sample\_calculation.Rmd

2024-01-21

Table of Contents

# EDA

data <- read.csv2("../data/team\_5.csv")  
data\_filtered <- data %>%  
 mutate\_if(is.character, as.factor) %>%  
 mutate(  
 V0\_DEM\_GEN = factor("ж"),  
 V0\_DEM\_AGE = V0\_DEM\_AGE + 20  
 )  
data\_filtered %>% glimpse()

## Rows: 52  
## Columns: 32  
## $ ID <int> 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29,…  
## $ V0\_GRP <fct> Мастодинон, Мастодинон, Мастодинон, Мастодинон, Мастод…  
## $ V0\_DEM\_GEN <fct> ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, ж, …  
## $ V0\_DEM\_AGE <dbl> 55, 47, 55, 45, 45, 51, 43, 50, 49, 54, 45, 46, 54, 46…  
## $ V1\_TIDES\_AMOUNT <int> 2, 5, 5, 6, 5, 6, 4, 7, 5, 5, 7, 1, 5, 4, 2, 6, 4, 8, …  
## $ V2\_TIDES\_AMOUNT <int> 4, 4, 8, 6, 0, 4, 3, 3, 6, 3, 5, 7, 5, 9, 1, 2, 7, 1, …  
## $ V1\_CB\_WBC <dbl> 4.3, 6.5, 5.8, 6.4, 6.6, 4.7, 5.9, 5.0, 9.0, 6.9, 7.9,…  
## $ V1\_CB\_RBC <dbl> 4.1, 5.1, 4.8, 4.2, 4.5, 5.2, 5.6, 4.6, 5.4, 4.5, 5.3,…  
## $ V1\_CB\_HGB <dbl> 131.8, 144.9, 147.1, 127.7, 137.7, 121.1, 122.5, 137.4…  
## $ V1\_CB\_HCT <dbl> 0.40, 0.41, 0.44, 0.45, 0.43, 0.39, 0.41, 0.42, 0.41, …  
## $ V1\_CB\_PLT <dbl> 269.7, 318.2, 295.0, 342.7, 291.4, 240.6, 315.5, 302.7…  
## $ V1\_CB\_NEUT. <dbl> 5.4, 2.8, 3.1, 4.1, 2.3, 3.3, 5.9, 4.9, 2.9, 2.6, 5.2,…  
## $ V1\_CB\_LYM. <dbl> 1.3, 1.2, 1.7, 2.2, 1.3, 1.4, 1.4, 2.1, 1.8, 2.2, 1.8,…  
## $ V1\_CB\_MON. <dbl> 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,…  
## $ V1\_CB\_EO. <dbl> 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,…  
## $ V1\_CB\_BAS. <dbl> 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,…  
## $ V1\_NORM\_ECG <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, …  
## $ V1\_NORM\_PHYS <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## $ V1\_NORM\_VIT <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## $ V2\_CB\_WBC <dbl> 5.4, 7.3, 4.0, 7.7, 4.5, 4.7, 6.8, 4.5, 5.1, 8.0, 5.2,…  
## $ V2\_CB\_RBC <dbl> 4.4, 4.0, 5.0, 5.5, 5.2, 4.4, 5.5, 4.1, 4.5, 5.0, 5.0,…  
## $ V2\_CB\_HGB <dbl> 126.0, 129.2, 122.9, 136.1, 124.4, 141.0, 149.5, 134.5…  
## $ V2\_CB\_HCT <dbl> 0.46, 0.40, 0.42, 0.42, 0.45, 0.48, 0.43, 0.41, 0.44, …  
## $ V2\_CB\_PLT <dbl> 302.1, 318.1, 290.8, 338.6, 314.4, 265.4, 347.8, 294.2…  
## $ V2\_CB\_NEUT. <dbl> 3.3, 2.0, 2.2, 2.6, 3.1, 3.6, 3.3, 2.5, 4.3, 3.2, 5.2,…  
## $ V2\_CB\_LYM. <dbl> 1.7, 1.2, 1.7, 1.2, 1.9, 1.1, 1.1, 1.9, 1.6, 2.0, 1.2,…  
## $ V2\_CB\_MON. <dbl> 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,…  
## $ V2\_CB\_EO. <dbl> 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,…  
## $ V2\_CB\_BAS. <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## $ V2\_NORM\_ECG <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …  
## $ V2\_NORM\_PHYS <int> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, …  
## $ V2\_NORM\_VIT <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …

# Анализ первичной конечной точки эффективности

## Проведем тест на нормальность

shapiro.test(data$V2\_TIDES\_AMOUNT)

##   
## Shapiro-Wilk normality test  
##   
## data: data$V2\_TIDES\_AMOUNT  
## W = 0.96633, p-value = 0.1472

Данные распределены нормально.

## t-test

Проведем t-test для независимых выборок с одинаковыми дисперсиями.

# Разделение данных на две группы  
group1 <- data %>%   
 filter(V0\_GRP == "Мастодинон")  
group2 <- data %>%   
 filter(V0\_GRP == "Плацебо")  
  
# Проведение t-теста  
t\_test\_result <- t.test(group1$V2\_TIDES\_AMOUNT, group2$V2\_TIDES\_AMOUNT, var.equal = T, alternative = "less", mu = -2)  
  
# Вывод результатов теста  
print(t\_test\_result)

##   
## Two Sample t-test  
##   
## data: group1$V2\_TIDES\_AMOUNT and group2$V2\_TIDES\_AMOUNT  
## t = 3.1724, df = 50, p-value = 0.9987  
## alternative hypothesis: true difference in means is less than -2  
## 95 percent confidence interval:  
## -Inf 0.938979  
## sample estimates:  
## mean of x mean of y   
## 4.423077 4.500000

# Анализ параметров безопасности

## Количественные переменные

data <- read.csv2("../data/team\_5.csv")  
data\_filtered <- data %>%  
 mutate\_if(is.character, as.factor) %>%  
 mutate(  
 V0\_DEM\_GEN = factor("ж"),  
 V0\_DEM\_AGE = V0\_DEM\_AGE + 20  
 ) %>%  
 mutate\_at(vars(contains('NORM')), as.factor)  
head(data\_filtered)

## ID V0\_GRP V0\_DEM\_GEN V0\_DEM\_AGE V1\_TIDES\_AMOUNT V2\_TIDES\_AMOUNT V1\_CB\_WBC  
## 1 1 Мастодинон ж 55 2 4 4.3  
## 2 3 Мастодинон ж 47 5 4 6.5  
## 3 5 Мастодинон ж 55 5 8 5.8  
## 4 7 Мастодинон ж 45 6 6 6.4  
## 5 9 Мастодинон ж 45 5 0 6.6  
## 6 11 Мастодинон ж 51 6 4 4.7  
## V1\_CB\_RBC V1\_CB\_HGB V1\_CB\_HCT V1\_CB\_PLT V1\_CB\_NEUT. V1\_CB\_LYM. V1\_CB\_MON.  
## 1 4.1 131.8 0.40 269.7 5.4 1.3 0.1  
## 2 5.1 144.9 0.41 318.2 2.8 1.2 0.1  
## 3 4.8 147.1 0.44 295.0 3.1 1.7 0.1  
## 4 4.2 127.7 0.45 342.7 4.1 2.2 0.1  
## 5 4.5 137.7 0.43 291.4 2.3 1.3 0.1  
## 6 5.2 121.1 0.39 240.6 3.3 1.4 0.1  
## V1\_CB\_EO. V1\_CB\_BAS. V1\_NORM\_ECG V1\_NORM\_PHYS V1\_NORM\_VIT V2\_CB\_WBC V2\_CB\_RBC  
## 1 0.1 0 0 0 0 5.4 4.4  
## 2 0.1 0 0 0 0 7.3 4.0  
## 3 0.1 0 0 0 0 4.0 5.0  
## 4 0.1 0 0 0 0 7.7 5.5  
## 5 0.1 0 0 0 0 4.5 5.2  
## 6 0.1 0 0 0 0 4.7 4.4  
## V2\_CB\_HGB V2\_CB\_HCT V2\_CB\_PLT V2\_CB\_NEUT. V2\_CB\_LYM. V2\_CB\_MON. V2\_CB\_EO.  
## 1 126.0 0.46 302.1 3.3 1.7 0.1 0.1  
## 2 129.2 0.40 318.1 2.0 1.2 0.1 0.1  
## 3 122.9 0.42 290.8 2.2 1.7 0.1 0.1  
## 4 136.1 0.42 338.6 2.6 1.2 0.1 0.1  
## 5 124.4 0.45 314.4 3.1 1.9 0.1 0.1  
## 6 141.0 0.48 265.4 3.6 1.1 0.1 0.1  
## V2\_CB\_BAS. V2\_NORM\_ECG V2\_NORM\_PHYS V2\_NORM\_VIT  
## 1 0 0 0 0  
## 2 0 0 0 0  
## 3 0 0 0 0  
## 4 0 0 0 0  
## 5 0 0 0 0  
## 6 0 0 1 0

non\_unique <- names(data\_filtered)[sapply(data\_filtered, function(x) length(unique(x)) == 1)]  
  
cat('Колонки с одинаковыми значениями, которые не подлежат сравнению:', paste(non\_unique, collapse = ', '), '\n')

## Колонки с одинаковыми значениями, которые не подлежат сравнению: V0\_DEM\_GEN, V1\_CB\_MON., V1\_CB\_EO., V1\_NORM\_VIT, V2\_CB\_MON., V2\_CB\_EO., V2\_NORM\_ECG, V2\_NORM\_VIT

“CB\_MON.”, “CB\_EO.” не отличаются, проверка по ним не проводилась

Проведем тест Уилкоксона для зависимых переменных

data\_cleaned <- data\_filtered  
data\_control <- data\_cleaned %>%  
 filter(V0\_GRP == "Мастодинон")  
  
names <- c("TIDES\_AMOUNT", "CB\_WBC", "CB\_RBC", "CB\_HGB", "CB\_HCT",   
 "CB\_PLT", "CB\_NEUT.", "CB\_LYM.", "CB\_BAS.")  
  
  
# Создаем пустой датафрейм для сохранения результатов  
wilcox\_results\_df <- data.frame()  
  
for (name in names) {  
 name\_1 <- paste0("V1\_", name)  
 name\_2 <- paste0("V2\_", name)  
 # Проводим тест и сохраняем результаты в датафрейме  
 result <- data\_control %>%  
 select(ID, name\_1, name\_2) %>%  
 pivot\_longer(!ID) %>%  
 wilcox\_test(value ~ name, paired = TRUE, detailed = TRUE)   
   
 # Добавляем результаты к датафрейму  
 wilcox\_results\_df <- bind\_rows(wilcox\_results\_df, result)  
}

## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.  
## ℹ Please use `all\_of()` or `any\_of()` instead.  
## # Was:  
## data %>% select(name\_1)  
##   
## # Now:  
## data %>% select(all\_of(name\_1))  
##   
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.  
## ℹ Please use `all\_of()` or `any\_of()` instead.  
## # Was:  
## data %>% select(name\_2)  
##   
## # Now:  
## data %>% select(all\_of(name\_2))  
##   
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

# Выводим весь датафрейм с результатами тестов  
wilcox\_results\_df %>%   
 select(group1, statistic, p, conf.low, conf.high)

## group1 statistic p conf.low conf.high  
## 1 V1\_TIDES\_AMOUNT 159.5 0.796 -1.00005443 1.500039296  
## 2 V1\_CB\_WBC 154.5 0.909 -0.69996090 0.949999084  
## 3 V1\_CB\_RBC 137.5 0.509 -0.49995523 0.299932649  
## 4 V1\_CB\_HGB 141.0 0.394 -8.40000000 3.600000000  
## 5 V1\_CB\_HCT 115.5 0.502 -0.02003553 0.009944855  
## 6 V1\_CB\_PLT 227.0 0.199 -8.15000000 23.450000000  
## 7 V1\_CB\_NEUT. 173.0 0.520 -0.44993862 0.849999311  
## 8 V1\_CB\_LYM. 162.0 0.474 -0.15001693 0.349970475  
## 9 V1\_CB\_BAS. 1.0 1.000 NaN NaN

wilcox\_results\_df %>%   
 select(group1)

## group1  
## 1 V1\_TIDES\_AMOUNT  
## 2 V1\_CB\_WBC  
## 3 V1\_CB\_RBC  
## 4 V1\_CB\_HGB  
## 5 V1\_CB\_HCT  
## 6 V1\_CB\_PLT  
## 7 V1\_CB\_NEUT.  
## 8 V1\_CB\_LYM.  
## 9 V1\_CB\_BAS.

## Катигориальные переменные

#Сабсет с группой Мастодинона:   
data2 <- data\_cleaned %>%  
 filter(V0\_GRP == "Мастодинон")  
  
#ЭКГ в группе Мастодинона:  
sopr2 <- data2 %>%   
 select(V1\_NORM\_ECG, V2\_NORM\_ECG)  
  
sopr\_ispravl <- matrix(c(sum(sopr2[,1] == 0 & sopr2[,2] == 0), sum(sopr2[,1] == 0 & sopr2[,2] == 1), sum(sopr2[,1] == 1 & sopr2[,2] == 0), sum(sopr2[,1] == 1 & sopr2[,2] == 1)), nrow = 2, ncol = 2, byrow = TRUE, dimnames = list(c("V1\_ECG\_NORM=0", "V1\_ECG\_NORM=1"), c("V2\_ECG\_NORM=0", "V2\_ECG\_NORM=1"))) %>% print()

## V2\_ECG\_NORM=0 V2\_ECG\_NORM=1  
## V1\_ECG\_NORM=0 25 0  
## V1\_ECG\_NORM=1 1 0

mcn1 <- sopr\_ispravl %>%  
 mcnemar.test() %>%  
 tidy()  
  
#Физикальный осмотр в группе Мастодинона:  
sopr2 <- data2 %>%   
 select(V1\_NORM\_PHYS, V2\_NORM\_PHYS)  
  
sopr\_ispravl <- matrix(c(sum(sopr2[,1] == 0 & sopr2[,2] == 0), sum(sopr2[,1] == 0 & sopr2[,2] == 1), sum(sopr2[,1] == 1 & sopr2[,2] == 0), sum(sopr2[,1] == 1 & sopr2[,2] == 1)), nrow = 2, ncol = 2, byrow = TRUE, dimnames = list(c("V1\_ECG\_PHYS=0", "V1\_ECG\_PHYS=1"), c("V2\_ECG\_PHYS=0", "V2\_ECG\_PHYS=1"))) %>% print()

## V2\_ECG\_PHYS=0 V2\_ECG\_PHYS=1  
## V1\_ECG\_PHYS=0 23 2  
## V1\_ECG\_PHYS=1 1 0

mcn2 <- sopr\_ispravl %>%  
 mcnemar.test() %>%  
 tidy()  
  
#Витальные показатели осмотр в группе Мастодинона:  
sopr2 <- data2 %>%   
 select(V1\_NORM\_VIT, V2\_NORM\_VIT)  
  
sopr\_ispravl <- matrix(c(sum(sopr2[,1] == 0 & sopr2[,2] == 0), sum(sopr2[,1] == 0 & sopr2[,2] == 1), sum(sopr2[,1] == 1 & sopr2[,2] == 0), sum(sopr2[,1] == 1 & sopr2[,2] == 1)), nrow = 2, ncol = 2, byrow = TRUE, dimnames = list(c("V1\_ECG\_VIT=0", "V1\_ECG\_VIT=1"), c("V2\_ECG\_VIT=0", "V2\_ECG\_VIT=1"))) %>% print()

## V2\_ECG\_VIT=0 V2\_ECG\_VIT=1  
## V1\_ECG\_VIT=0 26 0  
## V1\_ECG\_VIT=1 0 0

mcn3 <- sopr\_ispravl %>%  
 mcnemar.test() %>%  
 tidy()  
  
#Сабсет с группой Плацебо:   
data3 <- data\_cleaned %>%  
 filter(V0\_GRP == "Плацебо")  
  
#ЭКГ в группе Плацебо:  
sopr3 <- data3 %>%   
 select(V1\_NORM\_ECG, V2\_NORM\_ECG)  
  
sopr\_ispravl <- matrix(c(sum(sopr3[,1] == 0 & sopr3[,2] == 0), sum(sopr3[,1] == 0 & sopr3[,2] == 1), sum(sopr3[,1] == 1 & sopr3[,2] == 0), sum(sopr3[,1] == 1 & sopr3[,2] == 1)), nrow = 2, ncol = 2, byrow = TRUE, dimnames = list(c("V1\_ECG\_NORM=0", "V1\_ECG\_NORM=1"), c("V2\_ECG\_NORM=0", "V2\_ECG\_NORM=1"))) %>% print()

## V2\_ECG\_NORM=0 V2\_ECG\_NORM=1  
## V1\_ECG\_NORM=0 25 0  
## V1\_ECG\_NORM=1 1 0

mcn4 <- sopr\_ispravl %>%  
 mcnemar.test() %>%  
 tidy()  
  
#Физикальный осмотр в группе Плацебо:  
sopr3 <- data3 %>%   
 select(V1\_NORM\_PHYS, V2\_NORM\_PHYS)  
  
sopr\_ispravl <- matrix(c(sum(sopr3[,1] == 0 & sopr3[,2] == 0), sum(sopr3[,1] == 0 & sopr3[,2] == 1), sum(sopr3[,1] == 1 & sopr3[,2] == 0), sum(sopr3[,1] == 1 & sopr3[,2] == 1)), nrow = 2, ncol = 2, byrow = TRUE, dimnames = list(c("V1\_ECG\_PHYS=0", "V1\_ECG\_PHYS=1"), c("V2\_ECG\_PHYS=0", "V2\_ECG\_PHYS=1"))) %>% print()

## V2\_ECG\_PHYS=0 V2\_ECG\_PHYS=1  
## V1\_ECG\_PHYS=0 23 3  
## V1\_ECG\_PHYS=1 0 0

mcn5 <- sopr\_ispravl %>%  
 mcnemar.test() %>%  
 tidy()  
  
#Витальные показатели осмотр в группе Плацебо:  
sopr3 <- data3 %>%   
 select(V1\_NORM\_VIT, V2\_NORM\_VIT)  
  
sopr\_ispravl <- matrix(c(sum(sopr3[,1] == 0 & sopr3[,2] == 0), sum(sopr3[,1] == 0 & sopr3[,2] == 1), sum(sopr3[,1] == 1 & sopr3[,2] == 0), sum(sopr3[,1] == 1 & sopr3[,2] == 1)), nrow = 2, ncol = 2, byrow = TRUE, dimnames = list(c("V1\_ECG\_VIT=0", "V1\_ECG\_VIT=1"), c("V2\_ECG\_VIT=0", "V2\_ECG\_VIT=1"))) %>% print()

## V2\_ECG\_VIT=0 V2\_ECG\_VIT=1  
## V1\_ECG\_VIT=0 26 0  
## V1\_ECG\_VIT=1 0 0

mcn6 <- sopr\_ispravl %>%  
 mcnemar.test() %>%  
 tidy()  
  
# Создайте список из ваших датафреймов  
df\_list <- list(mcn1, mcn2, mcn3, mcn4, mcn5, mcn6)  
  
# Объедините датафреймы по строкам  
combined\_df <- bind\_rows(df\_list)  
  
# Посмотрите на объединенный датафрейм  
print(combined\_df)

## # A tibble: 6 × 4  
## statistic p.value parameter method   
## <dbl> <dbl> <dbl> <chr>   
## 1 0 1 1 McNemar's Chi-squared test with continuity correc…  
## 2 0 1 1 McNemar's Chi-squared test with continuity correc…  
## 3 NaN NaN 1 McNemar's Chi-squared test   
## 4 0 1 1 McNemar's Chi-squared test with continuity correc…  
## 5 1.33 0.248 1 McNemar's Chi-squared test with continuity correc…  
## 6 NaN NaN 1 McNemar's Chi-squared test