Compute Shapiro-Wilk statistics

Alex Koh

2024-07-26

The program calculates and returns the test statistics W used in a Shapiro–Wilk test used to determine if the sample is normally distributed

Github link: Github

```r
#' Compute Shapiro-Wilk Test Statistic
#'
#' This function calculates the Shapiro-Wilk W statistic to determine
#' if a sample is normally distributed.
#'
#' @param data A numeric vector of data.
#' @param plot_qq A logical value indicating whether to plot the QQ plot. Default is FALSE.
#'
#' @return The Shapiro-Wilk W statistic.
#' @export
#'
#' @examples
#' set.seed(42)
#' data <- rnorm(1000)
#' compute_shapiro_wilk_test(data, plot_qq = TRUE)

compute_shapiro_wilk_test <- function(data, plot_qq = FALSE) {
    #input validation
    if (!is.vector(data, mode = "numeric")) stop("Data must be a numeric vector.")
    if (any(is.na(data))) stop("Data contains NA values.")
    if (any(is.infinite(data))) stop("Data contains infinite values.")
    if (length(data) < 3) stop("Data must contain at least 3 observations.")
    if (!is.logical(plot_qq)) stop("plot_qq argument must be of type logical.")

    n <- length(data)
    sorted_data <- sort(data)

    # Calculate mean and standard deviation of the data
    mean_data <- mean(sorted_data)
    sd_data <- sd(sorted_data)

    # Calculate expected normal order statistics
    m <- qnorm((1:n - 0.375) / (n + 0.25))

    # Calculate weights
    c <- cov(sorted_data, m)
    a <- c / sqrt(sum(c^2))
```

```
    # Calculate Shapiro-Wilk W statistic
    W <- (sum(a * sorted_data) ^ 2) / sum((sorted_data - mean_data) ^ 2)

    if (plot_qq) {
        qqnorm(data, main="QQ Plot")
        qqline(data, col = "blue")
    }

    return(W)
}
# Example usage
set.seed(42)  # Set seed for reproducibility
data <- rnorm(1000)
compute_shapiro_wilk_test(data, plot_qq = TRUE)
```
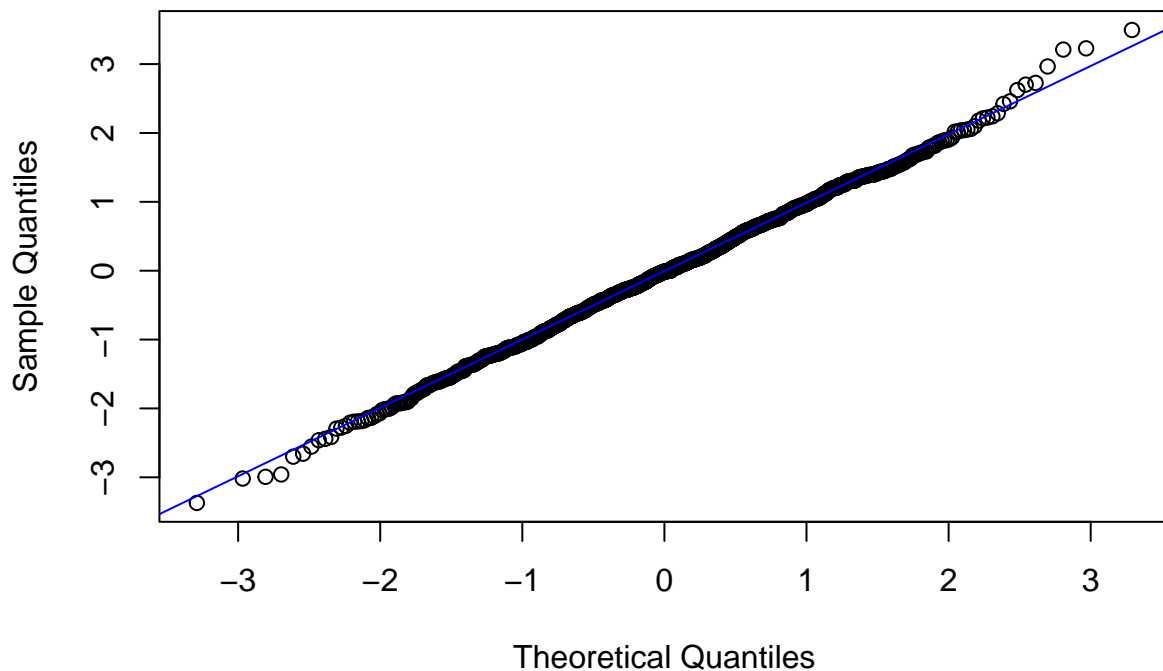
QQ Plot


```
## [1] 0.6642151
```

Unit tests

To ensure the function works correctly, several test cases have been implemented using the `testthat` package.

Test 1: Handling incorrect input format

This test checks if the function correctly handles inputs that are not in the expected format, such as a data frame.

```r
test_that("compute_shapiro_wilk_test() handles wrong format (data frame) as input", {
  invalid_data <- data.frame(a = 1:10)
  expect_error(compute_shapiro_wilk_test(invalid_data), "Data must be a numeric vector.")
})
```

```
## Test passed
```

Test 2: Handling invalid optional argument

This test checks if the function correctly handles invalid values for the optional `plot_qq` argument.

```r
test_that("compute_shapiro_wilk_test() handles invalid optional argument", {
  valid_data <- c(5,6,7)
  invalid_arg <- "x"
  expect_error(compute_shapiro_wilk_test(valid_data, plot_qq = invalid_arg),
               "plot_qq argument must be of type logical.")
})
```

```
## Test passed
```