Car_Results

Rose Determan

9/20/2021

Description

The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

Usage

mtcars ## Format A data frame with 32 observations on 11 (numeric) variables.

[, 1] mpg Miles/(US) gallon
[, 2] cyl Number of cylinders
[, 3] disp Displacement (cu.in.)
[, 4] hp Gross horsepower
[, 5] drat Rear axle ratio
[, 6] wt Weight (1000 lbs)
[, 7] qsec 1/4 mile time
[, 8] vs Engine (0 = V-shaped, 1 = straight)
[, 9] am Transmission (0 = automatic, 1 = manual)
[,10] gear Number of forward gears
[,11] carb Number of carburetors ## Note Henderson and Velleman (1981) comment in a footnote to Table 1: 'Hocking [original transcriber]'s noncrucial coding of the Mazda's rotary engine as a straight six-cylinder engine and the Porsche's flat engine as a V engine, as well as the inclusion of the diesel Mercedes 240D, have been retained to enable direct comparisons to be made with previous analyses.'

Source

Henderson and Velleman (1981), Building multiple regression models interactively. Biometrics, 37, 391–411.

Here we import and process the data called mtcars (motor trend car road tests 1974). This dataset contains 32 observations on 11 different variables related to fuel consumption, automobile design, and performance.In this analysis we only focus on cars which have a mpg of less than 20, and we are interested in the wt, am, mpg, cyl, disp, hp, and gears variables.

```
library(tidyverse)

# call built-in data mtcars.
data(mtcars)

# Select only car models where mpg<20
mtcars_mpg2 <- mtcars[mtcars$mpg < 20,]

# Reduce the variables to mpg, cyl, disp, hp, gears
mtcars_mpg2 <- mtcars_mpg2[, c(1,2,3,4,6,9,10)]
```

Here we import a filed called "hand_functions" and use it on the mtcars dataset.
hand_functions computes the mean, variance, covariance, and correlation for/ between each column of data.
We see the beginning of the hand_functions file below.

```
# read the R file hand_functions.R so that it can be used
# notice that with echo = TRUE
source(file = "hand_functions.R", echo = TRUE)
```

```
##
## > sum_special <- function(df_x) {
## +     try(if (!is.data.frame(df_x))
## +         stop("Input data must be a data frame."))
## +     sp_means <- apply(df_ .... [TRUNCATED]
```

```
# Now use the function from hand_functions.R
sp_out <- sum_special(mtcars_mpg2)

sp_out
```

```
## $sp_means
##         mpg         cyl        disp          hp          wt          am
##  15.9000000   7.5555556 313.8111111 191.9444444   3.8388333   0.1666667
##        gear
##   3.4444444
##
## $sp_var
##          mpg          cyl         disp           hp           wt           am
##    7.5258824    0.7320261 9438.7645752 3253.5849673    0.5565416    0.1470588
##         gear
##    0.6143791
##
## $sp_cov
##               mpg         cyl        disp          hp         wt          am
## mpg     7.5258824 -1.31764706 -188.795294  -75.811765 -1.4824706  0.16470588
## cyl    -1.3176471  0.73202614   64.711111   28.444444  0.2641569 -0.03921569
## disp -188.7952941 64.71111111 9438.764575 2679.600654 54.3675314 -8.49607843
## hp    -75.8117647 28.44444444 2679.600654 3253.584967  9.0403431 11.65686275
## wt     -1.4824706  0.26415686   54.367531    9.040343  0.5565416 -0.11802941
## am      0.1647059 -0.03921569   -8.496078   11.656863 -0.1180294  0.14705882
## gear    0.6352941 -0.26143791  -34.193464   15.202614 -0.2829804  0.27450980
##            gear
## mpg    0.6352941
## cyl   -0.2614379
## disp -34.1934641
## hp    15.2026144
## wt    -0.2829804
## am     0.2745098
## gear   0.6143791
##
## $sp_cor
##             mpg        cyl       disp         hp         wt         am
## mpg   1.0000000 -0.5613802 -0.7083614 -0.4844811 -0.7243669  0.1565615
## cyl  -0.5613802  1.0000000  0.7784989  0.5828450  0.4138565 -0.1195229
## disp -0.7083614  0.7784989  1.0000000  0.4835389  0.7501243 -0.2280423
## hp   -0.4844811  0.5828450  0.4835389  1.0000000  0.2124492  0.5329115
## wt   -0.7243669  0.4138565  0.7501243  0.2124492  1.0000000 -0.4125683
```

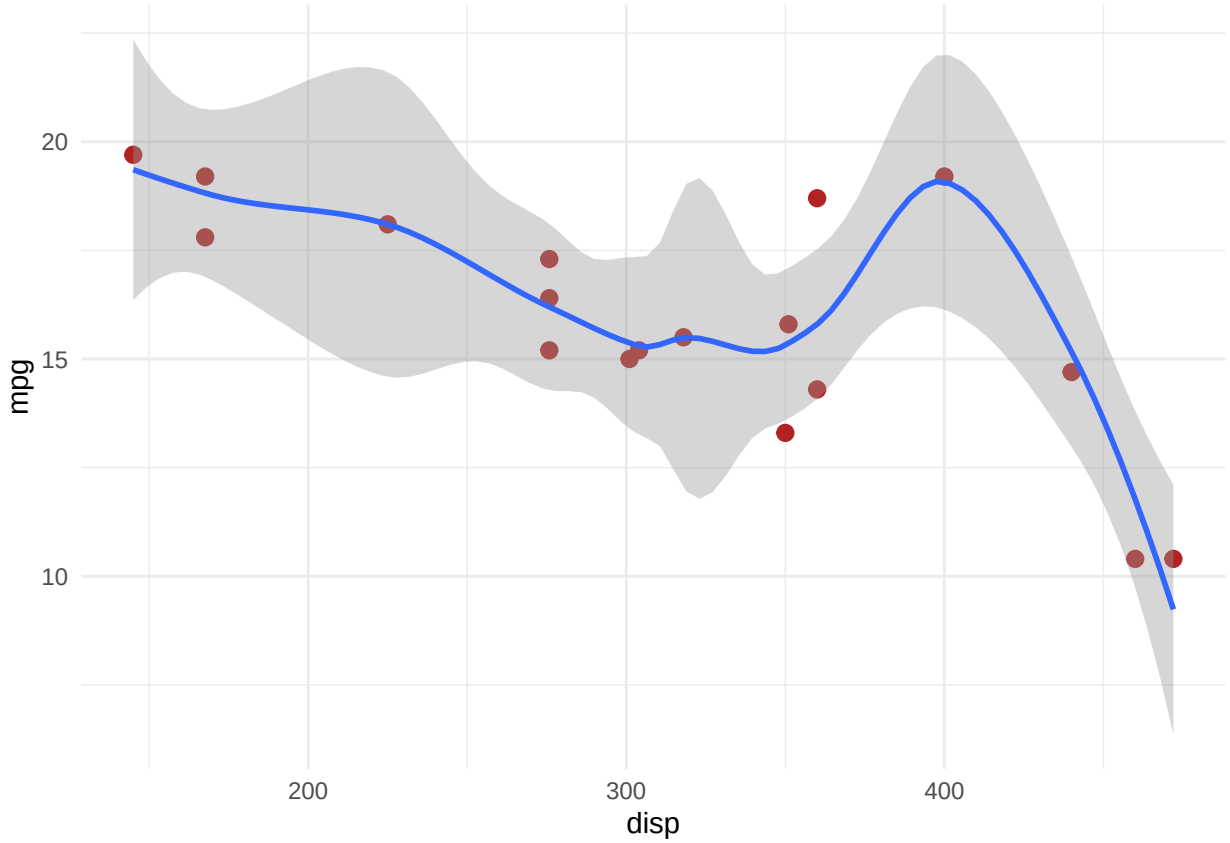
```
## am    0.1565615 -0.1195229 -0.2280423  0.5329115 -0.4125683  1.0000000
## gear  0.2954459 -0.3898406 -0.4490217  0.3400314 -0.4839376  0.9132593
##            gear
## mpg   0.2954459
## cyl  -0.3898406
## disp -0.4490217
## hp    0.3400314
## wt   -0.4839376
## am    0.9132593
## gear  1.0000000
```

Below is an optional chunk of code that allows you to view the dataset in a browser and explore ggplots of the data in an interactive manner.
The esquisse interactive page will provide you with ggplot2 code that you can paste and use in an R markdown file.

```
# library(esquisse)
#
# esquisser(data = mtcars_mpg2, viewer = "browser")
```

Plot of displacement vs mpg.
Each red point represents a datapoint from mtcars. The gray shaded area
represents a smoothed function that is calculated using geom_smooth's default
loess with a span of 0.5.
As shown in the plot, there is not a strong linear signal between displacement and mpg.

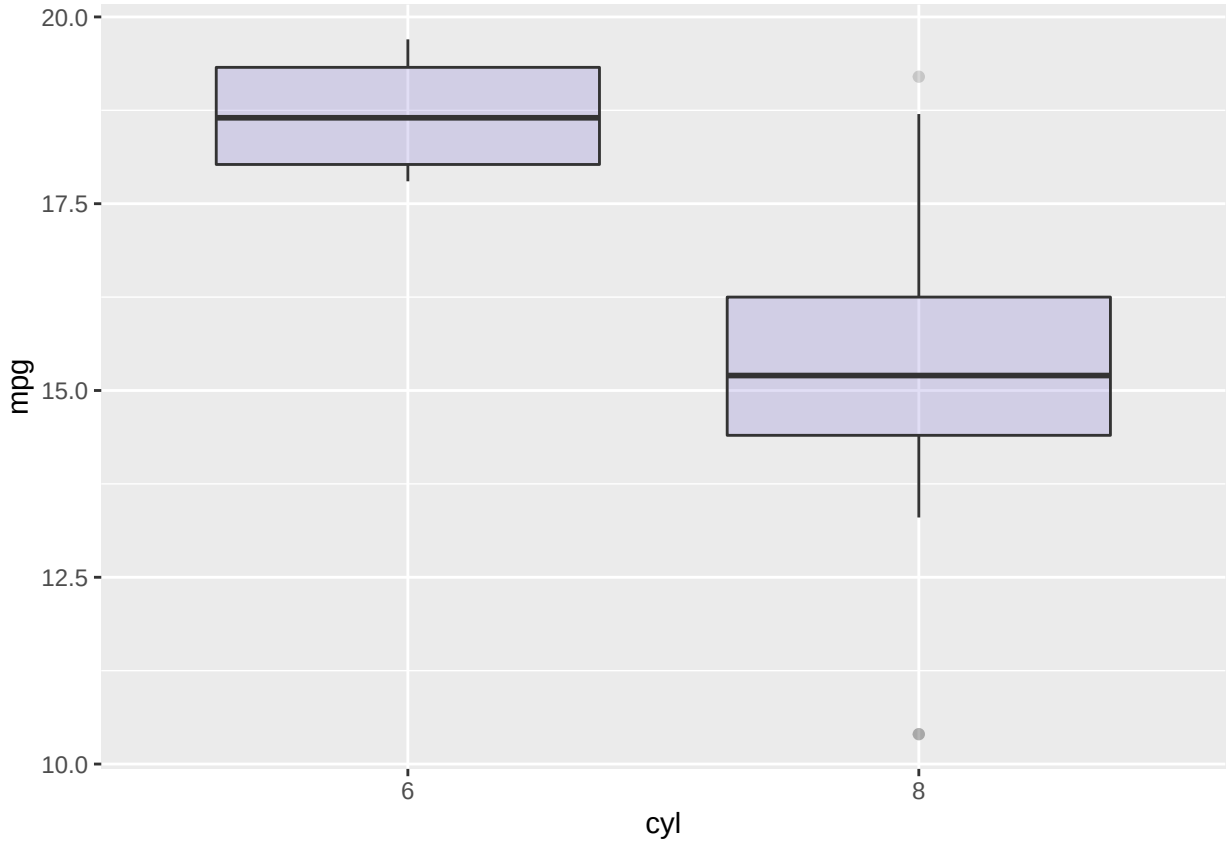
```
ggplot(mtcars_mpg2) +
  aes(x = disp, y = mpg) +
  geom_point(shape = "bullet", size = 4L, colour = "#B22222") +
  geom_smooth(span = 0.5, method = 'loess', formula = y~x) +
  theme_minimal()
```



Below, is a boxplot that compares the number of cylinders as a factor to mpg. A the bold line in the middle of each boxplot represents the median, while the lines above and below represent the 75th and 25th percentiles of the data. Here, we can see that a car with 6 cylinders has a higher median mpg than a car with 8 cylinders.

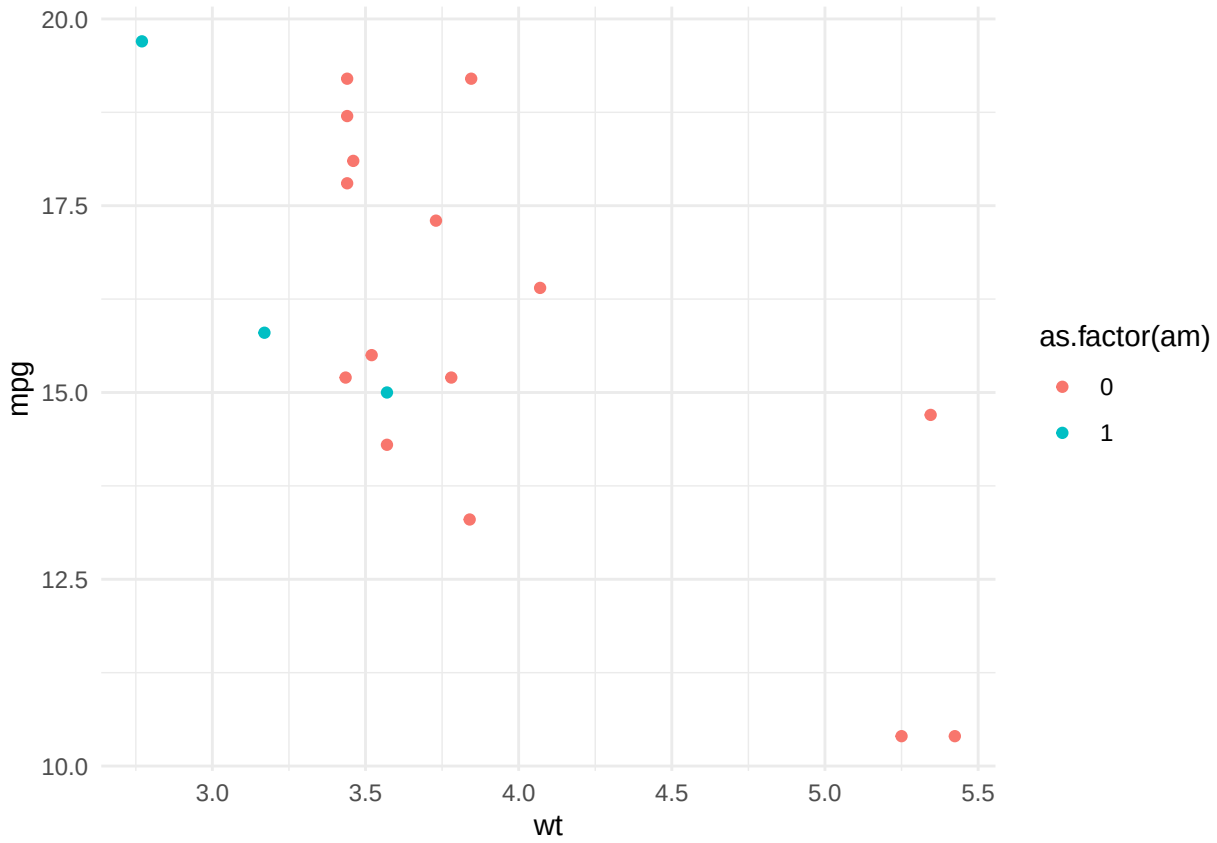
Note that this boxplot cannot be made with esquisse() unless the data is adjusted. In the dataframe, the cylinder column is stored as an integer, but for a boxplot the data must be stored as a factor.

```
ggplot(mtcars_mpg2, aes(x=as.factor(cyl), y=mpg)) +
  geom_boxplot(fill="slateblue", alpha=0.2) +
  xlab("cyl")
```



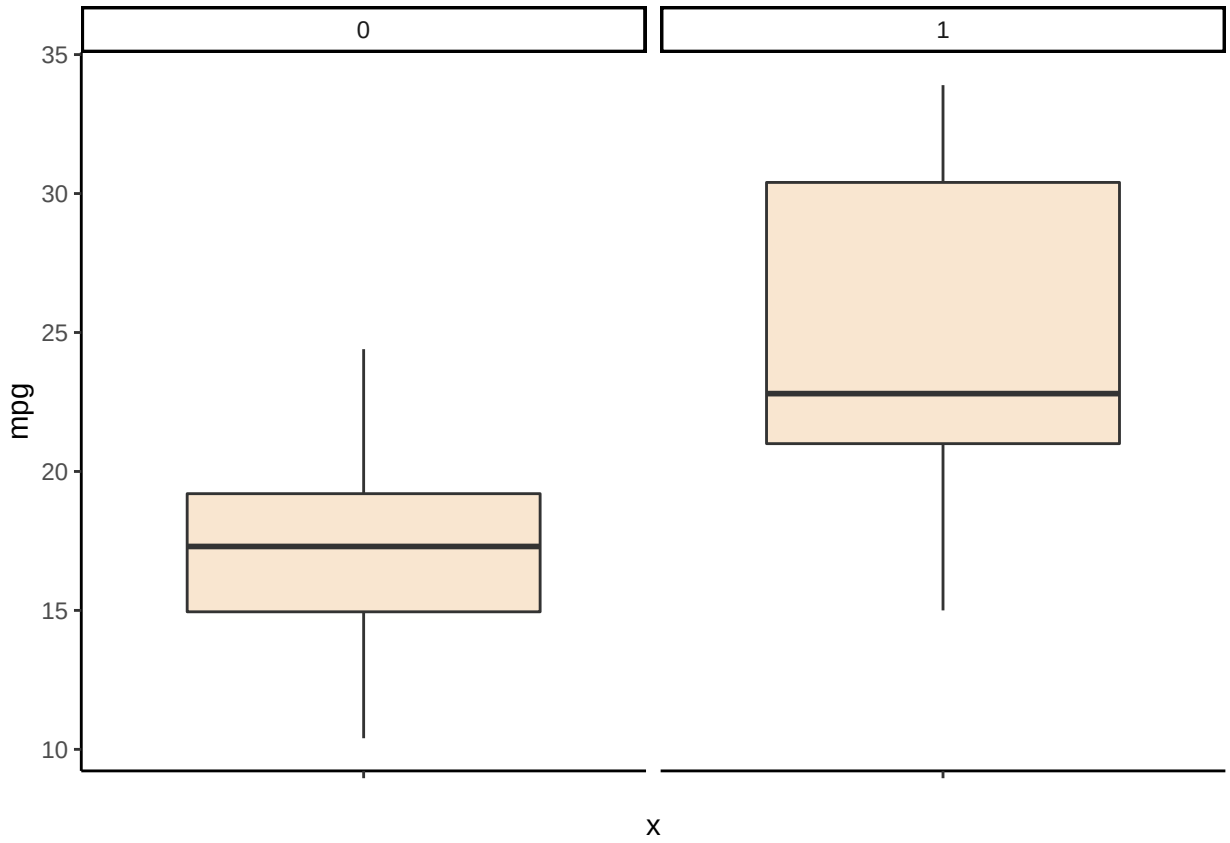
Below is a scatterplot that compares car weight and mpg. The plot suggests that heavier cars have lower mpg and lighter cars have a higher mpg. Additionally the data are color coded. Blue represents manual transmission vehicles and red represents automatic transmission vehicles. Although there are only three manual vehicles, they seem to have higher mpg and lower weight than their automatic counterparts.

```
#weight vs mpg; color = automatic/manual
ggplot(mtcars_mpg2) +
  aes(x = wt, y = mpg, colour = as.factor(am)) +
  geom_point(shape = "circle", size = 1.5) +
  scale_colour_discrete() +
  theme_minimal()
```



Below, we examine all of the mtcar data, including both high and low mpg vehicles. Again the data are separated by manual transmission (1) and automatic transmission (0). This plot further supports the claim that manual vehicles have a higher mpg than automatic ones.

```
#mpg; separated by automatic/manual
ggplot(mtcars) +
  aes(x = "", y = mpg) +
  geom_boxplot(shape = "circle filled", fill = "#F9E6D0") +
  theme_classic() +
  facet_wrap(vars(am))
```


0
1
35
30
25
20
15
10
mpg
x