Getting / cleaning data 2

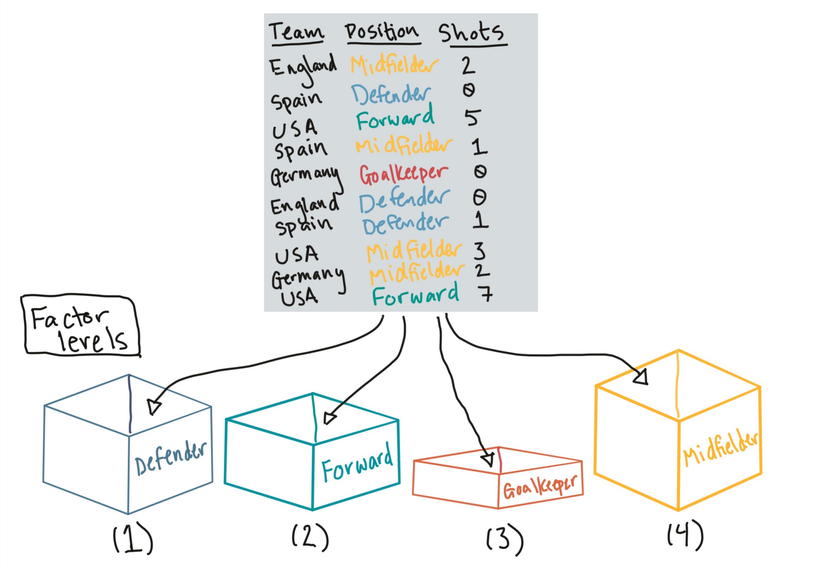
Working with factors

Working with factors

Hadley Wickham has developed a package called `forcats` that helps you work with factors.

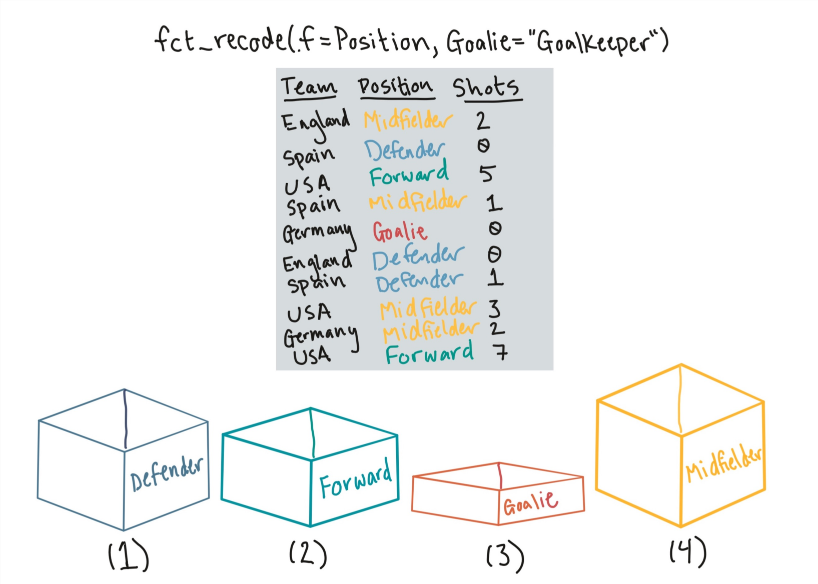
```r
library("forcats")
```

Factors



Team	Position	Shots
England	Midfielder	2
Spain	Defender	0
USA	Forward	5
Spain	Midfielder	1
Germany	Goalkeeper	0
England	Defender	0
Spain	Defender	1
USA	Midfielder	3
Germany	Midfielder	2
USA	Forward	7

fct_recode



```
fct_recode(.f = Position, Goalie = "GoalKeeper")
```

Team	Position	Shots
England	Midfielder	2
Spain	Defender	0
USA	Forward	5
Spain	Midfielder	1
Germany	Goalie	0
England	Defender	0
Spain	Defender	1
USA	Midfielder	3
Germany	Midfielder	2
USA	Forward	7

fct_recode

The `fct_recode` function can be used to change the labels of a function (along the lines of using `factor` with `levels` and `labels` to reset factor labels).

One big advantage is that `fct_recode` lets you change labels for some, but not all, levels. For example, here are the team names:

```
worldcup %>%
  filter(Team == "USA") %>%
  slice(1:3) %>% select(Team, Position, Time)
```

```
##           Team   Position Time
## Beasley    USA Midfielder   10
## Bocanegra  USA   Defender  390
## Bornstein  USA   Defender  200
```

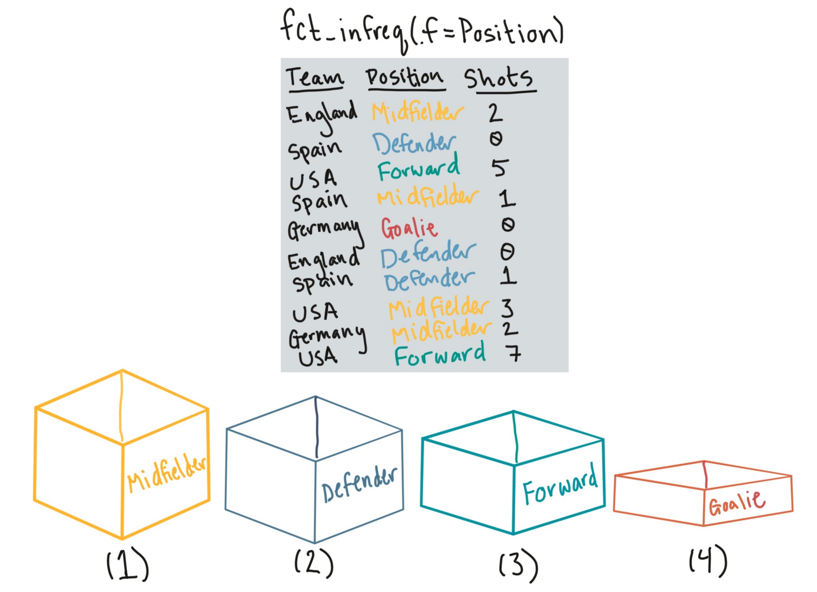
fct_recode

If you just want to change “USA” to “United States”, you can run:

```
worldcup <- worldcup %>%
  mutate(Team = fct_recode(Team, `United States` = "USA"))
worldcup %>%
  filter(Team == "United States") %>%
  slice(1:3) %>% select(Team, Position, Time)
```

```
##             Team   Position Time
## 1 United States Midfielder   10
## 2 United States   Defender  390
## 3 United States   Defender  200
```

fct_infreq



fct_infreq(.f=Position)

Team	Position	Shots
England	Midfielder	2
Spain	Defender	0
USA	Forward	5
Spain	Midfielder	1
Germany	Goalie	0
England	Defender	0
Spain	Defender	1
USA	Midfielder	3
Germany	Midfielder	2
USA	Forward	7

fct_infreq

You can use the `fct_infreq` function to reorder the levels of a factor from most common to least common:

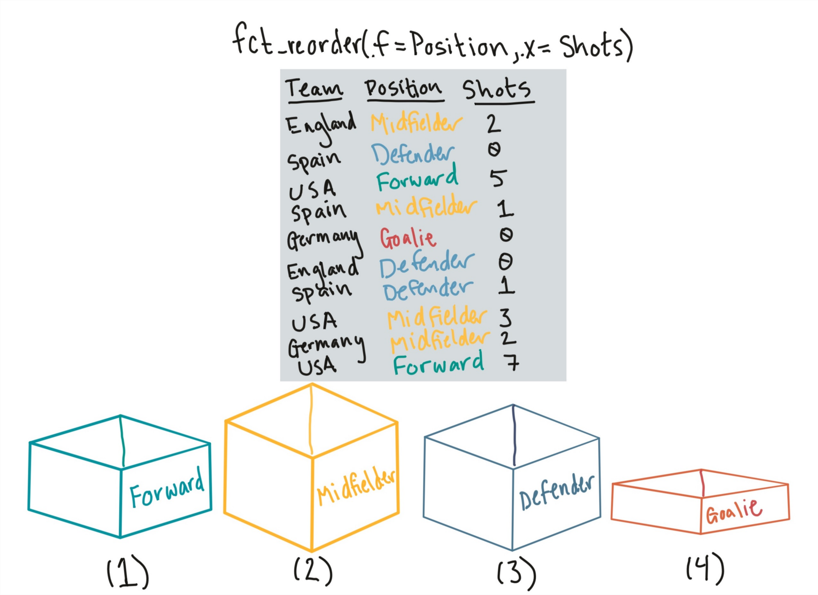
```r
levels(worldcup$Position)
```

```
## [1] "Defender"   "Forward"    "Goalkeeper" "Midfielder"
```

```r
worldcup <- worldcup %>%
  mutate(Position = fct_infreq(Position))
levels(worldcup$Position)
```

```
## [1] "Midfielder" "Defender"   "Forward"    "Goalkeeper"
```

fct_reorder



fct_reorder(.f = Position, .x = Shots)

Team	Position	Shots
England	Midfielder	2
Spain	Defender	0
USA	Forward	5
Spain	midfielder	1
Germany	Goalie	0
England	Defender	0
Spain	Defender	1
USA	Midfielder	3
Germany	midfielder	2
USA	Forward	7

fct_reorder

If you want to reorder one factor by another variable (ascending order), you can use `fct_reorder` (e.g., homework 3). For example, to re-level `Position` by the median shots on goals for each position, you can run:

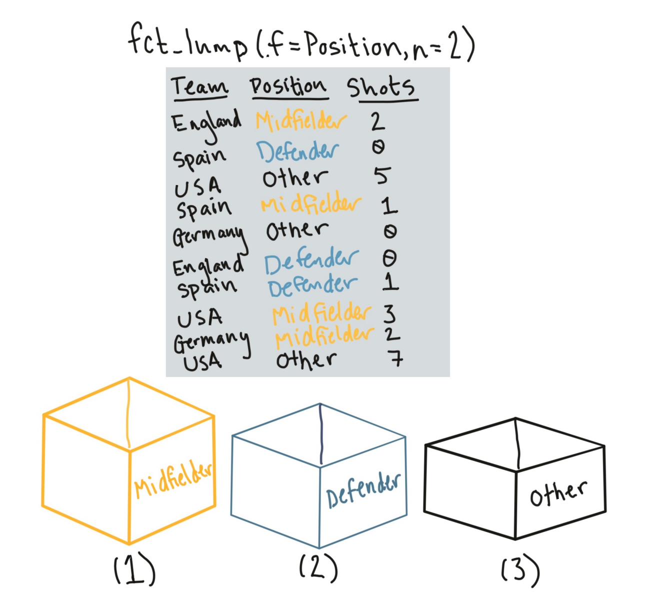
```
levels(worldcup$Position)
```

```
## [1] "Midfielder" "Defender"   "Forward"    "Goalkeeper"
```

```
worldcup <- worldcup %>%
  mutate(Position = fct_reorder(Position, Shots))
levels(worldcup$Position)
```

```
## [1] "Goalkeeper" "Defender"   "Midfielder" "Forward"
```

fct_lump



fct_lump(.f=Position,n=2)

Team	Position	Shots
England	Midfielder	2
Spain	Defender	0
USA	Other	5
Spain	midfielder	1
Germany	Other	0
England	Defender	0
Spain	Defender	1
USA	Midfielder	3
Germany	midfielder	2
USA	Other	7

fct_lump

You can use the fct_lump function to lump uncommon factors into an "Other" category. For example, to lump the two least common positions together, you can run (n specifies how many categories to keep outside of "Other"):

```r
worldcup %>%
  mutate(Position = fct_lump(Position, n = 2)) %>%
  count(Position)
```

```
##     Position   n
## 1   Defender 188
## 2 Midfielder 228
## 3      Other 179
```