Assignment_1_Jessie Xu

Jessie Xu

2021/9/22

```
library(tidyverse)
```

```
## -- Attaching packages -------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5      v purrr   0.3.4
## v tibble  3.1.4      v dplyr   1.0.7
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   2.0.1      v forcats 0.5.1
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

(1)After install the package once, we need to use library() to invoke the relative functions everytime we start our R.

(2)Using packman::p_load() can replace install.packages("A","B") and library(A,B) at the same time.

```
# 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(1,2,3,4,10)]
```

Pay attention to the expression of the questions, there are so many time that I read the question in reverse, like "reduce..to.." actually means reserve the things after "to".

```
# 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       gear
##   15.900000  7.555556 313.811111 191.944444   3.444444
```

```
## 
## $sp_var
##         mpg         cyl        disp          hp        gear
##   7.5258824   0.7320261 9438.7645752 3253.5849673   0.6143791
## 
## $sp_cov
##             mpg         cyl       disp         hp       gear
## mpg    7.5258824  -1.3176471 -188.79529  -75.81176  0.6352941
## cyl   -1.3176471   0.7320261   64.71111   28.44444 -0.2614379
## disp -188.7952941 64.7111111 9438.76458 2679.60065 -34.1934641
## hp    -75.8117647 28.4444444 2679.60065 3253.58497  15.2026144
## gear    0.6352941 -0.2614379  -34.19346   15.20261   0.6143791
## 
## $sp_cor
##             mpg         cyl       disp         hp       gear
## mpg    1.0000000 -0.5613802 -0.7083614 -0.4844811  0.2954459
## cyl   -0.5613802  1.0000000  0.7784989  0.5828450 -0.3898406
## disp -0.7083614  0.7784989  1.0000000  0.4835389 -0.4490217
## hp    -0.4844811  0.5828450  0.4835389  1.0000000  0.3400314
## gear   0.2954459 -0.3898406 -0.4490217  0.3400314  1.0000000
```

(1)"echo=TRUE" means that all setted code chunks will be included in the final rendered version.

(2)Using the new "sum_special()" function, we can get the value of the mean, the var, the cov and the cor of our target variables.

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

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```
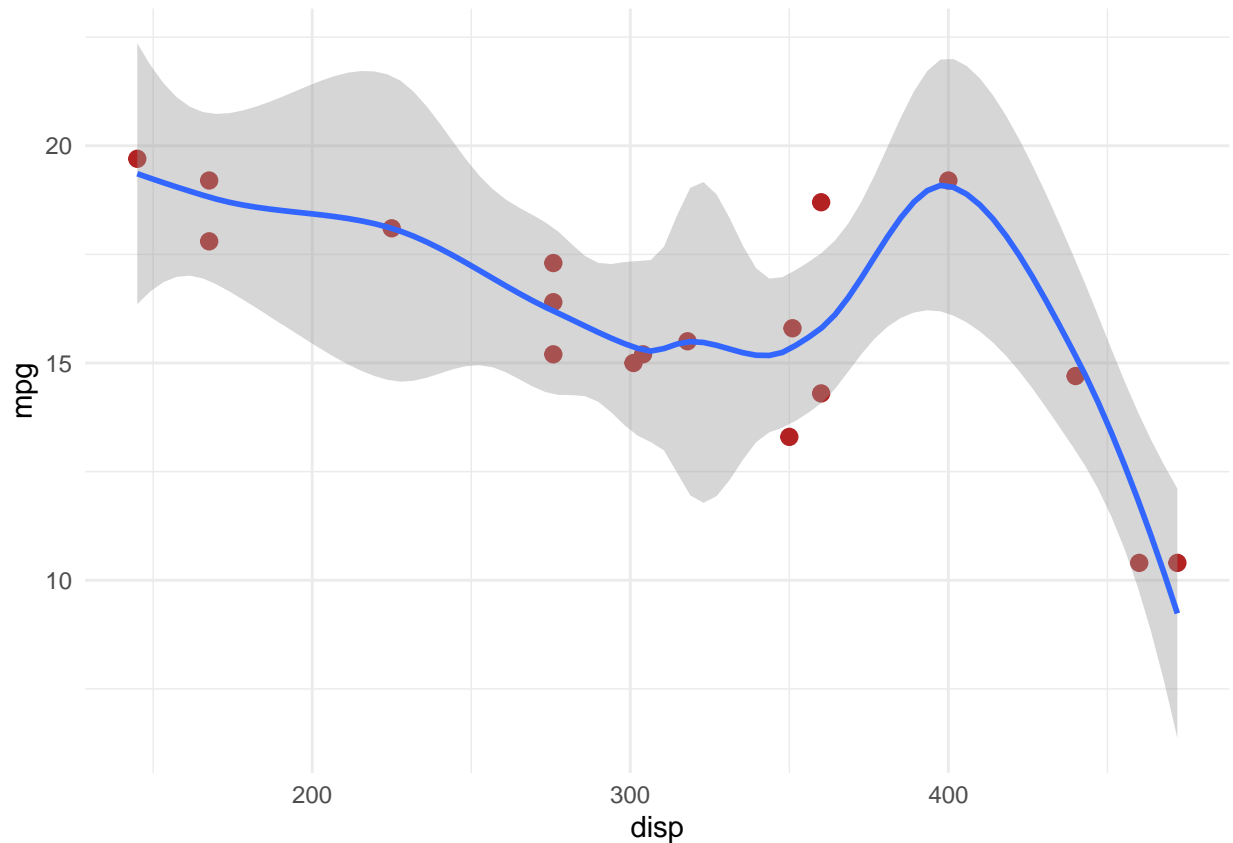
(1)"#B22222" is a great dark red shade.

(2)"span" controls the amount of smoothing for the default loess smoother. The larger the span the smoother the plot.

(3)The shade aroung the line means the confidence bands of the data.

(4)theme_minimal() means no context like no black grid lines and no dark gray color of the background.

```
# draw a boxplot by using geom_boxplot()
ggplot(mtcars_mpg2, aes(x=as.factor(cyl), y=mpg)) +
  geom_boxplot(fill="slateblue", alpha=0.2) +
  xlab("cyl")
```
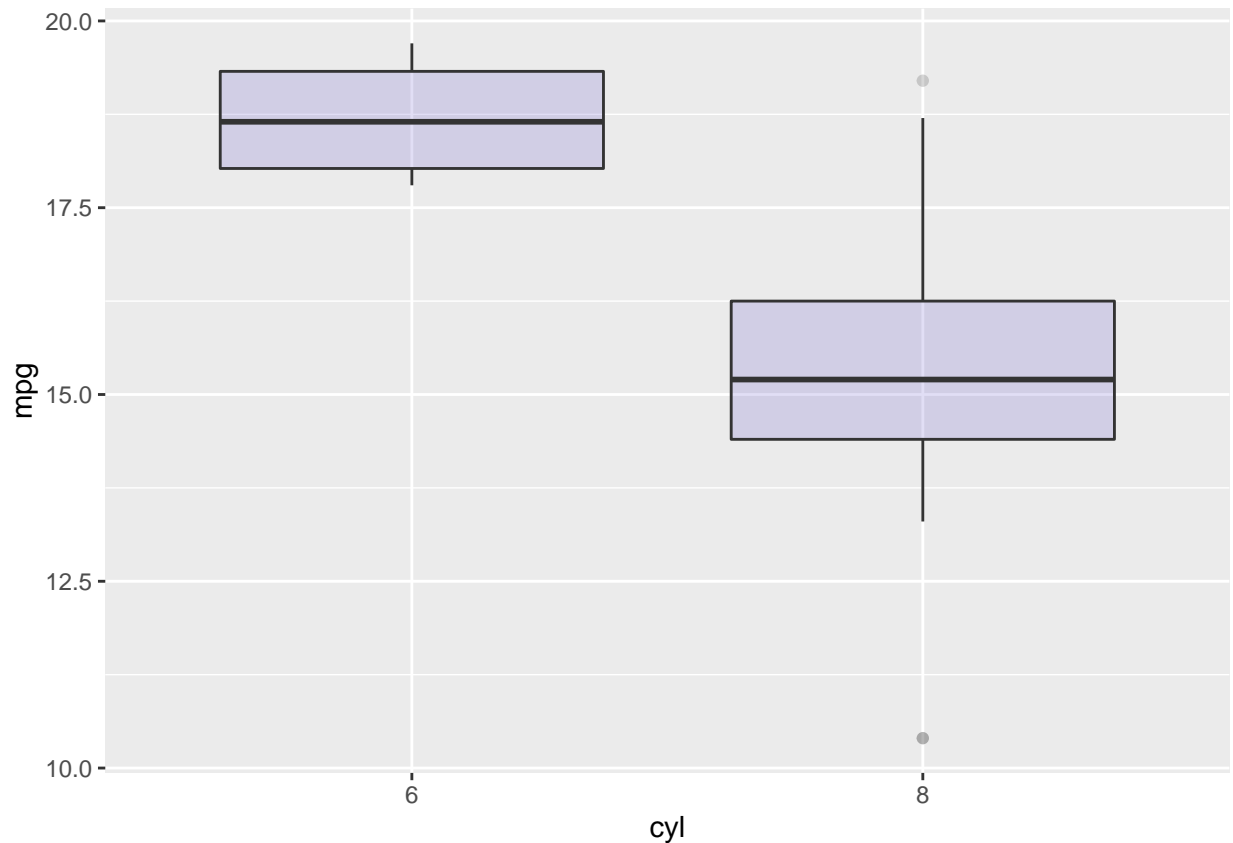
```
# install.packages("esquisse")
# library(esquisse)
# esquisser(data = mtcars_mpg2, viewer = "browser")

# note that this boxplot cannot be made with esquisse() unless
# the data is adjusted.  What adjustment is needed?
```

By running the code of line 58-60, I noticed that the presence or absence of "as.factor()" function can affect the graph we plot. So I try to generate a new colunm which add cyl as a classification variable.

```
mtcars_mpg2$factor_cyl <- as.factor(mtcars_mpg2$cyl)
# esquisser(data = mtcars_mpg2, viewer = "browser")
```

Just drag "factor_cyl" into the X box and drag "mpg" in to y box, we can plot this boxplot made with esquisse() successfully.