The lattice package

- better defaults

- ability to display multivariate relationships.

Trellis graphs - graphs that display a variable or the relationship between variables, conditioned on one or more other variables.

install and load Lattice Package

install.packages("lattice")

library(lattice)

Summary of Plots Covered in this section

Single Continuos Variable: (dataset Chickwts)

*1. Histogram - hist() || Lattice equivalent histogram()

*2. Density plot - plot() || Lattice equivalent densityplot()

*3. Box-Whisker Plot - boxplot() || Lattice equivalent bwplot() and the same for violin plot

*4. Bar Chart - barplot() || Lattice equivalent barchart()

*5. Scatter Plot - plot() || Lattice equivalent xyplot() and splom() scatter plot matrix | cloud() for 3D Plot

*6. Dot Chart - dotchart() || Lattice equivalent dotplot()

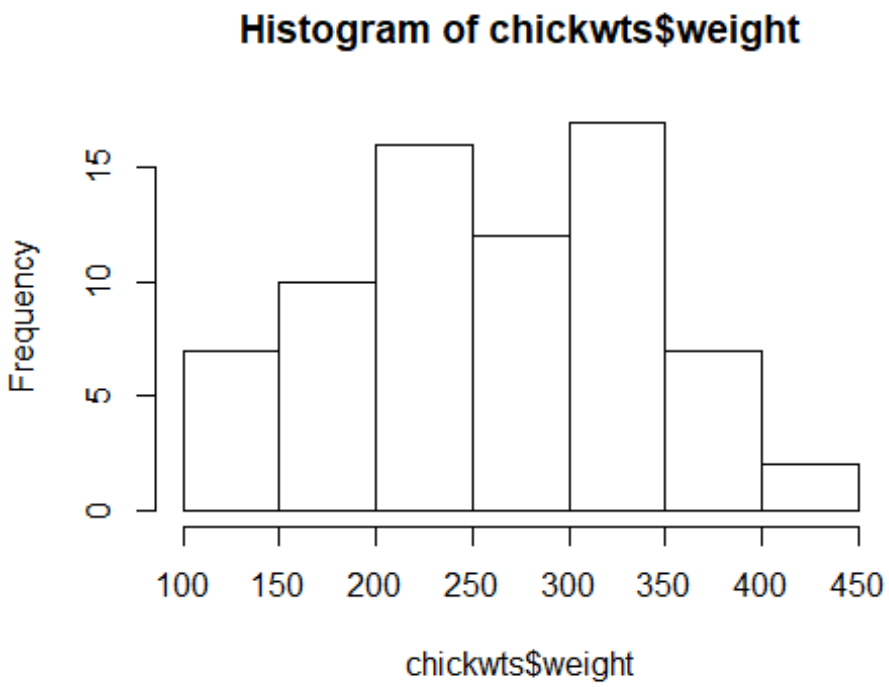
*7. Strip Chart - stripchart() || Lattice equivalent stripplot()

---

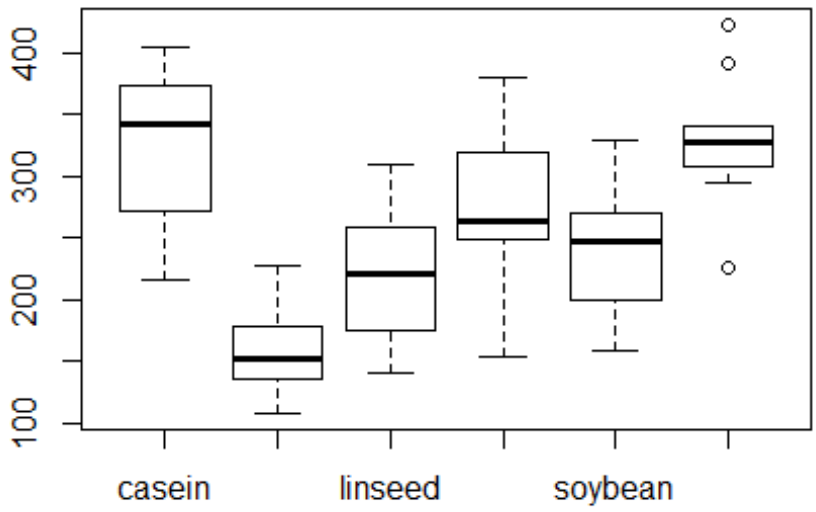
1. Histogram - hist() || Lattice equivalent histogram()

First example

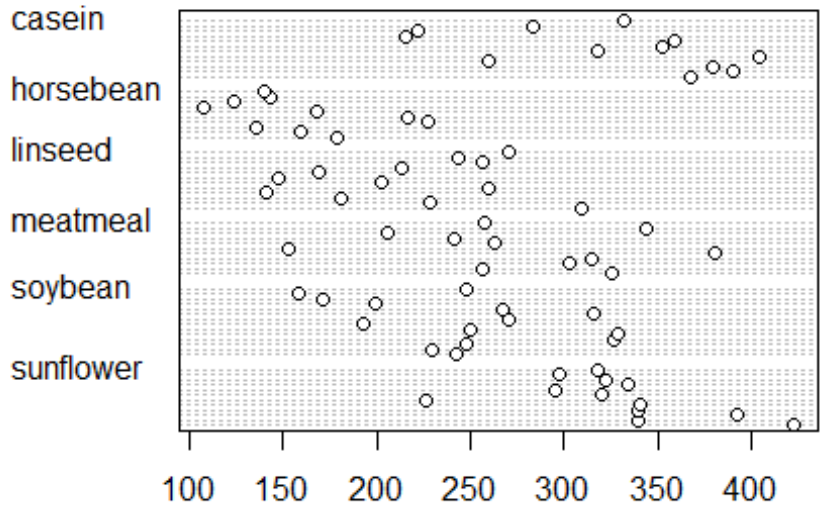
```
hist(chickwts$weight)
```



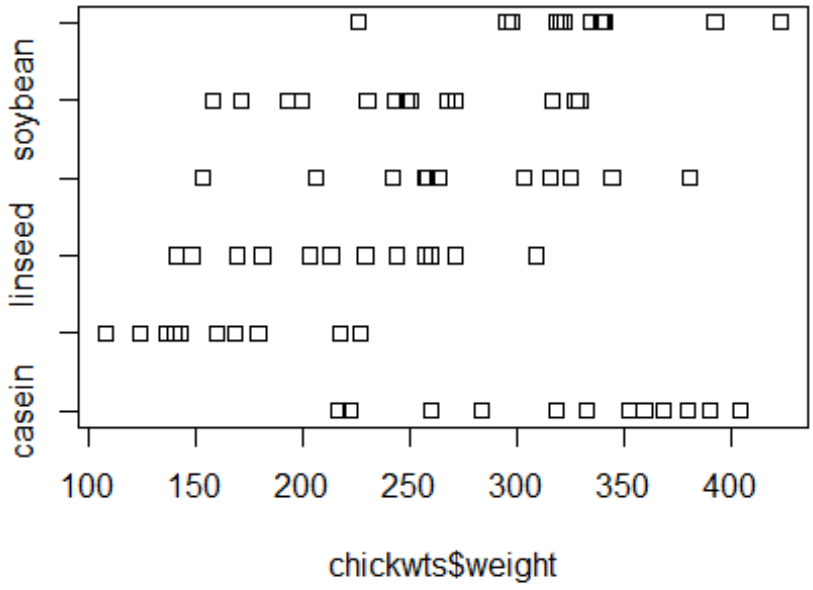

```
boxplot(chickwts$weight ~ chickwts$feed)
```




```
dotchart(chickwts$weight, groups = chickwts$feed)
```

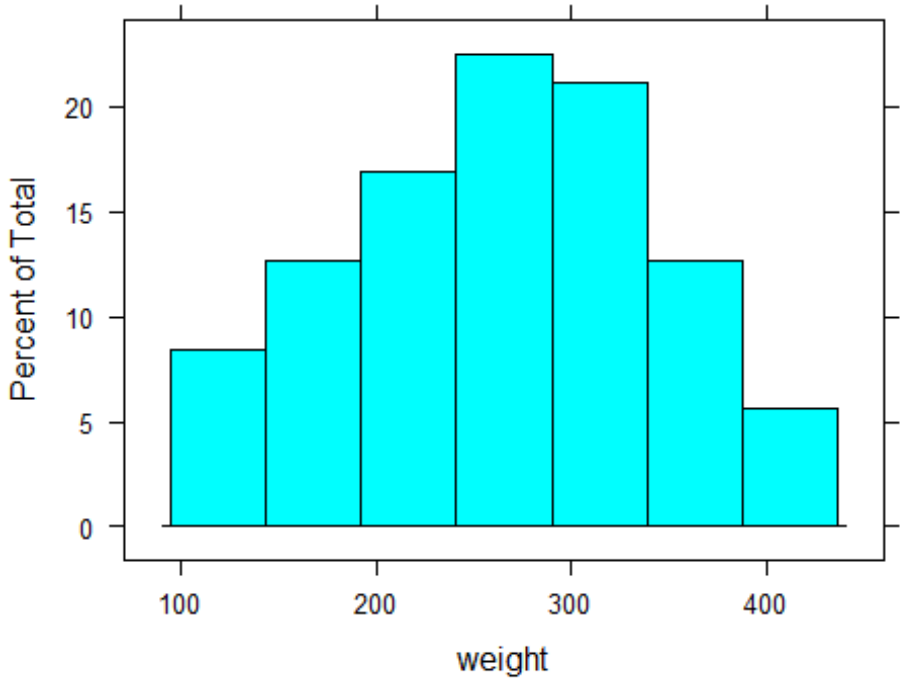



```
stripchart(chickwts$weight ~ chickwts$feed)
```

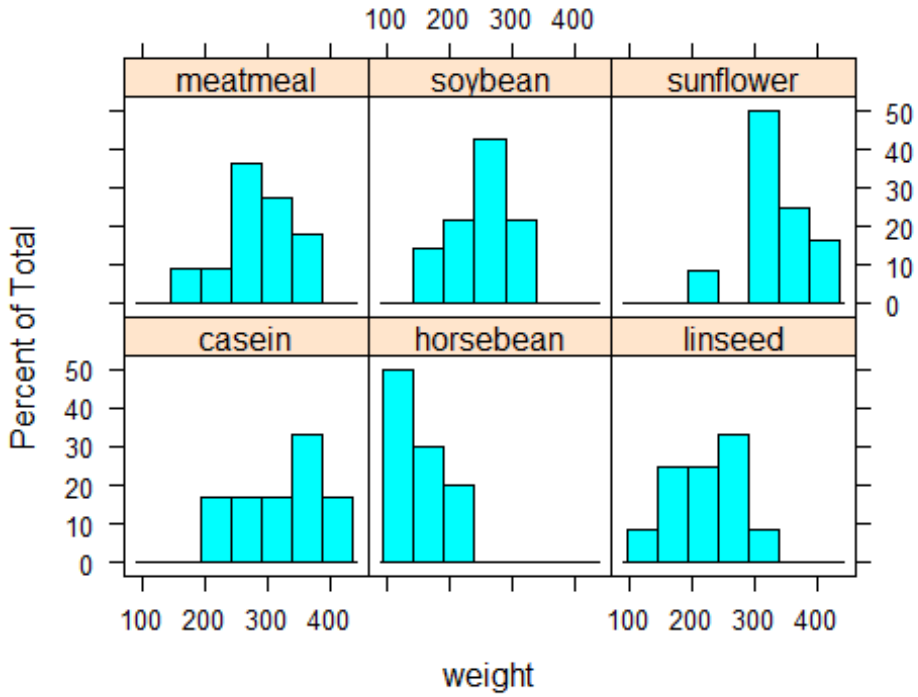


lattice

```
histogram(~weight, data=chickwts)
```



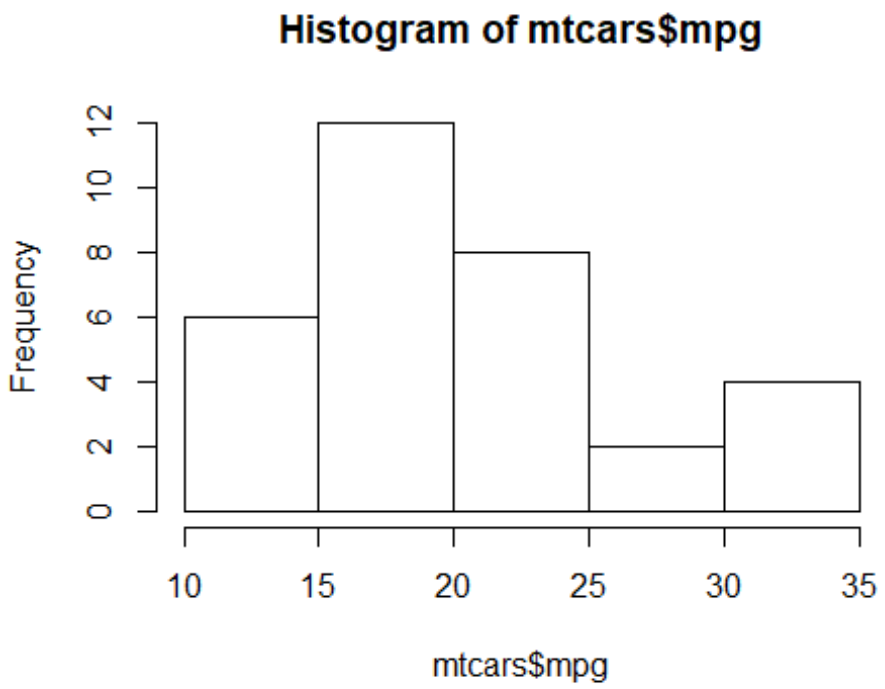

```
histogram(~weight | feed, data=chickwts)
```



Second Example

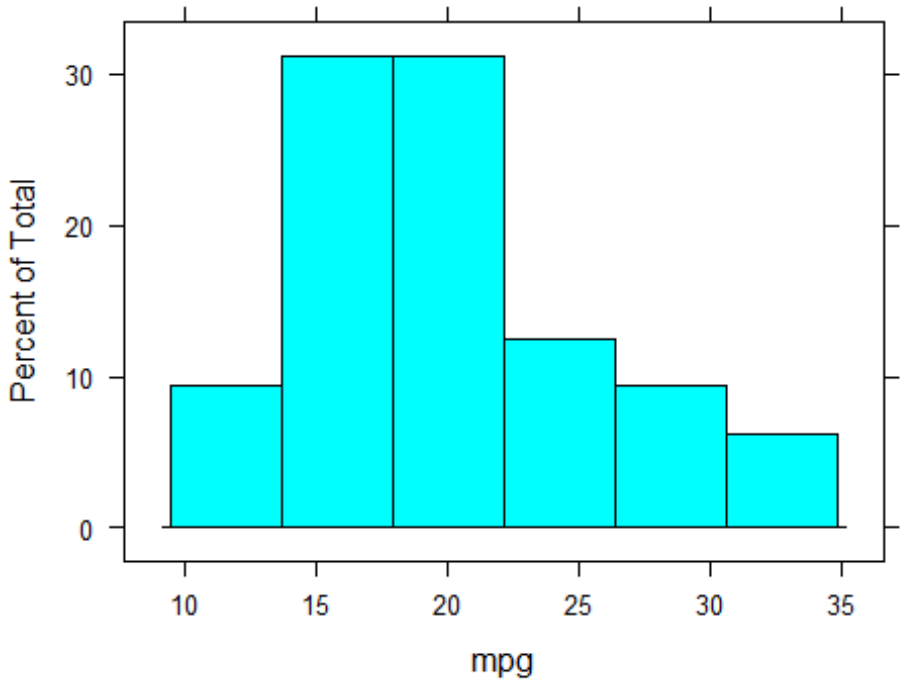
Base command

```
hist(mtcars$mpg)
```

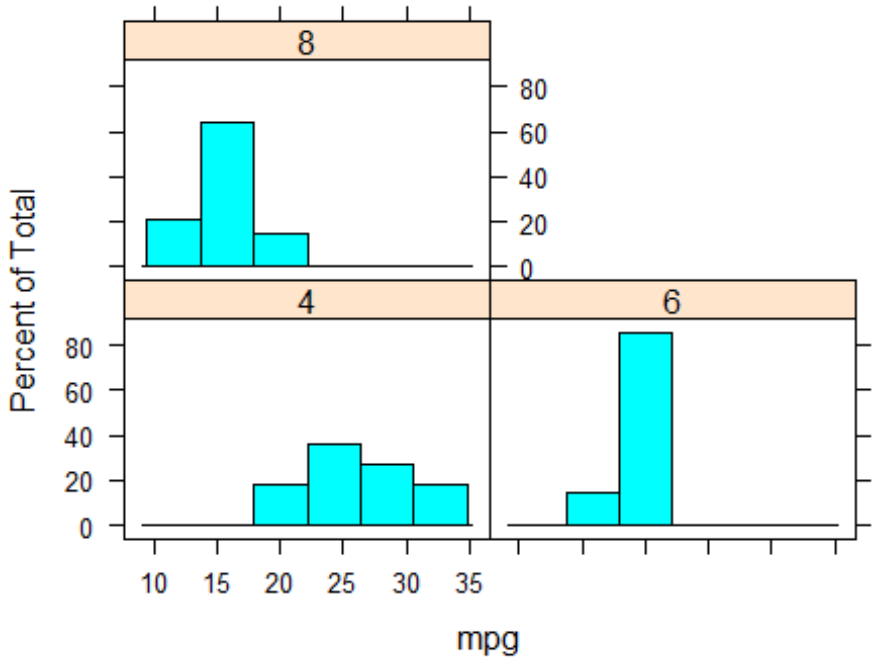


Lattice command

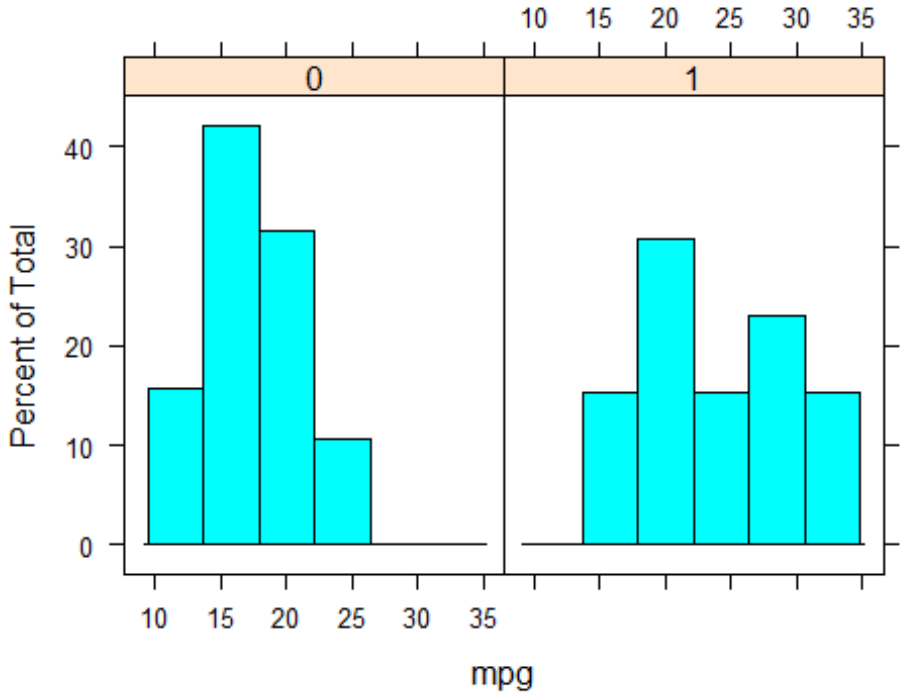
```
histogram(~mpg, data = mtcars)
```




```r
histogram(~mpg | factor(cyl), data = mtcars)
```

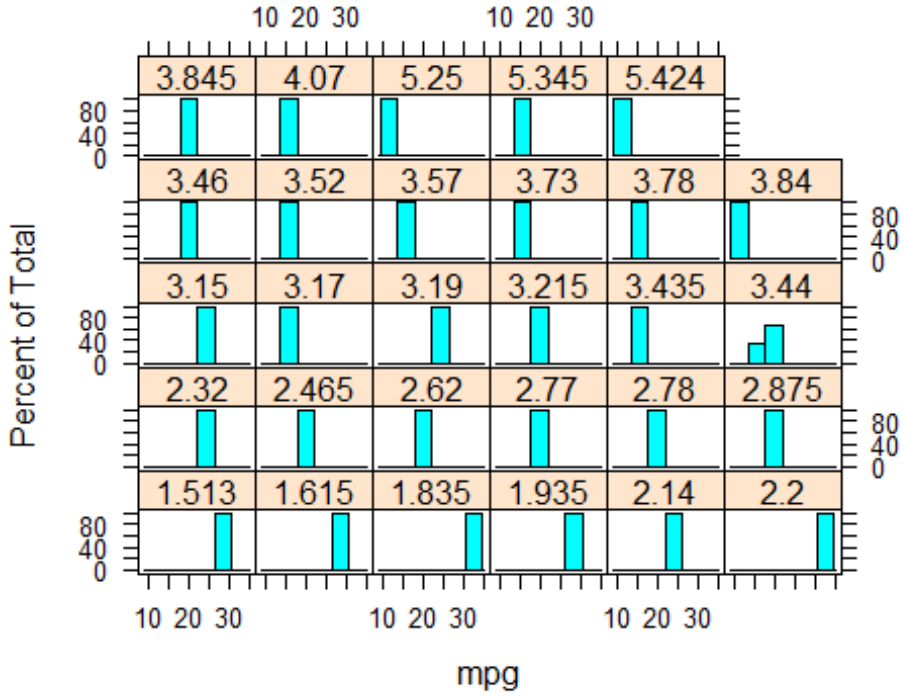



```r
histogram(~mpg | factor(am), data = mtcars)
```



If the factor is a continuous variable

```
histogram(~mpg | factor(wt), data = mtcars)
```

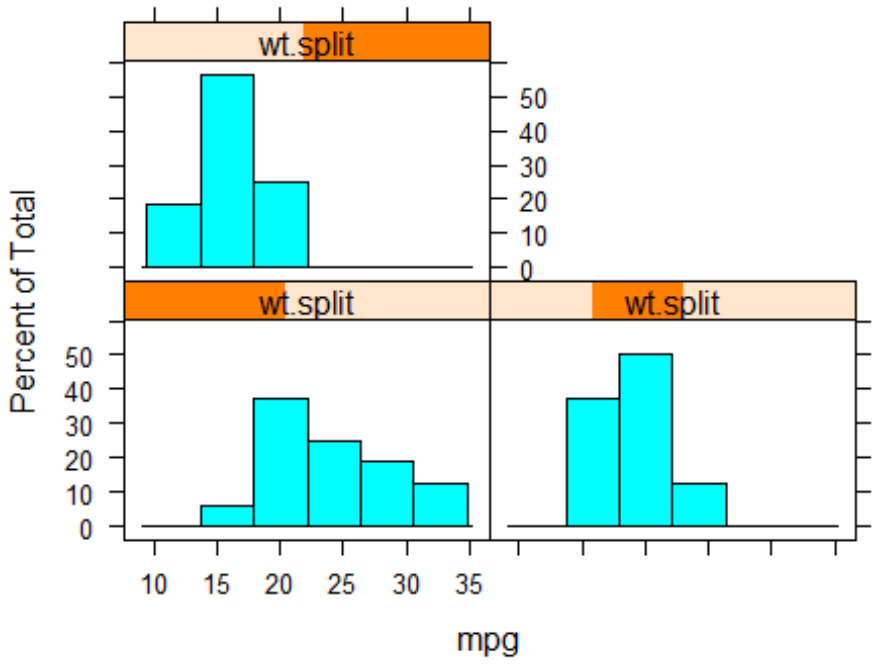


Lets split hp into three ranges

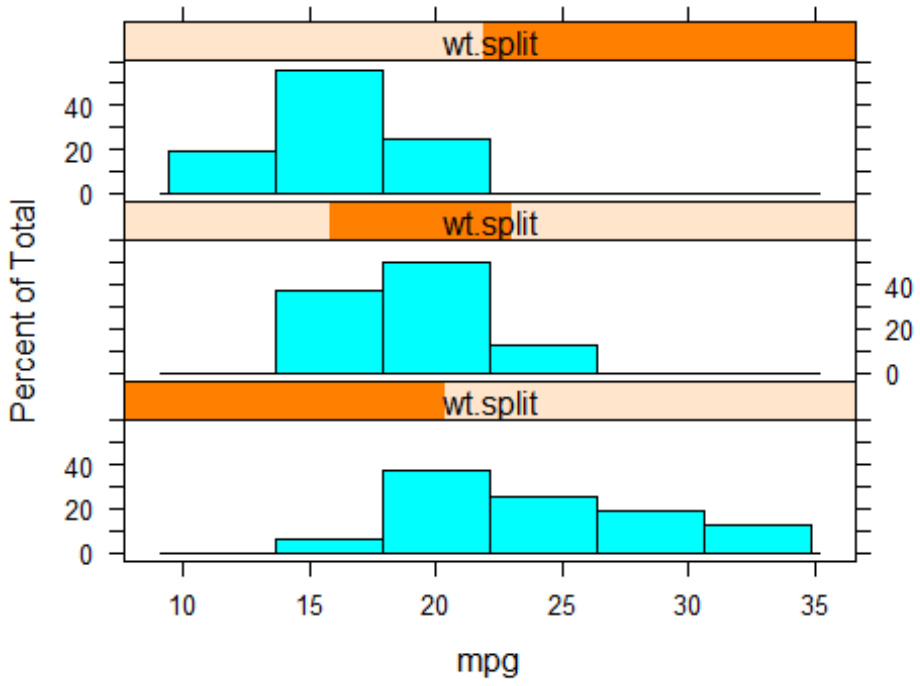
```
wt.split <- equal.count(mtcars$wt, 3)
wt.split
```

```
##
## Data:
##  [1] 2.620 2.875 2.320 3.215 3.440 3.460 3.570 3.190 3.150 3.440 3.440
## [12] 4.070 3.730 3.780 5.250 5.424 5.345 2.200 1.615 1.835 2.465 3.520
## [23] 3.435 3.840 3.845 1.935 2.140 1.513 3.170 2.770 3.570 2.780
##
## Intervals:
##      min    max count
## 1 1.5105 3.2175    16
## 2 2.6175 3.5725    16
## 3 3.4325 5.4265    16
##
## Overlap between adjacent intervals:
## [1] 8 8
```

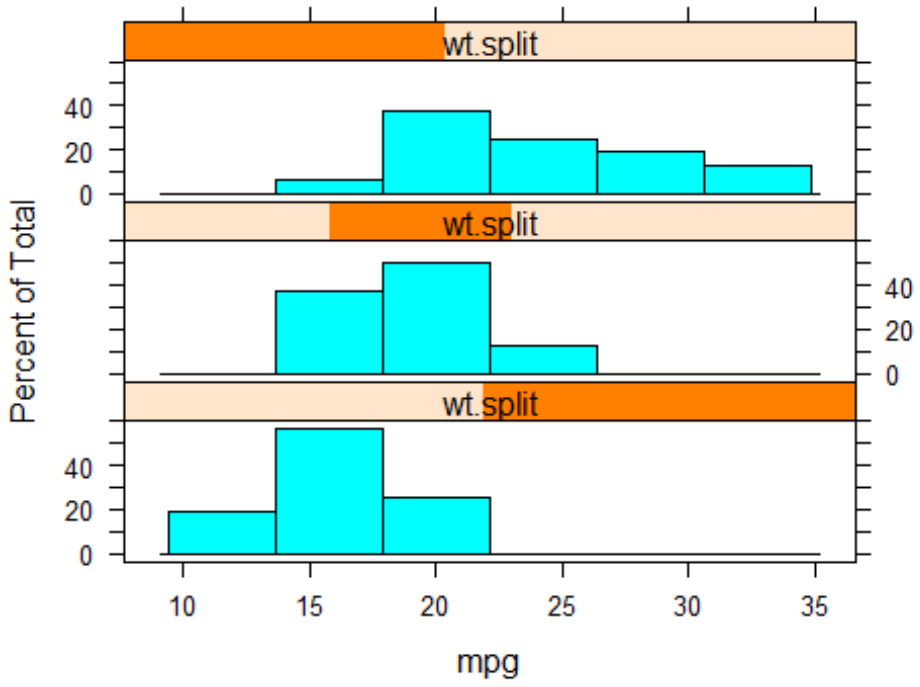
```
histogram(~mpg | wt.split, data = mtcars)
```



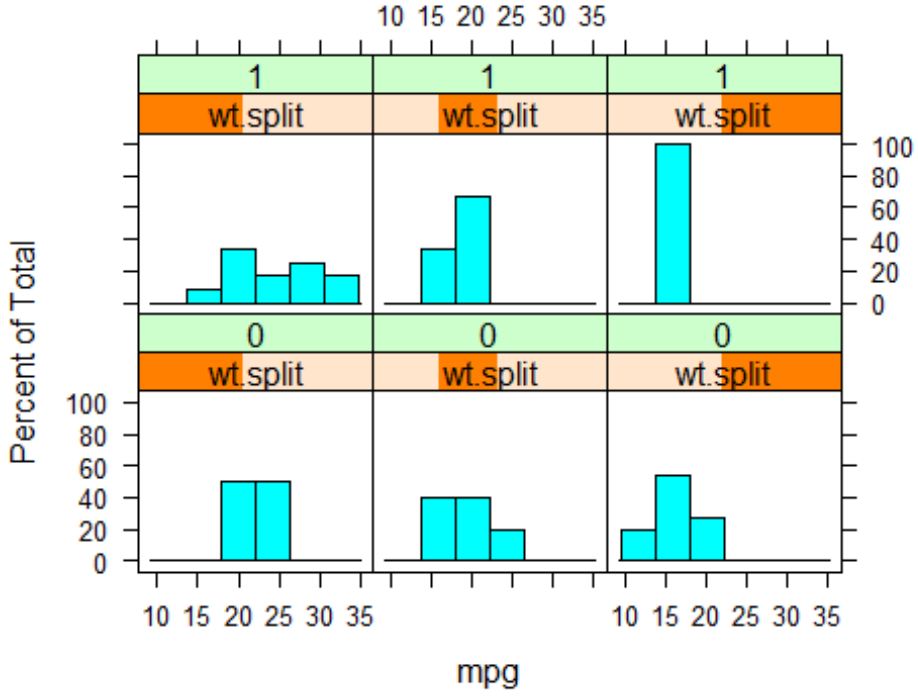

```
histogram(~mpg | wt.split, data = mtcars, layout = c(1,3))
```




```r
histogram(~mpg | wt.split, data = mtcars, layout = c(1,3), as.table=TRUE)
```

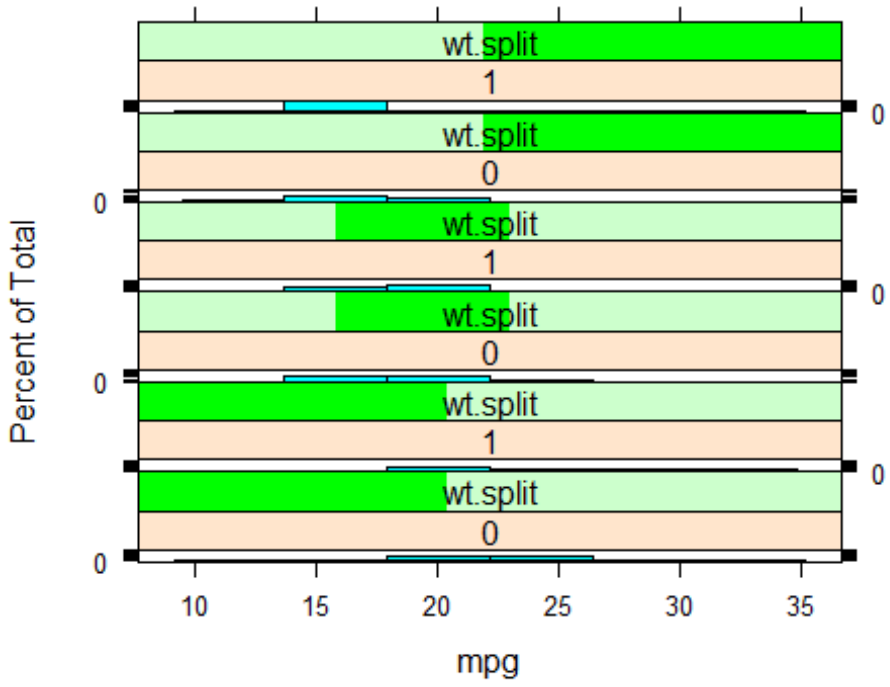



```r
histogram(~mpg | wt.split + factor(am), data = mtcars)
```



0 = automatic, 1 = manual

```
histogram(~mpg | factor(am) + wt.split , data = mtcars, layout = c(1,6))
```

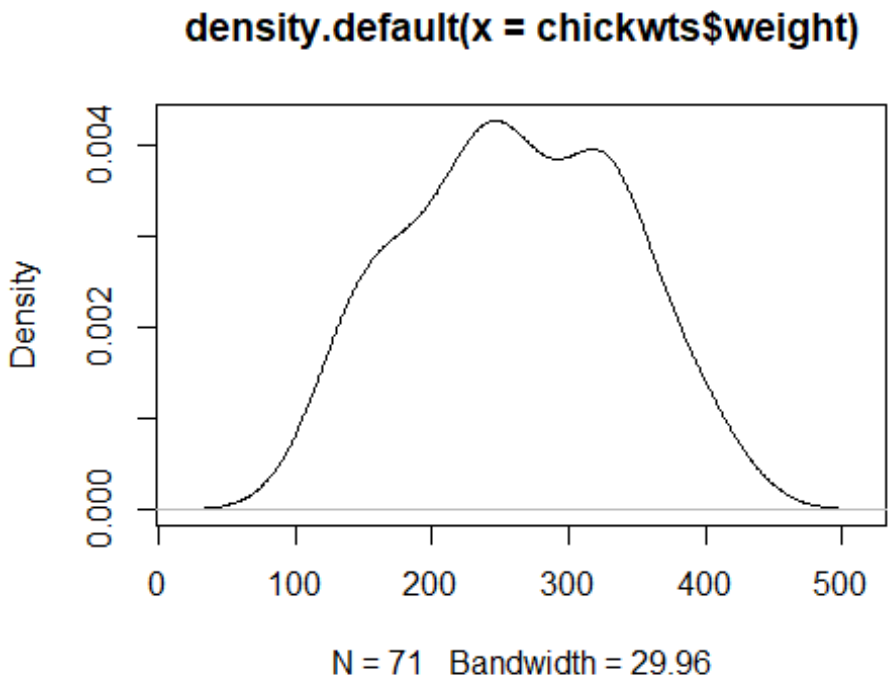


---

2. Density plot - plot() || Lattice equivalent densityplot()

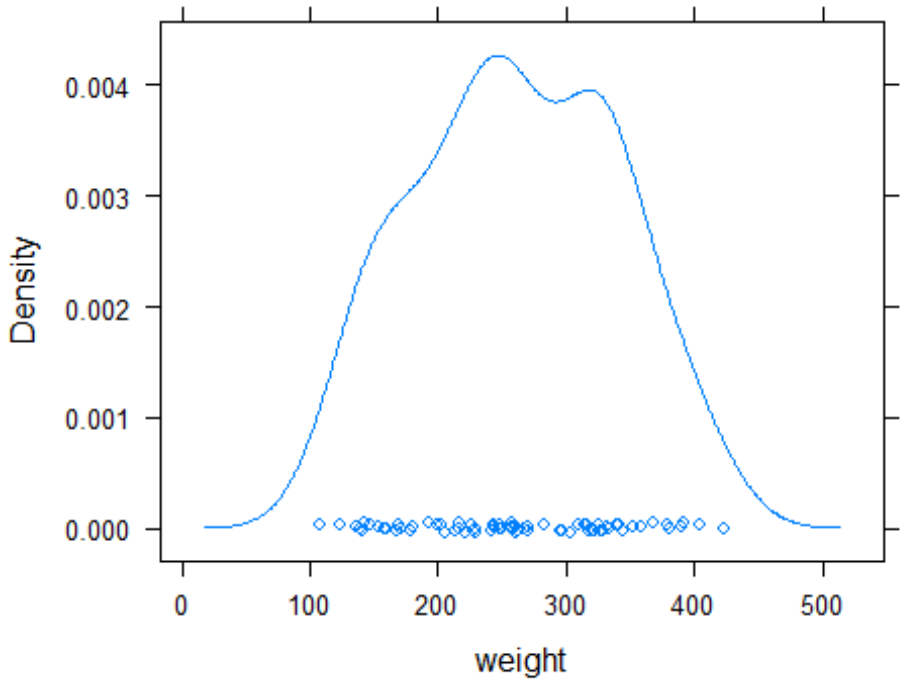
Base package commands

```
chden <- density(chickwts$weight)
plot(density(chickwts$weight))
```

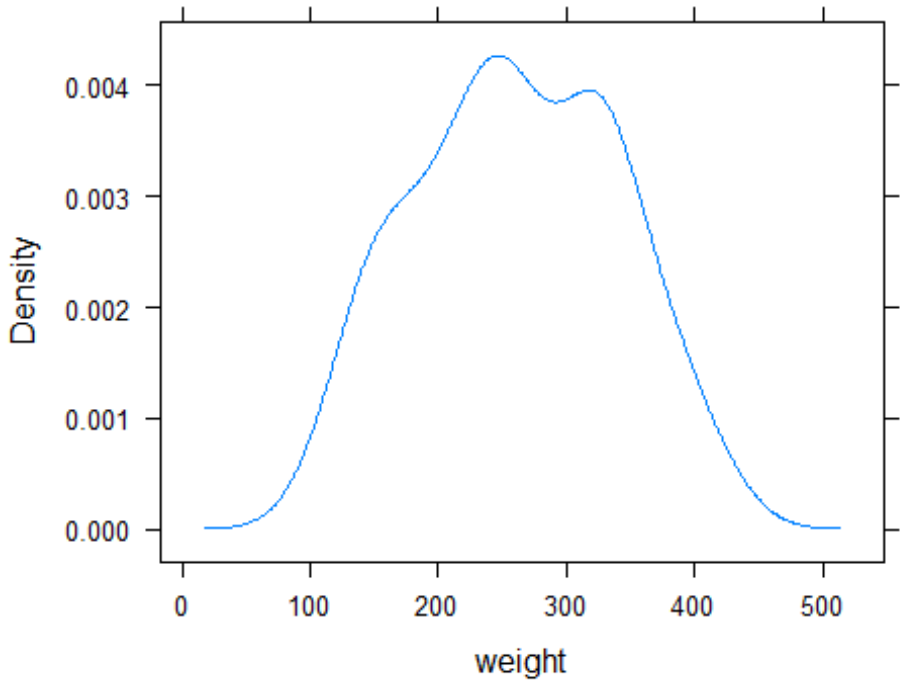


Lattice

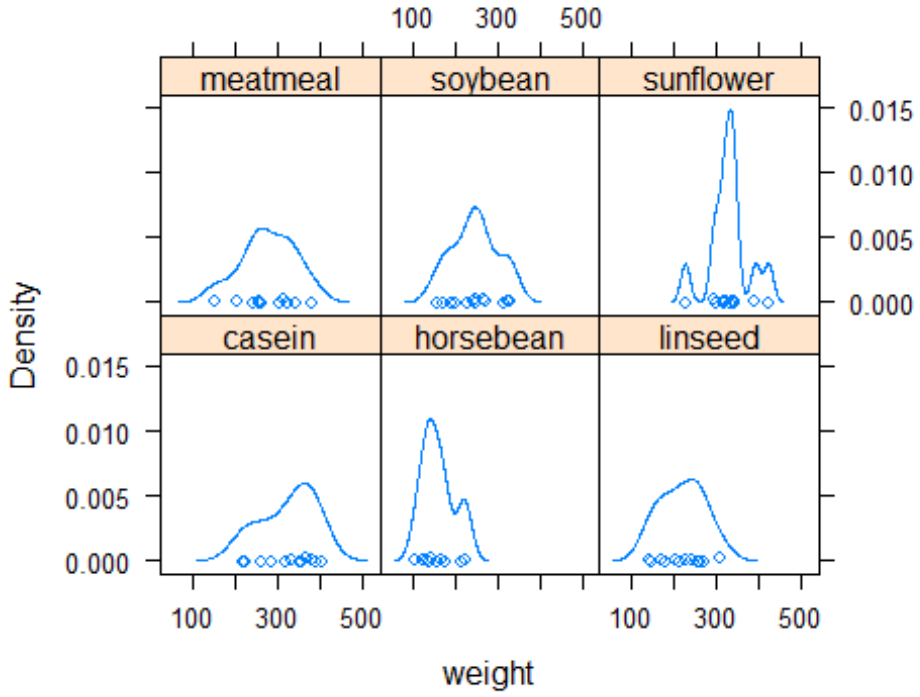
```
densityplot(~weight, data=chickwts)
```




```
densityplot(~weight, data=chickwts, plot.points = FALSE)
```

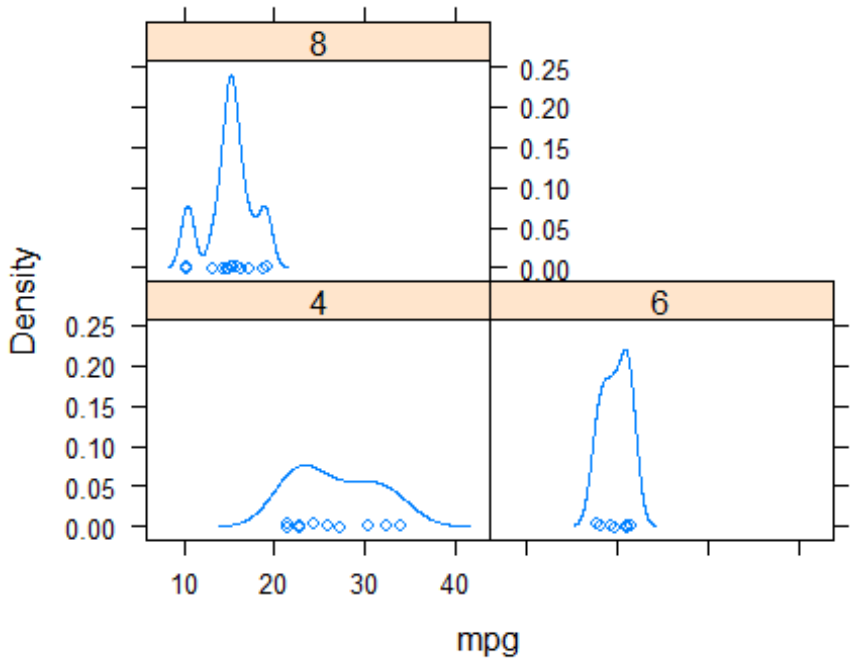



```
densityplot(~weight | feed, data=chickwts)
```

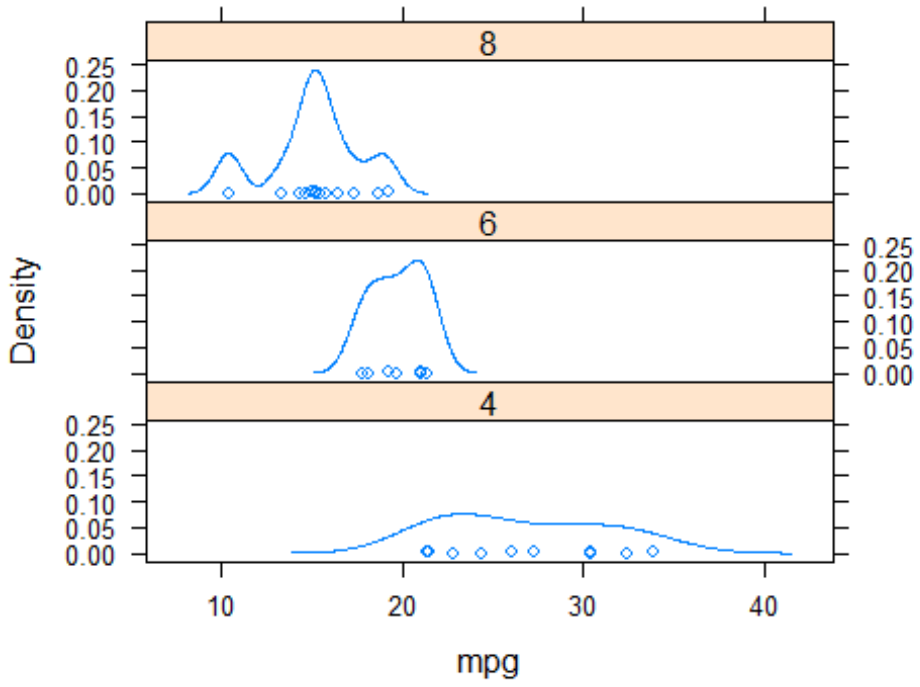


mtcars

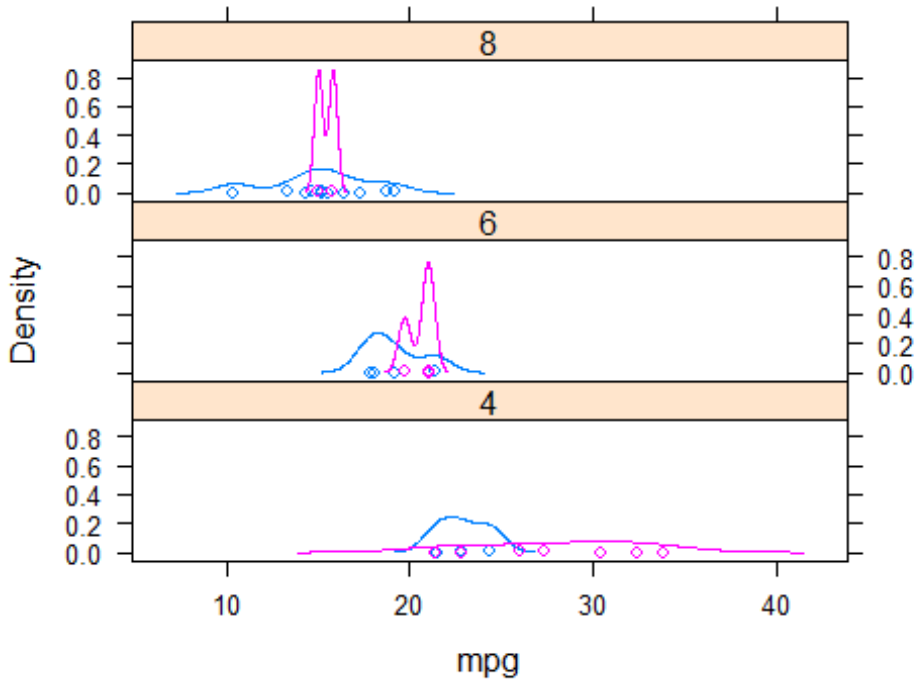
```
densityplot(~mpg | factor(cyl), data = mtcars)
```



```
densityplot(~mpg | factor(cyl), data = mtcars, layout = c(1,3))
```

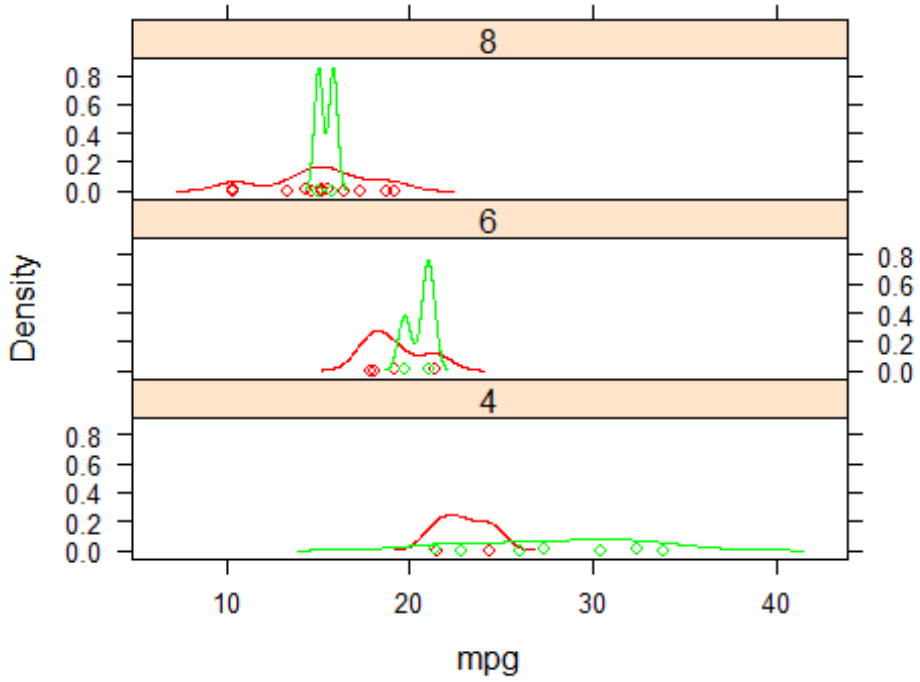



```
densityplot(~mpg | factor(cyl), data = mtcars, groups = factor(am),
            layout = c(1,3))
```



change color

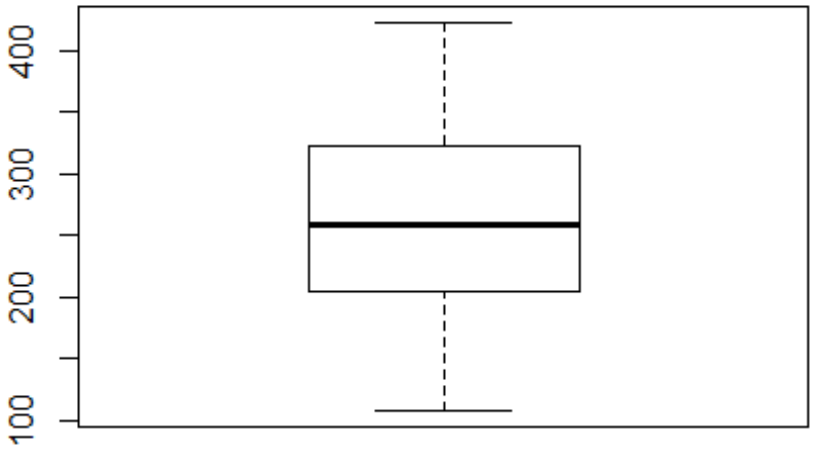
```
densityplot(~mpg | factor(cyl), data = mtcars, groups = factor(am),
            layout = c(1,3), col= c("red", "green"))
```



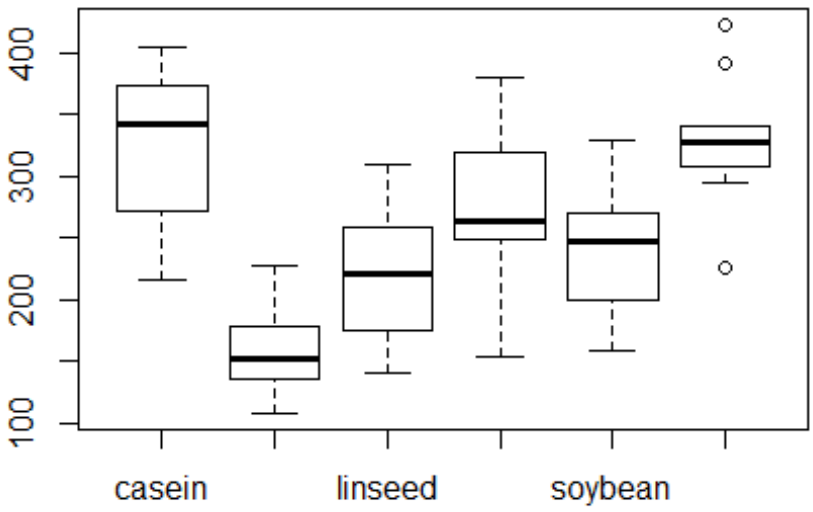
---

3. Box-Whisker Plot - boxplot() || Lattice equivalent bwplot()

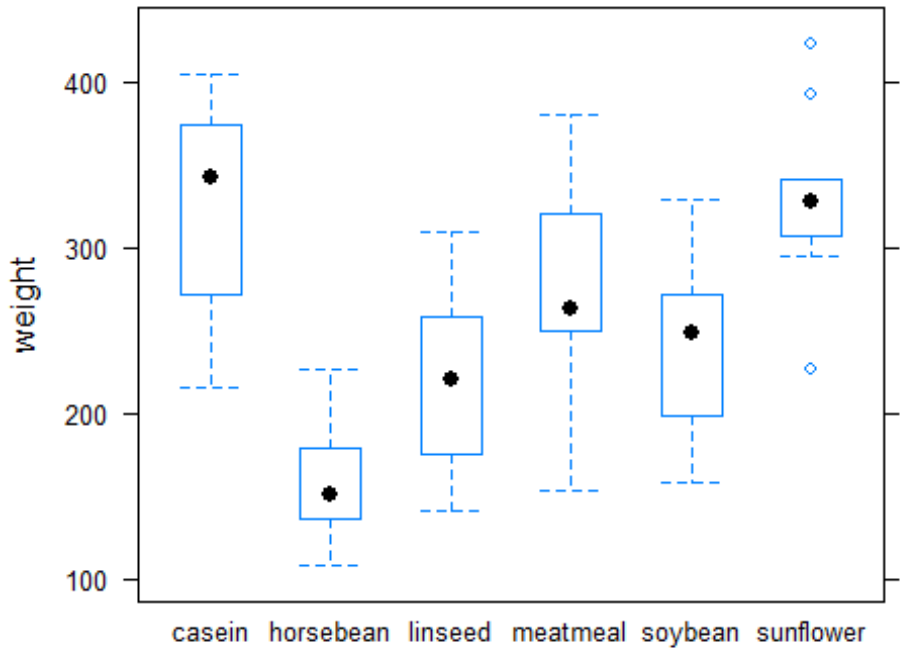
```
boxplot(chickwts$weight)
```



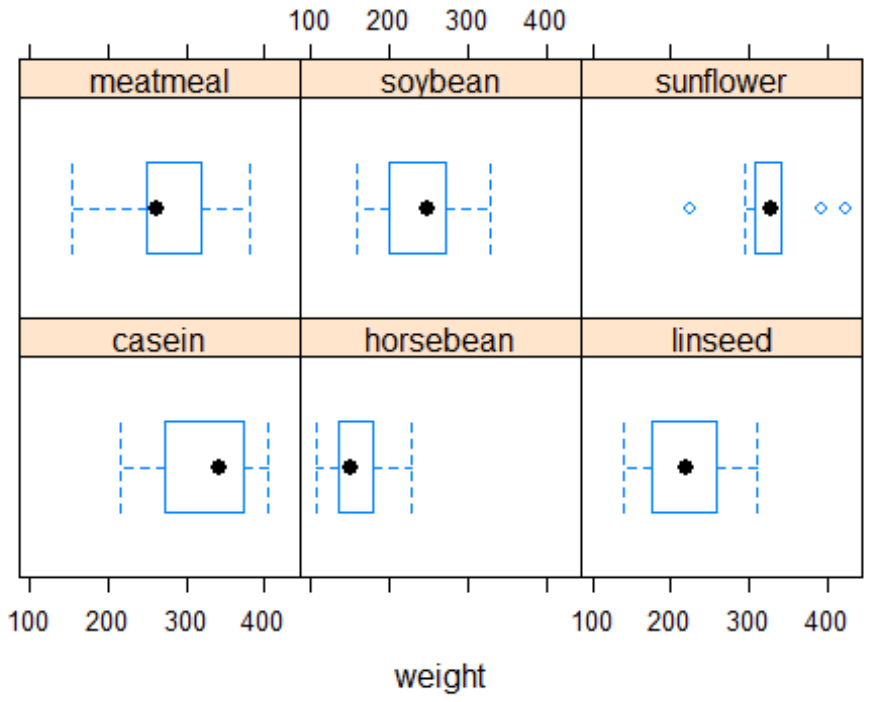

```r
boxplot(chickwts$weight ~ chickwts$feed)
```



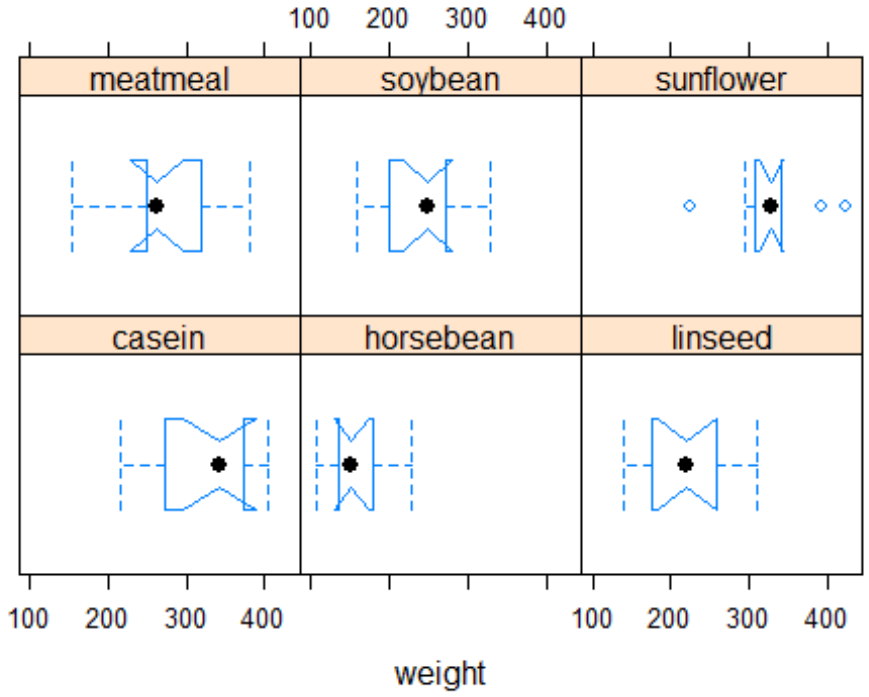

```r
bwplot(weight~feed, data = chickwts)
```



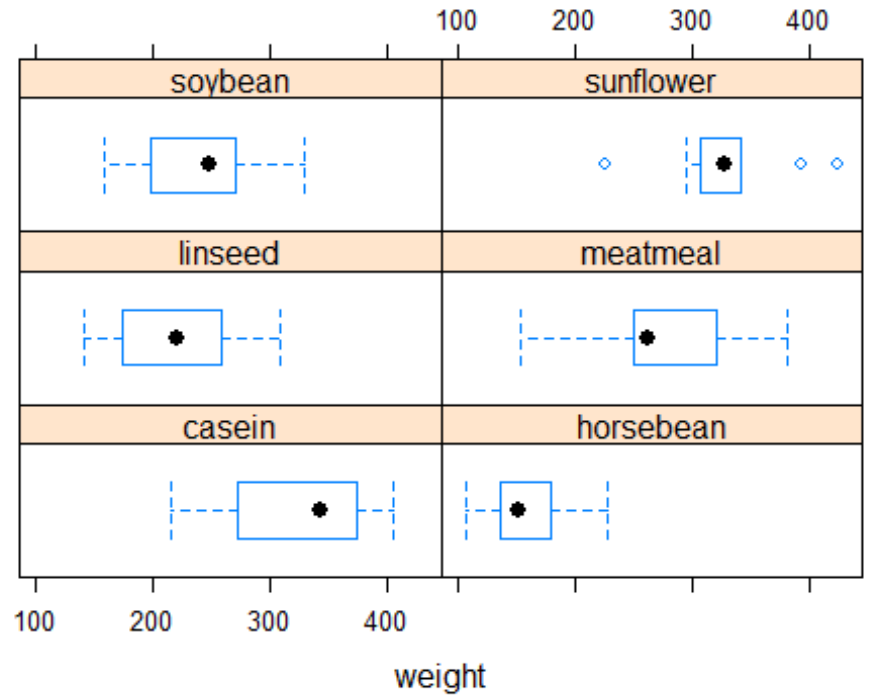

```
bwplot(~weight | factor(feed), data = chickwts)
```



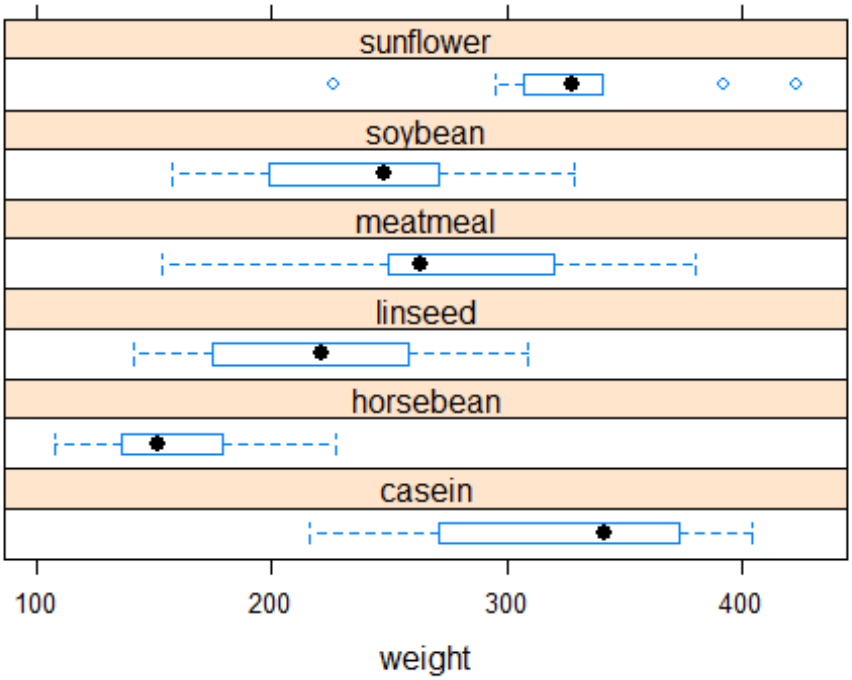

```
bwplot(~weight | factor(feed), data = chickwts, notch = TRUE)
```




```
bwplot(~weight | factor(feed), data = chickwts, layout = c(2,3))
```

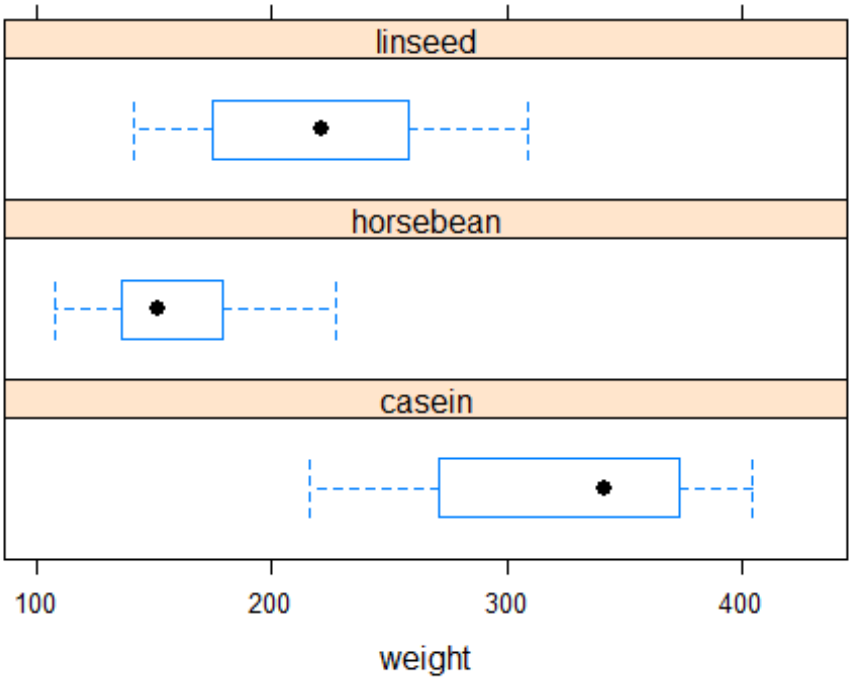


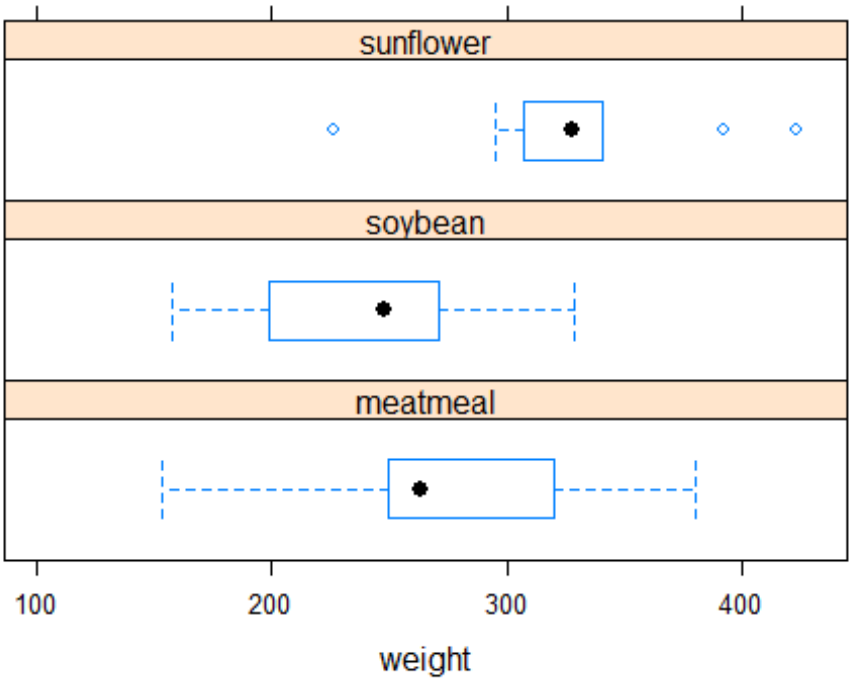

```
bwplot(~weight | factor(feed), data = chickwts, layout = c(1,6))
```



Layout c(column, rows, pages)

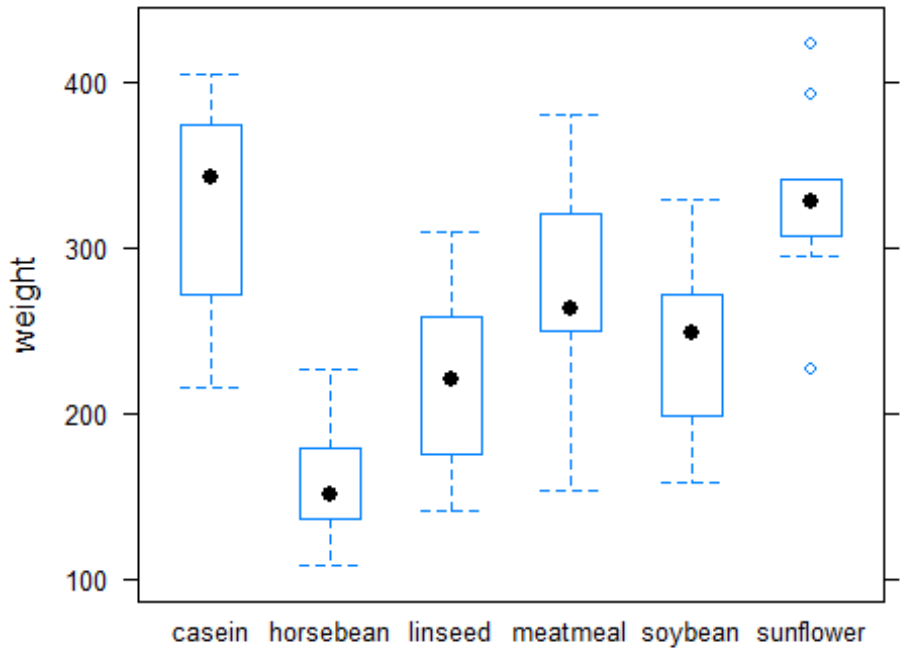
```
bwplot(~weight | factor(feed), data = chickwts, layout = c(1,3,2))
```



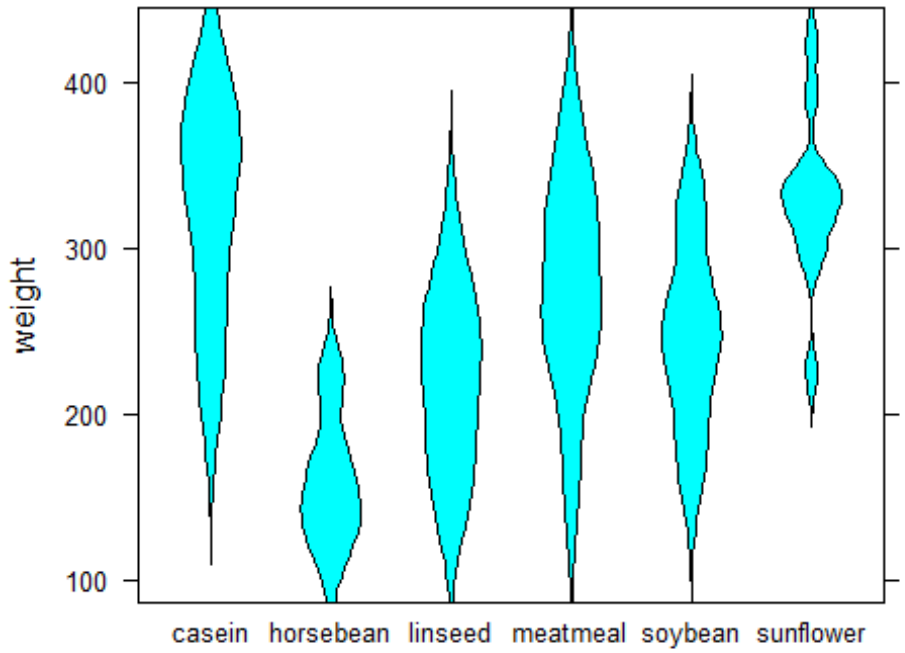


Violine Plot

```
bwplot(weight~feed, data = chickwts)
```




```
bwplot(weight~feed, data = chickwts,
       panel = panel.violin)
```

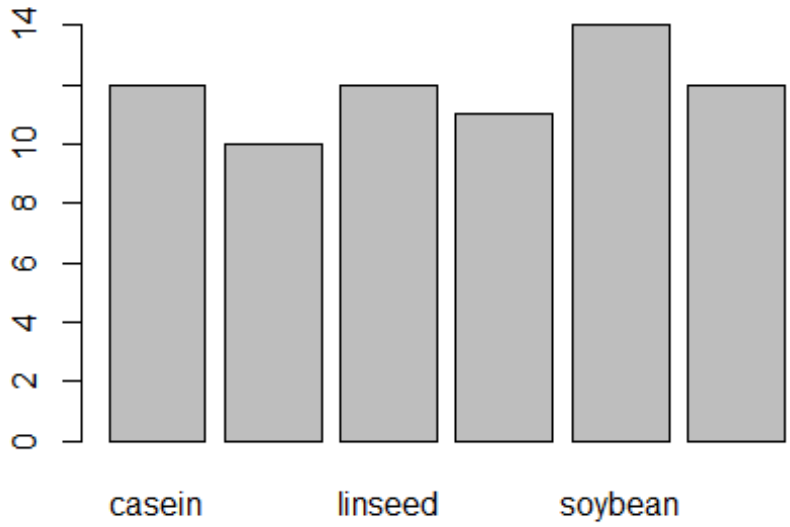


---

4. Bar Chart - barplot() || Lattice equivalent barchart()

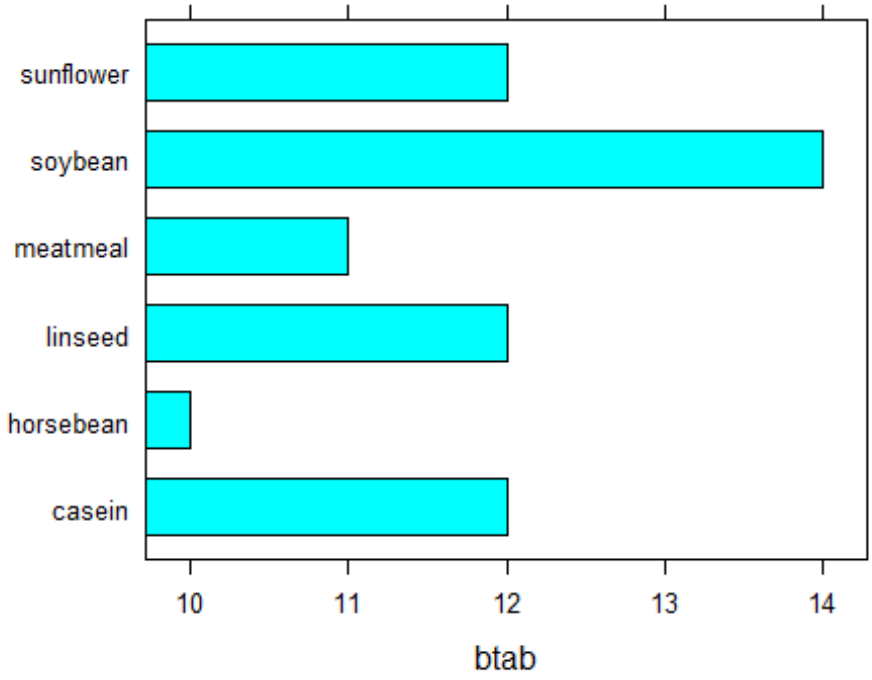
Base Command

```
table(chickwts$feed)

##
##    casein horsebean   linseed  meatmeal   soybean sunflower
##        12        10        12        11        14        12

barplot(table(chickwts$feed))
```



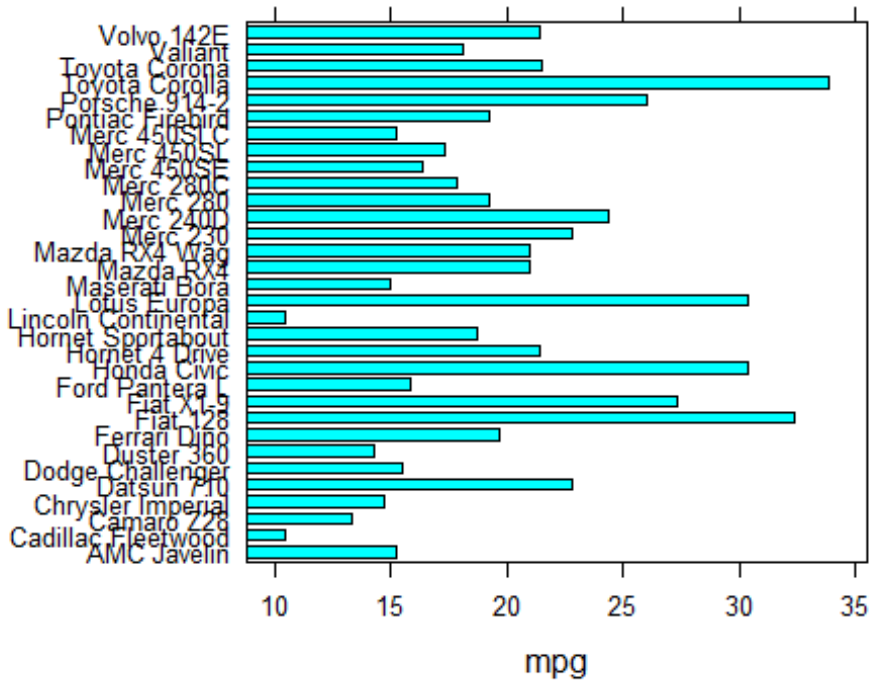
Lattice

```
btab <- table(chickwts$feed)
barchart(~btab, data=chickwts)
```

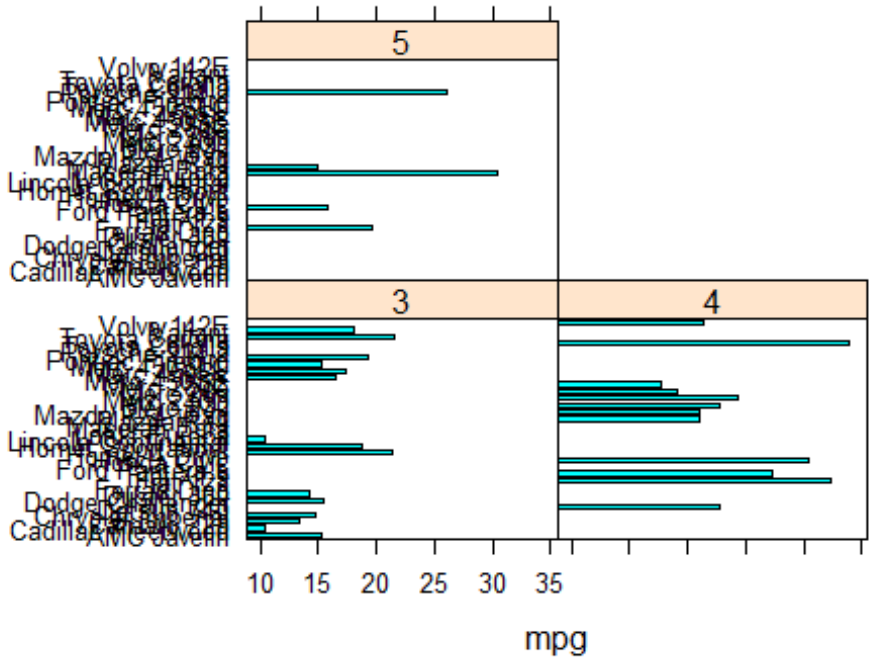


Second example

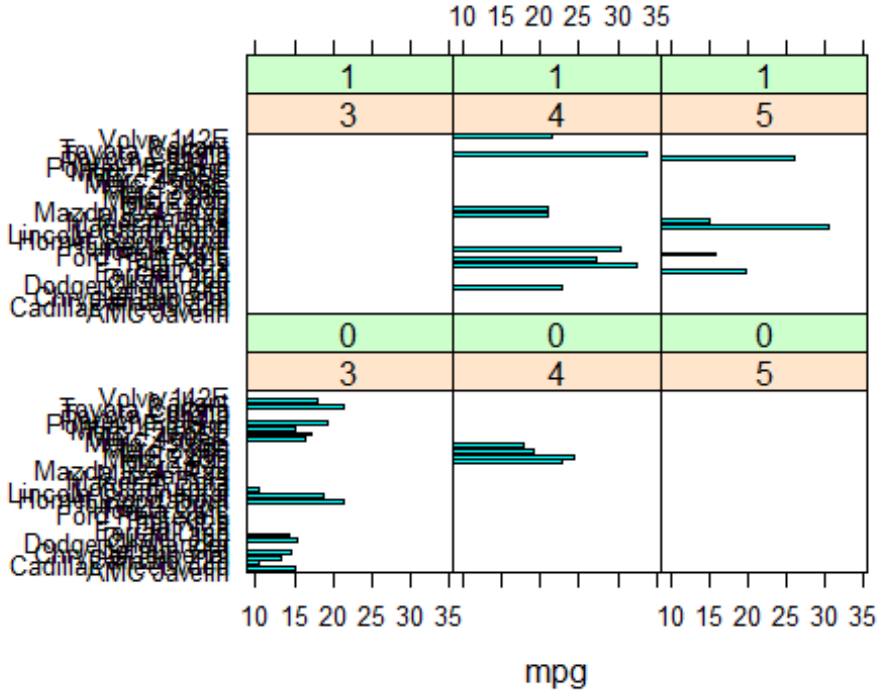
```
barchart(rownames(mtcars) ~ mpg, data=mtcars)
```



```
barchart(rownames(mtcars) ~ mpg | factor(gear), data=mtcars)
```




```
# 0 = automatic, 1 = manual

barchart(rownames(mtcars) ~ mpg | factor(gear) + factor (am), data=mtcars)
```

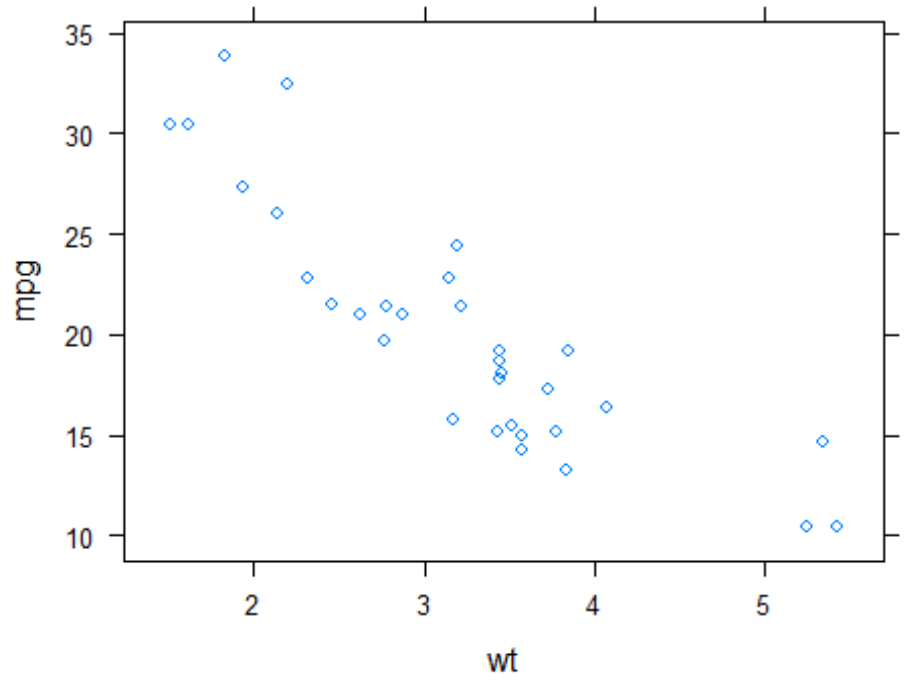


---

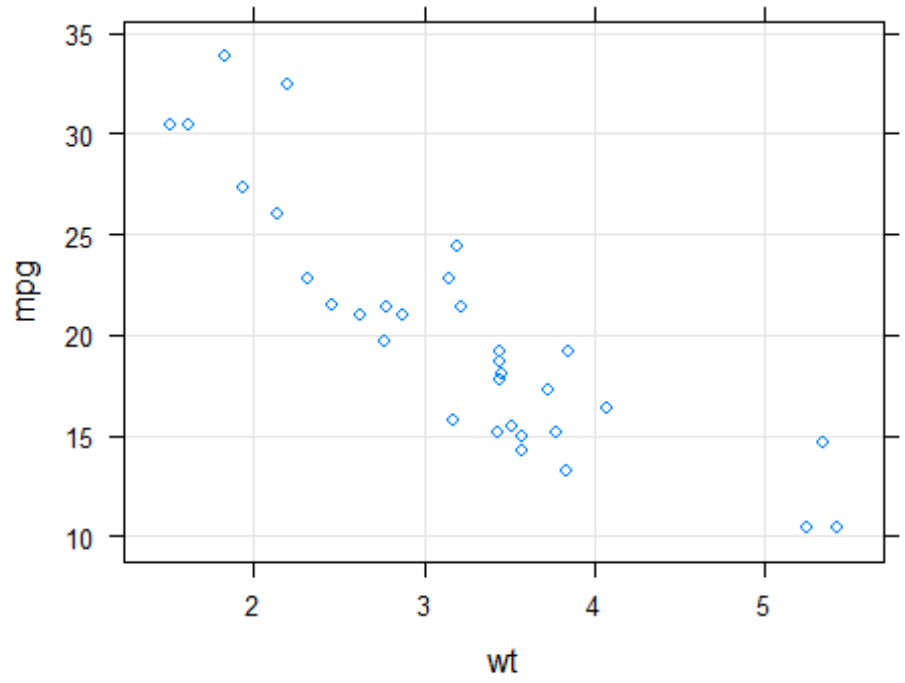
5. Scatter Plot - plot() || Lattice equivalent xyplot(),

splom() scatter matrix plot, cloud() 3D Plot

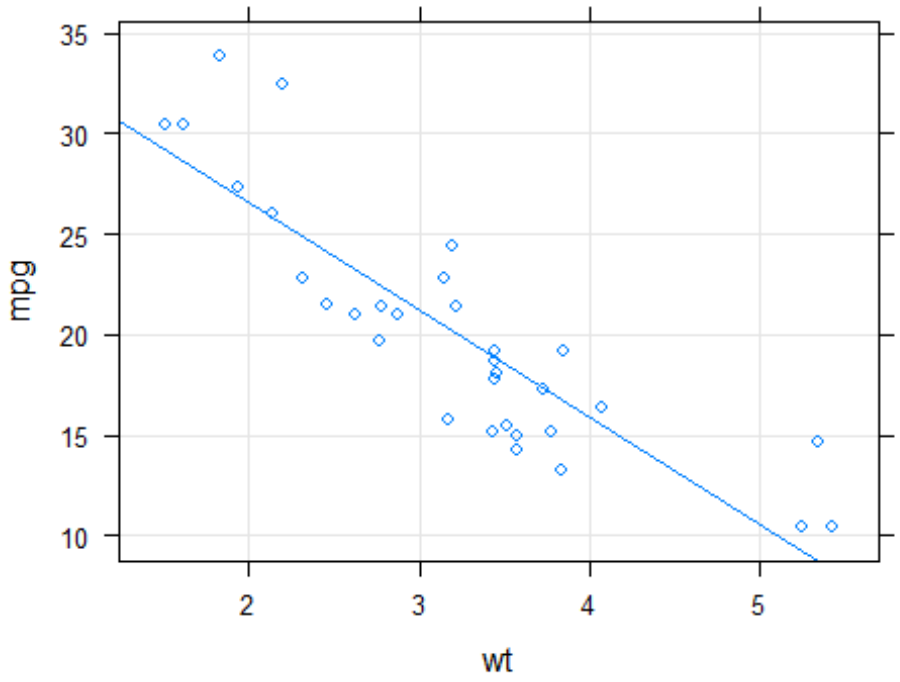
```
xyplot(mpg ~ wt, data = mtcars)
```




```
xyplot(mpg ~ wt, data = mtcars, grid=TRUE)
```

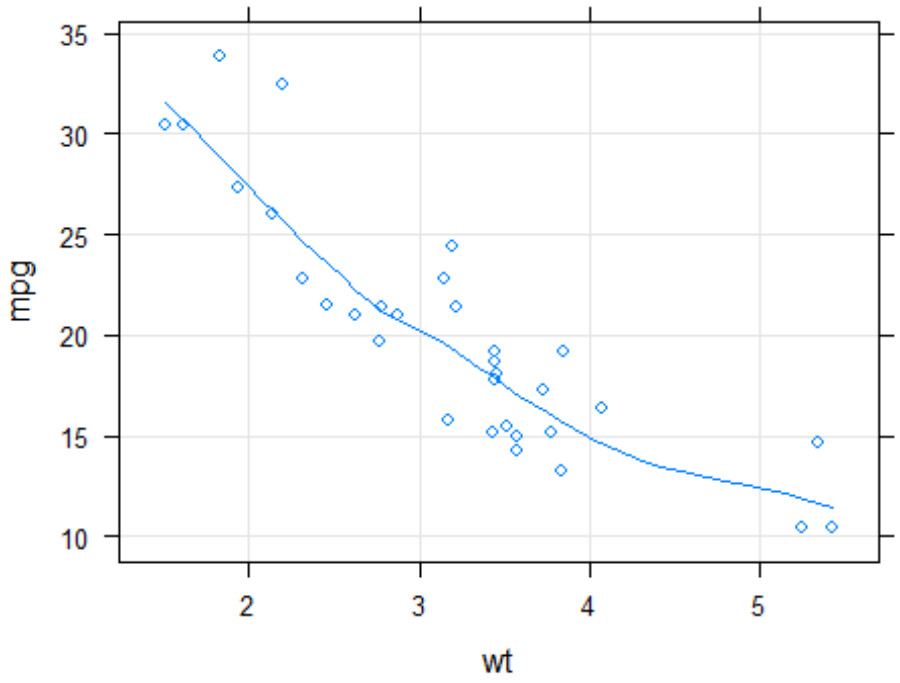



```
xyplot(mpg ~ wt, data = mtcars, grid=TRUE, type = c("p", "r"))
```

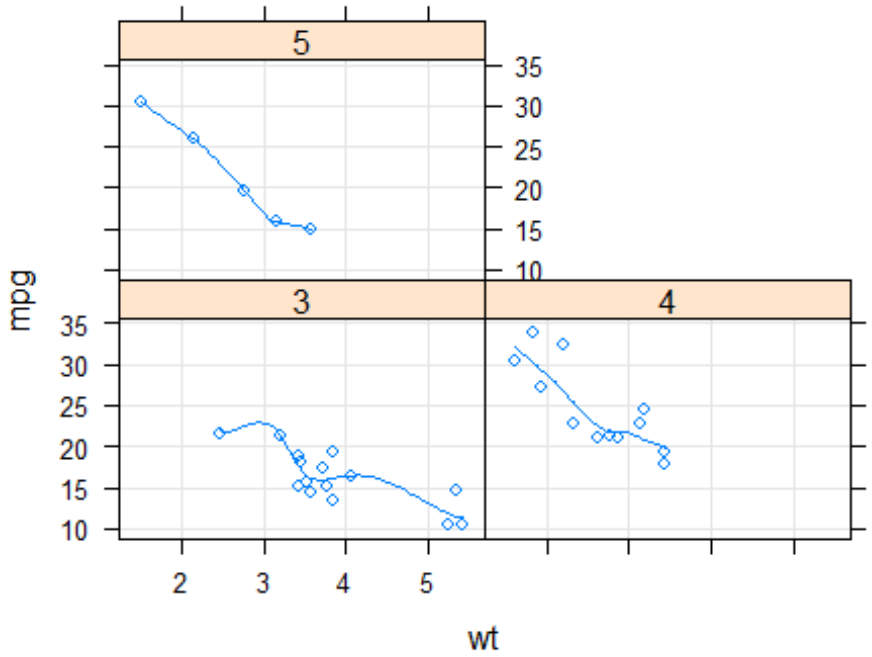


Types: p = point, r = linear regression, smooth = smooth regression

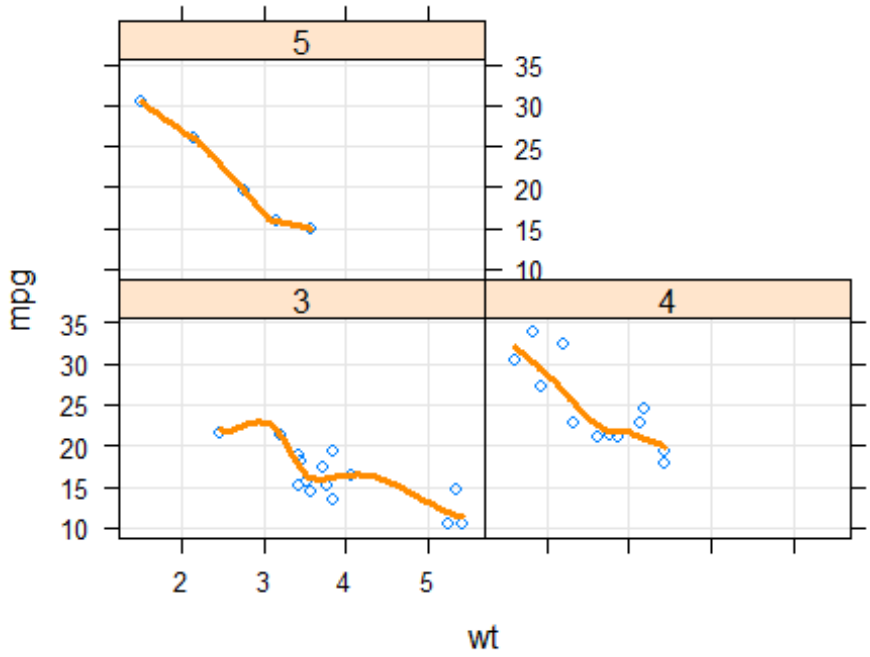
```
xyplot(mpg ~ wt, data = mtcars, grid=TRUE, type = c("p", "smooth"))
```



```
xyplot(mpg ~ wt | factor(gear), data = mtcars, grid=TRUE, type = c("p",
"smooth"))
```

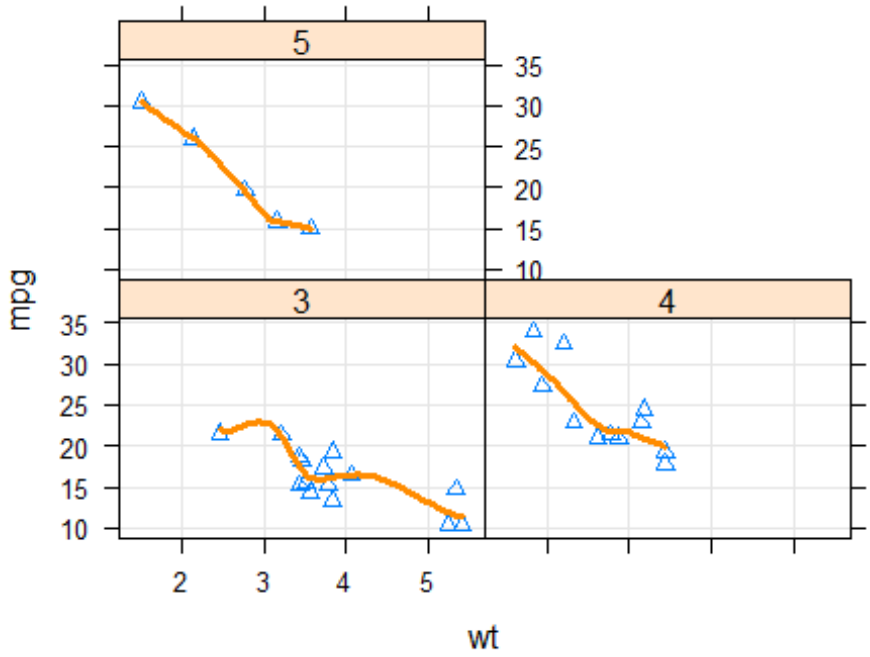



```
xyplot(mpg ~ wt | factor(gear), data = mtcars, grid=TRUE,
       type = c("p", "smooth"), col.line = "darkorange", lwd = 3)
```



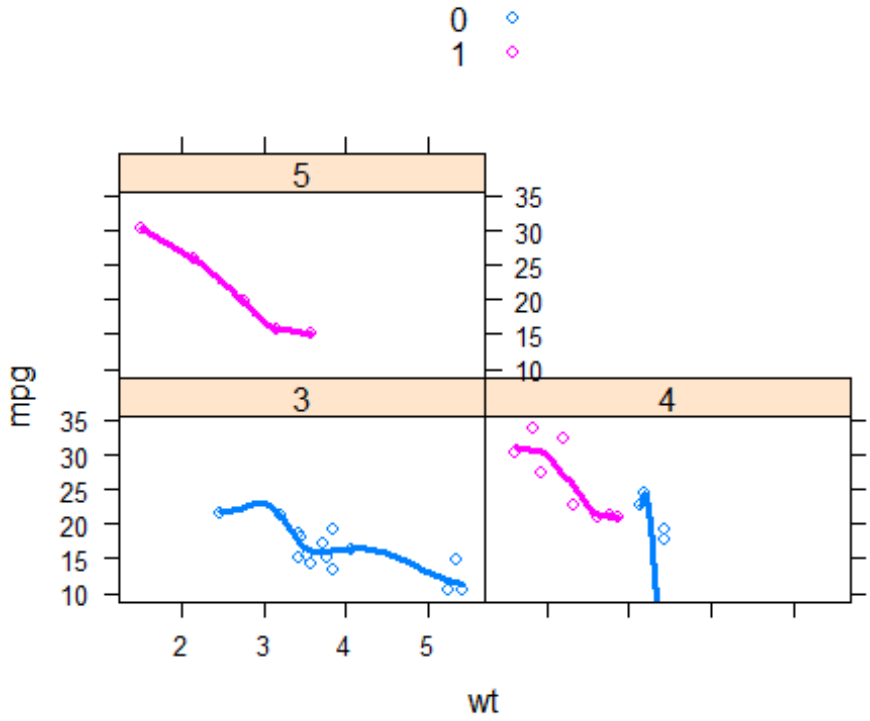
Changing Plot Character (pch=) 1. Circle 2. Triangle 3. Plus 4. Cross 5. Diamond, 6. Reverese triangle 7. Box and crossed

```
xyplot(mpg ~ wt | factor(gear), data = mtcars, grid=TRUE,
       type = c("p", "smooth"), col.line = "darkorange", lwd = 3, pch=2)
```



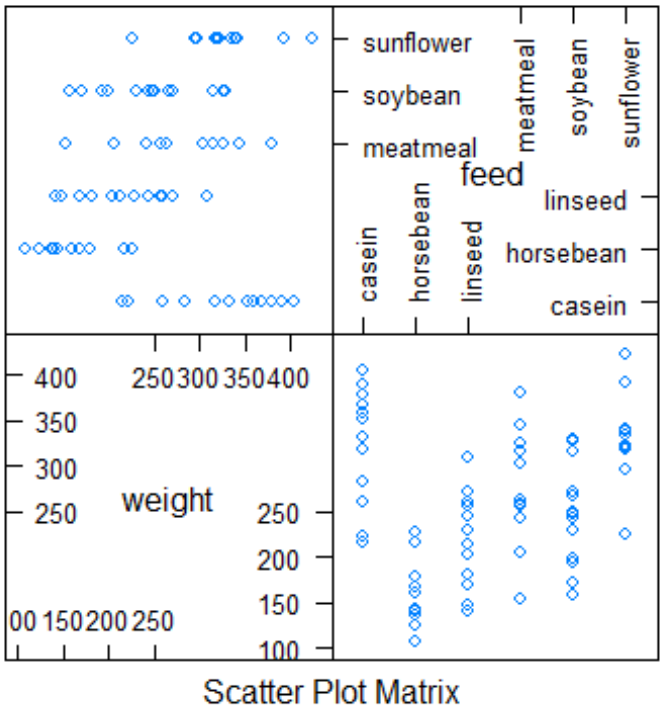
group by am

```
xyplot(mpg ~ wt | factor(gear), data = mtcars, type = c("p", "smooth"),
       group = factor(am), lwd = 3,auto.key = TRUE)
```

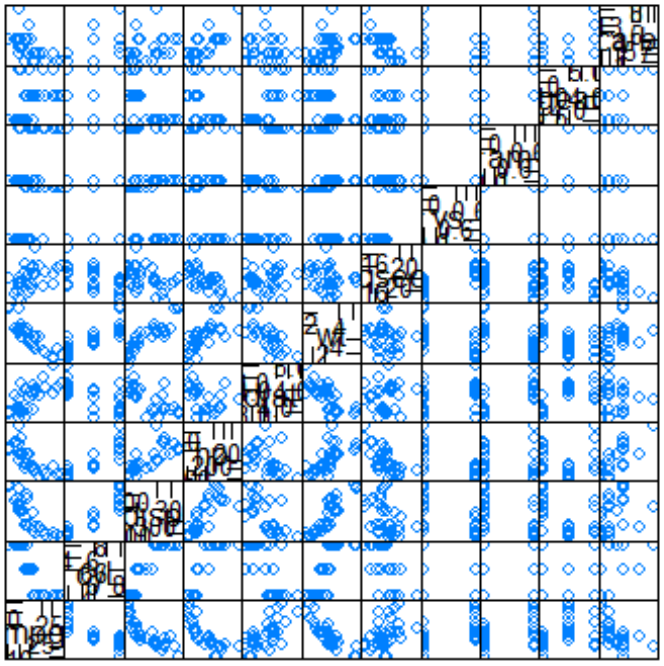


Scatter Plot Matrix - splom()

```
splom(chickwts)
```



```
splom(mtcars)
```

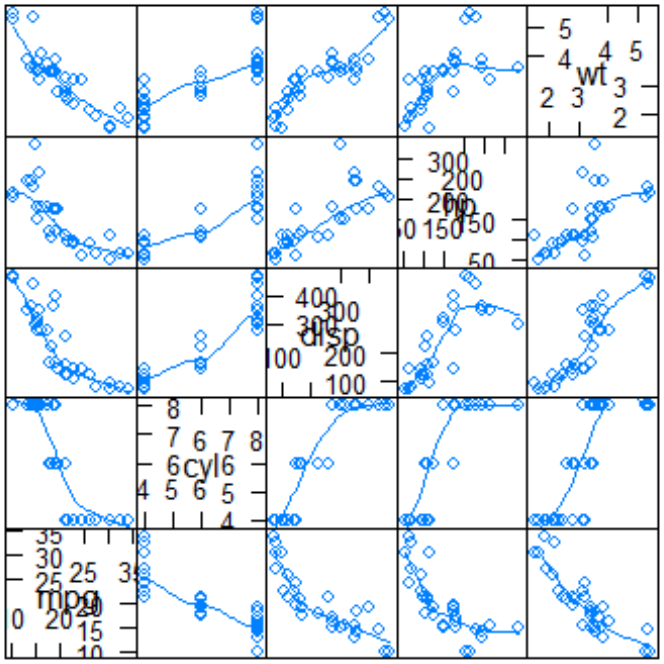


Scatter Plot Matrix

```
str(mtcars)
```

```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

```
splom(mtcars[c(1,2,3,4,6)], type = c("p", "smooth"))
```



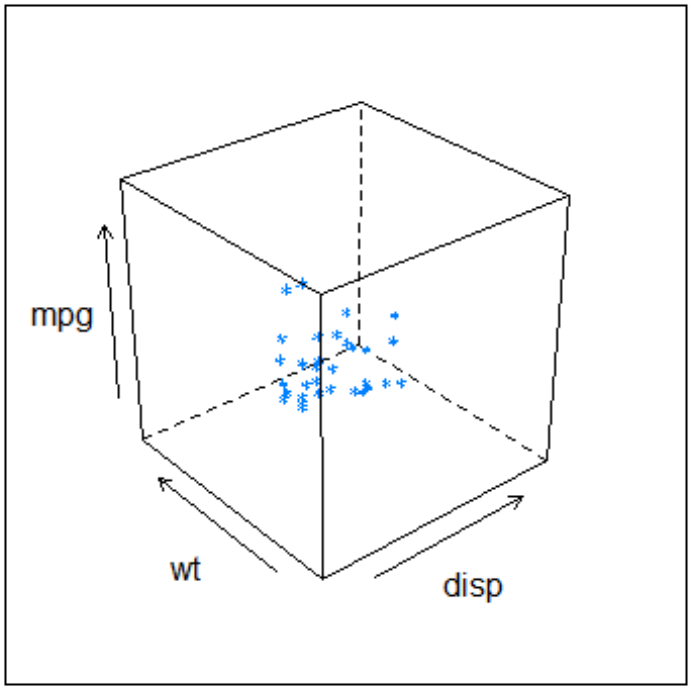
Scatter Plot Matrix

3D Plot - cloud() z ~ x * y

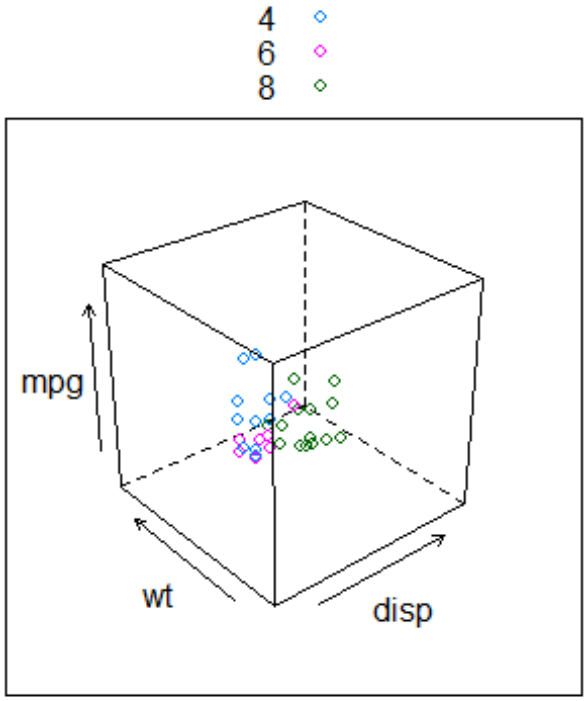
```
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
cloud(mpg ~ disp * wt, data = mtcars)
```




```r
cloud(mpg ~ disp * wt, data = mtcars, group = factor(cyl), auto.key = TRUE)
```

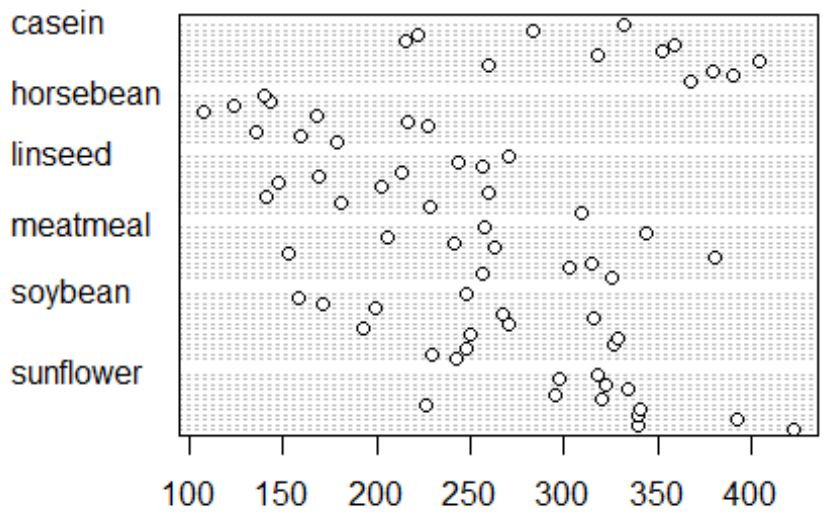


---

6 and 7. Dot and Strip Plot - dotplot() , stripplot()

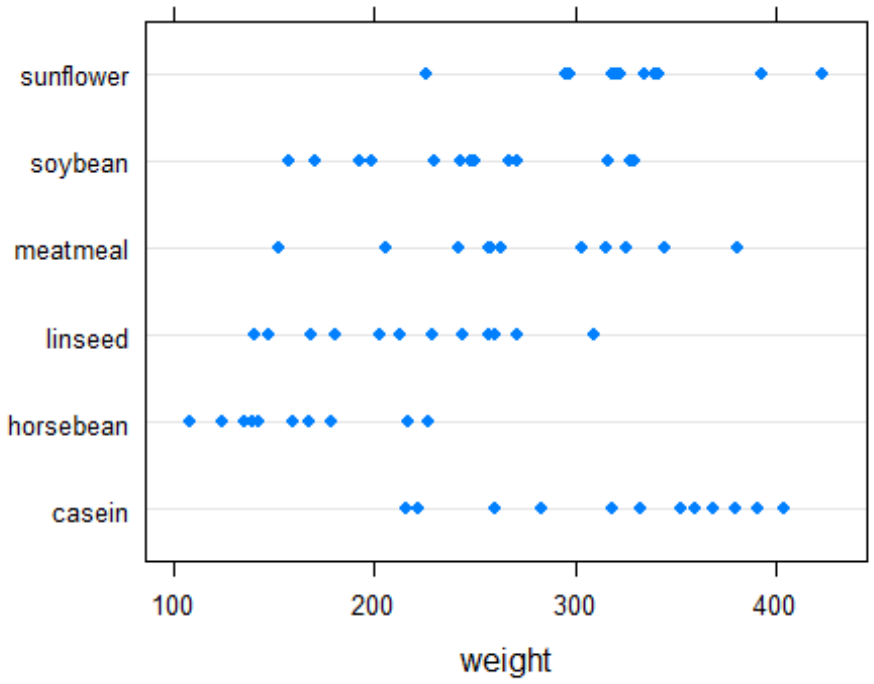
Base command

```
dotchart(chickwts$weight, groups = chickwts$feed)
```

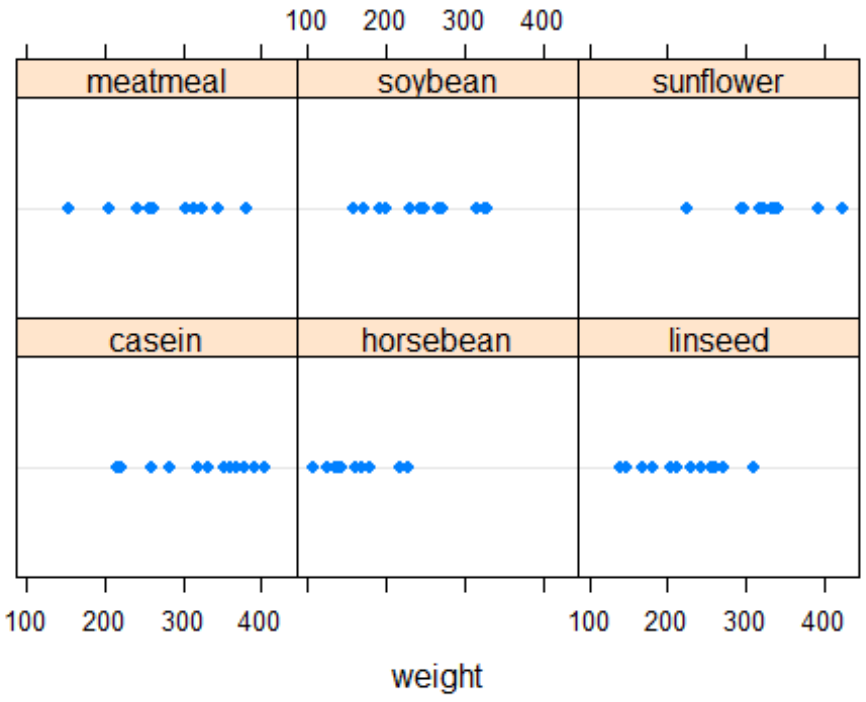


Lattice

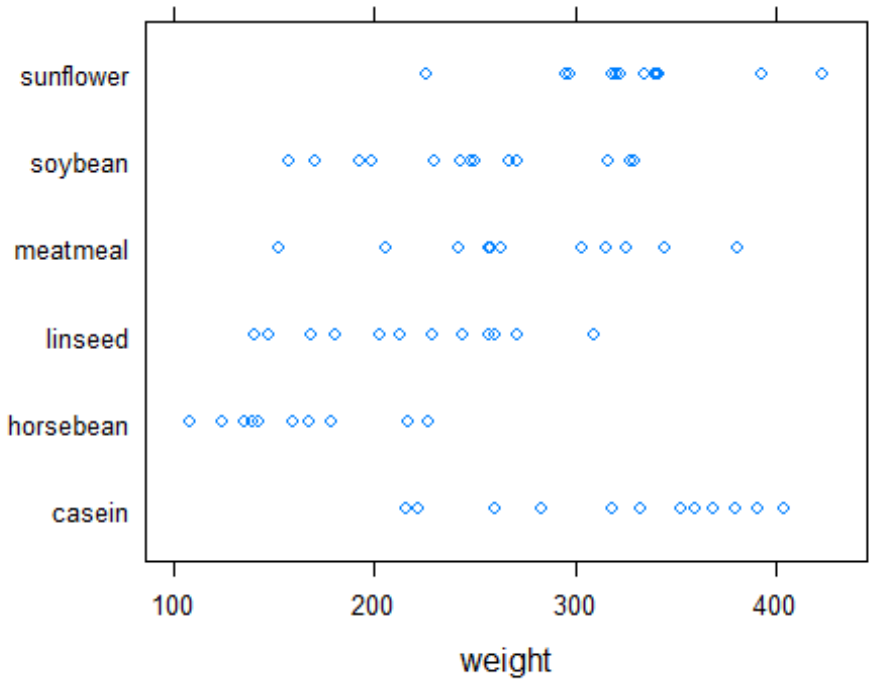
```
dotplot(feed~weight, data= chickwts)
```




```
dotplot(~weight | feed, data= chickwts)
```




```
stripplot(feed~weight, data= chickwts)
```


sunflower
soybean
meatmeal
linseed
horsebean
casein
100
200
300
400
weight