RWorksheet_urdas#6

Cindy Urdas

2022-12-05

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```
data(mpg)
as.data.frame(data(mpg))
```

```
##   data(mpg)
## 1       mpg
```

```
data(mpg)
mpg
```

```
## # A tibble: 234 x 11
##    manufacturer model      displ  year   cyl trans drv     cty   hwy fl    class
##    <chr>        <chr>      <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>
##  1 audi         a4           1.8  1999     4 auto~ f        18    29 p     comp~
##  2 audi         a4           1.8  1999     4 manu~ f        21    29 p     comp~
##  3 audi         a4           2    2008     4 manu~ f        20    31 p     comp~
##  4 audi         a4           2    2008     4 auto~ f        21    30 p     comp~
##  5 audi         a4           2.8  1999     6 auto~ f        16    26 p     comp~
##  6 audi         a4           2.8  1999     6 manu~ f        18    26 p     comp~
##  7 audi         a4           3.1  2008     6 auto~ f        18    27 p     comp~
##  8 audi         a4 quattro   1.8  1999     4 manu~ 4        18    26 p     comp~
##  9 audi         a4 quattro   1.8  1999     4 auto~ 4        16    25 p     comp~
## 10 audi         a4 quattro   2    2008     4 manu~ 4        20    28 p     comp~
## # ... with 224 more rows
```

```
str(mpg)
```

```
## tibble [234 x 11] (S3: tbl_df/tbl/data.frame)
##  $ manufacturer: chr [1:234] "audi" "audi" "audi" "audi" ...
##  $ model       : chr [1:234] "a4" "a4" "a4" "a4" ...
##  $ displ       : num [1:234] 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int [1:234] 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : int [1:234] 4 4 4 4 6 6 6 4 4 4 ...
##  $ trans       : chr [1:234] "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
##  $ drv         : chr [1:234] "f" "f" "f" "f" ...
##  $ cty         : int [1:234] 18 21 20 21 16 18 18 18 16 20 ...
```

```
##  $ hwy          : int [1:234] 29 29 31 30 26 26 27 26 25 28 ...
##  $ fl           : chr [1:234] "p" "p" "p" "p" ...
##  $ class        : chr [1:234] "compact" "compact" "compact" "compact" ...
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.2
```

```
##
## 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
```

```
glimpse(mpg)
```

```
## Rows: 234
## Columns: 11
## $ manufacturer <chr> "audi", "audi", "audi", "audi", "audi", "audi", "audi", "~
## $ model        <chr> "a4", "a4", "a4", "a4", "a4", "a4", "a4", "a4 quattro", "~
## $ displ        <dbl> 1.8, 1.8, 2.0, 2.0, 2.8, 2.8, 3.1, 1.8, 1.8, 2.0, 2.0, 2.~
## $ year         <int> 1999, 1999, 2008, 2008, 1999, 1999, 2008, 1999, 1999, 200~
## $ cyl          <int> 4, 4, 4, 4, 6, 6, 6, 4, 4, 4, 4, 6, 6, 6, 6, 6, 6, 8, 8, ~
## $ trans        <chr> "auto(l5)", "manual(m5)", "manual(m6)", "auto(av)", "auto~
## $ drv          <chr> "f", "f", "f", "f", "f", "f", "f", "4", "4", "4", "4", "4~
## $ cty          <int> 18, 21, 20, 21, 16, 18, 18, 18, 16, 20, 19, 15, 17, 17, 1~
## $ hwy          <int> 29, 29, 31, 30, 26, 26, 27, 26, 25, 28, 27, 25, 25, 25, 2~
## $ fl           <chr> "p", "p", "p", "p", "p", "p", "p", "p", "p", "p", "p", "p~
## $ class        <chr> "compact", "compact", "compact", "compact", "compact", "c~
```

1. How many columns are in mpg dataset? How about the number of rows? Show the codes and its result.

```
datampg <- glimpse(mpg)
```

```
## Rows: 234
## Columns: 11
## $ manufacturer <chr> "audi", "audi", "audi", "audi", "audi", "audi", "audi", "~
## $ model        <chr> "a4", "a4", "a4", "a4", "a4", "a4", "a4", "a4 quattro", "~
## $ displ        <dbl> 1.8, 1.8, 2.0, 2.0, 2.8, 2.8, 3.1, 1.8, 1.8, 2.0, 2.0, 2.~
## $ year         <int> 1999, 1999, 2008, 2008, 1999, 1999, 2008, 1999, 1999, 200~
## $ cyl          <int> 4, 4, 4, 4, 6, 6, 6, 4, 4, 4, 4, 6, 6, 6, 6, 6, 6, 8, 8, ~
## $ trans        <chr> "auto(l5)", "manual(m5)", "manual(m6)", "auto(av)", "auto~
## $ drv          <chr> "f", "f", "f", "f", "f", "f", "f", "4", "4", "4", "4", "4~
## $ cty          <int> 18, 21, 20, 21, 16, 18, 18, 18, 16, 20, 19, 15, 17, 17, 1~
## $ hwy          <int> 29, 29, 31, 30, 26, 26, 27, 26, 25, 28, 27, 25, 25, 25, 2~
## $ fl           <chr> "p", "p", "p", "p", "p", "p", "p", "p", "p", "p", "p", "p~
## $ class        <chr> "compact", "compact", "compact", "compact", "compact", "c~
```

```
nrow(mpg)
```

```
## [1] 234
```

```
ncol(mpg)
```

```
## [1] 11
```

The number of columns in the mpg dataset is 11, while the number of rows is 234.

2. Which manufacturer has the most models in this data set? Which model has the most variations?

```
num2 <- mpg %>%
  group_by(manufacturer) %>%
  tally(sort = TRUE)
num2
```

```
## # A tibble: 15 x 2
##    manufacturer     n
##    <chr>        <int>
##  1 dodge           37
##  2 toyota          34
##  3 volkswagen      27
##  4 ford            25
##  5 chevrolet       19
##  6 audi            18
##  7 hyundai         14
##  8 subaru          14
##  9 nissan          13
## 10 honda            9
## 11 jeep             8
## 12 pontiac          5
## 13 land rover       4
## 14 mercury          4
## 15 lincoln          3
```

```
mpg$manu_mod <- paste(mpg$manufacturer,mpg$model)
unique(mpg$manu_mod)
```

```
##  [1] "audi a4"                       "audi a4 quattro"
##  [3] "audi a6 quattro"               "chevrolet c1500 suburban 2wd"
##  [5] "chevrolet corvette"            "chevrolet k1500 tahoe 4wd"
##  [7] "chevrolet malibu"              "dodge caravan 2wd"
##  [9] "dodge dakota pickup 4wd"       "dodge durango 4wd"
## [11] "dodge ram 1500 pickup 4wd"     "ford expedition 2wd"
## [13] "ford explorer 4wd"             "ford f150 pickup 4wd"
## [15] "ford mustang"                  "honda civic"
## [17] "hyundai sonata"                "hyundai tiburon"
```

```
## [19] "jeep grand cherokee 4wd"       "land rover range rover"
## [21] "lincoln navigator 2wd"         "mercury mountaineer 4wd"
## [23] "nissan altima"                 "nissan maxima"
## [25] "nissan pathfinder 4wd"         "pontiac grand prix"
## [27] "subaru forester awd"           "subaru impreza awd"
## [29] "toyota 4runner 4wd"            "toyota camry"
## [31] "toyota camry solara"           "toyota corolla"
## [33] "toyota land cruiser wagon 4wd" "toyota toyota tacoma 4wd"
## [35] "volkswagen gti"                "volkswagen jetta"
## [37] "volkswagen new beetle"         "volkswagen passat"
```

In this dataset, Dodge appears the most often (37 times), whereas Toyota has the most variety (5 different model types).

a. Group the manufacturers and find the unique models. Copy the codes and result.

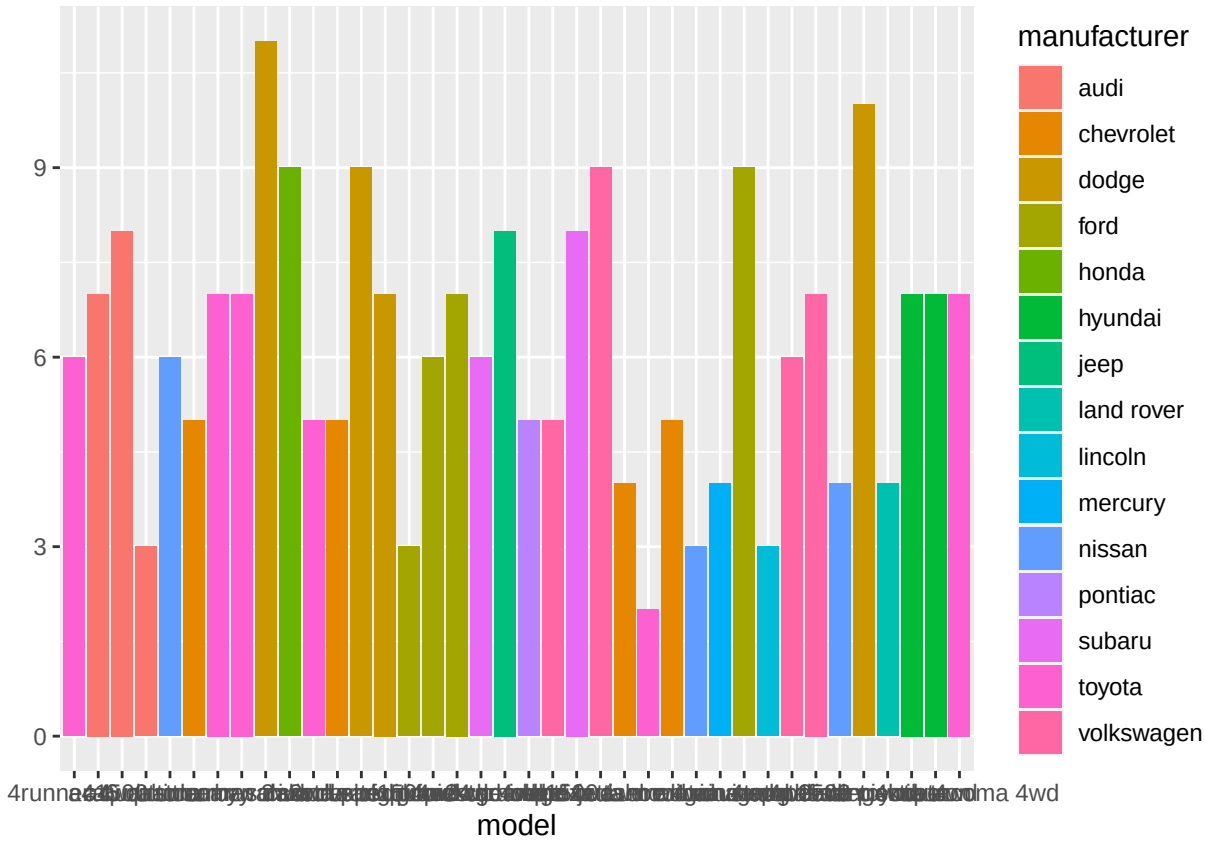
```
datampg <- mpg
uniq_mod <- datampg %>% group_by(manufacturer, model) %>%
distinct() %>% count()
uniq_mod
```

```
## # A tibble: 38 x 3
## # Groups:   manufacturer, model [38]
##    manufacturer model                  n
##    <chr>        <chr>              <int>
##  1 audi         a4                     7
##  2 audi         a4 quattro             8
##  3 audi         a6 quattro             3
##  4 chevrolet    c1500 suburban 2wd     4
##  5 chevrolet    corvette               5
##  6 chevrolet    k1500 tahoe 4wd        4
##  7 chevrolet    malibu                 5
##  8 dodge        caravan 2wd            9
##  9 dodge        dakota pickup 4wd      8
## 10 dodge        durango 4wd            6
## # ... with 28 more rows
```

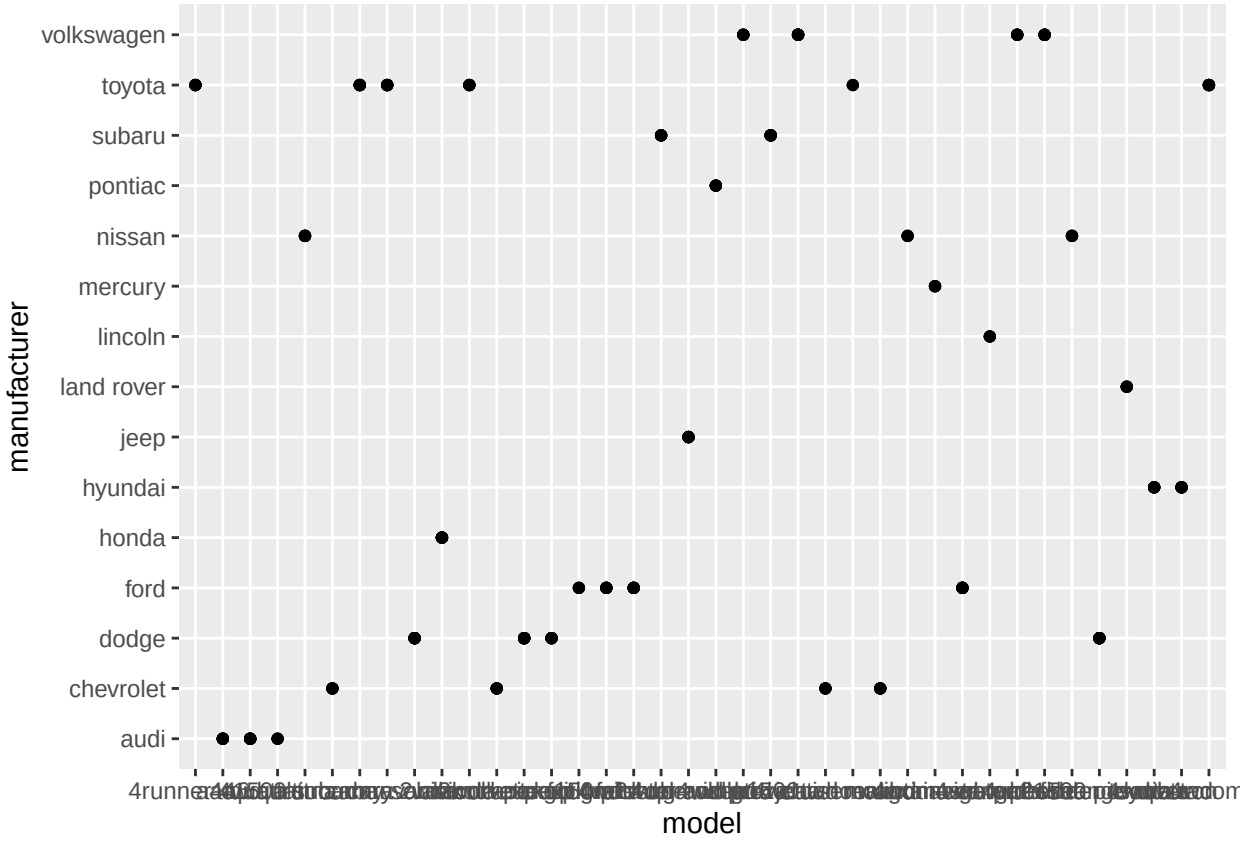
b. Graph the result by using plot() and ggplot(). Write the codes and its result.

plot()

```
## Warning: `qplot()` was deprecated in ggplot2 3.4.0.
```

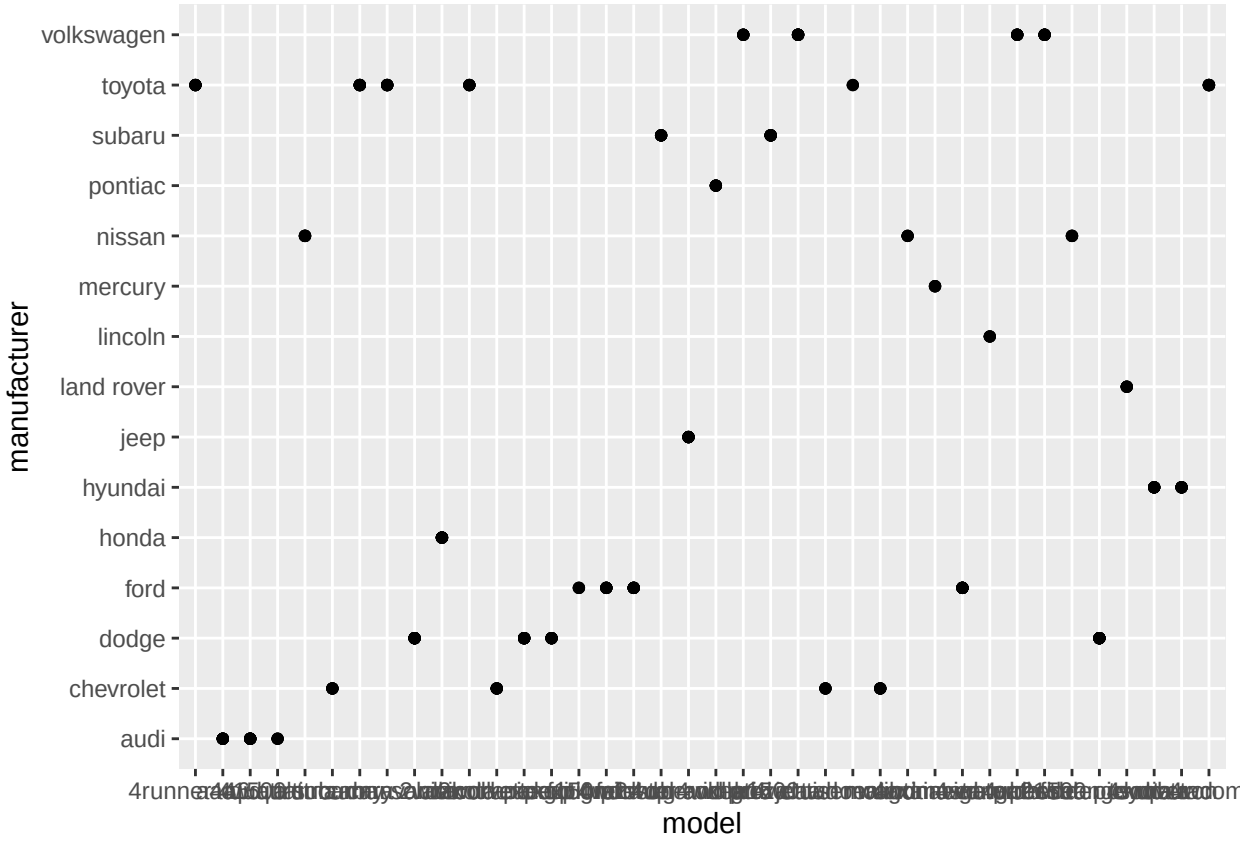

manufacturer
audi
chevrolet
dodge
ford
honda
hyundai
jeep
land rover
lincoln
mercury
nissan
pontiac
subaru
toyota
volkswagen
9
6
3
0
4runner 4wd
model

ggplot()

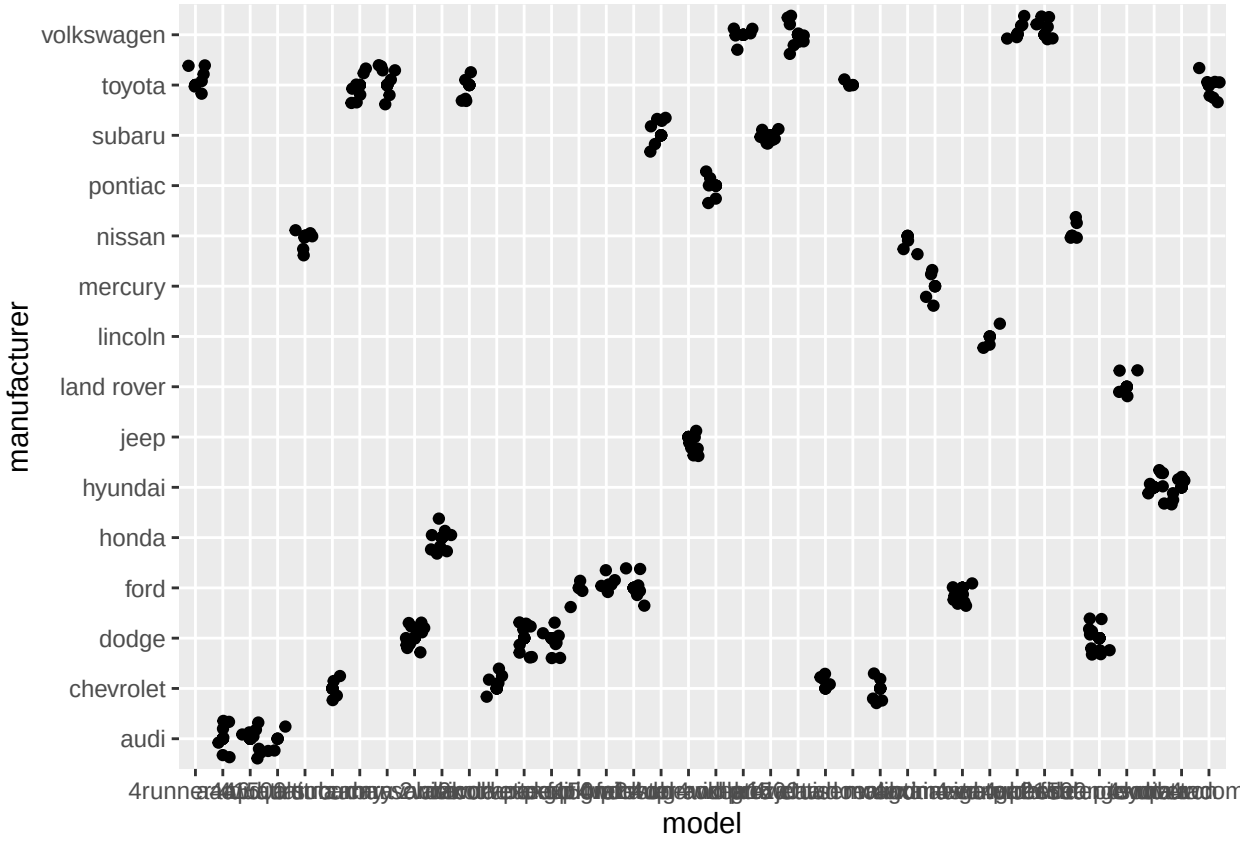


3. Same dataset will be used.You are going to show the relationship of the model and the manufacturer.

a. What does ggplot(mpg, aes(model, manufacturer)) + geom_point() show?



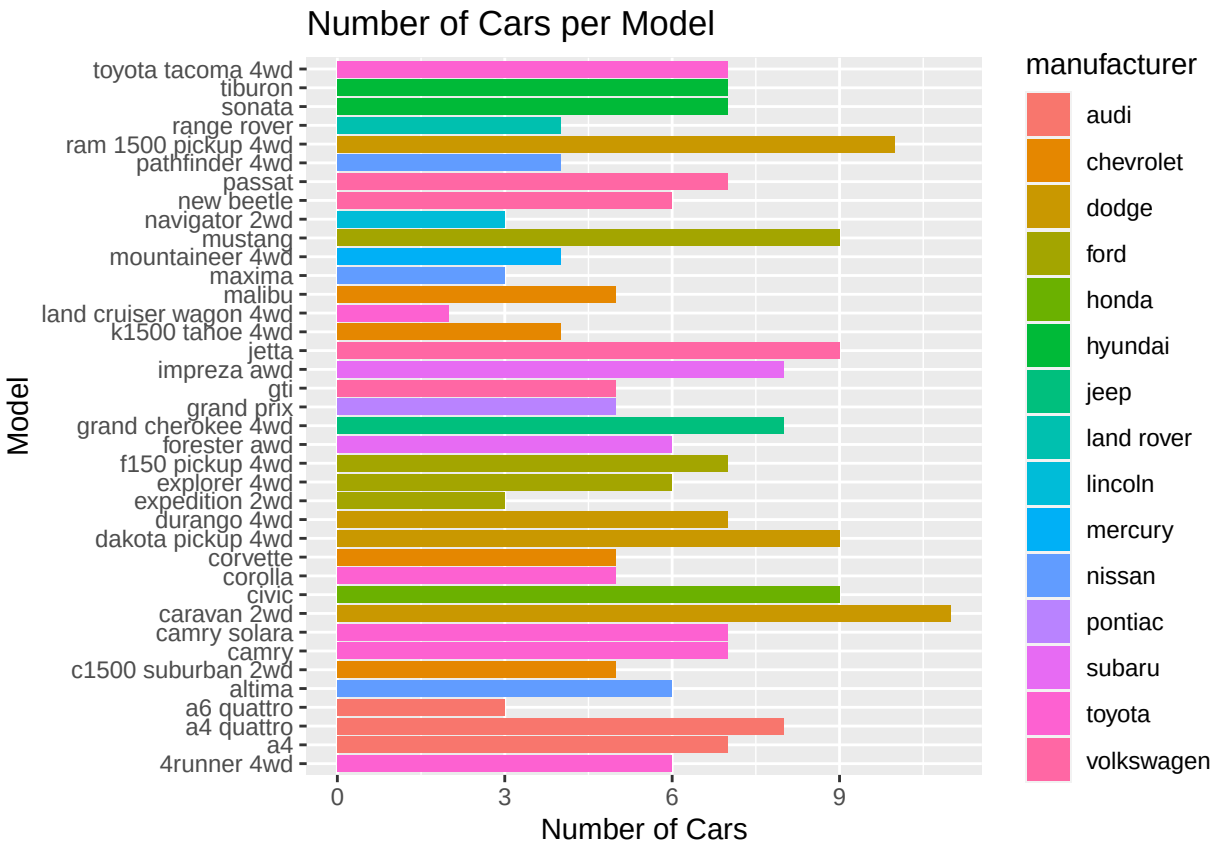
b. For you, is it useful? If not, how could you modify the data to make it more informative?



4. Using the pipe (%>%), group the model and get the number of cars per model. Show codes and its result.

```
## # A tibble: 38 x 2
## # Groups:   model [38]
##    model                  n
##    <chr>              <int>
##  1 4runner 4wd            1
##  2 a4                     1
##  3 a4 quattro             1
##  4 a6 quattro             1
##  5 altima                 1
##  6 c1500 suburban 2wd     1
##  7 camry                  1
##  8 camry solara           1
##  9 caravan 2wd            1
## 10 civic                  1
## # ... with 28 more rows
```

a. Plot using the geom_bar() + coord_flip() just like what is shown below. Show codes and its result.



b. Use only the top 20 observations. Show code and results.

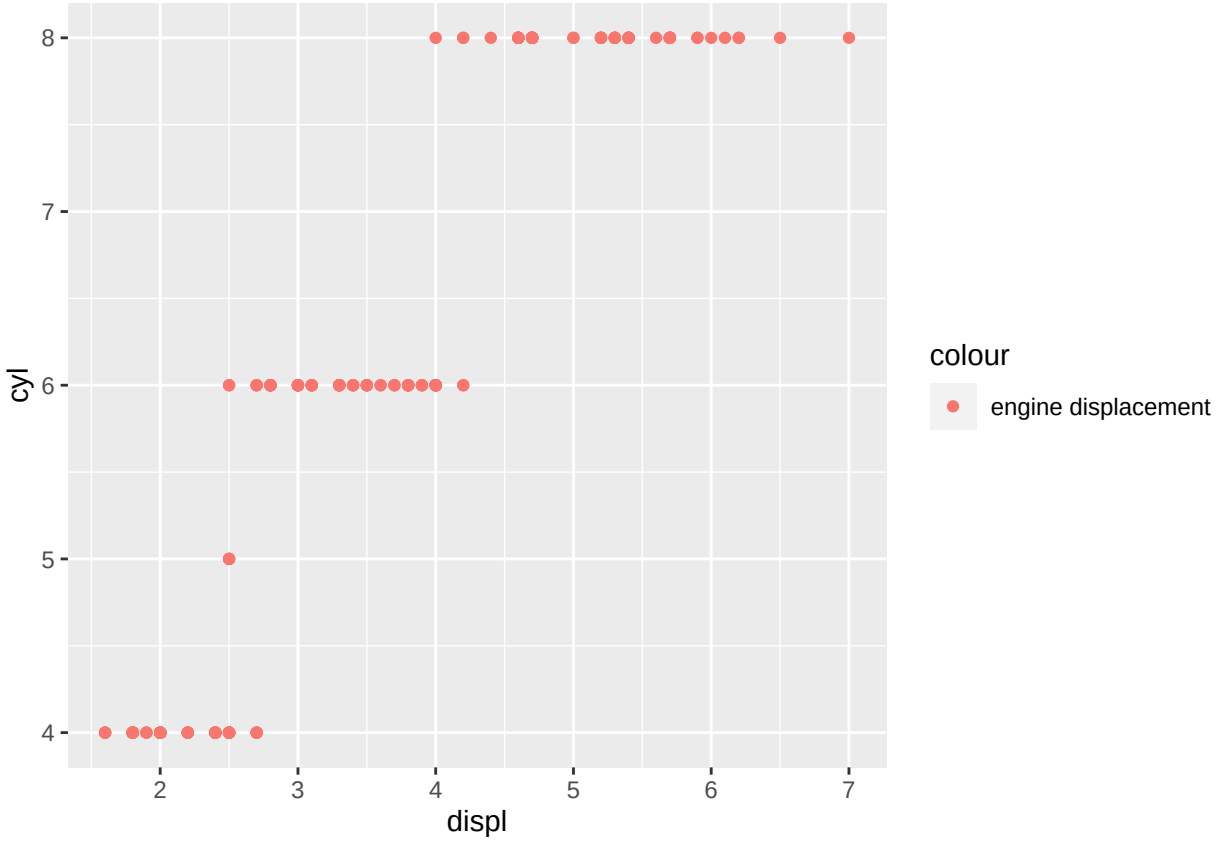
```
car_mods <- mpg %>%
  group_by(model) %>%
  tally(sort = TRUE)
car_mods %>% print(n = 20)
```

```
## # A tibble: 38 x 2
##    model                  n
##    <chr>              <int>
##  1 caravan 2wd           11
##  2 ram 1500 pickup 4wd   10
##  3 civic                  9
##  4 dakota pickup 4wd      9
##  5 jetta                  9
##  6 mustang                9
##  7 a4 quattro             8
##  8 grand cherokee 4wd     8
##  9 impreza awd            8
## 10 a4                     7
## 11 camry                  7
## 12 camry solara           7
## 13 durango 4wd            7
## 14 f150 pickup 4wd        7
## 15 passat                 7
```

```
## 16 sonata                  7
## 17 tiburon                 7
## 18 toyota tacoma 4wd       7
## 19 4runner 4wd             6
## 20 altima                  6
## # ... with 18 more rows
```

5. Plot the relationship between cyl - number of cylinders and displ - engine displace- ment using geom_point with aesthetic colour = engine displacement. Title should be "Relationship between No. of Cylinders and Engine Displacement".

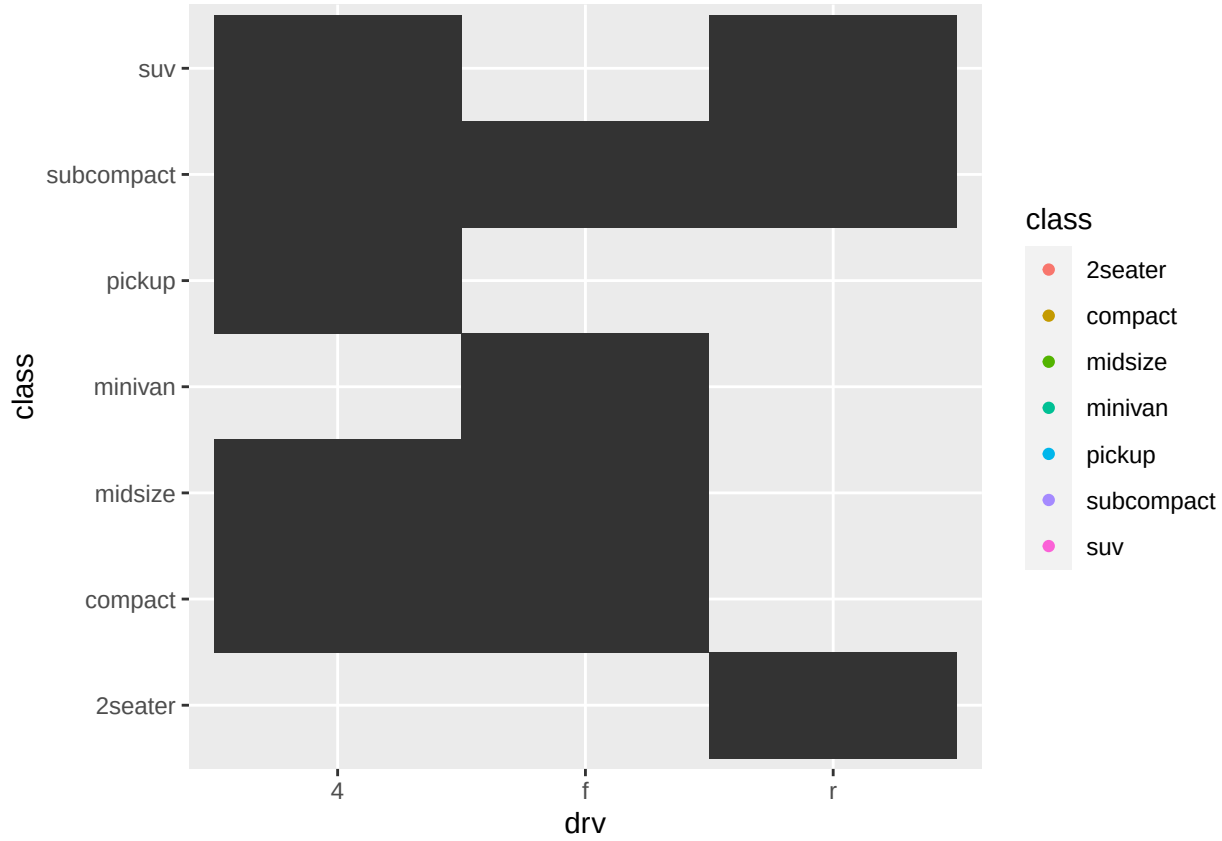
a. Show the codes and its result.



b. How would you describe its relationship?
While more cylinders widen and flatten the power band, displacement predicts and limits maximum power output.

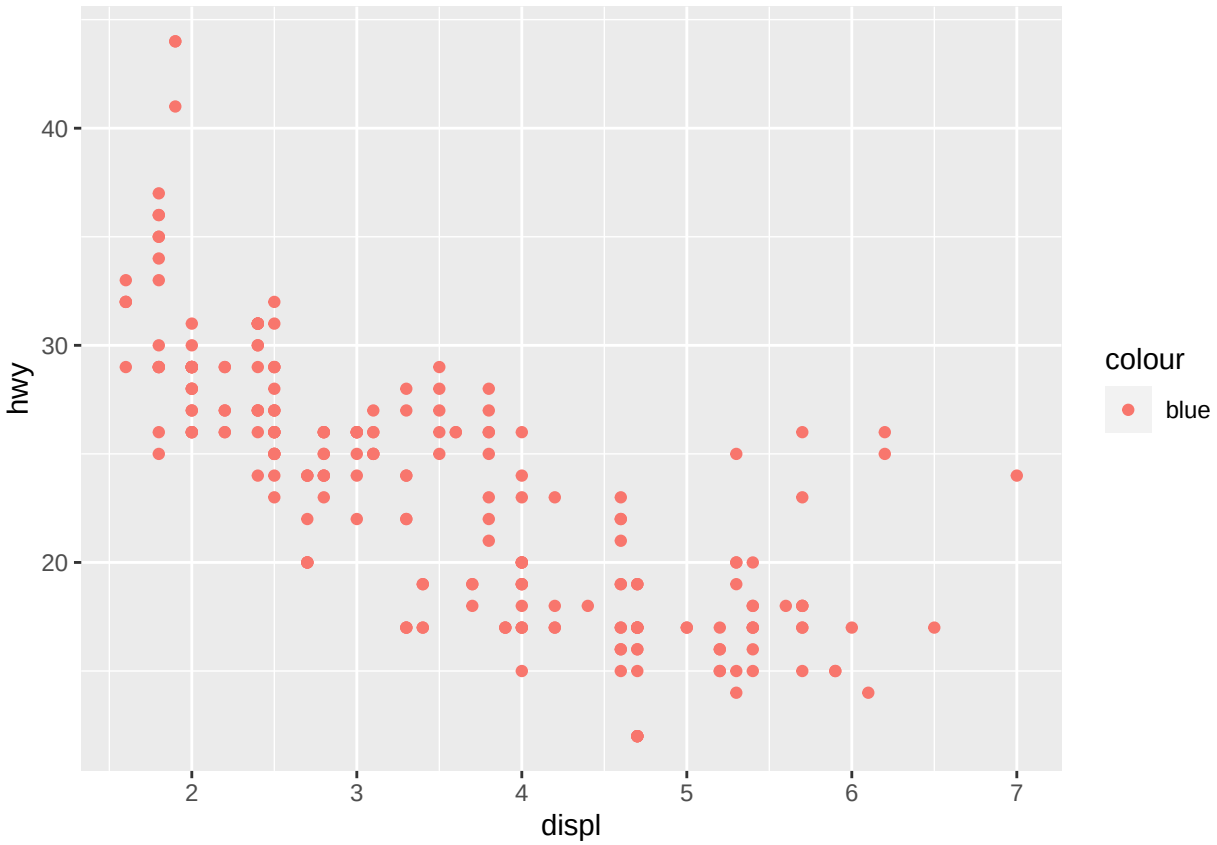
6. Get the total number of observations for drv - type of drive train (f = front-wheel drive, r = rear wheel drive, 4 = 4wd) and class - type of class (Example: suv, 2seater, etc.). Plot using the geom_tile() where the number of observations for class be used as a fill for aesthetics.

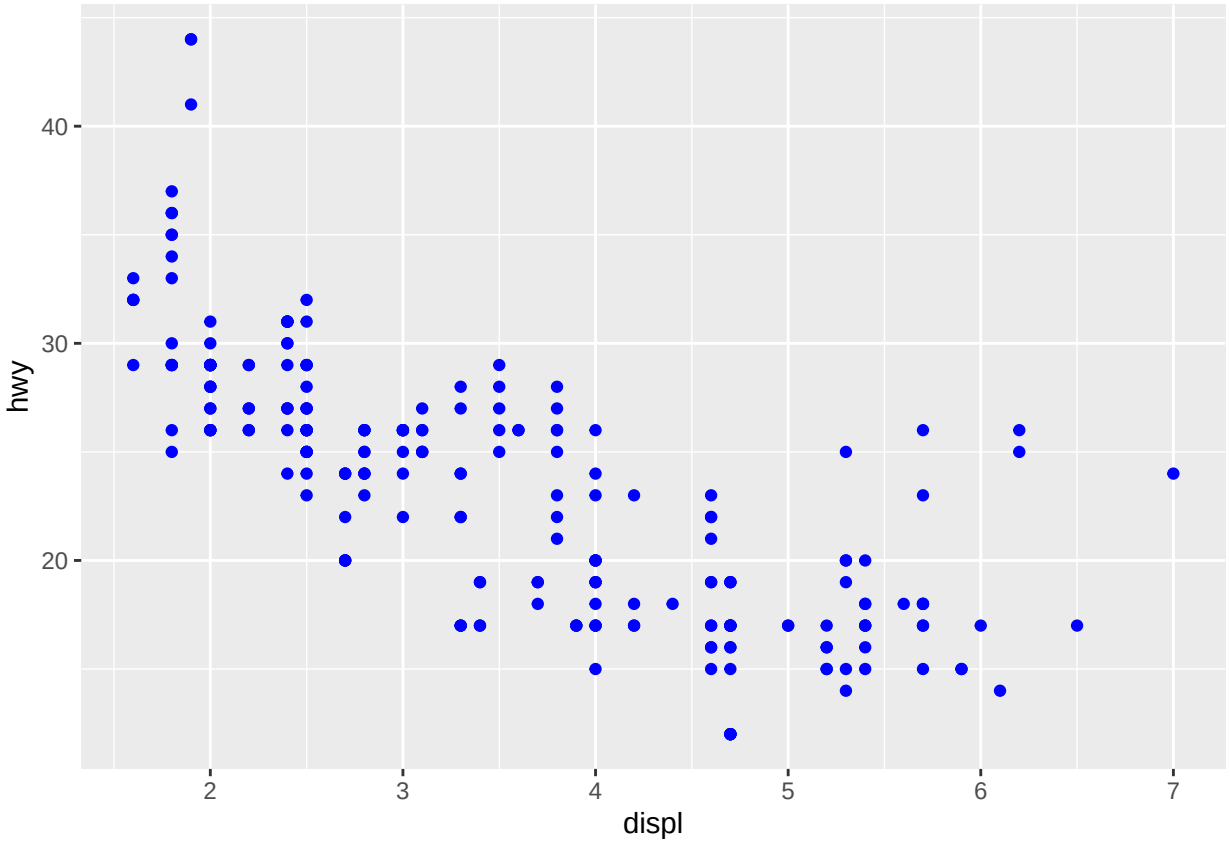
a. Show the codes and its result for the narrative in #6.



b. Interpret the result.
Areas covered with black are "mapped" using the mapping geometric point graph. y as class and x as drv.

7. Discuss the difference between these codes. Its outputs for each are shown below.


40
30
20
hwy
2
3
4
5
6
7
displ
colour
blue



In the first line of code, where the color goes inside aes(), you can see that it doesn't apply any coloring to the plot. As "blue" as a string doesn't exist in your data frame, it's not applied to your plot as a new coloring aesthetic. Instead, it will only be added to your legend. While, In the second line of code, the color goes outside aes(), and as you see, it works. In this case, with one color only, it doesn't show a legend.

8. Try to run the command ?mpg. What is the result of this command?

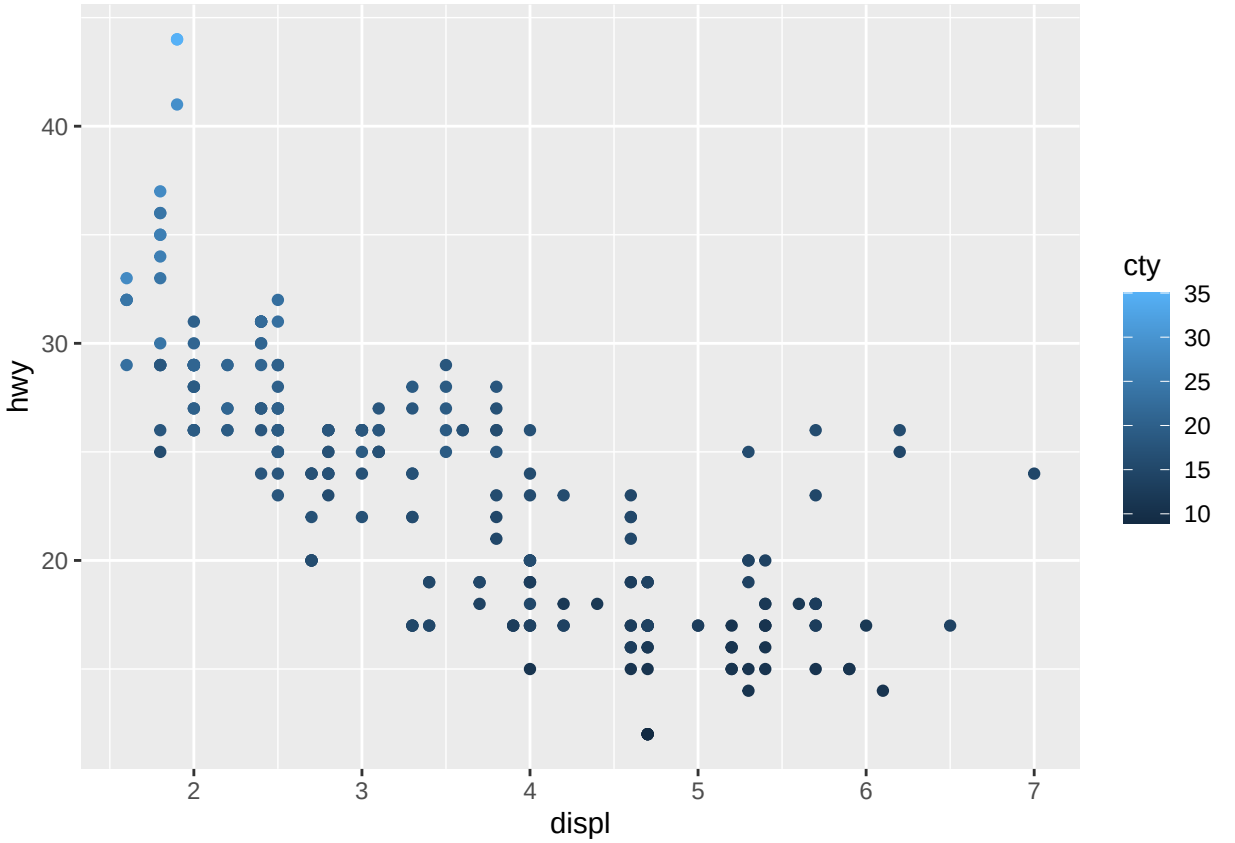
```
?mpg
```

```
## starting httpd help server ... done
```

a. Which variables from mpg dataset are categorical?
Categorical variables in mpg include: manufacturer, model, trans (type of transmission), drv (front-wheel drive, rear-wheel, 4wd), fl (fuel type), and class (type of car).

b. Which are continuous variables?
Continuous variables in mpg include: displ (engine displacement in litres), cyl (number of cylinders), cty (city miles/gallon), and hwy (highway gallons/mile).

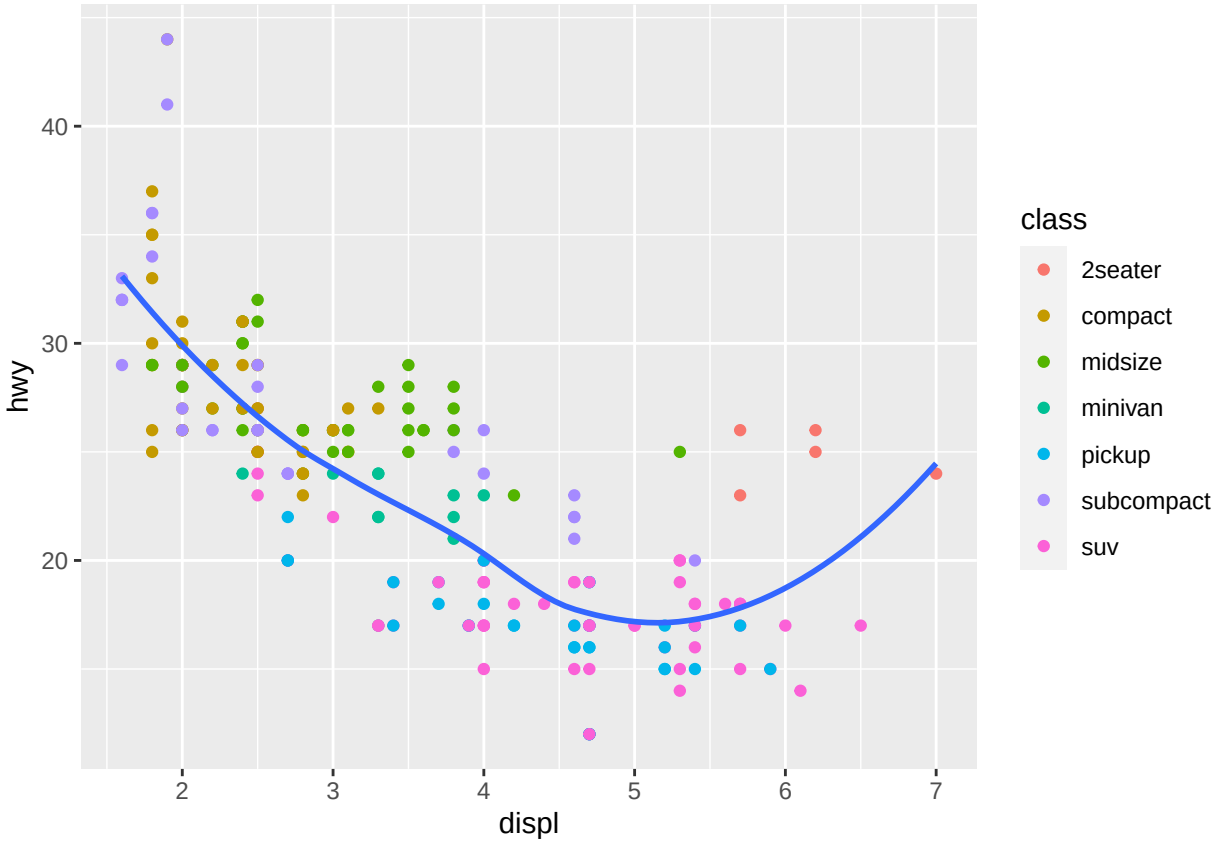
c. Plot the relationship between displ (engine displacement) and hwy(highway miles per gallon). Mapped it with a continuous variable you have identified in #5-b. What is its result? Why it produced such output?



City highway miles per gallon, or cty, is a continuous variable. The continuous variable uses a scale that ranges from a light to dark blue color instead of discrete colors.

9. Plot the relationship between displ (engine displacement) and hwy(highway miles per gallon) using geom_point(). Add a trend line over the existing plot using geom_smooth() with se = FALSE. Default method is "loess".

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



10. Using the relationship of displ and hwy, add a trend line over existing plot. Set the se = FALSE to remove the confidence interval and method = lm to check for linear modeling.

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : span too small. fewer data values than degrees of freedom.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : pseudoinverse used at 5.6935

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : neighborhood radius 0.5065

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : There are other near singularities as well. 0.65044

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : pseudoinverse used at 4.008

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : neighborhood radius 0.708
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : reciprocal condition number 0
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : There are other near singularities as well. 0.25
```