

КОМП'ЮТЕРНИЙ ПРАКТИКУМ №2

Oleksii Stroganov

2024-03-23

Мета роботи:

Навчитися виконувати статистичну обробку при незначних вибірках значень експериментальних даних.

Необхідні умови

```
library(tidyverse)
```

Створюємо таблицю стандартних значень критеріїв для зручності

```
# Use data.frame with standard values
standards <- data.frame(
  n = c(3:8),
  Q = c(0.970,0.829,0.710,0.625,0.568,0.526),
  beta = c(1.153,1.463,1.672,1.822,1.938,2.032),
  q = c(4.96,3.56,3.040,2.78,2.62,2.51),
  t = c(3.18,2.78,2.57,2.45,2.37,2.31)
)
tibble(standards)
```

```
## # A tibble: 6 x 5
##       n      Q  beta    q    t
##   <int> <dbl> <dbl> <dbl> <dbl>
## 1     3 0.97  1.15  4.96  3.18
## 2     4 0.829 1.46  3.56  2.78
## 3     5 0.71  1.67  3.04  2.57
## 4     6 0.625 1.82  2.78  2.45
## 5     7 0.568 1.94  2.62  2.37
## 6     8 0.526 2.03  2.51  2.31
```

Створюємо функцію do_tests

Дана функція використовується з метою упростити розрахунки та оцінку елементів вибірки на грубі похибки. Функція приймає 2 аргумента:

- **df** - дані, для яких виконуються розрахунки
- **ph_sus** - значення елемента, яке потрібно оцінити

Функцію включає в себе проведення очислень, необхідних для оцінки елемента **ph_sus** за Q-критерієм, критерієм Грабса та Романівського.

На виході функція виводить результат оцінки за наведеними критеріями, де значення **FAIL** вказує на грубу похибку, **PASS** вказує, що похибка є допустимою.

```

# Use function to simplify the calculations
do_tests <- function(df, ph_sus) {
  # Set values from standards
  standards <- standards |>
    filter(n == length(df$ph))

  Q_std <- standards[2]
  beta_std <- standards[3]
  q_std <- standards[4]

  # Calculate sd of ph without ph_sus
  ph_sd_star <- df |>
    filter(ph != ph_sus) |>
    pull(ph) |>
    sd()

  # Calculate avg, sd, and var for df
  ph_avg <- mean(df$ph)
  ph_sd <- sd(df$ph)
  ph_var <- ph_sd^2

  # Sort df
  df_sorted <- df |>
    arrange(ph)

  # Select close value for ph_sus from df
  ph_close <- ifelse(min(df$ph) == ph_sus,
                    df_sorted$ph[which(df_sorted$ph == ph_sus) + 1],
                    df_sorted$ph[which(df_sorted$ph == ph_sus) - 1])
  )

  # Calculate values for Q, Grubbs, and Rom using ph_sus
  Q_sus <- abs((ph_sus - ph_close) / (max(df_sorted$ph) - min(df_sorted$ph)))
  beta_sus <- abs((ph_sus - ph_avg) / ph_sd)
  x_edge <- ph_sd_star * q_std
  d_err <- abs(ph_sus - ph_avg)

  # Test ph_sus for Q, Grubbs, and Rom tests
  test_result <- c(
    ifelse(Q_sus > Q_std, "FAIL", "PASS"),
    ifelse(beta_sus > beta_std, "FAIL", "PASS"),
    ifelse(d_err > x_edge, "FAIL", "PASS")
  )

  print(paste0(
    "Tests: Q: ", test_result[1],
    "; Grubbs: ", test_result[2],
    "; Rom: ", test_result[3]))
}

```

Оцінка елементів вибірки

Використовуємо вибірку 23 варіанта

```
df_1 <- data.frame(ph = c(7.88, 7.86, 7.91, 7.89, 7.91, 7.88, 7.78, 7.53))
```

Спершу перевіряємо найменше значення вибірки - $pH = 7.53$.

```
ph_sus_1 <- 7.53  
# First test (all tests failed)  
do_tests(df_1, ph_sus_1)
```

```
## [1] "Tests: Q: FAIL; Grubbs: FAIL; Rom: FAIL"
```

Як можна побачити всі три оцінки дали результат FAIL, отже:

$pH = 7.53$ — Груба похибка

Наступним перевіряємо значення $pH = 7.78$.

```
df_2 <- df_1 |> filter(ph != ph_sus_1)  
ph_sus_2 <- 7.78  
# Second test (all tests failed)  
do_tests(df_2, ph_sus_2)
```

```
## [1] "Tests: Q: FAIL; Grubbs: FAIL; Rom: FAIL"
```

Як і в попередньому результаті похибка виявилась грубою:

$pH = 7.78$ — Груба похибка

Перевіряємо $pH = 7.86$

```
df_3 <- df_2 |> filter(ph != ph_sus_2)  
ph_sus_3 <- 7.86  
# Third test (all tests passed)  
do_tests(df_3, ph_sus_3)
```

```
## [1] "Tests: Q: PASS; Grubbs: PASS; Rom: PASS"
```

В цей раз елемент пройшов всі критерії, тому:

$pH = 7.86$ — Допустима похибка

Перевіряємо $pH = 7.91$

```
df_4 <- df_3  
ph_sus_4 <- 7.91  
# Second test (all tests passed)  
do_tests(df_4, ph_sus_4)
```

```
## [1] "Tests: Q: PASS; Grubbs: PASS; Rom: PASS"
```

І в цей раз елемент пройшов всі критерії, отже:

$pH = 7.91$ — Допустима похибка

Визначення довірчого інтервала

```
# Select the ph from last data.frame
ph <- df_4$ph

# Calculation to get delta_x
ph_sd <- sd(ph)
ph_avg <- mean(ph)
ph_s_x <- ph_sd / sqrt(length(ph))
# Pull the students t coefficient from standards
t <- standards |>
  filter(n == (length(ph) - 1)) |>
  pull(t)
delta_x <- ph_s_x * t
delta_x
```

```
## [1] 0.02036273
```

Остаточний результат

Останнім записуємо середнє значення рН з врахуванням тільки допустимих похибок та довірчого інтервалу:

```
print(paste0("pH = ",round(ph_avg,2),"±",round(delta_x,2)))
```

```
## [1] "pH = 7.89±0.02"
```

$$pH = 7.89 \pm 0.02$$