Factors

factor: categorical variables

```
position <- c('SG', 'PG', 'PF', 'SF', 'C')
```

Since factor ≠ character, the elements don't have `""`. `Levels` shows all categories.

```
position_factor <- factor(position)
position_factor
```

```
## [1] SG PG PF SF C
## Levels: C PF PG SF SG
```

`str()` checks a data structure. Notice that the categories are assigned integers.

```
class(position)
```

```
## [1] "character"
```

```
class(position_factor)
```

```
## [1] "factor"
```

```
str(position_factor)
```

```
##  Factor w/ 5 levels "C","PF","PG",..: 5 3 2 4 1
```

So its type is `integer`.

```
typeof(position_factor)
```

```
## [1] "integer"
```

Factor can be treated like a vector in manipulation.

```
position <- factor(position)

position[position == "SG"] # extract elemets whose value is "SG"
```

```
## [1] SG
## Levels: C PF PG SF SG
```

```r
position[position == "spectator"] # a factor vector of length 0
```

```
## factor(0)
## Levels: C PF PG SF SG
```

```r
position[1:3] # extract first 3 elements
```

```
## [1] SG PG PF
## Levels: C PF PG SF SG
```

```r
position[rep(2, 4)] # extract the second element 4 times
```

```
## [1] PG PG PG PG
## Levels: C PF PG SF SG
```

```r
x <- c("c", "b", "a", "b", "c", "b")
y <- factor(x)
y
```

```
## [1] c b a b c b
## Levels: a b c
```

```r
y[3]
```

```
## [1] a
## Levels: a b c
```

```r
y[y == "sun"]
```

```
## factor(0)
## Levels: a b c
```

```r
length(y)
```

```
## [1] 6
```

```r
sort(y)
```

```
## [1] a b b b c c
## Levels: a b c
```

```r
unique(y)
```

```
## [1] c b a
## Levels: a b c
```

But factor is not a vector. `attributes()` returns all levels (categories) and class.

```r
str(y)
```

```
##  Factor w/ 3 levels "a","b","c": 3 2 1 2 3 2
```

```r
is.vector(y) # factor is different from vector
```

```
## [1] FALSE
```

```r
attributes(y) # shows levels and class
```

```
## $levels
## [1] "a" "b" "c"
##
## $class
## [1] "factor"
```

Qualitative Variables

- Nominal: no orders.
  - city has no orders.

  ```
  cities <- c("SF", "Berkeley", "SJ", "SJ", "SF", "SF")
  ```

- Ordinal: with orders.
  - rating has orders. It can be sorted like "bad" < "average" < "good".

  ```
  rates <- c("good", "bad", "bad", "good", "average", "good", "average", "good")
  ```

```r
rates <- c("good", "bad", "bad", "good", "average", "good", "average", "good")

# not ordered
factor(rates)
```

```
## [1] good    bad     bad     good    average good    average good
## Levels: average bad good
```

Notice that `Levels` is sorted alphabetically as default. Set `levels` to change the order.

```r
# ordered levels
rating <- factor(rates,
                 levels = c("bad", "average", "good"))

rating
```

```
## [1] good    bad     bad     good    average good    average good
## Levels: bad average good
```

To give a quantative relationship, set `ordered`.

```
# ordered levels with "<"
ordered_rating <- factor(rates,
                  levels = c("bad", "average", "good"),
                  ordered = TRUE)
ordered_rating
```

```
## [1] good    bad     bad     good    average good    average good
## Levels: bad < average < good
```