ggplot

```
# ggplot2 includes a dataset "mpg"
#install.packages("ggplot2")
library(ggplot2)
# ? gives help on a function or dataset
?mpg
#### mpg dataset
# head() lists the first 6 rows of a data.frame
head(mpg)
```

```
## # A tibble: 6 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     1.80  1999     4 auto~ f        18    29 p     comp~
## 2 audi         a4     1.80  1999     4 manu~ f        21    29 p     comp~
## 3 audi         a4     2.00  2008     4 manu~ f        20    31 p     comp~
## 4 audi         a4     2.00  2008     4 auto~ f        21    30 p     comp~
## 5 audi         a4     2.80  1999     6 auto~ f        16    26 p     comp~
## 6 audi         a4     2.80  1999     6 manu~ f        18    26 p     comp~
```

```
# manufacturer: car manufacturer    15 manufacturers
# model: model name 38 models
# displ: numeric engine displacement in liters,
# 1.6 - 7.0, median: 3.3
# year: year of manufacturing   1999, 2008
# cyl   : number of cylinders   4, 5, 6, 8
# trans:    type of transmission    automatic, manual
# drv: drive type   f, r, 4, f=front wheel, r=rear wheel,
# 4=4 wheel
# cty: city mileage miles per gallon
# hwy: highway mileage  miles per gallon
# fl: fuel type 5 fuel types: diesel, petrol, electric, etc.
# class: vehicle class 7 types
#  (compact, SUV, minivan etc.)

# str() gives the structure of the object
str(mpg)
```

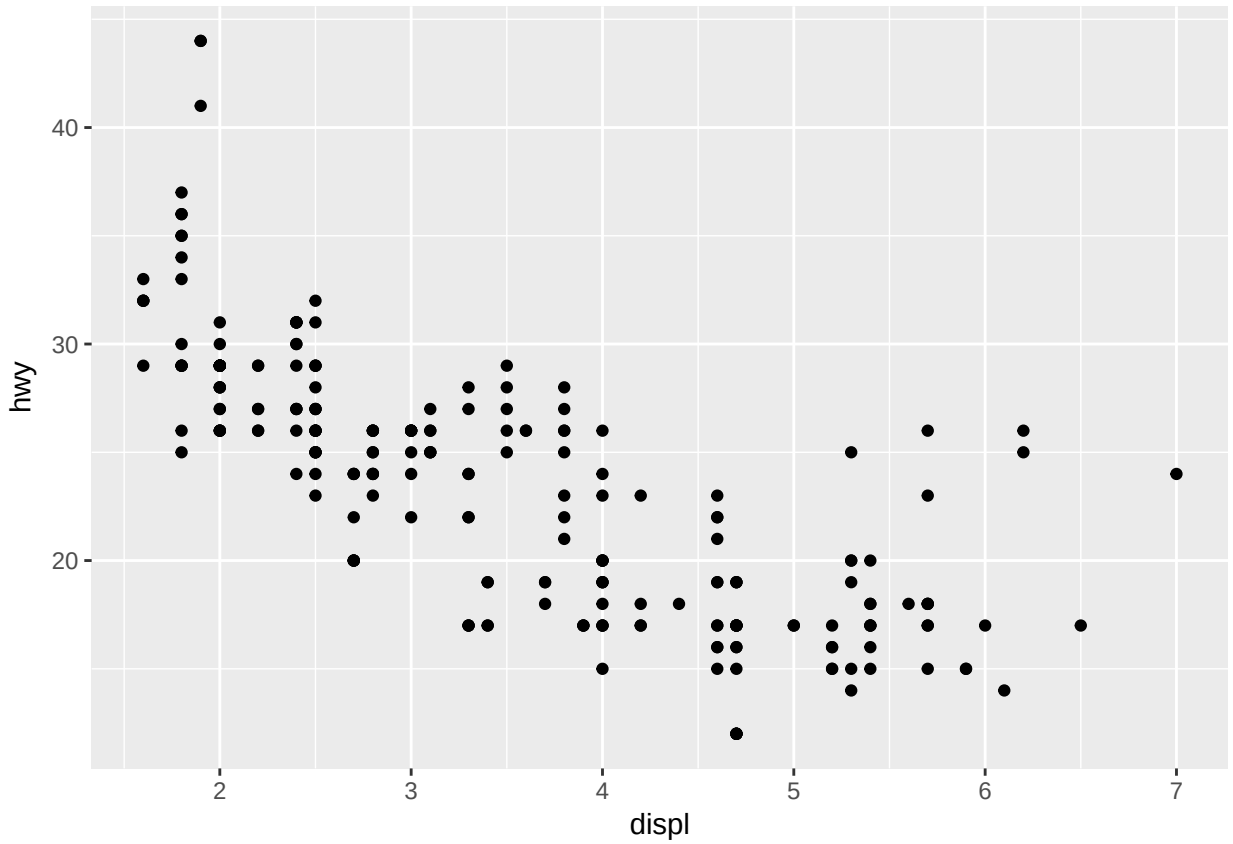
```
## Classes 'tbl_df', 'tbl' and 'data.frame':    234 obs. of  11 variables:
##  $ manufacturer: chr  "audi" "audi" "audi" "audi" ...
##  $ model       : chr  "a4" "a4" "a4" "a4" ...
##  $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int  1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
##  $ trans       : chr  "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
##  $ drv         : chr  "f" "f" "f" "f" ...
##  $ cty         : int  18 21 20 21 16 18 18 18 16 20 ...
##  $ hwy         : int  29 29 31 30 26 26 27 26 25 28 ...
##  $ fl          : chr  "p" "p" "p" "p" ...
##  $ class       : chr  "compact" "compact" "compact" "compact" ...
```

```
# summary() gives frequeny tables for categorical variables
#             and mean and five-number summaries for continuous variables
```

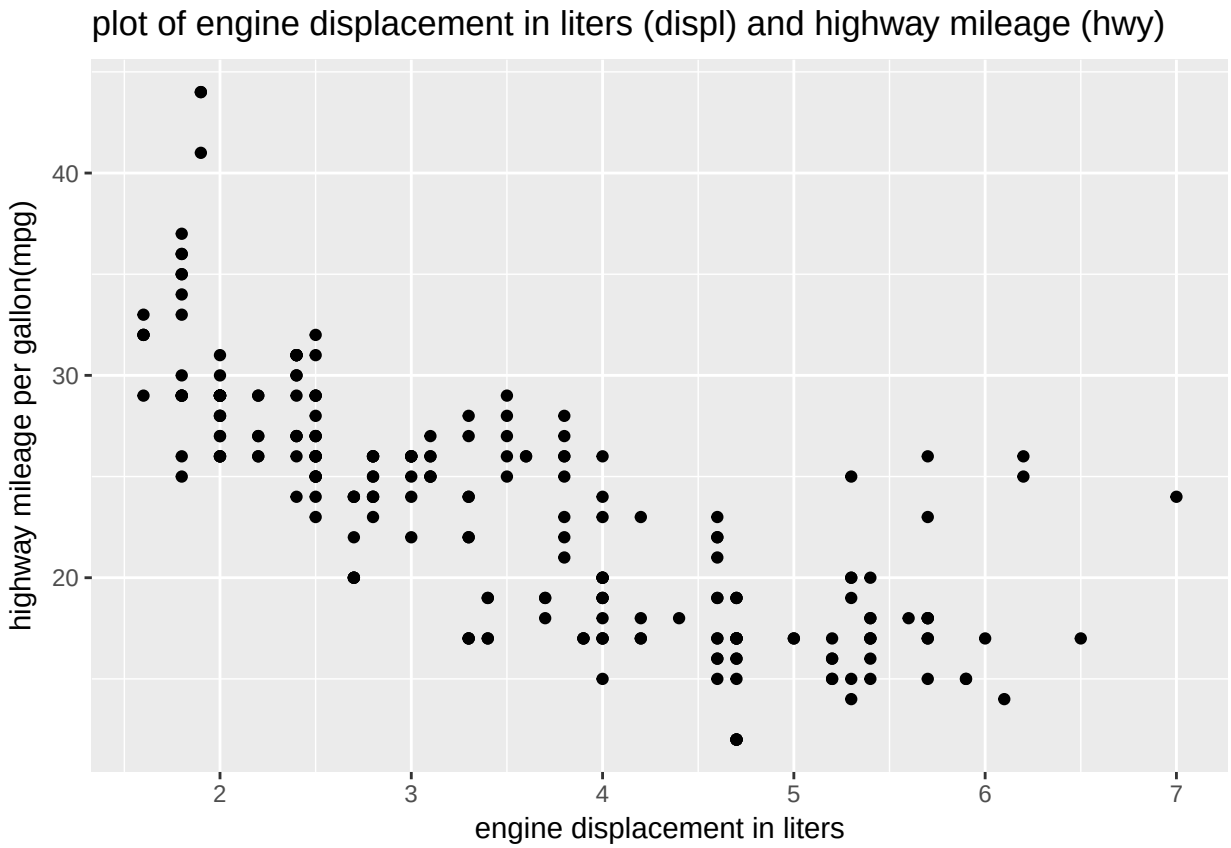
```r
summary(mpg)
```

```
##  manufacturer          model               displ            year
##  Length:234         Length:234         Min.   :1.600   Min.   :1999
##  Class :character   Class :character   1st Qu.:2.400   1st Qu.:1999
##  Mode  :character   Mode  :character   Median :3.300   Median :2004
##                                        Mean   :3.472   Mean   :2004
##                                        3rd Qu.:4.600   3rd Qu.:2008
##                                        Max.   :7.000   Max.   :2008
##       cyl           trans               drv                 cty
##  Min.   :4.000   Length:234         Length:234         Min.   : 9.00
##  1st Qu.:4.000   Class :character   Class :character   1st Qu.:14.00
##  Median :6.000   Mode  :character   Mode  :character   Median :17.00
##  Mean   :5.889                                         Mean   :16.86
##  3rd Qu.:8.000                                         3rd Qu.:19.00
##  Max.   :8.000                                         Max.   :35.00
##       hwy             fl                class
##  Min.   :12.00   Length:234         Length:234
##  1st Qu.:18.00   Class :character   Class :character
##  Median :24.00   Mode  :character   Mode  :character
##  Mean   :23.44
##  3rd Qu.:27.00
##  Max.   :44.00
```

```r
#### ggplot_mpg_displ_hwy, basic plot
#Geom: is the "type" of plot
#Aesthetics: shape, colour, size, alpha
#Faceting: "small multiples" displaying different subsets
# specify the dataset and variables
p <- ggplot(mpg, aes(x = displ, y = hwy))
p <- p + geom_point() # add a plot layer with points
print(p)
```



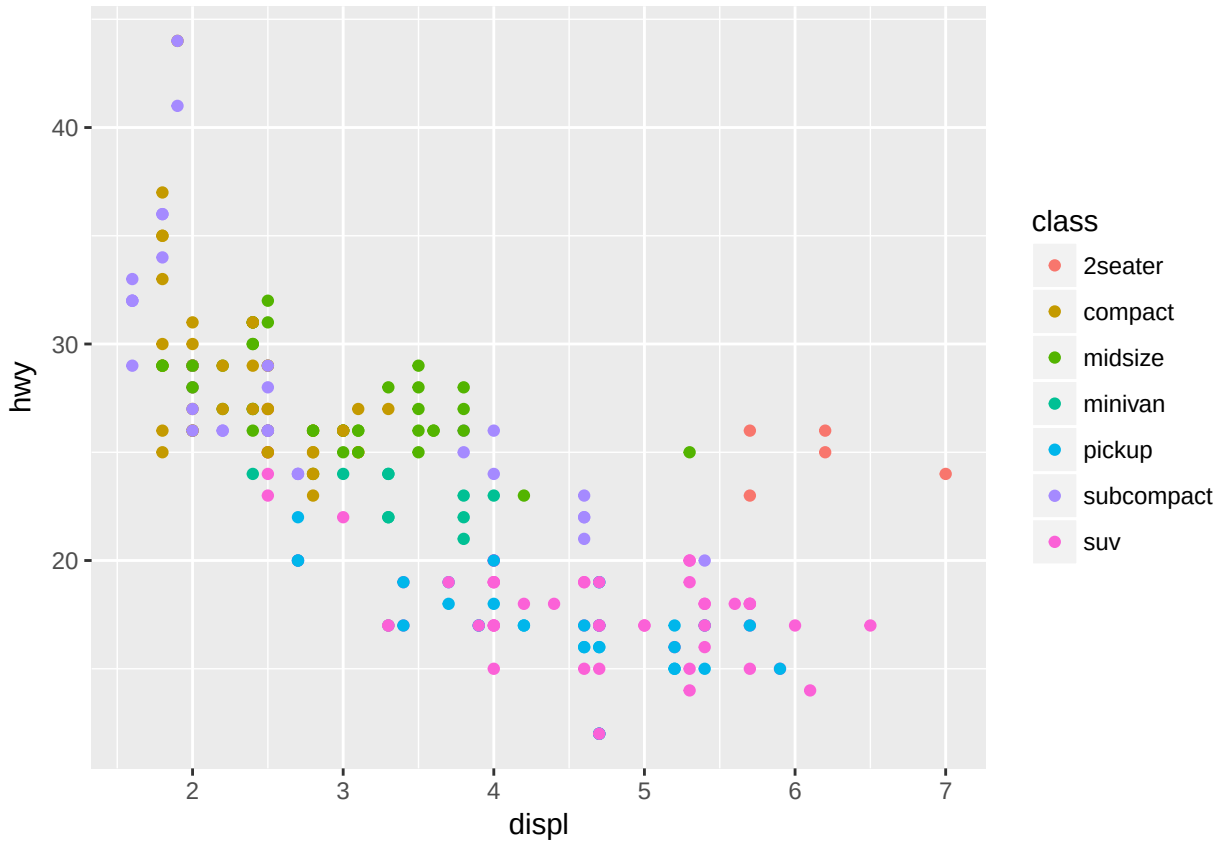

```
##adding titles and axis names
print(p+ggtitle("plot of engine displacement in liters (displ) and highway mileage (hwy)")
+labs(y="highway mileage per gallon(mpg)",x="engine displacement in liters" ))
```



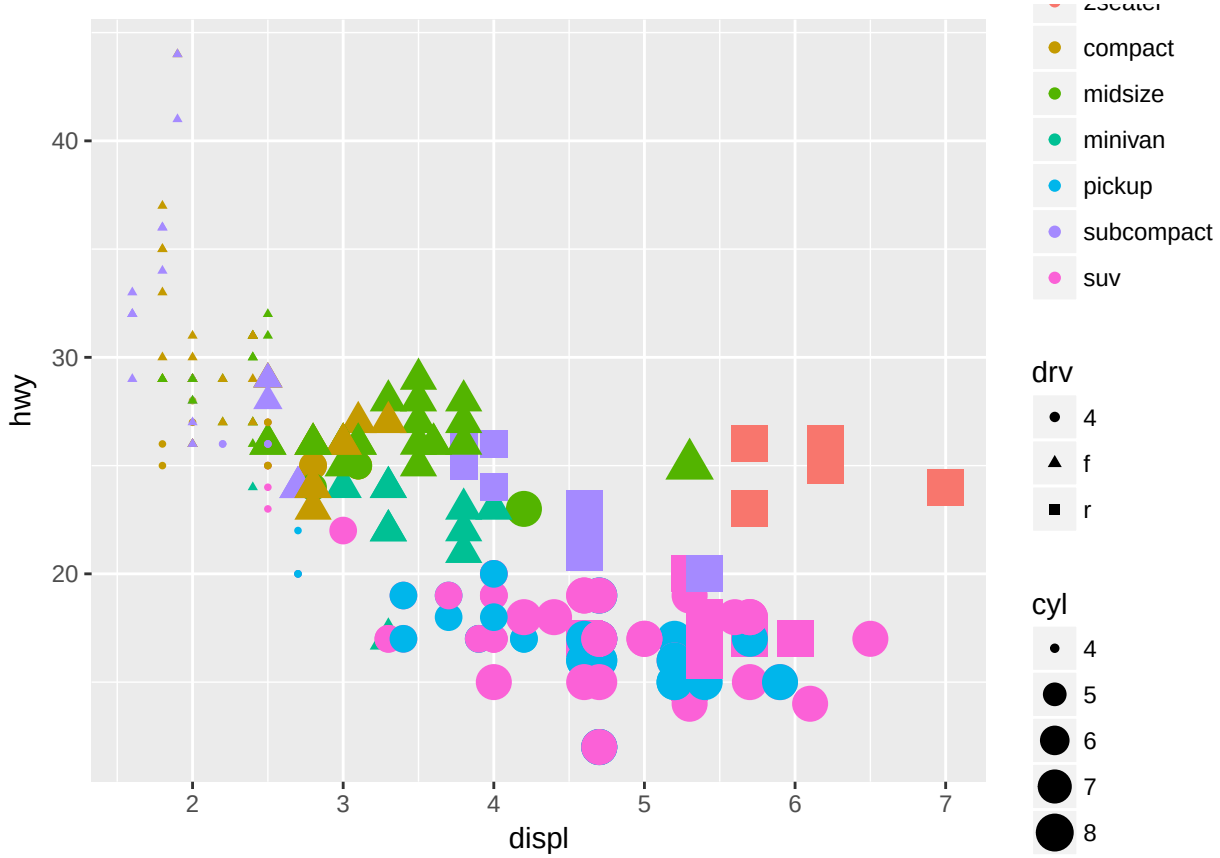

```
#### ggplot_mpg_displ_hwy_colour_class
#how many classes
unique(mpg$class)
```

```
## [1] "compact"    "midsize"    "suv"        "2seater"    "minivan"
## [6] "pickup"     "subcompact"
```

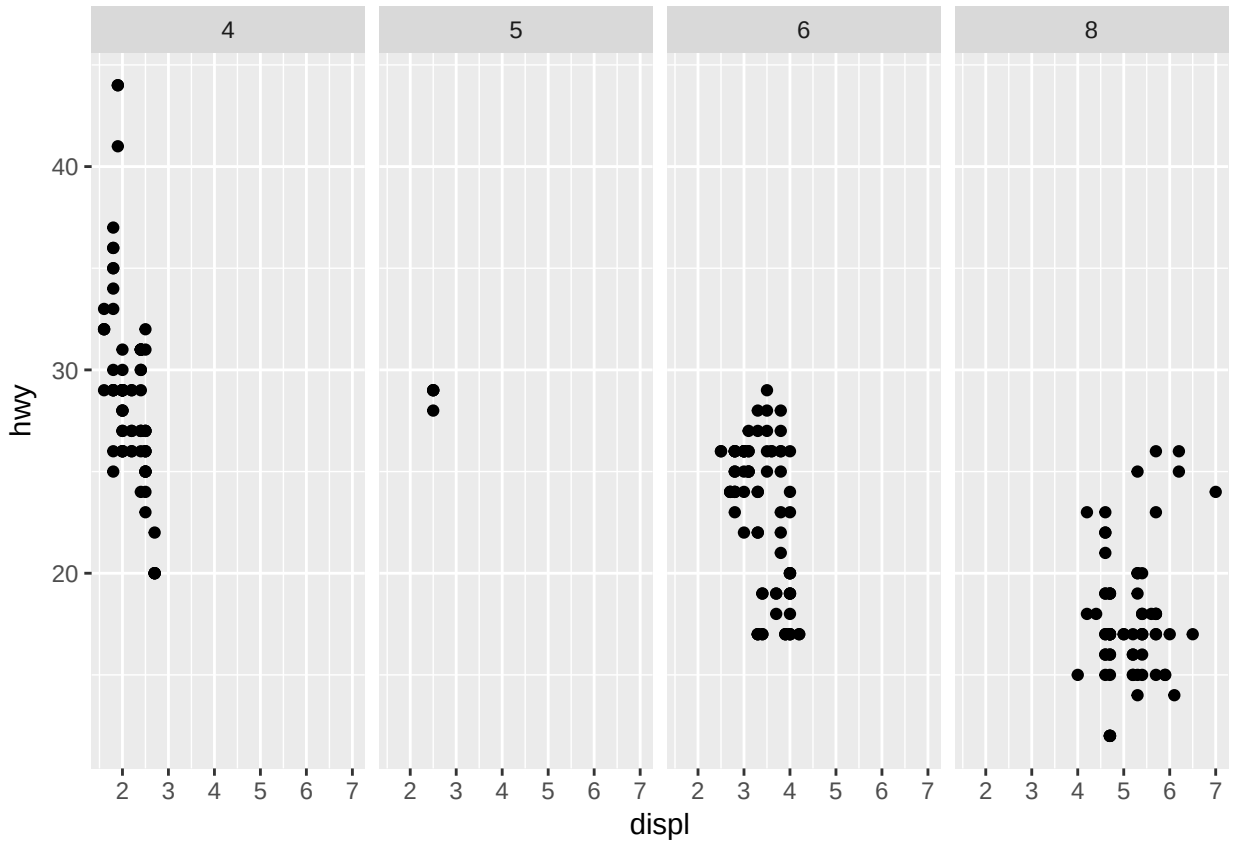
```
p <- ggplot(mpg, aes(x = displ, y = hwy))
p <- p + geom_point(aes(colour = class))
print(p)
```



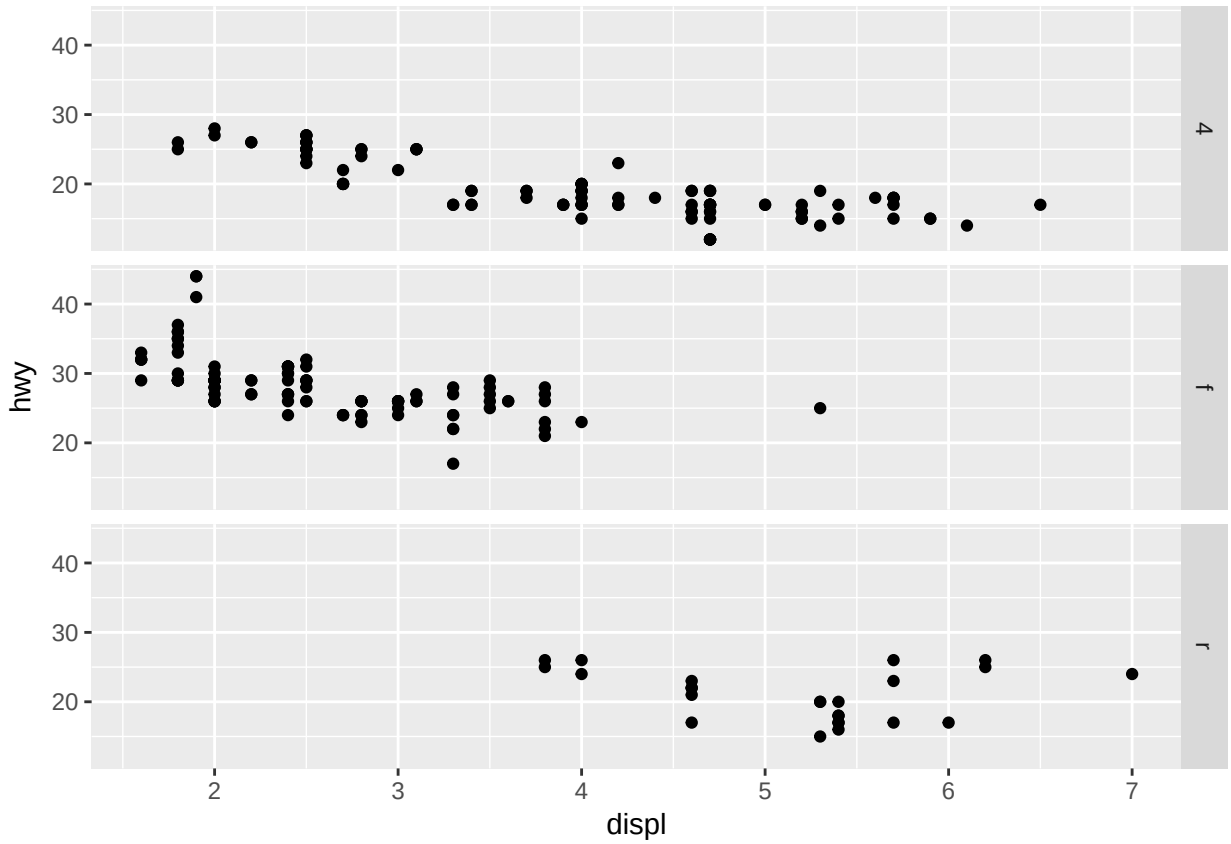

```
#### ggplot_mpg_displ_hwy_colour_class_size_cyl_shape_drv
p <- ggplot(mpg, aes(x = displ, y = hwy))
p <- p + geom_point(aes(colour = class, size = cyl, shape = drv))
print(p)
```



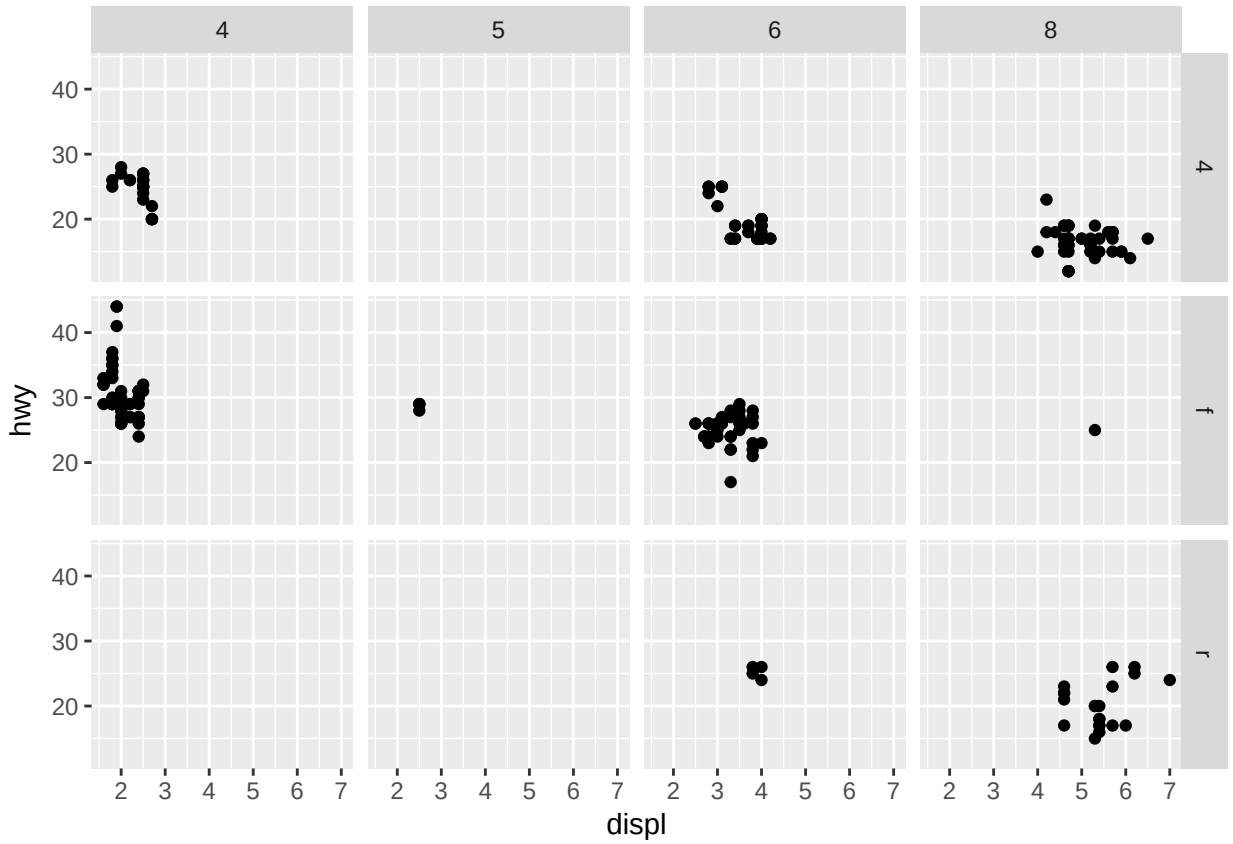

```
##Faceting A small multiple 2 (sometimes called faceting, trellis chart, lattice
#chart, grid chart, or panel chart) is a series or grid of small similar graphics or
#charts, allowing them to be easily compared.
#Typically, small multiples will display different subsets of the data.
#Useful strategy for exploring conditional relationships, especially for large
#data.
#### ggplot_mpg_displ_hwy_facet
# start by creating a basic scatterplot
p <- ggplot(mpg, aes(x = displ, y = hwy))
p <- p + geom_point()

## two methods
# facet_grid(rows ~ cols) for 2D grid, "." for no split.
# facet_wrap(~ var)       for 1D ribbon wrapped into 2D.

# examples of subsetting the scatterplot in facets
p1 <- p + facet_grid(. ~ cyl)     # columns are cyl categories
print(p1)
```



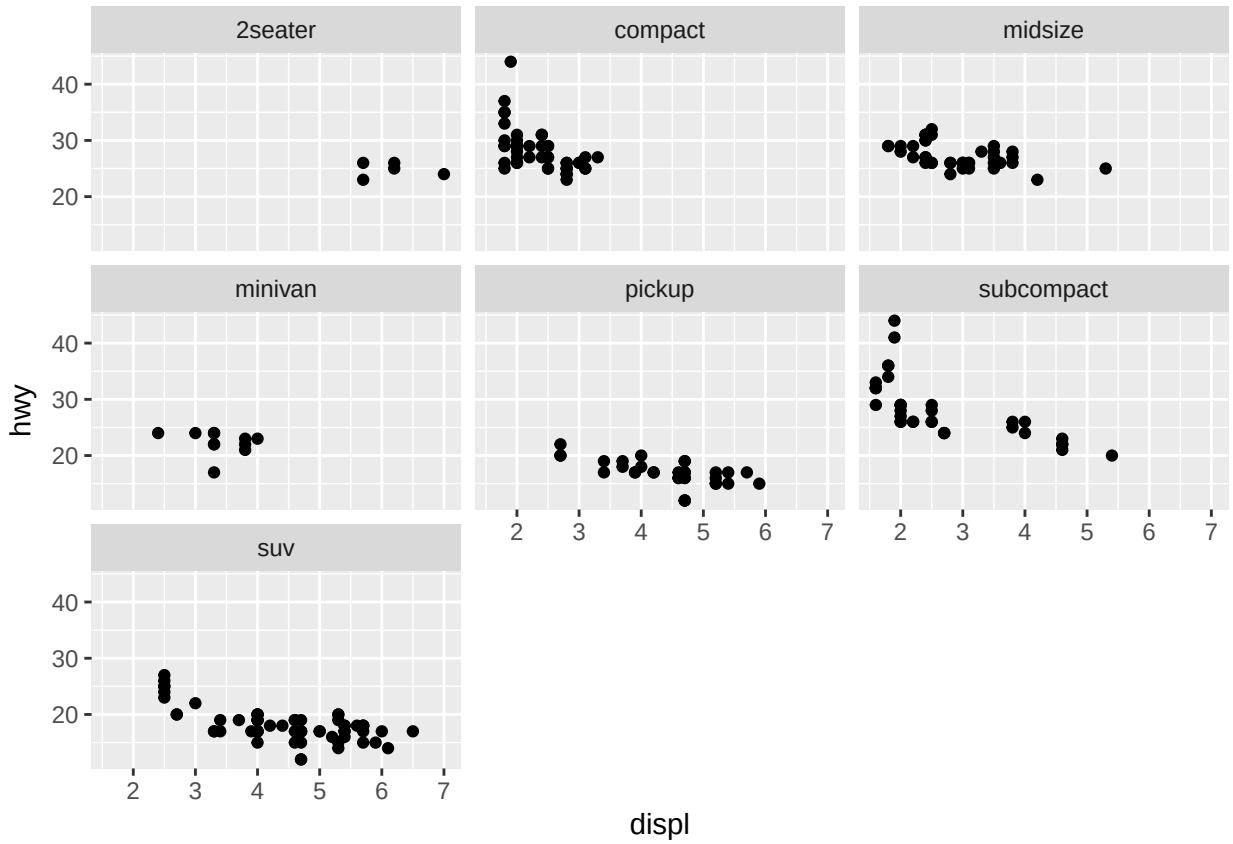

```
p2 <- p + facet_grid(drv ~ .)      # rows are drv categories
print(p2)
```



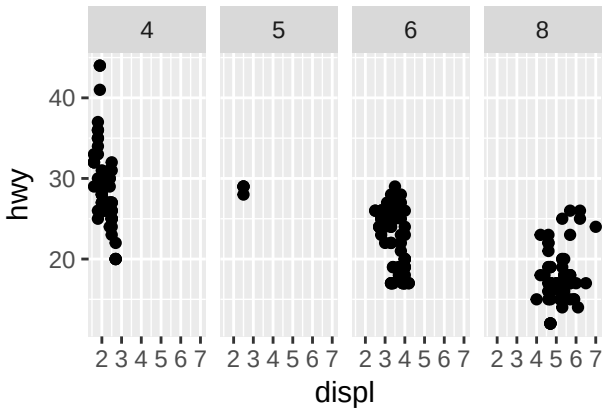

```
p3 <- p + facet_grid(drv ~ cyl)   # both
print(p3)
```

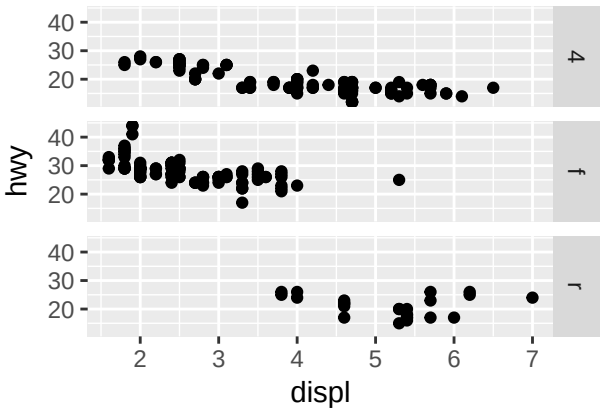


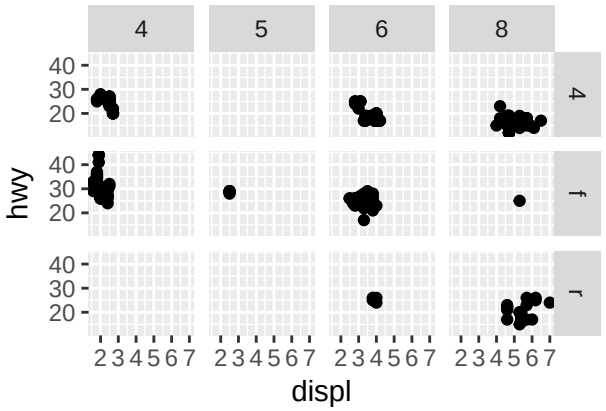

```
p4 <- p + facet_wrap(~ class)      # wrap plots by class category
print(p4)
```

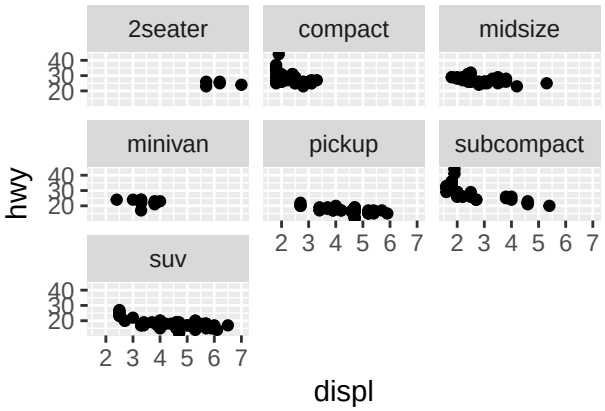



```
# plot all four in one arrangement
#install.packages("gridExtra")
library(gridExtra)
grid.arrange(p1, p2, p3, p4, ncol = 2)
```

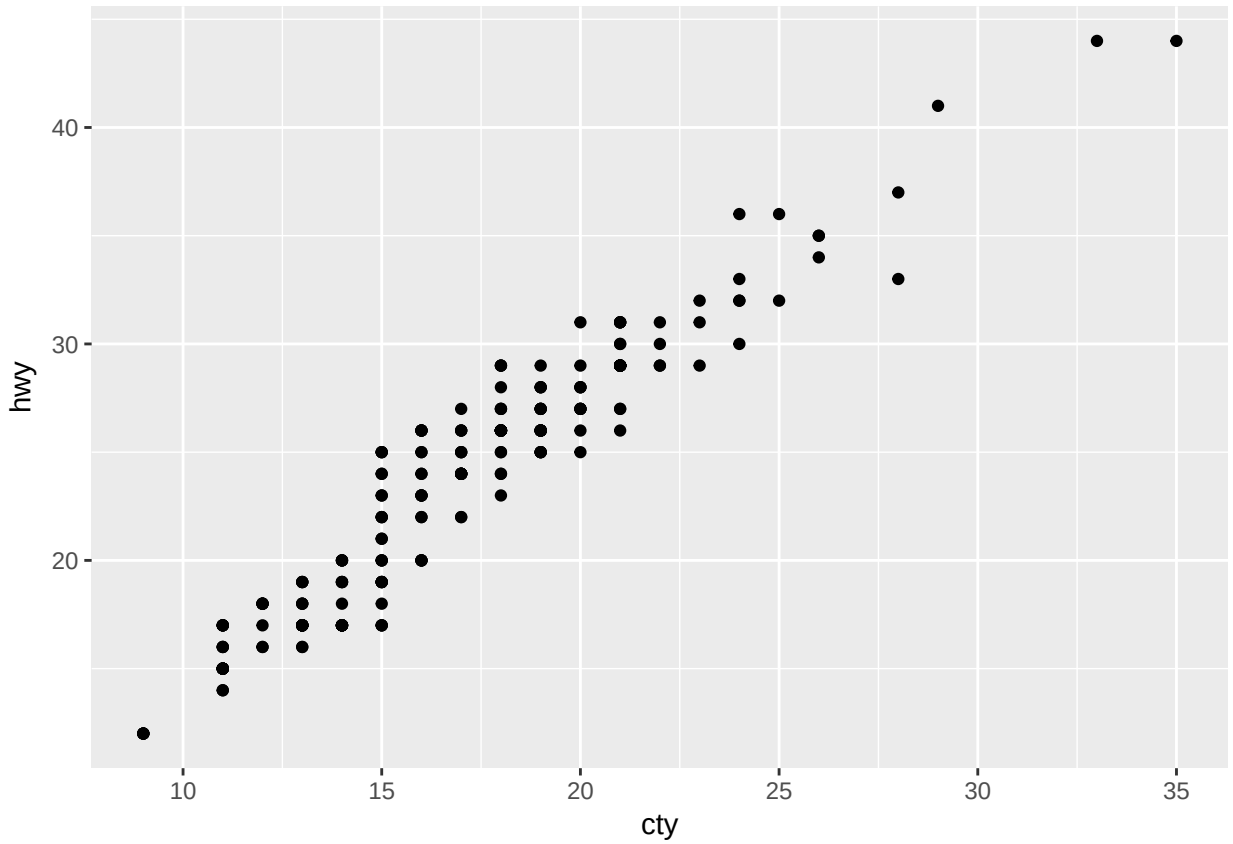




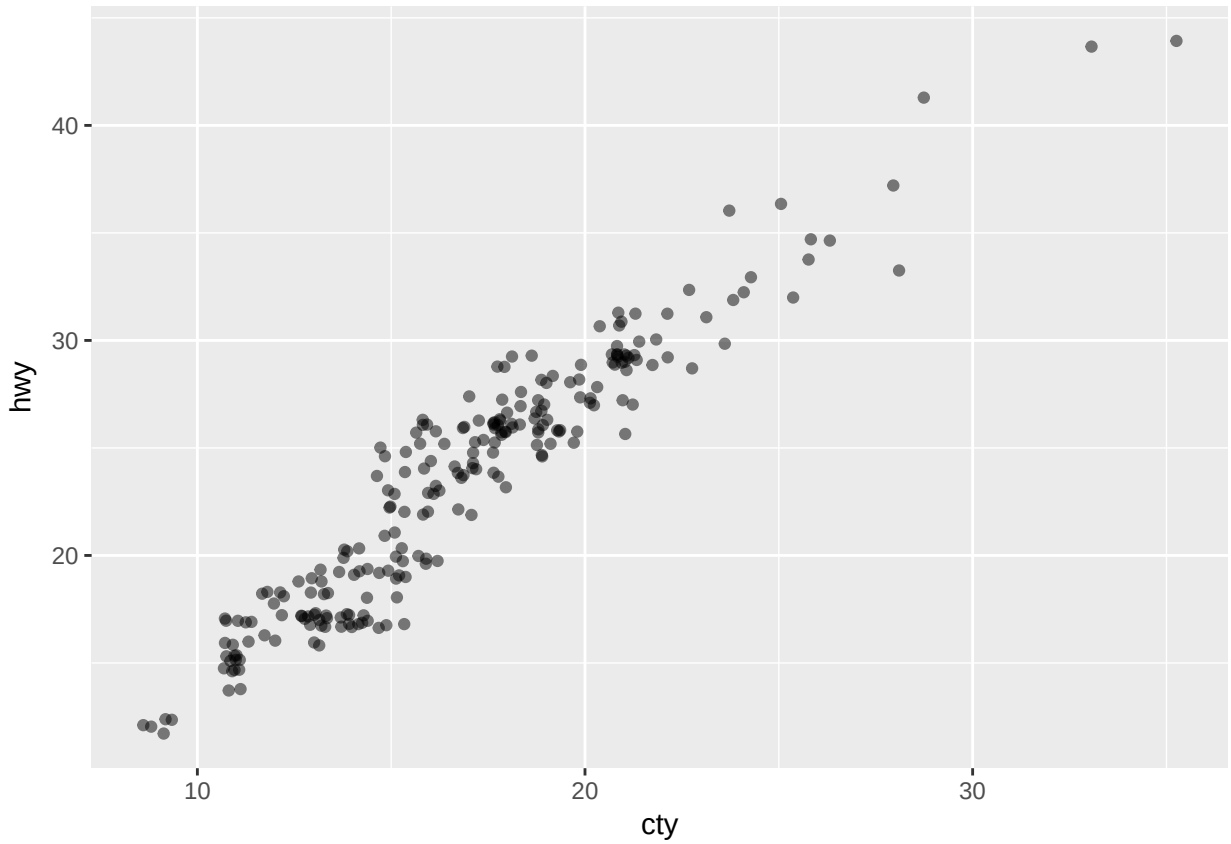




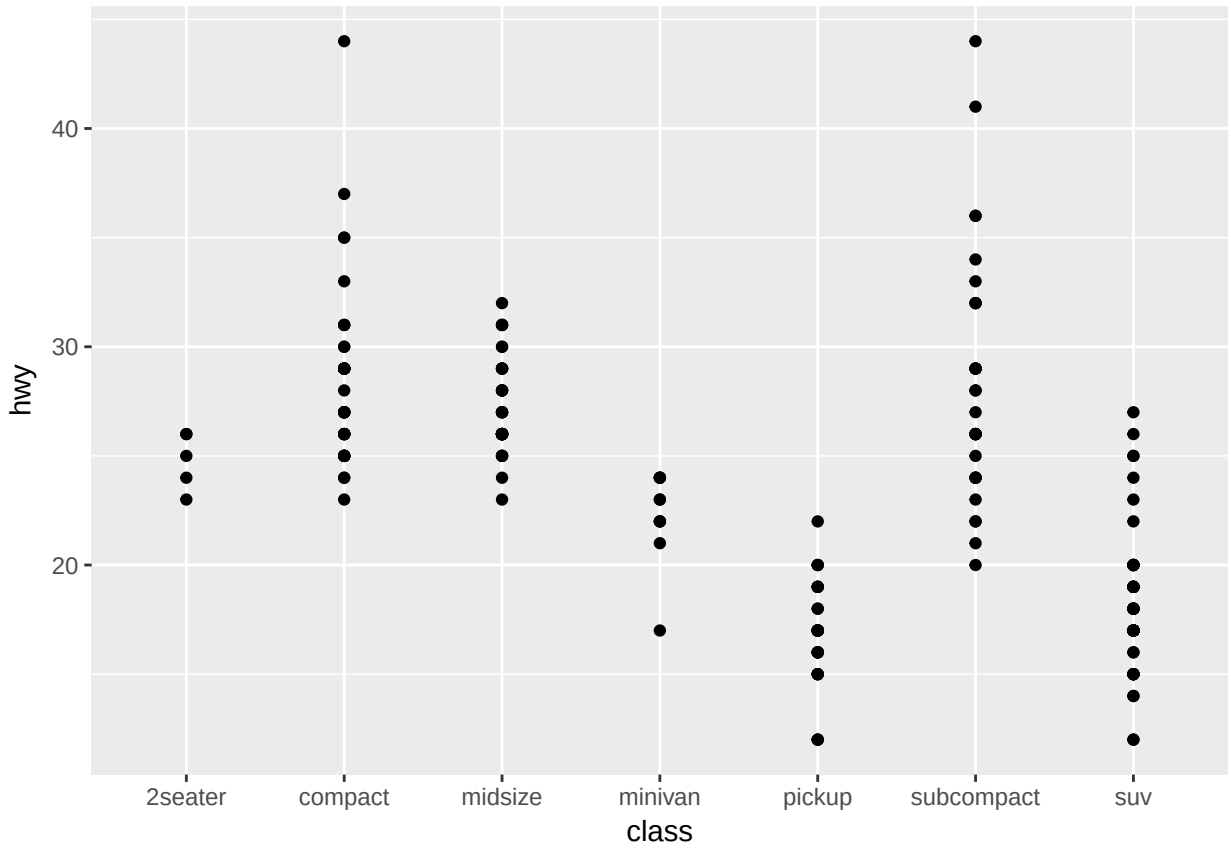

```
#### ggplot_mpg_cty_hwy
p <- ggplot(mpg, aes(x = cty, y = hwy))
p <- p + geom_point()
print(p)
```



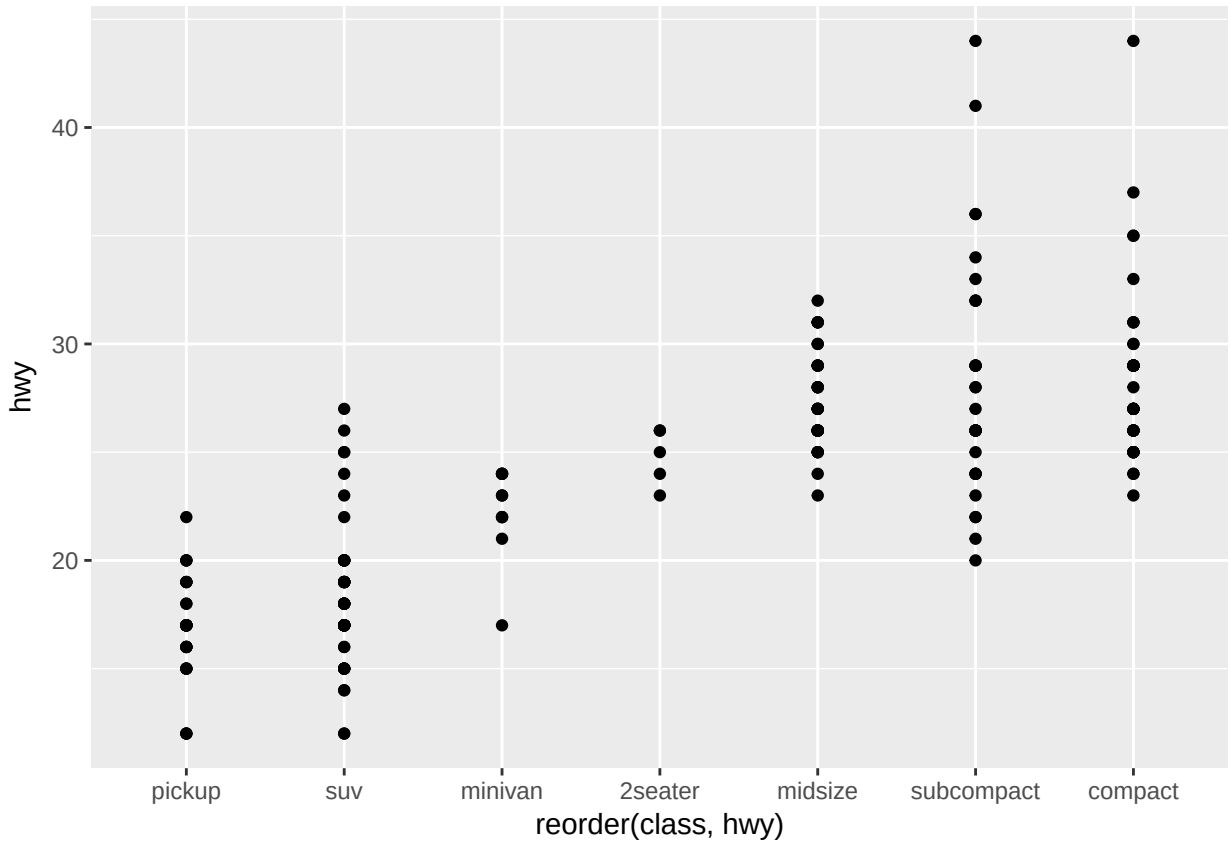

```
#observe a linear relationship

#problem: points lie on top of each other, so it's impossible to tell how many
#observations each point represents.
#A solution: Jitter the points to reveal the individual points and reduce the
#opacity to 1/2 to indicate when points overlap.

#### ggplot_mpg_cty_hwy_jitter
p <- ggplot(mpg, aes(x = cty, y = hwy))
p <- p + geom_point(position = "jitter", alpha = 1/2)
print(p)
```



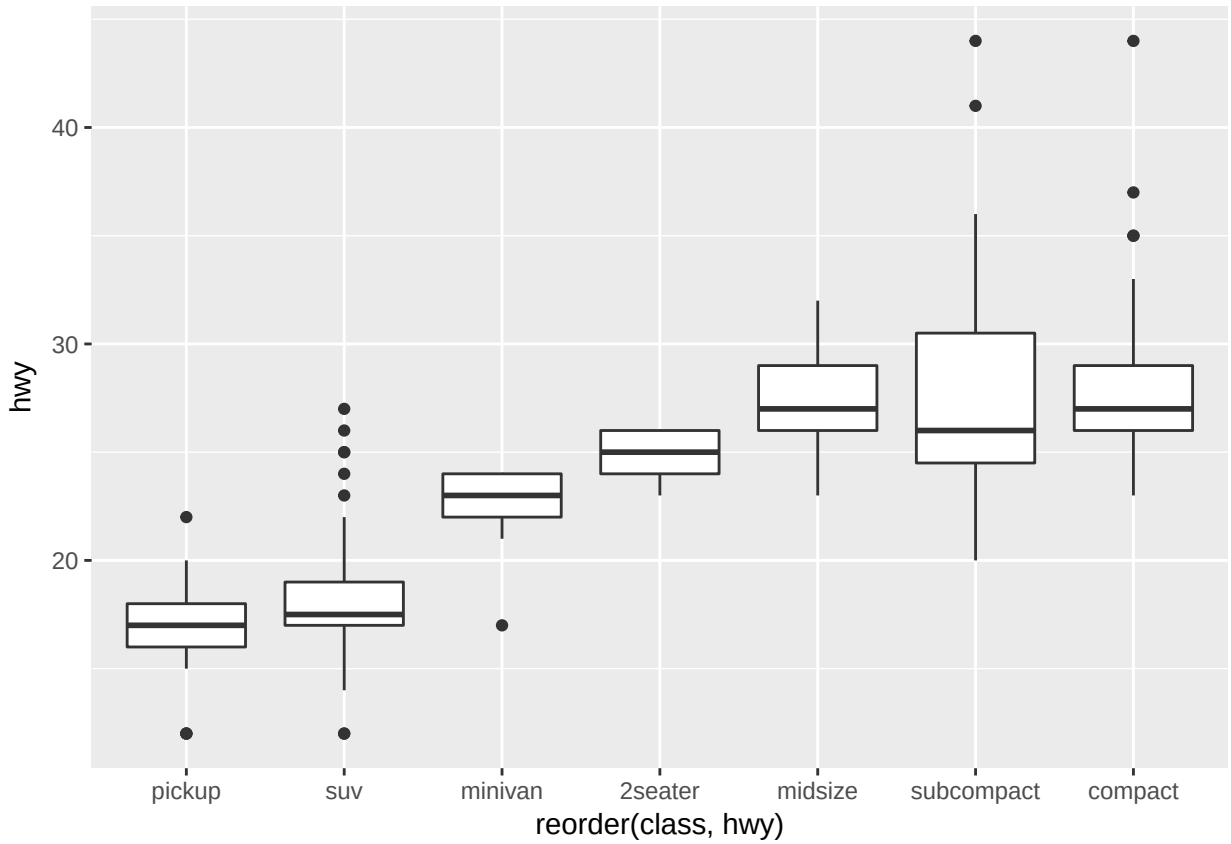

```
#### ggplot_mpg_class_hwy
p <- ggplot(mpg, aes(x = class, y = hwy))
p <- p + geom_point()
print(p)
```



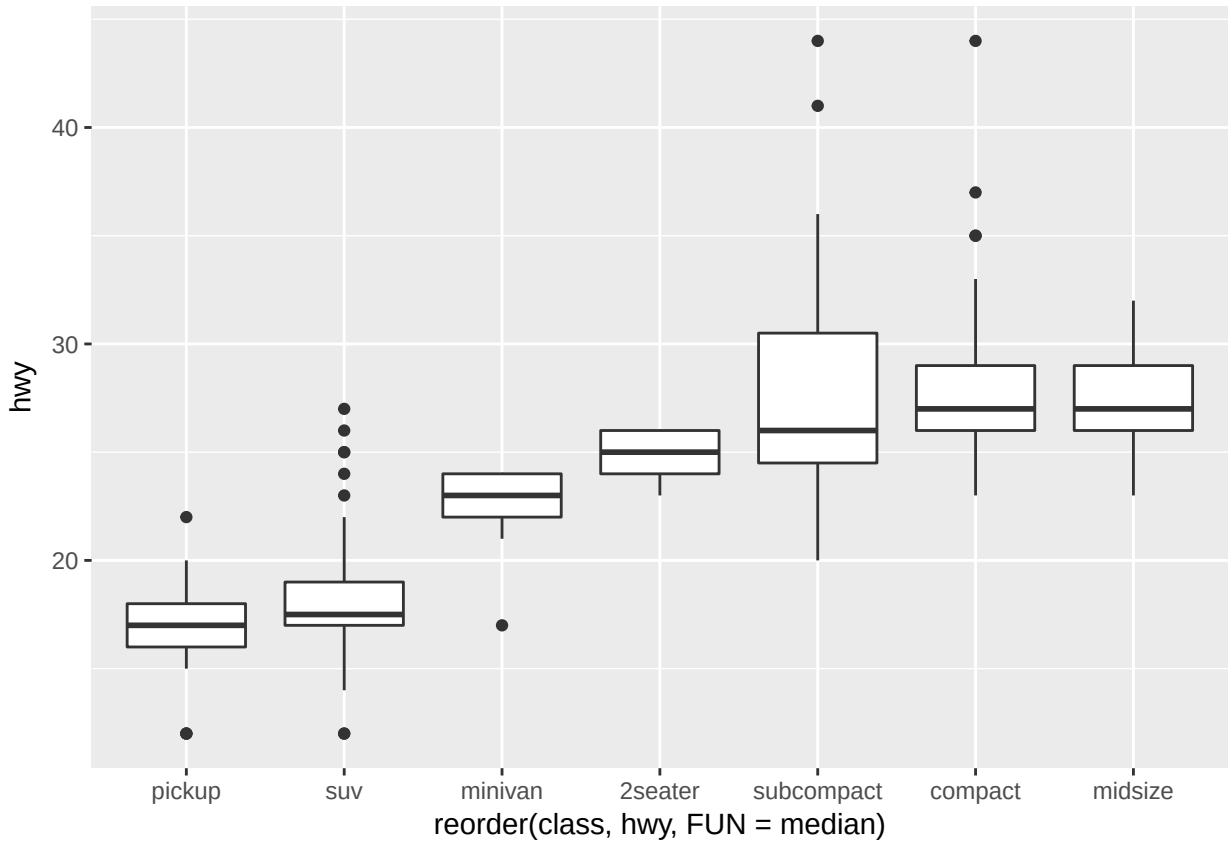

```
##Reorder the class variable by the mean hwy for a meaningful
##ordering. Get help with ?reorder to understand how this works.

#### ggplot_mpg_reorder_class_hwy
p <- ggplot(mpg, aes(x = reorder(class, hwy), y = hwy))
p <- p + geom_point()
print(p)
```




```
#### ggplot_mpg_reorder_class_hwy_boxplot
p <- ggplot(mpg, aes(x = reorder(class, hwy), y = hwy))
p <- p + geom_boxplot()
print(p)
```




```
#### ggplot_mpg_reorder_class_hwy_boxplot_median
##reorder by median() instead of mean() (mean is the default)
p <- ggplot(mpg, aes(x = reorder(class, hwy, FUN = median), y = hwy))
p <- p + geom_boxplot()
print(p)
```




```
## #### Review
## library(ggplot2)
## ?help
## head()
## str()
## summary()
## ggplot(df)
## geom_point()
##   aes()
##     colour, size, shape, alpha
##   position = "jitter"
##   geom_jitter(position = position_jitter(width = 0.1))
## geom_boxplot()
## facet_grid()
## facet_wrap()
## reorder()
##   median()
```