

Elab 4.9

[Computation/language/language102]

When the explanatory variable is categorical, you have a choice of the order used in displays. Most people assume that alphabetical order is called, but it's can be better to arrange the order to minimize haphazard ups and downs in the graph. By making the graph more orderly, not only is any overall pattern made clear but the eye is able to pick out deviations from the overall pattern.

The reorder operator will set things up so that R takes a categorical variable in a new order. To be precise, `reorder` returns a new categorical variable which can be used in place of the original. It takes several arguments and works in a way that's analogous to `group`:

1. The categorical variable to be re-ordered.
2. Another vector of the same length
3. A function

Like `group`, the `reorder` operator divides the vector into groups according to the levels of the categorical variable. It then applies the function to each group. The function should return a number or "score." The order of the levels in new categorical variable is determined by that score.

For example, consider the `sector` variable in the Current Population Survey data. By default, the order of the levels is alphabetical:

```
> cps = ISMdata("cps.csv")
> levels(cps$sector)
[1] "clerical" "const"      "manag"      "manuf"
[5] "other"    "prof"        "sales"      "service"
```

The following command reorders the levels of `sector` in the order of the median wage for each sector:

```
> new = reorder( cps$sector, cps$wage, median)
> levels(new)
[1] "service" "sales"    "manuf"    "other"
[5] "clerical" "const"    "prof"     "manag"
```

This order makes a somewhat more orderly boxplot.

```
> bwplot(wage~new, data=cps, scales=list(rot=45), ylim=c(0, 30))
```

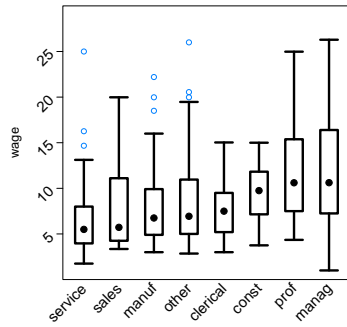


Figure out how to use `reorder` to put the categorical variable in order according to each of these criteria. Enter your commands for each:

1. mean value of wages
2. maximum value of wages
3. number of cases in each group (hint: `length`)
4. mean age in each group