

Lecture 5

2023-08-04

More about Tidyverse

Example

We want to know the maximum life expectancy of each country, and the increase in life expectancy (assume monotonic).

```
library(tidyverse)
library(gapminder)

# max life expectancy
View(gapminder %>% group_by(country) %>% summarize(max_life_exp = max(lifeExp)))

# increase in life expectancy
View(gapminder %>% group_by(country)
      %>% summarize(max_life_exp = max(lifeExp) - min(lifeExp)))
```

In general, to find any function of the **vector** life expectancy, replace **max** with **func**, the desired function. To find the maximum/minimum value, append **\$life_exp** (the name of the second column) to make it a vector, and **%>% max** for maximum, same for minimum.

Some Functions

We can arrange things with **arrange**.

```
library(tidyverse)
View(gapminder %>% arrange(year, gdpPercap))

# To sort in a descending order, we can take the additive inverse
View(gapminder %>% arrange(year, -gdpPercap))

# To find the maximum GDP per capita every year
View(gapminder %>% group_by(year) %>% filter(gdpPercap == max(gdpPercap))
      %>% arrange(year))
```

We can remove duplicates with **distinct**.

```
library(tidyverse)

df <- data.frame(A = c(1, 1, 2),
                  B = c(1, 1, 2),
                  C = c(2, 2, 2))
df %>% distinct %>% View # distinct entries in all
df %>% distinct(A, B) %>% View # distinct entries in A, B, deletes column C

library(babynames)
babynames %>% distinct(name) %>% View # distinct names
babynames %>% distinct(name, sex) %>% View # distinct name, sex pairs
```

We create new columns with `mutate`.

```
library(gapminder)
g_gdp <- gapminder %>% mutate(gdp = gdpPercap * pop/1000) %>% View
g_gdp_share <- gapminder %>% group_by(year) %>%
  mutate(gdp_share = (gdpPercap * pop)/sum(gdpPercap * pop)) %>% View
```

Select columns with `select`.

```
View(gapminder %>% select(gdpPercap, country))
```

Exercises

Compute the number of countries in `gapminder`

```
(gapminder %>% select(country) %>% distinct %>% dim)[1]
```

```
## [1] 142
```

Write a function that computes the number of countries in a `gapminder`-like dataset.

```
num_of_countries <- function(data){
  (data %>% select(country) %>% distinct %>% dim)[1]
}
num_of_countries(gapminder)
```

```
## [1] 142
```

Find the country with the largest life expectancy in the year 1982

```
(gapminder %>% filter(year == 1982) %>% arrange(-lifeExp))$country[1]
```

Write a function that find the largest life expectancy in a given year.

```
max_lifeExp <- function(yr){
  (gapminder %>% filter(year == yr) %>% arrange(-lifeExp))$country[1]
}
max_lifeExp(1987)
```