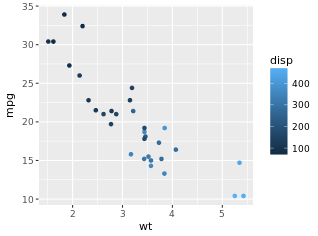
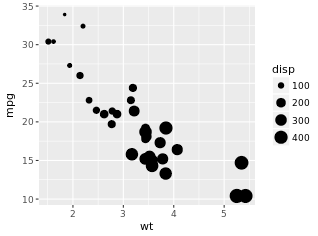
ggplot(mtcars, aes(x = wt, y = mpg, color = disp)) + geom\_point()



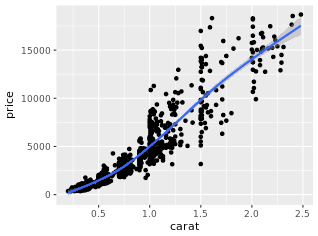
ggplot(mtcars, aes(x = wt, y = mpg, size = disp)) + geom\_point()



Shape instead of size for categorial variables

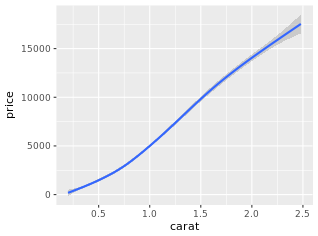
# 1 - The plot you created in the previous exercise

ggplot(diamonds, aes(x = carat, y = price)) + geom\_point() + geom\_smooth()



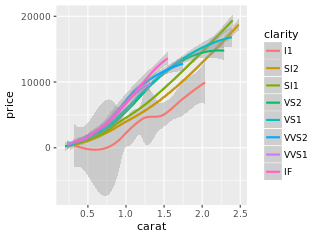
# 2 - Copy the above command but show only the smooth line

ggplot(diamonds, aes(x = carat, y = price)) + geom\_smooth()



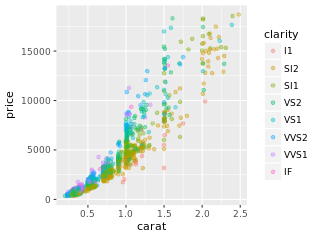
# 3 - Copy the above command and assign the correct value to col in aes()

ggplot(diamonds, aes(x = carat, y = price, color = clarity)) + geom\_smooth()



# 4 - Keep the color settings from previous command. Plot only the points with argument alpha

ggplot(diamonds, aes(x = carat, y = price, color = clarity)) + geom\_point(alpha = 0.4)

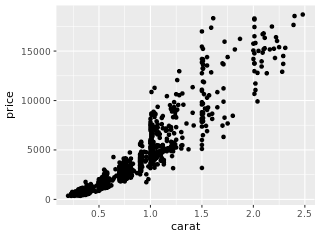


# Create the object containing the data and aes layers: dia\_plot

dia\_plot <- ggplot(diamonds, aes(x = carat, y =price))

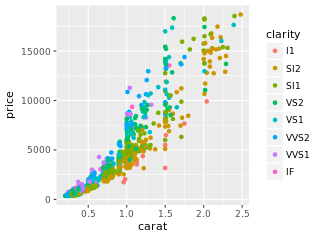
# Add a geom layer with + and geom\_point()

dia\_plot + geom\_point()



# Add the same geom layer, but with aes() inside

dia\_plot + geom\_point(aes(color = clarity))



# 1 - The dia\_plot object has been created for you

dia\_plot <- ggplot(diamonds, aes(x = carat, y = price))

# 2 - Expand dia\_plot by adding geom\_point() with alpha set to 0.2

dia\_plot <- dia\_plot + geom\_point(alpha = 0.2)

# 3 - Plot dia\_plot with additional geom\_smooth() with se set to FALSE

dia\_plot + geom\_smooth(se =FALSE)

# 4 - Copy the command from above and add aes() with the correct mapping to geom\_smooth()

dia\_plot + geom\_smooth(aes(col =clarity), se =FALSE)

ggplot(mammals, aes(x = body, y = brain)) +

geom\_point(alpha = 0.6) + stat\_smooth(method = "lm", col =

"red", se = FALSE)

#Green spaces

library(geometry)

library(measurements)

x = read.table(text = " lat long

105252 30°25.264 9°01.331

105253 30°39.237 8°10.811

105255 31°37.760 8°06.040

105258 31°41.190 8°06.557

105259 31°41.229 8°06.622

105260 31°38.891 8°06.281",

header = TRUE, stringsAsFactors = FALSE)

# change the degree symbol to a space

x$lat = gsub('°', ' ', x$lat)

x$long = gsub('°', ' ', x$long)

# convert from decimal minutes to decimal degrees

x$lat = measurements::conv\_unit(x$lat, from = 'deg\_dec\_min', to = 'dec\_deg')

x$long = measurements::conv\_unit(x$long, from = 'deg\_dec\_min', to = 'dec\_deg')