

Exercises day 1 – Introduction

Exercise 1. Download the birth1 data file (birth1.csv).

a) Start RStudio and open a new R Script file (R file)

```
# If you do not have the required packages already:  
install.packages(c("tidyverse", "ggthemes", "gtsummary",  
"lubridate"))
```

```
# Load packages  
library(tidyverse)  
library(ggthemes)  
library(gtsummary)  
library(lubridate)
```

b) Open/read the dataset and describe all the variables (**read_csv**).

```
# Load the .csv file  
birth1 <- read_csv("birth1.csv")
```

c) Save the R Script file for later use

Exercise 2. Explore the data and interpret the output.

a) Summarize continuous (mother's age) and categorical (sex) variables.

```
birth1 |>  
  summarise(  
    n = sum(!is.na(mage)),  
    mean = mean(mage, na.rm = TRUE),  
    sd = sd(mage, na.rm = TRUE),  
    median = median(mage, na.rm = TRUE),  
    q1 = quantile(mage, 0.25, na.rm = TRUE),  
    q3 = quantile(mage, 0.75, na.rm = TRUE),  
    min = min(mage, na.rm = TRUE),  
    max = max(mage, na.rm = TRUE)  
  )
```

```

birth1 |>
  count(sex) |>
  mutate(
    percent = 100 * n / sum(n)
  )

# or
birth1 |>
  select(mage, sex) |>
  tbl_summary(
    statistic = list(
      mage ~ "{mean} ({sd})",
      all_categorical() ~ "{n} ({p}%)"
    )
  )

```

b) Generate new variable (gestational age in weeks)

```

birth1 <- birth1 |>
  mutate(gest_week = floor(gest / 7))
glimpse(birth1)
count(birth1, gest_week)

```

c) Generate and format a new variable birth date, in date format, based on the three variables day, month and year.

```

birth1 <- birth1 |>
  mutate(
    birth_date = make_date(year, month, day)
  )

```

Exercise 3. Explore the data further by conducting bivariable analysis and checking for missing data. Interpret the results.

a) Summarize gestational age by sex

```
birth1 |>
  group_by(sex) |>
  summarise(
    n = n(),
    mean_ga = mean(gest_week, na.rm = TRUE),
    sd_ga = sd(gest_week, na.rm = TRUE),
    median_ga = median(gest_week, na.rm = TRUE),
    min_ga = min(gest_week, na.rm = TRUE),
    max_ga = max(gest_week, na.rm = TRUE)
  )

# or
birth1 |>
  select(sex, gest_week) |>
  tbl_summary(
    by = sex,
    statistic = all_continuous() ~ "{mean} ({sd})"
  )

# Fix implausible gestational weeks
birth1 <- birth1 |>
  mutate(
    gest_week = if_else(
      gest_week > 45,
      NA_real_,
      gest_week
    )
  )
```

b) Check missing data in gestational age

```
birth1 |>
  summarise(
    n_missing = sum(is.na(gest_week)),
    percent_missing = mean(is.na(gest_week)) * 100
  )

# or
birth1 |>
  group_by(sex) |>
  summarise(
    n_missing = sum(is.na(gest_week)),
    percent_missing = mean(is.na(gest_week)) * 100
  )

# or
birth1 |>
  count(missing_ga = is.na(gest_week))
```

Exercise 4. Visualize your data with R graphics and interpret the outputs.

a) Plot child's birth weight versus gestational age. Add a line of best fit.

```
ggplot(
  data = birth1,
  mapping = aes(x = gest_week, y = weight)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = TRUE, color = "blue") +
  labs(
    title = "Birth Weight vs Gestational Age",
    x = "Gestational age (weeks)",
    y = "Birth weight (g)"
  ) +
  theme_minimal()
```

b) Use an appropriate graph to visualize child's birth weight by sex

```
ggplot(  
  data = birth1,  
  mapping = aes(x = sex, y = weight)) +  
  geom_boxplot(outlier.shape = NA) +  
  geom_jitter(width = 0.2, alpha = 0.4) +  
  labs(  
    title = "Birth Weight by Sex",  
    x = "Sex",  
    y = "Birth weight (g)"  
  ) +  
  theme_minimal()
```

```
ggplot(  
  data = birth1,  
  mapping = aes(x = weight, fill = sex)) +  
  geom_density(alpha = 0.3) +  
  labs(  
    x = "Birth weight (g)",  
    y = "Density"  
  ) +  
  theme_minimal()
```