MATH7343 Final Project

Group B: Yiyan Meng, Fangyi Tian, Hesen Huang, Yapeng Guo, Boyang Zhao

2024-04-11

1. Load the Data

```
# Library the packages
library(dplyr)
library(ggplot2)
library(caret)
library(corrplot)
library(glmnet)
library(tidyr)
library(tidyverse)
library(psych)
library(colorspace)
library(stargazer)
library(PerformanceAnalytics)
library(MASS)
library(car)
library(nnet)
```

```
# Import the data
carbon <- read.csv("Carbon Emission.csv")
```

2. EDA - Exploratory Data Analysis

2.1 Data Overview

First, we can check the dimensions of the dataset.

```
# Get the dimensions of the dataset
dim(carbon)
```

```
## [1] 10000    20
```

```
# View the first few rows of the data
head(carbon)
```

	Body.Type <chr>	Sex <chr>	Diet <chr>	How.Often.Shower <chr>	Heating.Energy.Source <chr>	Transport <chr>	V <chr>
1	overweight	female	pescatarian	daily	coal	public	

	Body.Type <chr>	Sex <chr>	Diet <chr>	How.Often.Shower <chr>	Heating.Energy.Source <chr>	Transport <chr>	V <chr>
2	obese	female	vegetarian	less frequently	natural gas	walk/bicycle	
3	overweight	male	omnivore	more frequently	wood	private	p
4	overweight	male	omnivore	twice a day	wood	walk/bicycle	
5	obese	female	vegetarian	daily	coal	private	c
6	overweight	male	vegetarian	less frequently	wood	public	

6 rows | 1-8 of 21 columns

The output offers a preliminary view of the scale and scope of our data. We can know the dataset has 10000 rows (samples) and 20 columns (features) and get a glimpse of the data structure and content.

Then we can get some basic information of this dataset for future analysis.

```
# Inspect the data
summary(carbon)
```

```
##   Body.Type             Sex                Diet           How.Often.Shower
##  Length:10000       Length:10000       Length:10000       Length:10000
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  Heating.Energy.Source  Transport         Vehicle.Type       Social.Activity
##  Length:10000          Length:10000       Length:10000       Length:10000
##  Class :character      Class :character   Class :character   Class :character
##  Mode  :character      Mode  :character   Mode  :character   Mode  :character
##
##
##
##  Monthly.Grocery.Bill Frequency.of.Traveling.by.Air Vehicle.Monthly.Distance.Km
##  Min.   : 50.0        Length:10000                  Min.   :   0
##  1st Qu.:111.0        Class :character              1st Qu.:  69
##  Median :173.0        Mode  :character              Median : 823
##  Mean   :173.9                                      Mean   :2031
##  3rd Qu.:237.0                                      3rd Qu.:2517
##  Max.   :299.0                                      Max.   :9999
##  Waste.Bag.Size     Waste.Bag.Weekly.Count How.Long.TV.PC.Daily.Hour
##  Length:10000       Min.   :1.000          Min.   : 0.00
##  Class :character   1st Qu.:2.000          1st Qu.: 6.00
##  Mode  :character   Median :4.000          Median :12.00
##                     Mean   :4.025          Mean   :12.14
##                     3rd Qu.:6.000          3rd Qu.:18.00
##                     Max.   :7.000          Max.   :24.00
##  How.Many.New.Clothes.Monthly How.Long.Internet.Daily.Hour Energy.efficiency
##  Min.   : 0.00                Min.   : 0.00                Length:10000
##  1st Qu.:13.00                1st Qu.: 6.00                Class :character
##  Median :25.00                Median :12.00                Mode  :character
##  Mean   :25.11                Mean   :11.89
##  3rd Qu.:38.00                3rd Qu.:18.00
##  Max.   :50.00                Max.   :24.00
##   Recycling         Cooking_With       CarbonEmission
##  Length:10000       Length:10000       Min.   : 306
##  Class :character   Class :character   1st Qu.:1538
##  Mode  :character   Mode  :character   Median :2080
##                                        Mean   :2269
##                                        3rd Qu.:2768
##                                        Max.   :8377
```

```
str(carbon)
```

```
## 'data.frame':    10000 obs. of  20 variables:
##  $ Body.Type                    : chr  "overweight" "obese" "overweight" "overweight"
...
##  $ Sex                          : chr  "female" "female" "male" "male" ...
##  $ Diet                         : chr  "pescatarian" "vegetarian" "omnivore" "omnivor
e" ...
##  $ How.Often.Shower             : chr  "daily" "less frequently" "more frequently" "t
wice a day" ...
##  $ Heating.Energy.Source        : chr  "coal" "natural gas" "wood" "wood" ...
##  $ Transport                    : chr  "public" "walk/bicycle" "private" "walk/bicycl
e" ...
##  $ Vehicle.Type                 : chr  "" "" "petrol" "" ...
##  $ Social.Activity              : chr  "often" "often" "never" "sometimes" ...
##  $ Monthly.Grocery.Bill         : int  230 114 138 157 266 144 56 59 200 135 ...
##  $ Frequency.of.Traveling.by.Air: chr  "frequently" "rarely" "never" "rarely" ...
##  $ Vehicle.Monthly.Distance.Km  : int  210 9 2472 74 8457 658 5363 54 1376 440 ...
##  $ Waste.Bag.Size               : chr  "large" "extra large" "small" "medium" ...
##  $ Waste.Bag.Weekly.Count       : int  4 3 1 3 1 1 4 3 3 1 ...
##  $ How.Long.TV.PC.Daily.Hour    : int  7 9 14 20 3 22 9 5 3 8 ...
##  $ How.Many.New.Clothes.Monthly : int  26 38 47 5 5 18 11 39 31 23 ...
##  $ How.Long.Internet.Daily.Hour : int  1 5 6 7 6 9 19 15 15 18 ...
##  $ Energy.efficiency            : chr  "No" "No" "Sometimes" "Sometimes" ...
##  $ Recycling                    : chr  "['Metal']" "['Metal']" "['Metal']" "['Paper',
'Plastic', 'Glass', 'Metal']" ...
##  $ Cooking_With                 : chr  "['Stove', 'Oven']" "['Stove', 'Microwave']" "
['Oven', 'Microwave']" "['Microwave', 'Grill', 'Airfryer']" ...
##  $ CarbonEmission               : int  2238 1892 2595 1074 4743 1647 1832 2322 2494 1
178 ...
```

This step helps us understand the variables, their data types, and get a general idea of the data distribution.

Explanation:

We can get some specific data in each feature to understand the structure and summary statistics of the dataset. There are 20 variables in our dataset. 13 variables are of character type, indicating categorical data. 7 variables are integers, indicating numerical data.

Here are our 20 features.

`Body Type`' : Body type.

`Sex` : Gender.

`Diet` : Diet.

`How.Often.Shower` : Frequency of showering.

` `Heating.Energy.Source` : Residential heating energy.

`Transport` : Transportation preference.

`Vehicle.Type` : Vehicle fuel type.

`Social.Activity` : Frequency of participating in social activities.

`Monthly.Grocery.Bill` : Monthly amount spent on groceries, in dollars.

`Frequency.of.Traveling.by.Air` : Frequency of using aircraft in the last month.

`Vehicle.Monthly.Distance.Km` : The kilometers traveled by vehicle in the last month.

`Waste.Bag.Size` : Size of the garbage bag.

`Waste.Bag.Weekly.Count` : The amount of garbage thrown away in the last week.

`How.Long.TV.PC.Daily.Hour` : Daily time spent in front of TV or PC.

`How.Many.New.Clothes.Monthly` : Number of clothes purchased monthly.

`How.Long.Internet.Daily.Hour` : Time spent on the Internet daily.

`Energy.efficiency` : Whether or not you care about purchasing energy efficient devices.

`Recycling` : The wastes it recycles.

`Cooking_With` : Devices used in cooking.

`CarbonEmission` : Dependent variable, total carbon emissions.

2.2 Rename the Variables

We renamed the variable names and are adhered to a consistent naming convention for better readability.

```
# Rename the variables
carbon <- rename(carbon,
                 body.type = "Body.Type", sex = "Sex", diet = "Diet",
                 shower.freq = "How.Often.Shower", heat.src = "Heating.Energy.Source",
                 transport = "Transport", veh.type = "Vehicle.Type",
                 social.act = "Social.Activity", groc.bill = "Monthly.Grocery.Bill",
                 airtvl.freq = "Frequency.of.Traveling.by.Air",
                 veh.distance = "Vehicle.Monthly.Distance.Km",
                 wastebag.size = "Waste.Bag.Size", wastebag.num = "Waste.Bag.Weekly.Coun
t",
                 tvpc.hour = "How.Long.TV.PC.Daily.Hour",
                 new.clothes = "How.Many.New.Clothes.Monthly",
                 internet.hour = "How.Long.Internet.Daily.Hour",
                 energy.eff = "Energy.efficiency", recycling = "Recycling",
                 cooking = "Cooking_With", carbon.emission = "CarbonEmission")
# Check the frequency and names of the character variables
for (i in 1:20) {
  if(class(carbon[,i]) == "character"){
    print(table(carbon[,i]))
  }
}
```

```
##
##      normal       obese  overweight underweight
##        2473        2500        2487        2540
##
## female   male
##   5007   4993
##
##    omnivore pescatarian       vegan  vegetarian
##        2492        2554        2497        2457
##
##           daily less frequently more frequently     twice a day
##            2546            2487            2451            2516
##
##        coal electricity natural gas        wood
##        2523        2552        2462        2463
##
##      private       public walk/bicycle
##         3279         3294         3427
##
##            diesel electric   hybrid      lpg   petrol
##     6721      622      671      642      697      647
##
##     never     often sometimes
##      3406      3319      3275
##
##      frequently           never          rarely very frequently
##            2524            2459            2477            2540
##
## extra large       large      medium       small
##        2500        2501        2474        2525
##
##        No Sometimes       Yes
##      3221      3463      3316
##
##                     ['Glass', 'Metal']                              ['Glass']
##                                    645                                    587
##                               ['Metal']            ['Paper', 'Glass', 'Metal']
##                                    625                                    647
##                     ['Paper', 'Glass']                      ['Paper', 'Metal']
##                                    616                                    589
## ['Paper', 'Plastic', 'Glass', 'Metal']          ['Paper', 'Plastic', 'Glass']
##                                    637                                    588
##          ['Paper', 'Plastic', 'Metal']                   ['Paper', 'Plastic']
##                                    648                                    633
##                               ['Paper']          ['Plastic', 'Glass', 'Metal']
##                                    619                                    626
##                   ['Plastic', 'Glass']                   ['Plastic', 'Metal']
##                                    633                                    630
##                              ['Plastic']                                     []
##                                    602                                    675
##
##                                ['Grill', 'Airfryer']
```

```
##                                                 593
##                  ['Microwave', 'Grill', 'Airfryer']
##                                                 623
##                                       ['Microwave']
##                                                 621
##                        ['Oven', 'Grill', 'Airfryer']
##                                                 625
##          ['Oven', 'Microwave', 'Grill', 'Airfryer']
##                                                 638
##                                ['Oven', 'Microwave']
##                                                 649
##                                             ['Oven']
##                                                 607
##                       ['Stove', 'Grill', 'Airfryer']
##                                                 628
##         ['Stove', 'Microwave', 'Grill', 'Airfryer']
##                                                 652
##                                ['Stove', 'Microwave']
##                                                 625
##              ['Stove', 'Oven', 'Grill', 'Airfryer']
##                                                 596
## ['Stove', 'Oven', 'Microwave', 'Grill', 'Airfryer']
##                                                 637
##                       ['Stove', 'Oven', 'Microwave']
##                                                 628
##                                    ['Stove', 'Oven']
##                                                 670
##                                          ['Stove']
##                                                 605
##                                                  []
##                                                 603
```

2.3 Deal with NA

```
# Check the NAs
colSums(is.na(carbon))
```

```
##       body.type             sex            diet     shower.freq        heat.src
##               0               0               0               0               0
##       transport        veh.type      social.act       groc.bill     airtvl.freq
##               0               0               0               0               0
##    veh.distance   wastebag.size    wastebag.num       tvpc.hour     new.clothes
##               0               0               0               0               0
##   internet.hour      energy.eff       recycling         cooking carbon.emission
##               0               0               0               0               0
```

Although it shows there is no explicit NA in our dataset, we can notice that the column `veh.type` has some blank values. This variable means vehicle fuel type. So it is a easy job for us to know that these blank values are not real NAs. This is because if the person does not drive, they do not need to use vehicle fuel. So we can replace these values instead of deleting them.

```
# Replace the values
carbon$veh.type[carbon$veh.type == ""] <- "none"
```

2.4 Recode the Data

5 categorical variables `body.type` , `social.act` , `airtvl.freq` , `wastebag.size` and `energy.eff` have meaningful orders, so we decided to convert them to numeric.

The order of `body.type` is underweight < normal < overweight < obese.

The order of `social.act` is never < sometimes < often.

The order of `airtvl.freq` is never < rarely < frequently < very frequently.

The order of `wastebag.size` is small < medium < large < extra large.

The order of `energy.eff` is No < Sometimes < Yes.

```
# Convert character variables to numeric
carbon$body.type <- as.numeric(factor(carbon$body.type,
                    levels = c("underweight", "normal", "overweight", "obese")))
carbon$social.act <- as.numeric(factor(carbon$social.act,
                    levels = c("never", "sometimes", "often")))
carbon$airtvl.freq <- as.numeric(factor(carbon$airtvl.freq,
                    levels = c("never", "rarely", "frequently", "very frequently")))
carbon$wastebag.size <- as.numeric(factor(carbon$wastebag.size,
                    levels = c("small", "medium", "large", "extra large")))
carbon$energy.eff <- as.numeric(factor(carbon$energy.eff,
                    levels = c("No", "Sometimes", "Yes")))
```

And then we convert other character variables to factors.

```
char_vars <- c("sex", "diet", "shower.freq", "heat.src",
               "transport", "veh.type", "recycling", "cooking")
carbon[char_vars] <- lapply(carbon[char_vars], as.factor)
```

Now we can check the structure of the dataset again.

```
str(carbon)
```

```
## 'data.frame':    10000 obs. of  20 variables:
##  $ body.type      : num  3 4 3 3 4 3 1 1 3 1 ...
##  $ sex            : Factor w/ 2 levels "female","male": 1 1 2 2 1 2 1 1 2 1 ...
##  $ diet           : Factor w/ 4 levels "omnivore","pescatarian",..: 2 4 1 1 4 4 3 3 1
2 ...
##  $ shower.freq    : Factor w/ 4 levels "daily","less frequently",..: 1 2 3 4 1 2 2 3
1 1 ...
##  $ heat.src       : Factor w/ 4 levels "coal","electricity",..: 1 3 4 4 1 4 4 1 4 4
...
##  $ transport      : Factor w/ 3 levels "private","public",..: 2 3 1 3 1 2 1 3 2 2 ...
##  $ veh.type       : Factor w/ 6 levels "diesel","electric",..: 5 5 6 5 1 5 3 5 5 5
...
##  $ social.act     : num  3 3 1 2 3 2 1 2 1 3 ...
##  $ groc.bill      : int  230 114 138 157 266 144 56 59 200 135 ...
##  $ airtvl.freq    : num  3 2 1 2 4 3 2 4 3 2 ...
##  $ veh.distance   : int  210 9 2472 74 8457 658 5363 54 1376 440 ...
##  $ wastebag.size  : num  3 4 1 2 3 3 2 4 2 4 ...
##  $ wastebag.num   : int  4 3 1 3 1 1 4 3 3 1 ...
##  $ tvpc.hour      : int  7 9 14 20 3 22 9 5 3 8 ...
##  $ new.clothes    : int  26 38 47 5 5 18 11 39 31 23 ...
##  $ internet.hour  : int  1 5 6 7 6 9 19 15 15 18 ...
##  $ energy.eff     : num  1 1 2 2 3 2 2 1 3 2 ...
##  $ recycling      : Factor w/ 16 levels "['Glass', 'Metal']",..: 3 3 3 7 11 4 16 8 2
2 ...
##  $ cooking        : Factor w/ 16 levels "['Grill', 'Airfryer']",..: 14 10 6 2 7 13 1
10 2 2 ...
##  $ carbon.emission: int  2238 1892 2595 1074 4743 1647 1832 2322 2494 1178 ...
```

```
summary(carbon)
```

```
##    body.type         sex                diet               shower.freq
##  Min.   :1.000   female:5007   omnivore   :2492   daily          :2546
##  1st Qu.:1.000   male  :4993   pescatarian:2554   less frequently:2487
##  Median :2.000                 vegan      :2497   more frequently:2451
##  Mean   :2.495                 vegetarian :2457   twice a day    :2516
##  3rd Qu.:3.250
##  Max.   :4.000
##
##        heat.src           transport        veh.type      social.act
##  coal       :2523   private     :3279   diesel  : 622   Min.   :1.000
##  electricity:2552   public      :3294   electric: 671   1st Qu.:1.000
##  natural gas:2462   walk/bicycle:3427   hybrid  : 642   Median :2.000
##  wood       :2463                       lpg     : 697   Mean   :1.991
##                                         none    :6721   3rd Qu.:3.000
##                                         petrol  : 647   Max.   :3.000
##
##    groc.bill      airtvl.freq     veh.distance  wastebag.size    wastebag.num
##  Min.   : 50.0   Min.   :1.000   Min.   :   0   Min.   :1.000   Min.   :1.000
##  1st Qu.:111.0   1st Qu.:2.000   1st Qu.:  69   1st Qu.:1.000   1st Qu.:2.000
##  Median :173.0   Median :3.000   Median : 823   Median :3.000   Median :4.000
##  Mean   :173.9   Mean   :2.514   Mean   :2031   Mean   :2.498   Mean   :4.025
##  3rd Qu.:237.0   3rd Qu.:4.000   3rd Qu.:2517   3rd Qu.:3.250   3rd Qu.:6.000
##  Max.   :299.0   Max.   :4.000   Max.   :9999   Max.   :4.000   Max.   :7.000
##
##    tvpc.hour      new.clothes    internet.hour     energy.eff
##  Min.   : 0.00   Min.   : 0.00   Min.   : 0.00   Min.   :1.00
##  1st Qu.: 6.00   1st Qu.:13.00   1st Qu.: 6.00   1st Qu.:1.00
##  Median :12.00   Median :25.00   Median :12.00   Median :2.00
##  Mean   :12.14   Mean   :25.11   Mean   :11.89   Mean   :2.01
##  3rd Qu.:18.00   3rd Qu.:38.00   3rd Qu.:18.00   3rd Qu.:3.00
##  Max.   :24.00   Max.   :50.00   Max.   :24.00   Max.   :3.00
##
##                                  recycling
##  []                                   : 675
##  ['Paper', 'Plastic', 'Metal']        : 648
##  ['Paper', 'Glass', 'Metal']          : 647
##  ['Glass', 'Metal']                   : 645
##  ['Paper', 'Plastic', 'Glass', 'Metal']: 637
##  ['Paper', 'Plastic']                 : 633
##  (Other)                              :6115
##                                                  cooking     carbon.emission
##  ['Stove', 'Oven']                                   : 670   Min.   : 306
##  ['Stove', 'Microwave', 'Grill', 'Airfryer']         : 652   1st Qu.:1538
##  ['Oven', 'Microwave']                               : 649   Median :2080
##  ['Oven', 'Microwave', 'Grill', 'Airfryer']          : 638   Mean   :2269
##  ['Stove', 'Oven', 'Microwave', 'Grill', 'Airfryer']: 637   3rd Qu.:2768
##  ['Stove', 'Grill', 'Airfryer']                      : 628   Max.   :8377
##  (Other)                                             :6126
```

2.5 Analyze the Outliers

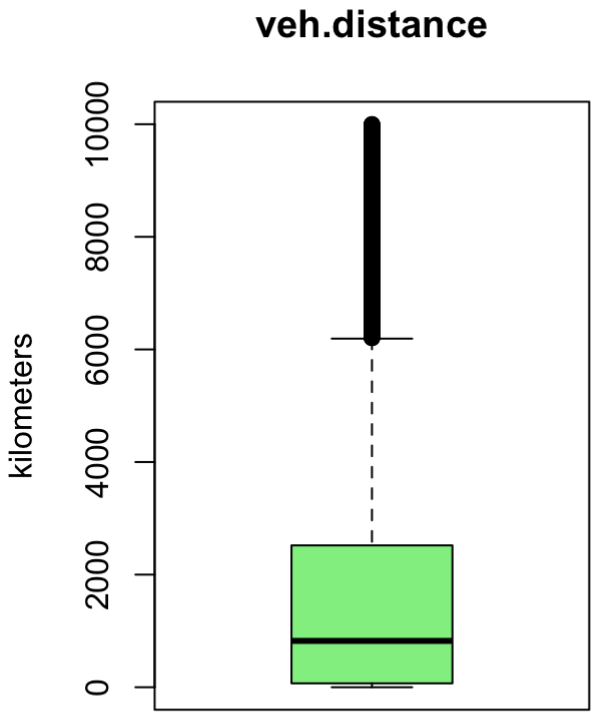
Now we tried to examine outliers for the original 7 numeric variables. Outliers are extreme values that deviate significantly from the majority of the data points and can potentially influence the analysis and modeling results. We first selected the numeric variables of interest. Then we used the interquartile range (IQR) method to identify them and counted the number of outliers.

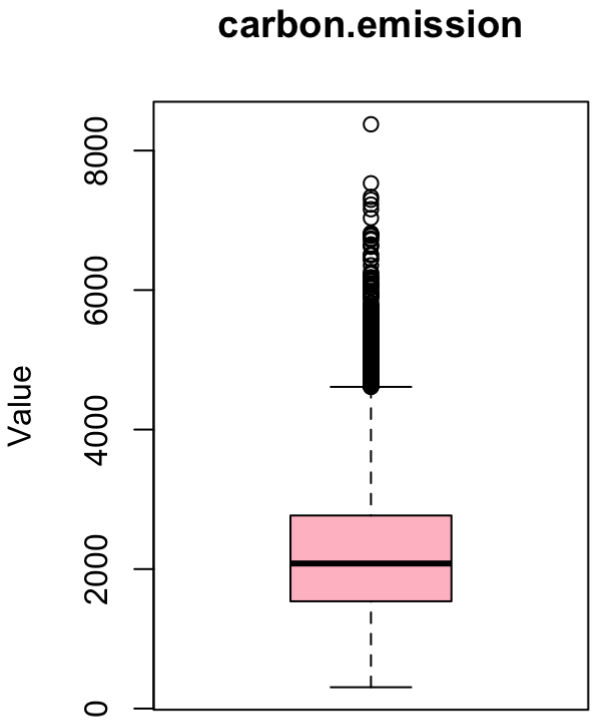
```
# Select the numeric variables
numeric_original <- c("groc.bill", "veh.distance", "wastebag.num","tvpc.hour",
                      "new.clothes", "internet.hour", "carbon.emission")

# Identify outliers using the IQR method
detect_outliers <- lapply(carbon[numeric_original], function(x) {
  Q1 <- quantile(x, 0.25)
  Q3 <- quantile(x, 0.75)
  IQR <- Q3 - Q1
  x < (Q1 - 1.5 * IQR) | x > (Q3 + 1.5 * IQR)
})

# Count the number of outliers for each variable
sapply(detect_outliers, sum)
```

```
##       groc.bill    veh.distance    wastebag.num       tvpc.hour     new.clothes
##               0            1294               0               0               0
##   internet.hour carbon.emission
##               0             346
```

We can see that only two variables `veh.distance` and `carbon.emission` have outliers. The former one has 1294 outliers and the latter one has 346 outliers. Then we visualize them.

```
# Create box plots for the variable containing outliers
par(mfrow = c(1, 2))
boxplot(carbon$veh.distance, col = "lightgreen",
        main = "veh.distance", ylab = "kilometers")
boxplot(carbon$carbon.emission, col = "pink",
        main = "carbon.emission", ylab = "Value")
```





So both groups are skewed to the right, with outliers concentrated on the side of larger values. We can take a look at the relevant data.

```
summary(carbon$veh.distance)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0      69     823    2031    2517    9999
```

```
summary(carbon$carbon.emission)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     306    1538    2080    2269    2768    8377
```

```
head(carbon[order(-carbon$carbon.emission), ], 10)
```

	body.type	sex	diet	shower.freq	heat.src	transport	veh.type	social.
	<dbl>	<fct>	<fct>	<fct>	<fct>	<fct>	<fct>	<c
9774	4	male	vegetarian	daily	coal	private	petrol	
4671	3	male	vegetarian	more frequently	coal	private	petrol	
1184	4	male	vegan	twice a day	wood	private	petrol	

	body.type	sex	diet	shower.freq	heat.src	transport	veh.type	social.
	<dbl>	<fct>	<fct>	<fct>	<fct>	<fct>	<fct>	<c
9395	4	male	omnivore	twice a day	natural gas	private	petrol	
1180	4	male	vegetarian	less frequently	coal	private	petrol	
221	3	male	vegan	less frequently	wood	private	petrol	
1065	4	male	omnivore	twice a day	coal	private	petrol	
734	4	male	pescatarian	less frequently	wood	private	lpg	
443	3	male	omnivore	more frequently	coal	private	petrol	
7417	3	male	pescatarian	less frequently	natural gas	private	petrol	

1-10 of 10 rows | 1-9 of 21 columns

```
head(carbon[order(-carbon$veh.distance), ], 10)
```

	body.type	sex	diet	shower.freq	heat.src	transport	veh.type	soc
	<dbl>	<fct>	<fct>	<fct>	<fct>	<fct>	<fct>	
3930	1	male	pescatarian	twice a day	natural gas	private	diesel	
2770	3	male	vegan	twice a day	electricity	private	hybrid	
8381	1	male	pescatarian	daily	wood	private	petrol	
6970	4	male	vegan	more frequently	wood	private	hybrid	
1861	1	male	omnivore	more frequently	coal	private	electric	
9698	3	female	omnivore	daily	coal	private	lpg	
5586	3	male	pescatarian	daily	wood	private	lpg	
6686	1	female	omnivore	twice a day	coal	private	hybrid	
2516	4	male	vegan	twice a day	wood	private	lpg	
2666	3	female	pescatarian	less frequently	electricity	private	electric	

1-10 of 10 rows | 1-9 of 21 columns

From the boxplots and data, the number of the outliers for `veh.distance` and `carbon.emission` are not small and they seem to be on the higher end but not necessarily errors. It is possible that a very high `veh.distance` could be associated with someone who travels extensively, which causes high energy consumption or frequent flying, then related to high `carbon.emission` . So we think these values are possible. Therefore, we decided not to remove them , because they might reveal interesting patterns.

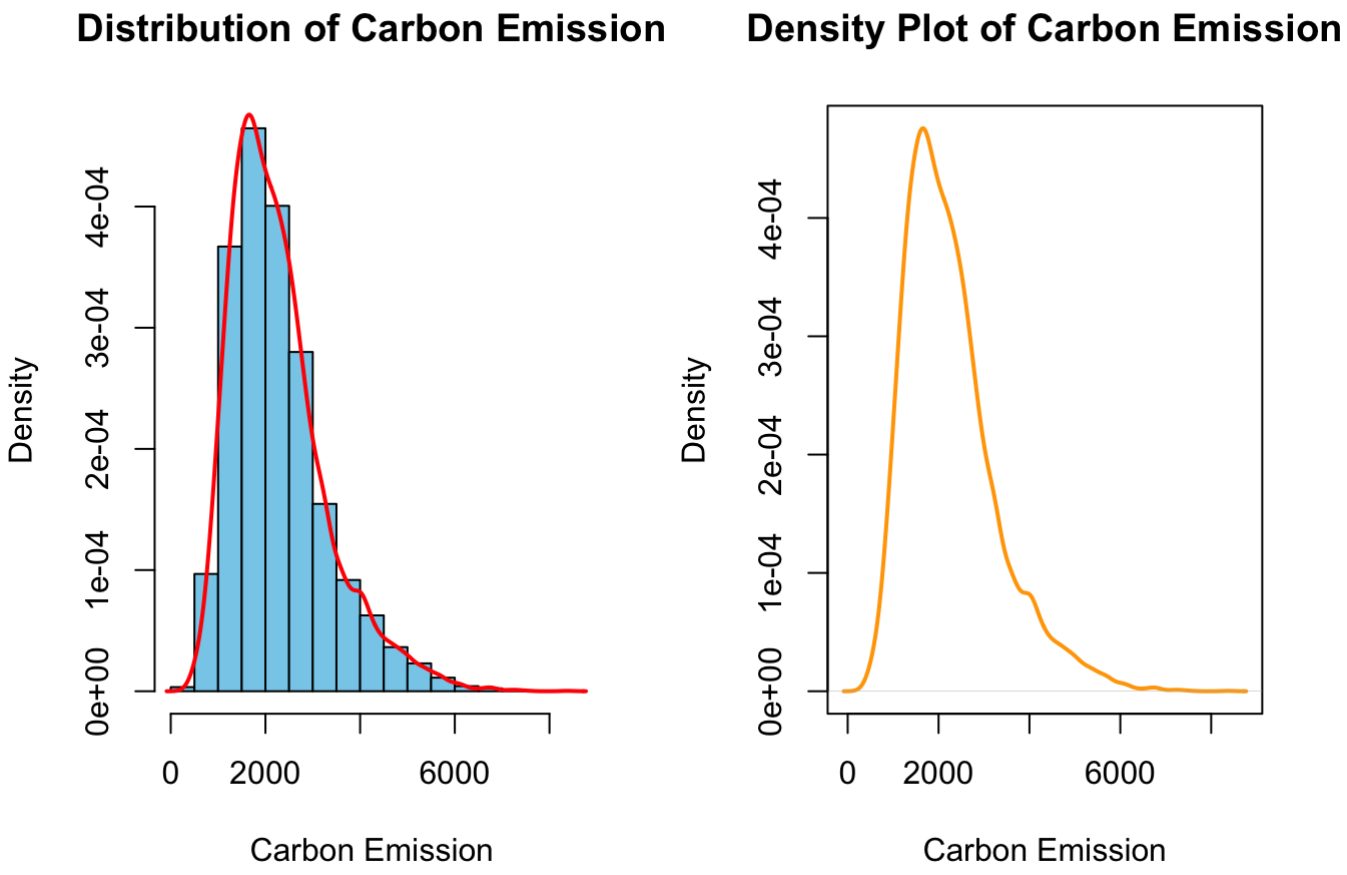
2.6 Data Visualization and Correlation

After getting some basic information and converting some data, now we want to use visualization to explore the basic relationship of carbon emission in different samples of different features.

2.6.1 Distribution of Dependent Variable

We can get some distribution of some features to help us to figure out whether these features follow some significant distribution or find the relationship between some features, and we can get the average carbon emission group by some features.

```
par(mfrow = c(1, 2))
# Histogram
hist(carbon$carbon.emission, col = "skyblue", probability = T,
     main = "Distribution of Carbon Emission", xlab = "Carbon Emission")
# Add density curve
lines(density(carbon$carbon.emission), col = "red", lwd = 2)
plot(density(carbon$carbon.emission), col = "orange", lwd = 2,
     main = "Density Plot of Carbon Emission", xlab = "Carbon Emission", ylab = "Densit
y")
```



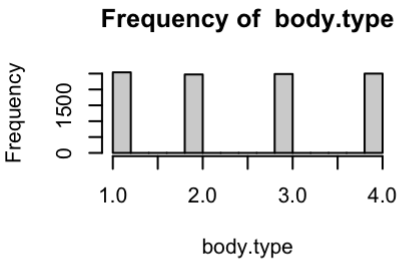
As can be seen from the figure, the distribution of carbon.emission is right-skewed, with most values concentrated in the lower range, but there are also some higher extreme values. We did not choose to delete these extremely high values.

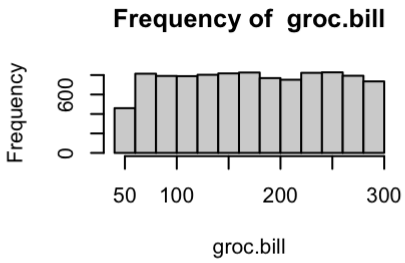
2.6.2 Independent Variables

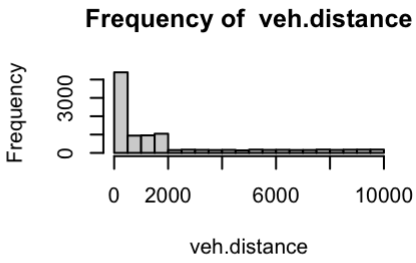
1. Numerical Variables

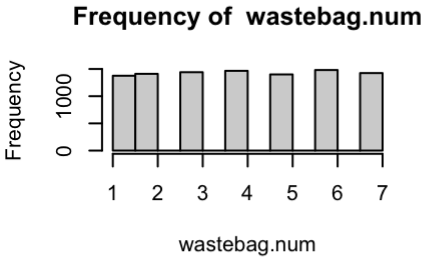
1. Histograms for numerical variables

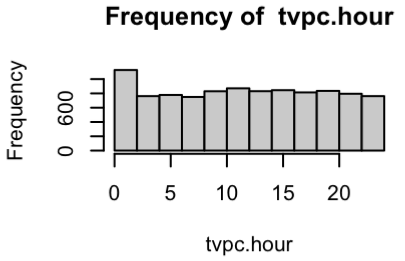
```
num_vars <- c("body.type", "groc.bill", "veh.distance", "wastebag.num",
              "tvpc.hour", "new.clothes", "internet.hour", "social.act",
              "airtvl.freq", "wastebag.size", "energy.eff")
# Histogram
par(mfrow = c(3, 3))
for (var in num_vars) {
    hist(carbon[[var]],
         xlab = var,
         main = paste("Frequency of ", var))
}
```

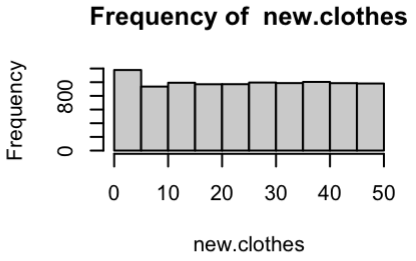

Frequency of body.type
Frequency
0 1500
1.0 2.0 3.0 4.0
body.type

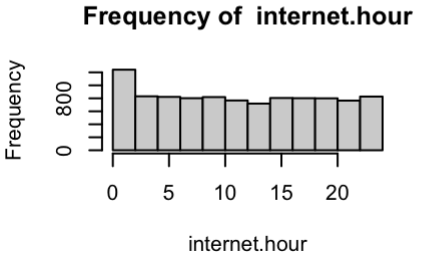

Frequency of groc.bill
Frequency
0 600
50 100 200 300
groc.bill

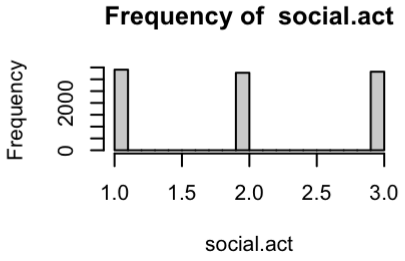

Frequency of veh.distance
Frequency
0 3000
0 2000 6000 10000
veh.distance

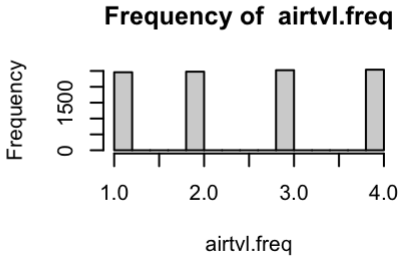

Frequency of wastebag.num
Frequency
0 1000
1 2 3 4 5 6 7
wastebag.num

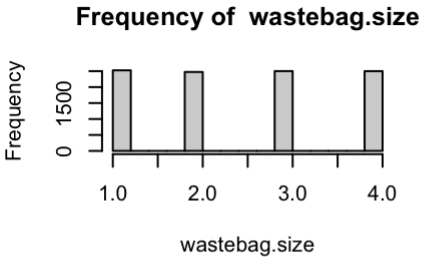

Frequency of tvpc.hour
Frequency
0 600
0 5 10 15 20
tvpc.hour

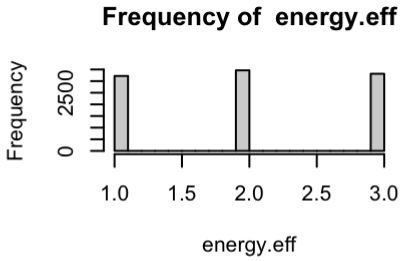

Frequency of new.clothes
Frequency
0 800
0 10 20 30 40 50
new.clothes


Frequency of internet.hour
Frequency
0 800
0 5 10 15 20
internet.hour


Frequency of social.act
Frequency
0 2000
1.0 1.5 2.0 2.5 3.0
social.act

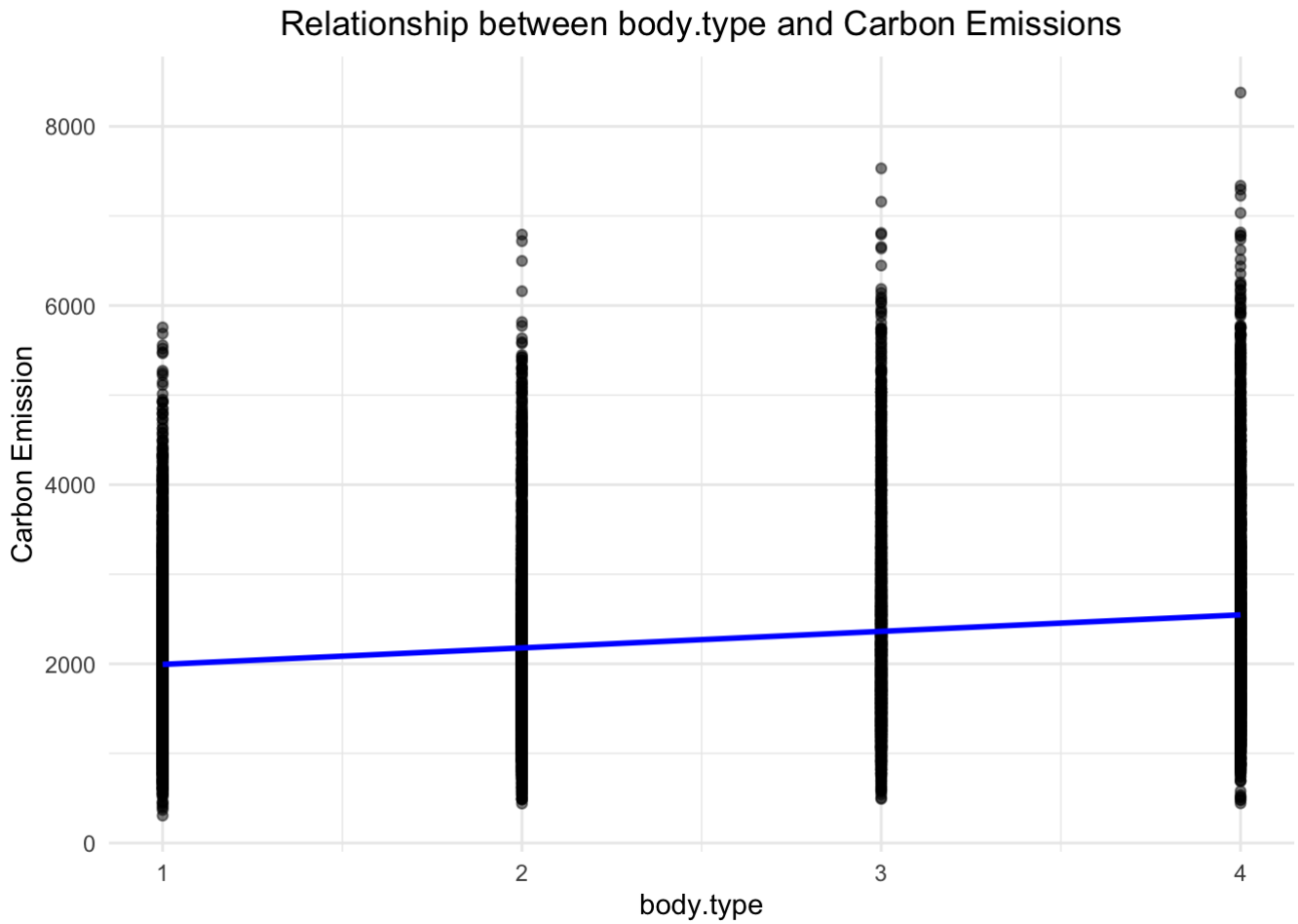

Frequency of airtvl.freq
Frequency
0 1500
1.0 2.0 3.0 4.0
airtvl.freq

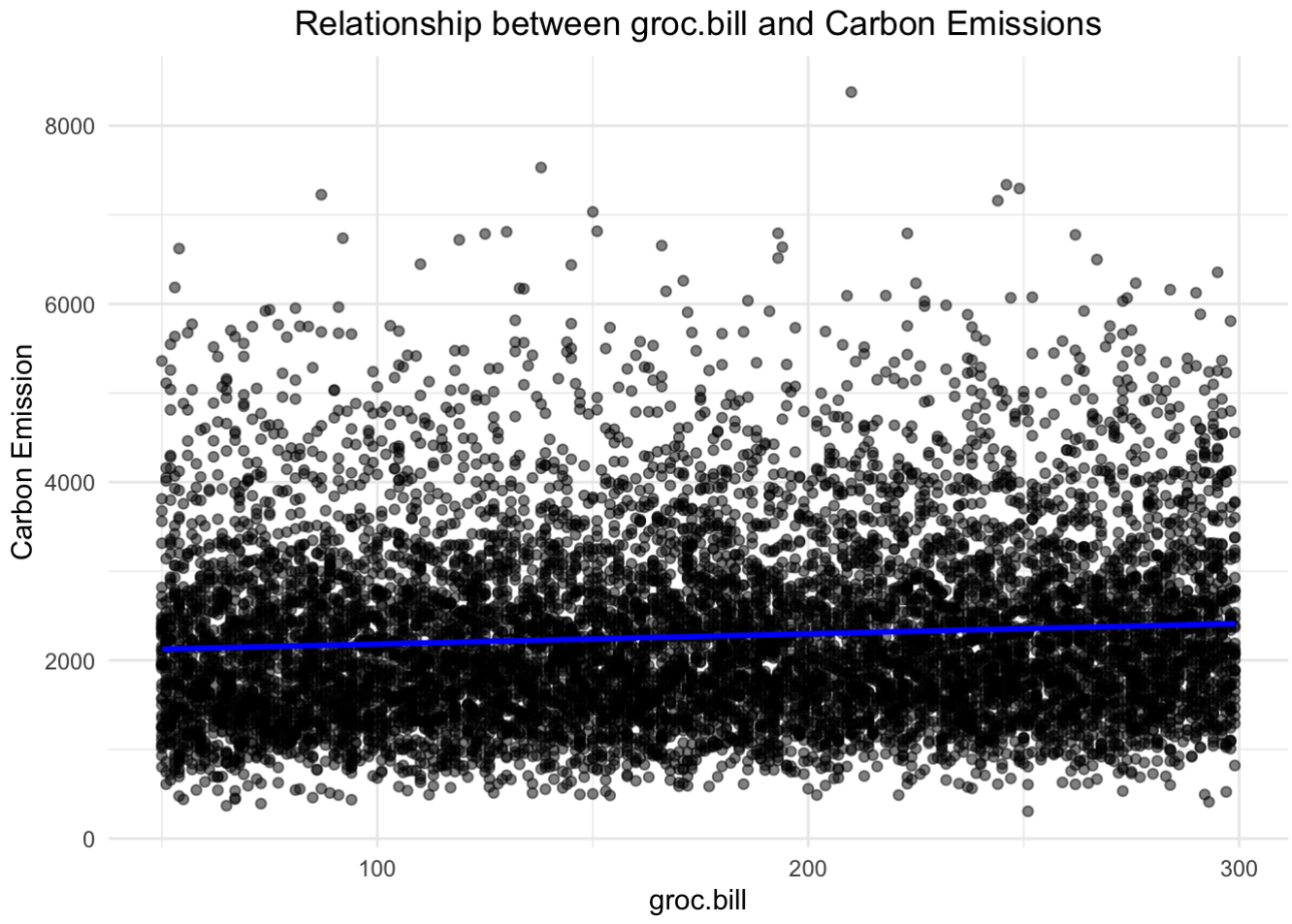

Frequency of wastebag.size
Frequency
0 1500
1.0 2.0 3.0 4.0
wastebag.size

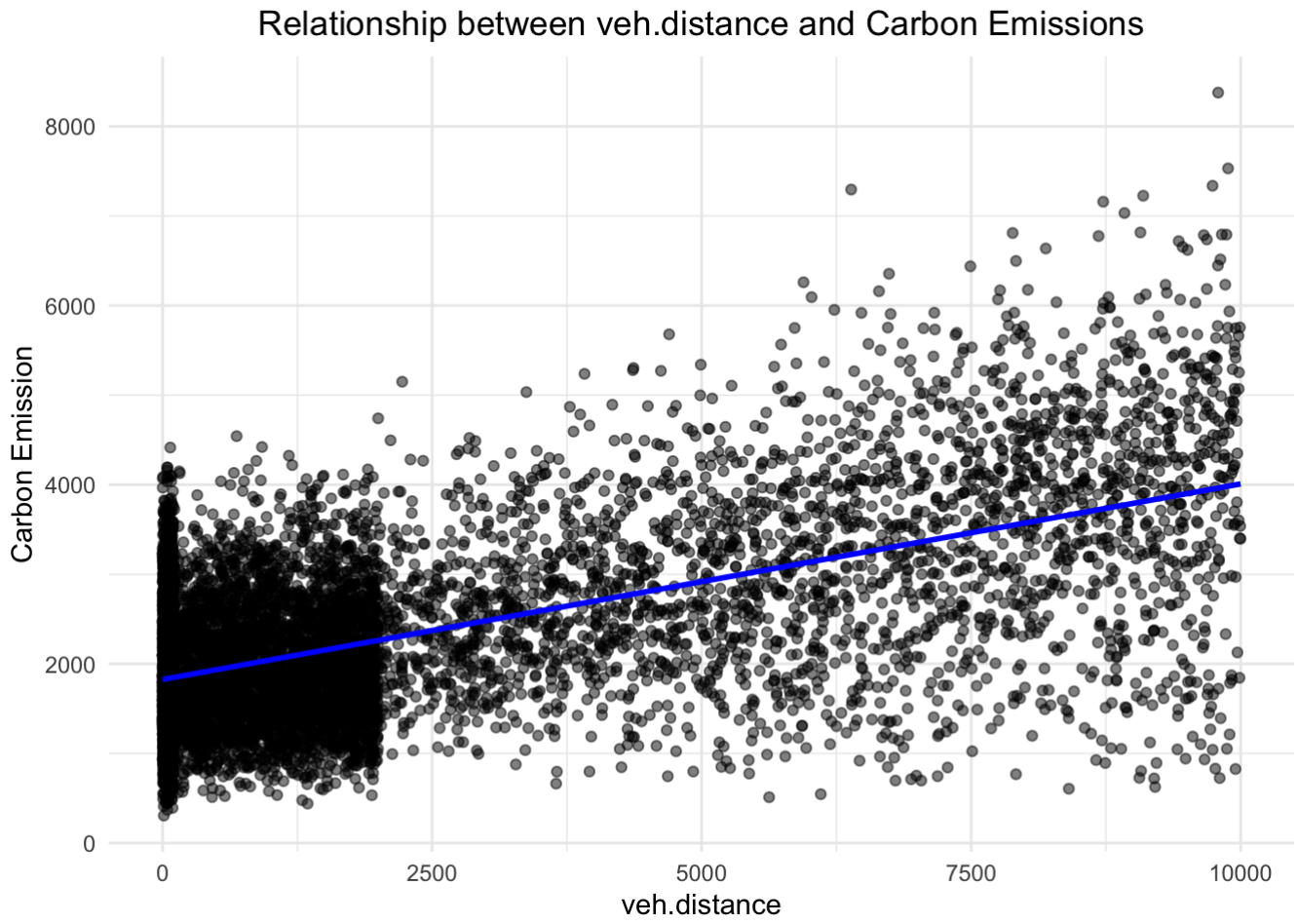

Frequency of energy.eff
Frequency
0 2500
1.0 1.5 2.0 2.5 3.0
energy.eff

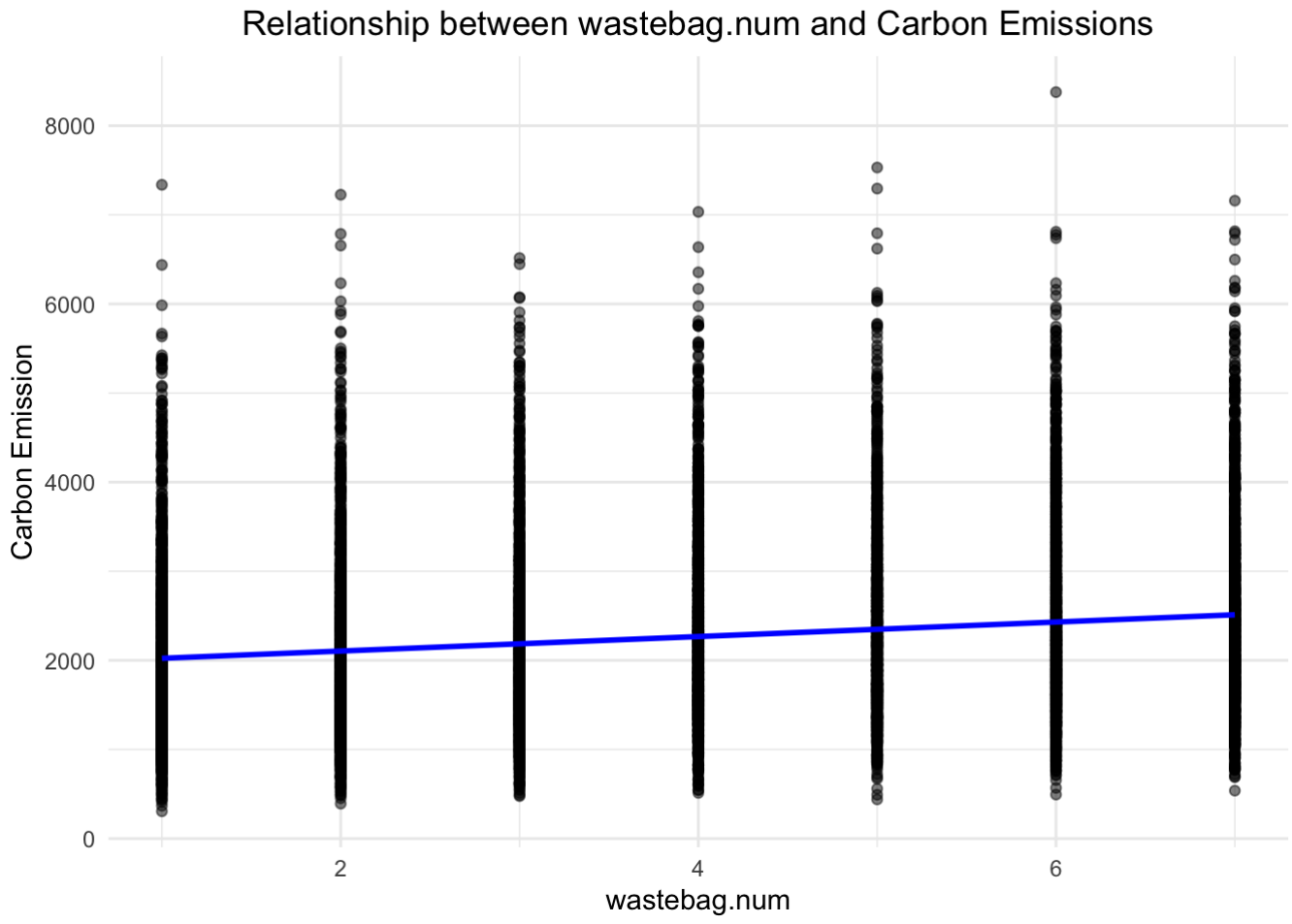
2. The relationship between numerical variables and carbon emissions

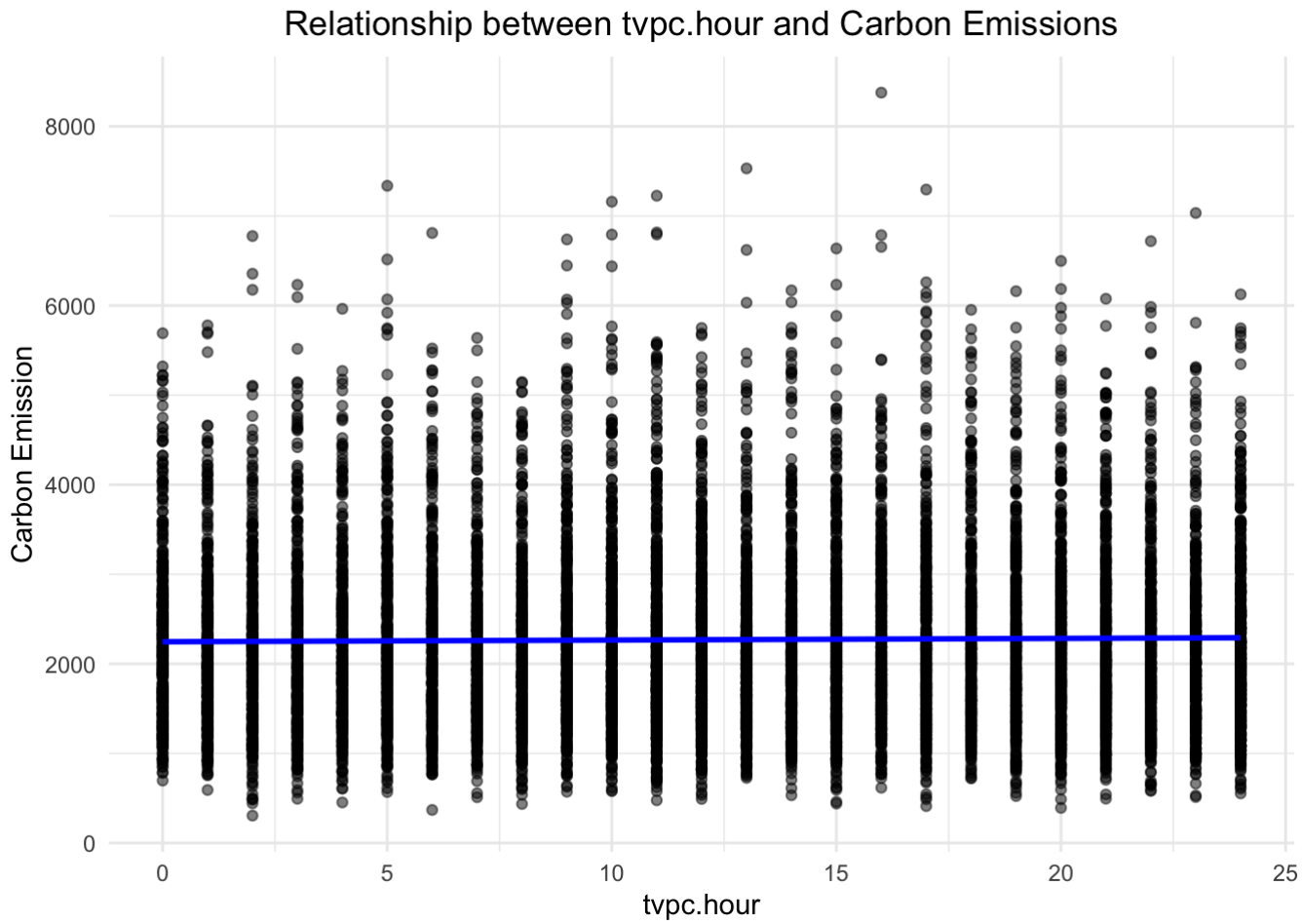
```
for (feature in num_vars) {
  print(ggplot(carbon, aes_string(x = feature, y = "carbon.emission")) +
          geom_point(alpha = 0.5) +
          geom_smooth(method = "lm", color = "blue", se = FALSE) +
          labs(title = paste("Relationship between", feature, "and Carbon Emissions"),
               x = feature, y = "Carbon Emission") +
          theme_minimal() +
          theme(plot.title = element_text(hjust = 0.5)) )
}
```

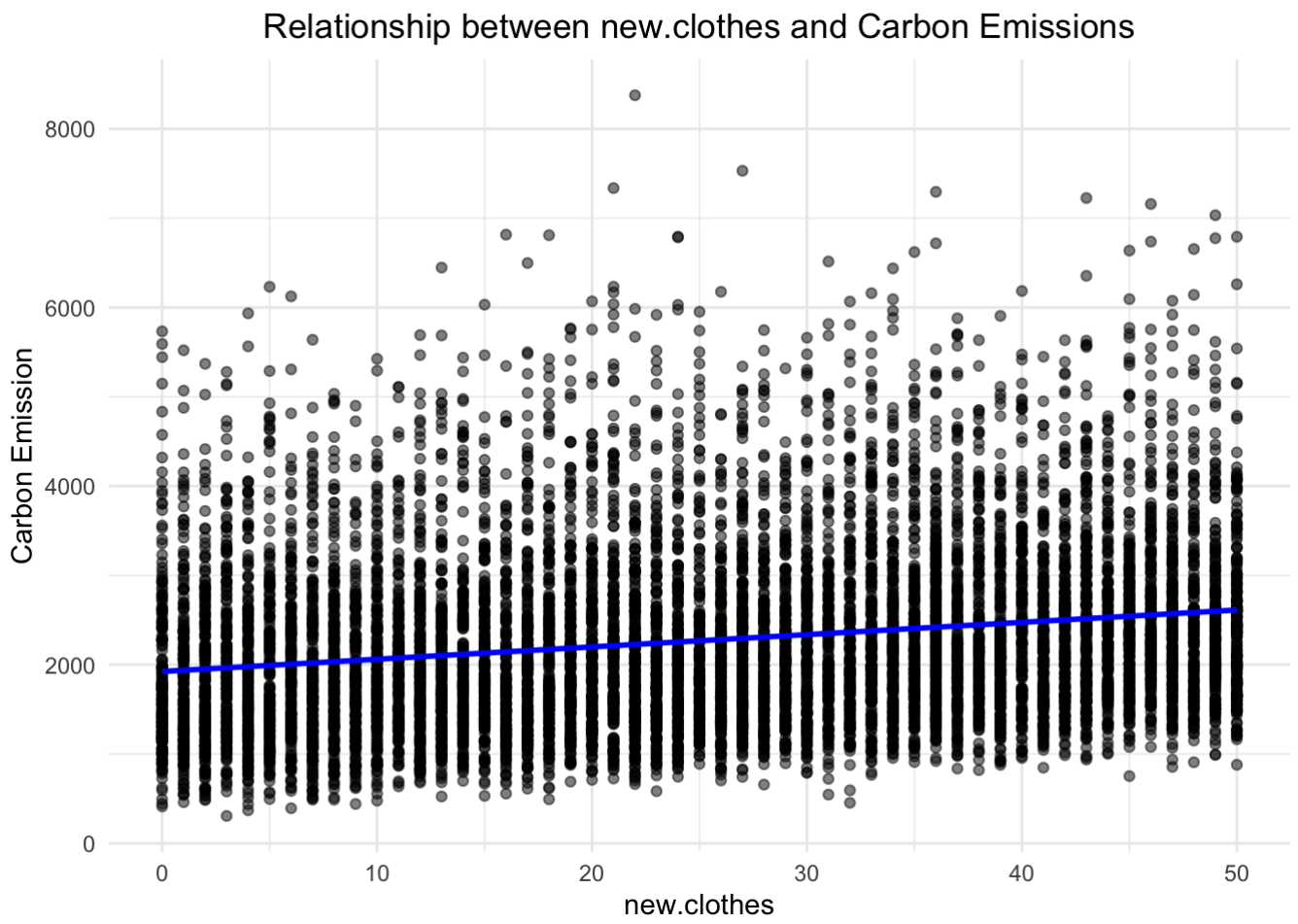

Relationship between body.type and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
1
2
3
4
body.type

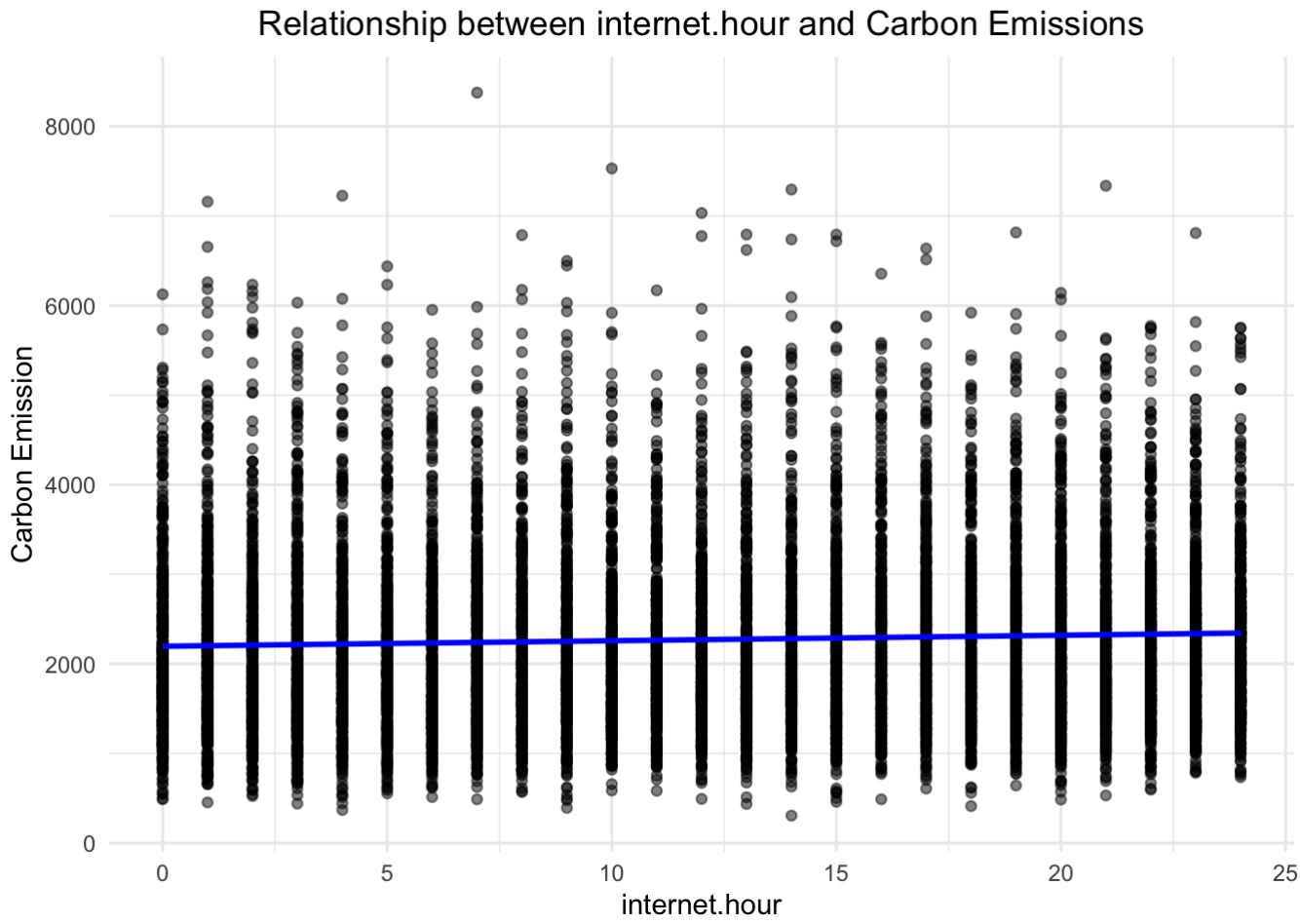

Relationship between groc.bill and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
100
200
300
groc.bill

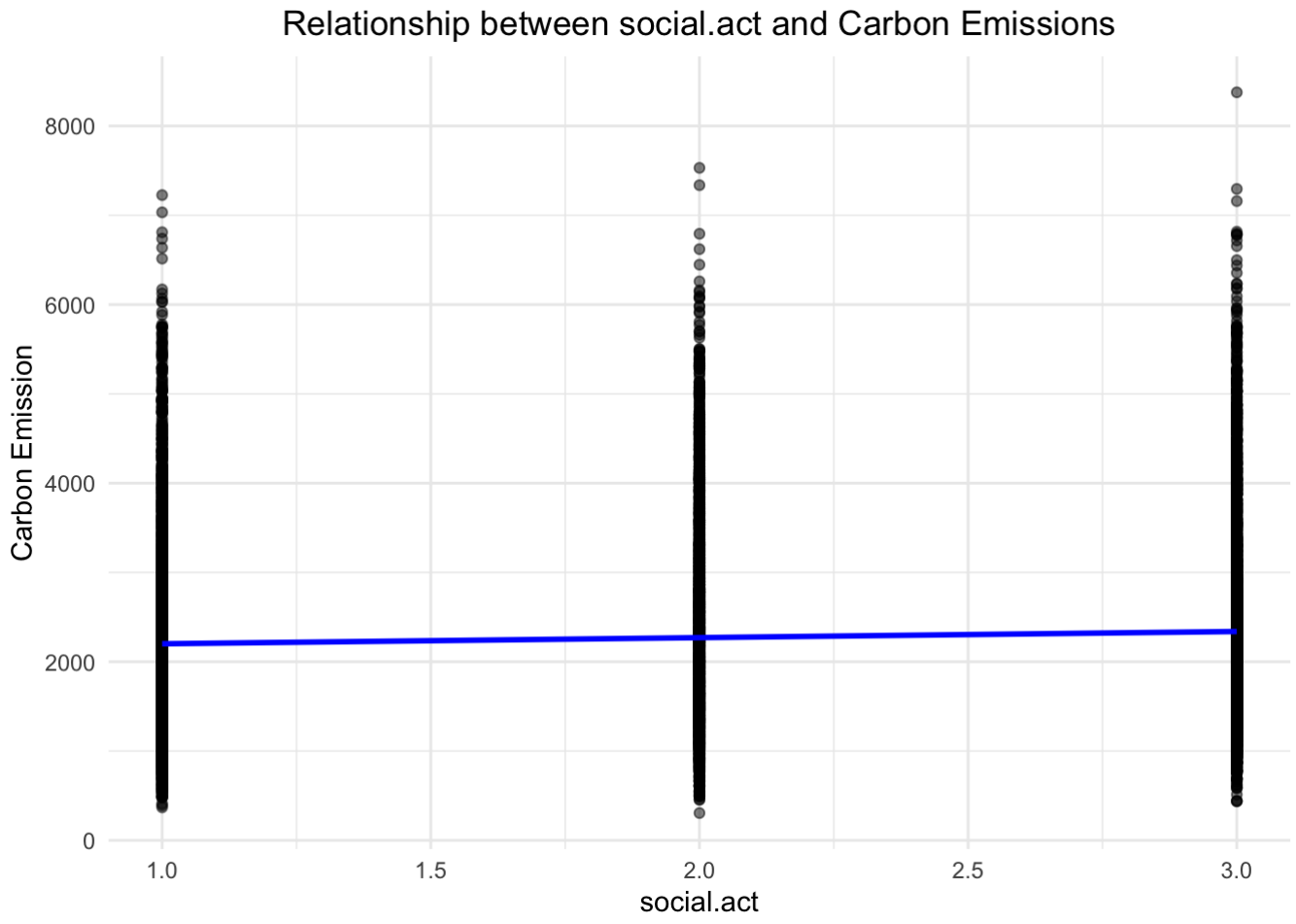

Relationship between veh.distance and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
0
2500
5000
7500
10000
veh.distance

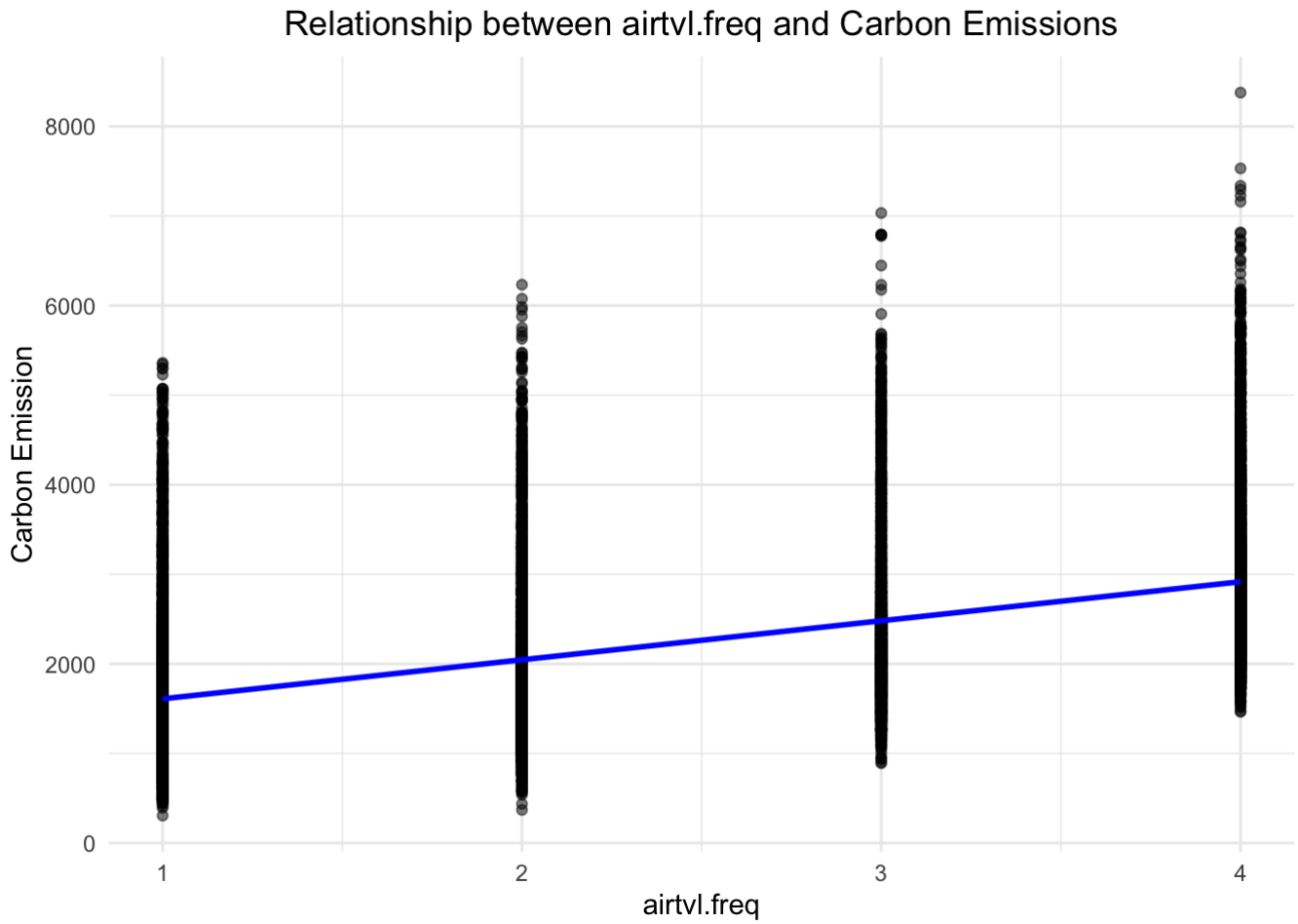

Relationship between wastebag.num and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
2
4
6
wastebag.num

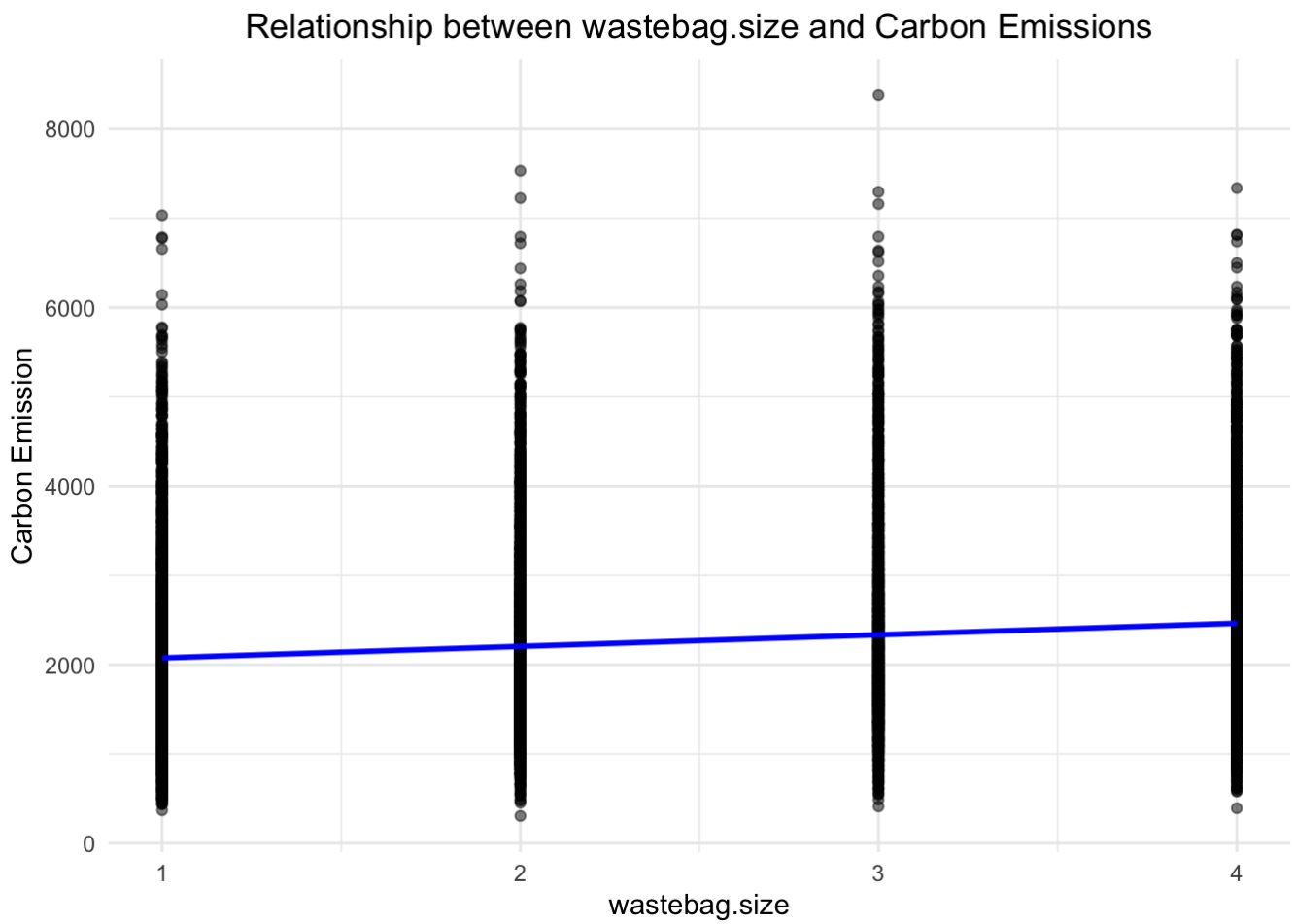

Relationship between tvpc.hour and Carbon Emissions
Carbon Emission
tvpc.hour
0
2000
4000
6000
8000
0
5
10
15
20
25

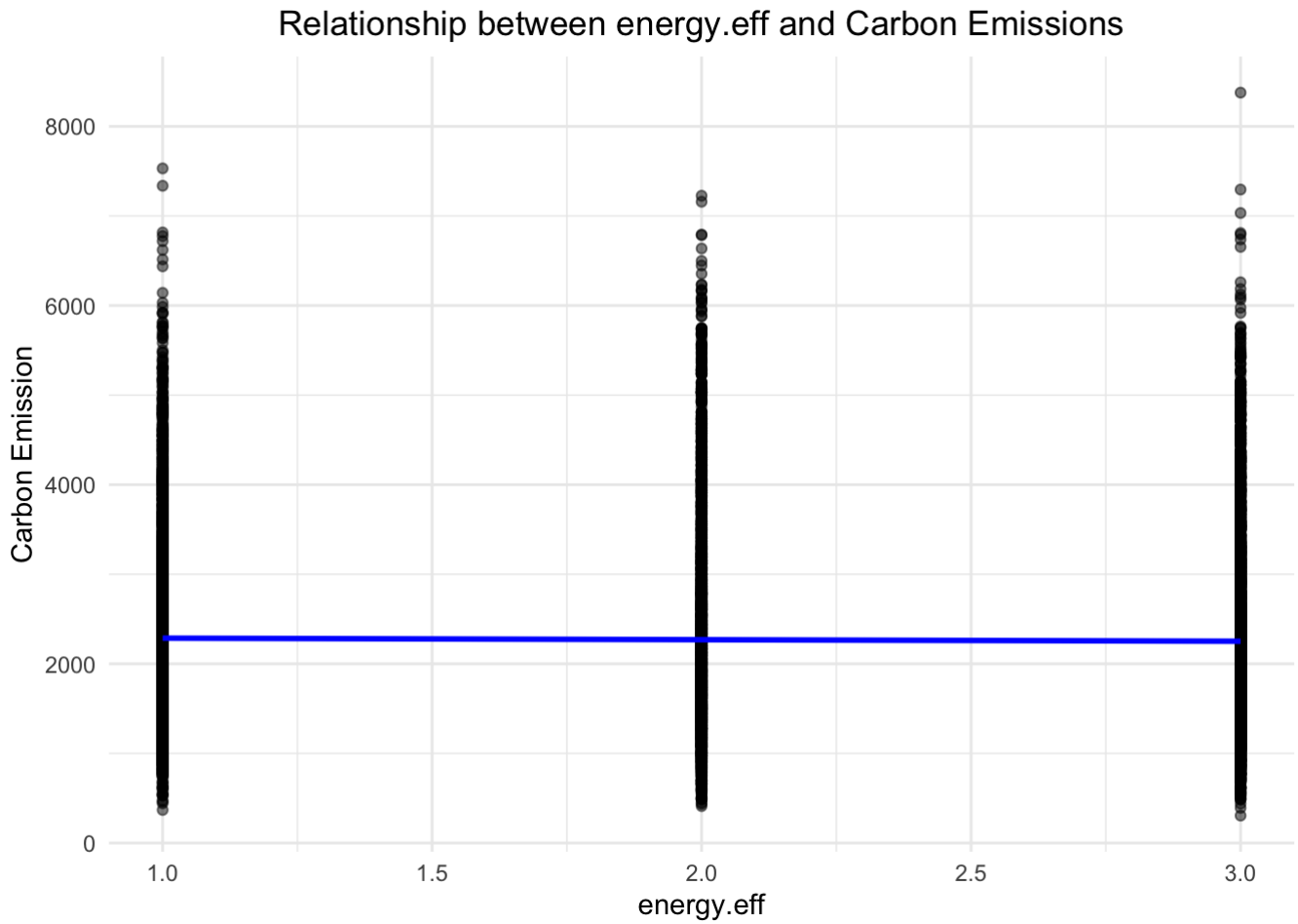

Relationship between new.clothes and Carbon Emissions
Carbon Emission
new.clothes
0
2000
4000
6000
8000
0
10
20
30
40
50


Relationship between internet.hour and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
0
5
10
15
20
25
internet.hour


Relationship between social.act and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
1.0
1.5
2.0
2.5
3.0
social.act

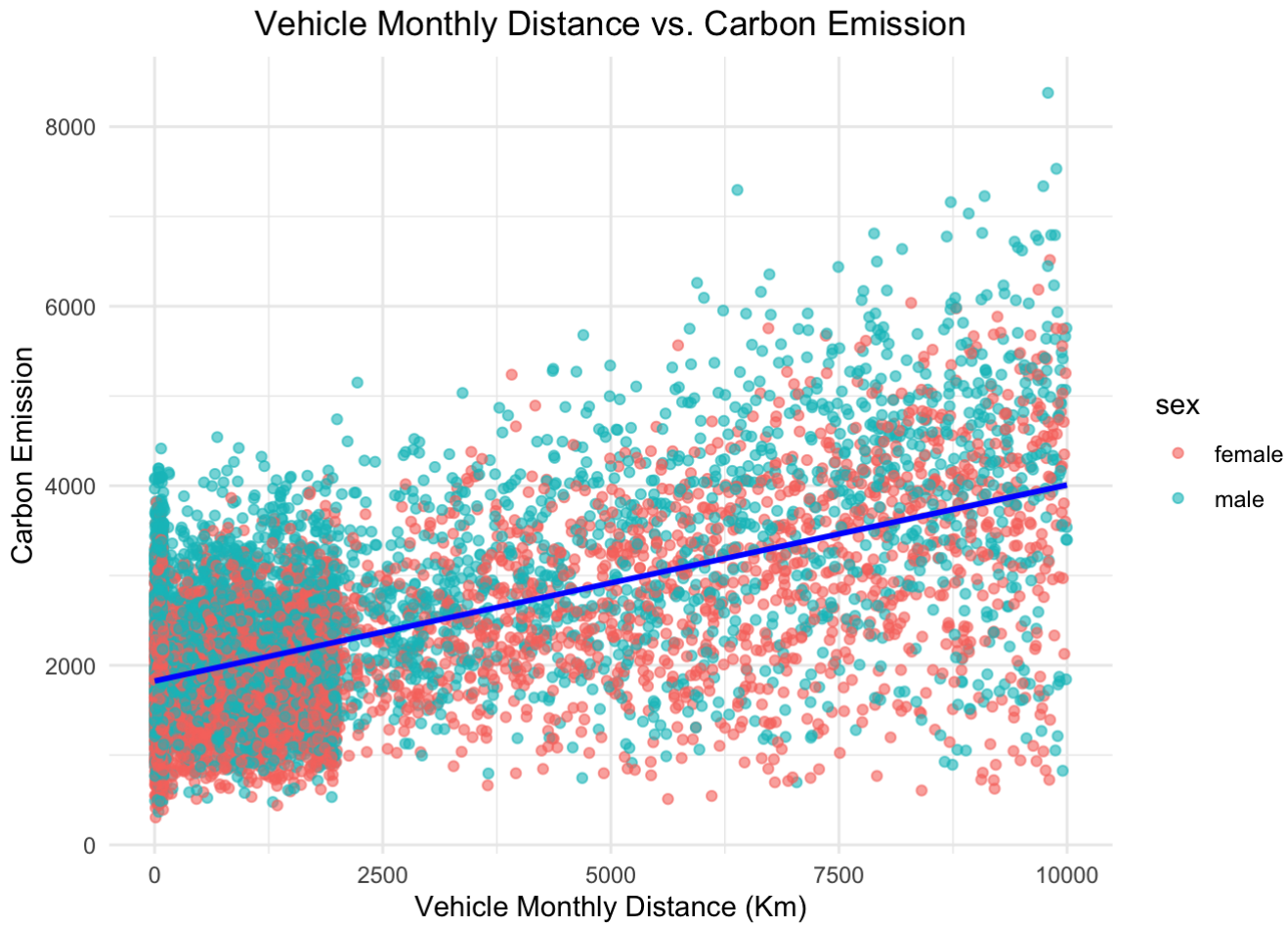

Relationship between airtvl.freq and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
1
2
3
4
airtvl.freq


Relationship between wastebag.size and Carbon Emissions
Carbon Emission
8000
6000
4000
2000
0
1
2
3
4
wastebag.size

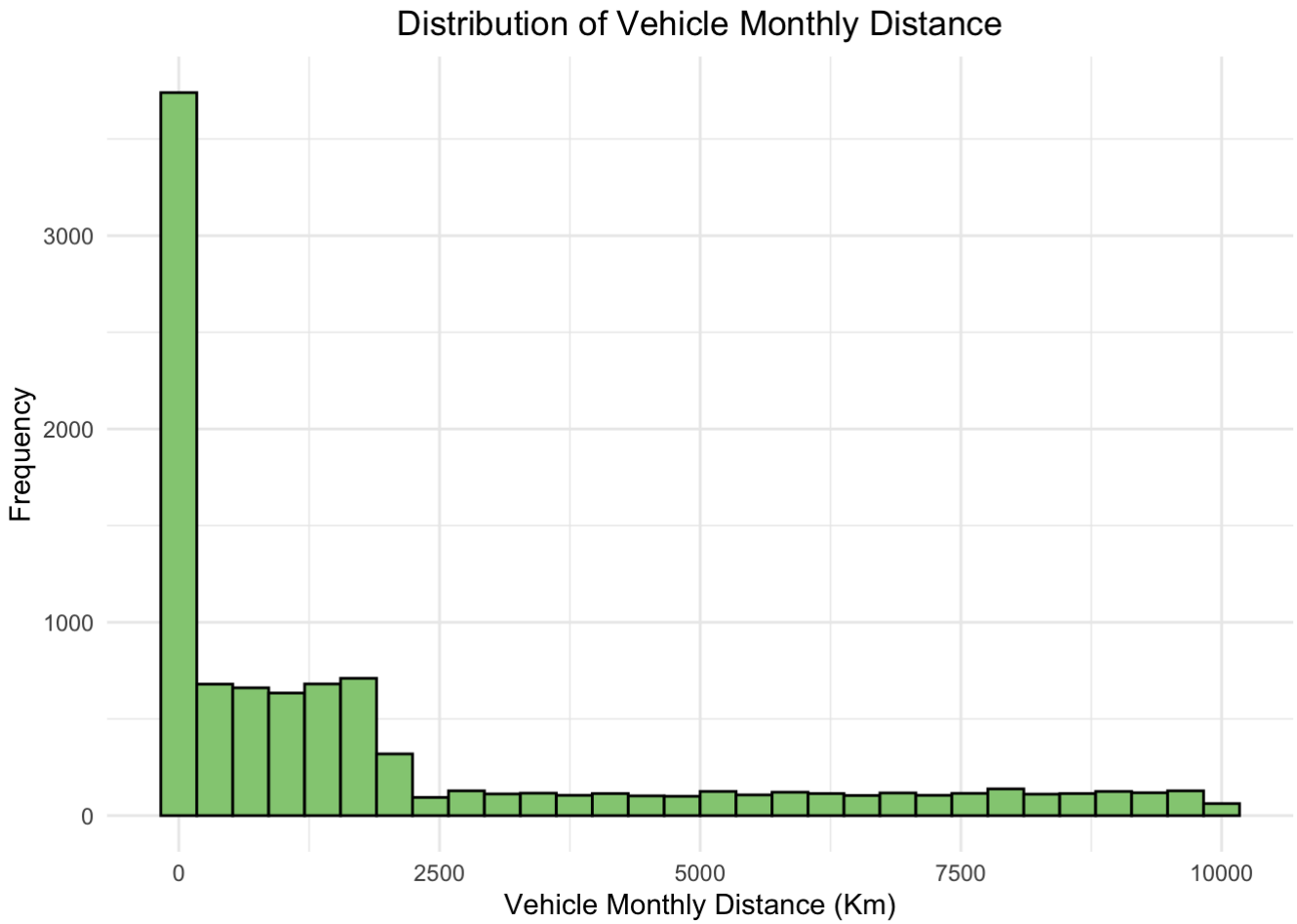


The numerical variable distribution patterns seem to be very uniform, and they all show some relationship with carbon emissions. However, the distribution of `veh.distance` is right-skewed and has an obvious direct relationship with `carbon.emission`.

```
# The relationship between veh.distance and carbon.emission
ggplot(carbon, aes(x = veh.distance, y = carbon.emission)) +
  geom_point(aes(color = sex), alpha = 0.6) +
  geom_smooth(method = "lm", color = "blue") +
  labs(title = "Vehicle Monthly Distance vs. Carbon Emission",
       x = "Vehicle Monthly Distance (Km)", y = "Carbon Emission") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```

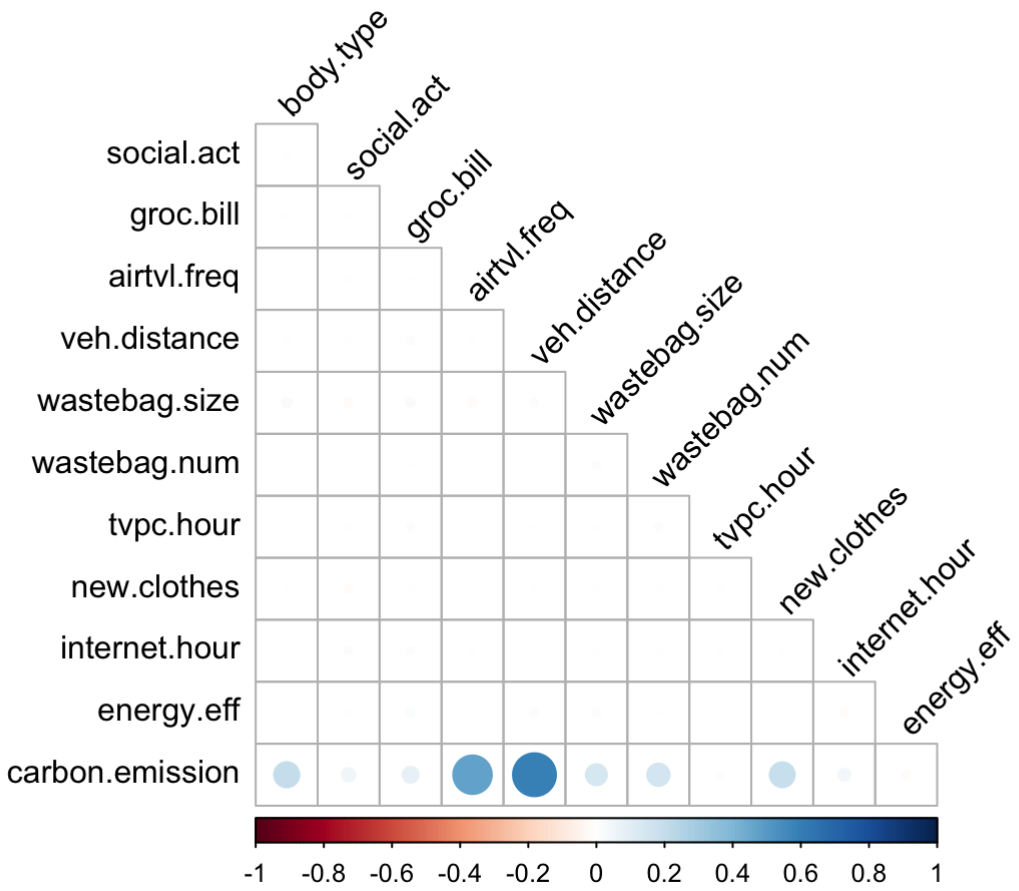



```
ggplot(carbon, aes(veh.distance)) +
  geom_histogram(fill = "#93cc82", color = "black") +
  labs(title = "Distribution of Vehicle Monthly Distance",
       x = "Vehicle Monthly Distance (Km)", y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```



3. Correlation

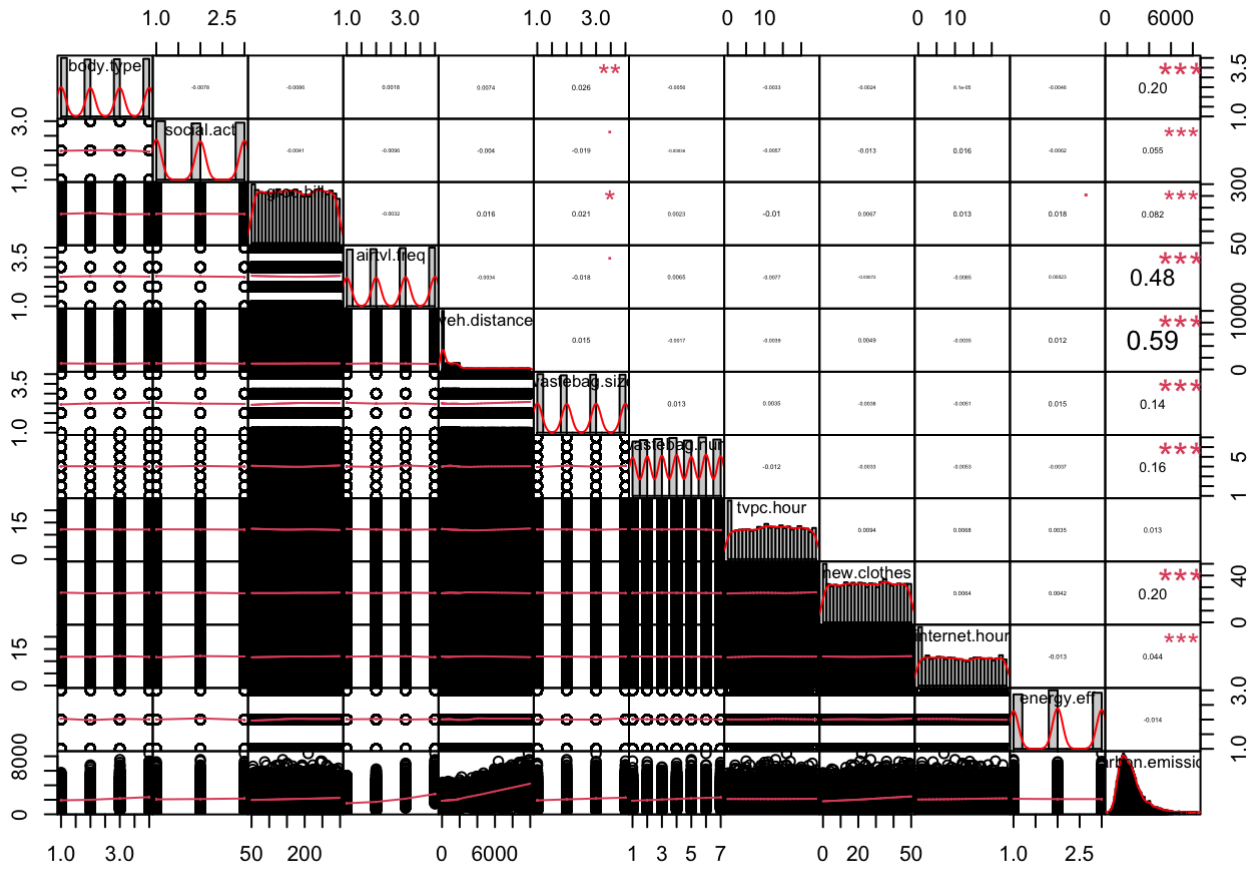
```
num_data <- carbon[, sapply(carbon, is.numeric)]
# Calculate the correlation matrix for the numeric variables
cor_matrix <- cor(num_data)
# Create a correlation plot
corrplot(cor_matrix, method = 'circle', type = 'lower',
         tl.col = "black", diag = F, tl.srt = 45)
```




```
# Extract the correlation coefficients
carbon_emission_cor <- cor_matrix["carbon.emission",] %>% sort(); carbon_emission_cor
```

```
##      energy.eff       tvpc.hour   internet.hour      social.act       groc.bill
##     -0.01371188      0.01298500      0.04387803      0.05537568      0.08158658
##   wastebag.size    wastebag.num     new.clothes       body.type     airtvl.freq
##      0.14239526      0.15919337      0.19888735      0.20316917      0.47848675
##    veh.distance carbon.emission
##      0.59417130      1.00000000
```

```
# Display the correlation matrix along with histograms and scatter plots
chart.Correlation(num_data, histogram = T)
```

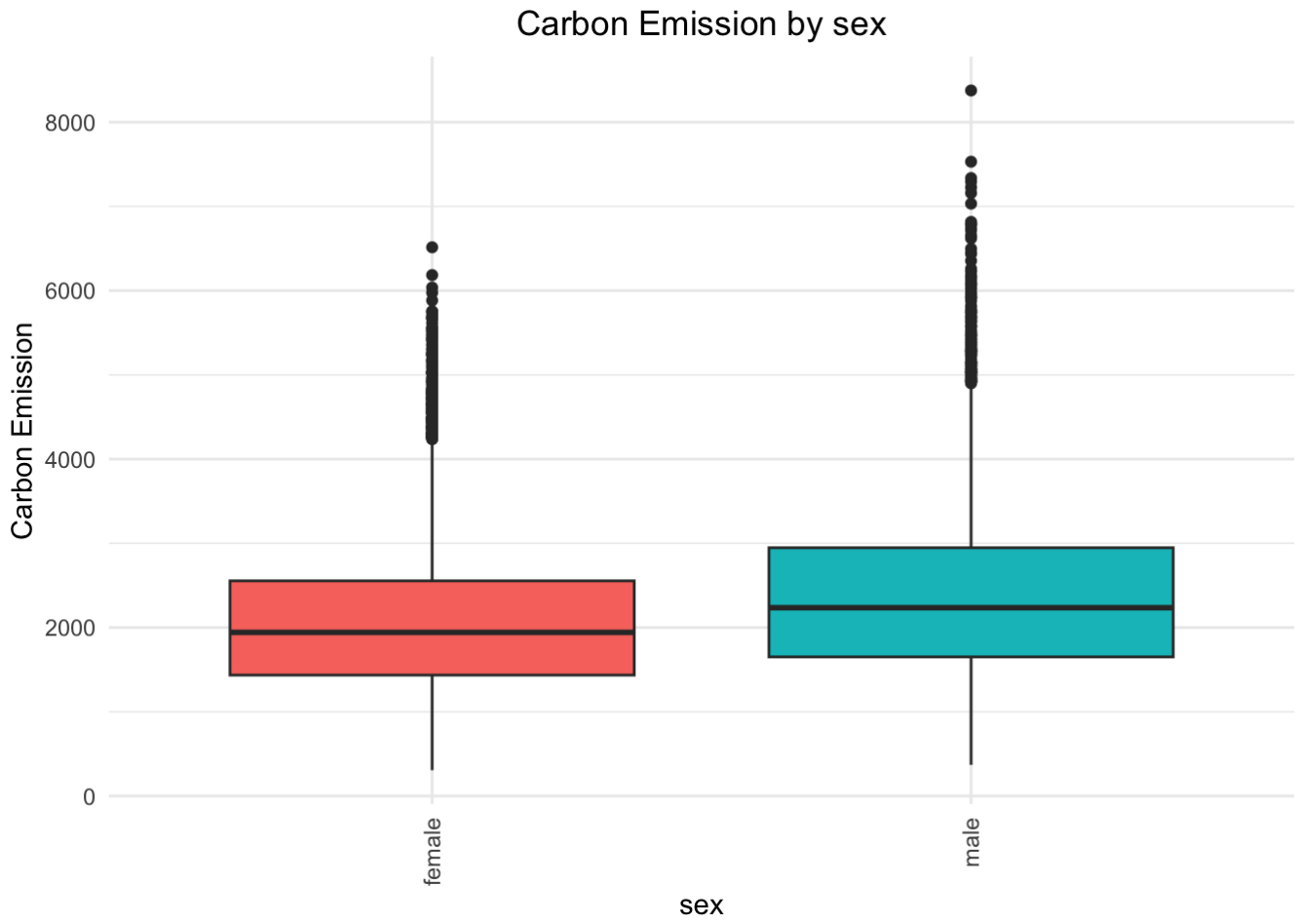


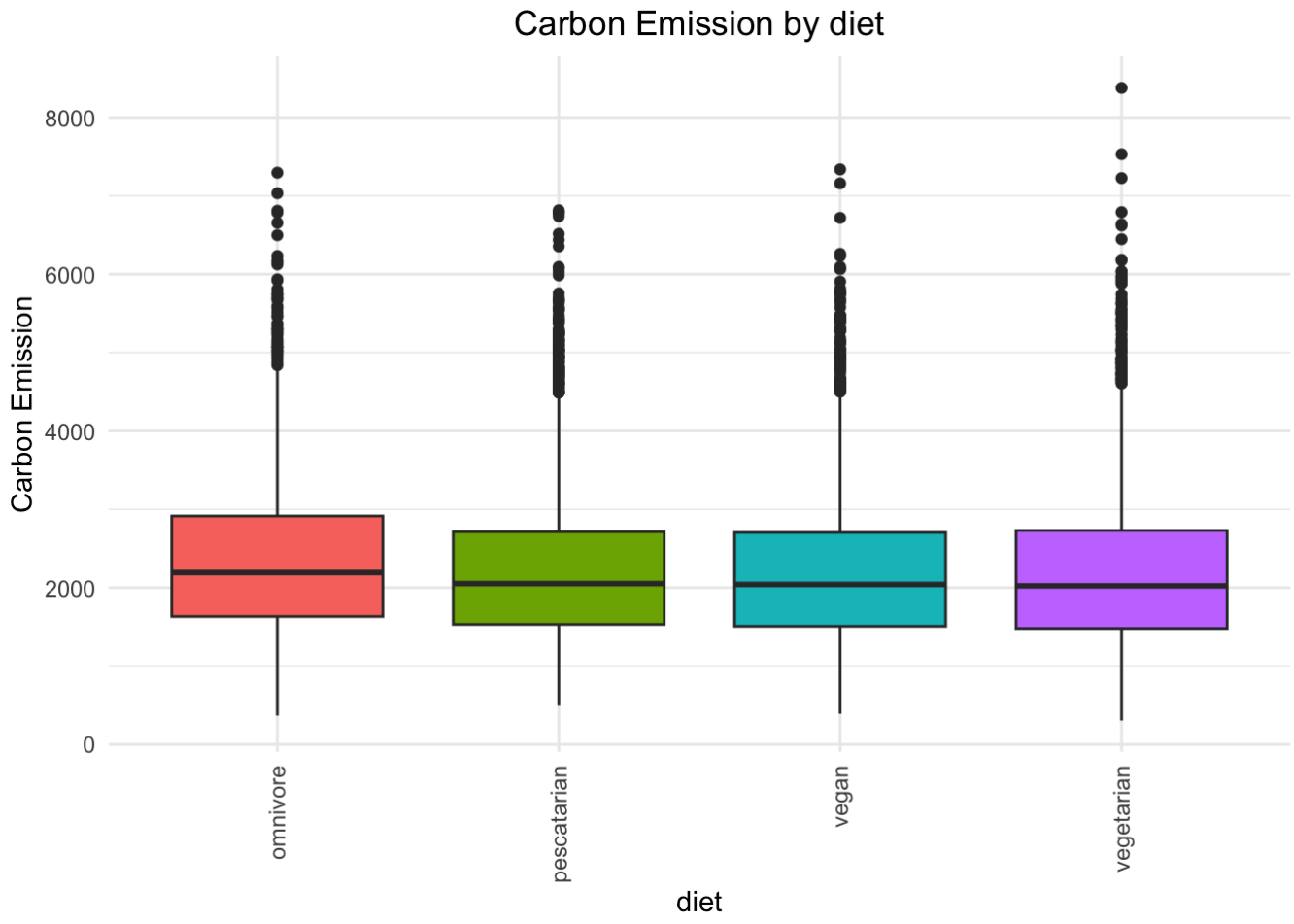
Although the correlation between `energy.eff` and the dependent variable `carbon.emission` is close to 0, but since it is the only negative value we decided to keep it. `tvpc.hour` does not seem to be related to any of the variables, so we decided to delete this variable when modeling.

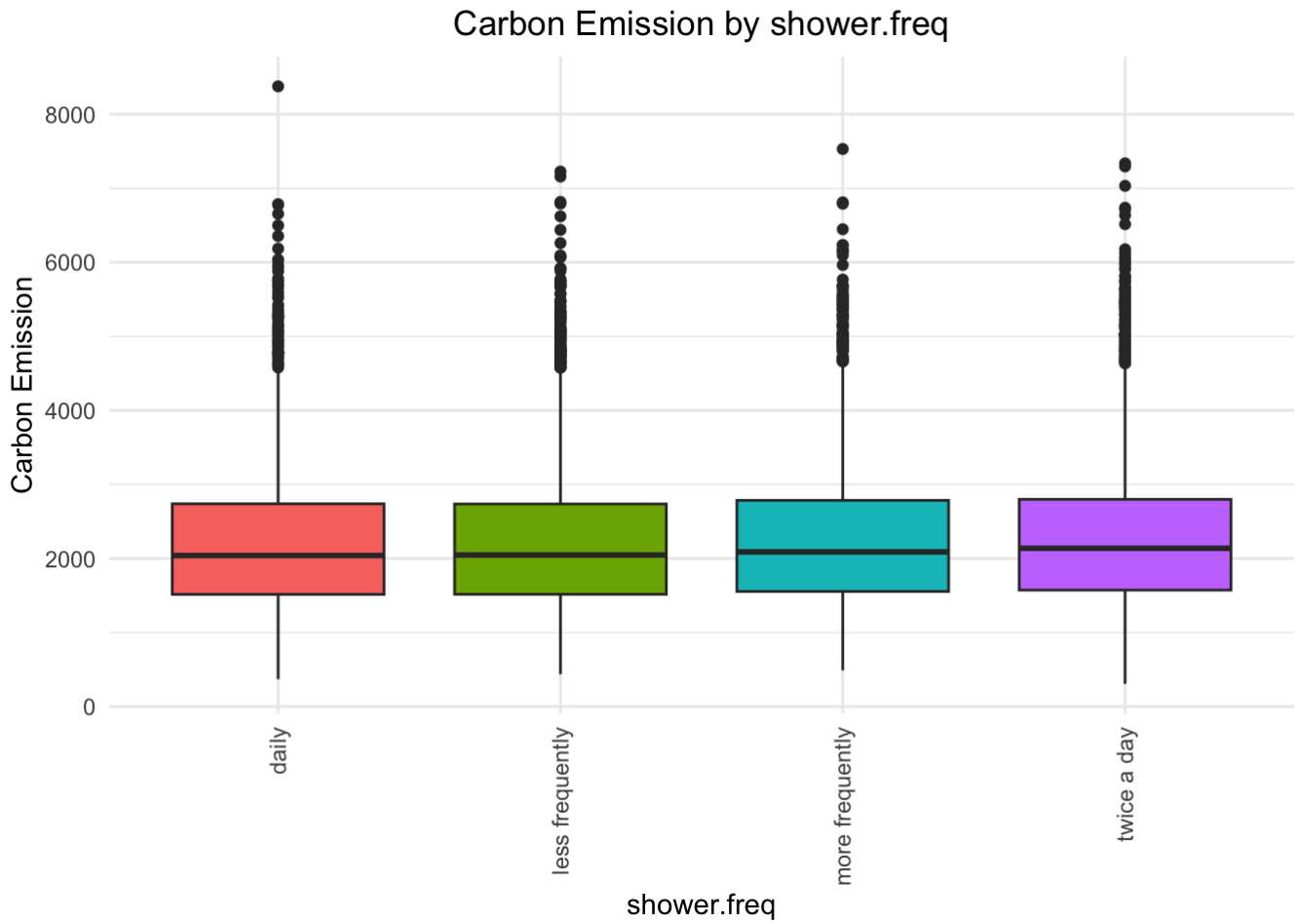
2. Category Variables

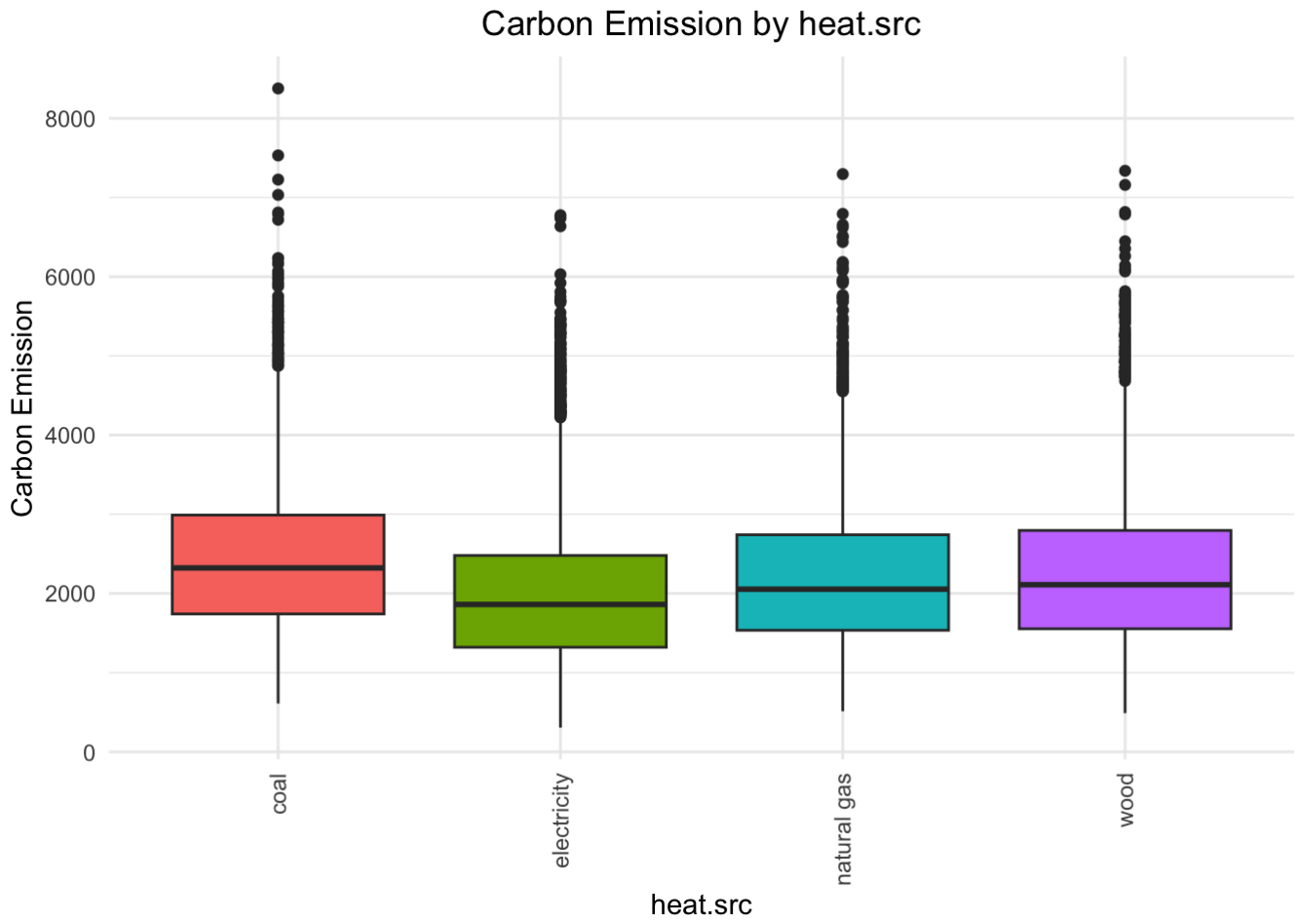
1. The relationship between category variables and carbon emissions

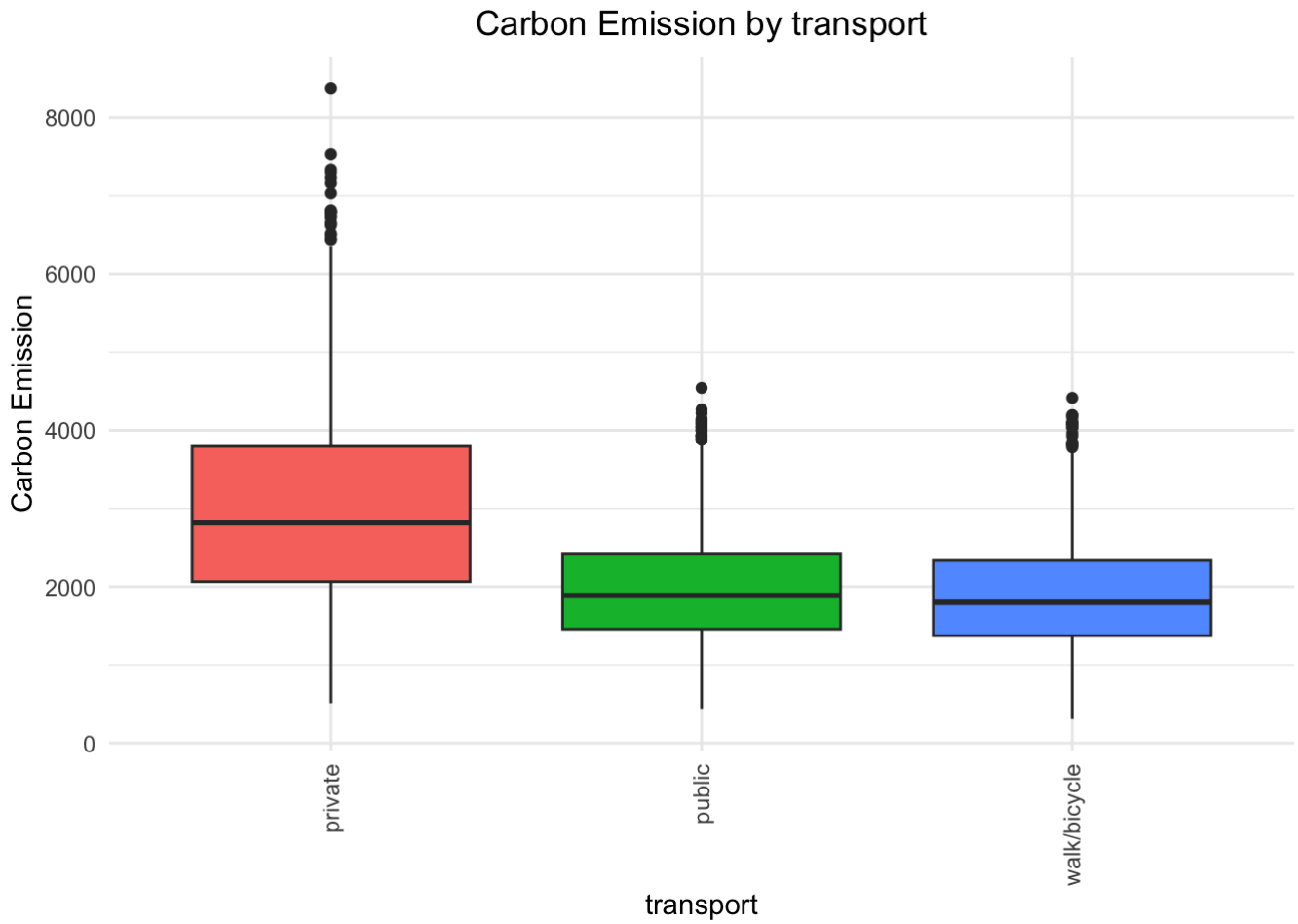
```
# Boxplot
for (var in char_vars) {
  print(ggplot(carbon, aes_string(x = var, y = "carbon.emission")) +
          geom_boxplot(aes_string(fill = var)) +
          labs(title = paste("Carbon Emission by", var),
               x = var, y = "Carbon Emission") +
          theme_minimal() +
          theme(plot.title = element_text(hjust = 0.5),
                axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5),
                legend.position = "none"))
}
```

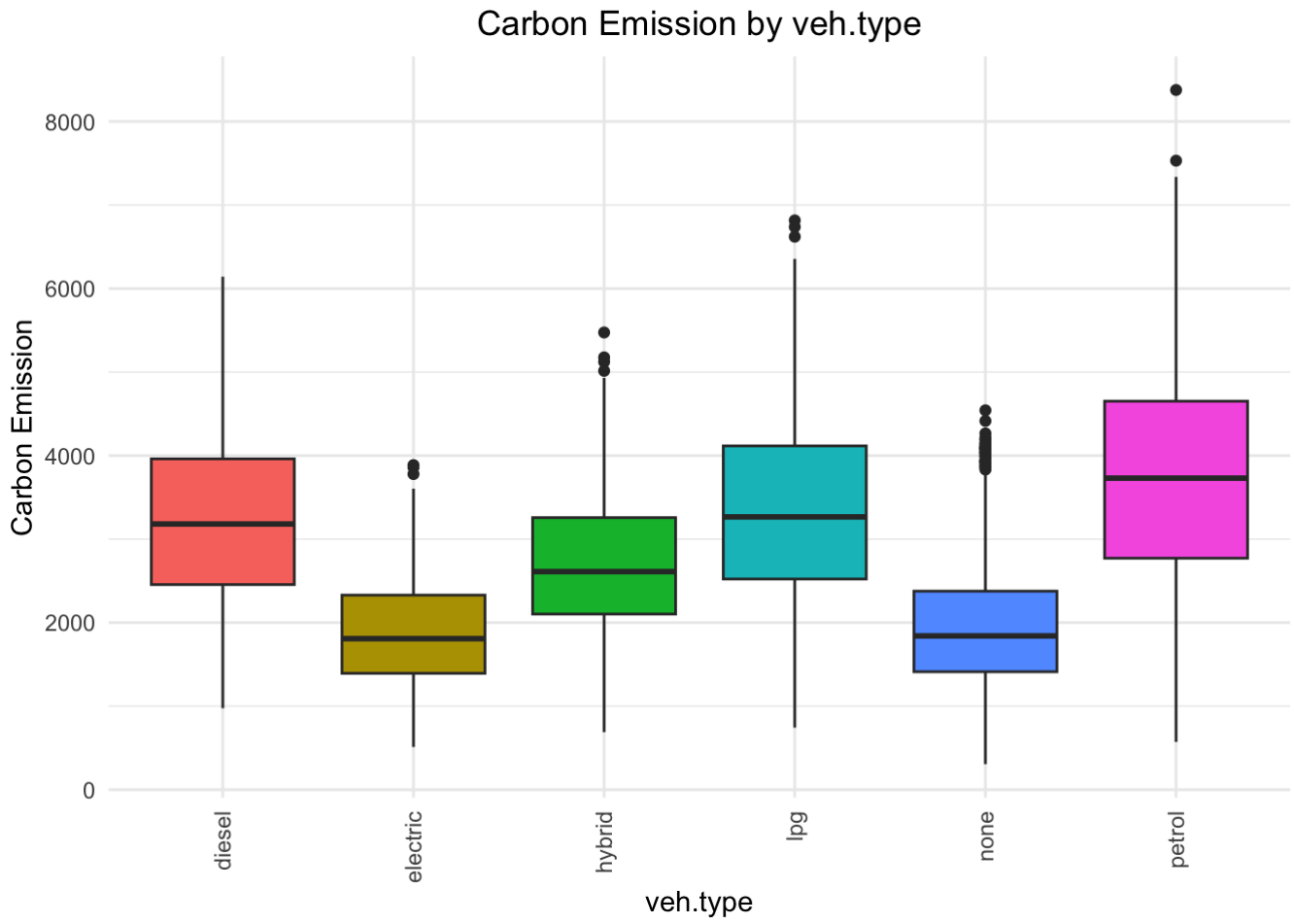

Carbon Emission by sex
Carbon Emission
8000
6000
4000
2000
0
female
male
sex

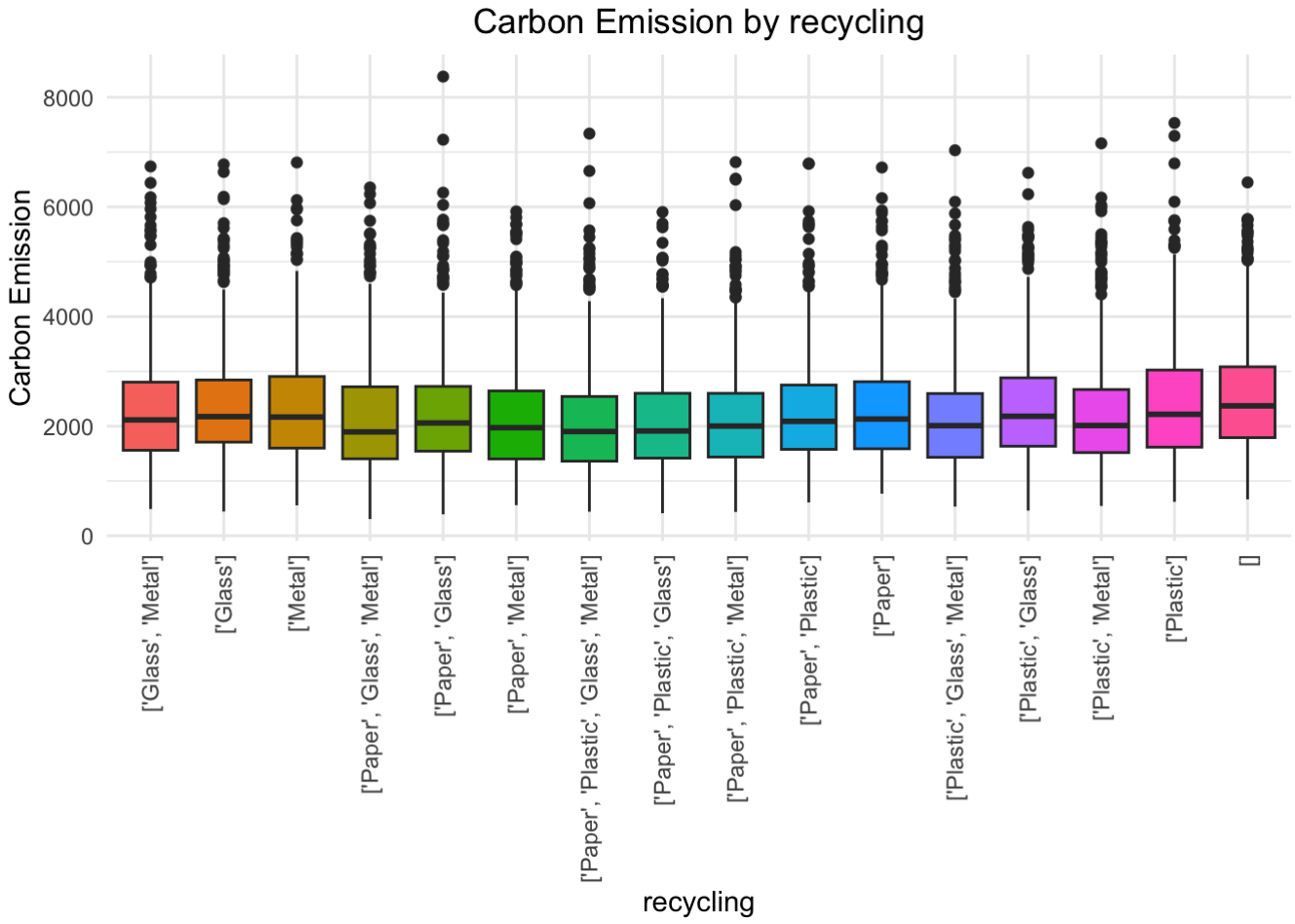

Carbon Emission by diet
Carbon Emission
8000
6000
4000
2000
0
omnivore
pescatarian
vegan
vegetarian
diet

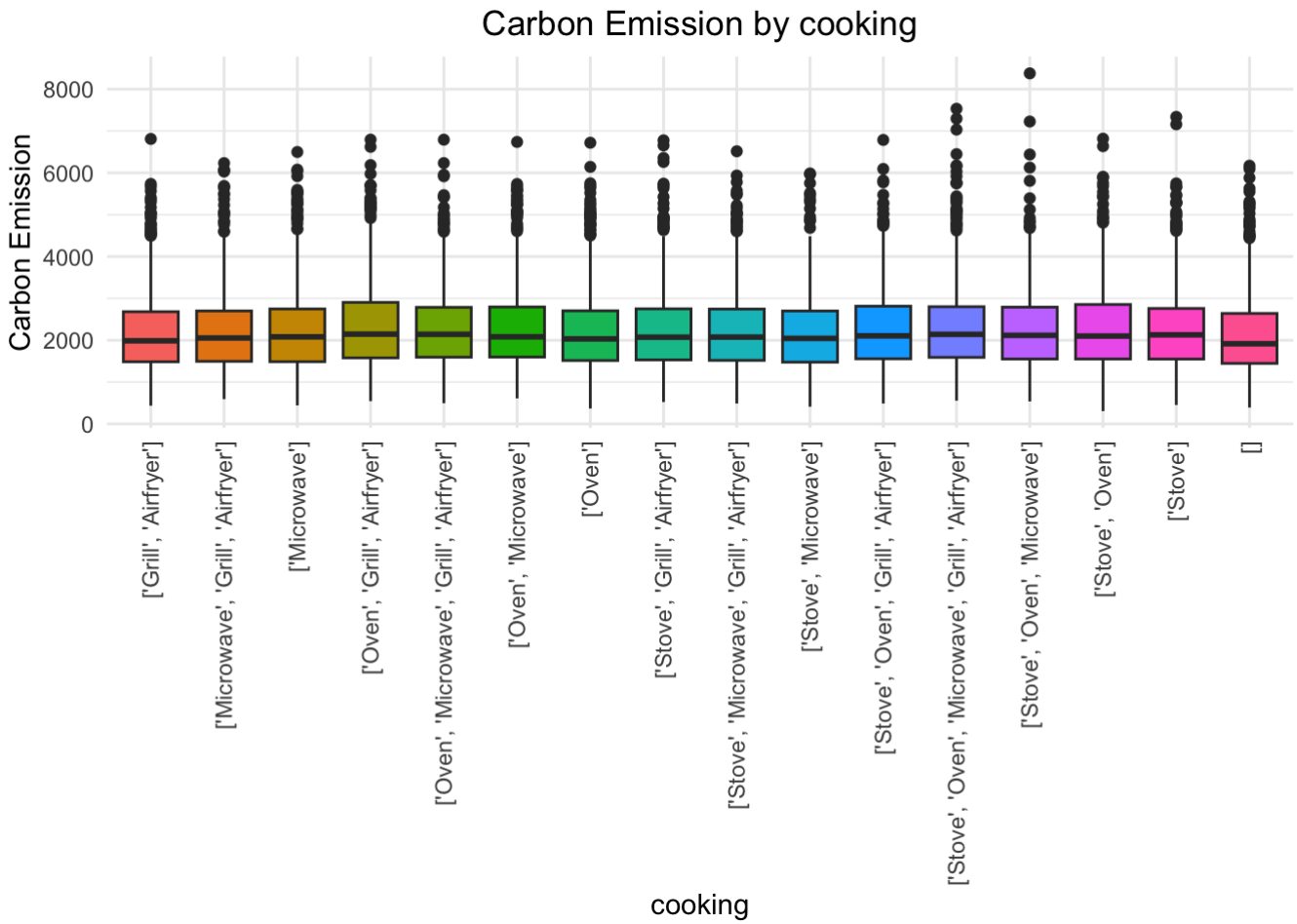

Carbon Emission by shower.freq
Carbon Emission
8000
6000
4000
2000
0
daily
less frequently
more frequently
twice a day
shower.freq


Carbon Emission by heat.src
Carbon Emission
8000
6000
4000
2000
0
coal
electricity
natural gas
wood
heat.src


Carbon Emission by transport
Carbon Emission
8000
6000
4000
2000
0
private
public
walk/bicycle
transport


Carbon Emission by veh.type
Carbon Emission
8000
6000
4000
2000
0
diesel
electric
hybrid
lpg
none
petrol
veh.type


Carbon Emission by recycling
Carbon Emission
8000
6000
4000
2000
0
['Glass', 'Metal']
['Glass']
['Metal']
['Paper', 'Glass', 'Metal']
['Paper', 'Glass']
['Paper', 'Metal']
['Paper', 'Plastic', 'Glass', 'Metal']
['Paper', 'Plastic', 'Glass']
['Paper', 'Plastic', 'Metal']
['Paper', 'Plastic']
['Paper']
['Plastic', 'Glass', 'Metal']
['Plastic', 'Glass']
['Plastic', 'Metal']
['Plastic']
[]
recycling


Carbon Emission by cooking
Carbon Emission
8000
6000
4000
2000
0
['Grill', 'Airfryer']
['Microwave', 'Grill', 'Airfryer']
['Microwave']
['Oven', 'Grill', 'Airfryer']
['Oven', 'Microwave', 'Grill', 'Airfryer']
['Oven', 'Microwave']
['Oven']
['Stove', 'Grill', 'Airfryer']
['Stove', 'Microwave', 'Grill', 'Airfryer']
['Stove', 'Microwave']
['Stove', 'Oven', 'Grill', 'Airfryer']
['Stove', 'Oven', 'Microwave', 'Grill', 'Airfryer']
['Stove', 'Oven', 'Microwave']
['Stove', 'Oven']
['Stove']
[]
cooking

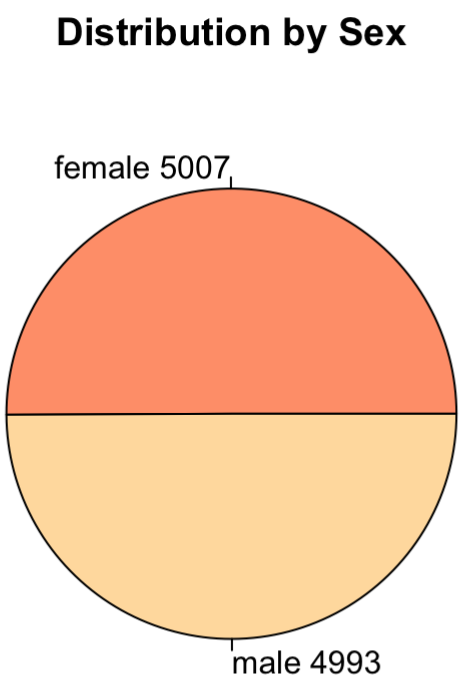
At this step, we can initially filter out some variables that we are not interested in.

From the boxplot, we can observe the following.

1. Variables with too many categories: `veh.type`, `recycling` and `cooking`. There are too many categories for these three variables, making it difficult to draw clear conclusions from the graph. Incorporating them all into a model may result in a model that is too complex to interpret and generalize.
2. The differences between groups are small and not very interpretable: `diet`, `shower.freq` and ``heat.src``. The difference in carbon emissions between different categories seems to be small. Incorporating these variables into the model may not significantly improve the model's predictive power.
3. Simplified model: Select a few key categorical variables - `sex` and `transport`. These two categorical variables seem to show a clear relationship with carbon emissions, so these two variables are retained. This helps keep the model simple and interpretable. Too many categorical variables may make the model complex and difficult to understand and apply. We can explore these two univariates further.

2. Sex

```
# Pie chart
pie(table(carbon$sex),
    labels = paste(unique(carbon$sex), " ", table(carbon$sex), sep=""),
    main = "Distribution by Sex",
    col = c("#FFA07A", "#FFDEAD"))
```



It can be seen from the histogram and pie chart that the gender distribution is roughly equal and can be included in the analysis as a meaningful binary variable.

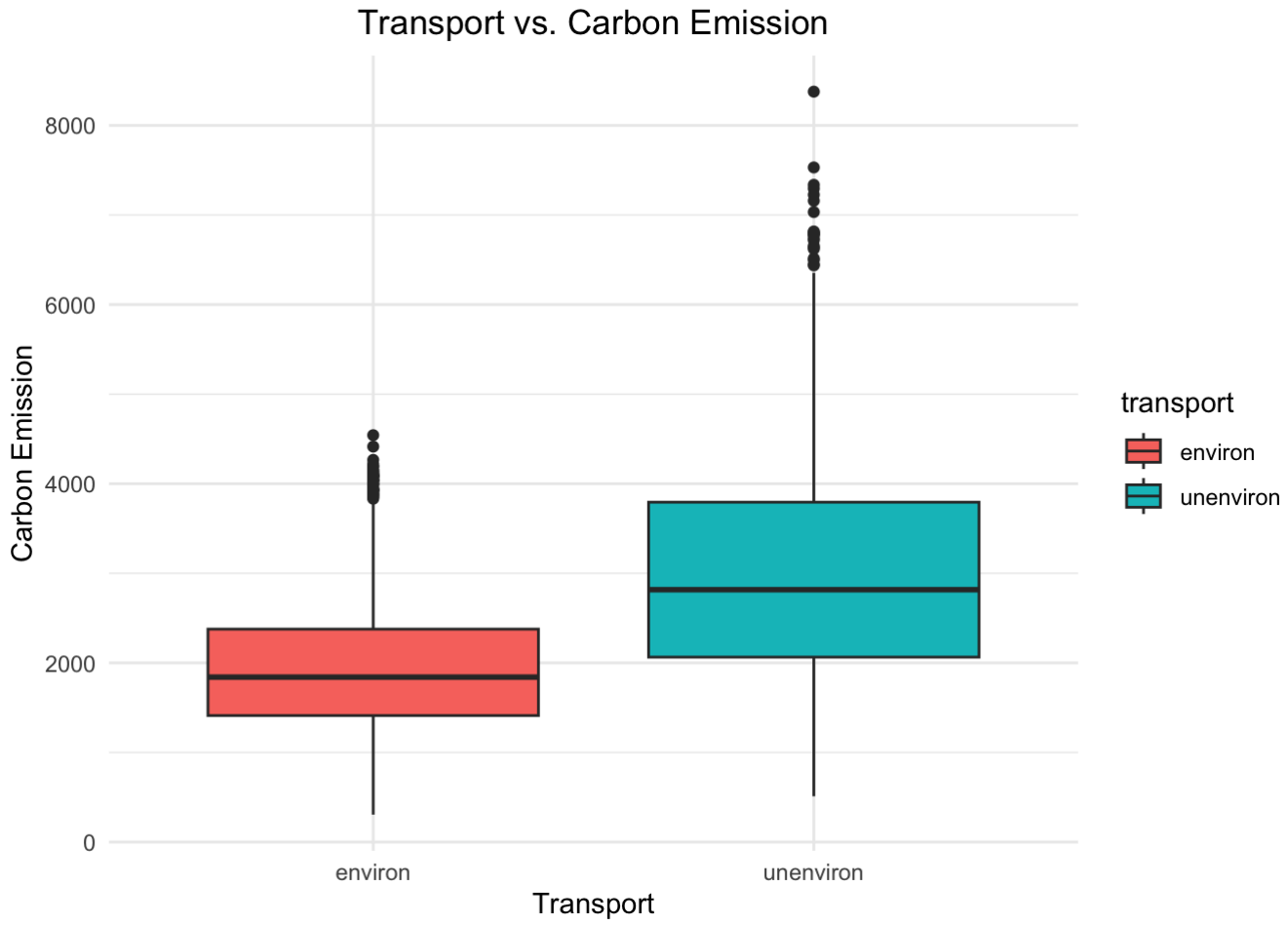
2. transport

We noticed that the variable `transport` , which means transportation preference, includes three categories: “private”, “public”, “walk/bicycle”. The first one is a less environmentally friendly form , the latter two are environmentally friendly modes of transportation.

So we decided to turn it into a dichotomous variable: “private” turned into “nonenviron”, “public” and “walk/bicycle” turned into “environ”.

```
carbon$transport <- ifelse(carbon$transport == "private", "unenviron", "environ")
carbon$transport <- as.factor(carbon$transport)
```

```
# Visualize the relationship between transport and carbon.emissions
ggplot(carbon, aes(x = transport, y = carbon.emission, fill = transport)) +
  geom_boxplot() +
  labs(title = "Transport vs. Carbon Emission", x = "Transport", y = "Carbon Emission")
+
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```




```
table(carbon$transport)
```

```
##
##   environ unenviron
##      6721      3279
```

```
prop.table(table(carbon$transport))
```

```
##
##   environ unenviron
##    0.6721    0.3279
```

We found that the number of samples in the two categories is very different, with 3279 samples in the ‘unenviron’ category and 6721 samples in the ‘environ’ category. This imbalance in sample size may affect the robustness of the analysis results.

Despite this limitation, we still analyzed because:

First, exploring the relationship between transportation modes and carbon emissions is important for understanding the impact of individual lifestyles on the environment. Secondly, the boxplot shows that the median carbon emission of the ‘Nonenviron’ category is significantly higher than that of the ‘Environ’ category, indicating that un-environmental friendly transportation modes may be associated with higher carbon emissions.

3. Hypothesis Test

We tried to determine

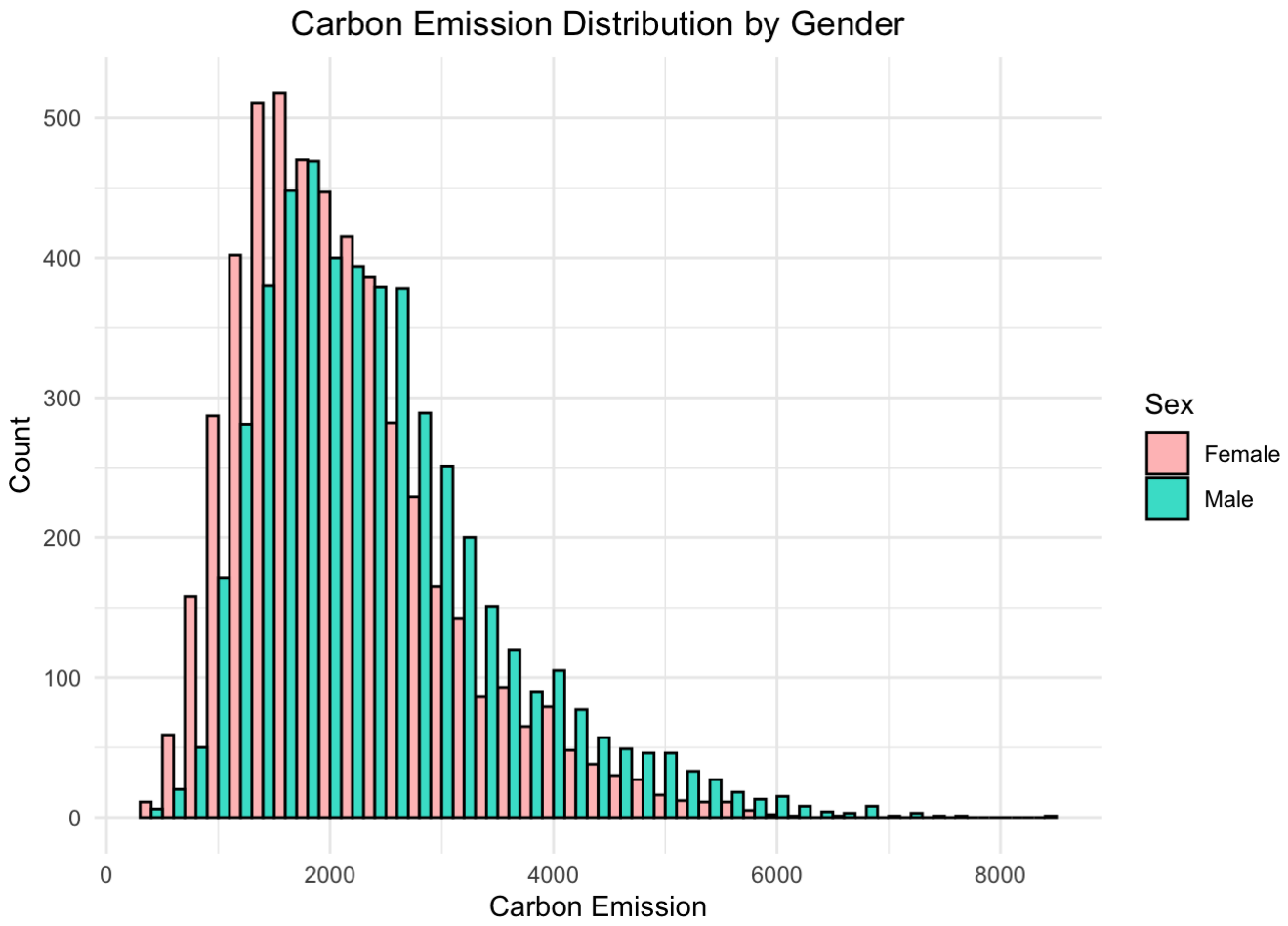
1. whether there is a significant difference in carbon emissions between male and female. (`sex`)
2. whether there is a significant difference in carbon emissions between environmentally friendly or not environmentally friendly modes of transportation. (`transport`)

3.1 Choose the Appropriate Test

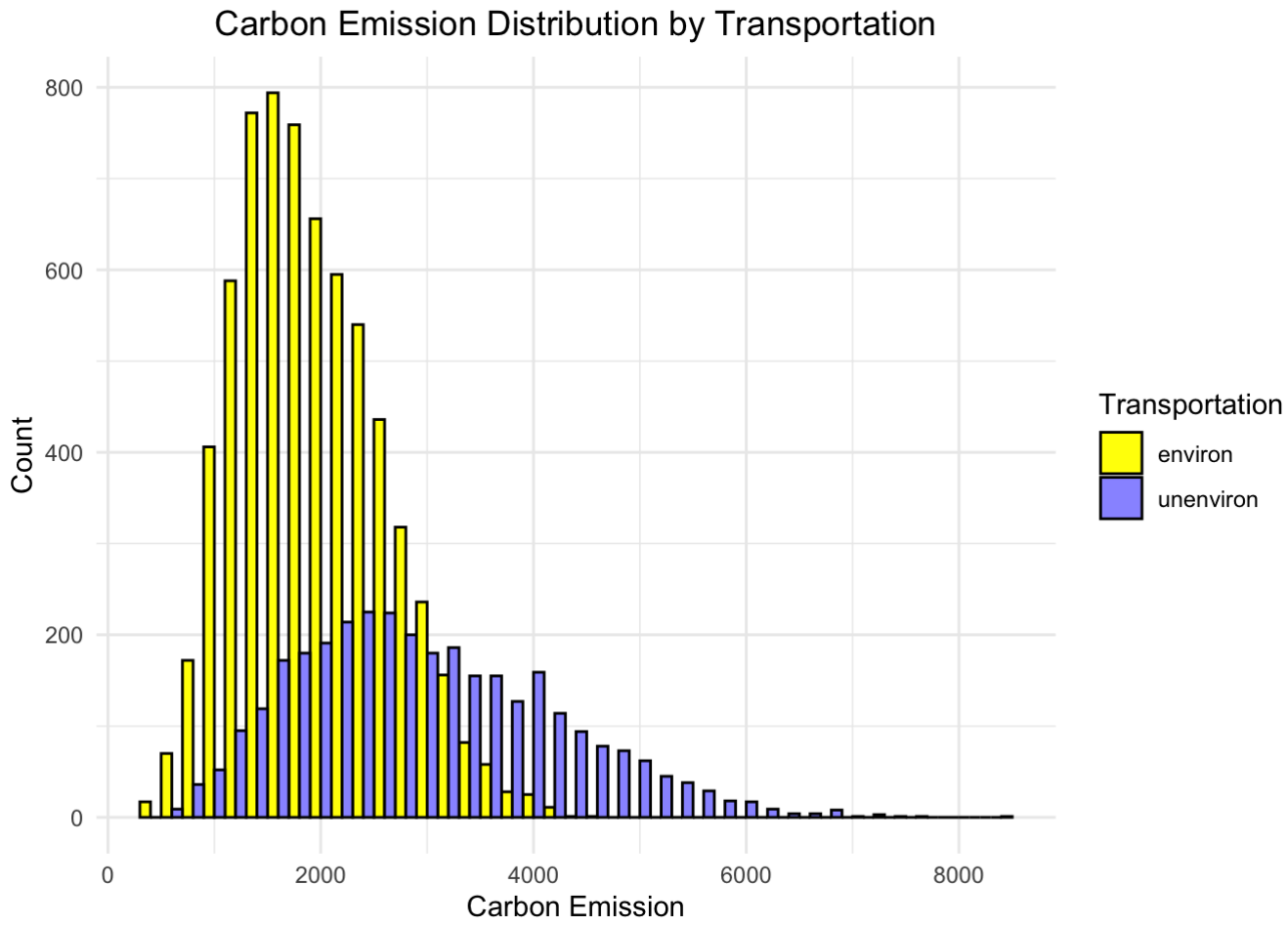
First, we should decide between a t-test and a Wilcoxon rank-sum test. The Wilcoxon rank-sum test is a non-parametric alternative to the t-test. It does not assume normality of data.

So let us check the normality.

```
# Grouped Histogram
# For sex
ggplot(carbon, aes(x = carbon.emission, fill = sex)) +
  geom_histogram(binwidth = 200, color = "black", position = "dodge") +
  theme_minimal() +
  labs(title = "Carbon Emission Distribution by Gender",
       x = "Carbon Emission", y = "Count") +
  scale_fill_manual(values = c("#FFC1C1", "#40E0D0"),
                    name = "Sex", labels = c("Female", "Male")) +
  theme(plot.title = element_text(hjust = 0.5))
```




```
# For transport
ggplot(carbon, aes(x = carbon.emission, fill = transport)) +
  geom_histogram(binwidth = 200, color = "black", position = "dodge") +
  theme_minimal() +
  labs(title = "Carbon Emission Distribution by Transportation",
       x = "Carbon Emission", y = "Count") +
  scale_fill_manual(values = c("#FFFF00", "#9999FF"),
                    name = "Transportation", labels = c("environ", "unenviron")) +
  theme(plot.title = element_text(hjust = 0.5))
```




```
# Check normality with Shapiro-Wilk test
# For sex
shapiro.test(carbon$carbon.emission[carbon$sex == "male"])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  carbon$carbon.emission[carbon$sex == "male"]
## W = 0.92683, p-value < 2.2e-16
```

```
shapiro.test(sample(carbon$carbon.emission[carbon$sex == "female"], 5000)) # Maximum sam
ple size 5000
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sample(carbon$carbon.emission[carbon$sex == "female"], 5000)
## W = 0.93508, p-value < 2.2e-16
```

```
# For transport
shapiro.test(sample(carbon$carbon.emission[carbon$transport == "environ"], 5000))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sample(carbon$carbon.emission[carbon$transport == "environ"], 5000)
## W = 0.98189, p-value < 2.2e-16
```

```
shapiro.test(sample(carbon$carbon.emission[carbon$transport == "unenviron"]))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sample(carbon$carbon.emission[carbon$transport == "unenviron"])
## W = 0.97658, p-value < 2.2e-16
```

The histograms indicate that the carbon emissions data for both `sex` and `transport` are right-skewed and do not follow a normal distribution. Our significance of level is 0.05. The Shapiro-Wilk tests used to assess the normality of data for both `sex` and `transport` have a p-value < 2.2e-16, which less than 0.05, indicating that the data significantly deviates from a normal distribution. This suggests that the t-test's assumption of normality is violated.

Given the results, the Wilcoxon rank-sum test is more appropriate, as it does not require normality of data.

3.2 Hypothesis Test

3.2.1 For sex

1. Hypothesis

Null Hypothesis H_0: $\text{Median}_{\text{male}} = \text{Median}_{\text{female}}$. The median carbon emissions for male is equal to the median carbon emissions for female. There is no significant difference in carbon emissions between males and females.

Alternative Hypothesis H_A: $\text{Median}_{\text{male}} \neq \text{Median}_{\text{female}}$. The median carbon emissions for male is not equal to the median carbon emissions for female. There is a significant difference in carbon emissions between males and females.

2. Wilcoxon Rank-Sum Test

```
wilcox.test(carbon.emission ~ sex, data = carbon,
            paired = F, alternative = "two.sided")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  carbon.emission by sex
## W = 10155931, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Assuming $\alpha = 0.05$, the Wilcoxon rank-sum test gives a p-value < 2.2e-16, which is less than our significance level of 0.05. This means we have sufficient evidence to reject the null hypothesis in favor of the alternative hypothesis. We can conclude that there is a significant difference in the median carbon emissions between males and females.

We can get a sense of the difference

```
tapply(carbon$carbon.emission, carbon$sex, summary)
```

```
## $female
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     306    1435    1942    2103    2554    6515
##
## $male
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     369    1651    2235    2436    2947    8377
```

The median carbon emission for males (2235) is slightly higher than for females (1942).

3.2.2 For transport

1. Hypothesis

Null Hypothesis H_0: $\text{Median}_{\text{environ}} = \text{Median}_{\text{unenviron}}$. The median carbon emissions for environmental friendly transportation is equal to the median carbon emissions for environmental not friendly transportation. There is no significant difference in carbon emissions between these two transportation.

Alternative Hypothesis H_A: $\text{Median}_{\text{environ}} \neq \text{Median}_{\text{unenviron}}$. The median carbon emissions for environmental friendly transportation is not equal to the median carbon emissions for environmental not friendly transportation. There is significant difference in carbon emissions between these two transportation.

2. Wilcoxon Rank-Sum Test

```
wilcox.test(carbon.emission ~ transport, data = carbon,
            paired = F, alternative = "two.sided")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  carbon.emission by transport
## W = 5093480, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Assuming $\alpha = 0.05$, the Wilcoxon rank-sum test gives a p-value < 2.2e-16, which is less than our significance level of 0.05. This means we have sufficient evidence to reject the null hypothesis in favor of the alternative hypothesis. We can conclude that there is a significant difference in the median carbon emissions between environmental friendly and environmental not friendly transportation. We can get a sense of the difference

```
tapply(carbon$carbon.emission, carbon$sex, summary)
```

```
## $female
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     306    1435    1942    2103    2554    6515
##
## $male
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     369    1651    2235    2436    2947    8377
```

```
tapply(carbon$carbon.emission, carbon$transport, summary)
```

```
## $environ
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     306    1412    1841    1922    2377    4542
##
## $unenviron
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     511    2064    2817    2981    3794    8377
```

The median carbon emission for males (2235) is slightly higher than for females (1942). The median carbon emission for environmental friendly transportation (1841) is quite lower than for not environmental friendly transportation (2817).

3.3 Conclusion

1. For `sex`

At 0.05 level of significance, based on the Wilcoxon rank-sum test, out p-value < 2.2e-16, which is less than 0.05, indicating that we can reject the null hypothesis. The median carbon emissions for male is not equal to the median carbon emissions for female. And we can conclude that there is a statistically significant difference in carbon emissions between males and females, with males (2235) having a higher median carbon emission than females (1942).

1. For `transport`

At 0.05 level of significance, based on the Wilcoxon rank-sum test, out p-value < 2.2e-16, which is less than 0.05, indicating that we can reject the null hypothesis. The median carbon emissions for environmental friendly transportation is not equal to the median carbon emissions for environmental not friendly transportation. And we can conclude that there is a statistically significant difference in carbon emissions between these two transportation, with environmental not friendly transportation (2817) having a higher median carbon emission than environmental friendly transportation (1841).

4. Analysis of Variance (ANOVA)

4.1 One-way ANOVA

1.

```
# One-way ANOVA for bodytype
anova_bodytype <- aov(carbon.emission ~ as.factor(body.type), data = carbon)
summary(anova_bodytype)
```

```
##                         Df    Sum Sq   Mean Sq F value Pr(>F)
## as.factor(body.type)     3 4.331e+08 144374189   145.4 <2e-16 ***
## Residuals             9996 9.922e+09    992644
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The one-way ANOVA results show that assuming $\alpha = 0.05$, the p-value is less than 2e-16, which is much smaller than the significance level of 0.05, indicating strong evidence against the null hypothesis that all body type groups have the same mean carbon emissions. So at least one mean is different.

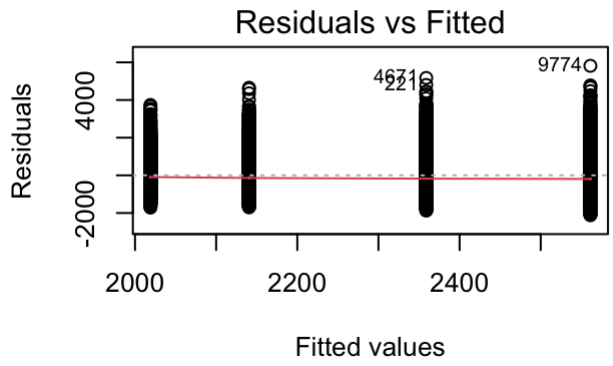
2. Post-hoc testing

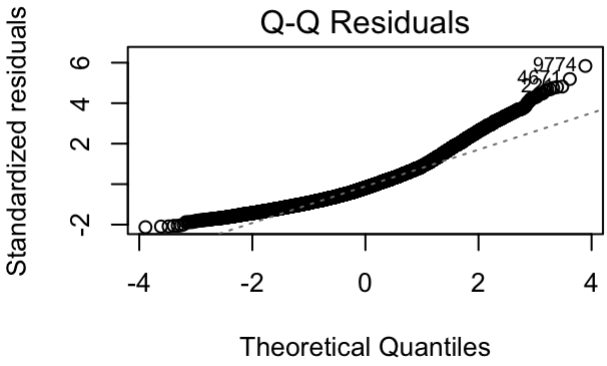
```
# Tukey's Honest Significant Difference (HSD) Test
TukeyHSD(anova_bodytype)
```

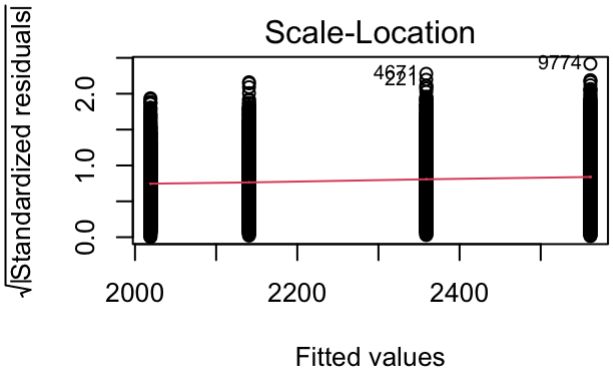
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = carbon.emission ~ as.factor(body.type), data = carbon)
##
## $`as.factor(body.type)`
##         diff      lwr      upr    p adj
## 2-1 121.4823  49.1622 193.8025 9.44e-05
## 3-1 339.8928 267.6759 412.1097 0.00e+00
## 4-1 542.1848 470.0628 614.3068 0.00e+00
## 3-2 218.4105 145.7112 291.1098 0.00e+00
## 4-2 420.7025 348.0975 493.3075 0.00e+00
## 4-3 202.2920 129.7898 274.7942 0.00e+00
```

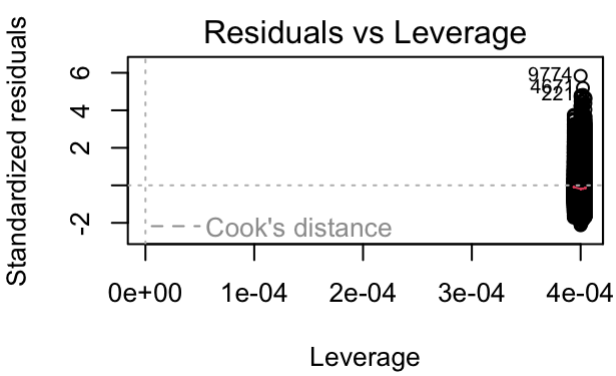
3. Diagnostic plots

```
# Visualization
par(mfrow=c(2,2))
plot(anova_bodytype)
```









4.2 Two-way ANOVA

Now, let us consider the impact of multiple independent factors simultaneously using a two-way ANOVA. We will examine the effects of `body.type` and `sex` on carbon emissions.

1.

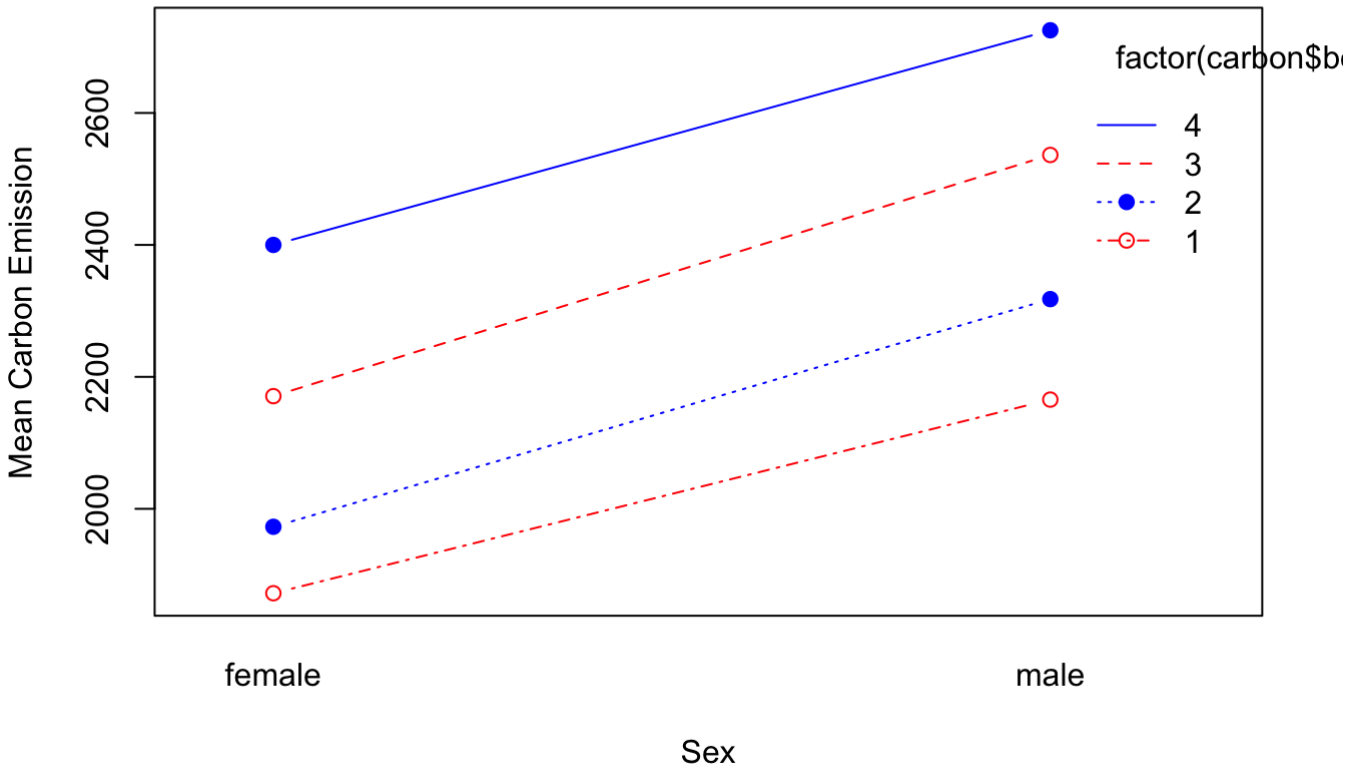
```
# Two-way ANOVA for body.type and sex
two_anova <- aov(carbon.emission ~ as.factor(body.type) * sex, data = carbon)
summary(two_anova)
```

```
##                             Df    Sum Sq   Mean Sq F value Pr(>F)
## as.factor(body.type)         3 4.331e+08 144374189 149.567 <2e-16 ***
## sex                          1 2.756e+08 275621045 285.535 <2e-16 ***
## as.factor(body.type):sex     3 1.776e+06    592071   0.613  0.606
## Residuals                 9992 9.645e+09    965280
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Assuming $\alpha = 0.05$, the two-way ANOVA results indicate that both `body.type` (p < 2e-16) and `sex` (p <2e-16) have significant main effects on carbon emissions respectively. However, the interaction effect between `body.type` and `sex` is not significant (p = 0.606). This suggests that the effect of sex on carbon emissions does not depend on body type, and vice versa.

2. Visualization

```
interaction.plot(x.factor = carbon$sex, trace.factor = factor(carbon$body.type),
                 response = carbon$carbon.emission, fun = mean,
                 type = "b", legend = TRUE,
                 xlab = "Sex", ylab = "Mean Carbon Emission",
                 pch = c(1, 19), col = c("red", "blue"))
```



3. Post-hoc testing

```
TukeyHSD(two_anova)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = carbon.emission ~ as.factor(body.type) * sex, data = carbon)
##
## $`as.factor(body.type)`
##         diff      lwr      upr    p adj
## 2-1 121.4823  50.1660 192.7987 7.17e-05
## 3-1 339.8928 268.6783 411.1074 0.00e+00
## 4-1 542.1848 471.0639 613.3058 0.00e+00
## 3-2 218.4105 146.7202 290.1007 0.00e+00
## 4-2 420.7025 349.1053 492.2997 0.00e+00
## 4-3 202.2920 130.7962 273.7879 0.00e+00
##
## $sex
##                 diff      lwr     upr p adj
## male-female 331.9666 293.4491 370.484     0
##
## $`as.factor(body.type):sex`
##                           diff        lwr       upr     p adj
## 2:female-1:female  100.726051  -17.49275  218.9449 0.1621394
## 3:female-1:female  298.709899  178.93607  418.4837 0.0000000
## 4:female-1:female  527.928978  409.47631  646.3816 0.0000000
## 1:male-1:female    293.465997  175.27040  411.6616 0.0000000
## 2:male-1:female    445.794646  325.89325  565.6960 0.0000000
## 3:male-1:female    664.273627  546.26257  782.2847 0.0000000
## 4:male-1:female    853.231501  734.25273  972.2103 0.0000000
## 3:female-2:female  197.983848   78.27897  317.6887 0.0000149
## 4:female-2:female  427.202927  308.81998  545.5859 0.0000000
## 1:male-2:female    192.739946   74.61423  310.8657 0.0000211
## 2:male-2:female    345.068595  225.23608  464.9011 0.0000000
## 3:male-2:female    563.547576  445.60650  681.4886 0.0000000
## 4:male-2:female    752.505450  633.59609  871.4148 0.0000000
## 4:female-3:female  229.219079  109.28323  349.1549 0.0000002
## 1:male-3:female     -5.243903 -124.92586  114.4381 1.0000000
## 2:male-3:female    147.084746   25.71788  268.4516 0.0058724
## 3:male-3:female    365.563728  246.06401  485.0634 0.0000000
## 4:male-3:female    554.521602  434.06613  674.9771 0.0000000
## 1:male-4:female   -234.462982 -352.82275 -116.1032 0.0000001
## 2:male-4:female    -82.134333 -202.19757   37.9289 0.4320714
## 3:male-4:female    136.344649   18.16916  254.5201 0.0111159
## 4:male-4:female    325.302523  206.16065  444.4444 0.0000000
## 2:male-1:male      152.328649   32.51904  272.1383 0.0029437
## 3:male-1:male      370.807631  252.88982  488.7254 0.0000000
## 4:male-1:male      559.765504  440.87922  678.6518 0.0000000
## 3:male-2:male      218.478982   98.85141  338.1066 0.0000009
## 4:male-2:male      407.436855  286.85454  528.0192 0.0000000
## 4:male-3:male      188.957874   70.25505  307.6607 0.0000387
```

5. Multiple Linear Regression

We finally chose 12 variables `body.type` , `sex` , `transport` , `internet.hour` , `social.act` , `groc.bill` , `wastebag.size` , `energy.eff` , `wastebag.num` , `new.clothes` , `airtvl.freq` and `veh.distance` as our independent variables.

Our dependent variables is `carbon.emission` .

```
carbon$sex.num <- ifelse(carbon$sex == "female", 0, 1)
carbon$trans.num <- ifelse(carbon$transport == "environ", 0, 1)

model_multiple <- lm(carbon.emission ~ body.type + sex.num + trans.num +
                     internet.hour + social.act + groc.bill + wastebag.size + energy.eff
+
                     wastebag.num + new.clothes + airtvl.freq + veh.distance,
                     data = carbon)
# Calculating VIF
vif_linear <- vif(model_multiple); vif_linear
```

```
##     body.type       sex.num     trans.num internet.hour    social.act
##      1.000997      1.000798      2.536330      1.000775      1.001090
##     groc.bill wastebag.size    energy.eff  wastebag.num   new.clothes
##      1.001879      1.002669      1.001335      1.000528      1.000329
##   airtvl.freq  veh.distance
##      1.000807      2.536657
```

```
# Our model
summary(model_multiple)
```

```
##
## Call:
## lm(formula = carbon.emission ~ body.type + sex.num + trans.num +
##     internet.hour + social.act + groc.bill + wastebag.size +
##     energy.eff + wastebag.num + new.clothes + airtvl.freq + veh.distance,
##     data = carbon)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2769.20  -228.33    14.16   252.39  3132.88
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.244e+03  3.423e+01 -36.343  < 2e-16 ***
## body.type      1.782e+02  4.556e+00  39.115  < 2e-16 ***
## sex.num        3.450e+02  1.022e+01  33.766  < 2e-16 ***
## trans.num      1.341e+02  1.733e+01   7.742 1.07e-14 ***
## internet.hour  6.650e+00  7.021e-01   9.471  < 2e-16 ***
## social.act     8.709e+01  6.232e+00  13.976  < 2e-16 ***
## groc.bill      9.718e-01  7.077e-02  13.731  < 2e-16 ***
## wastebag.size  1.257e+02  4.565e+00  27.541  < 2e-16 ***
## energy.eff    -3.156e+01  6.321e+00  -4.993 6.04e-07 ***
## wastebag.num   8.074e+01  2.567e+00  31.457  < 2e-16 ***
## new.clothes    1.369e+01  3.475e-01  39.403  < 2e-16 ***
## airtvl.freq    4.396e+02  4.571e+00  96.184  < 2e-16 ***
## veh.distance   1.999e-01  2.937e-03  68.079  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 510.7 on 9987 degrees of freedom
## Multiple R-squared:  0.7485, Adjusted R-squared:  0.7481
## F-statistic:  2476 on 12 and 9987 DF,  p-value: < 2.2e-16
```

VIF values provide an indication of multicollinearity among the independent variables. Generally, a VIF greater than 5 or 10 indicates high correlation and potential redundancy among predictor variables. In our model, all VIF values are low (almost each one are close to 1), which suggests there is no significant multicollinearity among our predictors. We think it is ideal and there is no need for us to add indicator variables.

6. Model Selection and Evaluation

6.1 Divide the Data

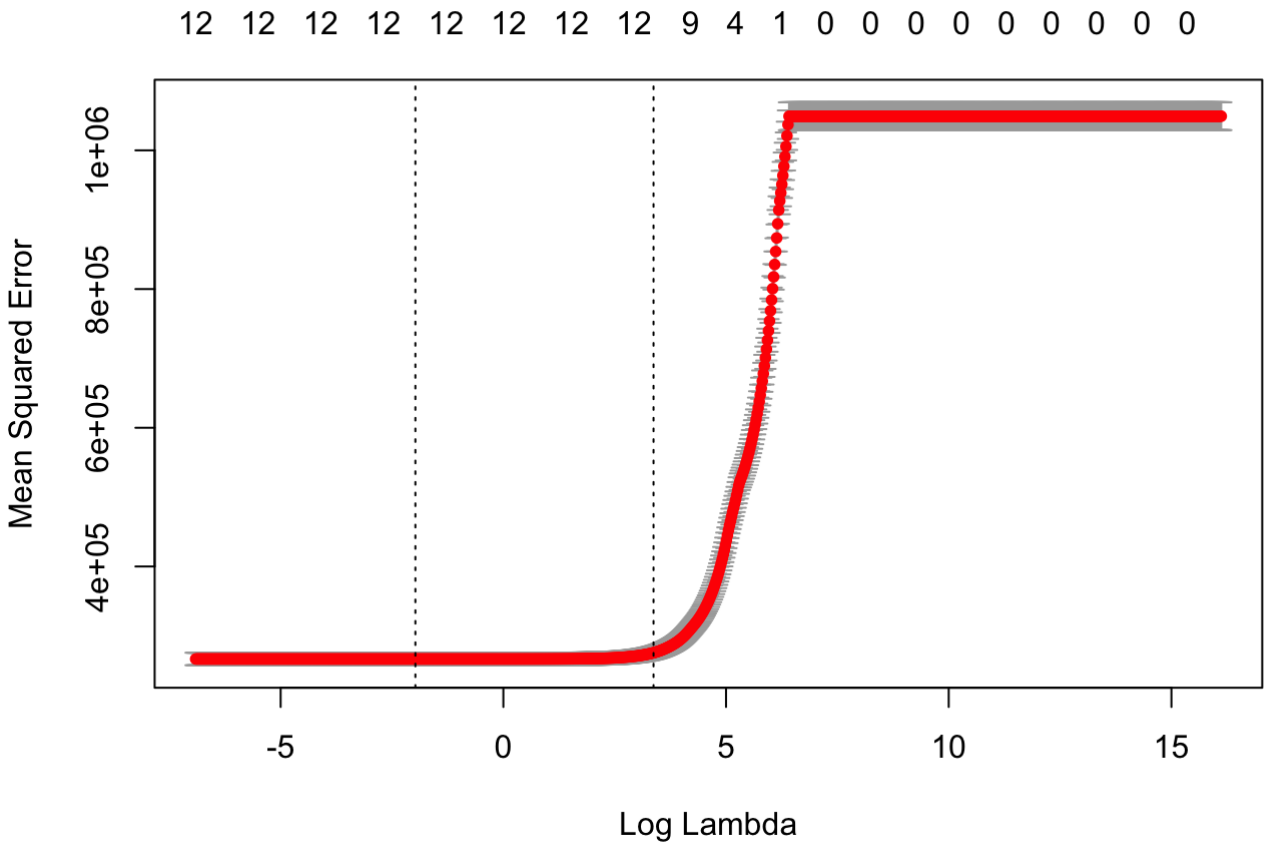
First we divide our data into two equal-sized samples, the in-sample and the out-sample.

```
set.seed(123)
carbon_data <- carbon[, sapply(carbon, is.numeric)]
carbon_data <- carbon_data[, -8]

# In-sample: Half the rows at random
train_rows <- sample(1:nrow(carbon), nrow(carbon)/2, replace = F)
# Out-sample: the other half
test_rows <- setdiff(1:nrow(carbon), train_rows)

train_data <- carbon[train_rows,]
test_data <- carbon[test_rows,]
# Convert the data to matrix and vectors

x <- as.matrix(carbon_data[, -11])
y <- carbon_data$carbon.emission

train_x <- x[train_rows, ]
train_y <- y[train_rows]
test_x <- x[test_rows, ]
test_y <- y[test_rows]
```

6.2 Lasso

Then applied Lasso model.

```
set.seed(1)
# Create a sequence of lambda values
lambdalevels <- 10 ^ seq(7 , -3, length = 1000)
# The lasso model
lasso_model <- glmnet(train_x, train_y, alpha = 1,
                      lambda = lambdalevels)

cvlasso <- cv.glmnet(train_x, train_y,
                     alpha = 1, lambda = lambdalevels)

# Plot the average MSE for each lambda level
plot(cvlasso, xlab = "Log Lambda", ylab = "Mean Squared Error")
```




```
cvlasso
```

```
##
## Call:  cv.glmnet(x = train_x, y = train_y, lambda = lambdalevels, alpha = 1)
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure   SE Nonzero
## min  0.139   786  266642 9017      12
## 1se 29.138   554  275638 9059      11
```

```
best_lambda <- cvlasso$lambda.min; best_lambda
```

```
## [1] 0.138721
```

The dotted vertical line on the left shows the lambda that gets the best MSE. So we can see the best fitting lambda level of the pure lasso model is 0.139 with 12 coefficients.

Now we looked at the coefficients.

```
predict(lasso_model, type = "coefficients",s = best_lambda)
```

```
## 13 x 1 sparse Matrix of class "dgCMatrix"
##                          s1
## (Intercept)   -1278.2237252
## body.type       180.9479380
## social.act       95.1291095
## groc.bill         0.8887496
## airtvl.freq     436.6782606
## veh.distance      0.2045259
## wastebag.size   135.6688514
## wastebag.num     79.0879241
## new.clothes      13.8683440
## internet.hour     5.5902901
## energy.eff      -25.7667757
## sex.num         364.0286311
## trans.num       124.2152576
```

6.3 Comparison

6.3.1 Regression

```
# MSE
lmout <- lm(train_y ~ train_x)
yhat_reg <- cbind(1, test_x) %*% lmout$coefficients
mse_reg <- sum((test_y - yhat_reg) ^ 2) / nrow(test_x); mse_reg
```

```
## [1] 256910.5
```

```
# Calculated R-squared by hand
tss <- sum((test_y - mean(test_y))^2)
sse_reg <- sum((test_y - yhat_reg)^2)
r2_reg <- (tss - sse_reg) / tss
r2_reg
```

```
## [1] 0.7486205
```

6.3.2 Lasso

```
# MSE
yhat_lasso <- predict(cvlasso$glmnet.fit,
                      s = cvlasso$lambda.min, newx = test_x)
mse_lasso <- sum((test_y - yhat_lasso) ^ 2) / nrow(test_x); mse_lasso
```

```
## [1] 256900
```

```
# Calculated R-squared by hand
sse_lasso <- sum((test_y - yhat_lasso)^2)
r2_lasso <- (tss - sse_lasso) / tss
r2_lasso
```

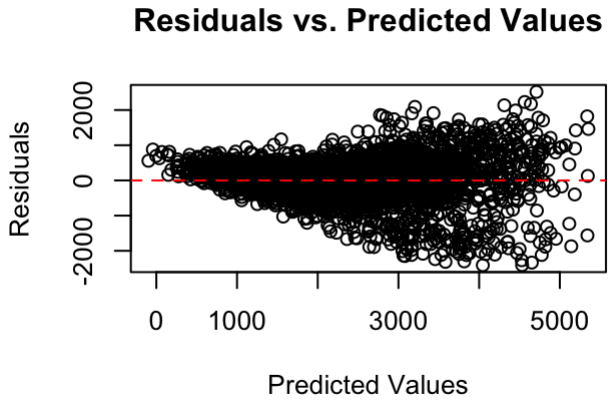
```
## [1] 0.7486306
```

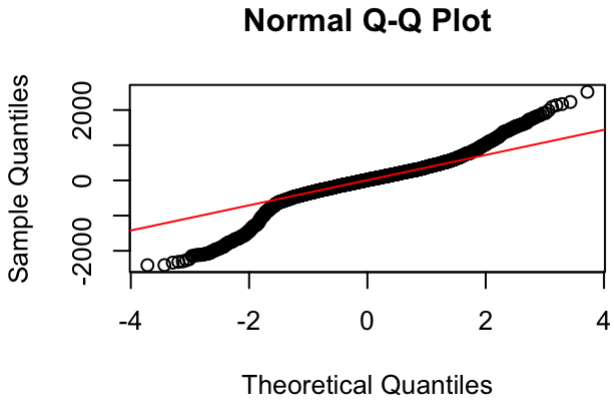
The MSE and R-squared of the two models is very, but the R-squared of the Lasso model 0.7486306 is slightly greater than the multiple linear regression model 0.7486205 and the MSE 256900 is smaller than the regression model 256910.5. So we will choose Lasso model.

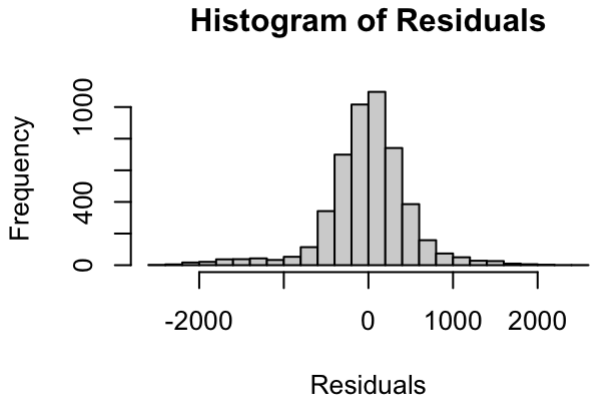
6.4 Residual Analysis

We have already got the R-squared 0.7486306 and MSE 256900 in previous question. Now we can do residual analysis.

```
residuals_lasso <- test_y - yhat_lasso
par(mfrow = c(2, 2))
plot(yhat_lasso, residuals_lasso,
     xlab = "Predicted Values", ylab = "Residuals",
     main = "Residuals vs. Predicted Values")
abline(h = 0, col = "red", lty = 2)

qqnorm(residuals_lasso)
qqline(residuals_lasso, col = "red")

hist(residuals_lasso, breaks = 30,
     main = "Histogram of Residuals", xlab = "Residuals")
```







So our model is:

carbon.emission = -1278.22 + 180.95 × body.type + 95.13 × social.act + 0.89 × groc.bill + 436.68 × airtvl.freq + 0.20 × veh.distance + 135.67 × wastebag.size + 79.09 × wastebag.num + 13.87 × new.clothes + 5.59 × internet.hour - 25.77 × energy.eff + 364.03 × sex.num + 124.22 × trans.num