

RBS-R Psychometric Analysis: Reproducible Report

Contents

1	Settings	2
1.1	Clear working environment	2
1.2	Set random seed	2
1.3	Import and inspect data attributes	2
2	Empirical statistical power estimation	3
2.1	Define Variables and parameters	3
2.2	Power Analysis for Detecting a Location Shift	4

1 Settings

1.1 Clear working environment

```
# Clear R environment  
rm(list = ls())
```

1.2 Set random seed

```
set.seed(123)
```

1.3 Import and inspect data attributes

```
# Read the data file  
migraine_df <- read.csv("migraine_screening_dataset.csv")  
  
attributes(migraine_df) [names(attributes(migraine_df)) != "row.names"]  
  
## $names  
## [1] "ID"                 "m4"                 "msq"                "ichd3b"  
## [5] "migraine_status"    "gh"                 "hc"                 "pf"  
## [9] "rf_p"                "rf_e"                "sf"                 "bp"  
## [13] "ef"                  "ew"  
##  
## $class  
## [1] "data.frame"
```

2 Empirical statistical power estimation

2.1 Define Variables and parameters

```
# Outcome and grouping variables
outcomes <- c("gh", "hc", "pf", "rf_p", "rf_e", "sf", "bp", "ef", "ew")
group_vars <- c("m4", "msq", "ichd3b", "migraine_status")

# Parameters
Nsim <- 5000      # Number of simulations
alpha <- 0.05       # Significance level
shift <- 5          # Location shift of interest under the alternative
```

2.2 Power Analysis for Detecting a Location Shift

```
estimate_empirical_power <- function(data, outcome, group_col,
                                      shift = 5, Nsim = 5000, alpha = 0.05) {
  # Identify the two comparison groups
  if (group_col == "migraine_status") {

    # Exclude "Negative" rows, then store the remaining 2 levels
    data <- subset(data, migraine_status != "Negative")
    groups <- unique(data[[group_col]])

  } else {

    # For other grouping variables directly store the 2 levels
    groups <- c("Negative", "Positive")

  }

  # Make sure that no other grouping having >2 levels
  if (length(groups) != 2) {
    stop(sprintf(
      "Grouping variable '%s' must have exactly 2 levels after filtering; found %d.",
      group_col, length(groups)
    ))
  }

  g1 <- groups[1]
  g2 <- groups[2]

  # Pull outcome values for each group
  group1 <- data[data[[group_col]] == g1, outcome]
  group2 <- data[data[[group_col]] == g2, outcome]

  n1 <- length(group1)
  n2 <- length(group2)

  # Compute Wilcoxon W statistic for two samples
  wilcox_W <- function(x, y) {
    wilcox.test(x, y, alternative = "two.sided", exact = FALSE)$statistic
  }

  # Null distribution via bootstrap (resample within each group)
  null_W <- replicate(Nsim, {
    samp1 <- sample(group1, size = n1, replace = TRUE)
    samp2 <- sample(group2, size = n2, replace = TRUE)
    wilcox_W(samp1, samp2)
  })

  crit_low <- quantile(null_W, probs = alpha / 2)
  crit_high <- quantile(null_W, probs = 1 - alpha / 2)

  # Alternative distribution: shift group2 by +shift, then test
  alt_reject <- replicate(Nsim, {
```

```

samp1 <- sample(group1, size = n1, replace = TRUE)
samp2 <- sample(group2, size = n2, replace = TRUE) + shift

W <- wilcox_W(samp1, samp2)
(W <= crit_low) || (W >= crit_high)
})

power <- mean(alt_reject)

# Monte Carlo CI for the estimated power
se <- sqrt(power * (1 - power) / Nsim)
ci_low <- max(0, power - 1.96 * se)
ci_high <- min(1, power + 1.96 * se)

# Return results
data.frame(Outcome      = outcome,
           Grouping     = group_col,
           Level1       = g1,
           Level2       = g2,
           Power        = power,
           CI_low       = ci_low,
           CI_high      = ci_high
         )
}

```

```

# Run power estimation across all outcomes and grouping variables
for (outcome in outcomes) {

  outcome_results <- lapply(group_vars, function(group_col) {
    estimate_empirical_power(data      = migraine_df,
                              outcome   = outcome,
                              group_col = group_col,
                              shift     = shift,
                              Nsim      = Nsim,
                              alpha     = alpha
    )
  })

  outcome_results <- do.call(rbind, outcome_results)
  print(outcome_results)
}

```

```

##   Outcome      Grouping      Level1      Level2 Power
## 1   gh          m4        Negative Positive 0.9958
## 2   gh          msq       Negative Positive 0.9972
## 3   gh          ichd3b    Negative Positive 0.9856
## 4   gh migraine_status screening positive criterion positive 0.9784
##   CI_low   CI_high
## 1 0.9940074 0.9975926
## 2 0.9957353 0.9986647
## 3 0.9822978 0.9889022
## 4 0.9743705 0.9824295
##   Outcome      Grouping      Level1      Level2 Power CI_low
## 1   hc          m4        Negative Positive 1       1
## 2   hc          msq       Negative Positive 1       1
## 3   hc          ichd3b    Negative Positive 1       1
## 4   hc migraine_status screening positive criterion positive 1       1
##   CI_high
## 1   1
## 2   1
## 3   1
## 4   1
##   Outcome      Grouping      Level1      Level2 Power
## 1   pf          m4        Negative Positive 0.8638
## 2   pf          msq       Negative Positive 0.8860
## 3   pf          ichd3b    Negative Positive 0.7500
## 4   pf migraine_status screening positive criterion positive 0.6808
##   CI_low   CI_high
## 1 0.8542925 0.8733075
## 2 0.8771907 0.8948093
## 3 0.7379975 0.7620025
## 4 0.6678785 0.6937215
##   Outcome      Grouping      Level1      Level2 Power CI_low
## 1   rf_p        m4        Negative Positive 1       1
## 2   rf_p        msq       Negative Positive 1       1
## 3   rf_p        ichd3b    Negative Positive 1       1
## 4   rf_p migraine_status screening positive criterion positive 1       1
##   CI_high

```

```

## 1      1
## 2      1
## 3      1
## 4      1
##   Outcome      Grouping      Level1      Level2 Power CI_low
## 1   rf_e        m4        Negative    Positive    1     1
## 2   rf_e        msq       Negative    Positive    1     1
## 3   rf_e        ichd3b    Negative    Positive    1     1
## 4   rf_e migraine_status screening positive criterion positive    1     1
##   CI_high
## 1      1
## 2      1
## 3      1
## 4      1
##   Outcome      Grouping      Level1      Level2 Power
## 1   sf          m4        Negative    Positive 0.9694
## 2   sf          msq       Negative    Positive 0.9726
## 3   sf          ichd3b    Negative    Positive 0.9532
## 4   sf migraine_status screening positive criterion positive 0.9376
##   CI_low  CI_high
## 1 0.9646260 0.9741740
## 2 0.9680751 0.9771249
## 3 0.9473456 0.9590544
## 4 0.9308954 0.9443046
##   Outcome      Grouping      Level1      Level2 Power
## 1   bp          m4        Negative    Positive 0.9530
## 2   bp          msq       Negative    Positive 0.9446
## 3   bp          ichd3b    Negative    Positive 0.9128
## 4   bp migraine_status screening positive criterion positive 0.9012
##   CI_low  CI_high
## 1 0.9471337 0.9588663
## 2 0.9382591 0.9509409
## 3 0.9049798 0.9206202
## 4 0.8929290 0.9094710
##   Outcome      Grouping      Level1      Level2 Power
## 1   ef          m4        Negative    Positive 0.9902
## 2   ef          msq       Negative    Positive 0.9914
## 3   ef          ichd3b    Negative    Positive 0.9680
## 4   ef migraine_status screening positive criterion positive 0.9478
##   CI_low  CI_high
## 1 0.9874695 0.9929305
## 2 0.9888406 0.9939594
## 3 0.9631215 0.9728785
## 4 0.9416346 0.9539654
##   Outcome      Grouping      Level1      Level2 Power
## 1   ew          m4        Negative    Positive 0.9952
## 2   ew          msq       Negative    Positive 0.9978
## 3   ew          ichd3b    Negative    Positive 0.9924
## 4   ew migraine_status screening positive criterion positive 0.9854
##   CI_low  CI_high
## 1 0.9932842 0.9971158
## 2 0.9965013 0.9990987
## 3 0.9899927 0.9948073
## 4 0.9820753 0.9887247

```