Multivariate Data

In this section we are going to review the relationship between more than 2 variables. The analysis follow the logic presented in univariate and bivariate data analysis section.

Categorical data

We have already reviewed frequency tables and proportion tables on 1 categorical variable and on 2 categorical variables using table and prop.table functions. It is the same for 3 and more categorical variables.

For the next examples let's consider Cars93 data frame from MASS package presenting data from 93 cars on sale in the USA in 1993.

```
> library(MASS)
> attach(Cars93)
> head(Cars93)
  Manufacturer   Model    Type Min.Price Price Max.Price MPG.city MPG.highway
1        Acura Integra   Small      12.9  15.9      18.8       25          31
2        Acura  Legend Midsize      29.2  33.9      38.7       18          25
3         Audi      90 Compact      25.9  29.1      32.3       20          26
4         Audi     100 Midsize      30.8  37.7      44.6       19          26
5          BMW    535i Midsize      23.7  30.0      36.2       22          30
6        Buick Century Midsize      14.2  15.7      17.3       22          31
             AirBags DriveTrain Cylinders EngineSize Horsepower  RPM
1               None      Front         4        1.8        140 6300
2 Driver & Passenger      Front         6        3.2        200 5500
3        Driver only      Front         6        2.8        172 5500
4 Driver & Passenger      Front         6        2.8        172 5500
5        Driver only       Rear         4        3.5        208 5700
6        Driver only      Front         4        2.2        110 5200
  Rev.per.mile Man.trans.avail Fuel.tank.capacity Passengers Length Wheelbase
1         2890             Yes               13.2          5    177       102
2         2335             Yes               18.0          5    195       115
3         2280             Yes               16.9          5    180       102
4         2535             Yes               21.1          6    193       106
5         2545             Yes               21.1          4    186       109
6         2565              No               16.4          6    189       105
  Width Turn.circle Rear.seat.room Luggage.room Weight  Origin          Make
1    68          37           26.5           11   2705 non-USA Acura Integra
2    71          38           30.0           15   3560 non-USA  Acura Legend
3    67          37           28.0           14   3375 non-USA       Audi 90
4    70          37           31.0           17   3405 non-USA      Audi 100
5    69          39           27.0           13   3640 non-USA      BMW 535i
6    69          41           28.0           16   2880     USA Buick Century
```

Car's Manufacturer, Origin, Type and AirBags are categorical variables, so we can see their frequency tables

```
> table(Manufacturer)
Manufacturer
        Acura          Audi           BMW         Buick      Cadillac
            2             2             1             4             2
    Chevrolet      Chrylser      Chrysler         Dodge         Eagle
            8             1             2             6             2
         Ford           Geo         Honda       Hyundai      Infiniti
            8             2             3             4             1
        Lexus       Lincoln         Mazda Mercedes-Benz       Mercury
            2             2             5             2             2
   Mitsubishi        Nissan    Oldsmobile      Plymouth       Pontiac
            2             4             4             1             5
         Saab        Saturn        Subaru        Suzuki        Toyota
            1             1             3             1             4
   Volkswagen         Volvo
            4             2

> sort(table(Manufacturer), decreasing = TRUE)
Manufacturer
    Chevrolet          Ford         Dodge         Mazda       Pontiac
            8             8             6             5             5
        Buick       Hyundai        Nissan    Oldsmobile        Toyota
            4             4             4             4             4
   Volkswagen         Honda        Subaru         Acura          Audi
            4             3             3             2             2
     Cadillac      Chrysler         Eagle           Geo         Lexus
            2             2             2             2             2
      Lincoln Mercedes-Benz       Mercury    Mitsubishi         Volvo
            2             2             2             2             2
          BMW      Chrylser      Infiniti      Plymouth          Saab
            1             1             1             1             1
       Saturn        Suzuki
            1             1
```

In our data set most of the cars are manufactured by Chevrolet, Ford, Dodge.

```
> table(Origin)
Origin
    USA non-USA
     48      45
> table(Origin) / length(Origin)
Origin
     USA  non-USA
0.516129 0.483871
> prop.table(table(Origin))
Origin
     USA  non-USA
0.516129 0.483871
> table(Type)
Type
Compact   Large Midsize   Small  Sporty     Van
     16      11      22      21      14       9
```

```
> sort(table(Type), decreasing = TRUE)
Type
Midsize   Small Compact  Sporty   Large     Van 
     22      21      16      14      11       9 
> Type <- factor(Type, levels = c("Small", "Compact", "Midsize", "Large", "Sporty", "Van"))
> table(Type)
Type
  Small Compact Midsize   Large  Sporty     Van 
     21      16      22      11      14       9 
```

Most of the cars in our data set are middle and small size.

```
> table(AirBags)
AirBags
Driver & Passenger        Driver only               None 
                16                 43                 34 
> sort(table(AirBags), decreasing = TRUE)
AirBags
       Driver only               None Driver & Passenger 
                43                 34                 16 
```

Most of the cars in our data set are with airbag only for the driver

We can also see how the cars from one manufacturer are distributed upon origin, type and airbag type.

```
> table(Origin, Manufacturer)
         Manufacturer
Origin    Acura Audi BMW Buick Cadillac Chevrolet Chrylser Chrysler Dodge Eagle
  USA         0    0   0     4        2         8        1        2     6     2
  non-USA     2    2   1     0        0         0        0        0     0     0
         Manufacturer
Origin    Ford Geo Honda Hyundai Infiniti Lexus Lincoln Mazda Mercedes-Benz
  USA        8   0     0       0        0     0       2     0             0
  non-USA    0   2     3       4        1     2       0     5             2
         Manufacturer
Origin    Mercury Mitsubishi Nissan Oldsmobile Plymouth Pontiac Saab Saturn
  USA           2          0      0          4        1       5    0      1
  non-USA       0          2      4          0        0       0    1      0
         Manufacturer
Origin    Subaru Suzuki Toyota Volkswagen Volvo
  USA          0      0      0          0     0
  non-USA      3      1      4          4     2

> table(Type, Manufacturer)
         Manufacturer
Type      Acura Audi BMW Buick Cadillac Chevrolet Chrylser Chrysler Dodge Eagle
  Small       1    0   0     0        0         0        0        0     2     1
  Compact     0    1   0     0        0         2        0        1     1     0
  Midsize     1    1   1     2        1         1        0        0     1     0
  Large       0    0   0     2        1         1        1        1     0     1
  Sporty      0    0   0     0        0         2        0        0     1     0
```

```
  Van            0    0  0    0      0       2      0      0    1    0
         Manufacturer
Type      Ford Geo Honda Hyundai Infiniti Lexus Lincoln Mazda Mercedes-Benz
  Small      2   1     1       2        0     0       0     2             0
  Compact    1   0     1       0        0     0       0     1             1
  Midsize    1   0     0       1        1     2       1     0             1
  Large      1   0     0       0        0     0       1     0             0
  Sporty     2   1     1       1        0     0       0     1             0
  Van        1   0     0       0        0     0       0     1             0
         Manufacturer
Type      Mercury Mitsubishi Nissan Oldsmobile Plymouth Pontiac Saab Saturn
  Small         0          1      1          0        0       1    0      1
  Compact       0          0      1          1        0       1    1      0
  Midsize       1          1      1          1        0       1    0      0
  Large         0          0      0          1        0       1    0      0
  Sporty        1          0      0          0        1       1    0      0
  Van           0          0      1          1        0       0    0      0
         Manufacturer
Type      Subaru Suzuki Toyota Volkswagen Volvo
  Small        2      1      1          1     0
  Compact      1      0      0          1     1
  Midsize      0      0      1          0     1
  Large        0      0      0          0     0
  Sporty       0      0      1          1     0
  Van          0      0      1          1     0
```

```
> table(AirBags, Manufacturer)
                    Manufacturer
AirBags              Acura Audi BMW Buick Cadillac Chevrolet Chrylser Chrysler
  Driver & Passenger     1    1   0     0        1         1        1        1
  Driver only            0    1   1     4        1         3        0        1
  None                   1    0   0     0        0         4        0        0
                    Manufacturer
AirBags              Dodge Eagle Ford Geo Honda Hyundai Infiniti Lexus Lincoln
  Driver & Passenger     0     1    0   0     2       0        0     1       2
  Driver only            5     0    5   1     1       0        1     1       0
  None                   1     1    3   1     0       4        0     0       0
                    Manufacturer
AirBags              Mazda Mercedes-Benz Mercury Mitsubishi Nissan Oldsmobile
  Driver & Passenger     0             1       0          0      0          0
  Driver only            2             1       1          1      3          2
  None                   3             0       1          1      1          2
                    Manufacturer
AirBags              Plymouth Pontiac Saab Saturn Subaru Suzuki Toyota
  Driver & Passenger        0       2    0      0      0      0      0
  Driver only               0       0    1      1      1      0      4
  None                      1       3    0      0      2      1      0
                    Manufacturer
AirBags              Volkswagen Volvo
  Driver & Passenger          0     1
  Driver only                 0     1
  None                        4     0
```

Let's see how the different car types are distributed upon different origins and airbag types. We will also show their proportion tables.

By default significant digits are 7, for the next examples let's set them to 2.

```
> getOption("digits")
[1] 7
> options(digits = 2)
> table(Origin, Type)
         Type
Origin    Small Compact Midsize Large Sporty Van
  USA         7       7      10    11      8   5
  non-USA    14       9      12     0      6   4

> prop.table(table(Origin, Type))
         Type
Origin    Small Compact Midsize Large Sporty   Van
  USA     0.075   0.075   0.108 0.118  0.086 0.054
  non-USA 0.151   0.097   0.129 0.000  0.065 0.043

> prop.table(table(Origin, Type), 1)
         Type
Origin    Small Compact Midsize Large Sporty   Van
  USA     0.146   0.146   0.208 0.229  0.167 0.104
  non-USA 0.311   0.200   0.267 0.000  0.133 0.089

> prop.table(table(Origin, Type), 2)
         Type
Origin    Small Compact Midsize Large Sporty  Van
  USA      0.33    0.44    0.45  1.00   0.57 0.56
  non-USA  0.67    0.56    0.55  0.00   0.43 0.44

> table(AirBags, Type)
                    Type
AirBags              Small Compact Midsize Large Sporty Van
  Driver & Passenger     0       2       7     4      3   0
  Driver only            5       9      11     7      8   3
  None                  16       5       4     0      3   6

> prop.table(table(AirBags, Type))
                    Type
AirBags              Small Compact Midsize Large Sporty   Van
  Driver & Passenger 0.000   0.022   0.075 0.043  0.032 0.000
  Driver only        0.054   0.097   0.118 0.075  0.086 0.032
  None               0.172   0.054   0.043 0.000  0.032 0.065

> prop.table(table(AirBags, Type), 1)
                    Type
AirBags              Small Compact Midsize Large Sporty   Van
  Driver & Passenger 0.000   0.125   0.438 0.250  0.188 0.000
  Driver only        0.116   0.209   0.256 0.163  0.186 0.070
  None               0.471   0.147   0.118 0.000  0.088 0.176
```

```
> prop.table(table(AirBags, Type), 2)
                  Type
AirBags              Small Compact Midsize Large Sporty  Van
  Driver & Passenger      0.00    0.12    0.32  0.36   0.21 0.00
  Driver only             0.24    0.56    0.50  0.64   0.57 0.33
  None                    0.76    0.31    0.18  0.00   0.21 0.67
```

We can see that most of the Small and Van cars in our data set don't have an airbag, more than one half of the others have airbag only for the driver and 44% of the cars that have airbag for the driver and passenger are with midsize type and we don't have cars from the small and Van type that have airbag for the driver and passenger.

Similarly we can generate a frequency table for 3 and more categorical variables.

```
> table(DriveTrain, Type, Man.trans.avail)
, , Man.trans.avail = No

          Type
DriveTrain Small Compact Midsize Large Sporty Van
     4WD           0       0       0     0      0   3
     Front         0       2      10     7      0   3
     Rear          0       0       3     4      0   0

, , Man.trans.avail = Yes

          Type
DriveTrain Small Compact Midsize Large Sporty Van
     4WD           2       1       0     0      2   2
     Front        19      11       7     0      7   1
     Rear          0       2       2     0      5   0
```

We can also use xtabs function with the formula syntax

```
> xtabs(~Type)
Type
  Small Compact Midsize   Large  Sporty     Van
     21      16      22      11      14       9
```

```
> xtabs(~Manufacturer+Type)
               Type
Manufacturer    Small Compact Midsize Large Sporty Van
  Acura             1       0       1     0      0   0
  Audi              0       1       1     0      0   0
  BMW               0       0       1     0      0   0
  Buick             0       0       2     2      0   0
  Cadillac          0       0       1     1      0   0
  Chevrolet         0       2       1     1      2   2
  Chrylser          0       0       0     1      0   0
  Chrysler          0       1       0     1      0   0
  Dodge             2       1       1     0      1   1
  Eagle             1       0       0     1      0   0
  Ford              2       1       1     1      2   1
```

```
  Geo             1    0    0    0    1  0
  Honda           1    1    0    0    1  0
  Hyundai         2    0    1    0    1  0
  Infiniti        0    0    1    0    0  0
  Lexus           0    0    2    0    0  0
  Lincoln         0    0    1    1    0  0
  Mazda           2    1    0    0    1  1
  Mercedes-Benz   0    1    1    0    0  0
  Mercury         0    0    1    0    1  0
  Mitsubishi      1    0    1    0    0  0
  Nissan          1    1    1    0    0  1
  Oldsmobile      0    1    1    1    0  1
  Plymouth        0    0    0    0    1  0
  Pontiac         1    1    1    1    1  0
  Saab            0    1    0    0    0  0
  Saturn          1    0    0    0    0  0
  Subaru          2    1    0    0    0  0
  Suzuki          1    0    0    0    0  0
  Toyota          1    0    1    0    1  1
  Volkswagen      1    1    0    0    1  1
  Volvo           0    1    1    0    0  0
```

```
> xtabs(~Manufacturer+AirBags)
               AirBags
Manufacturer    Driver & Passenger Driver only None
  Acura                          1           0    1
  Audi                           1           1    0
  BMW                            0           1    0
  Buick                          0           4    0
  Cadillac                       1           1    0
  Chevrolet                      1           3    4
  Chrylser                       1           0    0
  Chrysler                       1           1    0
  Dodge                          0           5    1
  Eagle                          1           0    1
  Ford                           0           5    3
  Geo                            0           1    1
  Honda                          2           1    0
  Hyundai                        0           0    4
  Infiniti                       0           1    0
  Lexus                          1           1    0
  Lincoln                        2           0    0
  Mazda                          0           2    3
  Mercedes-Benz                  1           1    0
  Mercury                        0           1    1
  Mitsubishi                     0           1    1
  Nissan                         0           3    1
  Oldsmobile                     0           2    2
  Plymouth                       0           0    1
  Pontiac                        2           0    3
  Saab                           0           1    0
  Saturn                         0           1    0
```

Subaru	0	1	2
Suzuki	0	0	1
Toyota	0	4	0
Volkswagen	0	0	4
Volvo	1	1	0

```
> xtabs(~AirBags+Type)
```

AirBags \ Type	Small	Compact	Midsize	Large	Sporty	Van
Driver & Passenger	0	2	7	4	3	0
Driver only	5	9	11	7	8	3
None	16	5	4	0	3	6

or ftable function

```
> ftable(Type)
```

Type	Small	Compact	Midsize	Large	Sporty	Van
	21	16	22	11	14	9

```
> ftable(Manufacturer, Type)
```

Manufacturer \ Type	Small	Compact	Midsize	Large	Sporty	Van
Acura	1	0	1	0	0	0
Audi	0	1	1	0	0	0
BMW	0	0	1	0	0	0
Buick	0	0	2	2	0	0
Cadillac	0	0	1	1	0	0
Chevrolet	0	2	1	1	2	2
Chrylser	0	0	0	1	0	0
Chrysler	0	1	0	1	0	0
Dodge	2	1	1	0	1	1
Eagle	1	0	0	1	0	0
Ford	2	1	1	1	2	1
Geo	1	0	0	0	1	0
Honda	1	1	0	0	1	0
Hyundai	2	0	1	0	1	0
Infiniti	0	0	1	0	0	0
Lexus	0	0	2	0	0	0
Lincoln	0	0	1	1	0	0
Mazda	2	1	0	0	1	1
Mercedes-Benz	0	1	1	0	0	0
Mercury	0	0	1	0	1	0
Mitsubishi	1	0	1	0	0	0
Nissan	1	1	1	0	0	1
Oldsmobile	0	1	1	1	0	1
Plymouth	0	0	0	0	1	0
Pontiac	1	1	1	1	1	0
Saab	0	1	0	0	0	0
Saturn	1	0	0	0	0	0
Subaru	2	1	0	0	0	0
Suzuki	1	0	0	0	0	0

```
Toyota           1    0    1    0    1  1
Volkswagen       1    1    0    0    1  1
Volvo            0    1    1    0    0  0
```

```
> ftable(Manufacturer, AirBags)
              AirBags Driver & Passenger Driver only None
Manufacturer
Acura                                  1           0    1
Audi                                   1           1    0
BMW                                    0           1    0
Buick                                  0           4    0
Cadillac                               1           1    0
Chevrolet                              1           3    4
Chrylser                               1           0    0
Chrysler                               1           1    0
Dodge                                  0           5    1
Eagle                                  1           0    1
Ford                                   0           5    3
Geo                                    0           1    1
Honda                                  2           1    0
Hyundai                                0           0    4
Infiniti                               0           1    0
Lexus                                  1           1    0
Lincoln                                2           0    0
Mazda                                  0           2    3
Mercedes-Benz                          1           1    0
Mercury                                0           1    1
Mitsubishi                             0           1    1
Nissan                                 0           3    1
Oldsmobile                             0           2    2
Plymouth                               0           0    1
Pontiac                                2           0    3
Saab                                   0           1    0
Saturn                                 0           1    0
Subaru                                 0           1    2
Suzuki                                 0           0    1
Toyota                                 0           4    0
Volkswagen                             0           0    4
Volvo                                  1           1    0
```

```
> ftable(AirBags, Type)
                   Type Small Compact Midsize Large Sporty Van
AirBags
Driver & Passenger        0       2       7     4      3   0
Driver only               5       9      11     7      8   3
None                     16       5       4     0      3   6
```

```
> ftable(Manufacturer, AirBags, Type)
                                 Type Small Compact Midsize Large Sporty Van
Manufacturer  AirBags
Acura         Driver & Passenger          0       0       1     0      0   0
              Driver only                 0       0       0     0      0   0
```

	None	1	0	0	0	0	0
Audi	Driver & Passenger	0	0	1	0	0	0
	Driver only	0	1	0	0	0	0
	None	0	0	0	0	0	0
BMW	Driver & Passenger	0	0	0	0	0	0
	Driver only	0	0	1	0	0	0
	None	0	0	0	0	0	0
Buick	Driver & Passenger	0	0	0	0	0	0
	Driver only	0	0	2	2	0	0
	None	0	0	0	0	0	0
Cadillac	Driver & Passenger	0	0	1	0	0	0
	Driver only	0	0	0	1	0	0
	None	0	0	0	0	0	0
Chevrolet	Driver & Passenger	0	0	0	0	1	0
	Driver only	0	1	0	1	1	0
	None	0	1	1	0	0	2
Chrylser	Driver & Passenger	0	0	0	1	0	0
	Driver only	0	0	0	0	0	0
	None	0	0	0	0	0	0
Chrysler	Driver & Passenger	0	1	0	0	0	0
	Driver only	0	0	0	1	0	0
	None	0	0	0	0	0	0
Dodge	Driver & Passenger	0	0	0	0	0	0
	Driver only	1	1	1	0	1	1
	None	1	0	0	0	0	0
Eagle	Driver & Passenger	0	0	0	1	0	0
	Driver only	0	0	0	0	0	0
	None	1	0	0	0	0	0
Ford	Driver & Passenger	0	0	0	0	0	0
	Driver only	0	0	1	1	2	1
	None	2	1	0	0	0	0
Geo	Driver & Passenger	0	0	0	0	0	0
	Driver only	0	0	0	0	1	0
	None	1	0	0	0	0	0
Honda	Driver & Passenger	0	1	0	0	1	0
	Driver only	1	0	0	0	0	0
	None	0	0	0	0	0	0
Hyundai	Driver & Passenger	0	0	0	0	0	0
	Driver only	0	0	0	0	0	0
	None	2	0	1	0	1	0
Infiniti	Driver & Passenger	0	0	0	0	0	0
	Driver only	0	0	1	0	0	0
	None	0	0	0	0	0	0
Lexus	Driver & Passenger	0	0	1	0	0	0
	Driver only	0	0	1	0	0	0
	None	0	0	0	0	0	0
Lincoln	Driver & Passenger	0	0	1	1	0	0
	Driver only	0	0	0	0	0	0
	None	0	0	0	0	0	0
Mazda	Driver & Passenger	0	0	0	0	0	0
	Driver only	0	1	0	0	1	0
	None	2	0	0	0	0	1

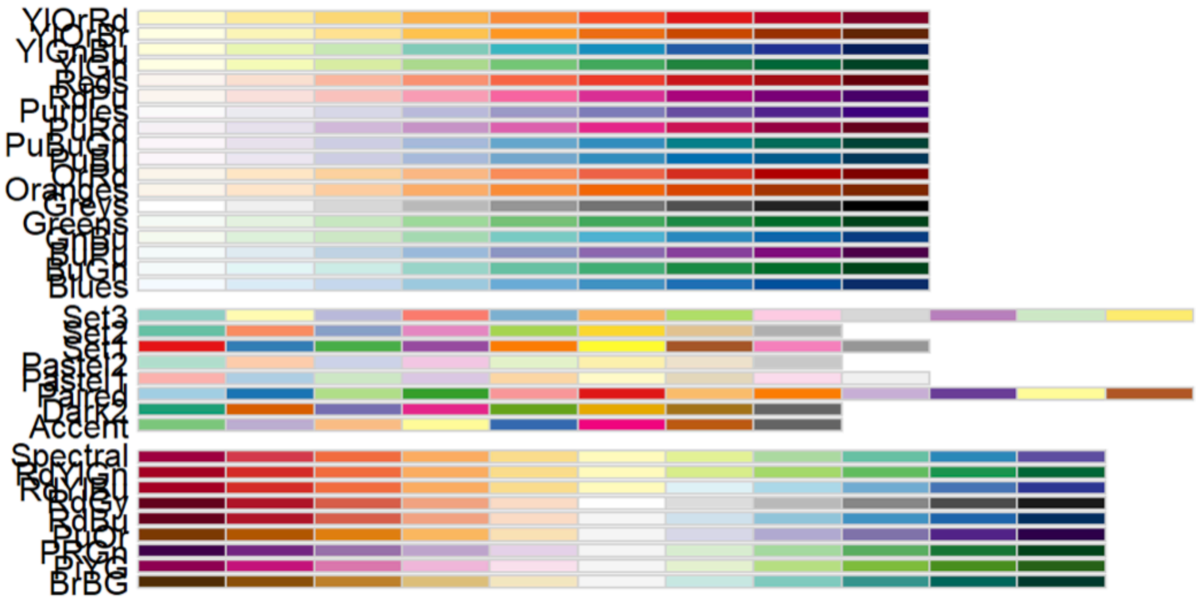
```
Mercedes-Benz Driver & Passenger   0  0  1  0  0 0
              Driver only          0  1  0  0  0 0
              None                 0  0  0  0  0 0
Mercury       Driver & Passenger   0  0  0  0  0 0
              Driver only          0  0  0  0  1 0
              None                 0  0  1  0  0 0
Mitsubishi    Driver & Passenger   0  0  0  0  0 0
              Driver only          0  0  1  0  0 0
              None                 1  0  0  0  0 0
Nissan        Driver & Passenger   0  0  0  0  0 0
              Driver only          1  1  1  0  0 0
              None                 0  0  0  0  0 1
Oldsmobile    Driver & Passenger   0  0  0  0  0 0
              Driver only          0  0  1  1  0 0
              None                 0  1  0  0  0 1
Plymouth      Driver & Passenger   0  0  0  0  0 0
              Driver only          0  0  0  0  0 0
              None                 0  0  0  0  1 0
Pontiac       Driver & Passenger   0  0  0  1  1 0
              Driver only          0  0  0  0  0 0
              None                 1  1  1  0  0 0
Saab          Driver & Passenger   0  0  0  0  0 0
              Driver only          0  1  0  0  0 0
              None                 0  0  0  0  0 0
Saturn        Driver & Passenger   0  0  0  0  0 0
              Driver only          1  0  0  0  0 0
              None                 0  0  0  0  0 0
Subaru        Driver & Passenger   0  0  0  0  0 0
              Driver only          0  1  0  0  0 0
              None                 2  0  0  0  0 0
Suzuki        Driver & Passenger   0  0  0  0  0 0
              Driver only          0  0  0  0  0 0
              None                 1  0  0  0  0 0
Toyota        Driver & Passenger   0  0  0  0  0 0
              Driver only          1  0  1  0  1 1
              None                 0  0  0  0  0 0
Volkswagen    Driver & Passenger   0  0  0  0  0 0
              Driver only          0  0  0  0  0 0
              None                 1  1  0  0  1 1
Volvo         Driver & Passenger   0  0  1  0  0 0
              Driver only          0  1  0  0  0 0
              None                 0  0  0  0  0 0
```

The plots that we use for categorical variables are barplot and piechart.
We are going to use RColorBrewer package for some predefined color palettes.

```
> install.packages("RColorBrewer")
> ? RColorBrewer
```

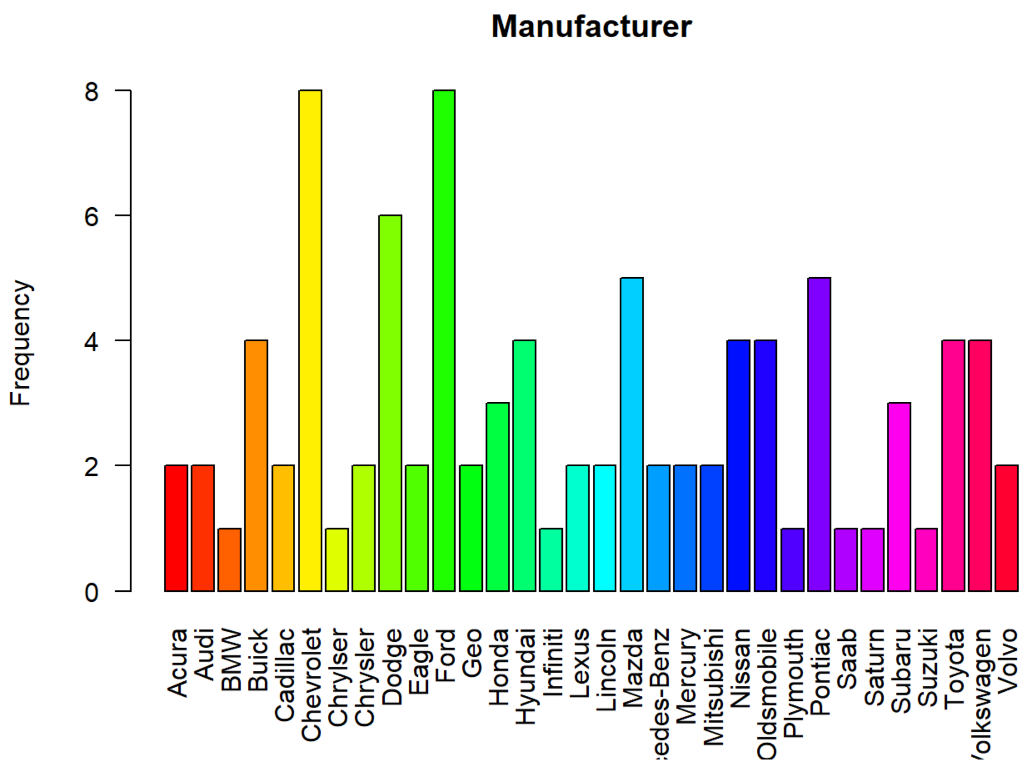
You can visualize the palettes using the display.brewer.all() function.

> library(RColorBrewer)
> display.brewer.all()

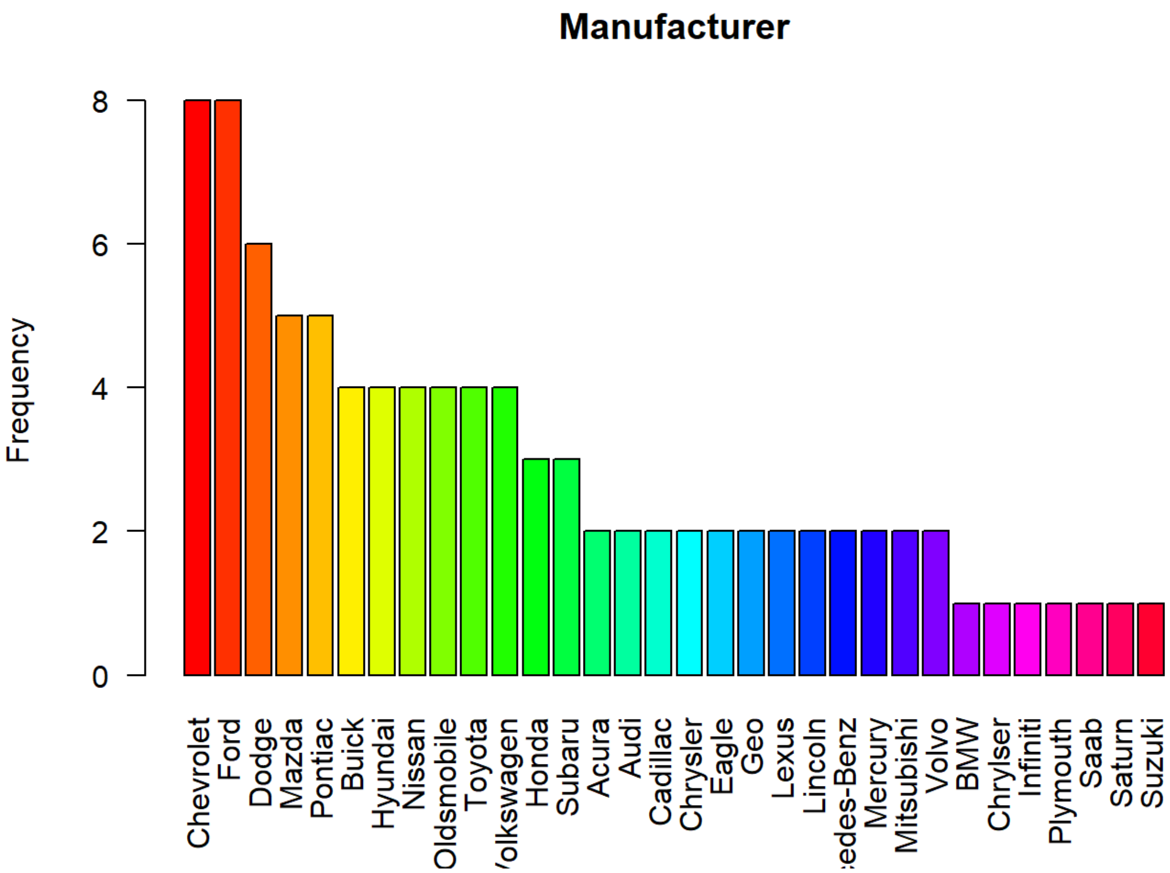


Following are the barplots of these categorical variables.

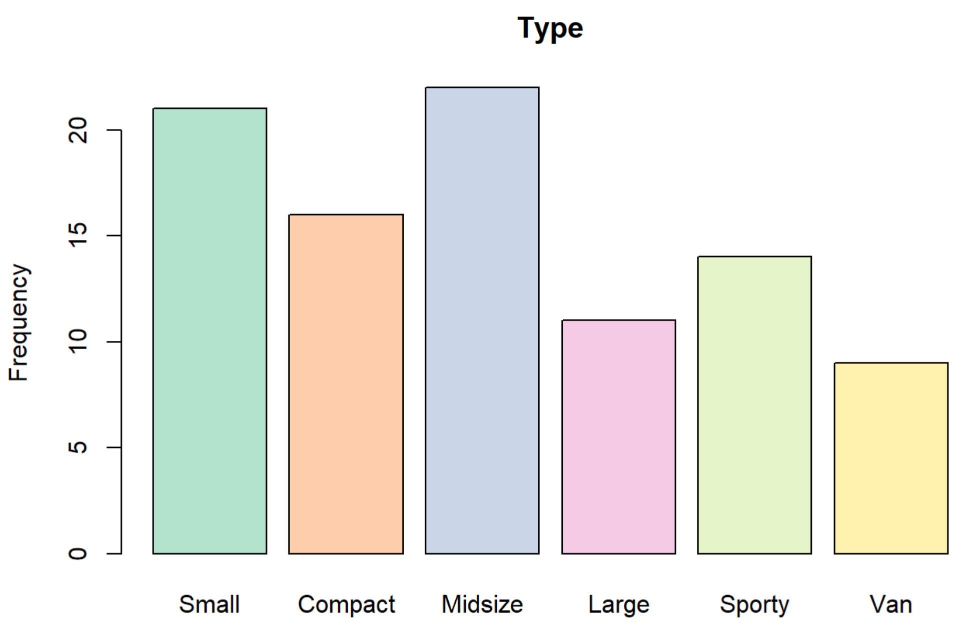
> barplot(table(Manufacturer), main = "Manufacturer", ylab = "Frequency", las = 2, col = rainbow(length(levels(Manufacturer))))



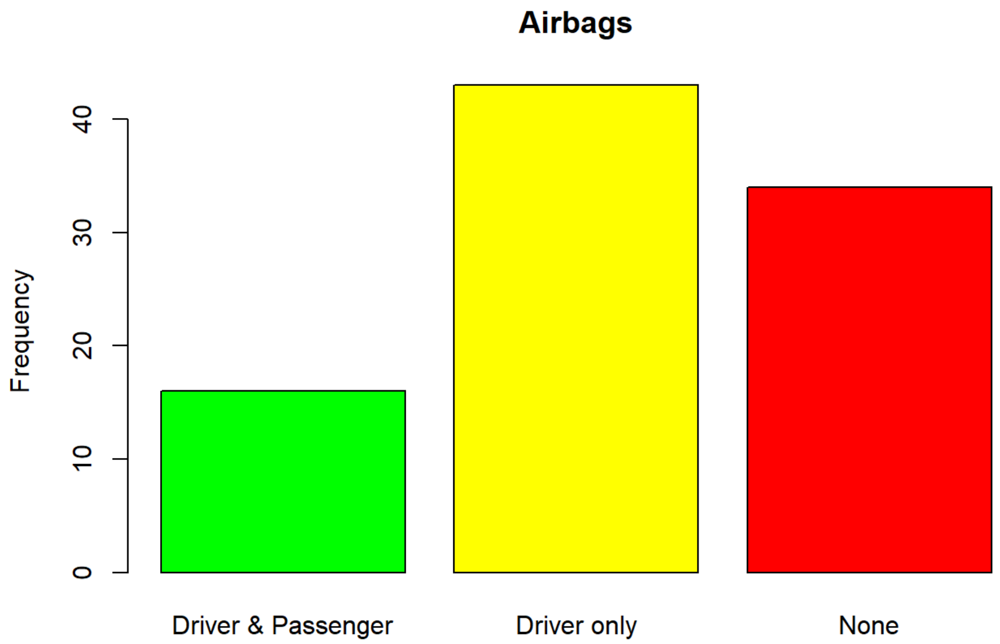
```
> barplot(sort(table(Manufacturer), decreasing = TRUE),
+         main = "Manufacturer",
+         ylab = "Frequency",
+         las = 2,
+         col = rainbow(length(levels(Manufacturer))))
```



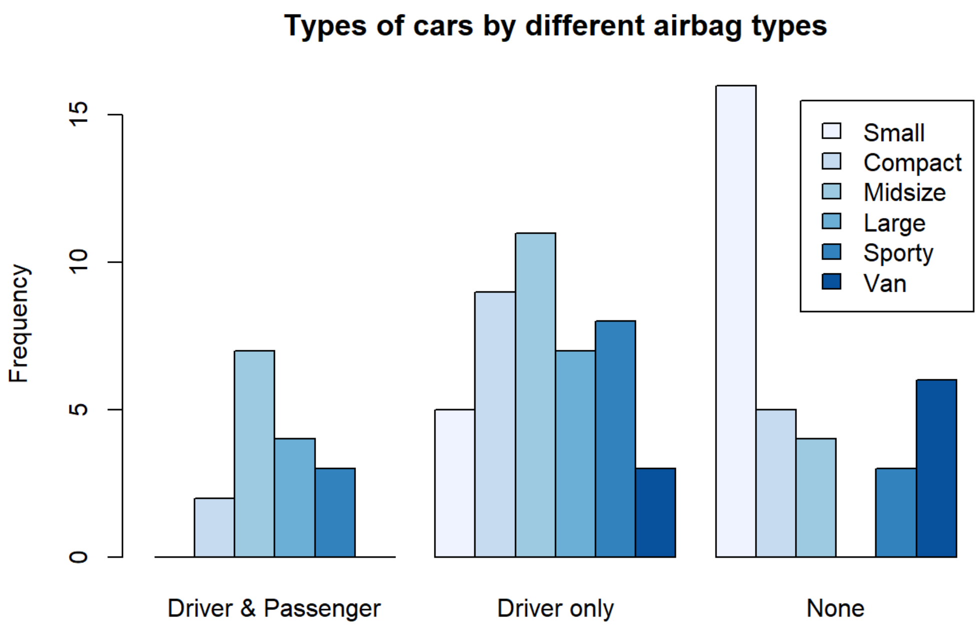

```
> barplot(table(Type),
+         main = "Type",
+         ylab = "Frequency",
+         col = brewer.pal(n = 6, name = "Pastel2"))
```



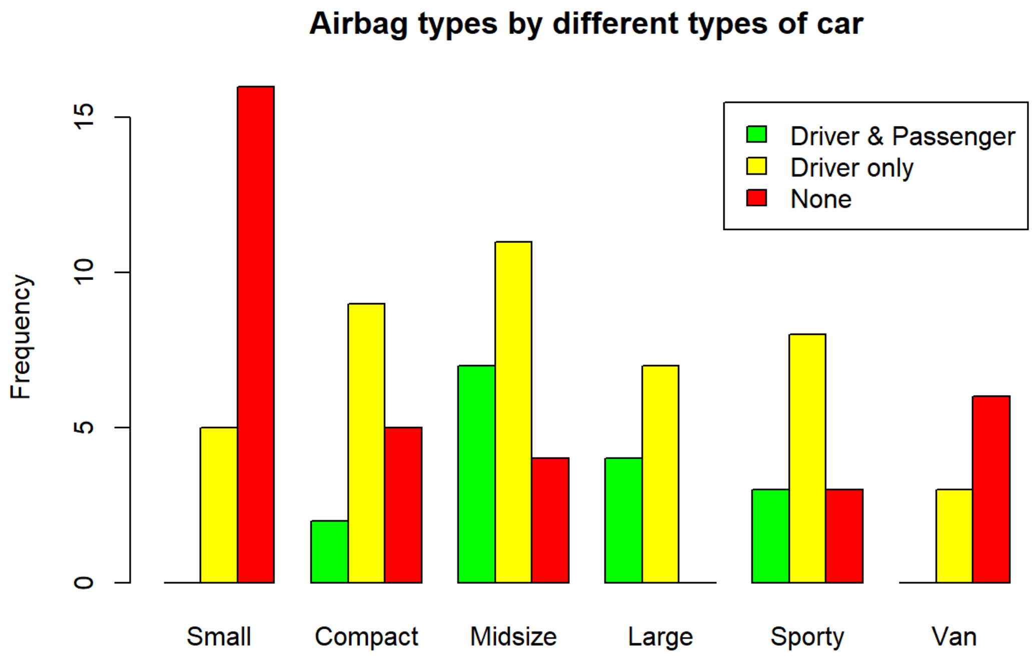
```
> barplot(table(AirBags),
+         main = "Airbags",
+         ylab = "Frequency",
+         col = c("Green", "Yellow", "Red"))
```




```
> barplot(table(Type, AirBags),
+         main = "Types of cars by different airbag types",
+         beside = TRUE,
+         ylab = "Frequency",
+         col = brewer.pal(n = 6, name = "Blues"),
+         legend.text = TRUE)
```



```
> barplot(table(AirBags, Type),
+         main = "Airbag types by different types of car",
+         beside = TRUE,
+         ylab = "Frequency",
+         col = c("Green", "Yellow", "Red"),
+         legend.text = TRUE)
```

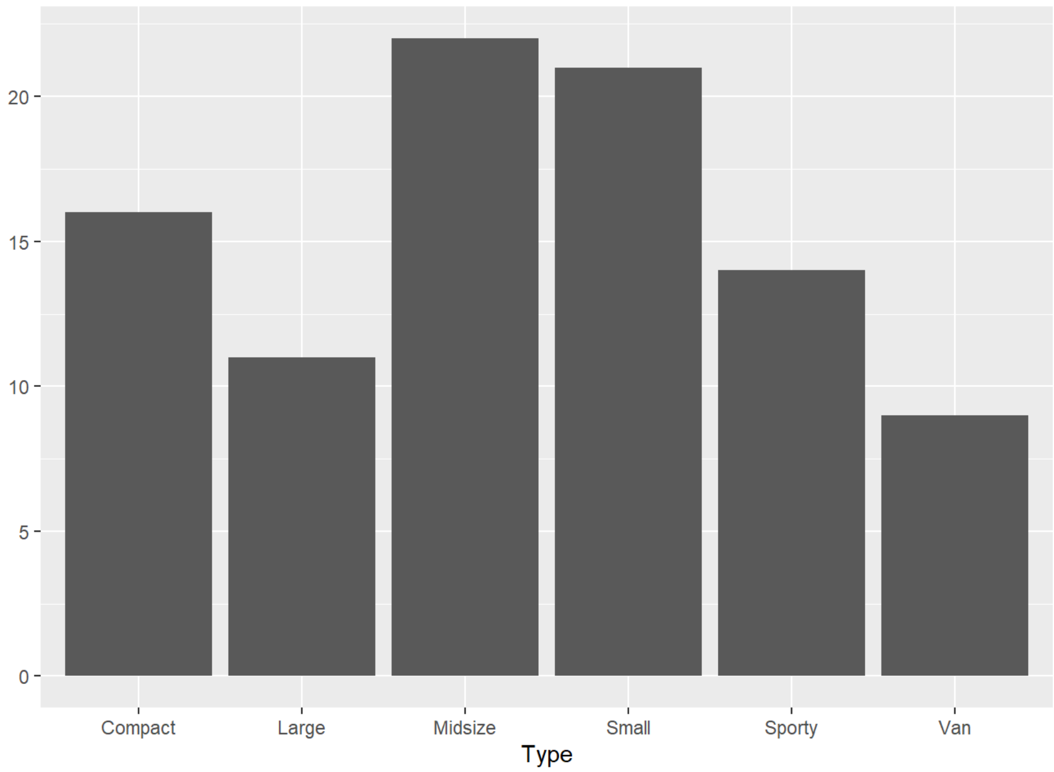


We can also use the ggplot2 package for creating graphs

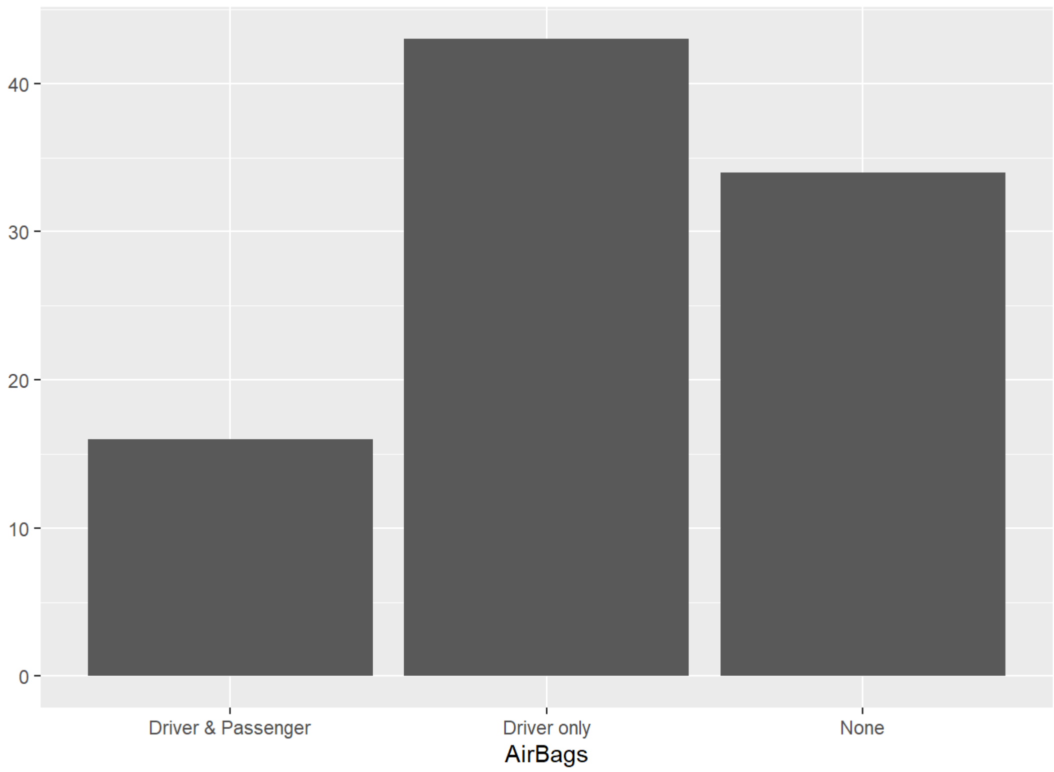
```
> install.packages("ggplot2")
> ? ggplot2
```

We will start by reviewing the quick plot qplot function and ggplot function

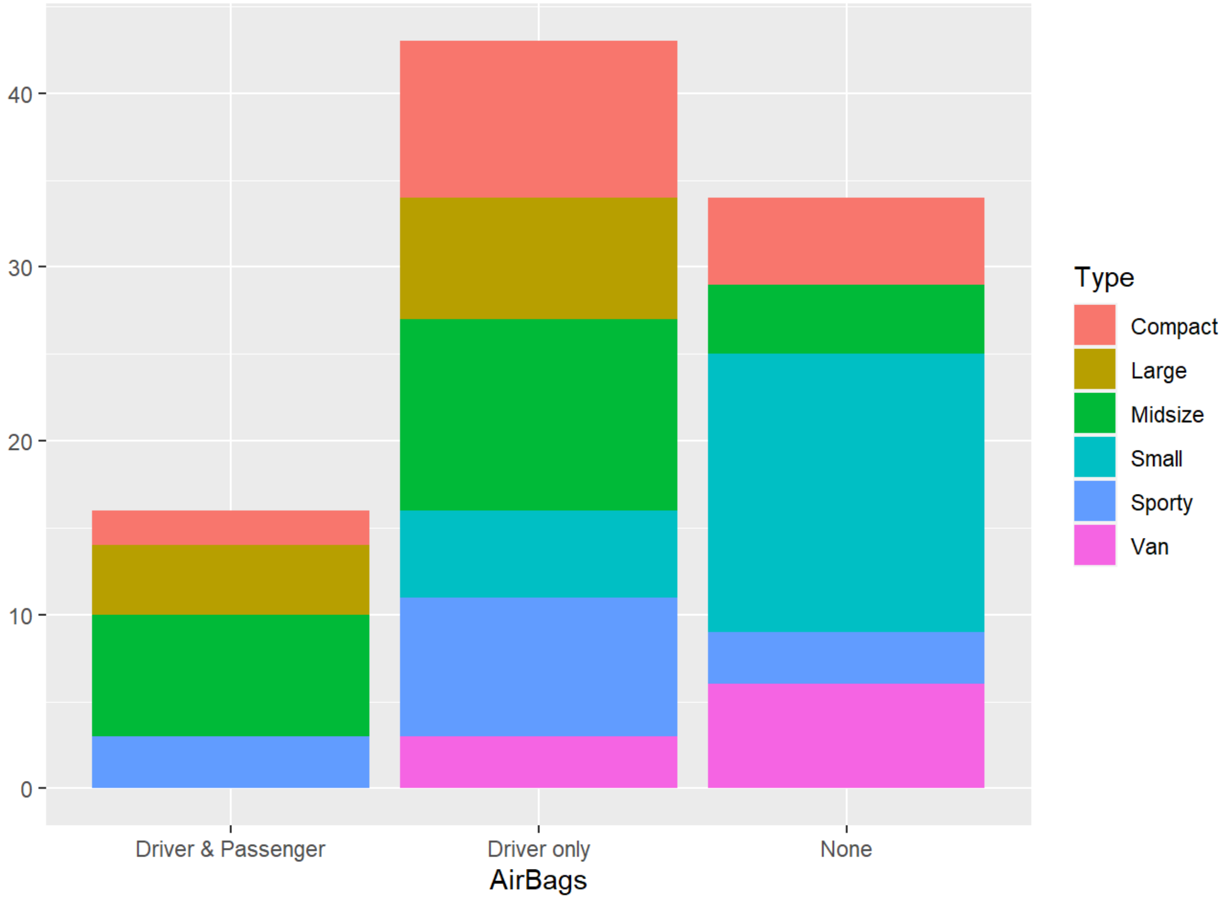
```
> library(ggplot2)
> qplot(Type, data = Cars93)
```



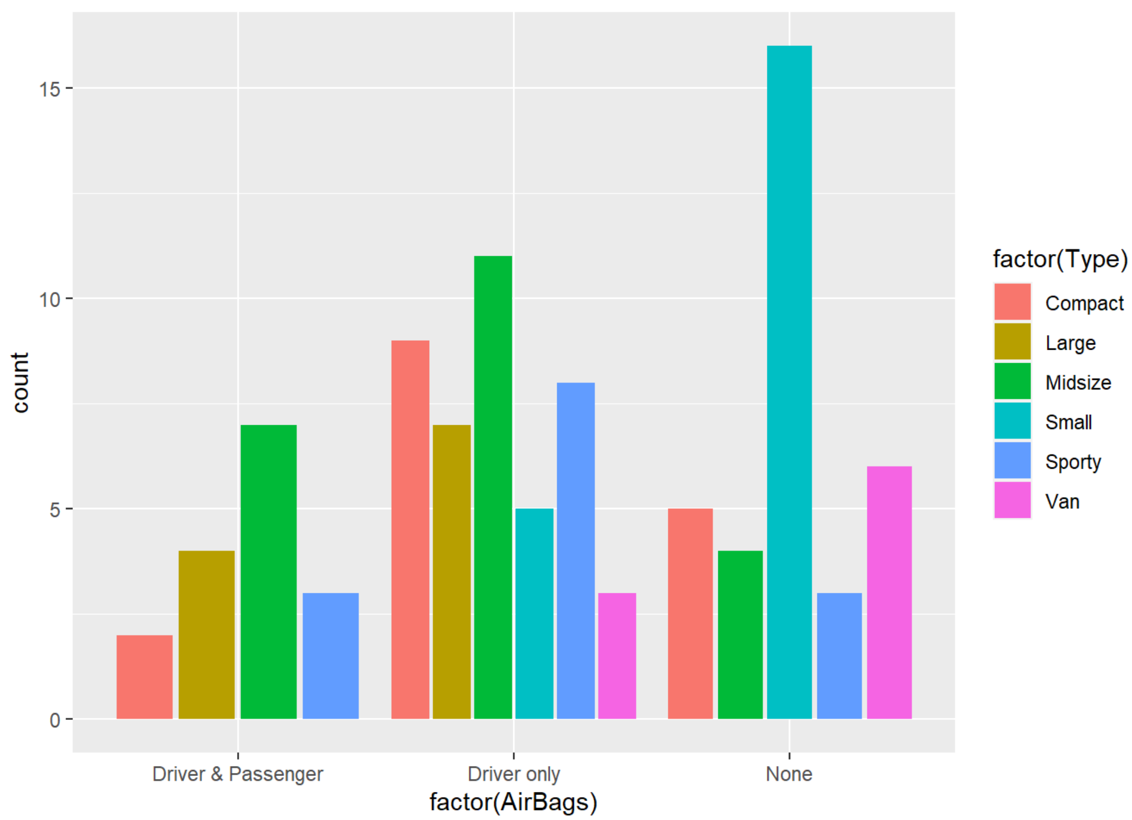
> qplot(AirBags, data = Cars93)



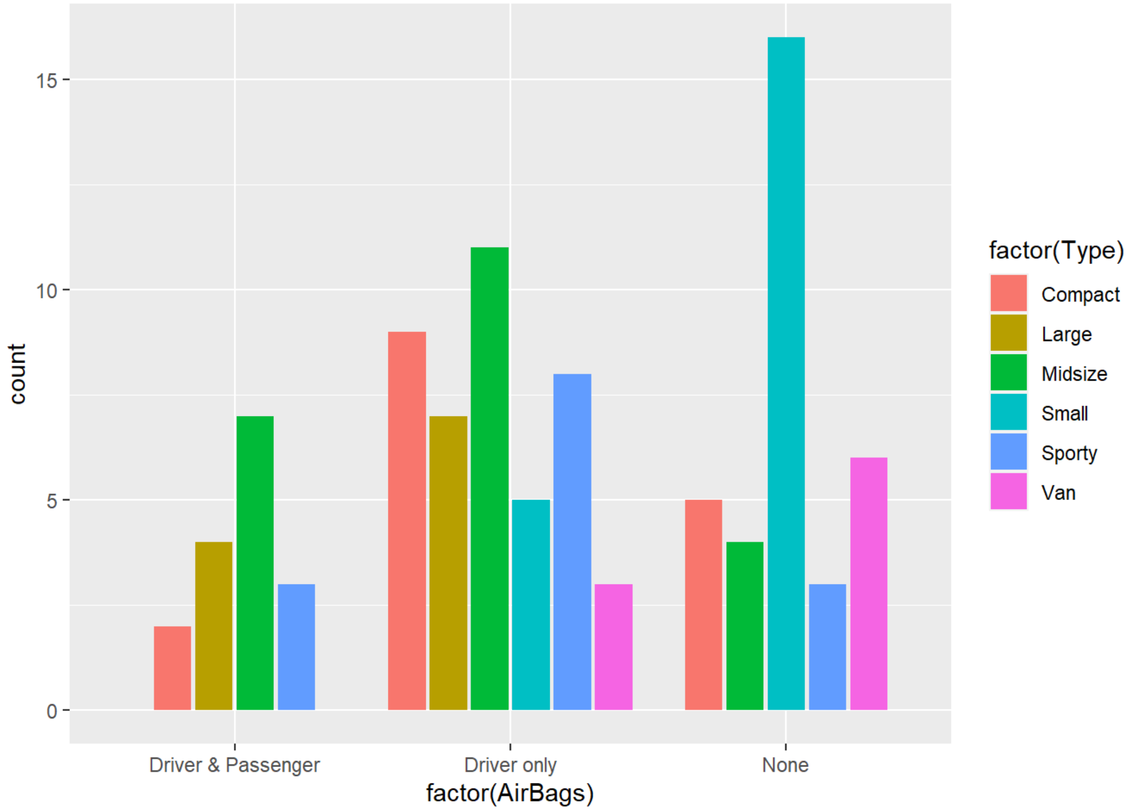
> qplot(AirBags, data = Cars93, fill = Type)



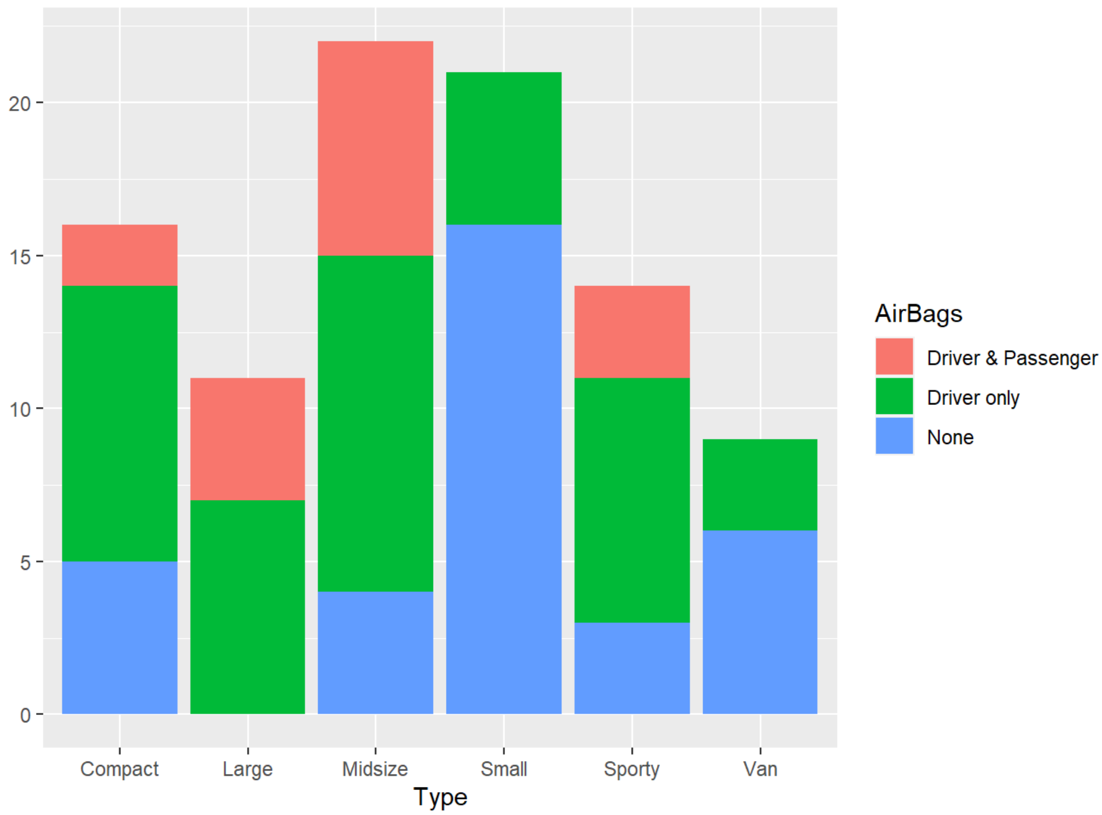
> ggplot(Cars93, aes(factor(AirBags), fill = factor(Type))) + geom_bar(position = "dodge2")



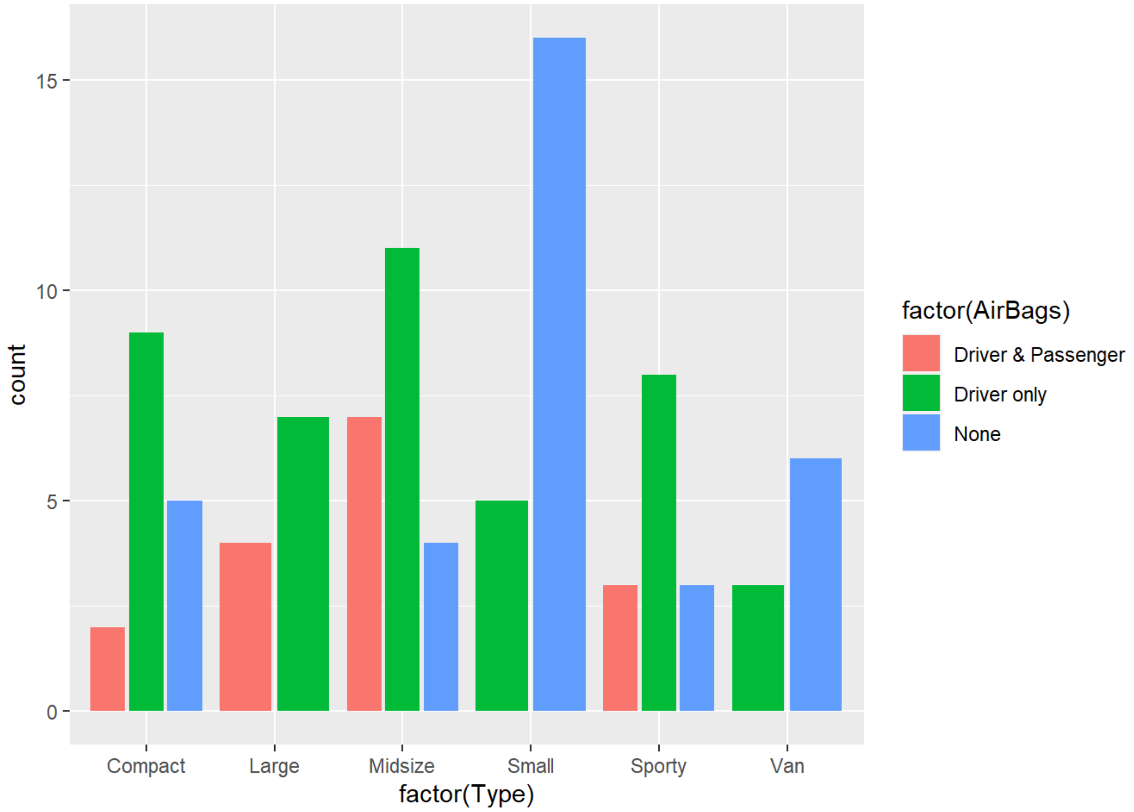
```
> ggplot(Cars93, aes(factor(AirBags), fill = factor(Type))) +
+   geom_bar(position = position_dodge2(preserve = "single"))
```



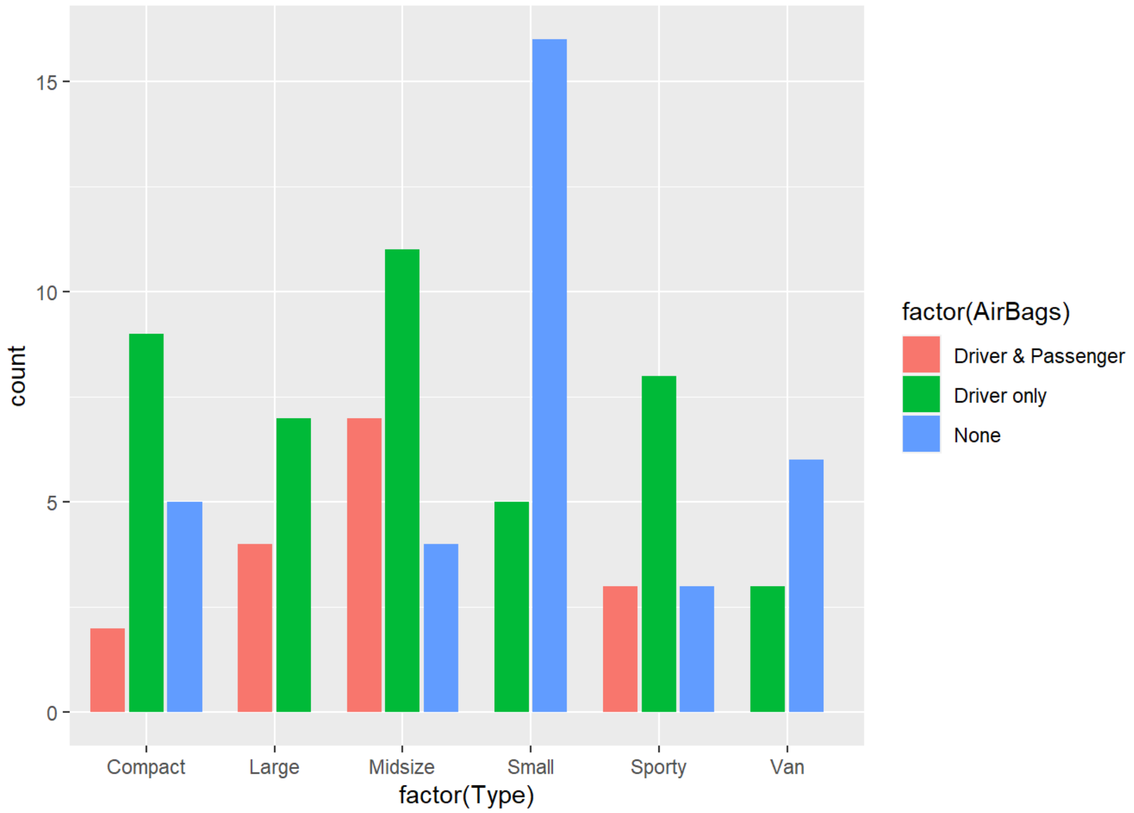

```
> qplot(Type, data = Cars93, fill = AirBags)
```



```
> ggplot(Cars93, aes(factor(Type), fill = factor(AirBags))) +
+   geom_bar(position = "dodge2")
```




```
> ggplot(Cars93, aes(factor(Type), fill = factor(AirBags))) +
+   geom_bar(position = position_dodge2(preserve = "single"))
```



Another example

```
> hair <- c("blond", "blond", "black", "blond", "brown", "brown",
+         "brown", "brown", "black", "brown", "black", "brown",
+         "black", "black", "black", "brown", "brown", "brown",
+         "brown", "brown", "black", "brown", "black", "brown",
+         "blond", "blond", "black", "blond", "brown", "brown",
+         "brown", "brown", "black", "brown", "black", "brown",
+         "brown", "brown", "black", "brown", "black", "brown",
+         "blond", "blond", "black", "blond", "brown", "brown")
> eyes <- c("blue", "green", "brown",  "blue", "green", "brown",
+         "brown", "black", "black", "green", "brown", "brown",
+         "green", "black", "black", "brown", "brown", "black",
+         "green", "black", "black", "brown", "brown", "black",
+         "brown",  "blue", "green", "brown", "brown", "black",
+         "black", "green", "brown",  "blue", "green", "brown",
+         "brown", "black", "brown",  "blue", "green", "brown",
+         "blue", "green", "brown",  "blue", "green", "brown")
> sex <- c("female", "male", "female", "female", "female", "male",
+        "male", "male", "female", "female", "male", "male",
+        "male", "male", "female", "male", "male", "female",
+        "female", "male", "female", "female", "female", "male",
+        "male", "male", "female", "female", "male", "male",
+        "male", "male", "female", "male", "male", "female",
+        "female", "male", "female", "female", "female", "male",
+        "male", "male", "female", "female", "male", "male")
> student <- c("yes", "no", "no", "no", "no", "yes",
+            "yes", "no", "yes", "no", "yes", "no",
+            "no", "yes", "yes", "yes", "no", "yes",
+            "yes", "no", "no", "no", "no", "yes",
+            "yes", "no", "yes", "no", "yes", "no",
+            "no", "yes", "yes", "yes", "no", "yes",
+            "yes", "no", "no", "no", "no", "yes",
+            "yes", "no", "yes", "no", "yes", "no")
> table(hair, eyes, sex, student)
, , sex = female, student = no

       eyes
hair     black blue brown green
  black      1    0     3     1
  blond      0    2     1     0
  brown      0    1     1     2

, , sex = male, student = no

       eyes
hair     black blue brown green
  black      0    0     0     2
  blond      0    1     0     2
  brown      5    0     3     0

, , sex = female, student = yes
```

```
      eyes
hair    black blue brown green
  black         2    0    2    1
  blond         0    1    0    0
  brown         1    0    2    1

, , sex = male, student = yes

      eyes
hair    black blue brown green
  black         1    0    1    0
  blond         0    1    1    0
  brown         1    1    5    2

> ftable(hair, eyes, sex, student)
                  student no yes
hair  eyes  sex
black black female         1  2
            male           0  1
      blue  female         0  0
            male           0  0
      brown female         3  2
            male           0  1
      green female         1  1
            male           2  0
blond black female         0  0
            male           0  0
      blue  female         2  1
            male           1  1
      brown female         1  0
            male           0  1
      green female         0  0
            male           2  0
brown black female         0  1
            male           5  1
      blue  female         1  0
            male           0  1
      brown female         1  2
            male           3  5
      green female         2  1
            male           0  2

> ftable(student, sex, hair, eyes)
                   eyes black blue brown green
student sex    hair
no      female black          1    0    3    1
               blond          0    2    1    0
               brown          0    1    1    2
        male   black          0    0    0    2
               blond          0    1    0    2
               brown          5    0    3    0
```

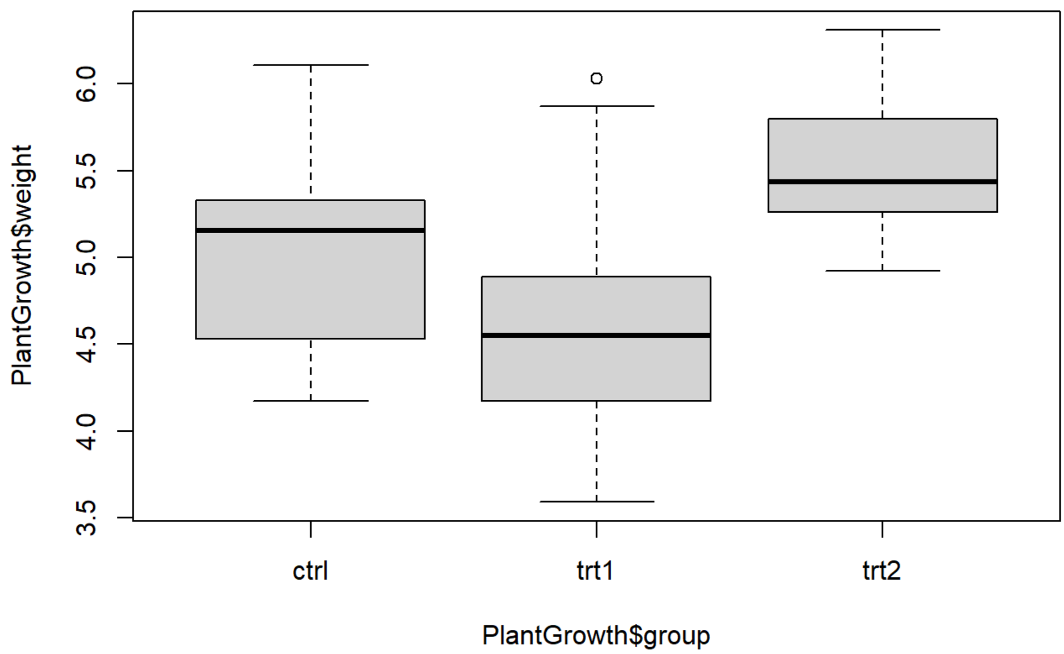
```
yes     female black        2   0   2   1
               blond        0   1   0   0
               brown        1   0   2   1
        male   black        1   0   1   0
               blond        0   1   1   0
               brown        1   1   5   2
```

Categorical and numerical data

Let's review the PlantGrowth data frame

```
> head(PlantGrowth)
  weight group
1    4.2  ctrl
2    5.6  ctrl
3    5.2  ctrl
4    6.1  ctrl
5    4.5  ctrl
6    4.6  ctrl
```

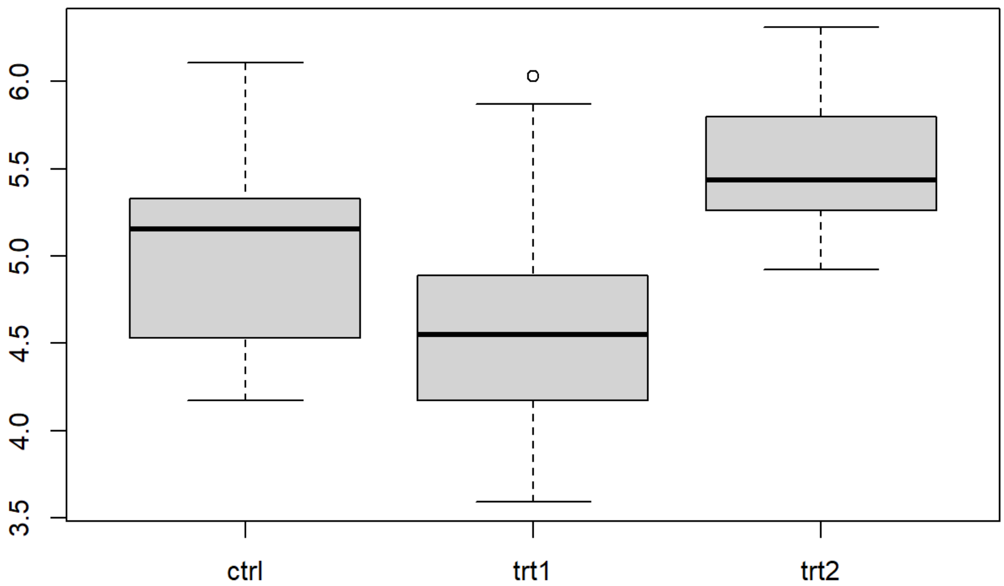
```
> boxplot(PlantGrowth$weight ~ PlantGrowth$group)
```



Another way is using the unstack function

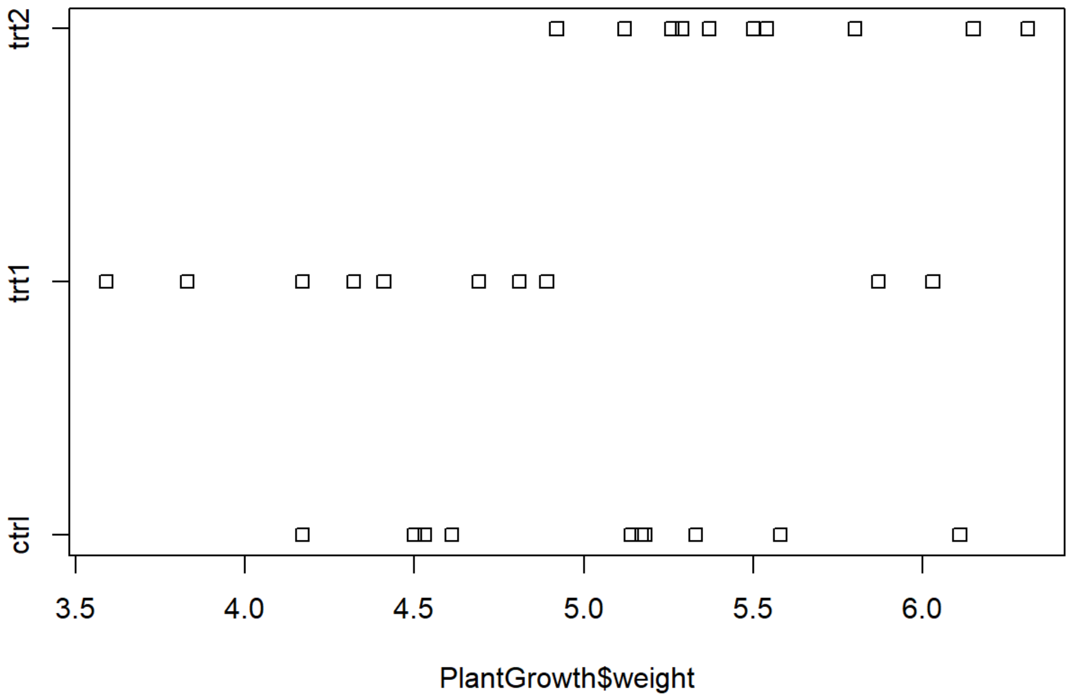
```
> unstack(PlantGrowth)
   ctrl trt1 trt2
1   4.2  4.8  6.3
2   5.6  4.2  5.1
3   5.2  4.4  5.5
4   6.1  3.6  5.5
5   4.5  5.9  5.4
6   4.6  3.8  5.3
7   5.2  6.0  4.9
8   4.5  4.9  6.2
9   5.3  4.3  5.8
10  5.1  4.7  5.3

> boxplot(unstack(PlantGrowth))
```



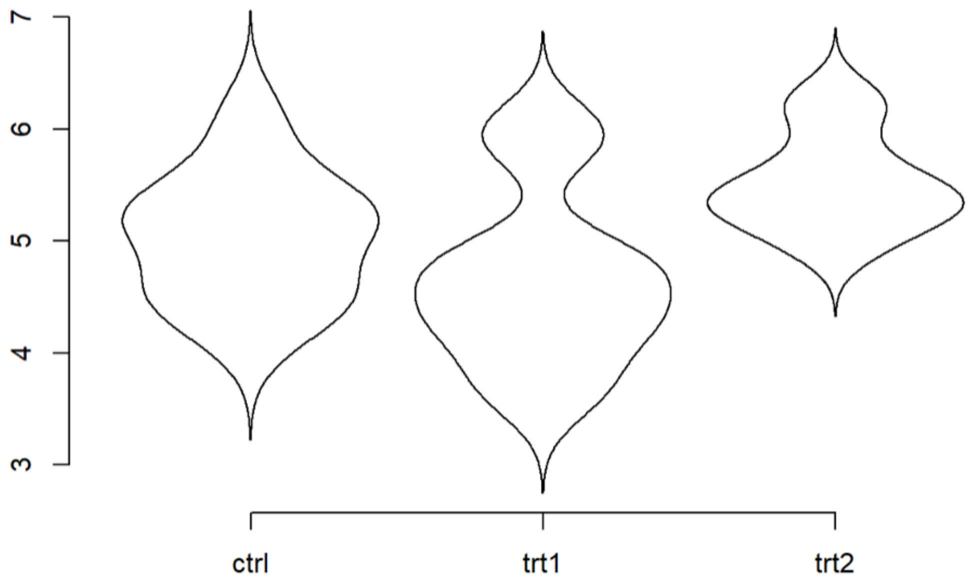
We can also visualize it using stripchart

```
> stripchart(PlantGrowth$weight ~ PlantGrowth$group)
```

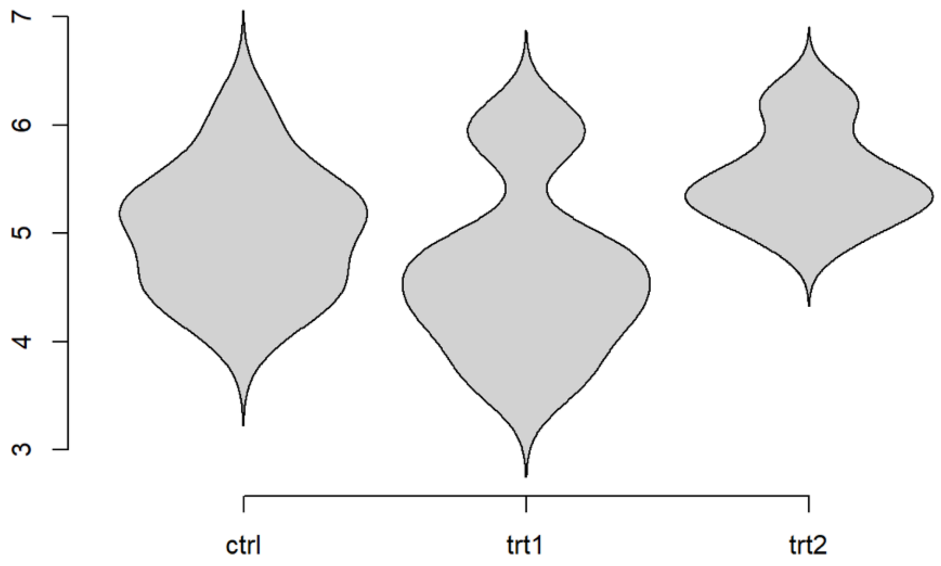


or with violinplot using the simple.violinplot function from UsingR package

```
> library(UsingR)
> simple.violinplot(PlantGrowth$weight ~ PlantGrowth$group)
```

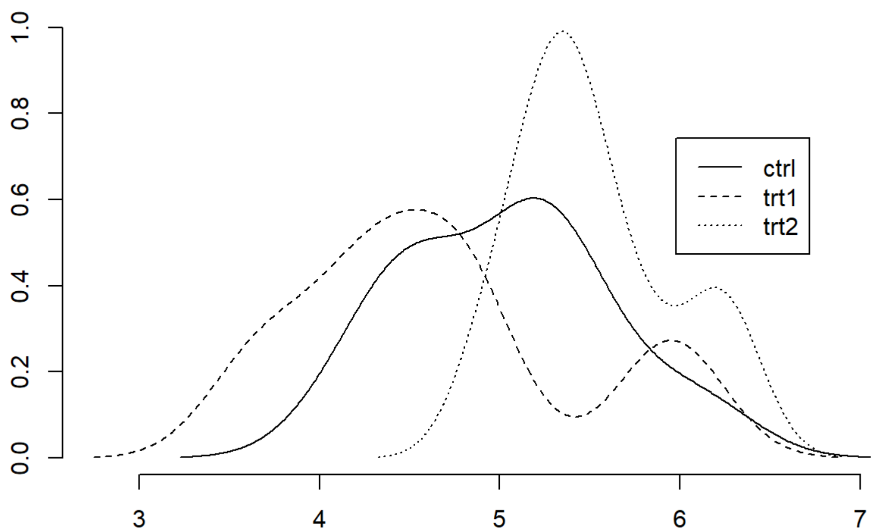


```
> simple.violinplot(PlantGrowth$weight ~ PlantGrowth$group, col = "lightgray")
```



or with densities using the simple.densityplot function from UsingR package

```
> simple.densityplot(PlantGrowth$weight ~ PlantGrowth$group)
```

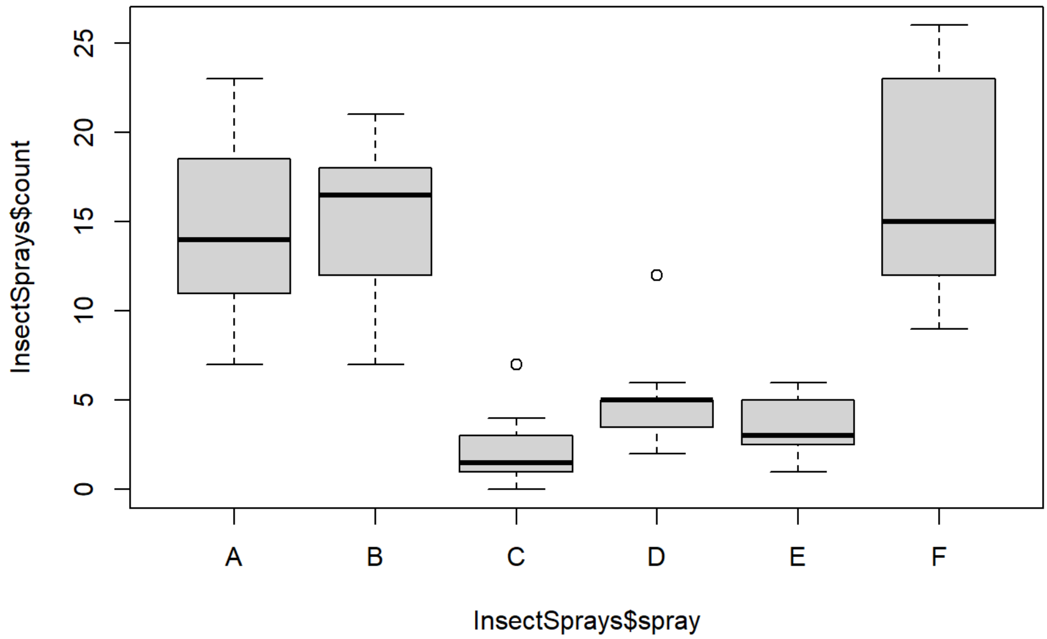


Another example. Let's review the InsectSpray's data frame presenting the counts of insects in agricultural experiment units treated with different insecticides.

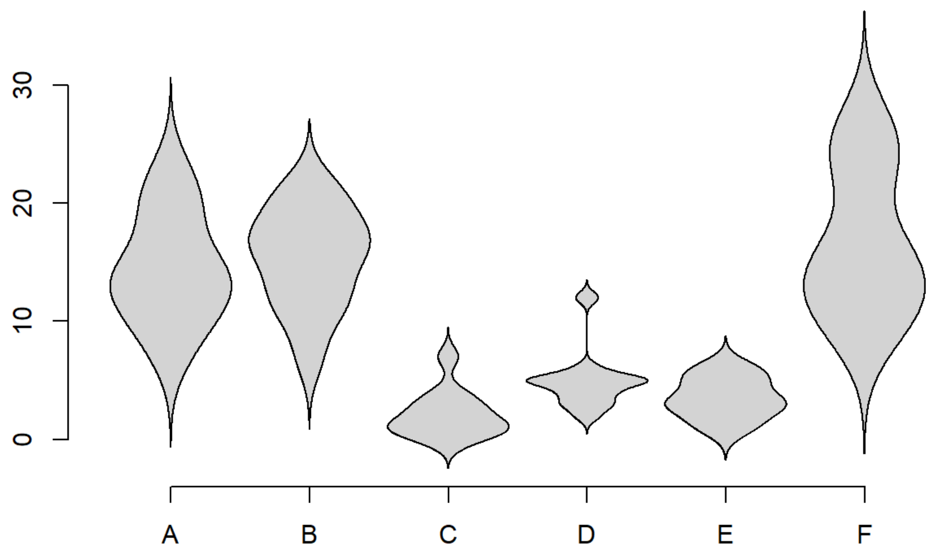
```
> head(InsectSprays)
  count spray
1    10     A
2     7     A
3    20     A
4    14     A
```

5 14 A
6 12 A

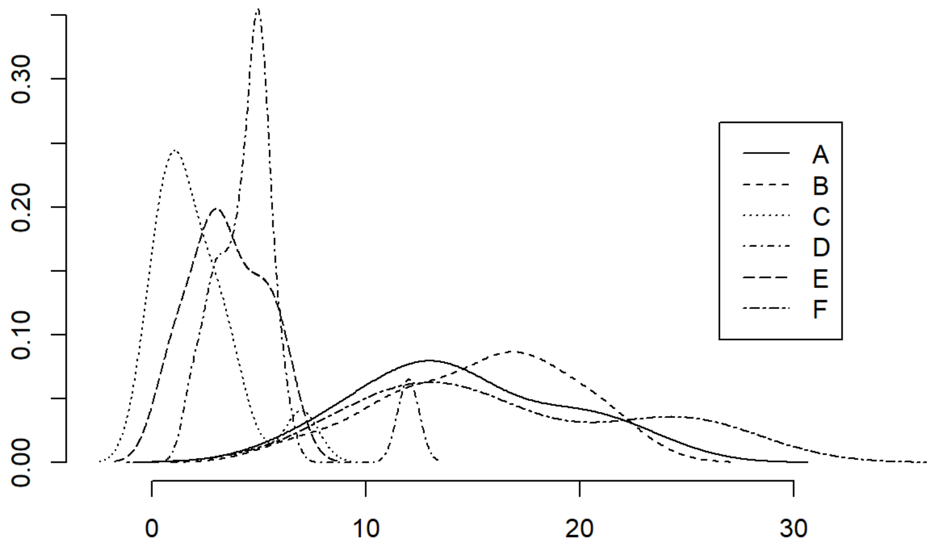
> boxplot(InsectSprays$count ~ InsectSprays$spray)



> simple.violinplot(InsectSprays$count ~ InsectSprays$spray, col = "lightgray")



```
> simple.densityplot(InsectSprays$count ~ InsectSprays$spray)
```

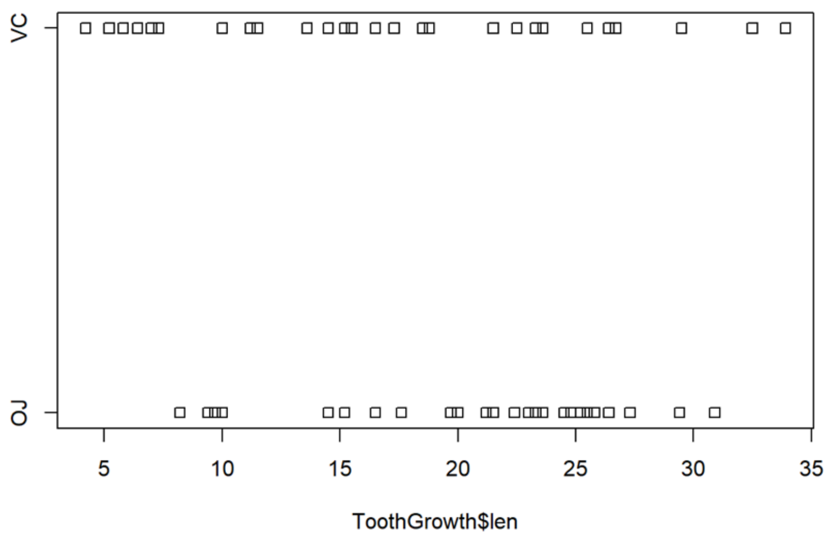


One more example with ToothGrowth data frame presenting the effect of vitamin C on tooth growth in guinea pigs. Here we have 1 numerical and 2 categorical variables.

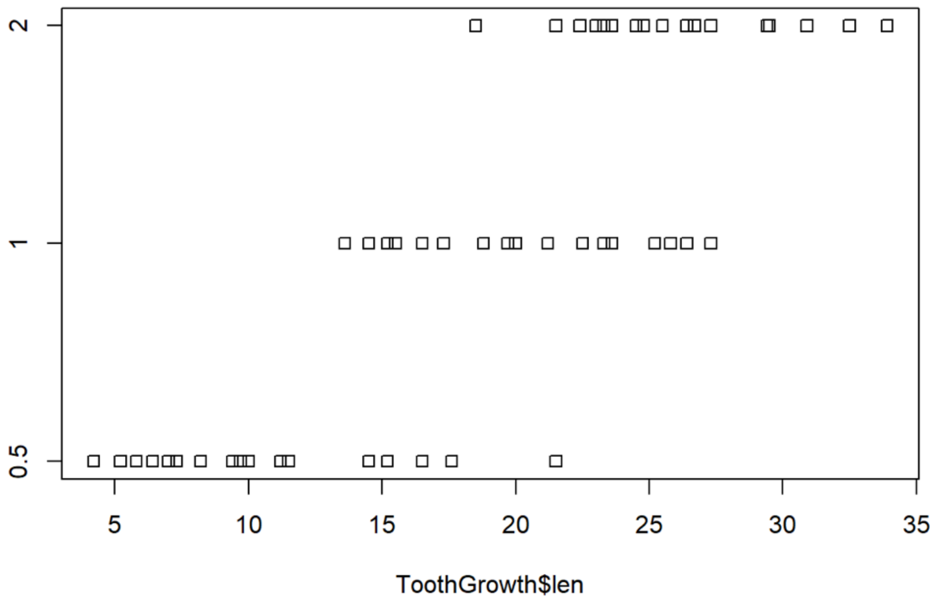
```
> head(ToothGrowth)
   len supp dose
1  4.2   VC  0.5
2 11.5   VC  0.5
3  7.3   VC  0.5
4  5.8   VC  0.5
5  6.4   VC  0.5
6 10.0   VC  0.5
```

We can use stripchart to show the effect of each of the categorical variables.

```
> stripchart(ToothGrowth$len ~ ToothGrowth$supp)
```

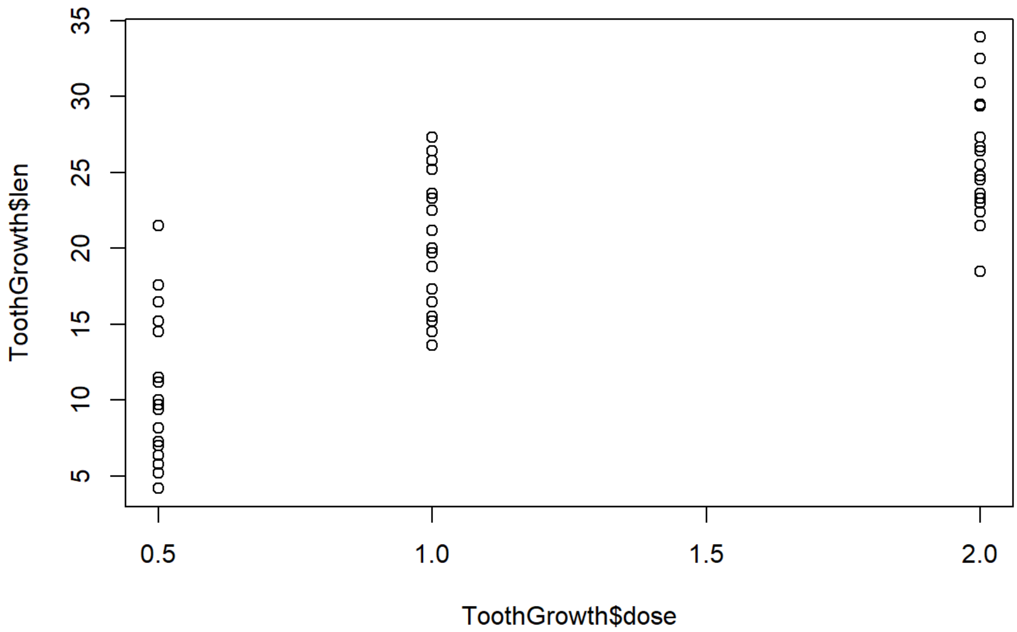


```
> stripchart(ToothGrowth$len ~ ToothGrowth$dose)
```



But how can we show the effect of both categorical variables on one graph?

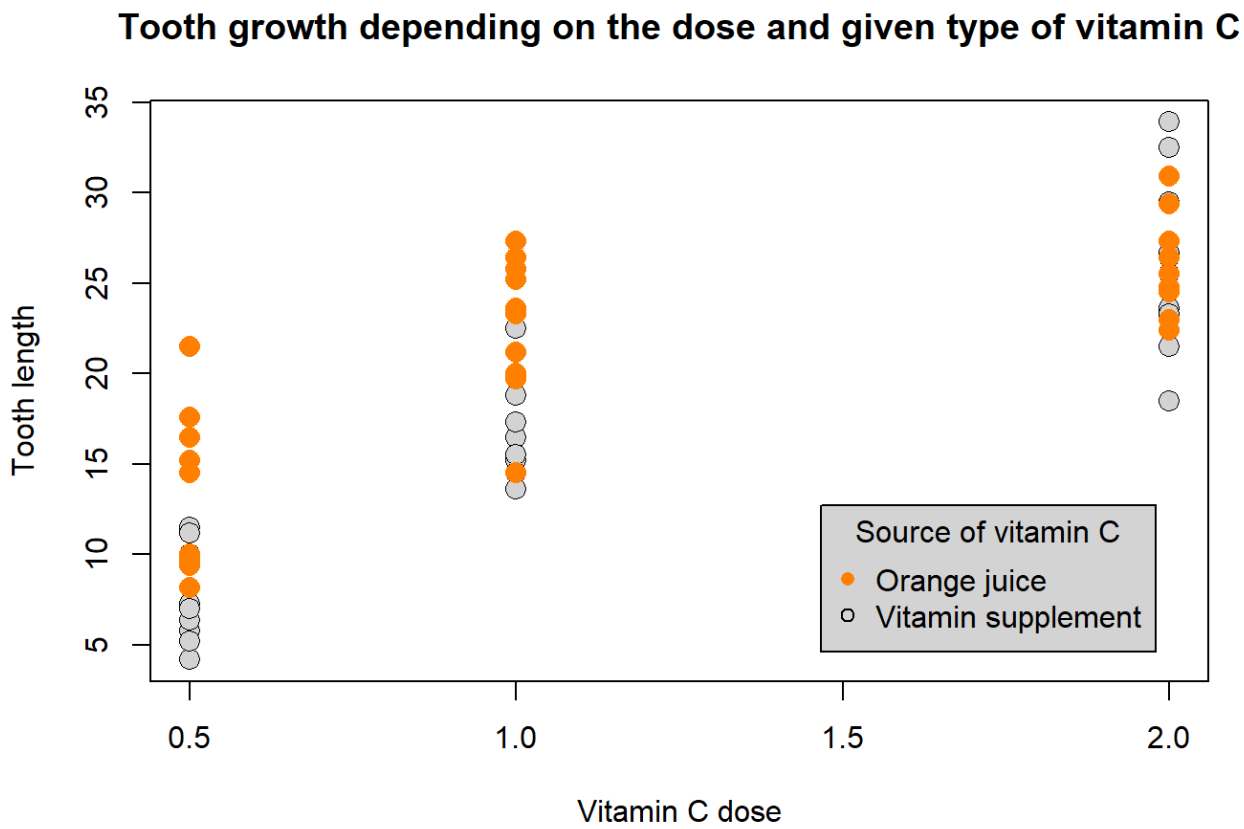
```
> plot(ToothGrowth$len ~ ToothGrowth$dose)
```



We can plot the other categorical variable on the above plot using different signs or colors marking the observations.

We can use the pch and col parameters in plot function to specify the point symbols and colors.

```
> colors <- c("darkorange1", "black")
> shapes <- c(16, 21)
> plot(ToothGrowth$len ~ ToothGrowth$dose,
+     main = "Tooth growth depending on the dose and given type of vitamin C",
+     xlab = "Vitamin C dose",
+     ylab = "Tooth length",
+     col = colors[ToothGrowth$supp],
+     bg = "lightgray",
+     pch = shapes[ToothGrowth$supp],
+     lwd = 0.9,
+     cex = 1.5)
> legend("bottomright", inset = 0.05,
+      title="Source of vitamin C",
+      legend = c("Orange juice", "Vitamin supplement"),
+      col = colors,
+      bg = "lightgray",
+      pch = shapes)
```



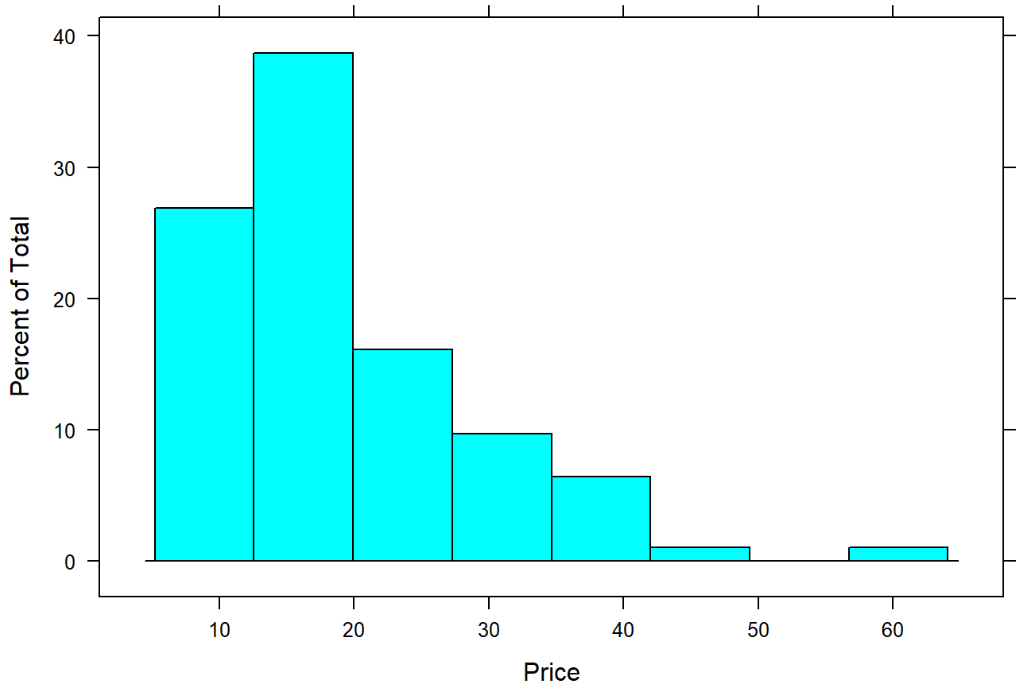
We see that in the groups taking 0.5 mg and 1.0 mg doses, the orange juice source was more effective on the tooth growth.

lattice is another useful package for easily displaying multivariate graphics.

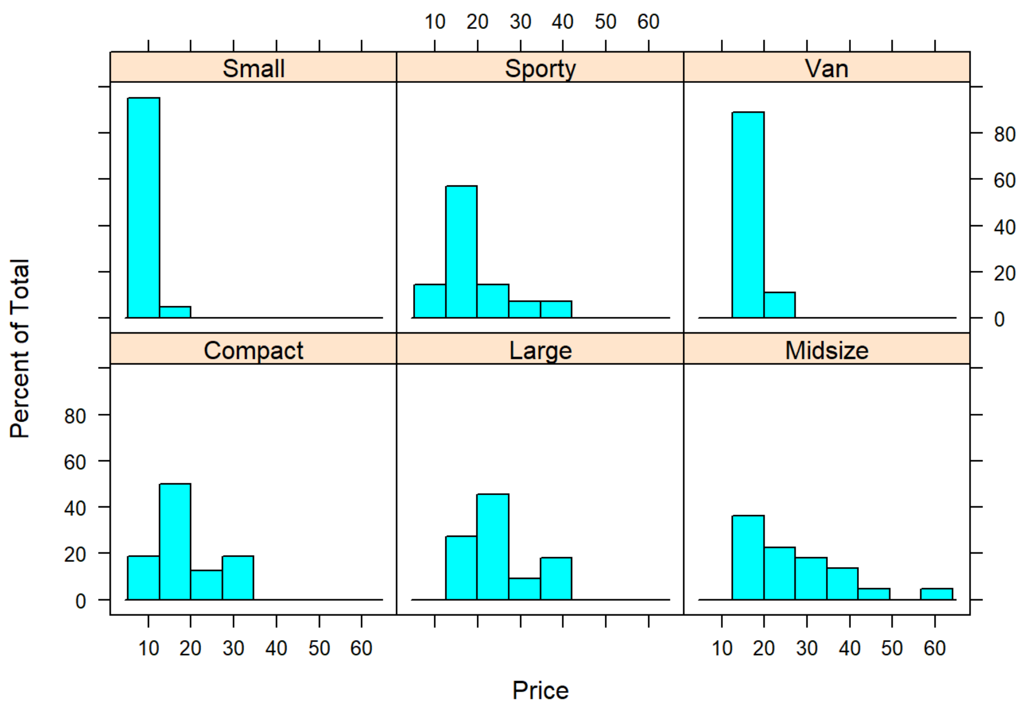
```
> install.packages("lattice")
> ? lattice
> library(lattice)
```

Continuing the Cars93 example we can make a histogram for the Price depending on the Type of the car

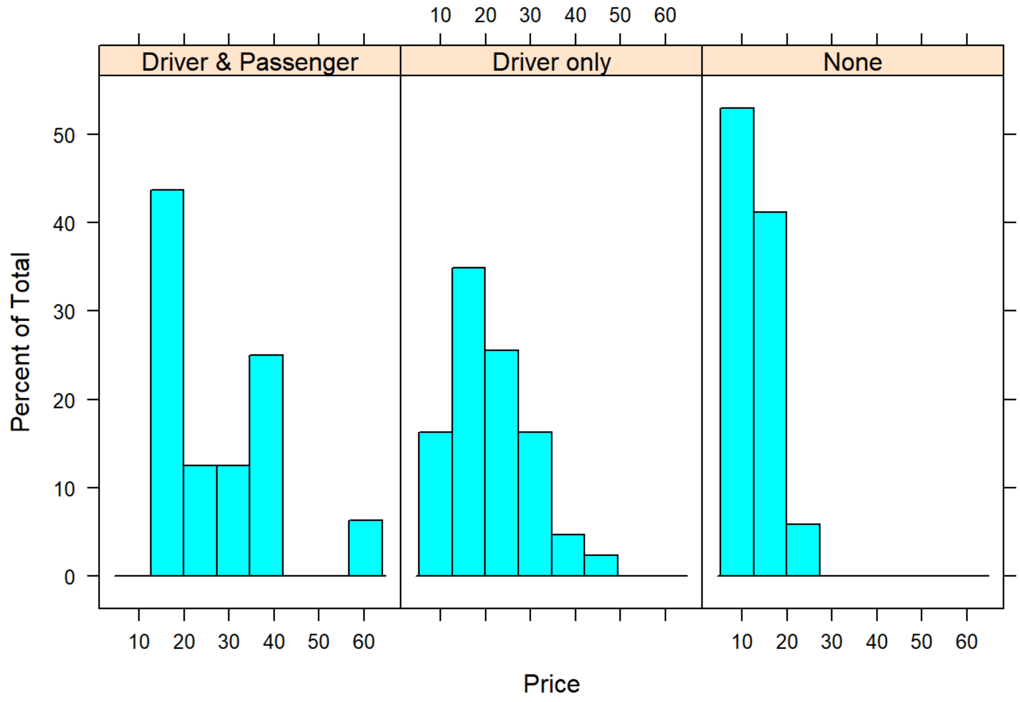
```
> histogram( ~ Price, data = Cars93)
```



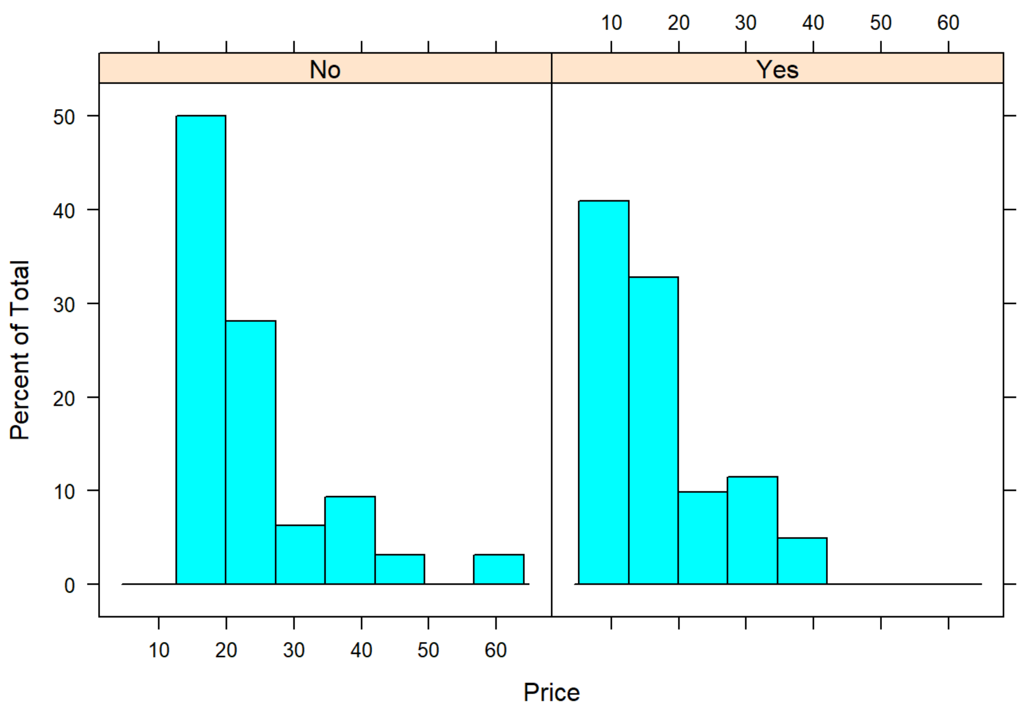

```
> histogram( ~ Price | Type, data = Cars93)
```



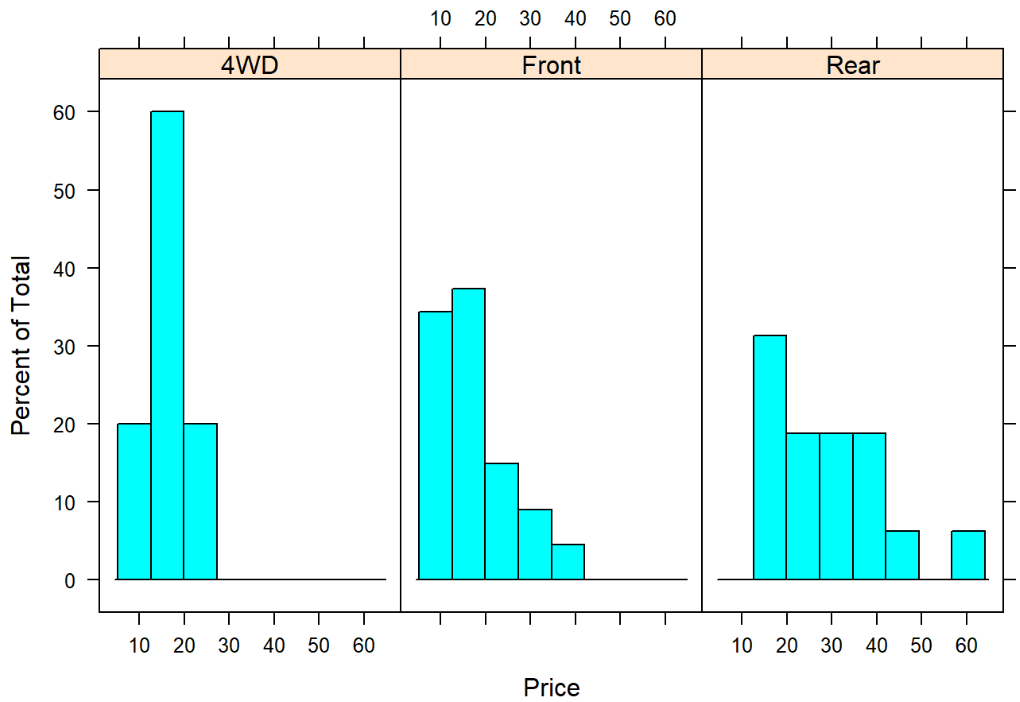
```
> histogram( ~ Price | AirBags, data = Cars93)
```



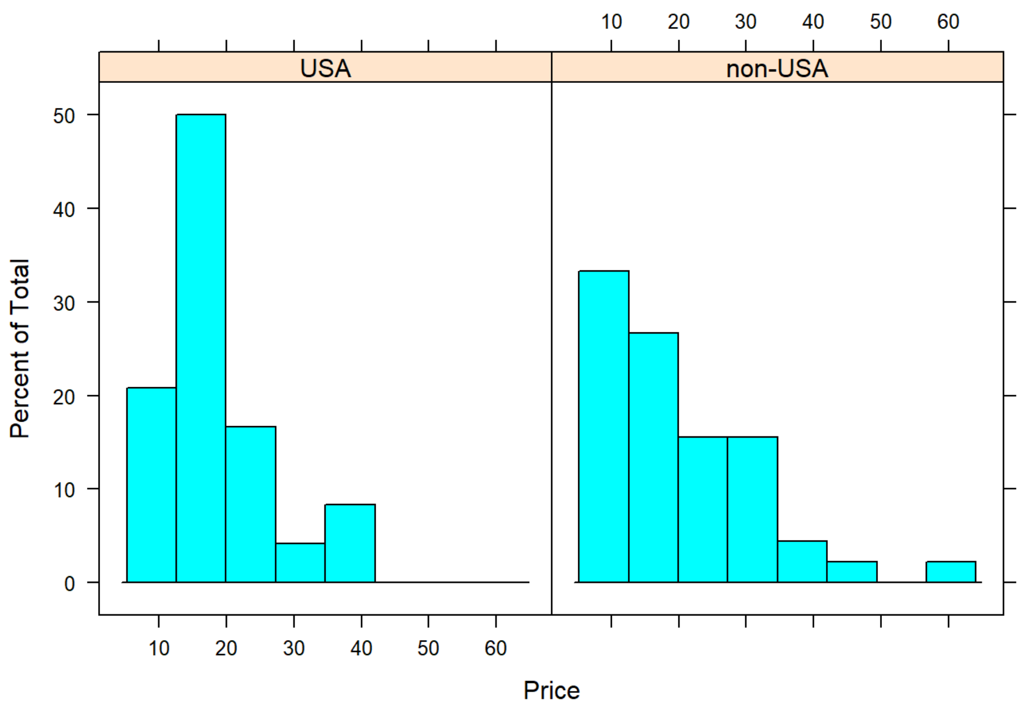

```
> histogram( ~ Price | Man.trans.avail, data = Cars93)
```



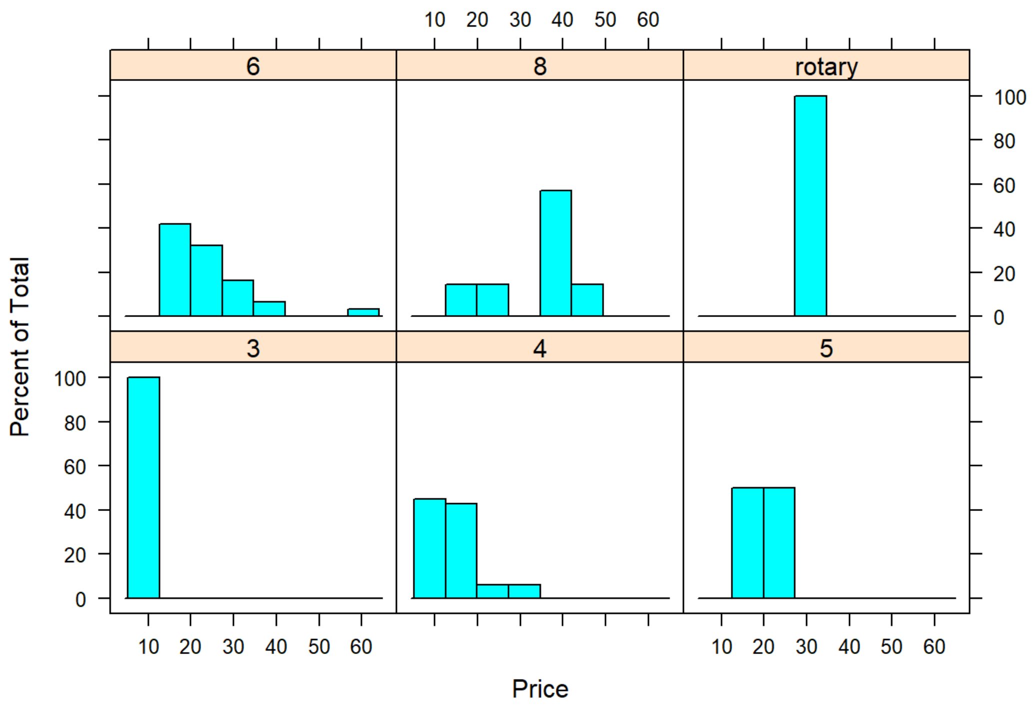
> histogram(~ Price | DriveTrain, data = Cars93)



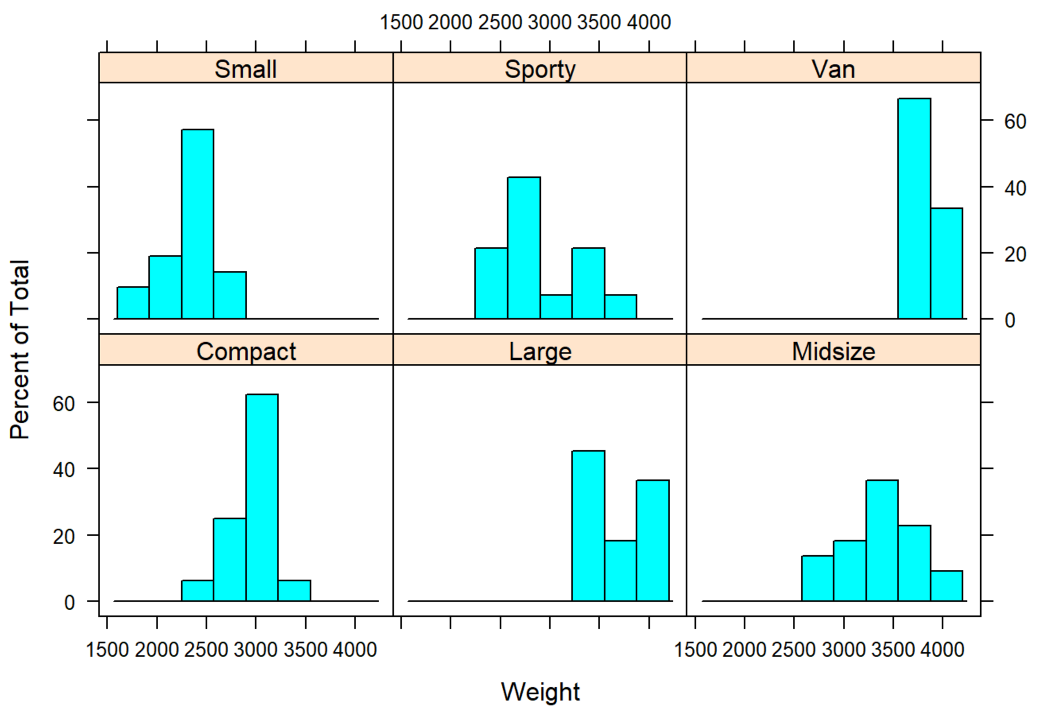
> histogram(~ Price | Origin, data = Cars93)



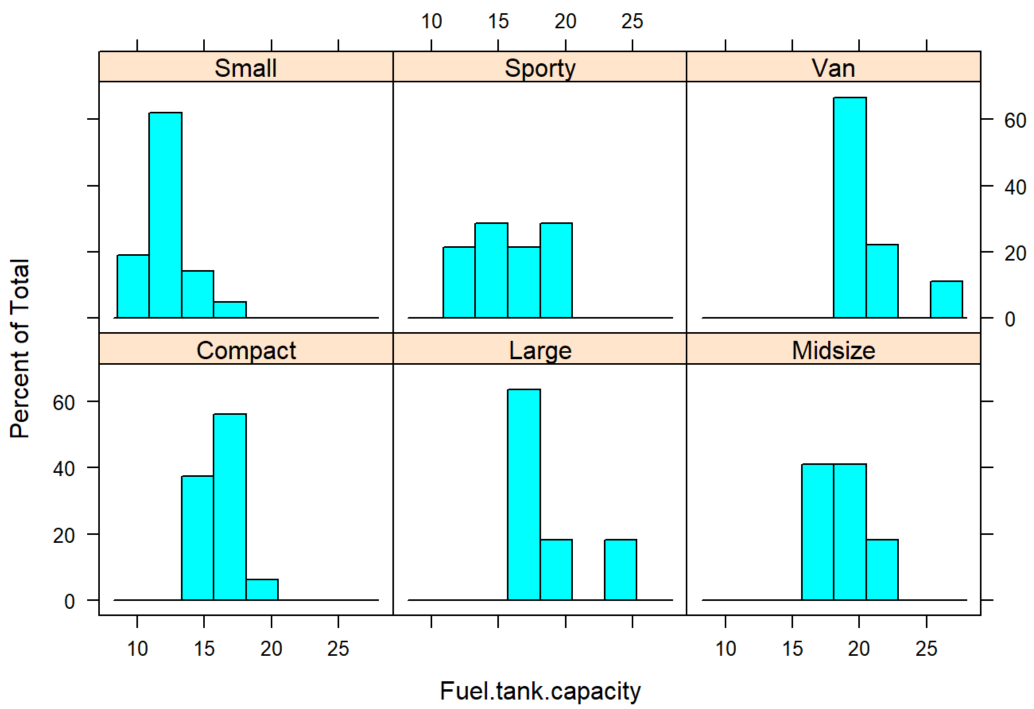
> histogram(~ Price | Cylinders , data = Cars93)



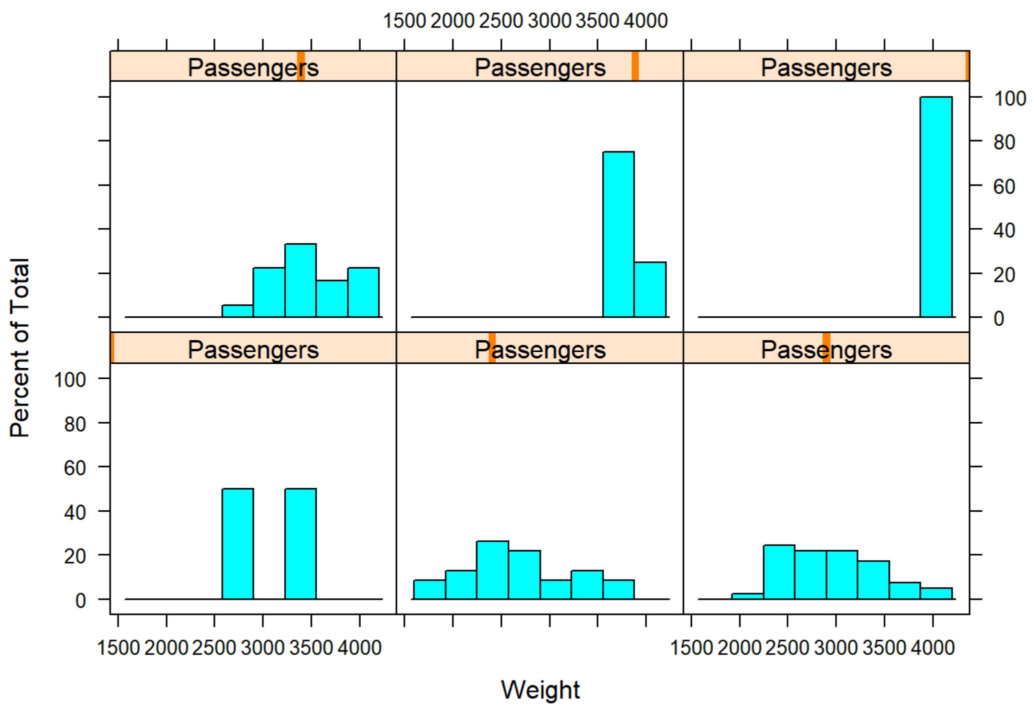
> histogram(~ Weight | Type , data = Cars93)



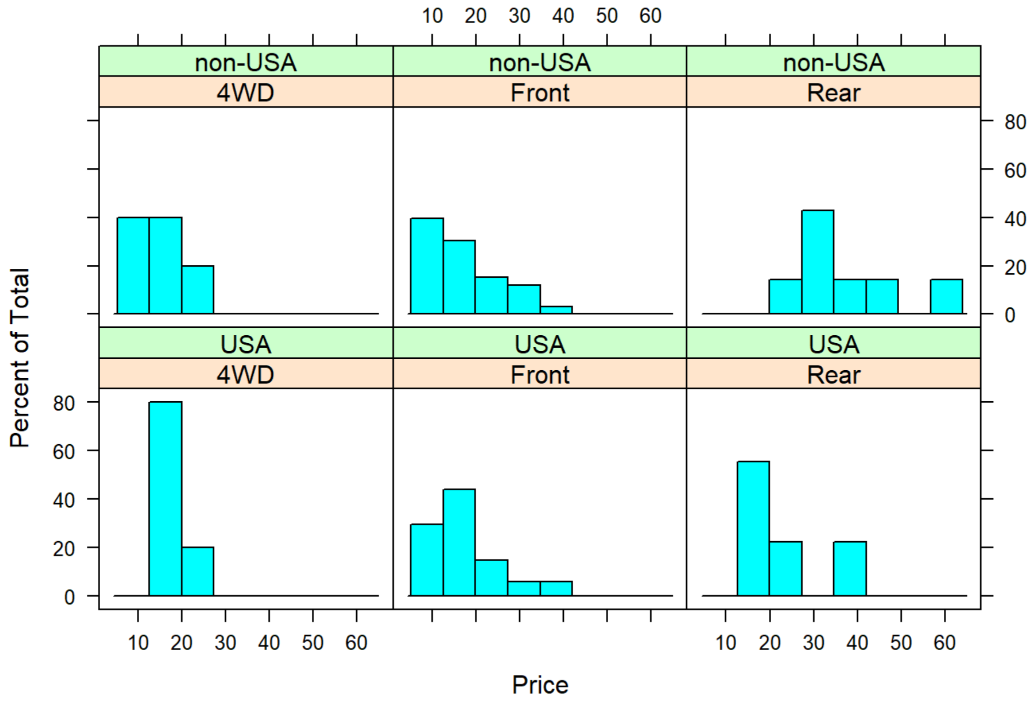
> histogram(~ Fuel.tank.capacity | Type , data = Cars93)



> histogram(~ Weight | Passengers , data = Cars93)

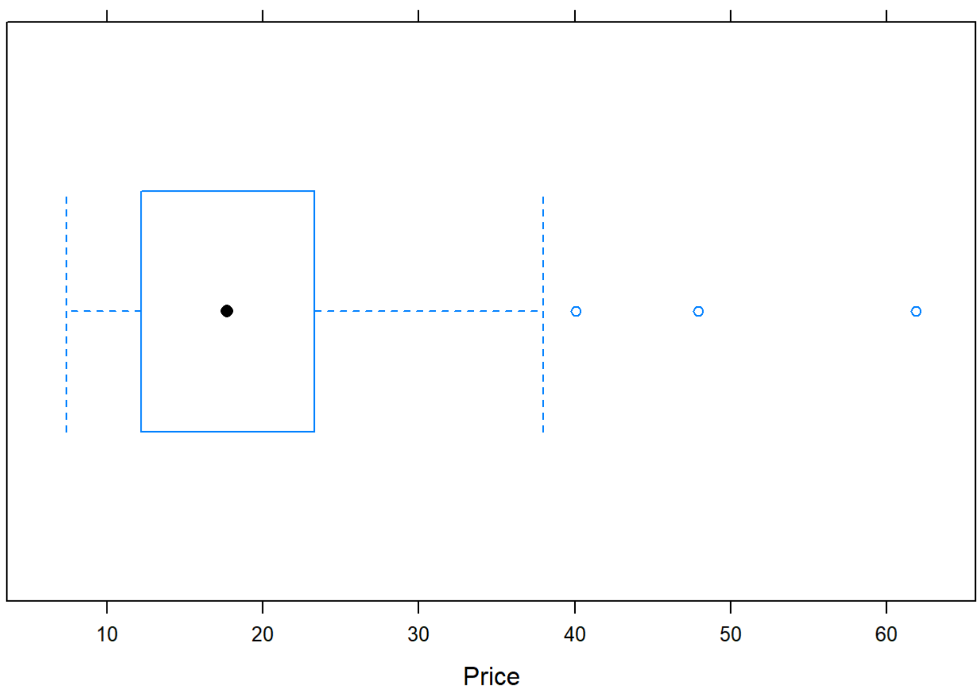


```
> histogram( ~ Price | DriveTrain * Origin, data = Cars93)
```

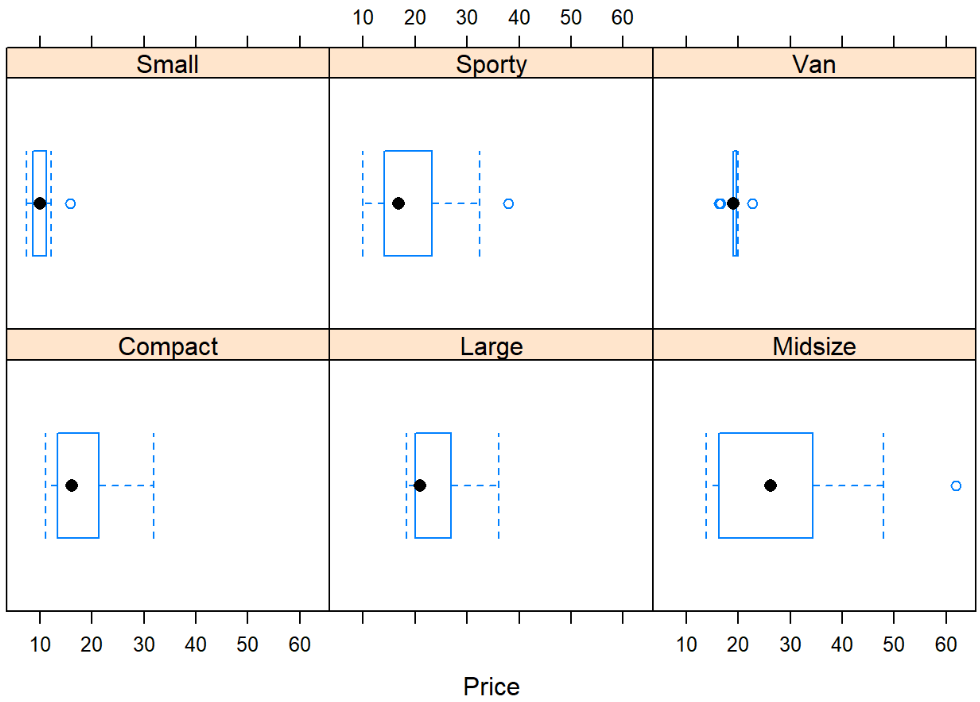


We can similarly make the boxplot lattice graphics using the bwplot function

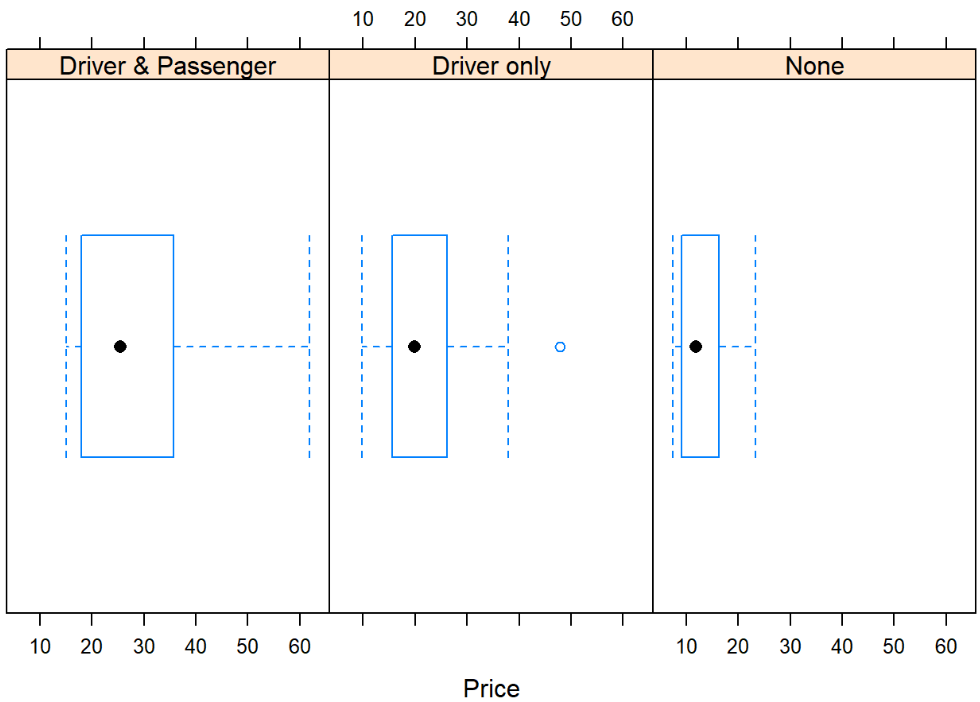
```
> bwplot( ~ Price, data = Cars93)
```



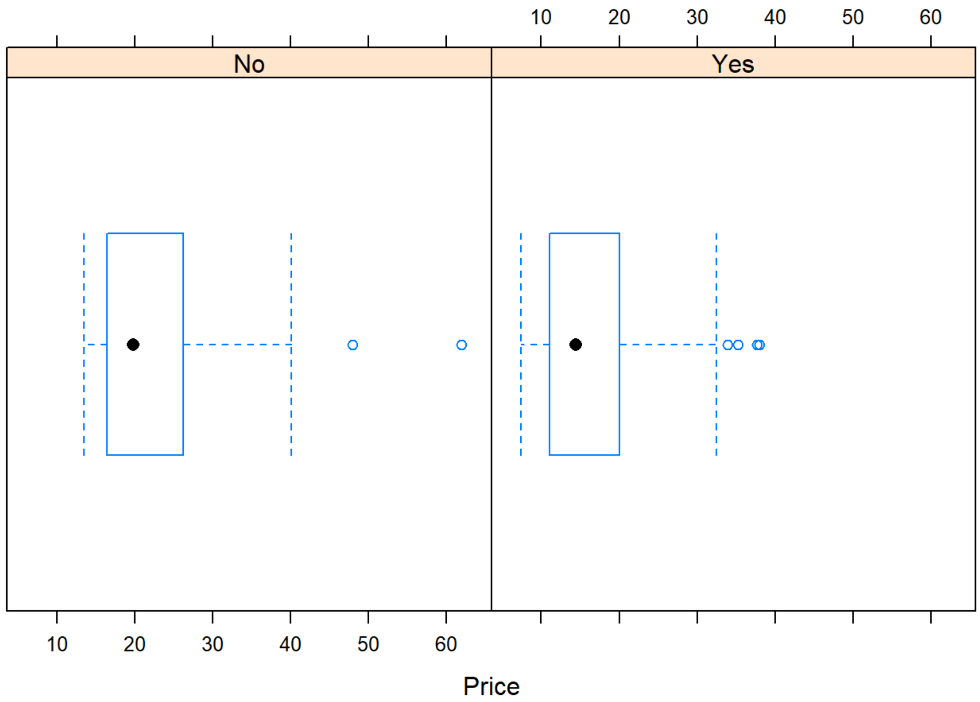
> bwplot(~ Price | Type, data = Cars93)



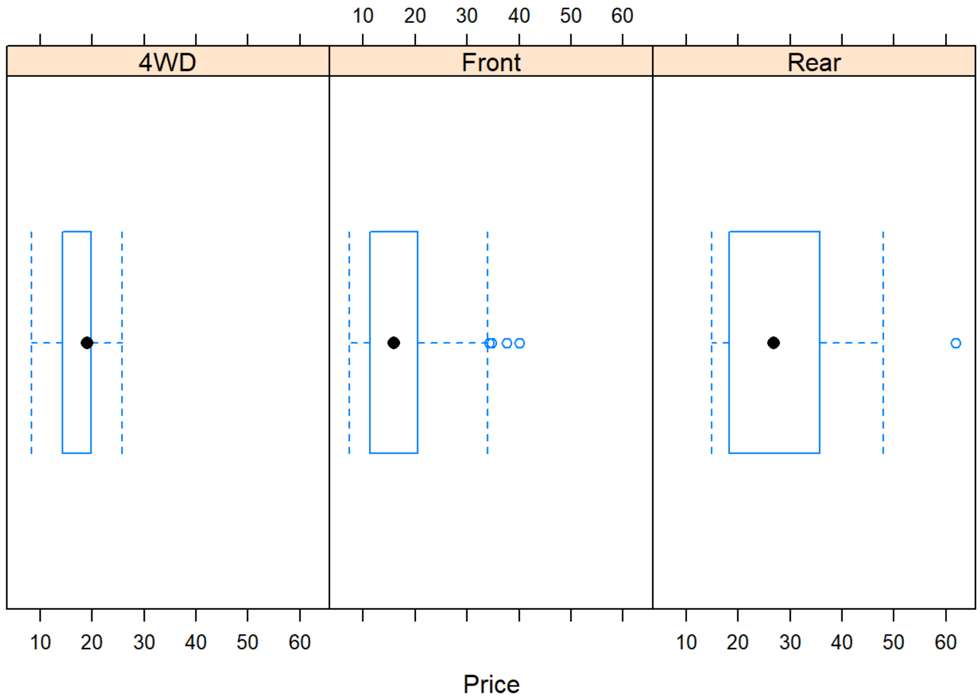
> bwplot(~ Price | AirBags, data = Cars93)



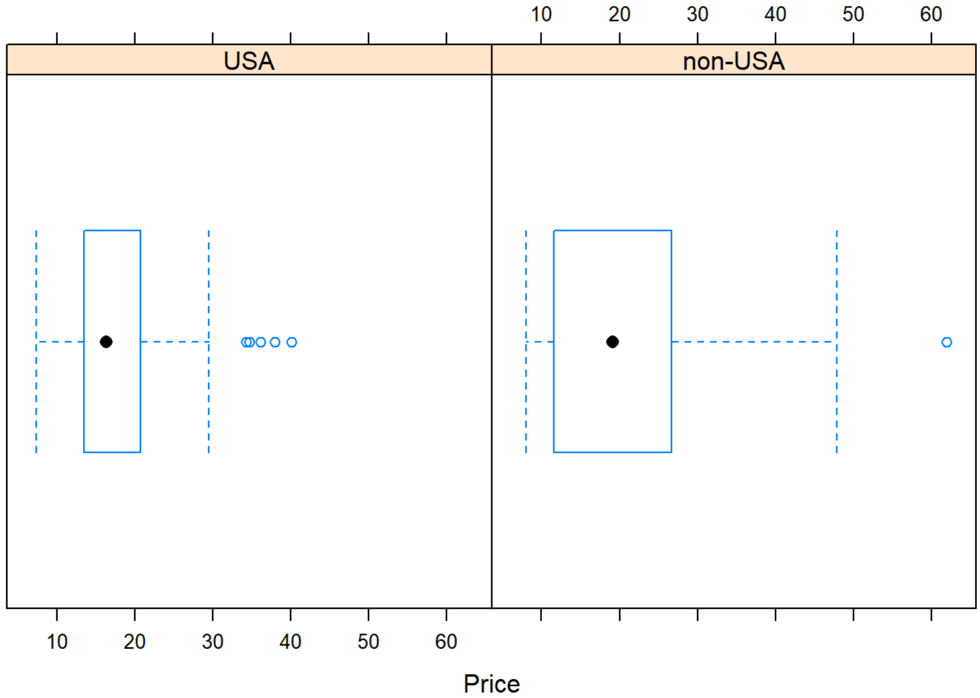
> bwplot(~ Price | Man.trans.avail, data = Cars93)



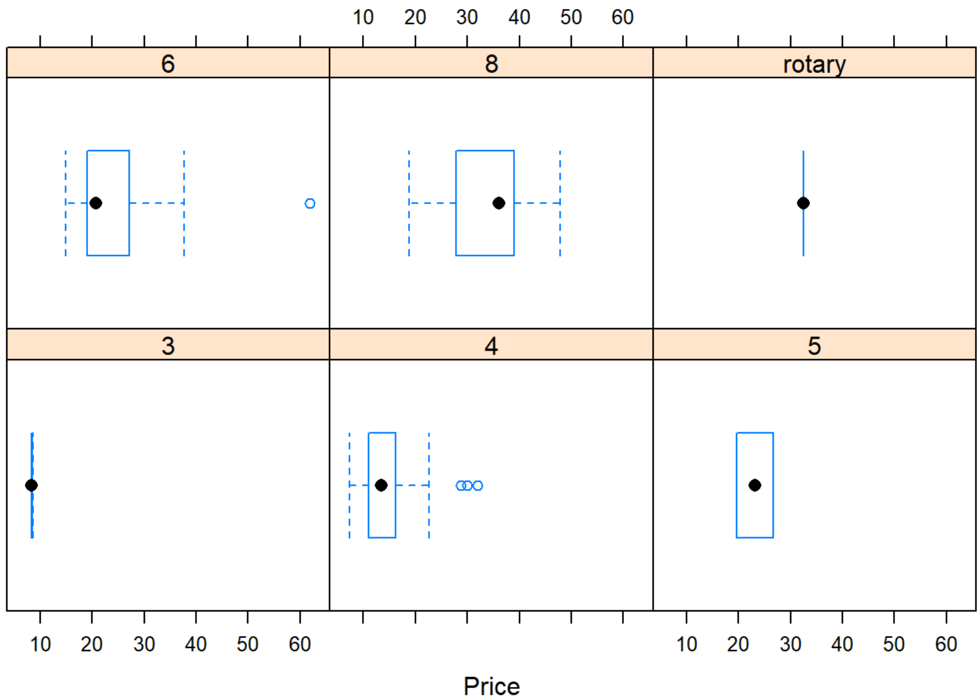
> bwplot(~ Price | DriveTrain, data = Cars93)



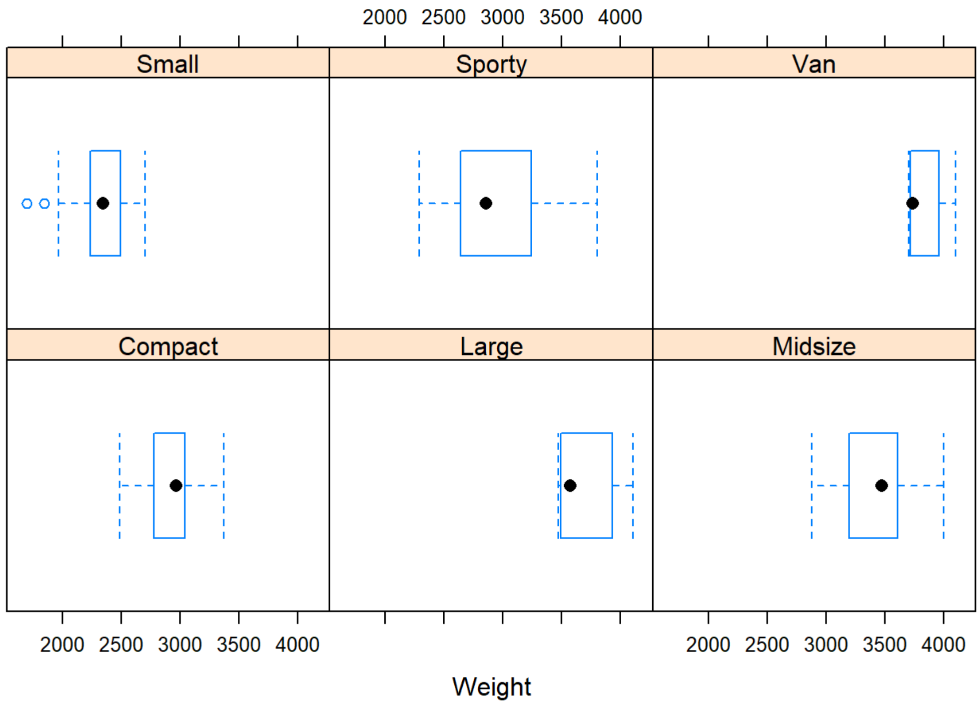
```
> bwplot( ~ Price | Origin, data = Cars93)
```



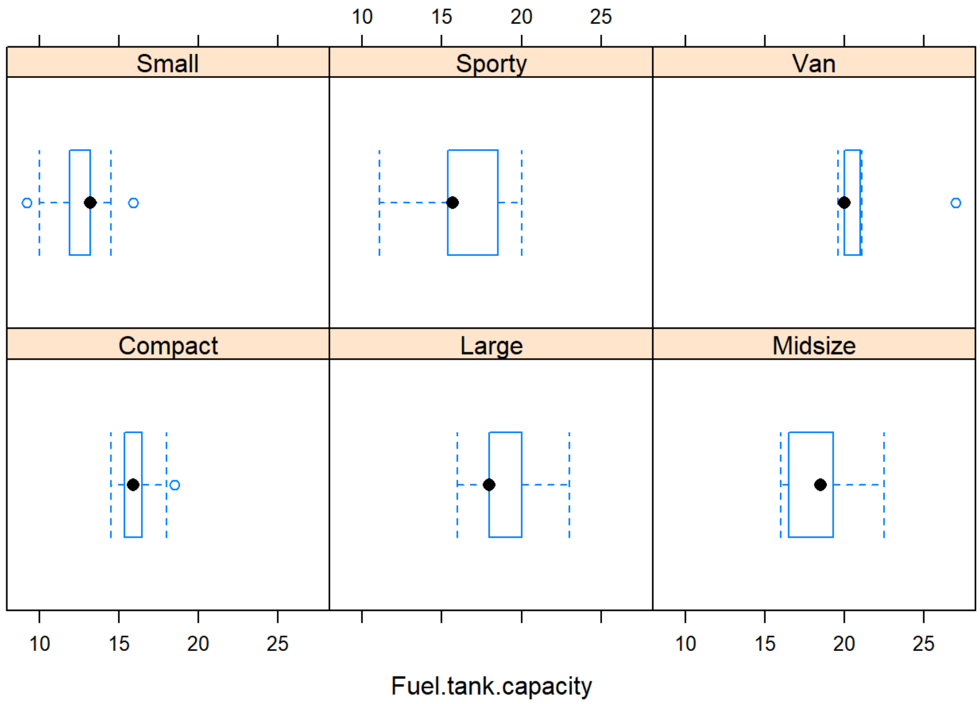

```
> bwplot( ~ Price | Cylinders , data = Cars93)
```



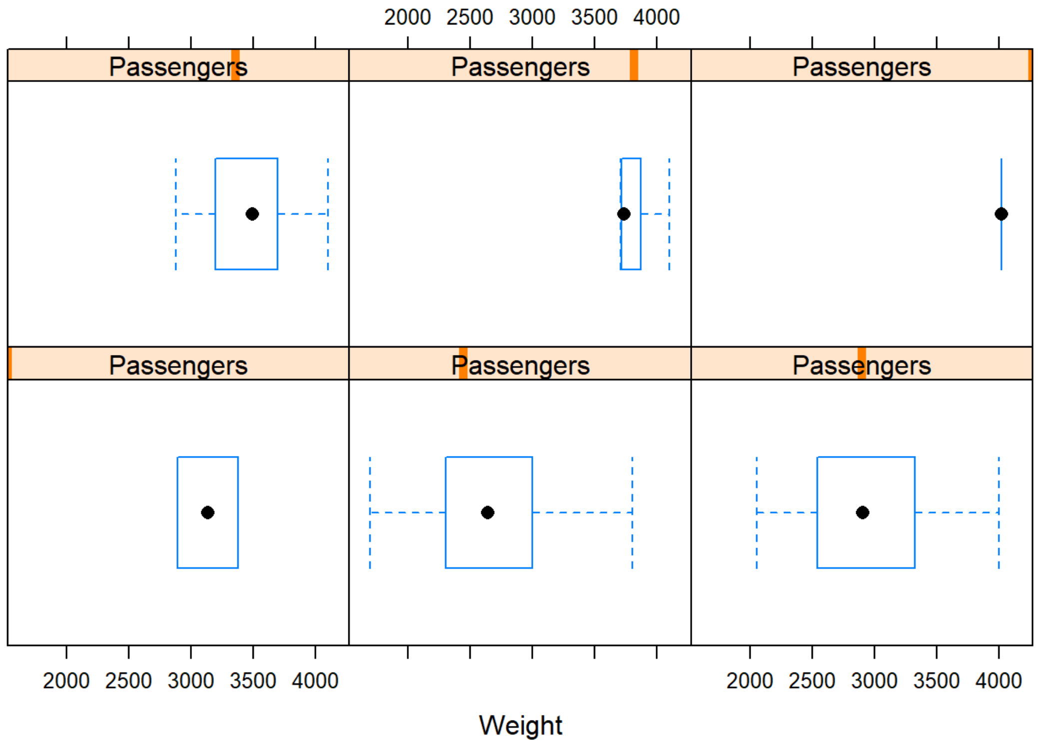
```
> bwplot( ~ Weight | Type , data = Cars93)
```



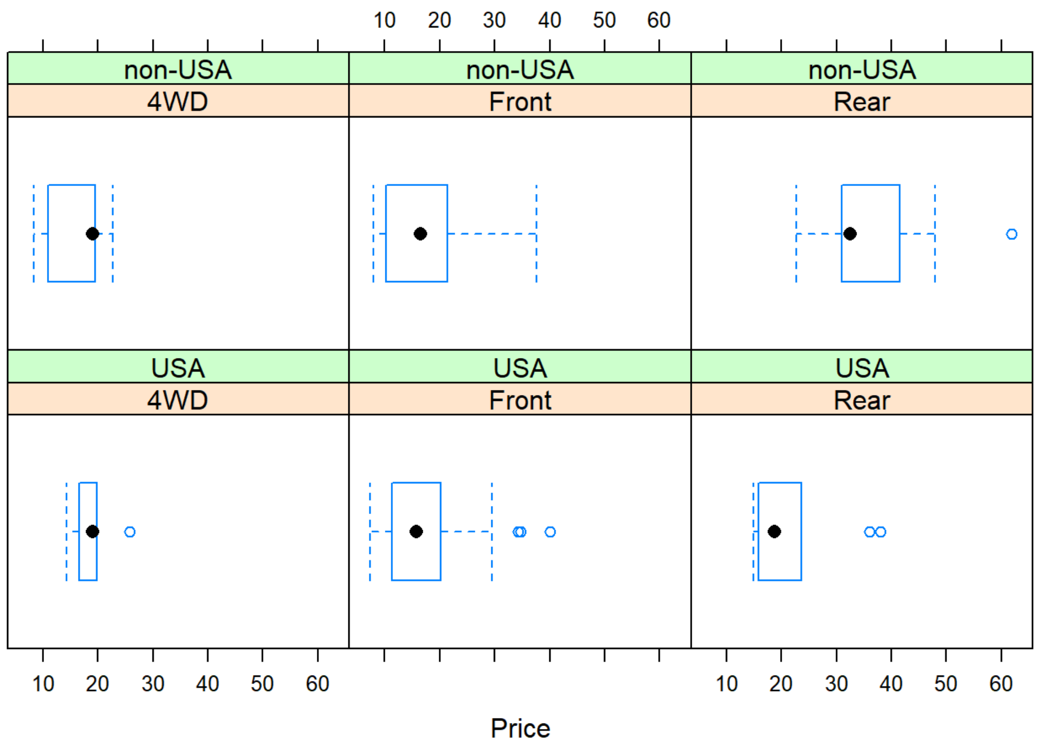

```
> bwplot( ~ Fuel.tank.capacity | Type , data = Cars93)
```



> bwplot(~ Weight | Passengers , data = Cars93)



> bwplot(~ Price | DriveTrain * Origin, data = Cars93)



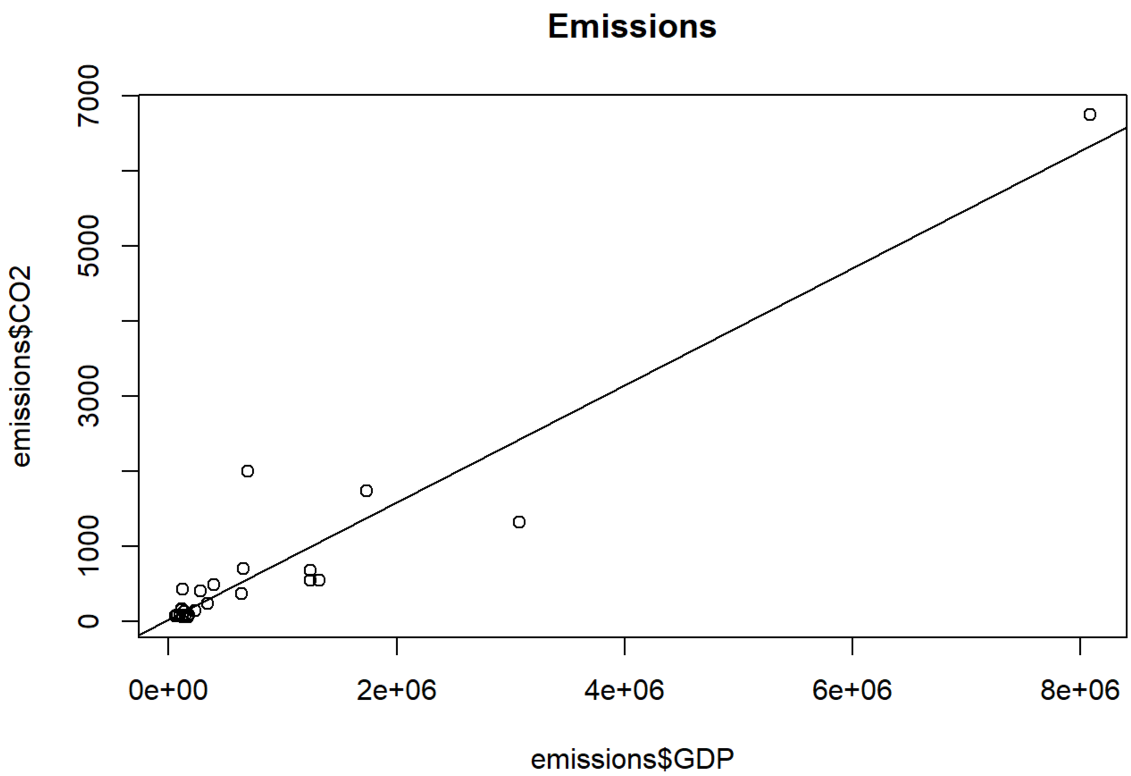
Numerical data

For the next example let's review the emissions data set containing gross domestic product (GDP), gross domestic product (GDP) per capita and CO2 emissions estimated for 26 countries in 1999.

```
> head(emissions)
                  GDP perCapita  CO2
UnitedStates  8083000     29647 6750
Japan         3080000     24409 1320
Germany       1740000     21197 1740
France        1320000     22381  550
UnitedKingdom 1242000     21010  675
Italy         1240000     21856  540
```

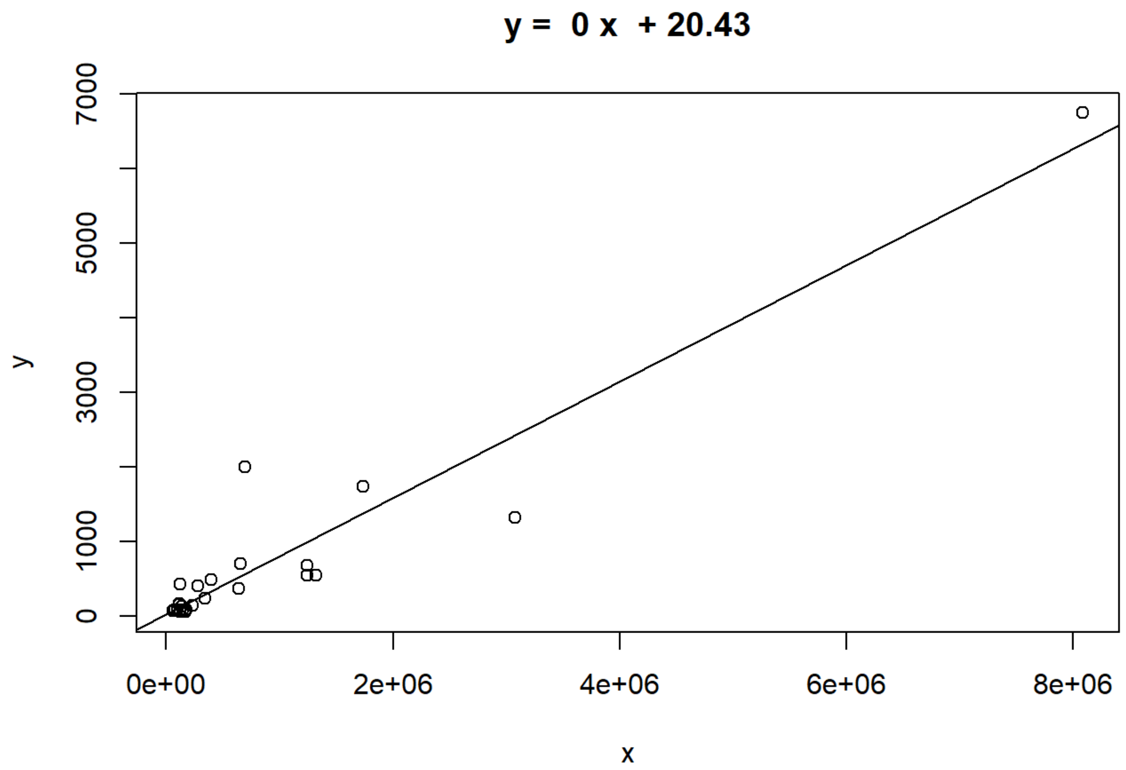
Is there a relationship between the gross domestic product and the CO2 emissions?

```
> cor(emissions$GDP, emissions$CO2)
[1] 0.95
> plot(emissions$CO2 ~ emissions$GDP, main = "Emissions")
> abline(lm(emissions$CO2 ~ emissions$GDP))
```



Another way was using simple.lm function from the UsingR package

> simple.lm(emissions$GDP, emissions$CO2)

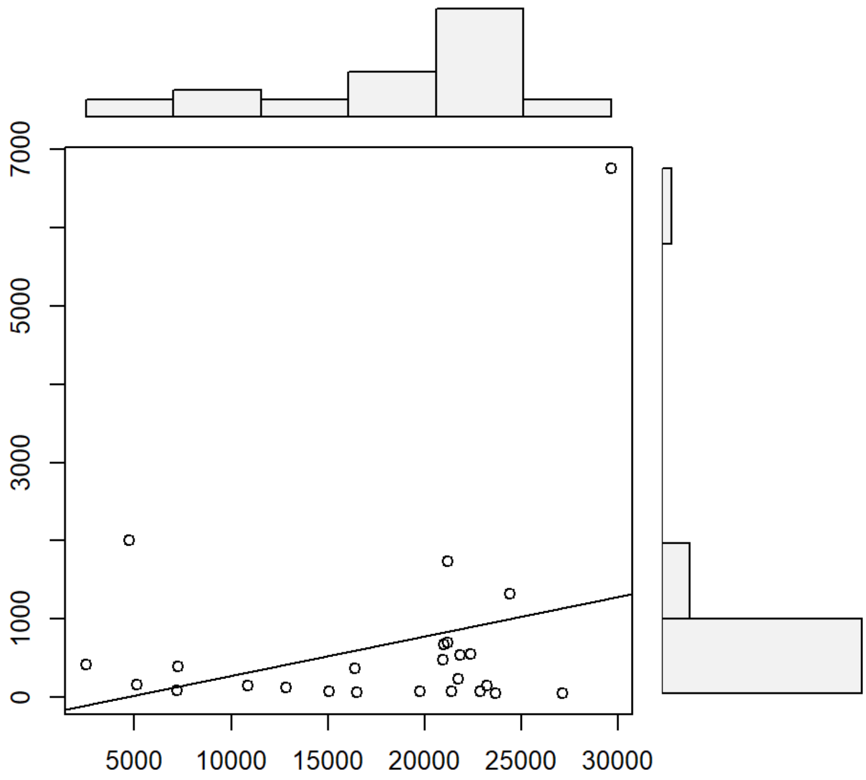


Call:
lm(formula = y ~ x)

Coefficients:
(Intercept) x
2.04e+01 7.81e-04

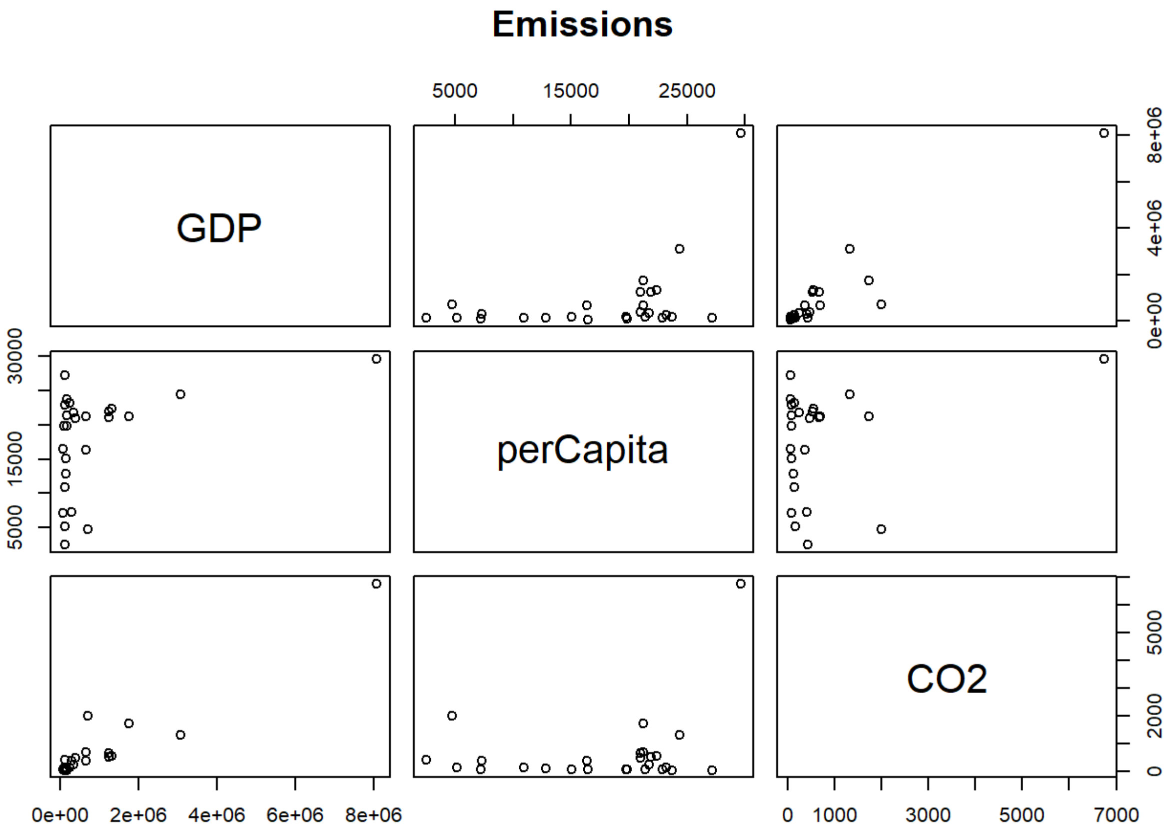
We can also see the distributions and the relation ship at the same time, we can use simple.scatterplot function.

```
> simple.scatterplot(emissions$perCapita, emissions$CO2)
```



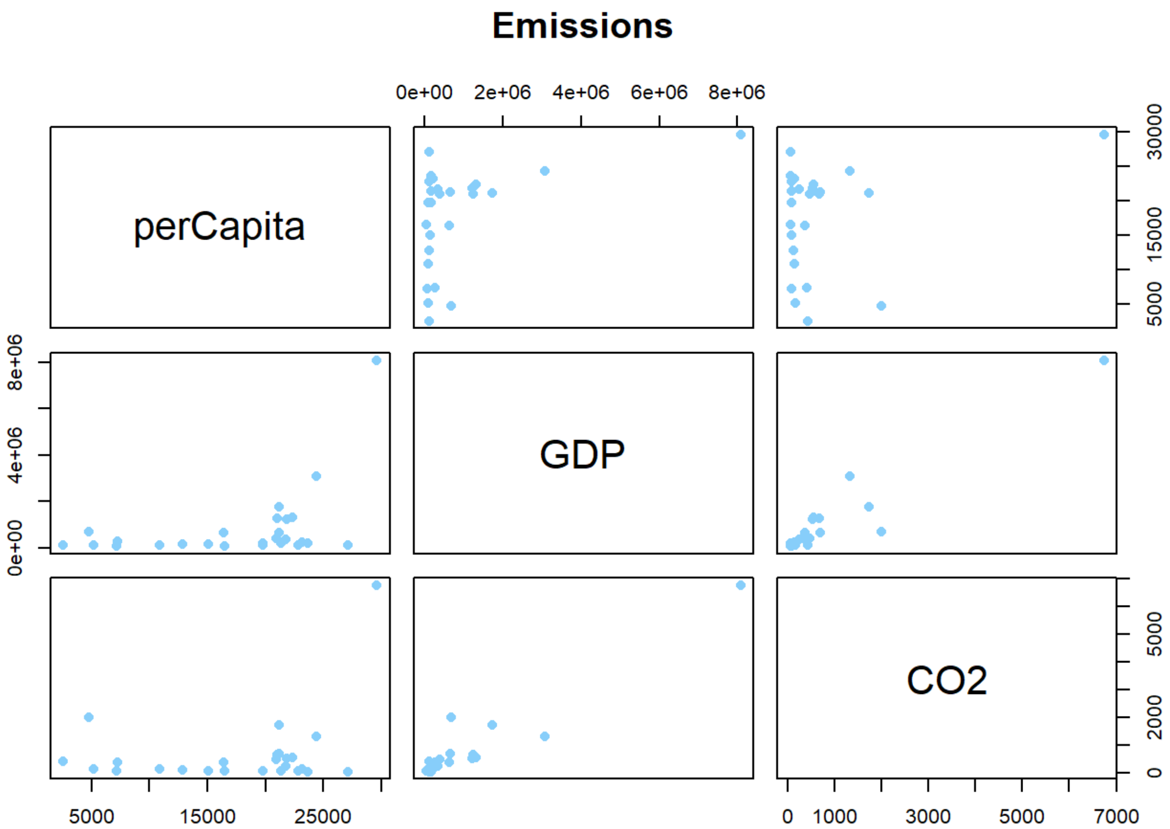
And we see that we can make plots for all the variables in the data frame simultaneously using the pairs function, there are a lot customize attributes for this function

```
> pairs(emissions, main = "Emissions")
```



For example using this syntax we can chose which variables to include in the model and how to order them. Also as we already saw by using col and pch we can change the color and the symbol of data representation.

```
> pairs( ~ perCapita + GDP + CO2, data = emissions,
+       main = "Emissions",
+       col = "lightskyblue",
+       pch = 16)
```



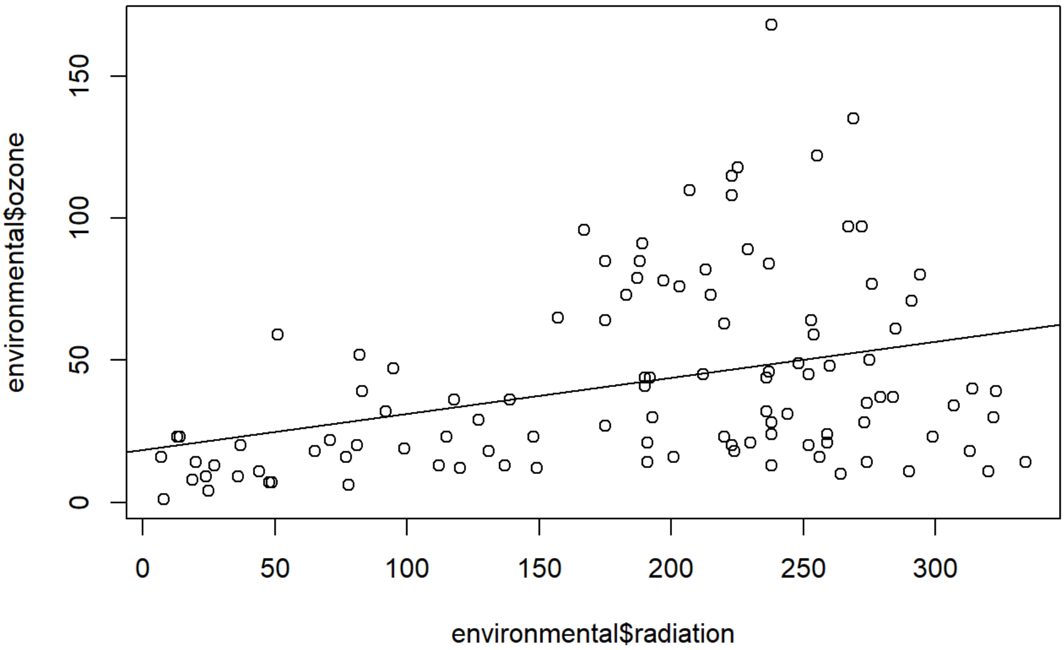
For the next example we are going to review environmental data frame from lattice package showing daily measurements of average ozone concentration (of hourly measurements) in parts per billion, solar radiation (from 08:00 to 12:00) in langleys, maximum temperature in Fahrenheit and average wind speed (at 07:00 and 10:00) in miles per hour in New York City from May to September of 1973.

```
> head(environmental)
  ozone radiation temperature wind
1    41       190          67  7.4
2    36       118          72  8.0
3    12       149          74 12.6
4    18       313          62 11.5
5    23       299          65  8.6
6    19        99          59 13.8
```

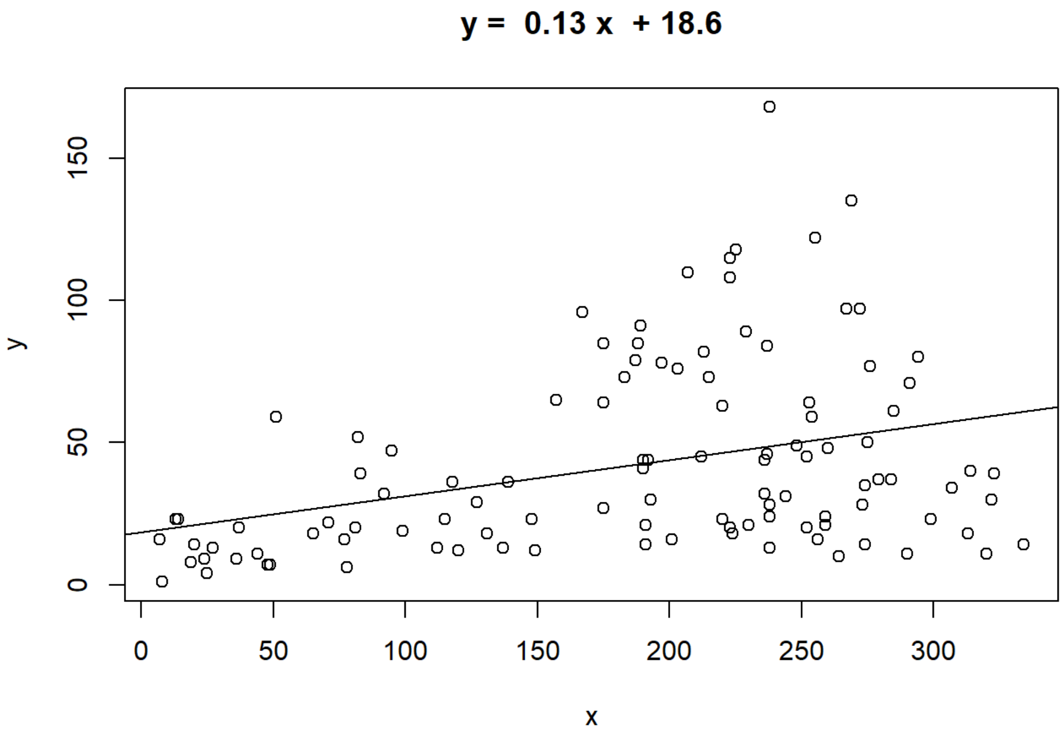
By using the base plotting system we can plot the data for ozone and radiation as above

```
> cor(environmental$radiation, environmental$ozone)
[1] 0.35
```

```
> plot(environmental$ozone ~ environmental$radiation)
> abline(lm(environmental$ozone ~ environmental$radiation))
```



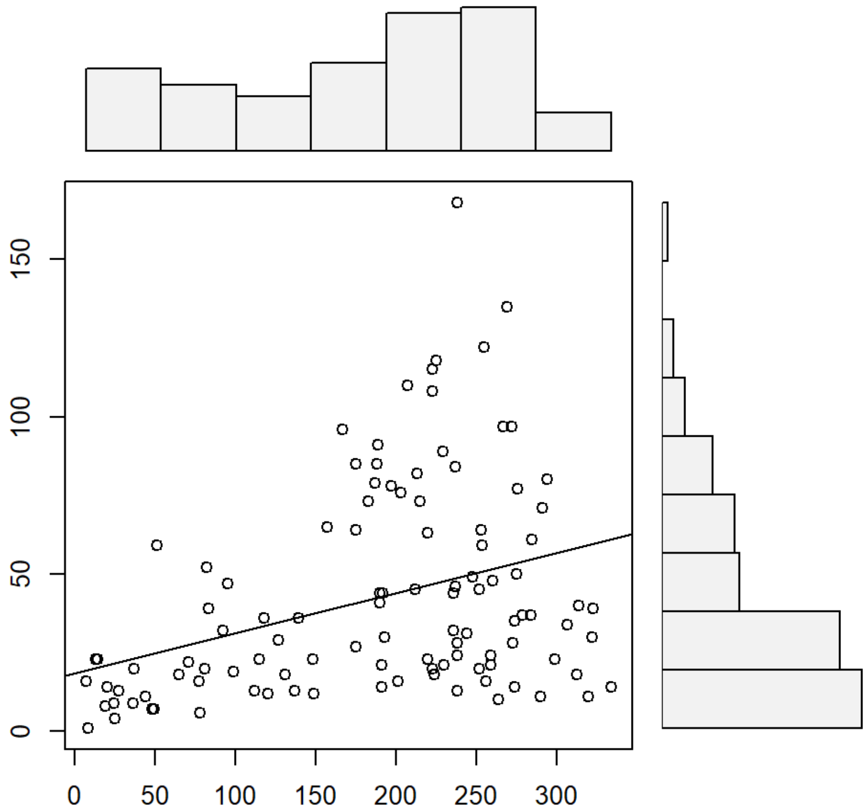
> simple.lm(environmental$radiation, environmental$ozone)



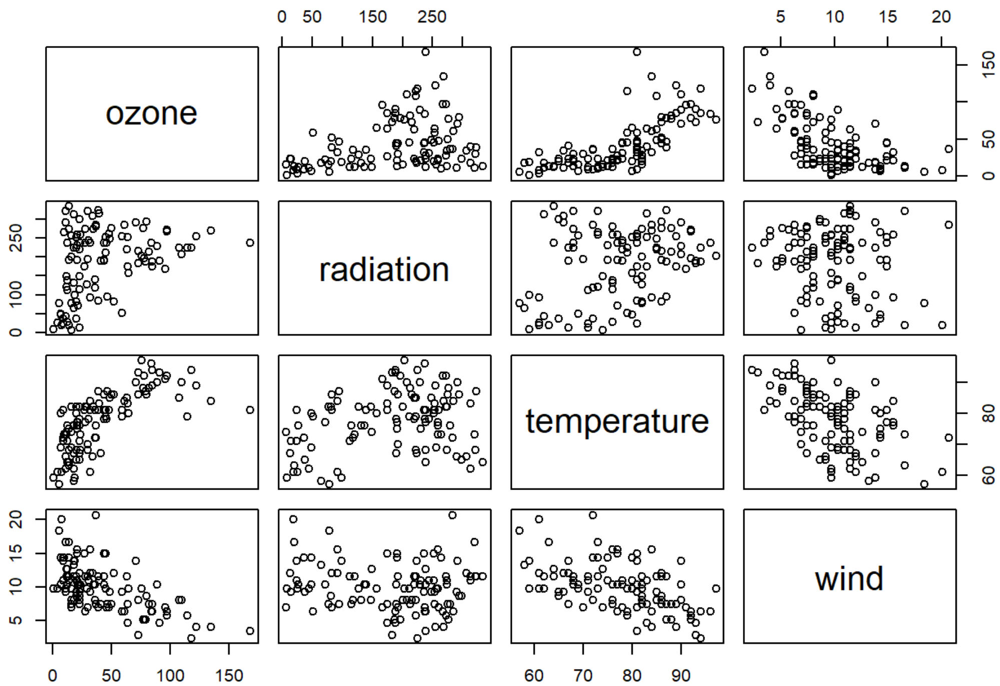

```
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)           x
     18.599       0.127
```

```
> simple.scatterplot(environmental$radiation, environmental$ozone)
```

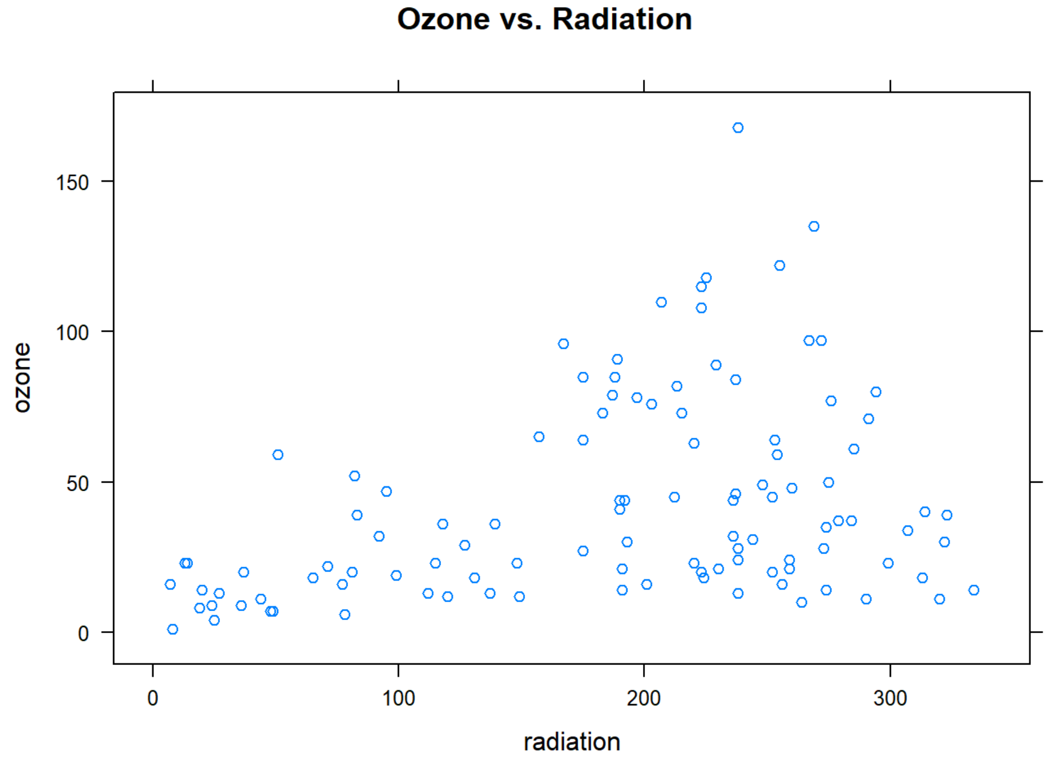



```
> pairs(environmental)
```



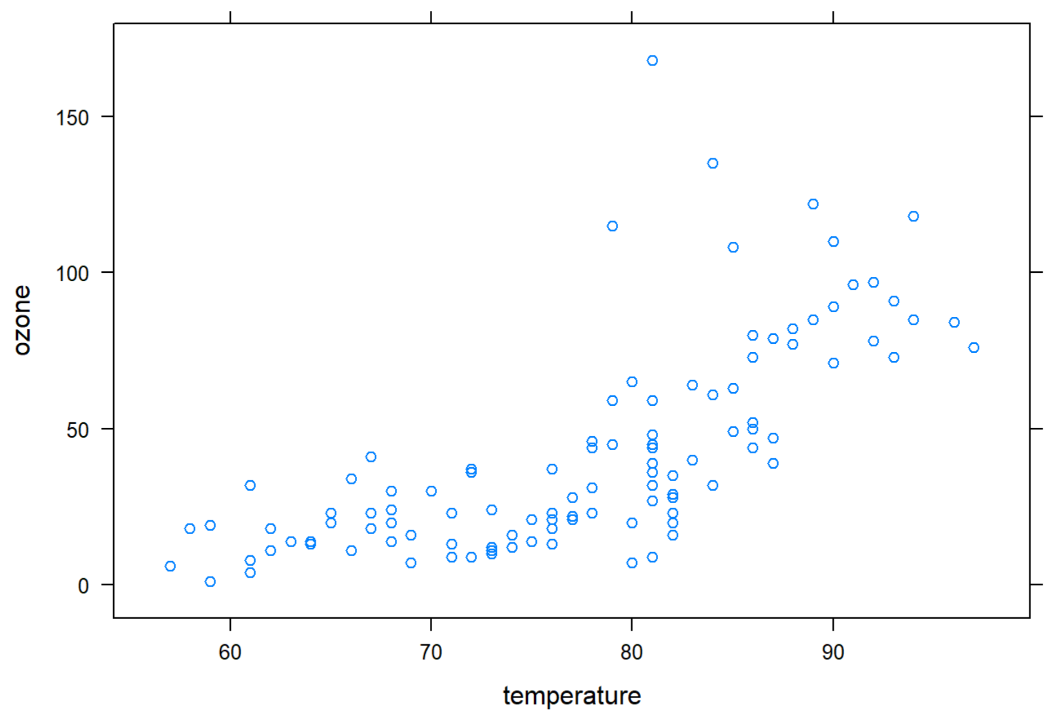
We can use xyplot from lattice package to make the scatterplot of ozone and radiation. We use the same syntax as above response ~ predictor

```
> xyplot(ozone ~ radiation, data = environmental,
+        main = "Ozone vs. Radiation")
```



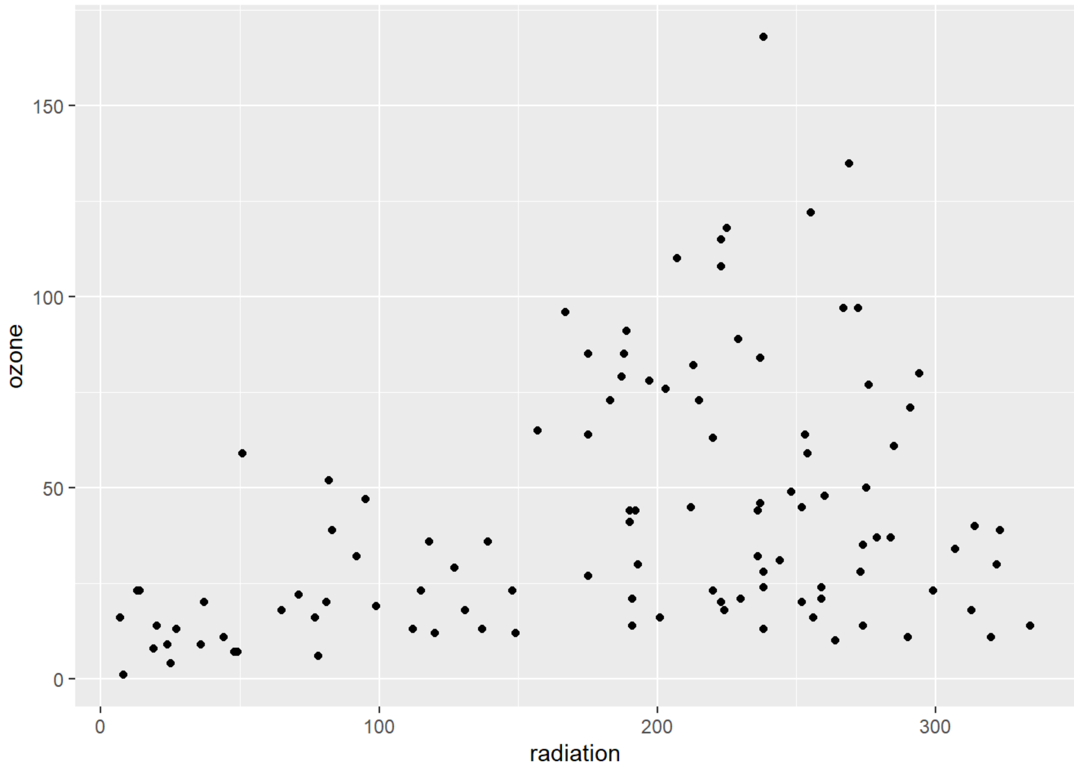
Also we can see the relation between ozone and temperature

```
> xyplot(ozone ~ temperature, data = environmental)
```

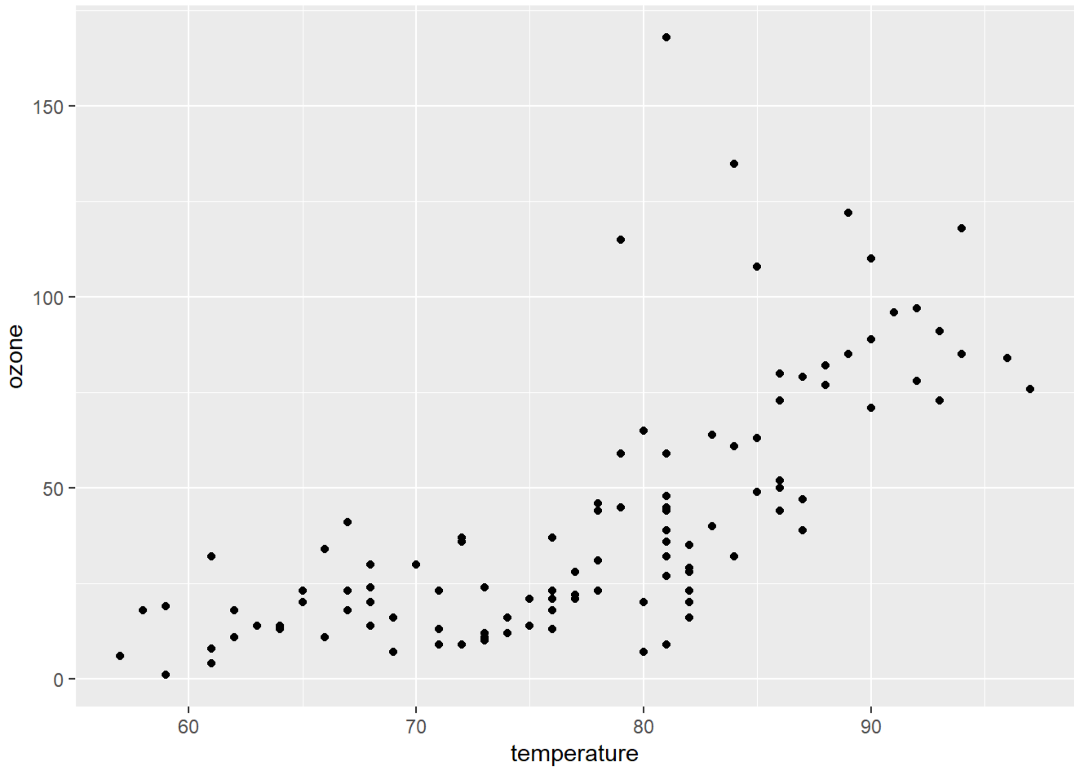


We can also use qplot from ggplot package to make the scatterplot of ozone and radiation.

```
> qplot(radiation, ozone, data = environmental)
```




```
> qplot(temperature, ozone, data = environmental)
```



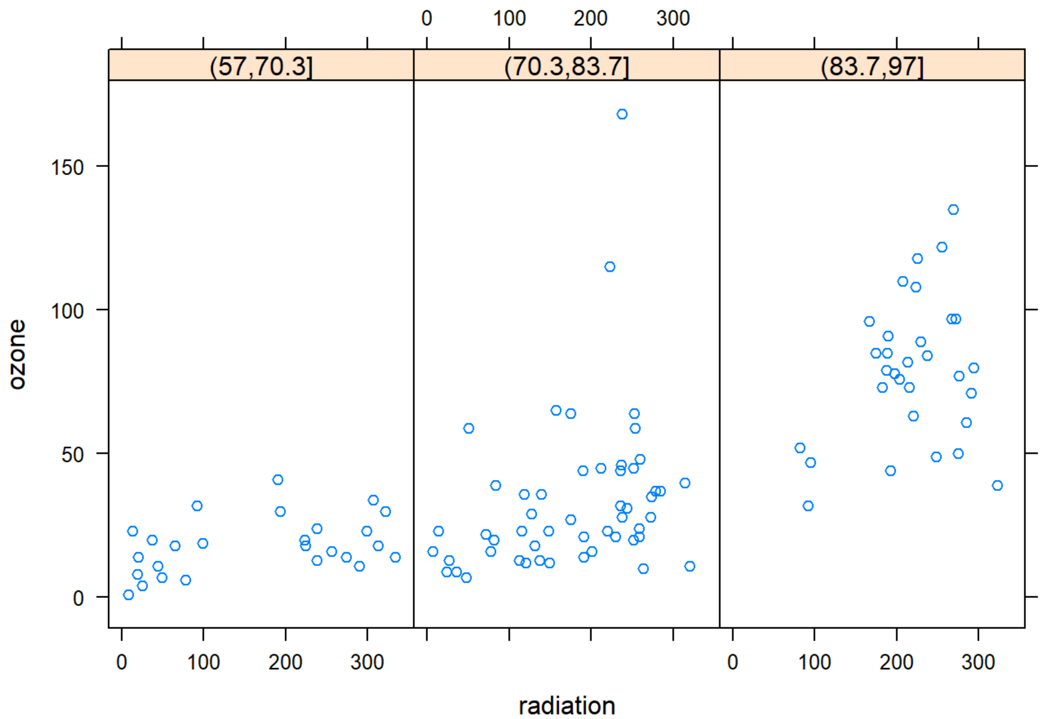
Does the relationship between ozone and radiation change as the temperature changes? We can cut the temperature interval in 3 subsets

```
> summary(environmental$temperature)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
     57      71      79      78      84      97 
```

```
> temperature.cut <- cut(environmental$temperature, 3)
```

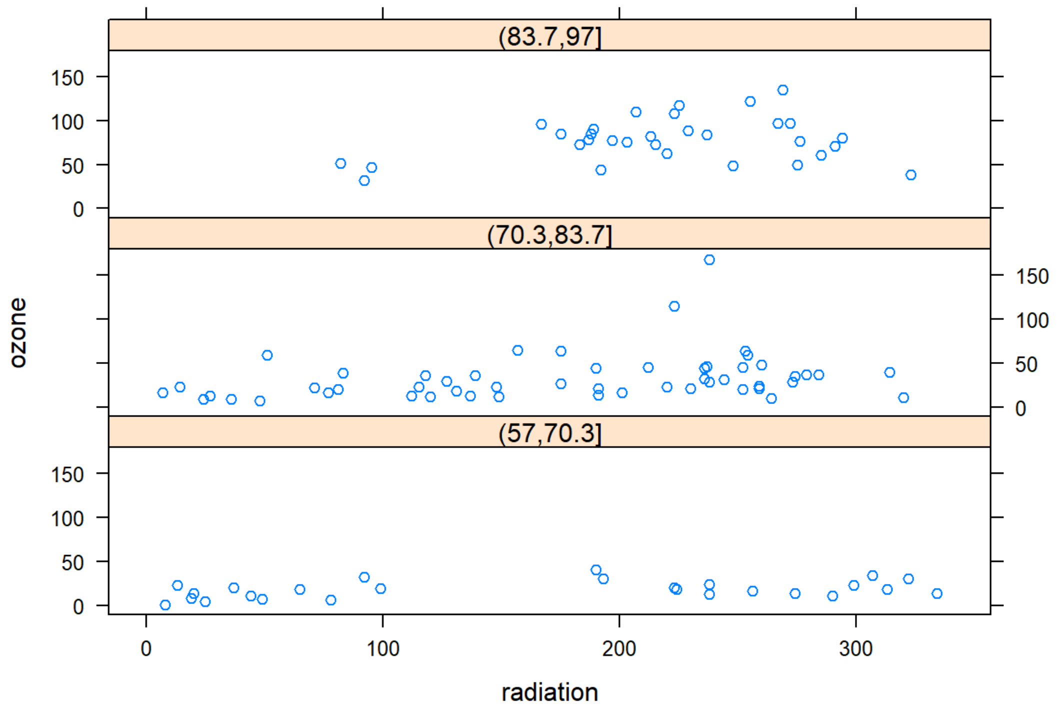
Continuing with lattice package we can make a scatterplot for every subset of temperature intervals

```
> xyplot(ozone ~ radiation | temperature.cut, data = environmental)
```



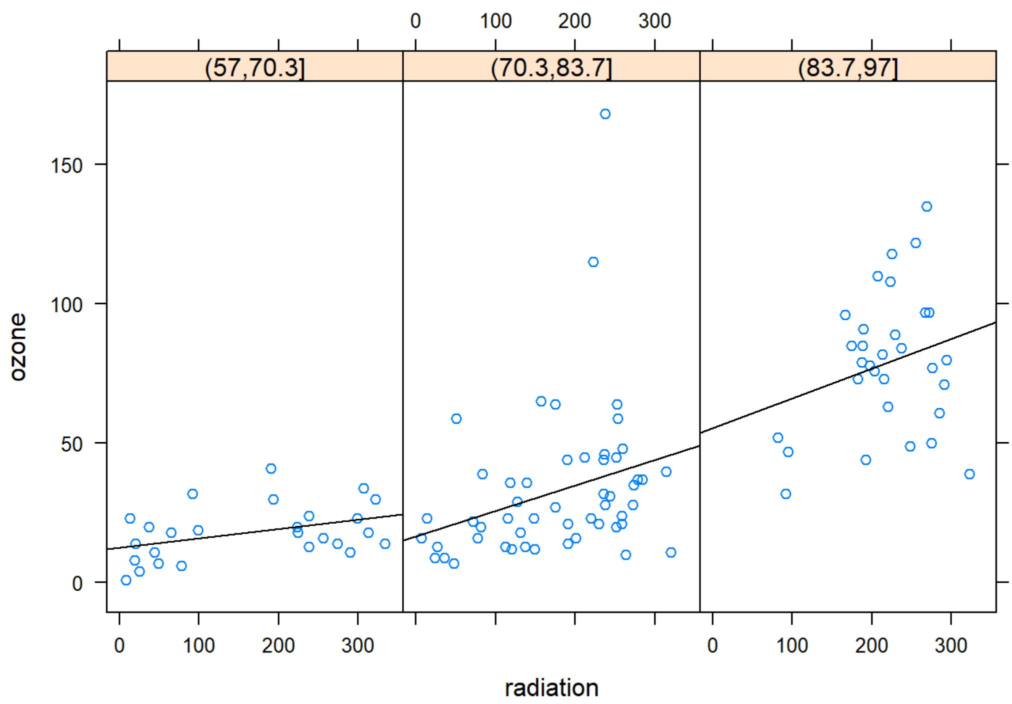
We can tune it up if we want. We can order the graphics in one column for example.

```
> xyplot(ozone ~ radiation | temperature.cut, data = environmental, layout = c(1, 3))
```



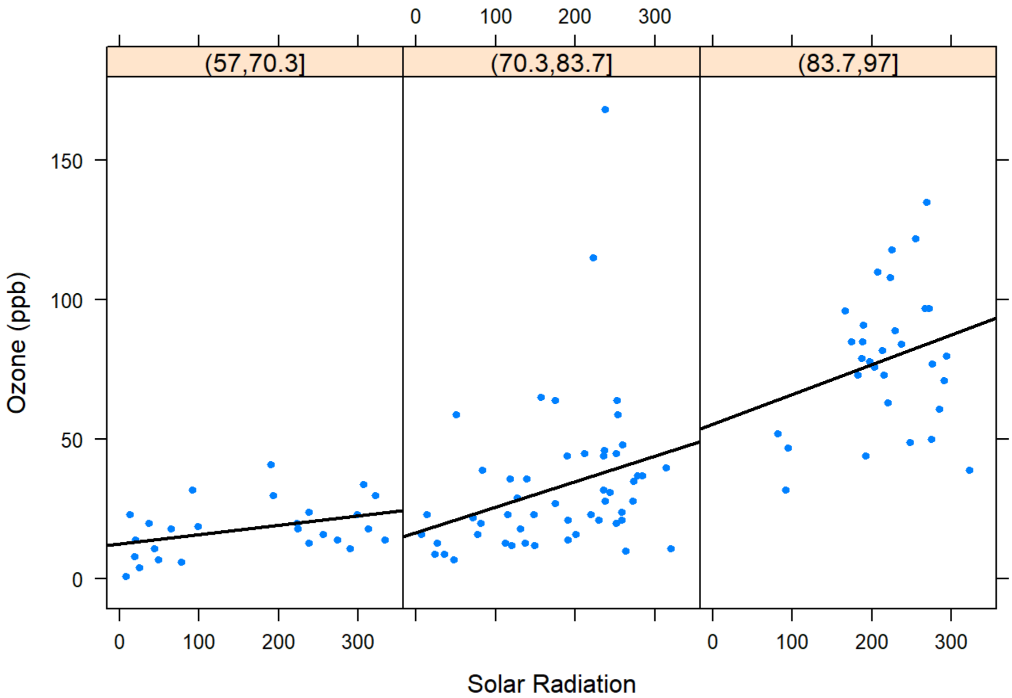
We can also add additional function to the panels. In this example we calculate a linear model and add a linear regression line to the panels.

```
> xyplot(ozone ~ radiation | temperature.cut, data = environmental,
+       panel = function(x, y, ...){
+         panel.xyplot(x, y, ...)
+         fit <- lm(y ~ x)
+         panel.abline(fit)
+       })
```



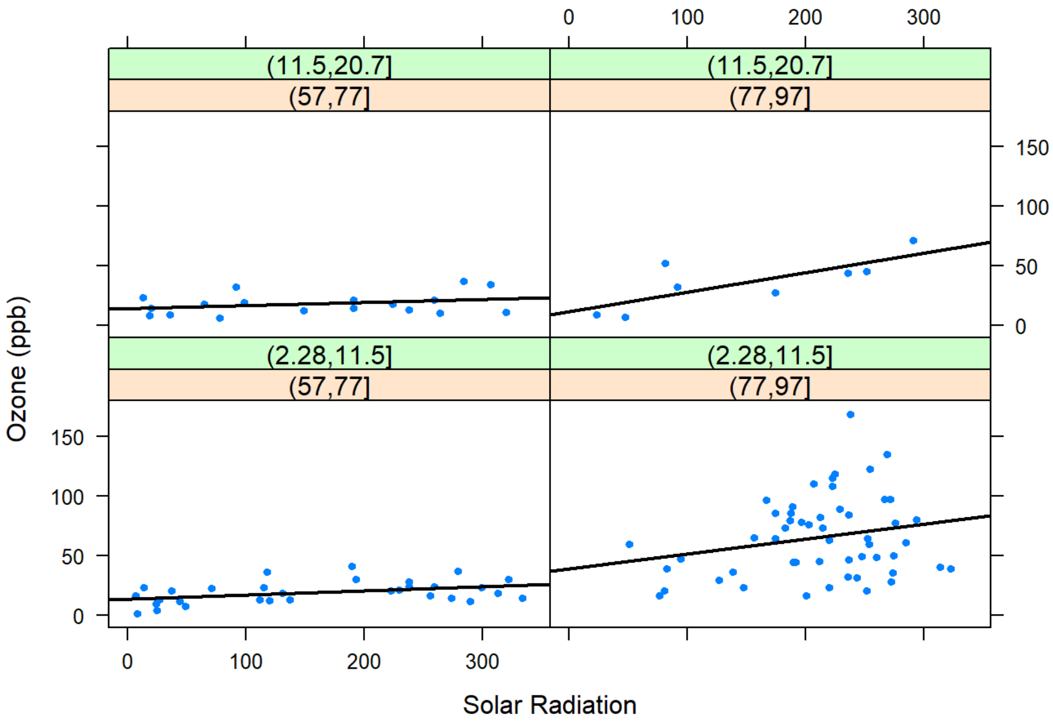
We can also change graphic's labels, colors and symbols.

```
> xyplot(ozone ~ radiation | temperature.cut, data = environmental,
+       panel = function(x, y, ...){
+         panel.xyplot(x, y, ...)
+         fit <- lm(y ~ x)
+         panel.abline(fit, lwd = 2)
+       },
+       xlab = "Solar Radiation",
+       ylab = "Ozone (ppb)",
+       pch = 20)
```



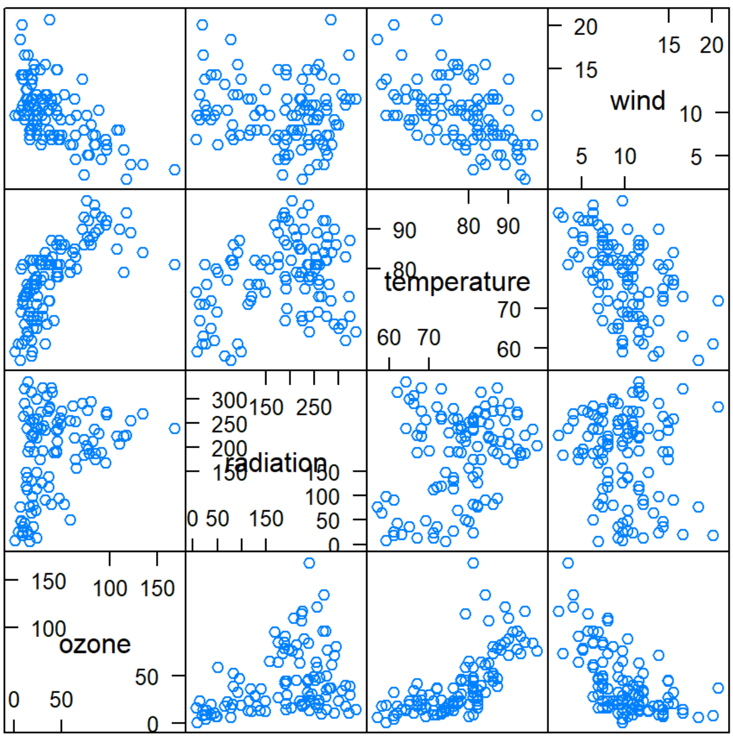
Let's see how ozone and solar radiation change when both temperature and wind change?

```
> temperature.cut <- cut(environmental$temperature, 2)
> wind.cut <- cut(environmental$wind, 2)
> xyplot(ozone ~ radiation | temperature.cut * wind.cut, data = environmental,
+       panel = function(x, y, ...){
+         panel.xyplot(x, y, ...)
+         fit <- lm(y ~ x)
+         panel.abline(fit, lwd = 2)
+       },
+       xlab = "Solar Radiation",
+       ylab = "Ozone (ppb)",
+       pch = 20)
```



As you see xyplot function is useful of making this kind of conditioning plots where we plot the relationship between two variables by conditioning on the values of third variable. splom form lattice is similar to pairs function base graphics package

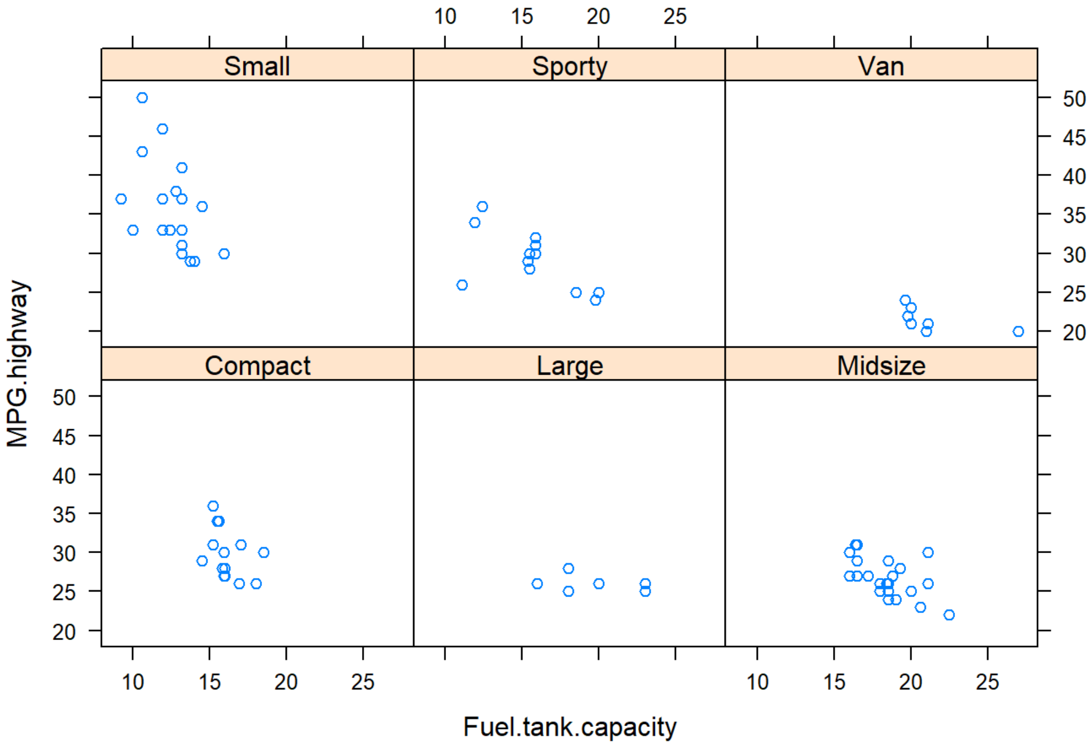
> splom(~ environmental)



Scatter Plot Matrix

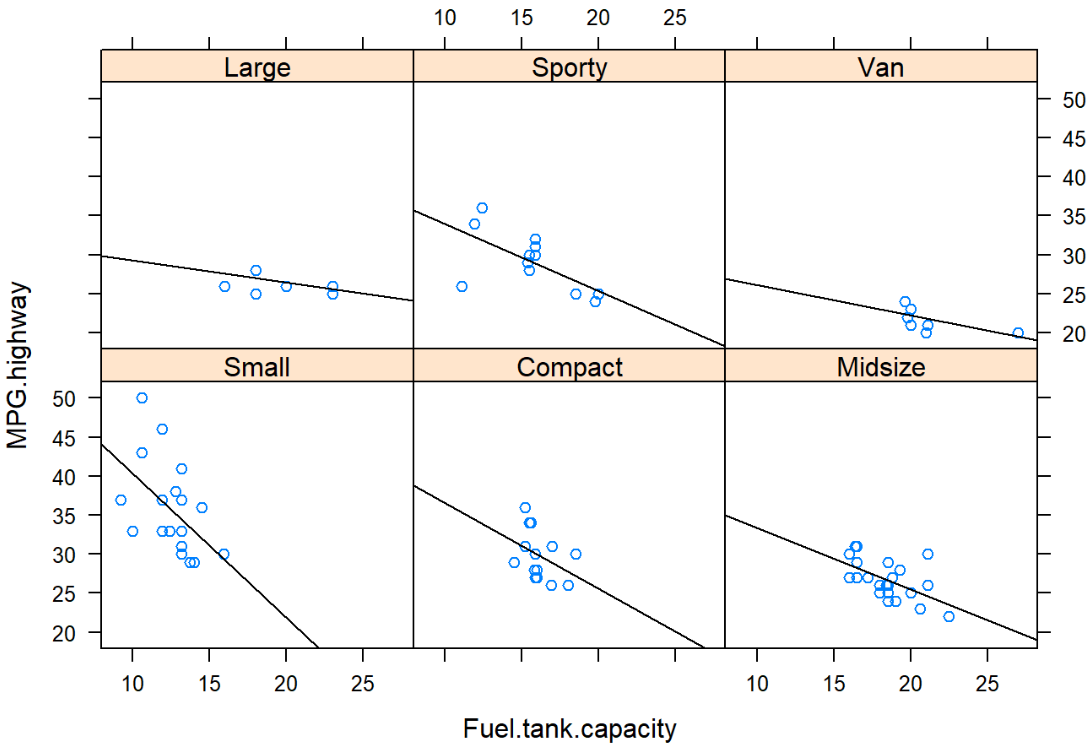
Another example again with Cars93 data frame for MPG.highway and Fuel.tank.capacity

```
> xyplot(MPG.highway ~ Fuel.tank.capacity | Type, data = Cars93)
```



Let's also add the linear regression lines

```
> plot.regression = function(x, y) {
+   panel.xyplot(x, y)
+   panel.abline(lm(y~x))
+ }
> xyplot(MPG.highway ~ Fuel.tank.capacity | Type, panel = plot.regression)
```

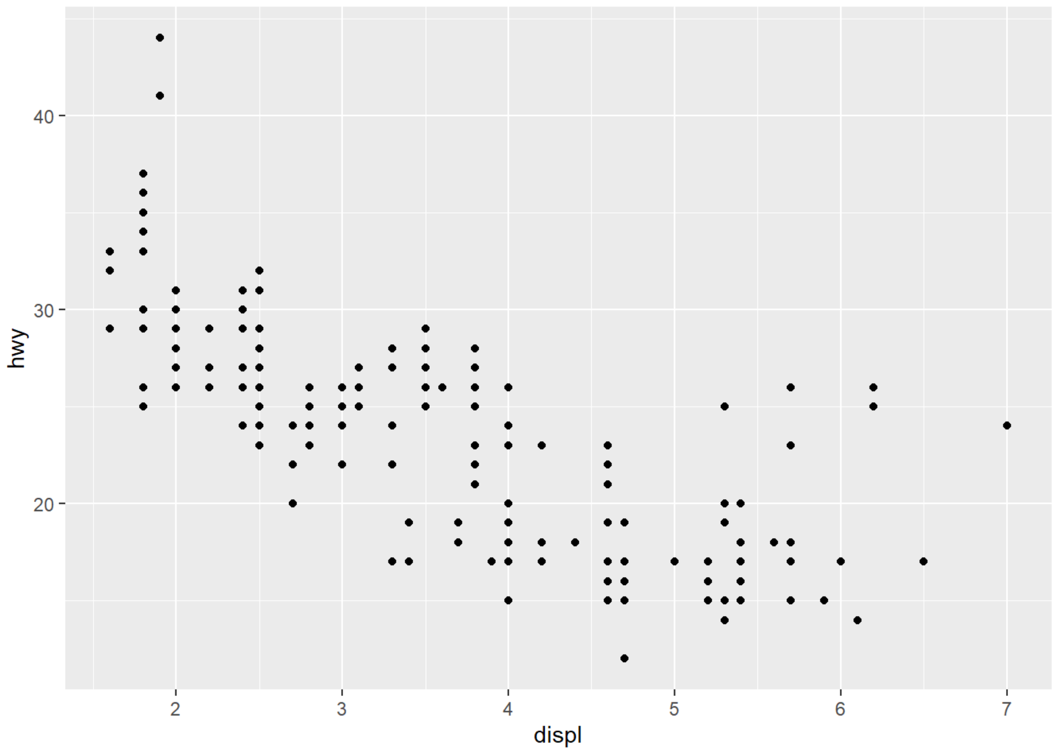


ggplot2 examples

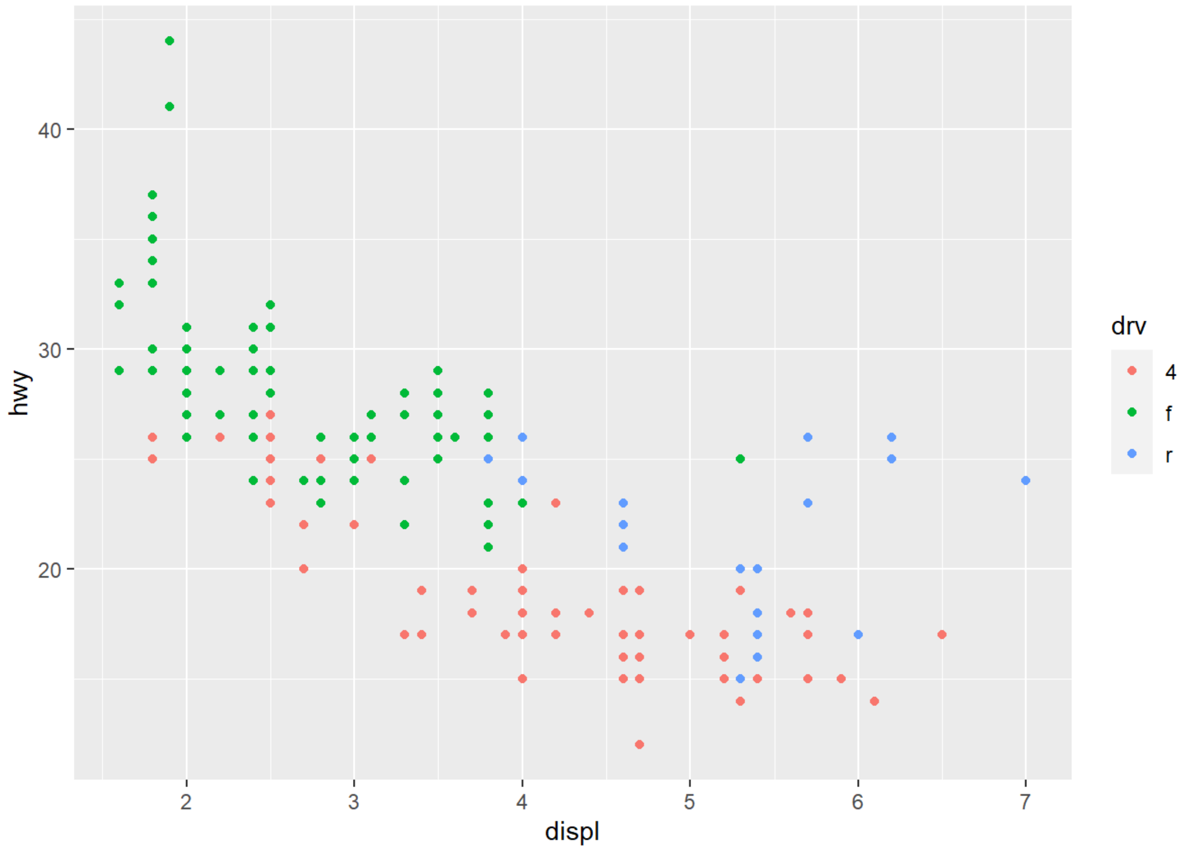
Another example with the mpg data frame reviewed using qplot function from ggplot package

```
> 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      1.8  1999     4 auto(l5)   f        18    29 p     compa~
2 audi         a4      1.8  1999     4 manual(m5) f        21    29 p     compa~
3 audi         a4      2    2008     4 manual(m6) f        20    31 p     compa~
4 audi         a4      2    2008     4 auto(av)   f        21    30 p     compa~
5 audi         a4      2.8  1999     6 auto(l5)   f        16    26 p     compa~
6 audi         a4      2.8  1999     6 manual(m5) f        18    26 p     compa~
```

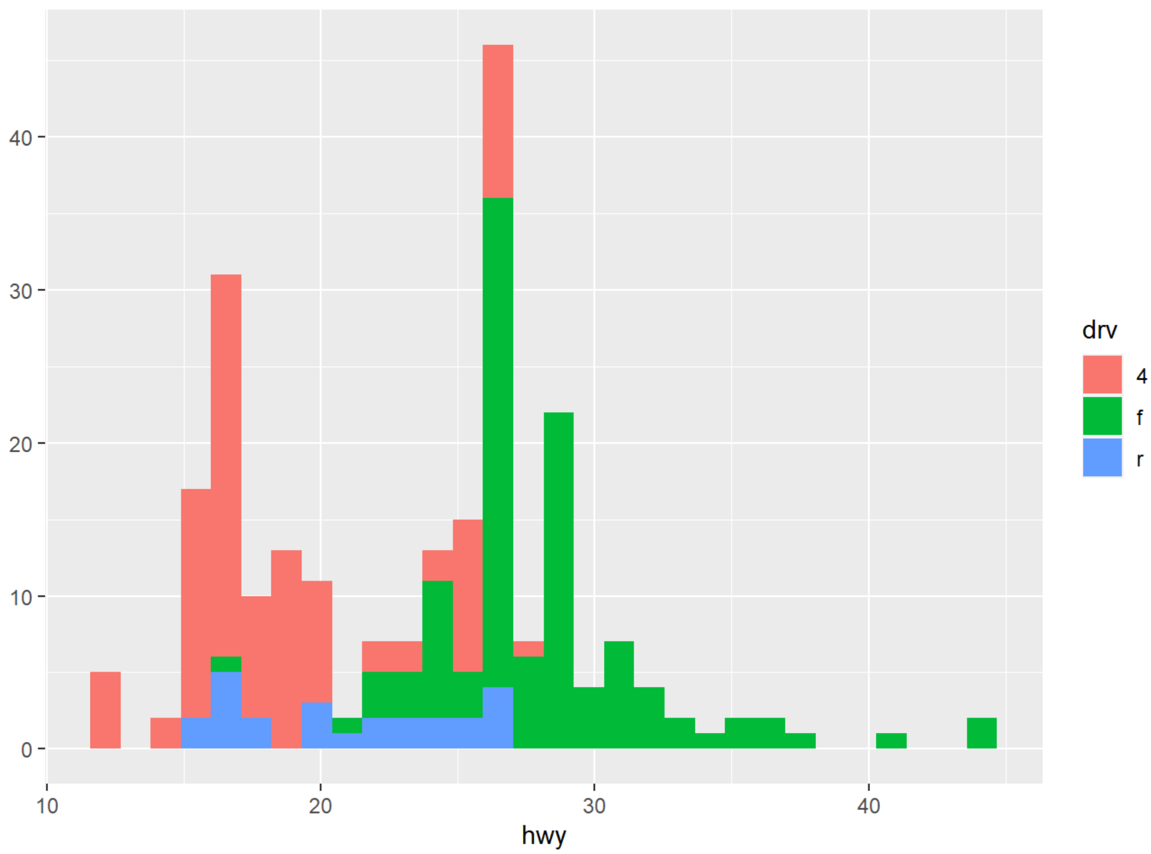
```
> qplot(displ, hwy, data = mpg)
```



```
> qplot(displ, hwy, data = mpg, color = drv)
```

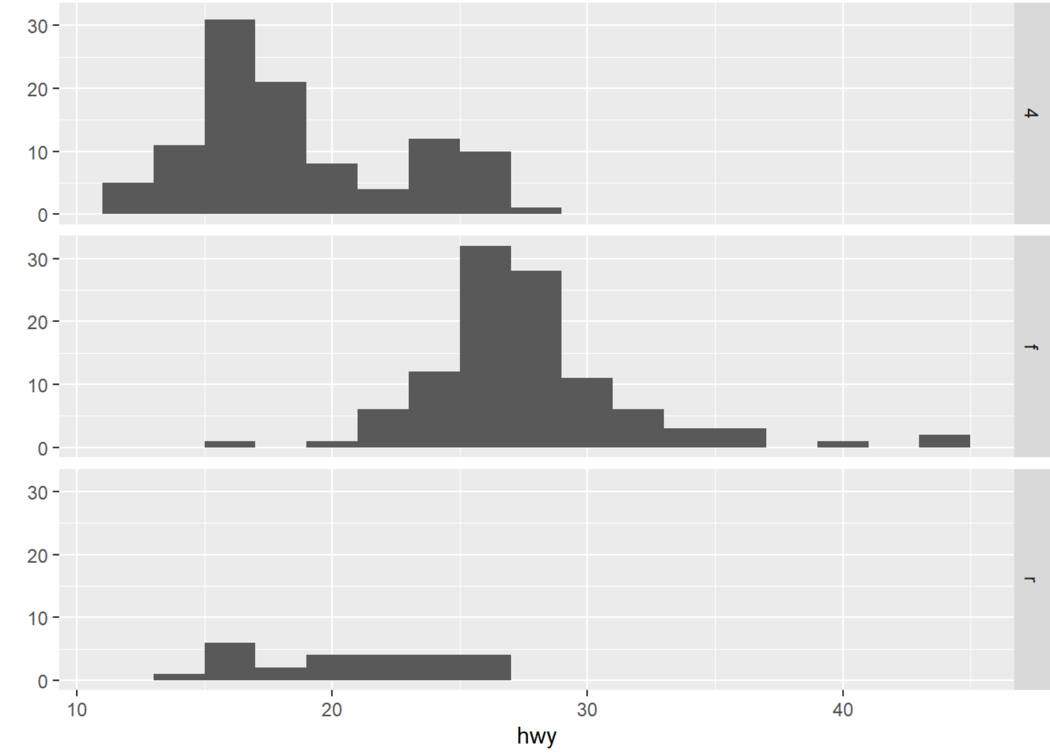



```
> qplot(hwy, data = mpg, fill = drv)
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

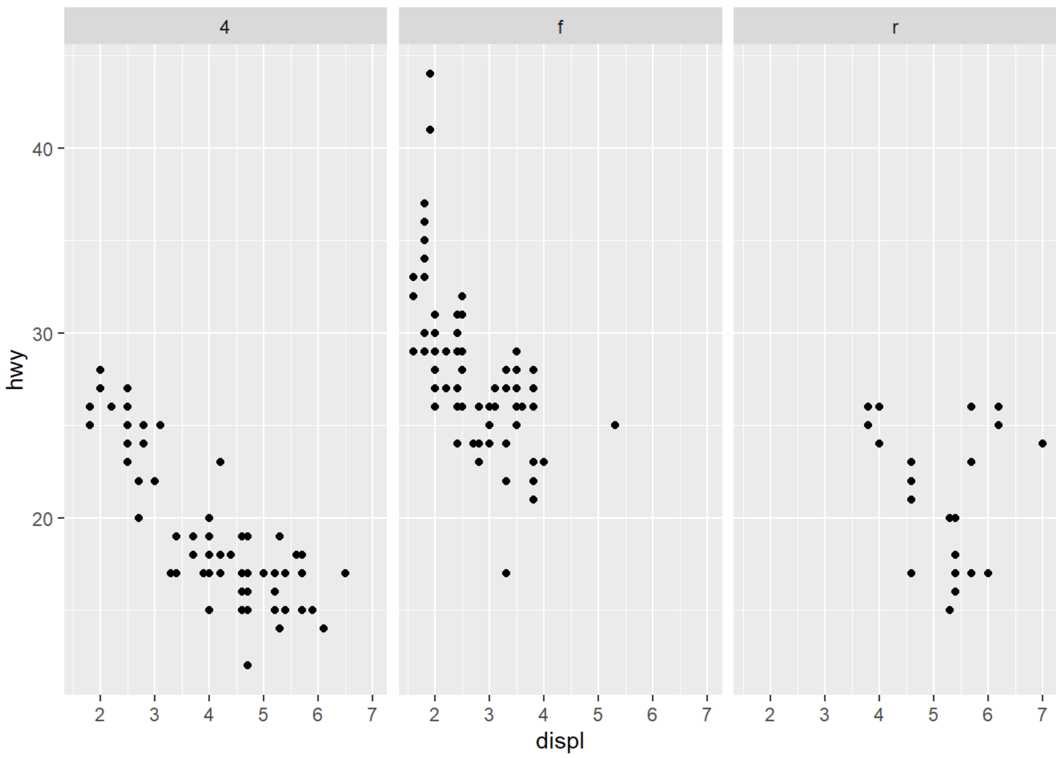


facets are like panels in lattice. The idea is that you can create a separate plot for any subset of your data which are determined by a factor variable. All of this plots are depicted in different panels in one and the same figure. When the factor variable is on the left side of the ~ its categories determine the facets in the different rows. When it is on the right side it determines the facets in the different columns. . indicates that we don’t have an argument, it can also be empty.

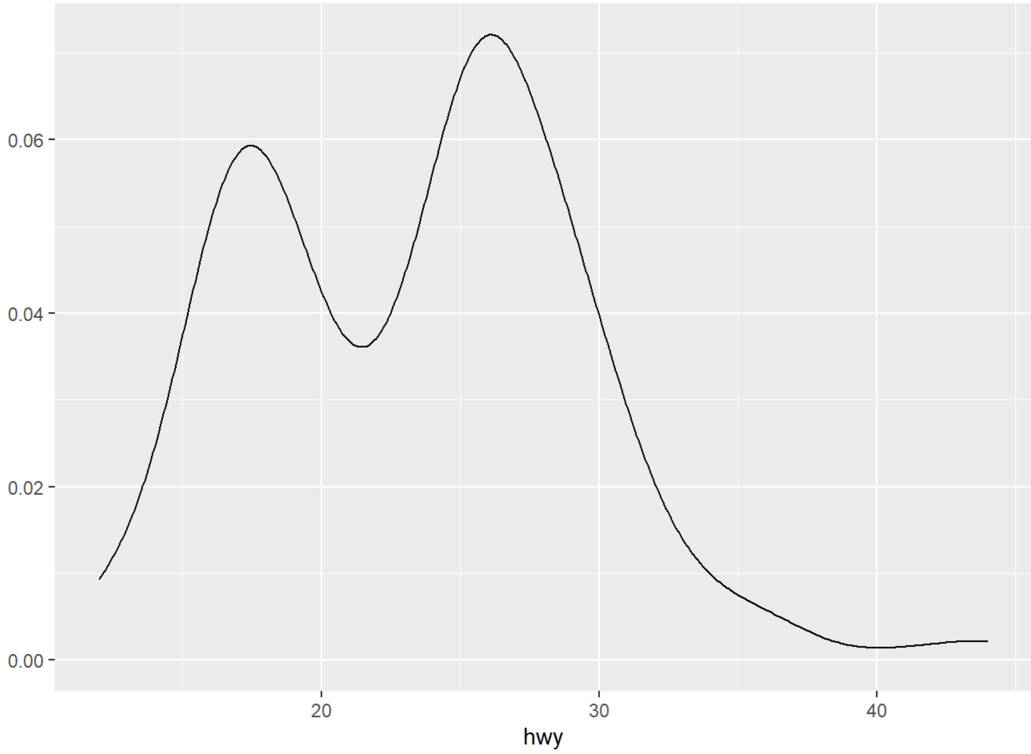
```
> qplot(hwy, data = mpg, facets = drv~., binwidth = 2)
```



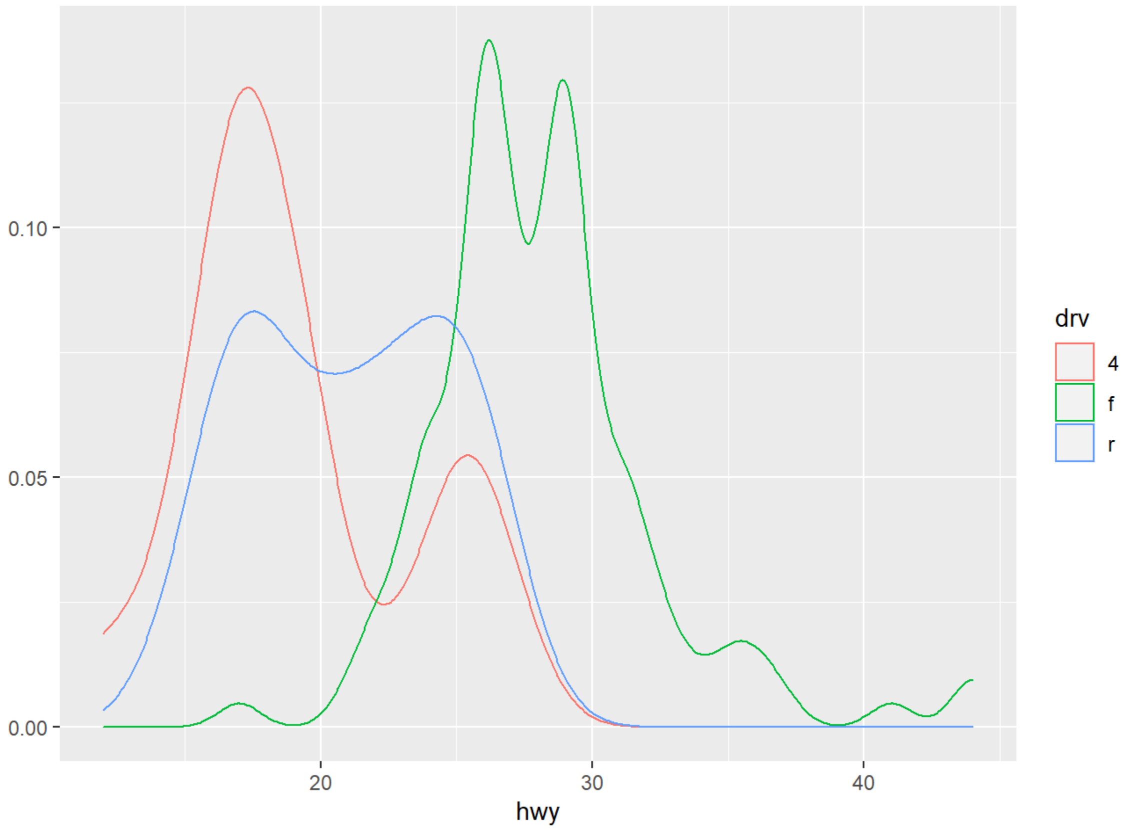

```
> qplot(displ, hwy, data = mpg, facets = .~drv)
```



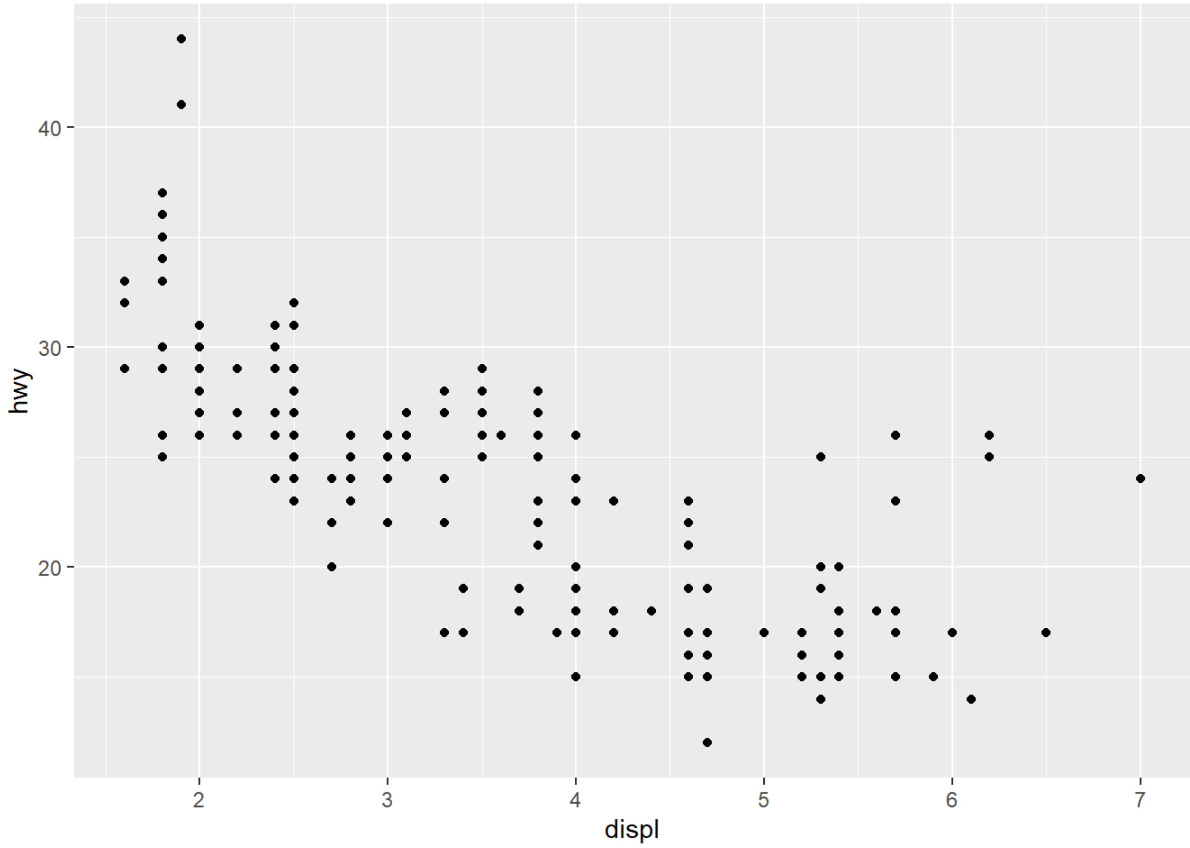
> qplot(hwy, data = mpg, geom = "density")



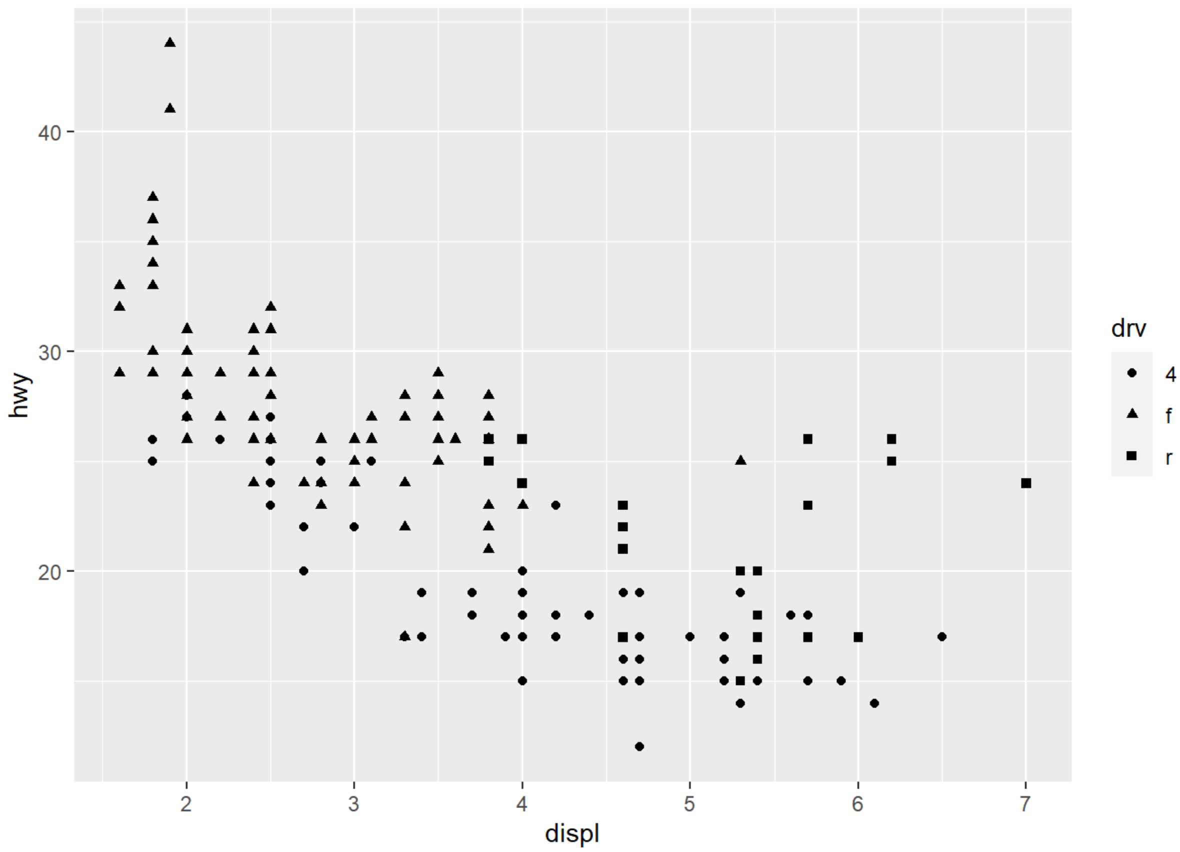
> qplot(hwy, data = mpg, geom = "density", color = drv)



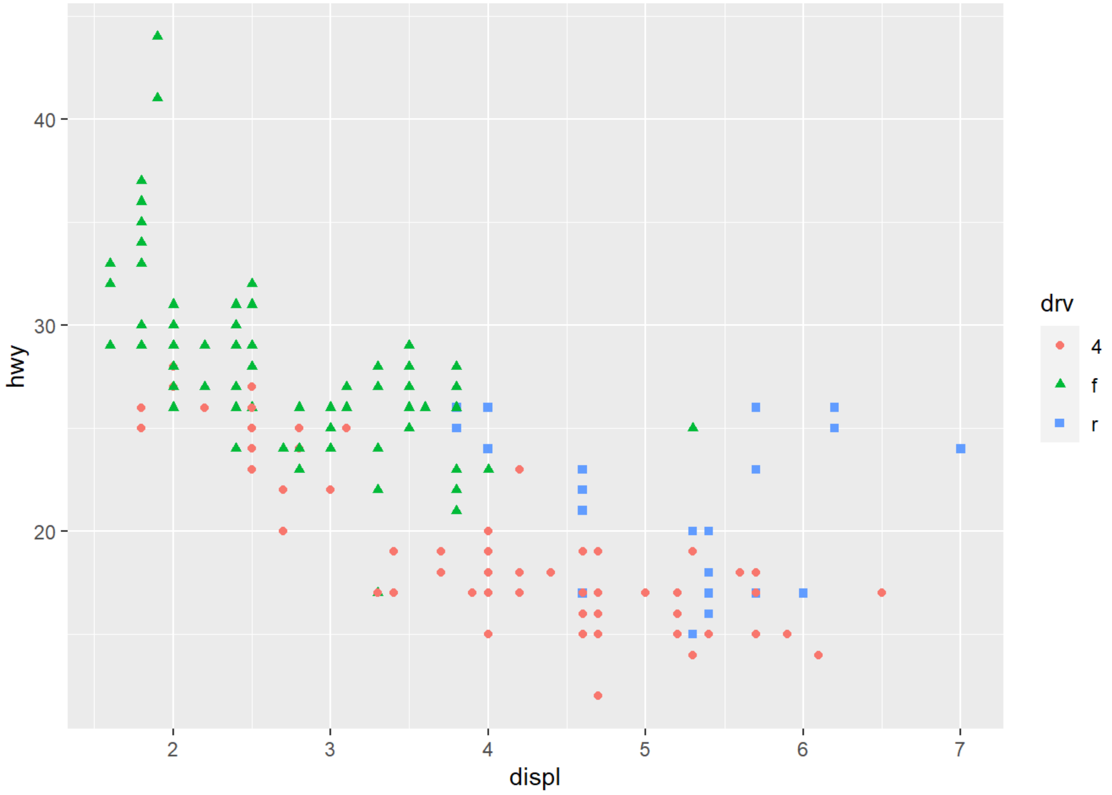
```
> qplot(displ, hwy, data = mpg)
```



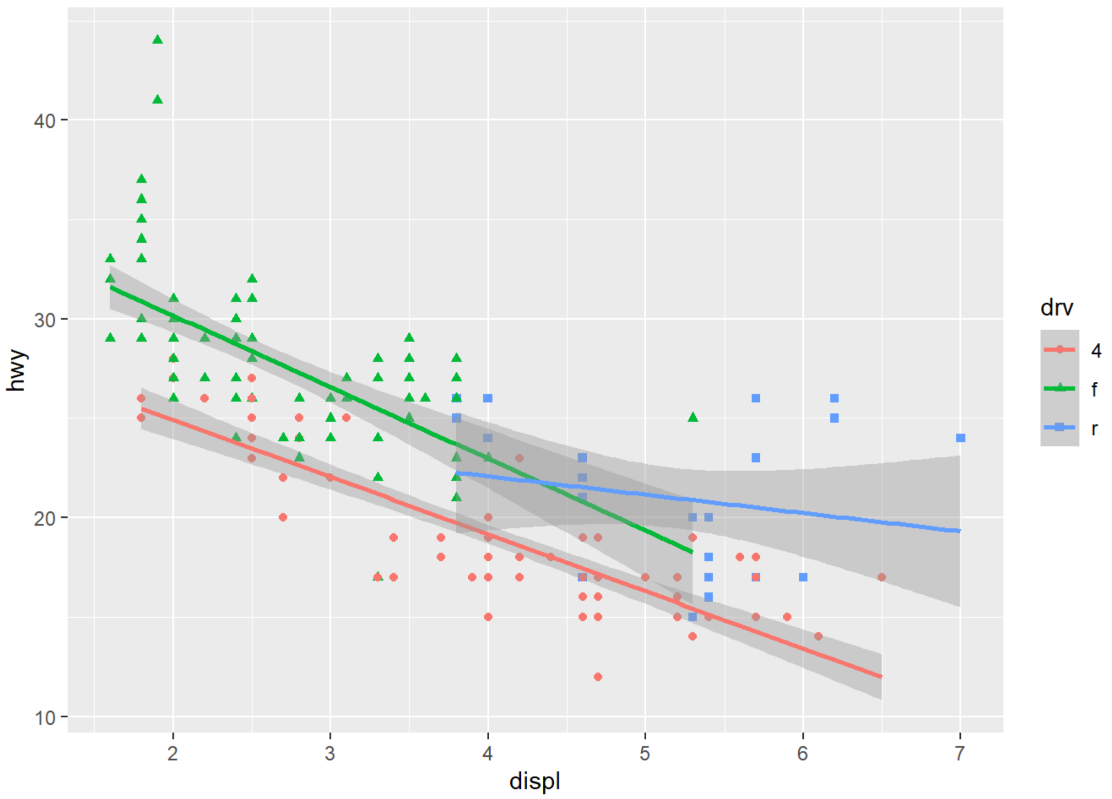

```
> qplot(displ, hwy, data = mpg, shape = drv)
```



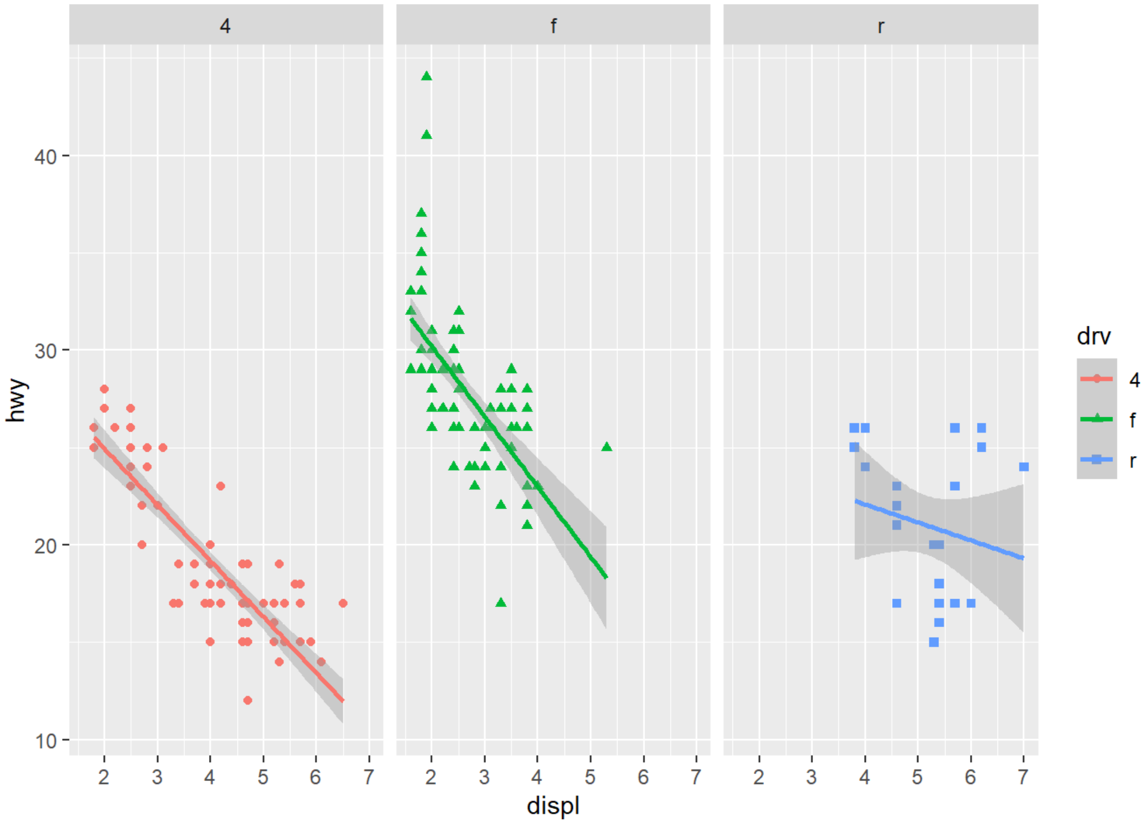
```
> qplot(displ, hwy, data = mpg, shape = drv, color = drv)
```




```
> qplot(displ, hwy, data = mpg, shape = drv, color = drv,
+       geom = c("point", "smooth"), method = "lm")
```

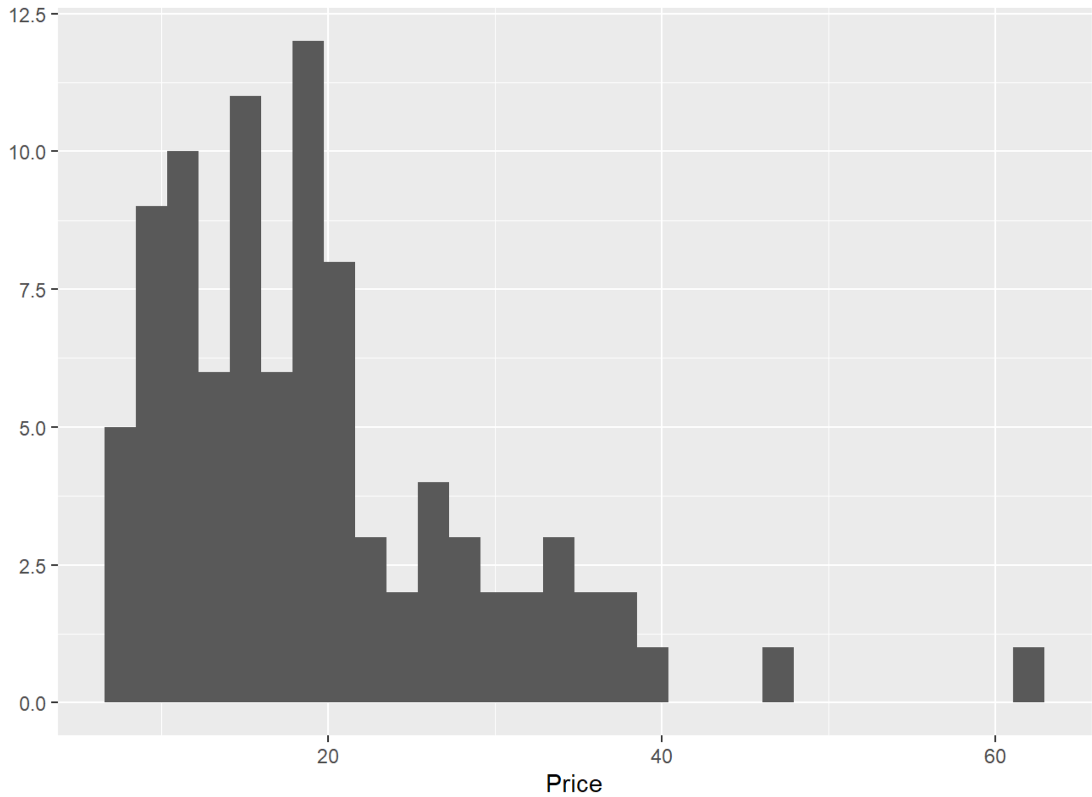


```
> qplot(displ, hwy, data = mpg, shape = drv, color = drv,
+       geom = c("point", "smooth"), method = "lm", facets = .~drv)
```

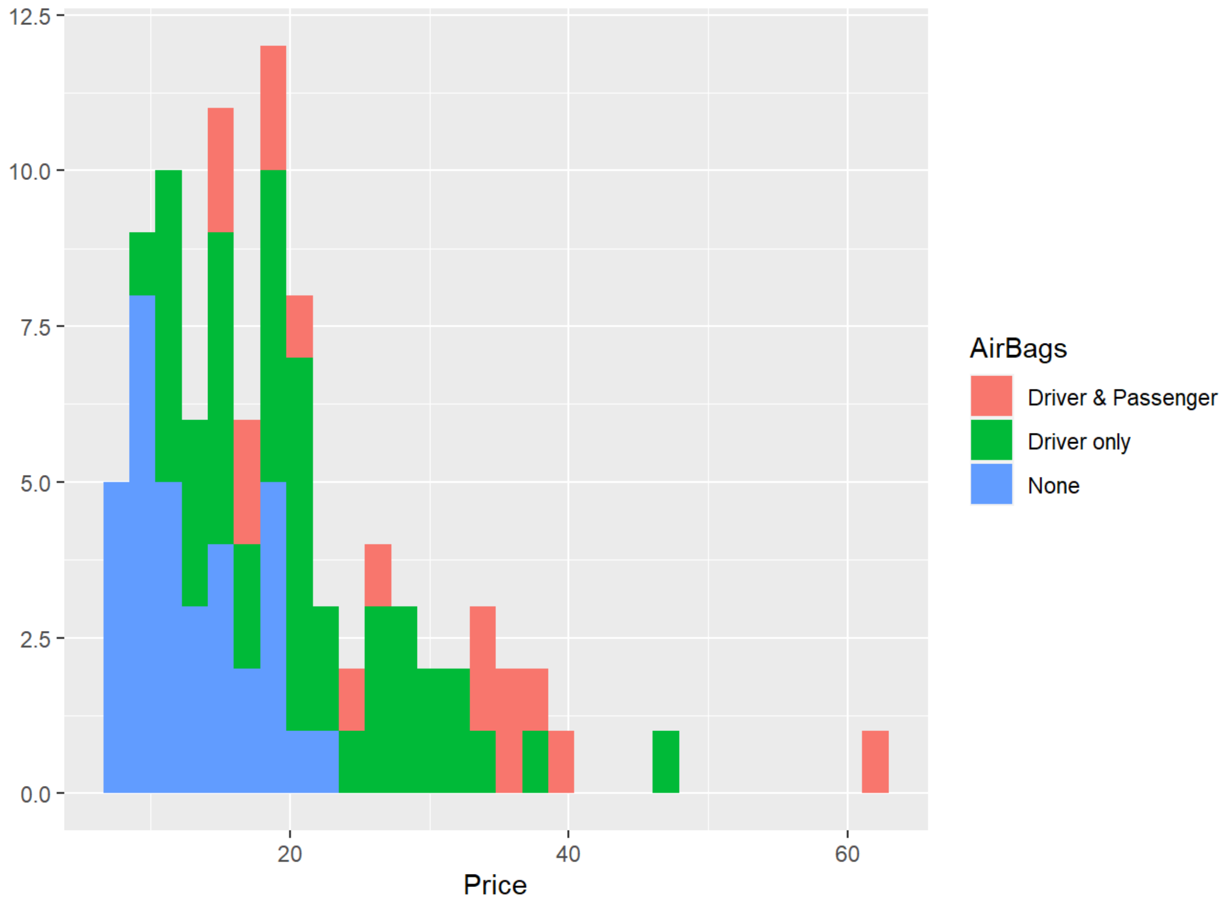


Using the Cars93 data frame

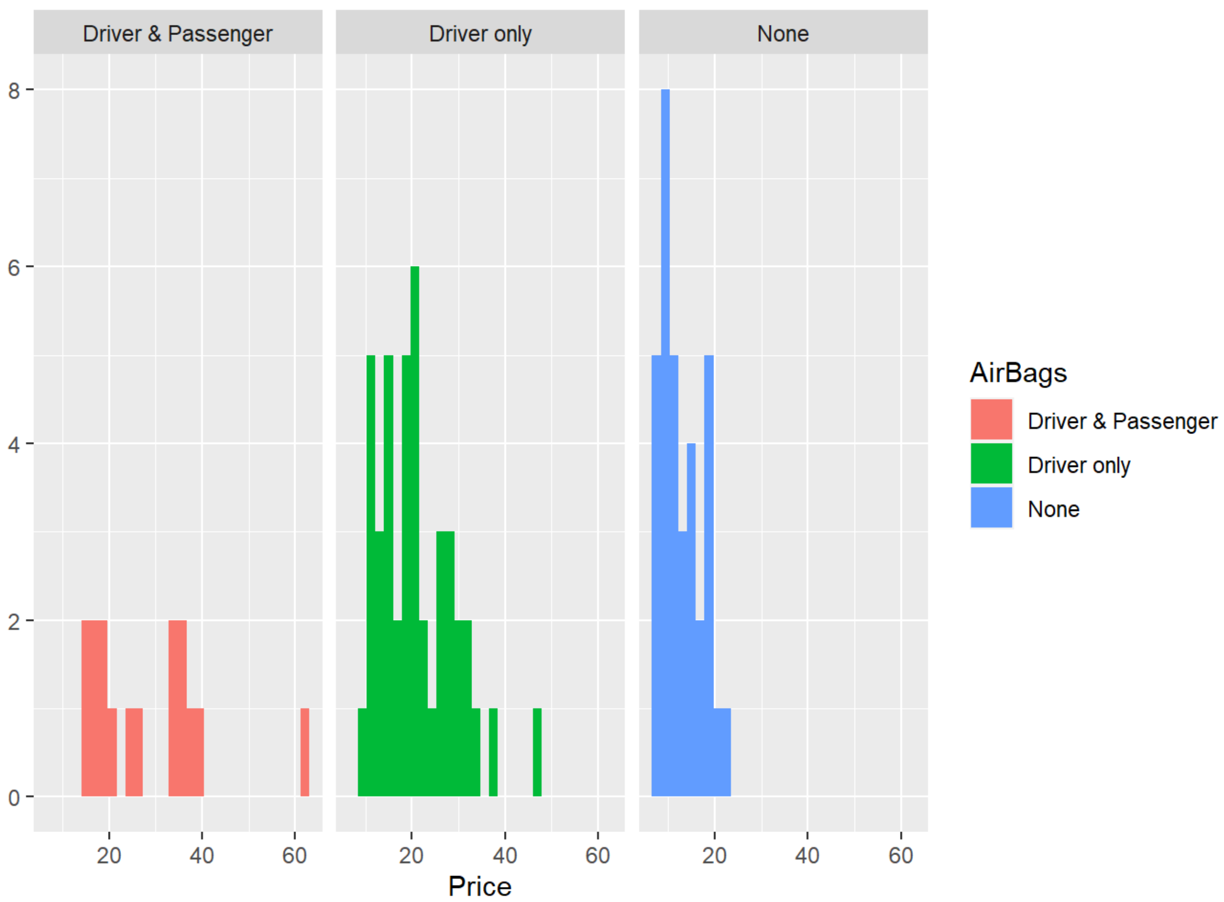
```
> qplot(Price, data = Cars93, geom= "histogram")
```



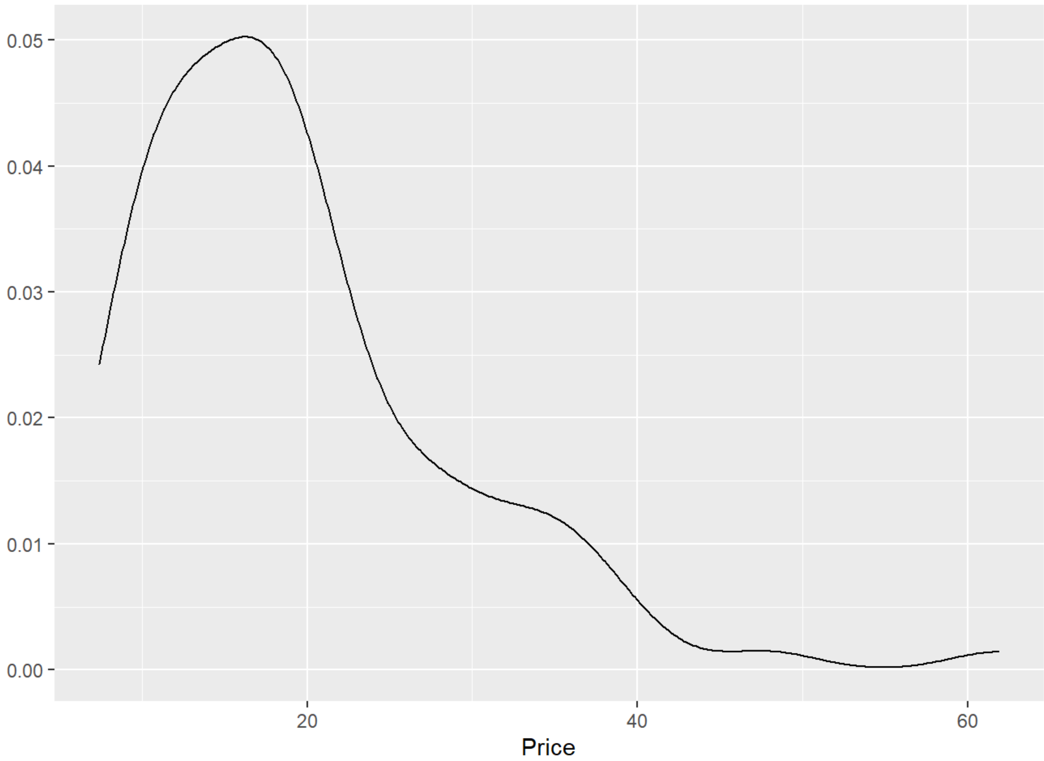
> qplot(Price, data = Cars93, geom= "histogram", fill = AirBags)



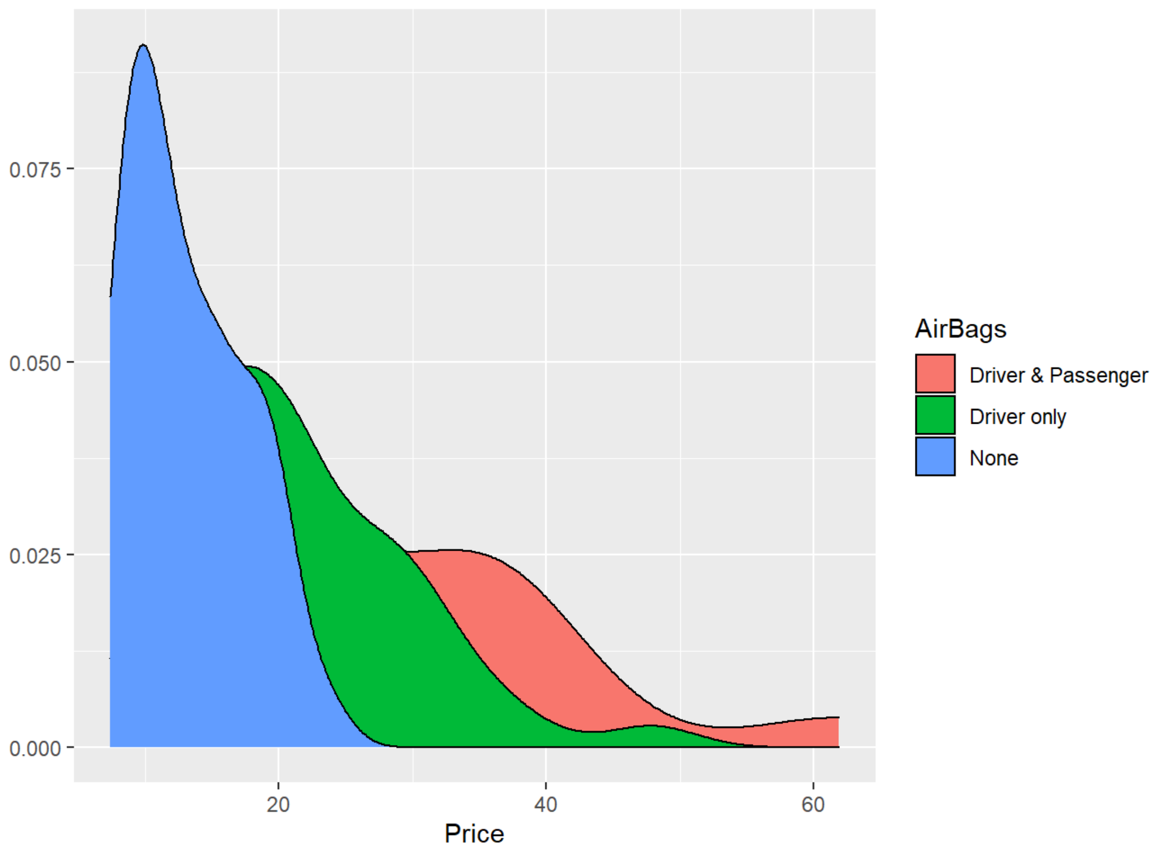
> qplot(Price, data = Cars93, geom= "histogram", fill = AirBags, facets = .~AirBags)



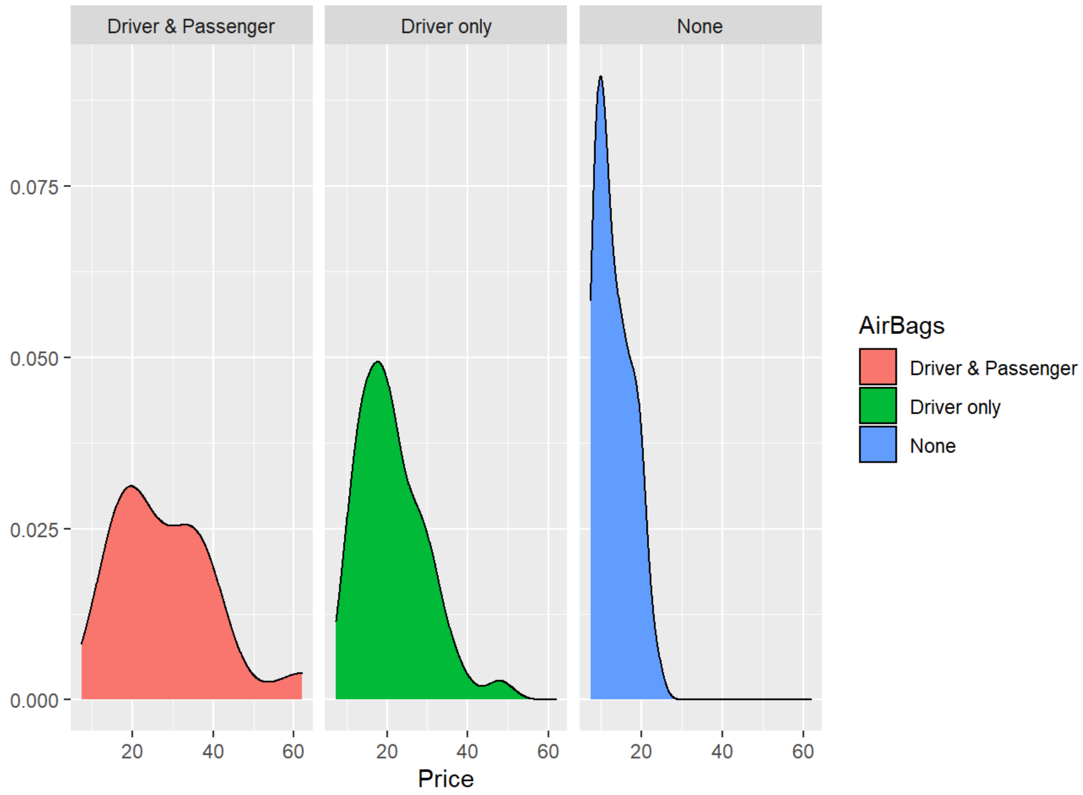
```
> qplot(Price, data = Cars93, geom= "density")
```




```
> qplot(Price, data = Cars93, geom= "density", fill = AirBags)
```

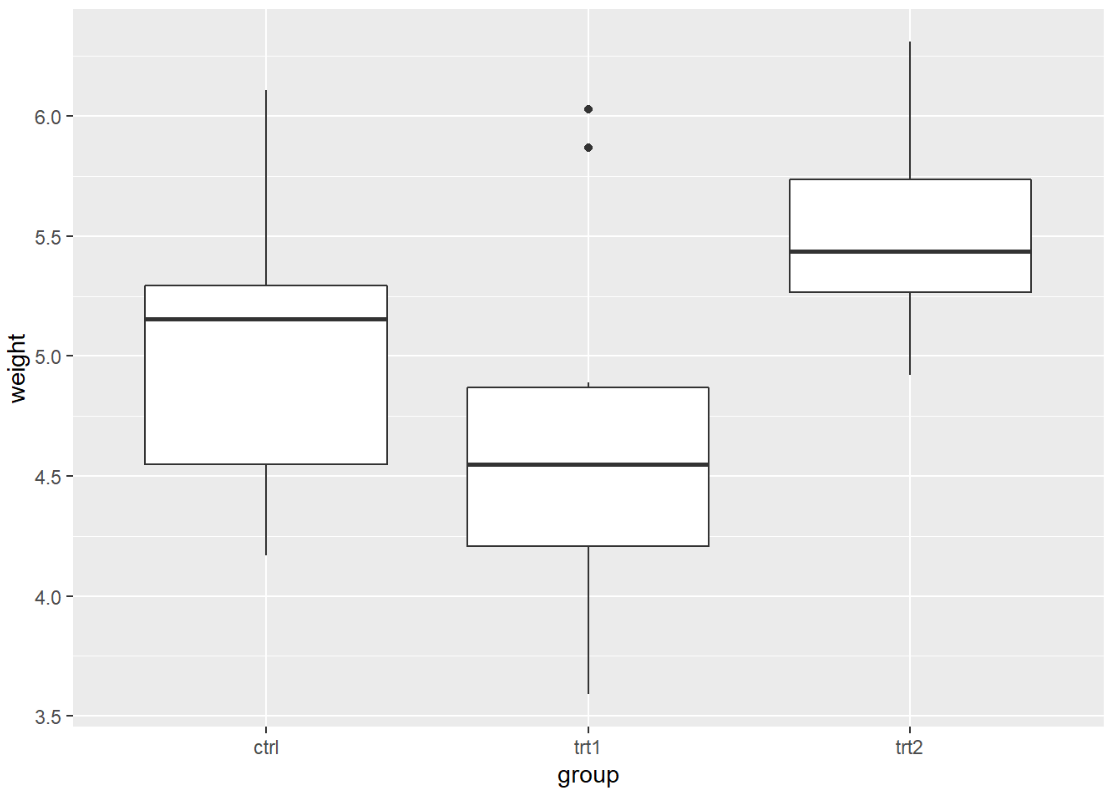


```
> qplot(Price, data = Cars93, geom= "density", fill = AirBags, facets = .~AirBags)
```

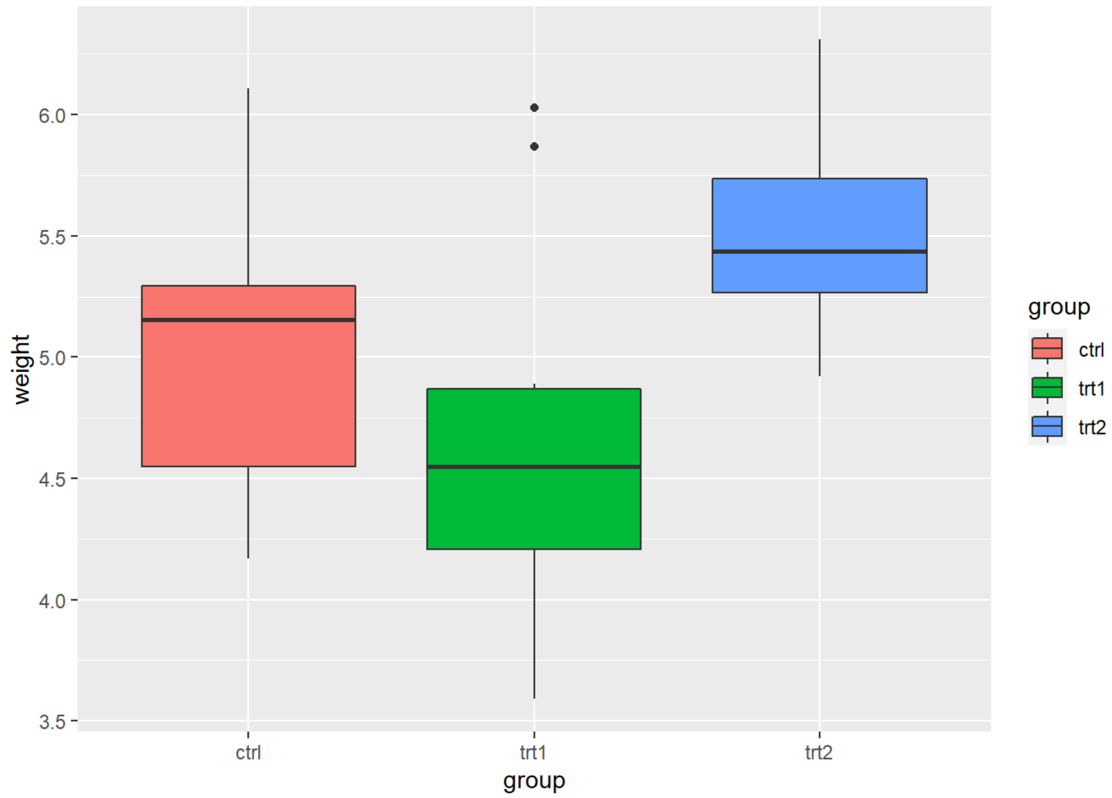


Boxplots from the PlanthGrowth example

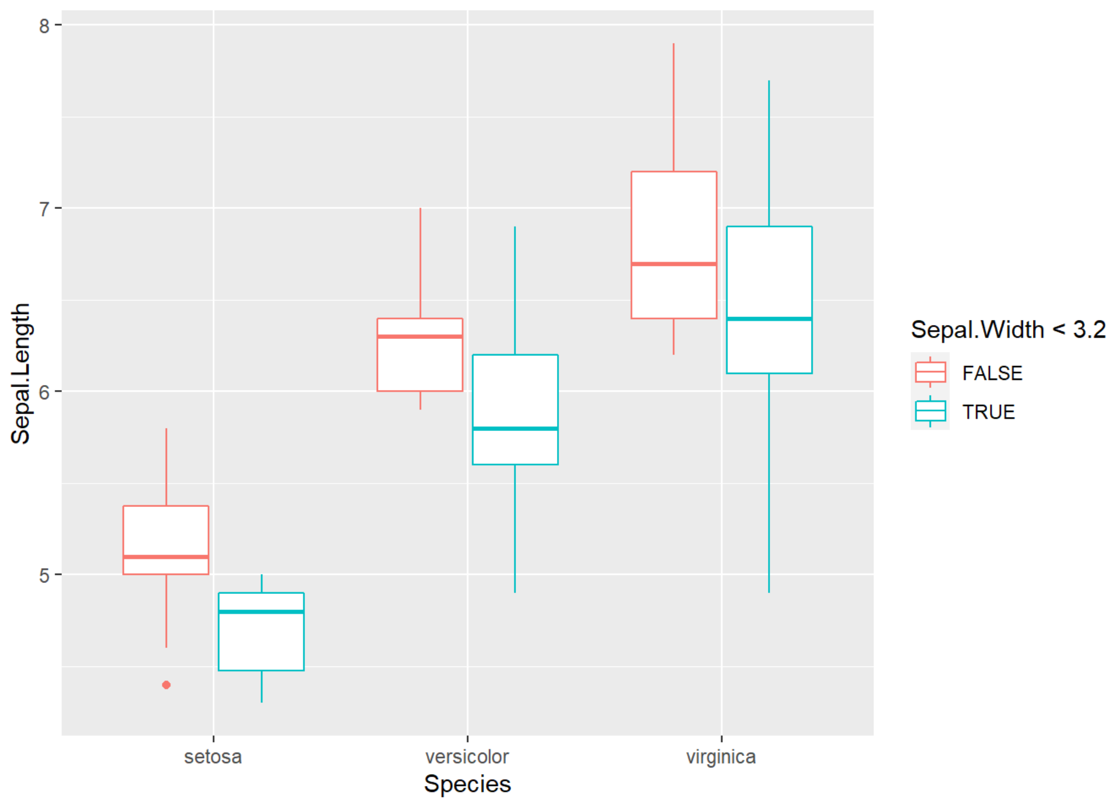
```
> qplot(group, weight, data = PlantGrowth, geom = "boxplot")
```



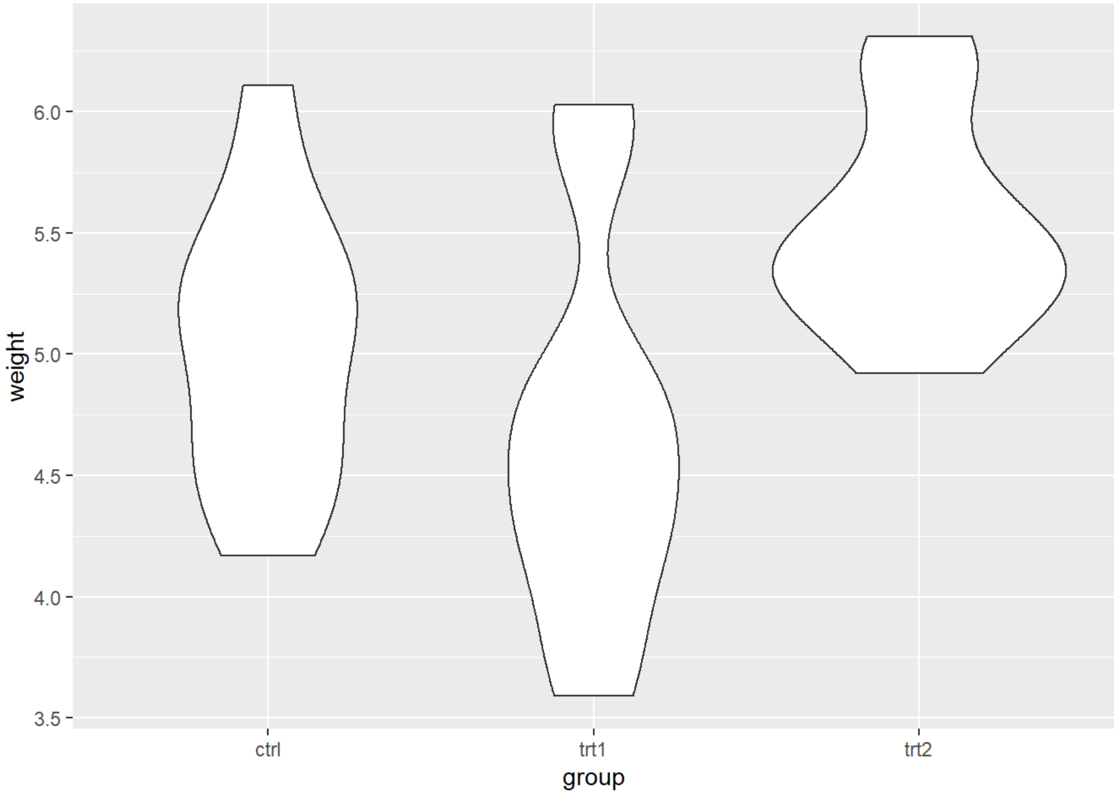
```
> qplot(group, weight, data = PlantGrowth, geom = "boxplot", fill = group)
```



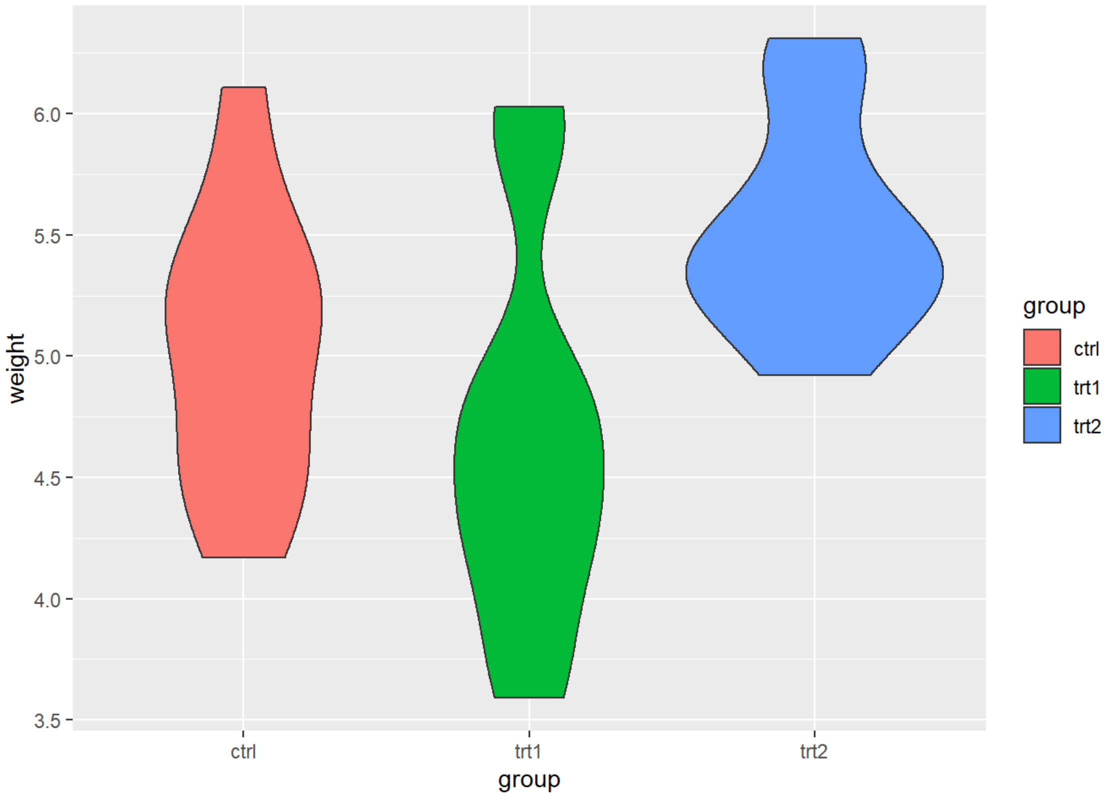

```
> ggplot(data = iris, aes(Species, Sepal.Length)) +
+   geom_boxplot(aes(colour = Sepal.Width < 3.2))
```



> qplot(group, weight, data = PlantGrowth, geom = "violin")



> qplot(group, weight, data = PlantGrowth, geom = "violin", fill = group)



Let’s review the prices of 50 000 diamonds from the dimonds data frame from ggplot2 package, where price is the price of diamond in US dollars and cut is the quality of cut (Fair, Good, Very Good, Premium, Ideal). Make a frequency polygon for the prices depending on the cut quality.

```
> ggplot(diamonds, aes(price, colour = cut)) + geom_freqpoly( )
```