

Will Speak too fast and mumble words

Please ask Siva to repeat if you do not understand

- R is an open-source compute programming language and runs on Linux, Windows, and Mac-OS.
- R is FREE.
- The R project web page
<http://www.r-project.org>

• - www - help

available

• R - blogger

- R is modeled after S and S-Plus. The S language was developed in the late 1980s at AT & T labs.
- The R project was started by Robert Gentleman and Ross Ihaka of the Statistics Department of the University of Auckland in 1995. [*Journal of Computational and Graphical Statistics*,5:3, pp. 299-314. 1996.]
- R is now a collaborative project with many contributors and is maintained by the R core-development team.

Installing R

- To download R visit <https://cloud.r-project.org>

- Rstudio is an Integrated Development Environment, or IDE for R. To download Rstudio visit <http://www.rstudio.com/download>

- Data and its analysis has a rich and wide literature.
- Three kinds of Data:
 - Categorical Data
 - Discrete Numeric Data
 - Continuous Numeric Data
- *On the Theory of Scales of Measurement* By S. S. Stevens Science 07 Jun 1946: Vol. 103, Issue 2684, pp. 677-680, gave a broad classification of data from measurements into 9 categories.

{ 0 or 1
A or B ...

Numbers !

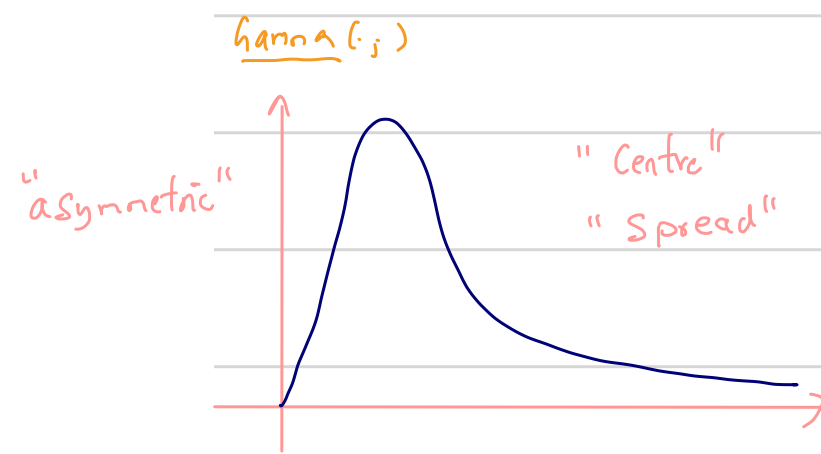
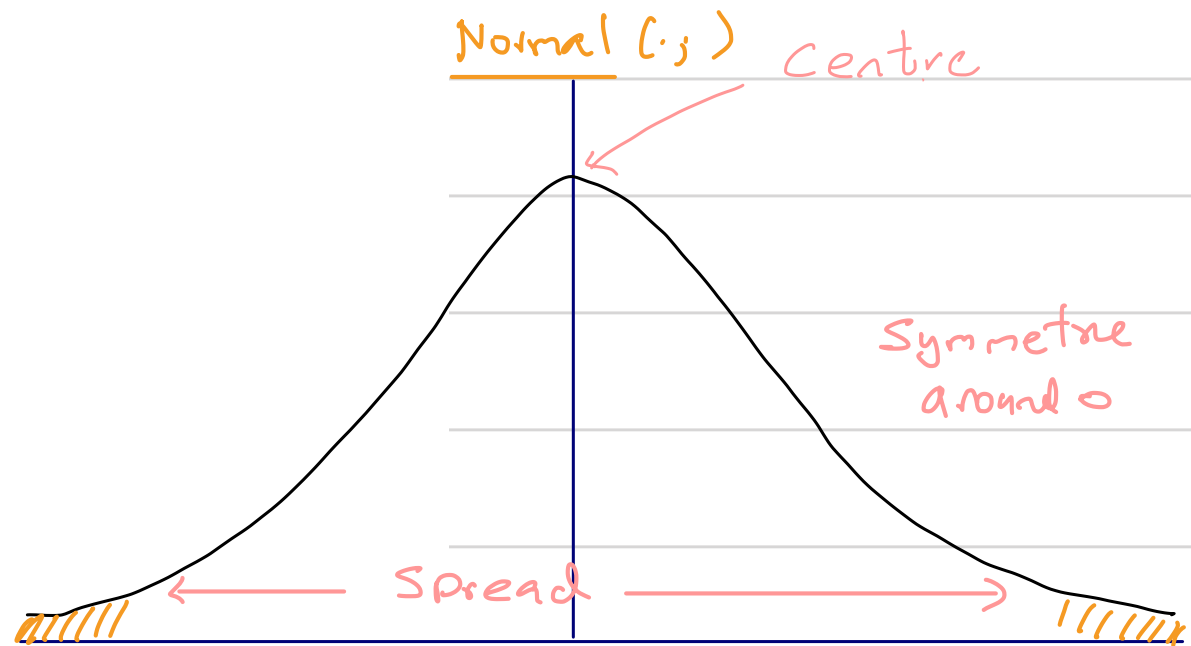
Discrete Numerical Data

- Many data are described in terms of numbers.
- Many variables naturally take on only discrete values.
- Boxplot and Histograms are used to visualise such data.

- Daily data of Temperature
- Earnings in a stock-market

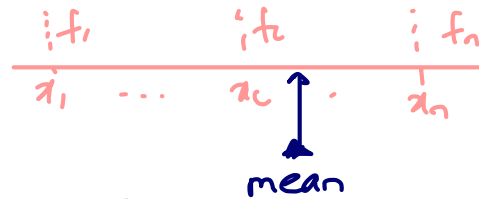
Discrete Numerical Data: Key features

- Center
- Spread
- Shape

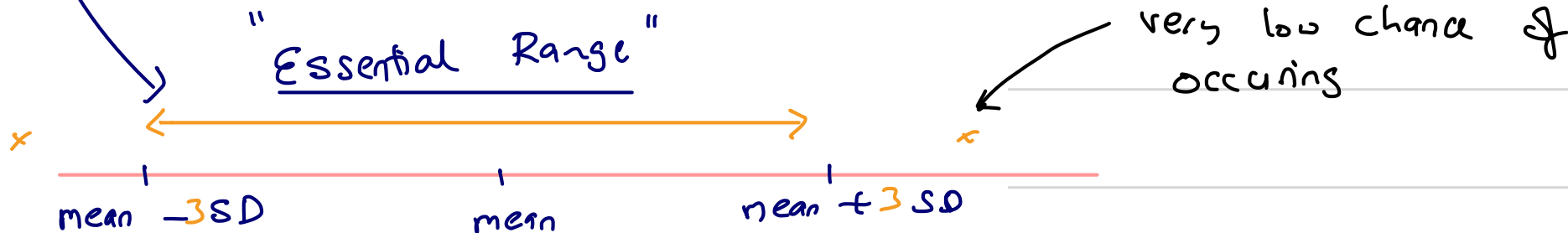


Discrete Numerical Data: Key features

$$\text{mean} = \frac{\sum_{i=1}^n f_i x_i}{\sum_{i=1}^n f_i}$$



- **Center** Widely used measure of centre is the **mean** or the average of the data set. Other measures include the **median** and the **mode**
- **Spread** Understanding variability of the given data is very important. If one were to understand **mean** as specifying the center then the range of the data set around it is determined by its variability or spread. It is often measured by the variance (**var**) or standard deviation (**sd**) or the inter-quartile range (**IQR**).
- **Shape** To understand various distributional aspects of the dataset one needs to understand its "shape". For e.g. if it is symmetric or skewed around its mean. Other aspects include among the data points which are more likely than others.



R has a lot inbuilt Datasets that one can use. The command :

```
> data()
```

will list currently installed data sets.

- Useful collection

- strongly encourage

to find a dataset

and repeat

experiments done

in a class.

R has a lot inbuilt Datasets that one can use. The command :

```
> data()
```

will list currently installed data sets.

Provides a comprehensive
list.

- R stores many datasets as data frame (often).
- A data frame is a rectangular collection of variables (in the columns) and observations (in the rows).

Let us learn about real data stored as data frame.

```
> ?airquality
```

- Describes the data set

airquality

- 153 - Rows

6 - Columns

Let us learn about `airquality` dataset a bit more.

- we could print the entire data set on the screen

```
>airquality
```

but this is too much information.

- Let us try the `head()` function

```
> head(airquality)
```

This provides the first six rows.

airquality in R

- Below provides the first ten rows.

```
> head(airquality, n = 10)
```

- Data can be called using row and column number

```
> airquality[148,4]
```

```
[1] 63
```

- We can use the variable name for the given column and call it by its position.

```
> airquality$Temp[148]
```

```
[1] 63
```

- Provides an entire row

```
> airquality[148,]
```

- Provides Ozone Temp columns

```
> airquality[,c(1,4)]
```

using `c()` function we can form any vector and that will enable display of the respective columns. We did not specify the row, so all rows will be displayed.

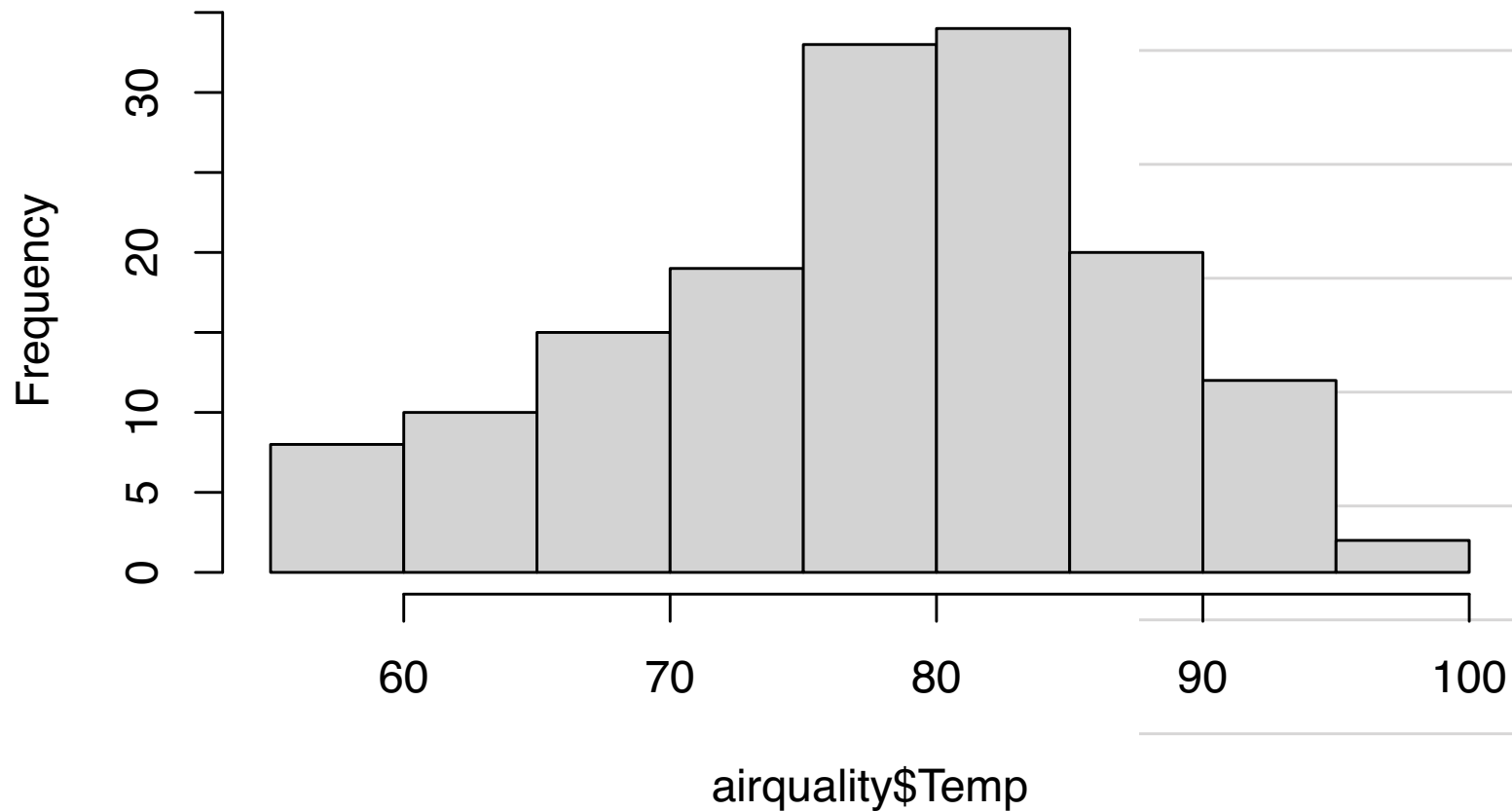
Five Number Summary and Histograms

```
> summary(airquality$Temp)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
56.00	72.00	79.00	77.88	85.00	97.00

```
> hist(airquality$Temp)
```

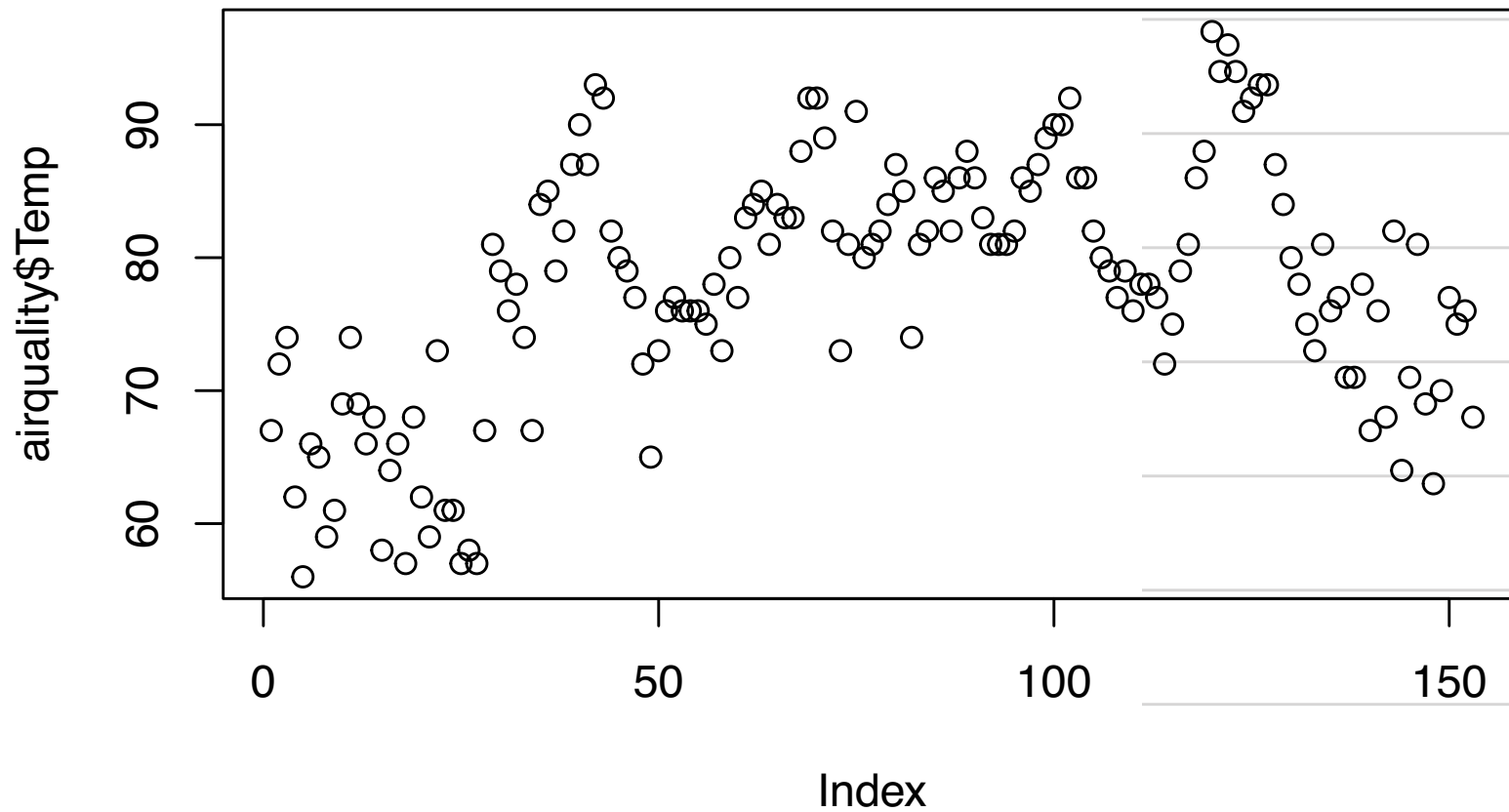
Histogram of airquality\$Temp



Plot

We can use the `plot` function to just plot.

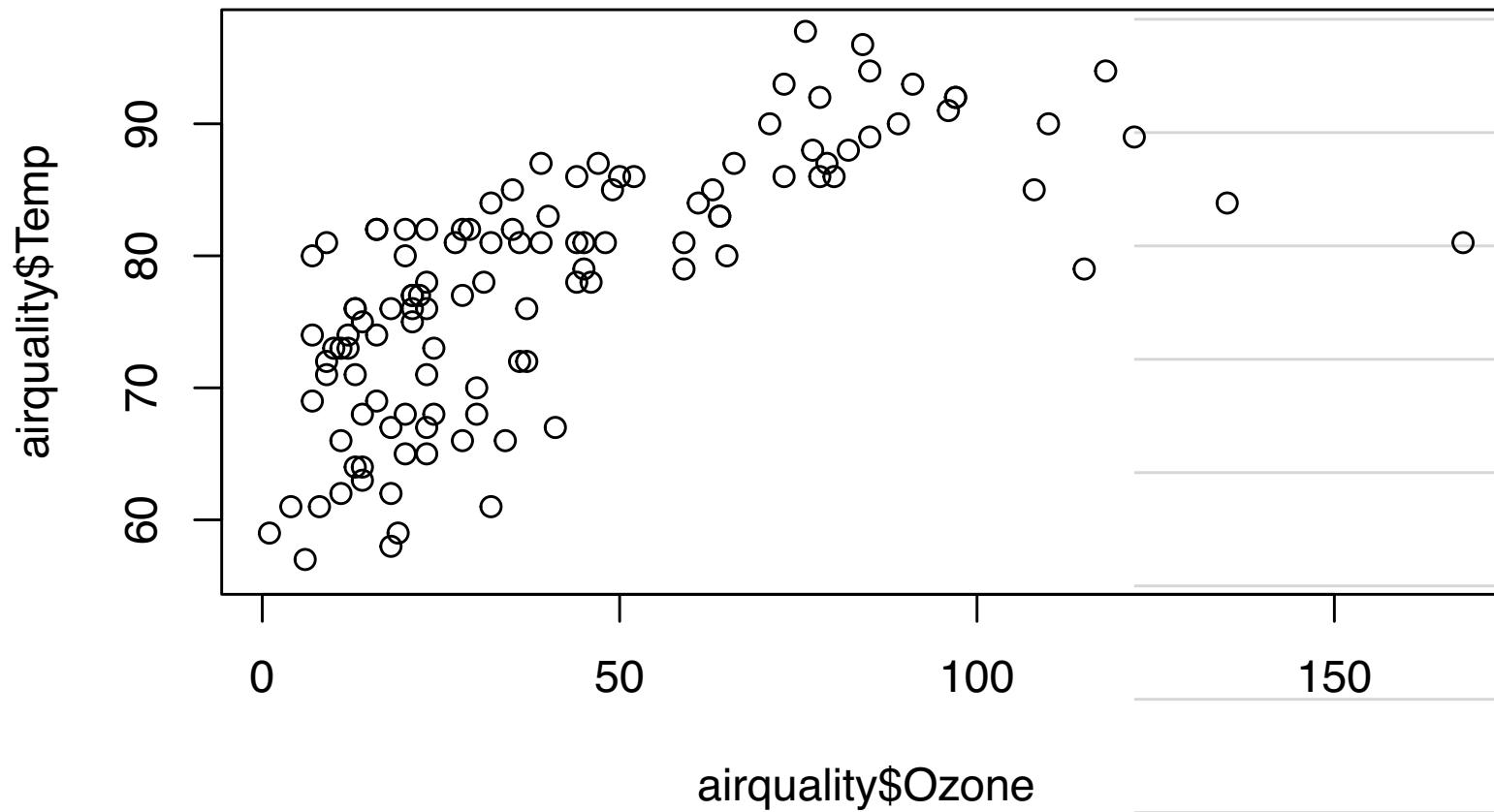
```
> plot(airquality$Temp)
```



Scatter Plot

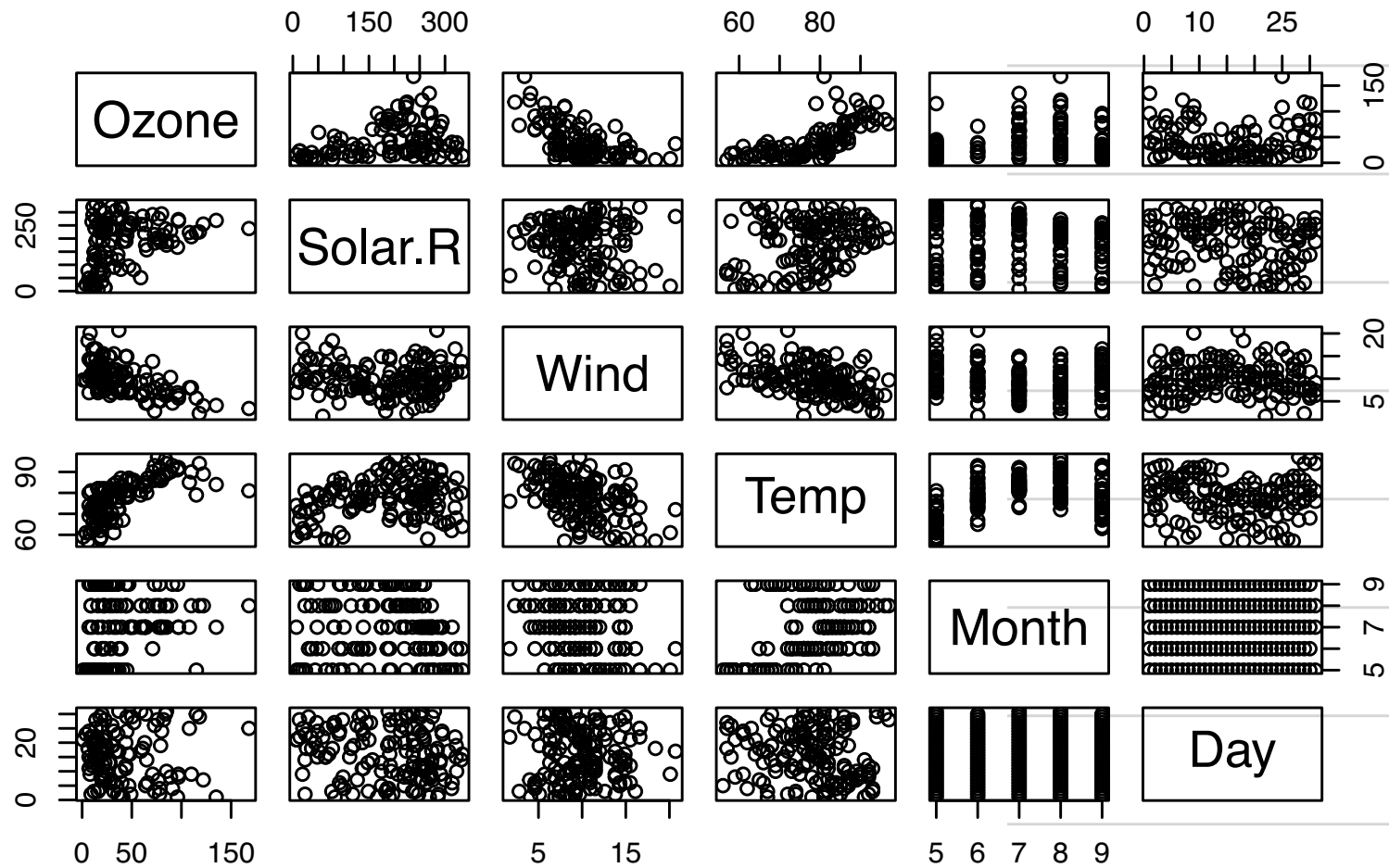
We can use the `plot` function to get a Scatter plot.

```
> plot(airquality$Ozone, airquality$Temp)
```



Plot!

```
> plot(airquality)
```



External Packages in R

R has can be enhanced with a lot of external packages that are available. The package `UsingR` has many datasets loaded in it.

```
> install.packages("UsingR")
```

Once installed then to add to current workspace

```
> library("UsingR")
```

ggplot2 implements grammar of graphics

```
> install.packages("tidyverse")
```

Once installed then to add to current workspace

```
> library("tidyverse")
```

Grammar of graphics

- system for describing
 & building graphs

- learn one system and
 apply it at many
 places.

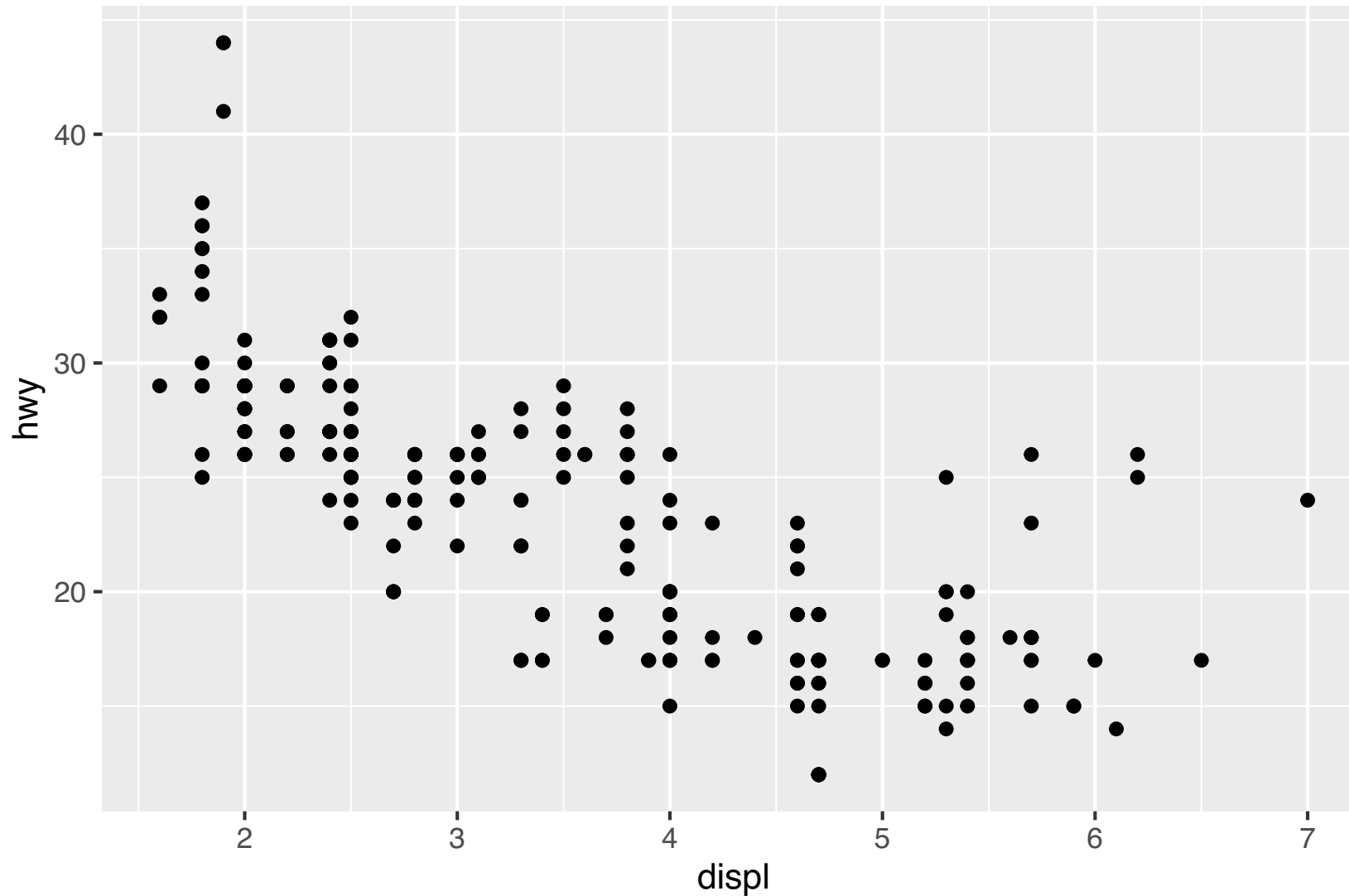
Dataset in tidyverse

> mpg

Observations collected by US Environment Protection Agency on 38 models of cars.

ggplot2- Data Visualisation

```
> ggplot(data=mpg) +  
+   geom_point(mapping=aes(x=displ, y=hwy))
```



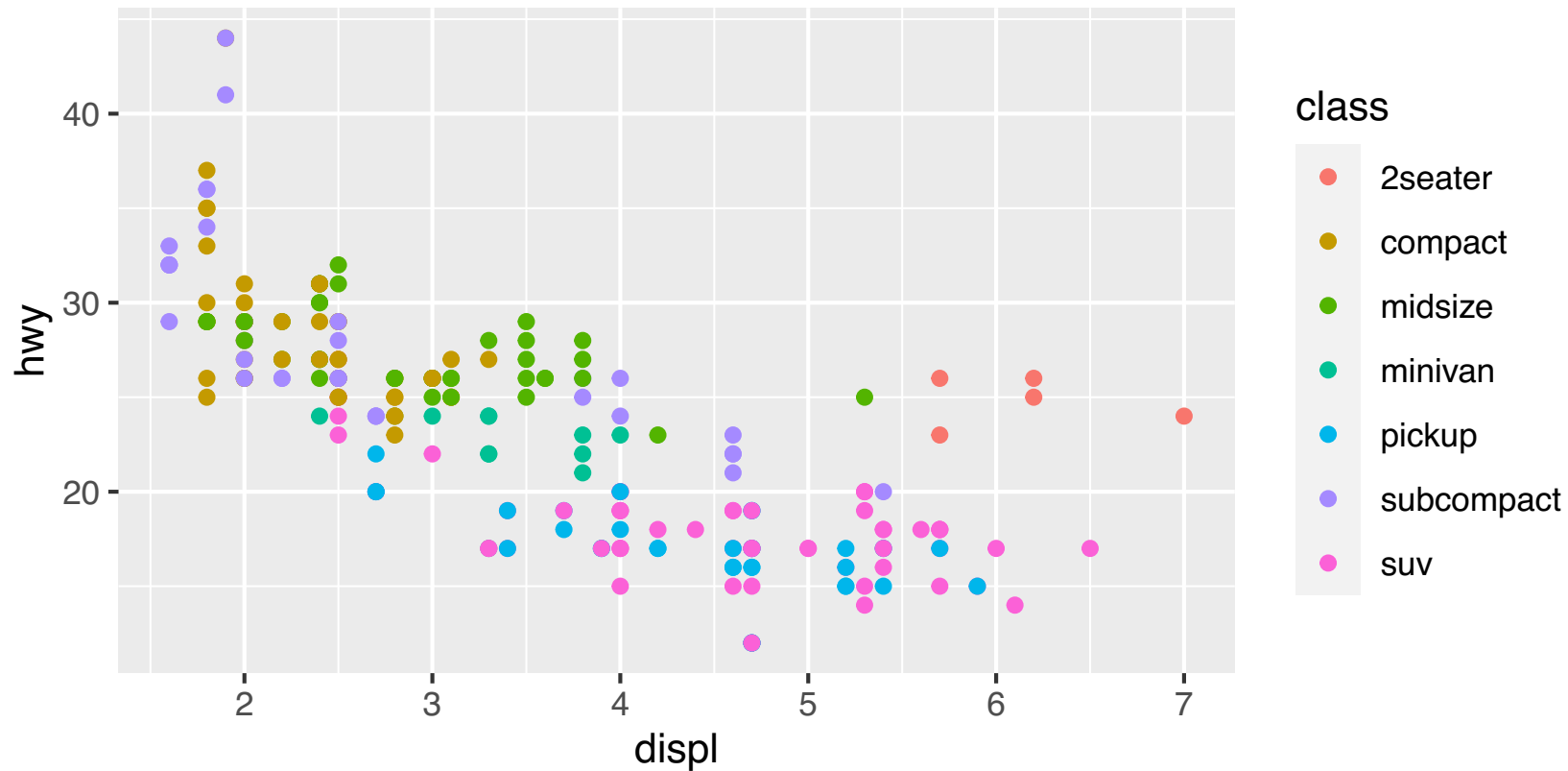
The plots negative relationship between Engine Size and Fuel Efficiency.

ggplot()

- Begins with a function `ggplot()`- creates a coordinate system that you can add `addlayers` to. The first argument is the data set to use `ggplot(data=mpg)` creates an empty graph.
- Add layers to `ggplot()`- the function `geom_point()` adds a layer of points to your plot
- Each geom function takes a `mapping` argument. The `mapping` argument is always paired with `aes()`
- `ggplot(data= <DATA>)+ GEOM-FUNCTION(mapping=aes(<MAPPINGS>))`
- We will learn how to complete and extend this basic template.

ggplot2- Aesthetics Mappings

```
> ggplot(data=mpg) +  
+   geom_point(mapping=aes(x=displ, y=hwy, colour=class))
```



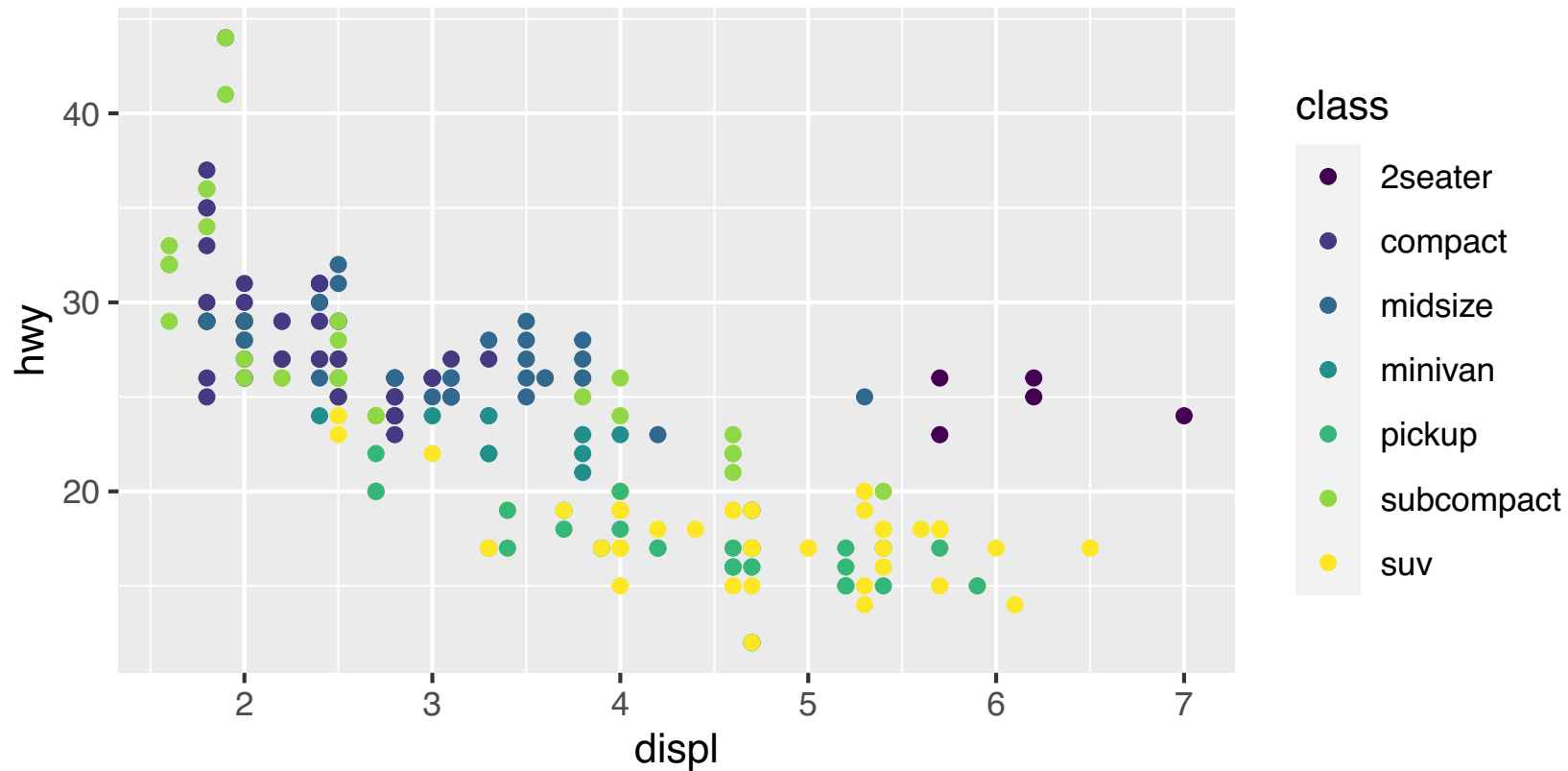
Added a third variable called `class` to a 2-D scatter plot by mapping it to an aesthetic.

- Map an aesthetic to a variable
- Associate the `name` of the aesthetic to the name of the `variable`.
- Above example `Name=colour` and `Variable=class`.
- **Scaling:** `ggplot2()` will assign a unique level of the aesthetic `colour` to a unique level to the variable `class`.
- `ggplot2` will also add a legend explaining the levels
- Other aesthetics include : shape and size.

- The viridis scales provide colour maps that are perceptually uniform in both colour and black-and-white.
- They are also designed to be perceived by viewers with common forms of colour blindness.
- See also <https://bids.github.io/colormap/>.

ggplot()-viridis options

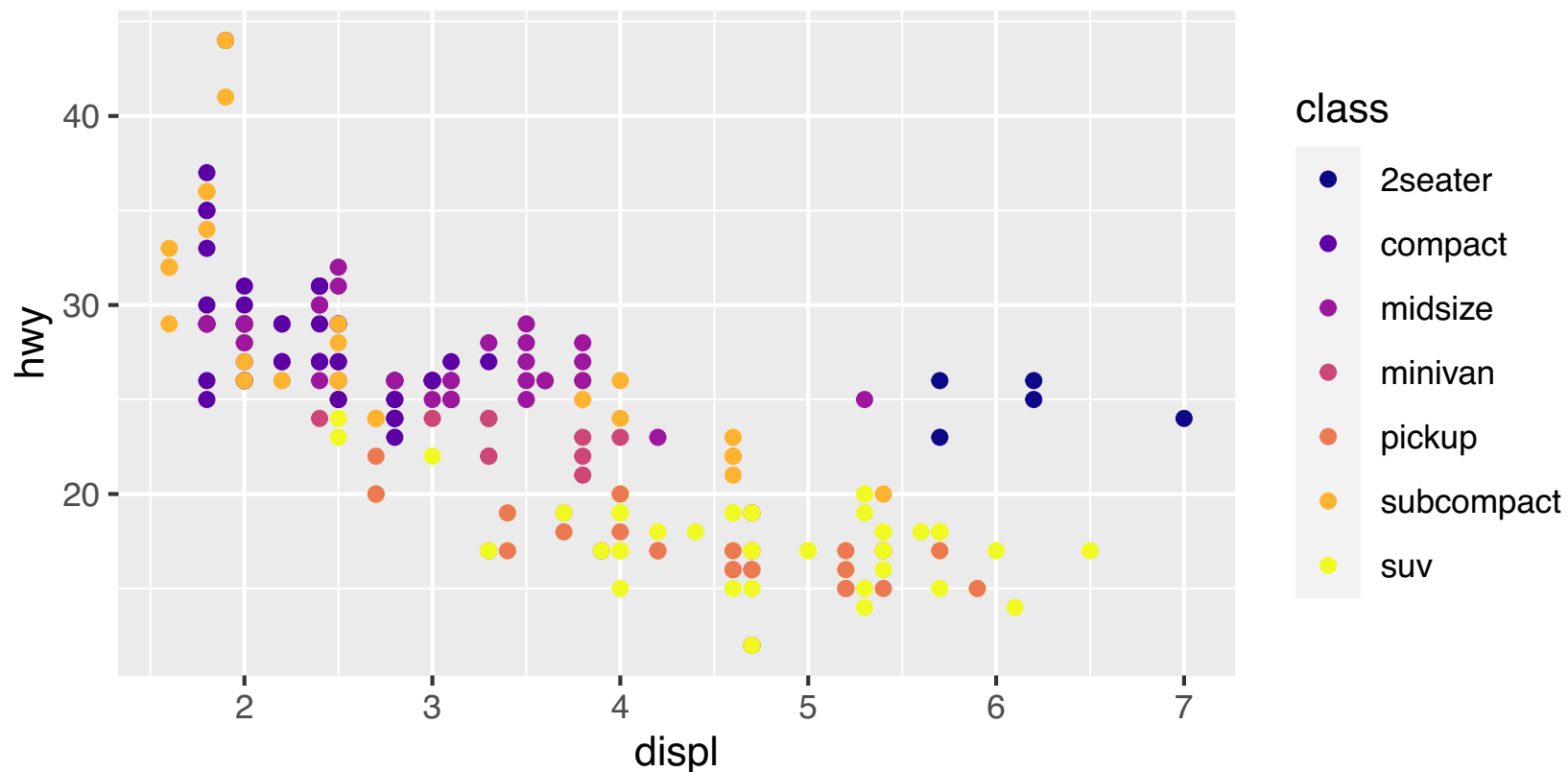
```
> ggplot(data=mpg) +  
+   geom_point(mapping=aes(x=displ, y=hwy, colour=class))+  
+   scale_colour_viridis_d()
```



Using colour palette from viridis package (colour blind colours).

ggplot()-viridis options

```
> ggplot(data=mpg) +  
+   geom_point(mapping=aes(x=displ, y=hwy, colour=class))+  
+   scale_colour_viridis_d(option = "plasma")
```



Using colour palette from viridis package (colour blind colours).

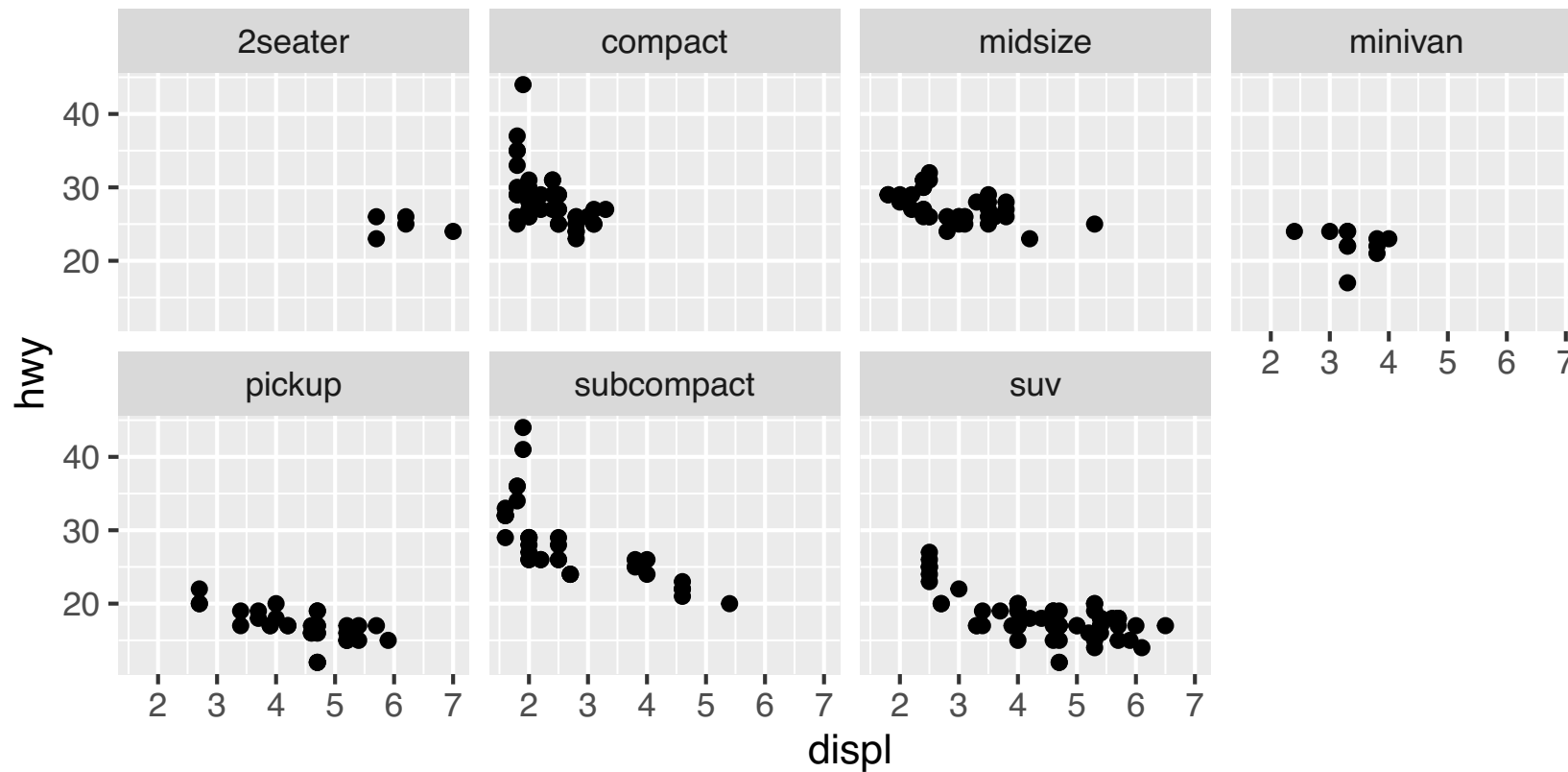
Can add

• aesthetic = shape

- As `aesthetics` was used to add an additional variable to the plot, another way is to add facets (useful for categorical variables).
- `facet_wrap` splits plot by a single variable into subplots that each display one subset of the data.

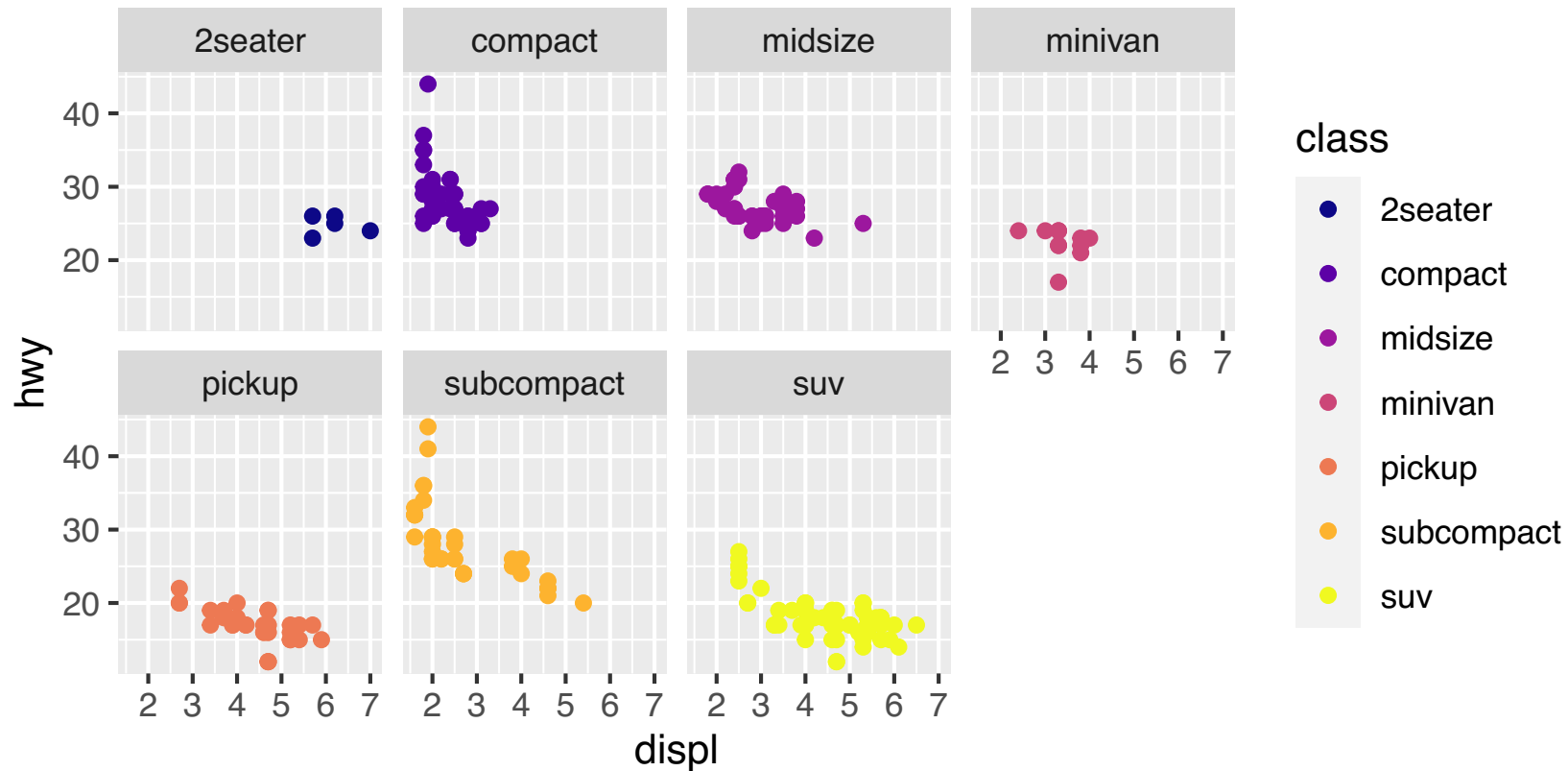
ggplot()-facets

```
> ggplot(data=mpg) +  
+   geom_point(mapping=aes(x=displ, y=hwy))+  
+   facet_wrap(~ class, nrow=2)
```



ggplot()-facets and Viridis

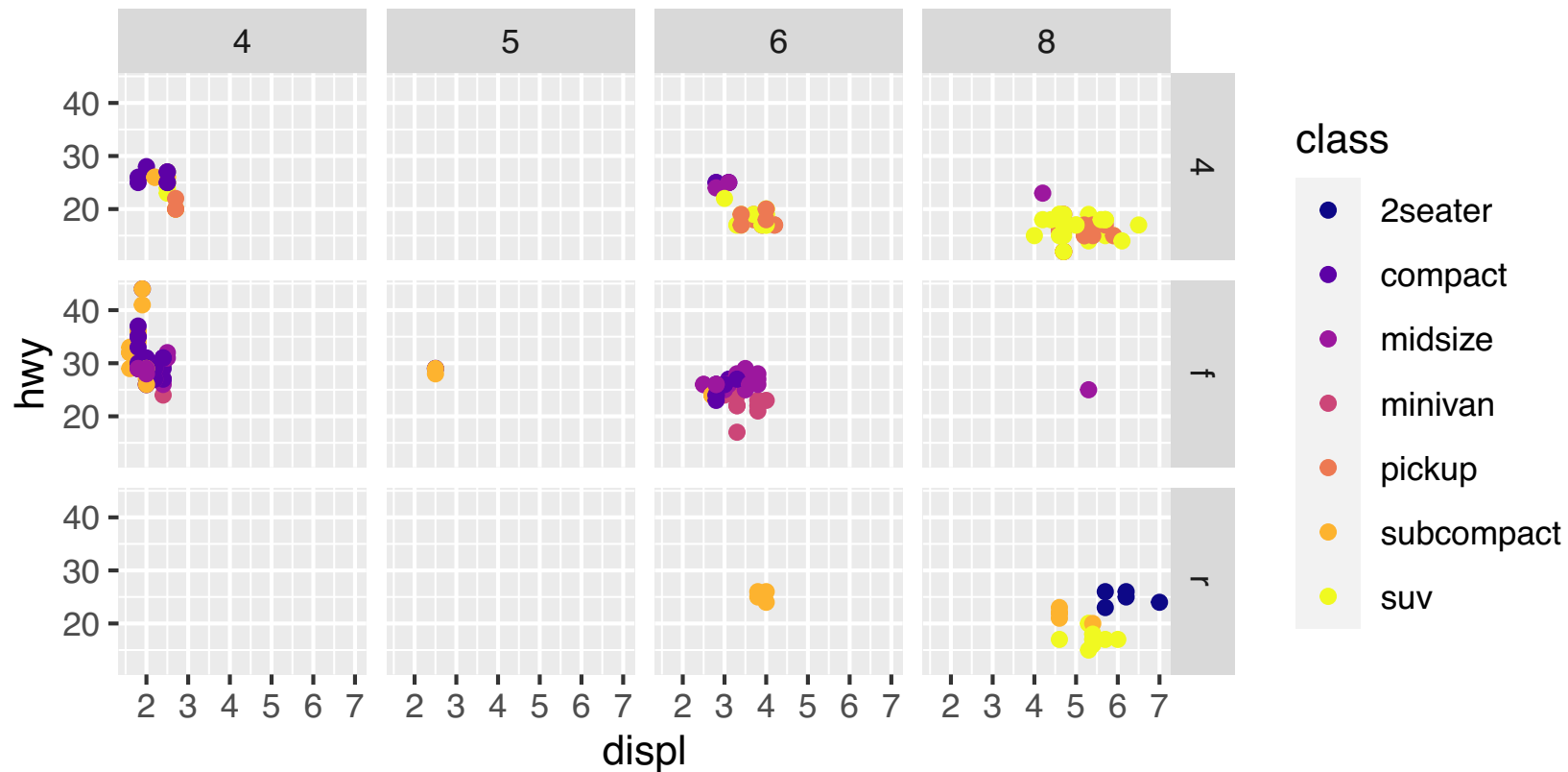
```
> ggplot(data=mpg) +  
+   geom_point(mapping=aes(x=displ, y=hwy, colour=class))+  
+   scale_colour_viridis_d(option = "plasma")+  
+   facet_wrap(~class, nrow=2)
```



`facet_wrap` splits plot by a single variable into subplots that each display one subset of the data.

ggplot()-facets

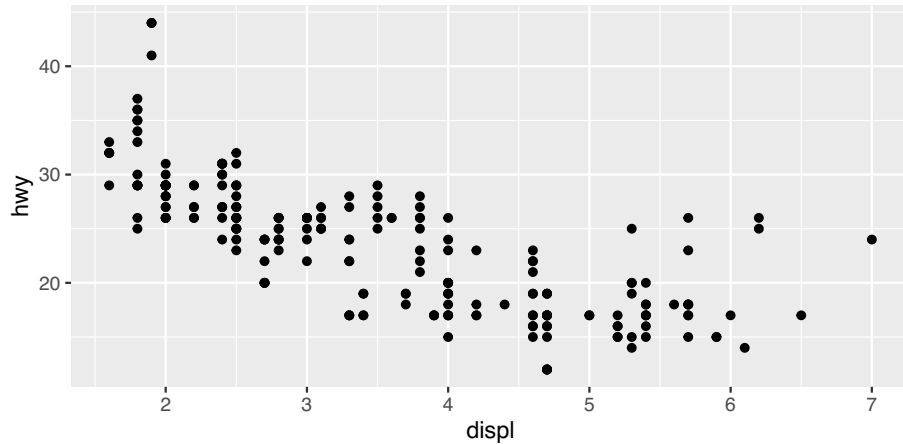
```
> ggplot(data=mpg) +  
+   geom_point(mapping=aes(x=displ, y=hwy, colour=class))+  
+   scale_colour_viridis_d(option = "plasma")+  
+   facet_grid(drv~cyl)
```



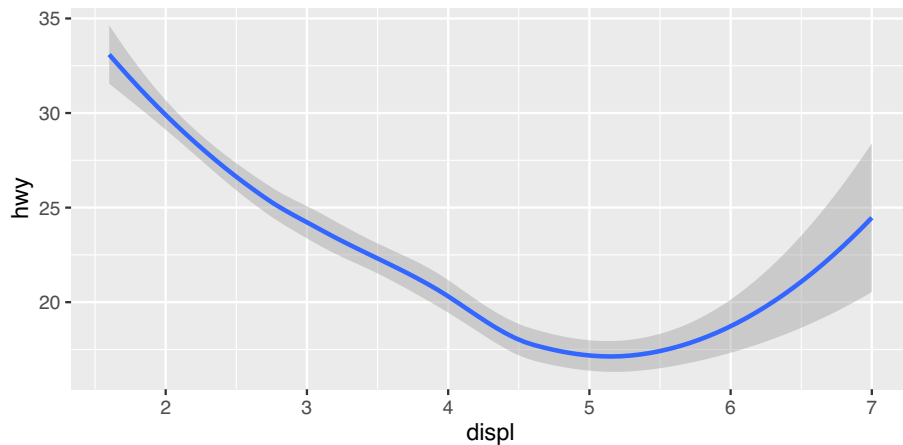
`facet_grid` splits plot by a combination of two variables into subplots that each display one subset of the data.

ggplot()-geoms

```
> ggplot(data = mpg) +  
+   geom_point(mapping = aes(x = displ, y = hwy))
```



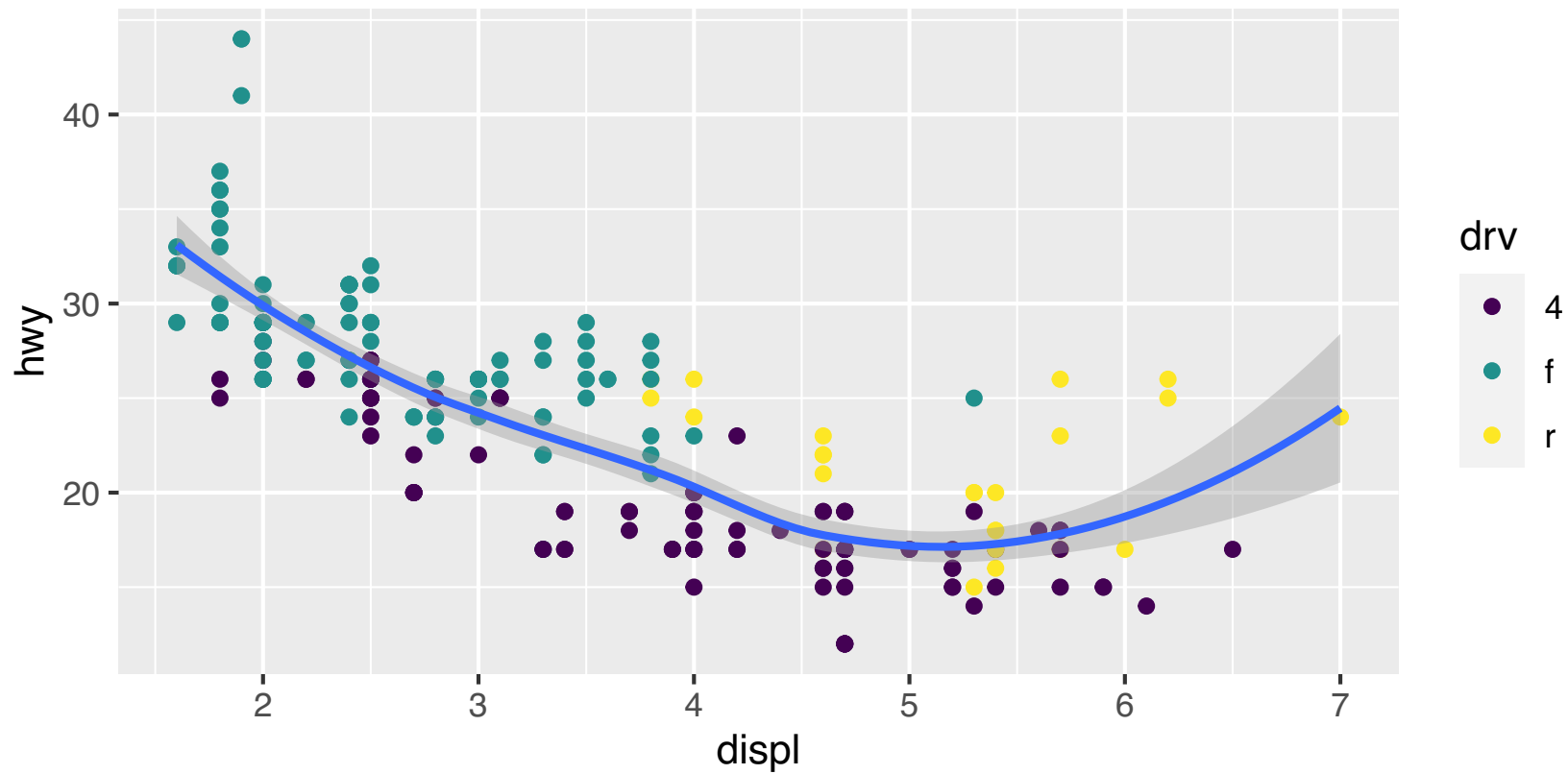
```
> ggplot(data = mpg) +  
+   geom_smooth(mapping = aes(x = displ, y = hwy))
```



a geom is the
geometrical object
that a plot uses
to represent
data.

ggplot()-geoms

```
> ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
+   geom_point(mapping = aes(color=drv))+  
+   geom_smooth()+  
+   scale_colour_viridis_d()
```



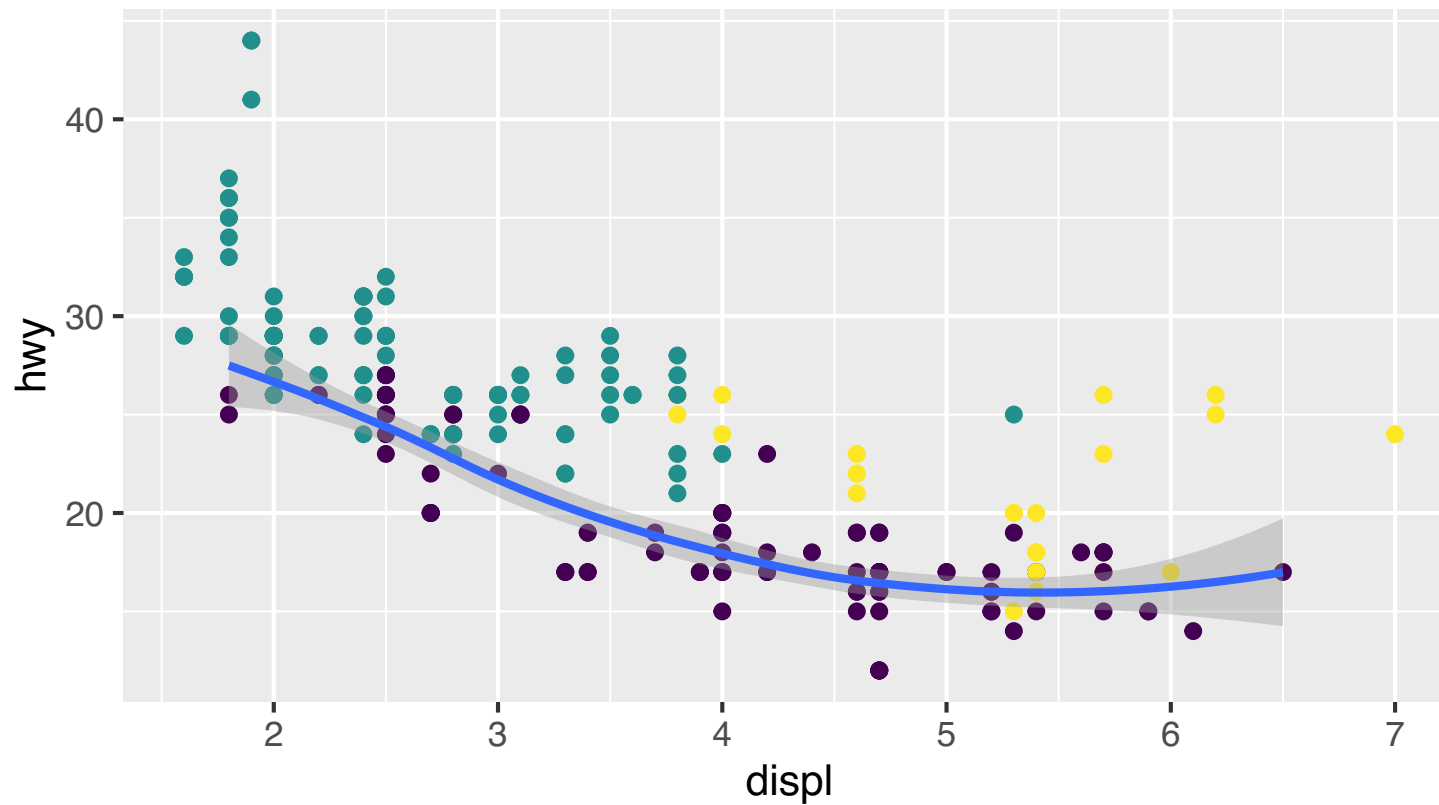
ggplot()-geoms

```
> ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
+   geom_point(mapping = aes(color=drv))+  
+   geom_smooth(data = filter(mpg, drv == "4"))+  
+   scale_colour_viridis_d()
```

filter command

in this layer

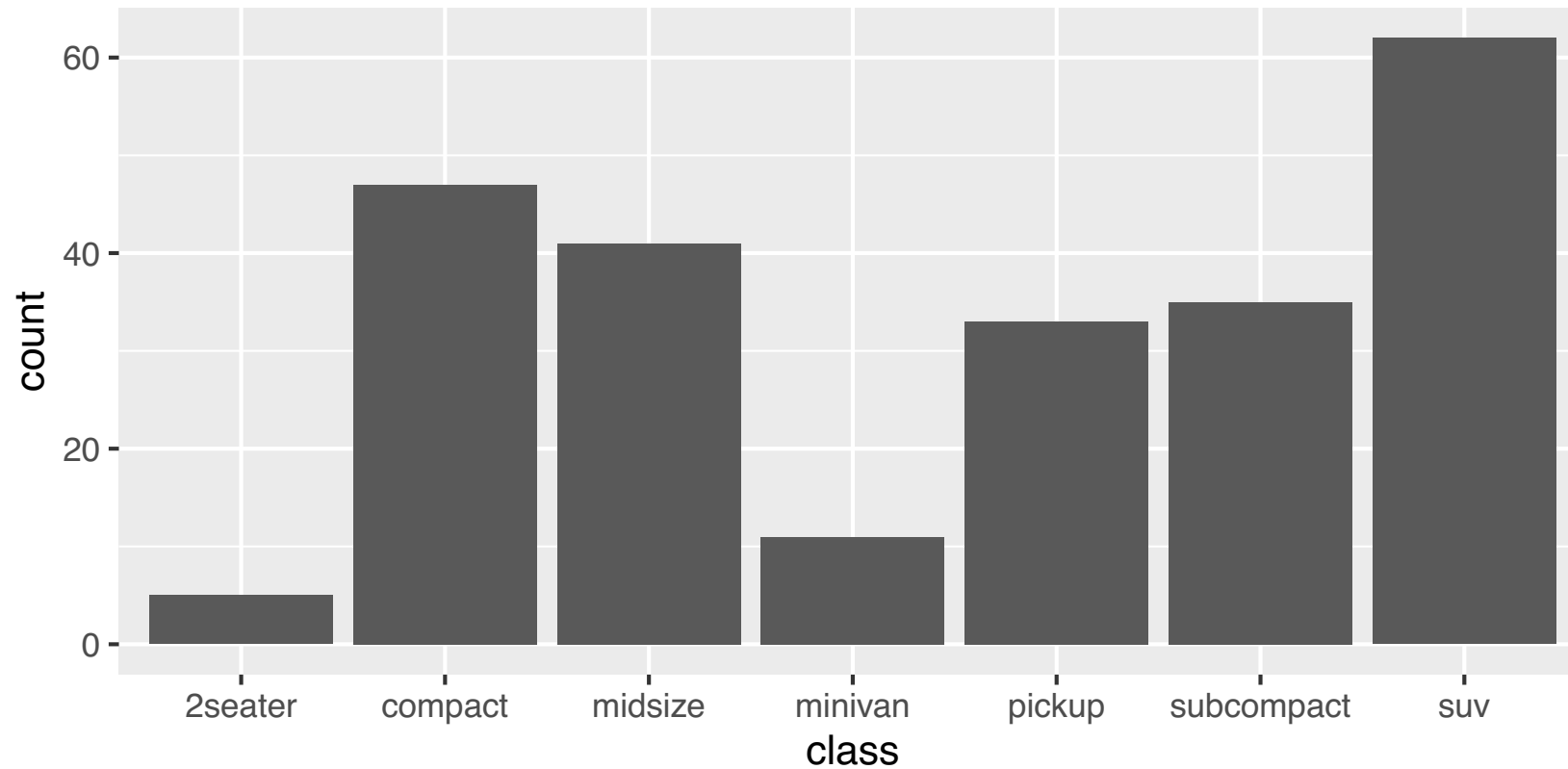
geom-smooth was
applied to only
4-wheel Drive



ggplot()-barchart

"Statistical transformation of data before display"

```
> ggplot(data = mpg) +  
+ stat_count(mapping= aes(x = class))
```

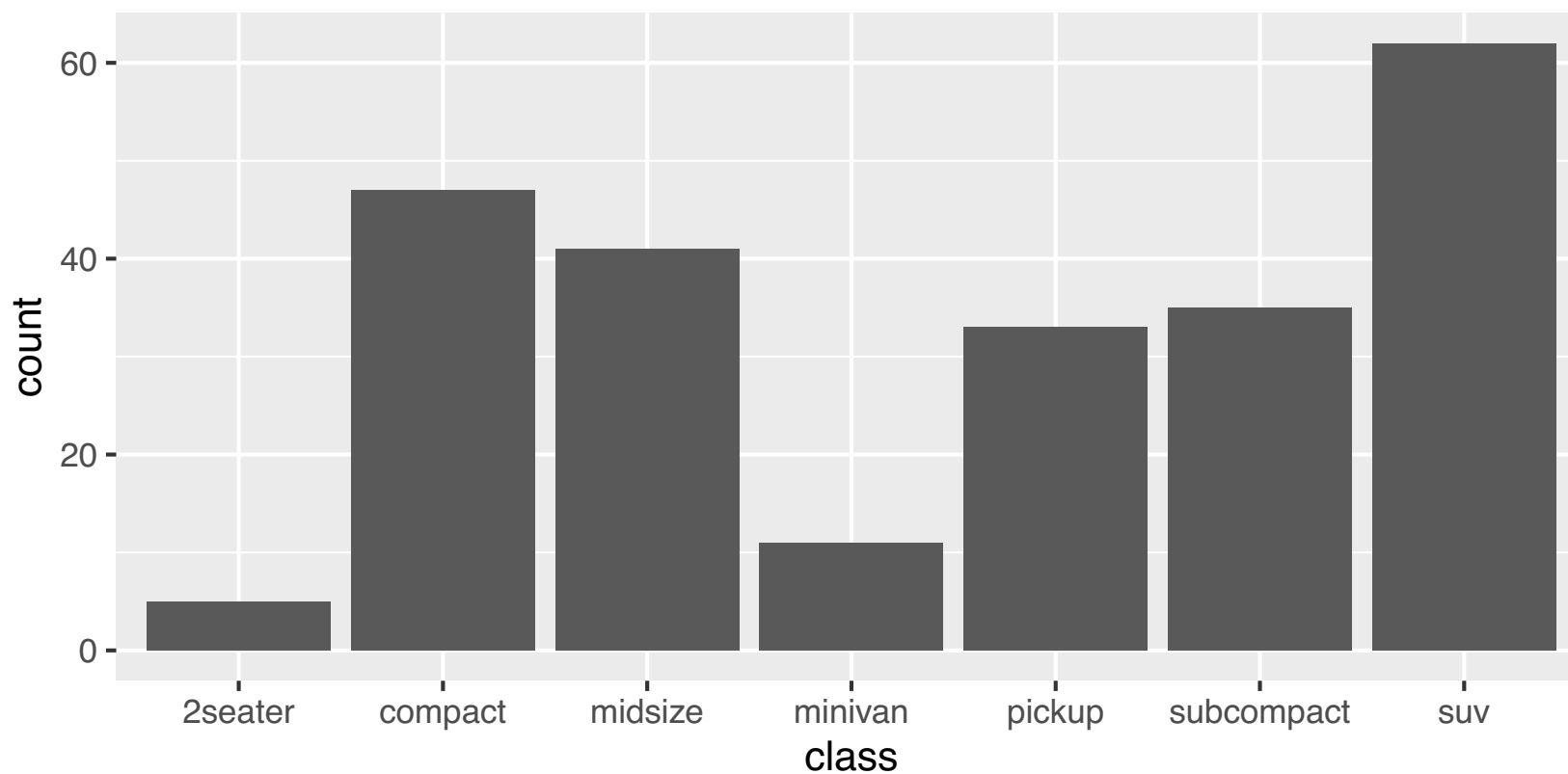


ggplot()-barchart

```
> table(mpg$class)
```

2seater	compact	midsize	minivan	pickup
5	47	41	11	33
subcompact	suv			
35	62			

```
> ggplot(data = mpg) +  
+ geom_bar( mapping= aes(x = class))
```



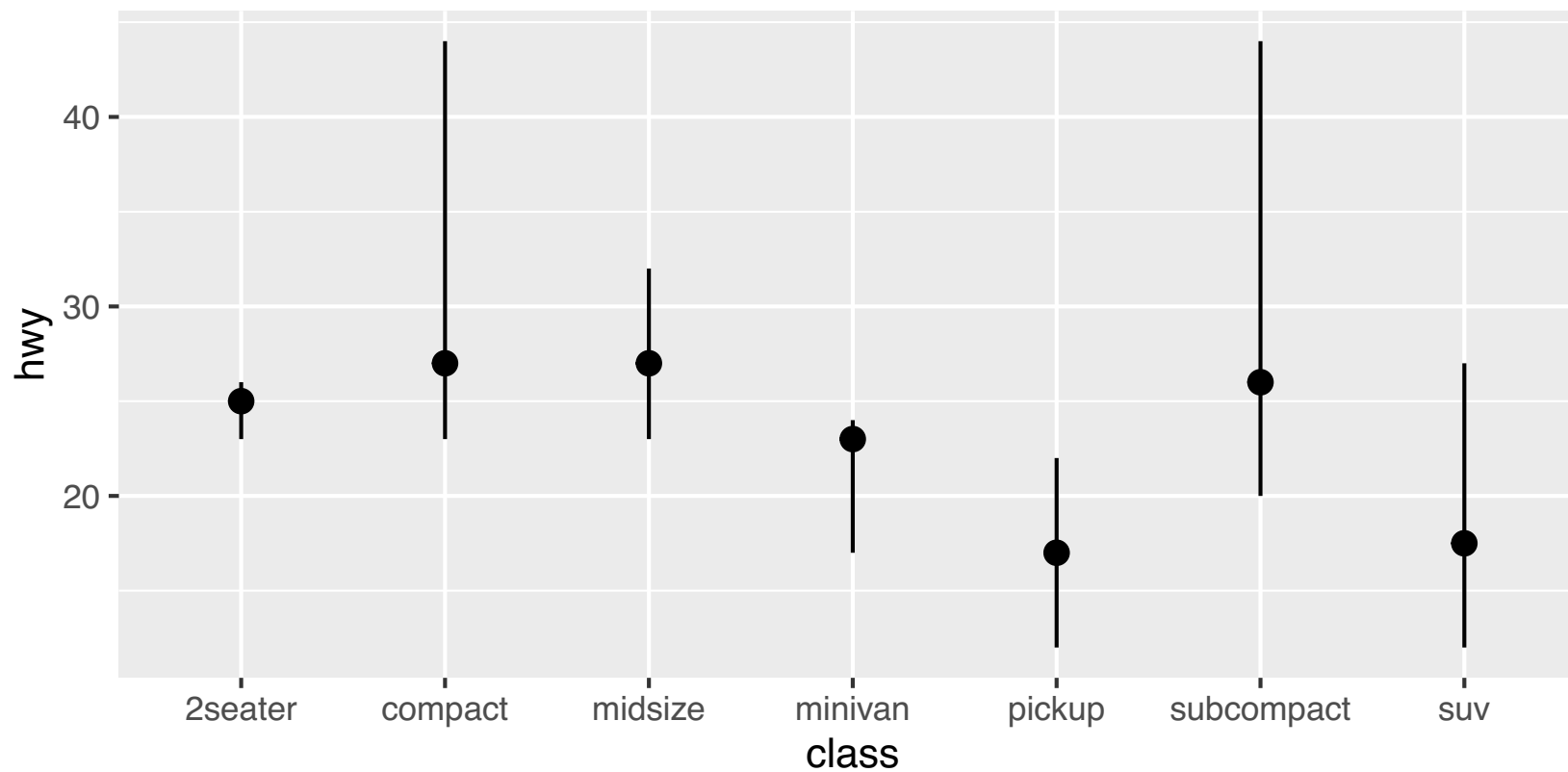
stat-count } & are
geom-bar } the
same.

ggplot()-barchart

```
> ggplot(data = mpg) +  
+   stat_summary(mapping=aes(x=class, y=hwy),  
+     fun.min = min,  
+     fun.max = max,  
+     fun = median  
+   )  
>
```

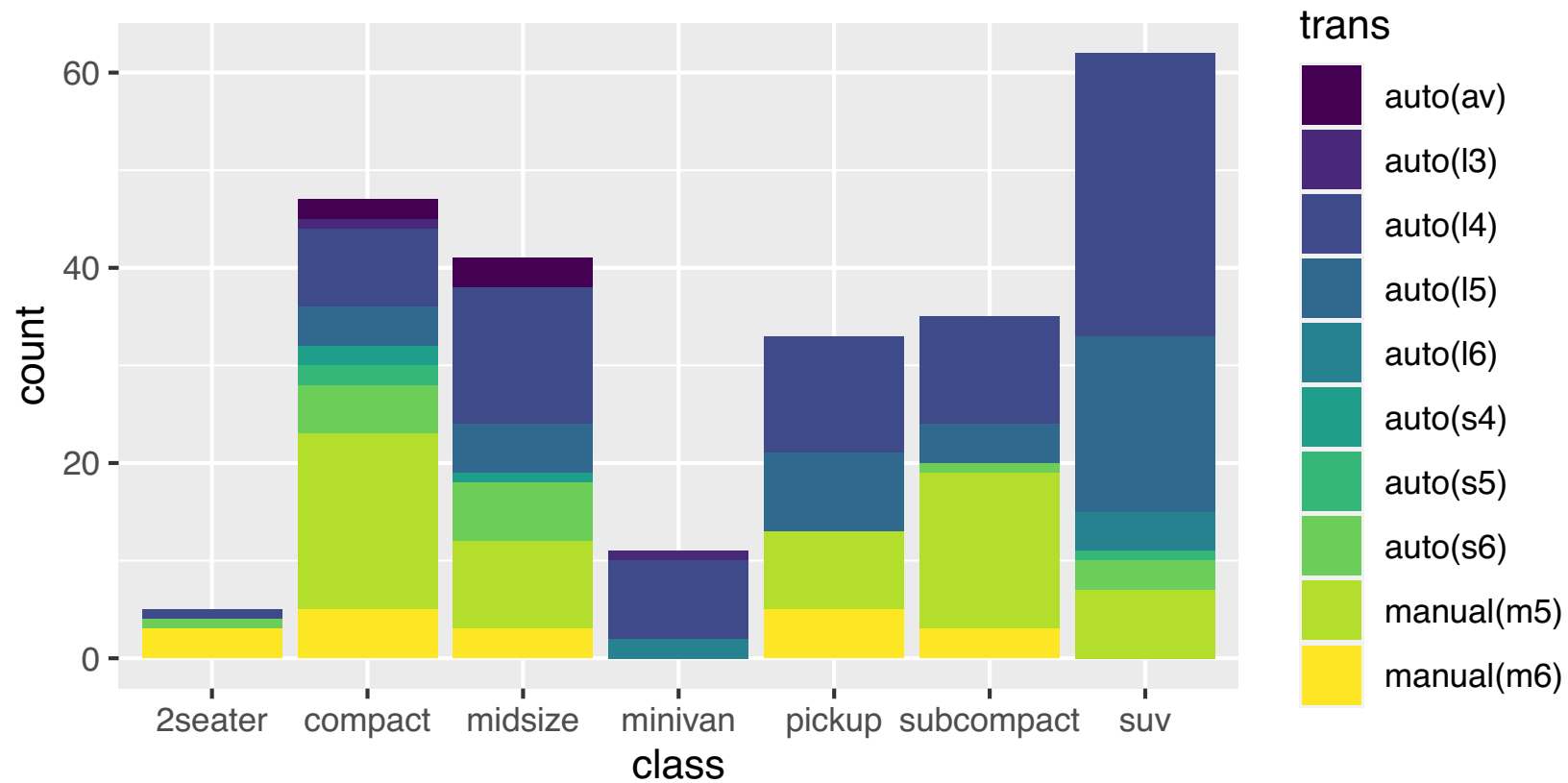
ggplot has over
20 stats.

? stat-bin



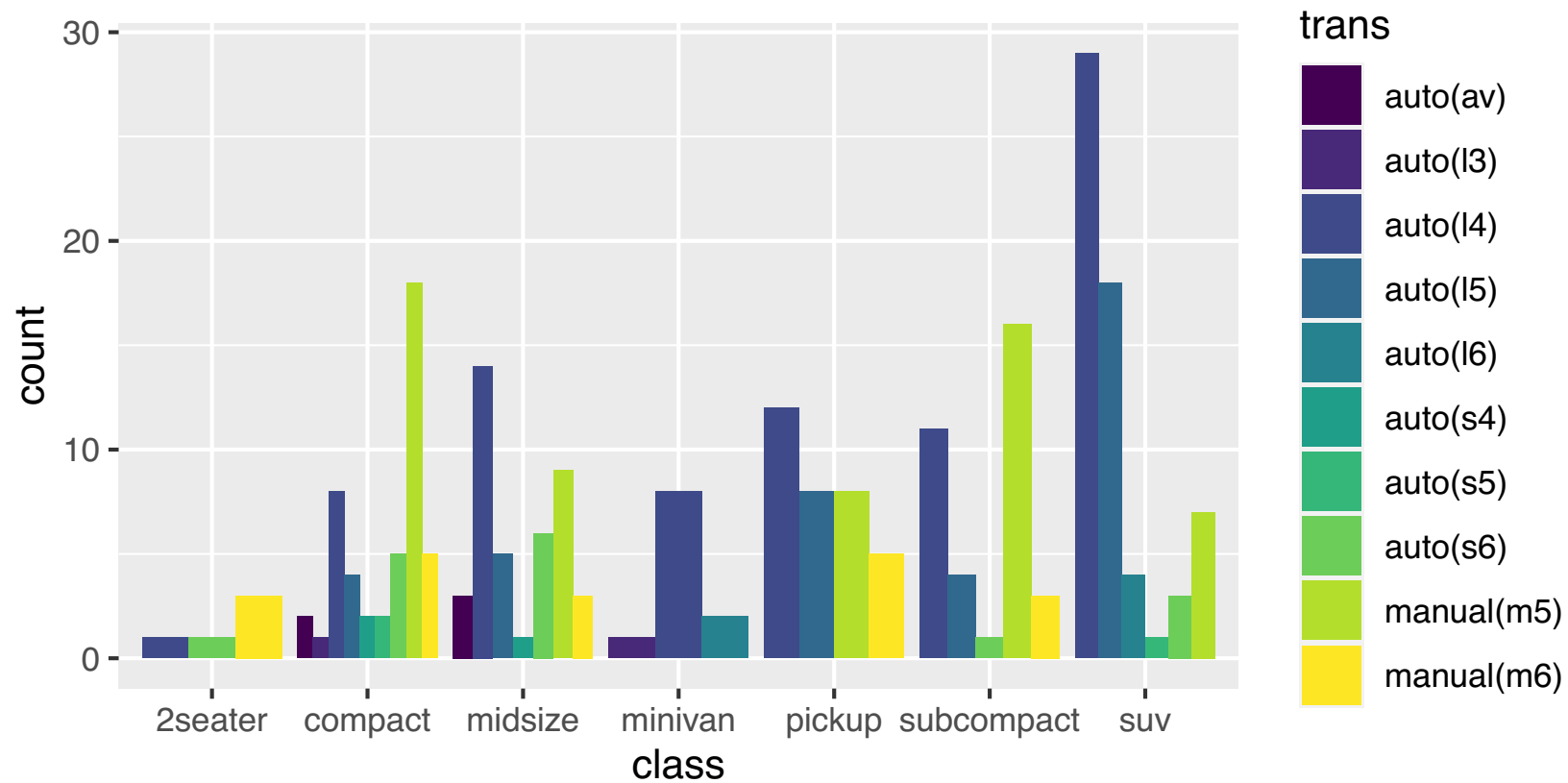
ggplot()-barchart

```
> ggplot(data = mpg) +  
+   geom_bar(mapping=aes(x=class,fill=trans))+  
+   scale_fill_viridis_d()  
>
```



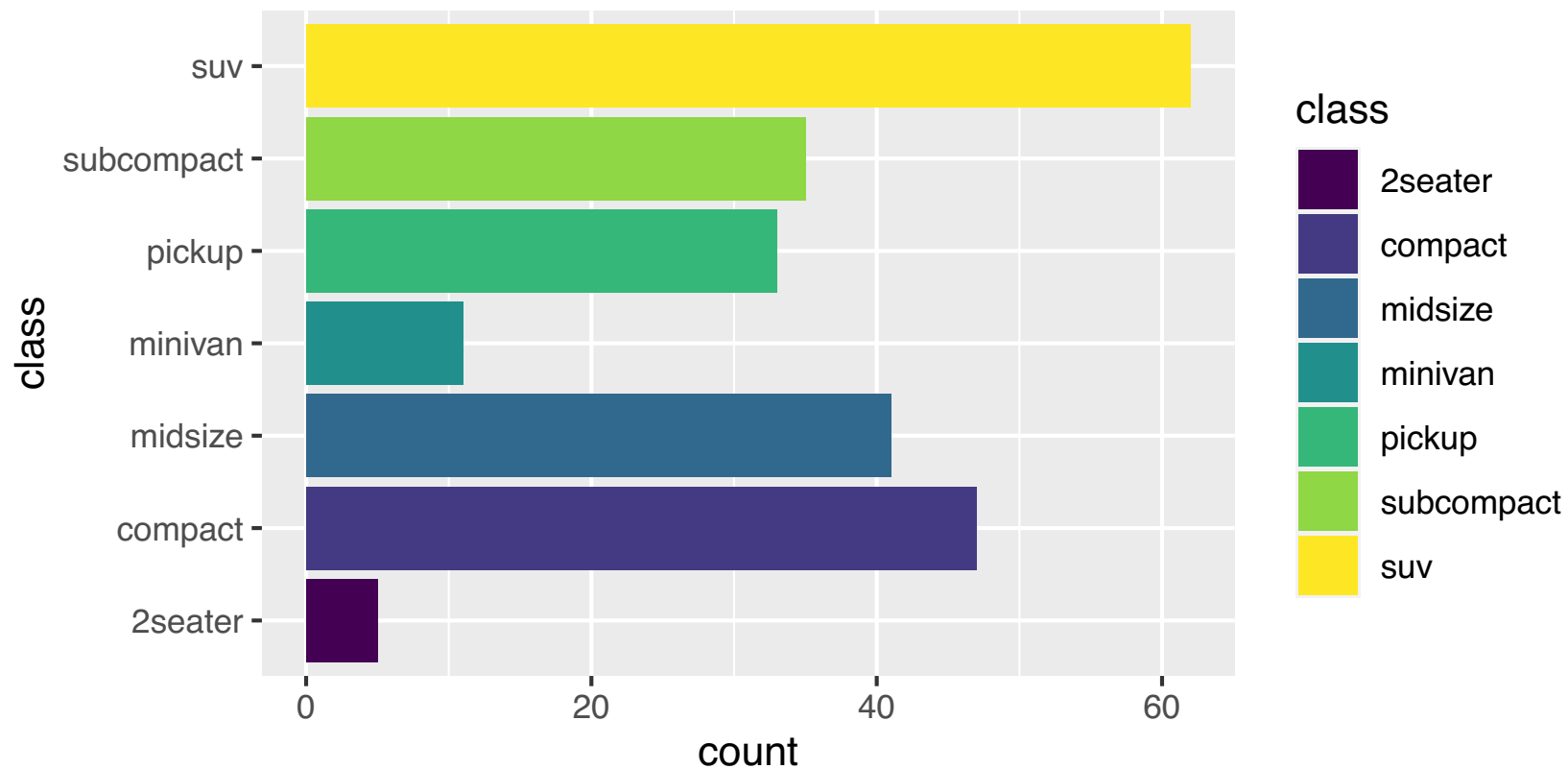
ggplot()-barchart

```
> ggplot(data = mpg) +  
+   geom_bar(mapping=aes(x=class,fill=trans),position="dodge")+  
+   scale_fill_viridis_d()  
>
```



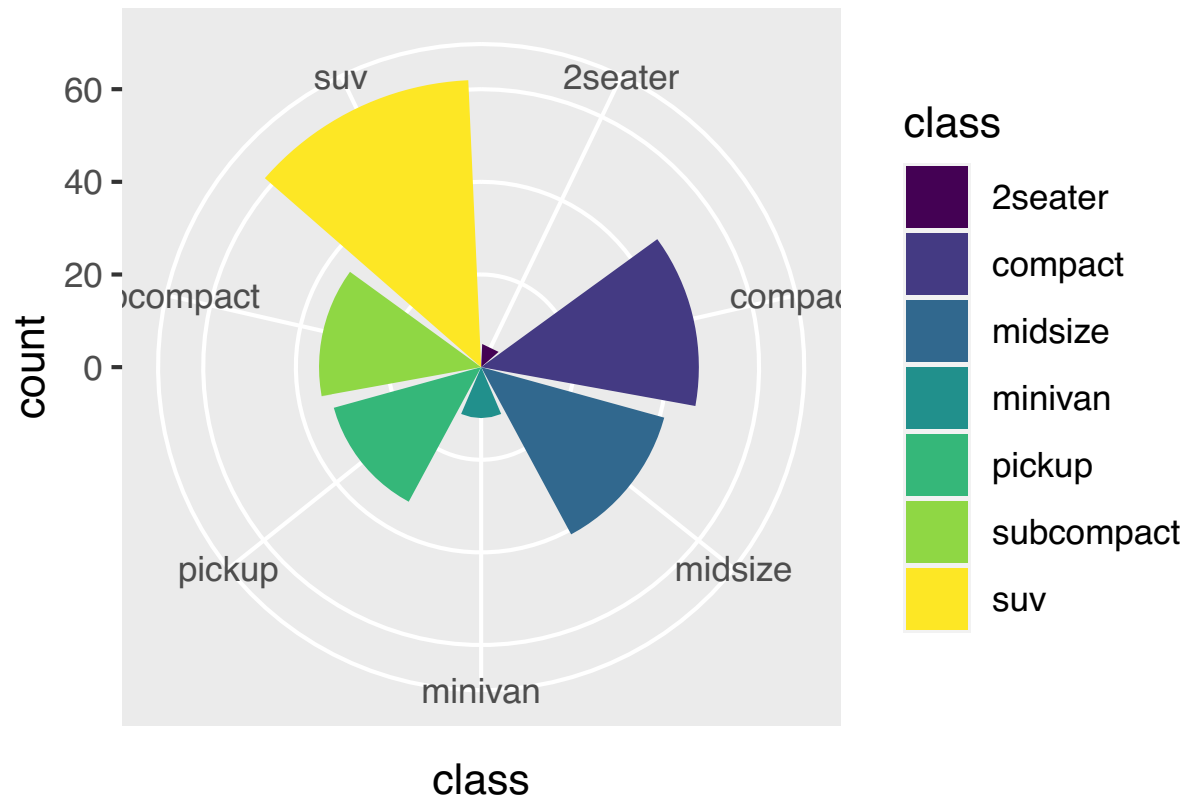
ggplot()-barchart

```
> ggplot(data = mpg) +  
+   geom_bar(mapping=aes(x=class,fill=class))+  
+   scale_fill_viridis_d() + coord_flip()
```



ggplot()-barchart

```
> ggplot(data = mpg) +  
+   geom_bar(mapping=aes(x=class,fill=class))+  
+   scale_fill_viridis_d() + coord_polar()
```



ggplot()- layered grammar of graphics

- `ggplot(data = <DATA>) +`
 - `<GEOM_FUNCTION>(
 mapping = aes(<MAPPINGS>),
 stat = <STAT>,
 position = <POSITION>
) +`
`<COORDINATE_FUNCTION> +`
`<FACET_FUNCTION>`

• Scatter plot

• Bar charts

• `geom`s

$\left. \begin{array}{l} : \\ : \\ : \end{array} \right\} \begin{array}{l} \text{many} \\ \text{more} \end{array}$

• `viridis` colour

• `facets`

• `coordinate`

ggplot()- layered grammar of graphics

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION>(  
    mapping = aes(<MAPPINGS>),  
    stat = <STAT>,  
    position = <POSITION>  
  ) +  
  <COORDINATE_FUNCTION> +  
  <FACET_FUNCTION>
```

