Introduction to Tidyverse

Claire Liow

5/24/2022

Tidyverse: A dialect of R

- Subsetting Data: slice(), select(), filter()
- Pipe operator: %>%
- Adding new variables: mutate()
- Summarizing data frames

Why Tidyverse?

```r
#install.packages(tidyverse)
library("tidyverse")
```

1. Follows logic
2. Intuitive
3. Easier to follow codes

Subsetting Data and Pipe operator

```r
setwd("~/Dropbox/GitHub/qss-inst-tidyverse/Introduction")
UNpop <- read.csv("data/UNpop.csv")

glimpse(UNpop)
dim(UNpop)

## Subset the first three rows of UNpop with tidyverse
slice(UNpop, n = 1:3)

## Extract the world.pop variable
select(UNpop, world.pop)

## Pipe operator %>% to link commands together
UNpop %>%
  select(world.pop) %>%
  slice(1:3)
```

Adding new variables: mutate()

```r
## Create an additional column based on existing ones
UNpop.mill <- UNpop %>%
                mutate(world.pop.mill = world.pop / 1000)

## Conditional statement: if_else()
UNpop.mill <- UNpop.mill %>%
  mutate(after.1980 = if_else(year >= 1980, "yes", "no"))

## Conditional symbol: %in%
target.years <- c(1950, 1980, 2000)

UNpop.mill <- UNpop.mill %>%
  mutate(year.of.interest = if_else(year %in% target.years, "y
```

Summarizing data frames: summarize()

```r
## Base R
summary(UNpop.mill)
mean(UNpop.mill$world.pop)

## Tidyverse
UNpop.mill %>%
  summarize(mean.pop = mean(world.pop.mill),
            median.pop = median(world.pop.mill))

UNpop.mill %>%
  group_by(after.1980) %>%
  summarize(mean.pop = mean(world.pop.mill))
```