

```

### Teste de Wilcoxon para uma amostra
### Exemplos em que teta0 é diferente de xi, i=1,...,n.

## 1. Dados
# Tabela 3.11, p. 83, em Hollander & Wolf (1999, 2nd ed.)
# H1 unilateral à direita
x <- c(254, 171, 345, 134, 190, 447, 106, 173, 449, 198)
n <- length(x)
cat("\n Tamanho da amostra:", n, "\n")
teta0 <- 175

z <- x - teta0
r <- rank(abs(z))
rs <- ifelse(z > 0, r, -r)
cat("\n Postos com sinais: \n", rs, "\n")
smais <- sum(r[z > 0])
cat("\n S+ = ", smais, "\n")

# Dist. exata (default quando não há xi - teta0 = 0 e sem empates)
(wilcox.test(x, mu = teta0, alternative = "greater", conf.int =
TRUE))

# Valor-p com a dist. de S+
cat("\n S+ = ", smais, "(p =", psignrank(smais - 1, n, lower.tail =
FALSE), ")")

# Dist. aproximada com correção de continuidade
e0 <- n * (n + 1) / 4
var0 <- n * (n + 1) * (2 * n + 1) / 24
spad <- (smais - e0 - 0.5) / sqrt(var0)
cat("\n Correção de continuidade \n S+ padronizada =", spad, "(p =",
pnorm(spad, lower.tail = FALSE), ")")

(wilcox.test(x, mu = teta0, alternative = "greater", exact = FALSE,
correct = TRUE))

(wilcox.test(x, mu = teta0, alternative = "greater", exact = FALSE,
correct = FALSE))

## 2. Dados
# Tabela 3.9, p. 82, em Hollander & Wolf (1999, 2nd ed.)
# H1 bilateral
x <- c(17.4, 17.9, 17.6, 18.1, 17.6, 18.9, 16.9, 17.5, 17.8, 17.4,
24.6, 26.0)
n <- length(x)
cat("\n Tamanho da amostra:", n, "\n")
teta0 <- 18

z <- x - teta0
r <- rank(abs(z))
rs <- ifelse(z > 0, r, -r)
cat("\n Postos com sinais: \n", rs, "\n")
smais <- sum(r[z > 0])
cat("\n S+ = ", smais, "\n")

```

```

# Dist. aproximada com correção de continuidade
e0 <- n * (n + 1) / 4
# Correção em var0 (Hollander & Wolf (1999, 2nd ed., p. 38)
tj <- as.numeric(table(rs))
correcao <- 0.5 * sum(tj * (tj - 1) * (tj + 1))
var0 <- (n * (n + 1) * (2 * n + 1) - correcao) / 24
spad <- (smais - e0 - 0.5) / sqrt(var0)
p1 <- pnorm(spad, lower.tail = FALSE)
spad <- (smais - e0 + 0.5) / sqrt(var0)
p2 <- pnorm(spad)
cat("\n Correção de continuidade \n (p =", 2 * min(p1, p2), ")")

# Default com empates: H1 bilateral e correção de continuidade
(wilcox.test(x, mu = teta0, alternative = "two.sided", conf.int =
TRUE))

(wilcox.test(x, mu = teta0, correct = FALSE, conf.int = TRUE))

## Exercício. Utilizando os dados do exemplo 2, efetue o teste com
teta0 = 18,1.

```