HW2_Vancraeynest_Vanhauwe

Toon Vancraeynest, Jonas Vanhauwe

2024-11-11

Question 1: Construction of median test based on permutation strategy

We consider the following hypotheses:

$$H_0 : F_1 = F_2$$

against

$$H_A : median_1 \neq median_2$$

The sample median was chosen as the test-statistic. As required, the test follows a permutation strategy to determine the p-value. The code for this test calculates the exact permutation null distribution if the number of permutations is less than 500. Otherwise, the permutation null distribution is approximated:

```r
median.test.two_sided <- function(vec1, vec2, N=1000, exact=FALSE) {

  observed_test_statistic <- abs(median(vec1) - median(vec2))


  full_vec <- c(vec1, vec2)

  # Check if the number of combinations is less than N - default N = 1000
  if (choose(length(full_vec), length(vec1)) < N | exact) {

    # Generate the null distribution by exact combinations
    t_star <- combn(length(full_vec), length(vec1), function(x) {
      group1 <- full_vec[x]
      group2 <- full_vec[-x]  # all elements not in x
      test_stats <- abs(median(group1) - median(group2))
      test_stats
    })
  } else {


    # Approximate null distribution by permutation if combinations exceed N
    Groupvec <- c(rep(1, length(vec1)),
                  rep(2, length(vec2)))  # Create group vector

    perm_data <- data.frame(value = full_vec,
                            group = Groupvec)  # Create data frame for permutations
    t_star <- replicate(N, {
      # Shuffle the group labels:
      perm_data$group <- sample(perm_data$group)
      # Calculate the test statistic for shuffled groups:
      test_stats <- abs(median(perm_data$value[perm_data$group == 1])
                        - median(perm_data$value[perm_data$group == 2]))
      test_stats
    })
  }

  # Calculate the p-value
  p_value <- mean(t_star >= observed_test_statistic)

  p_value
}
```

Question 2: Power simulation set-up

Following the required set-up, we compare the power of our newly developed median test based on a permutation strategy with the permutation t-test and the Wilcoxon-Mann-Whitney test using simulations. The power can be expressed as $P(\text{Reject } H_0 \mid H_a \text{ is true})$. Therefore, we can count the proportion of times that the test rejects H_0. In other words, we set up the simulation under H_a and count the proportion of times that the p-value is smaller than the significance level $\alpha = 0.05$. Note that in this comparison, we assume that the location shift model holds, since in this case, there is a relationship between the alternative hypothesis of the Mann-Whitney-U test and the (permutation) t-test. In Appendix 1, we show the Q-Q Plots used to verify whether this assumption holds for all distributions considered when the sample size is 20. If the data are parallel to the diagonal, the location shift model holds, because this means that the quantile function of the first and second group are the same in shape, but shifted with delta.

The comparison of the Mann-Whitney-U test, the permutation t-test and the developed median test is especially important for symmetrical distributions, because for symmetrical distributions, the mean equals the median, meaning that we have the following relationships:

$$P(X_1 < X_2) = \frac{1}{2} \quad \Leftrightarrow \quad \mu_1 = \mu_2 \quad \Leftrightarrow \quad median_1 = median_2$$

$$P(X_1 < X_2) > \frac{1}{2} \quad \Leftrightarrow \quad \mu_1 < \mu_2 \quad \Leftrightarrow \quad median_1 < median_2$$

$$P(X_1 < X_2) < \frac{1}{2} \quad \Leftrightarrow \quad \mu_1 > \mu_2 \quad \Leftrightarrow \quad median_1 > median_2$$

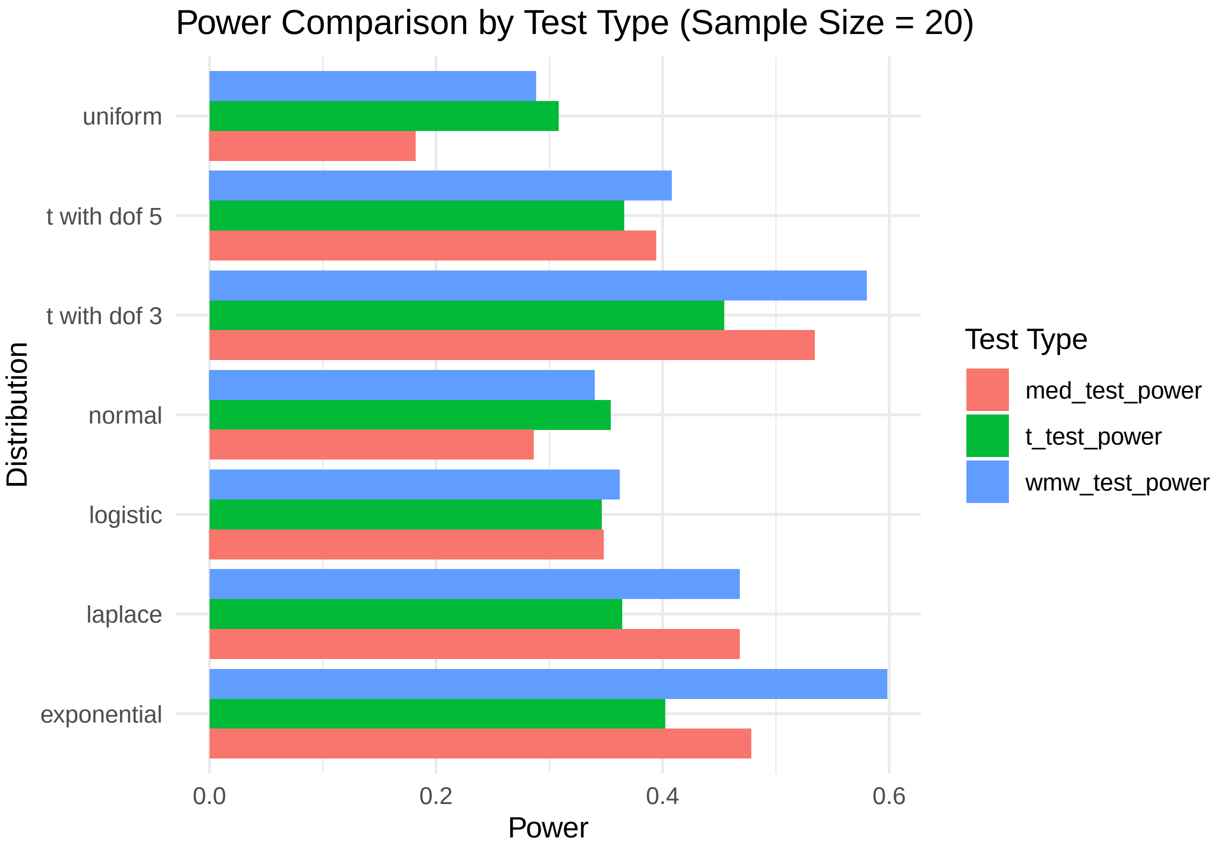
By running this simulation for 8 distributions and 5 sample sizes (5, 10, 20, 50 and 100), we obtain a comprehensive view on the performance of the tests. The delta was always chosen as proposed in the course notes: $\delta = \frac{\sqrt{\text{variance}}}{2}$ This is a good choice because it did not result in a power of 0% or 100% for the distributions and sample sizes considered.

From the QQ-Plots in Appendix 1, we infer that the location shift model does clearly not hold for the uniform distribution. We will take this into account in the remainder of our analysis when drawing conclusions. For the Logistic distribution and the Exponential distribution, the location shift relationship is not as explicit as for the other distributions - t3, laplace, t5 and normal - but we cannot clearly state that the location shift model does not hold.

To contain the required computation time, we chose to run 500 simulations per combination of distribution and sample size. The results are summarized in Appendix 2. Generally, we observe that for all tests, the power increases substantially with the sample size, which is expected. Further discussion will be provided in answering Question 3. Finally, we present the code used for the simulation of the t-distribution with 3 degrees of freedom at sample size 20. The code for all simulations can be found in the R-File attached.

```r
set.seed(1234) #For reproductibility
nsim <- 500
p.t <- p.wmw <- p.med <- numeric(nsim) #empty vector of p values for each test
n <- 20 # n observations in each group
#variance t-distribution = dof/(dof-2)
#with dof the degrees of freedom
delta <- sqrt(3)/2 #delta = sqrt(variance)/2
for(i in 1:nsim)
{
  Y1 <- rt(n, 3)
  Y2 <- rt(n, 3) + delta
  X <- factor(c(rep("A",n),rep("B",n)))
  Y <- c(Y1, Y2)
  #permutation t-test
  p.t[i] <- pvalue(oneway_test(Y ~ X,
                               distribution=approximate(nresample=1000)))
  #Wilcoxon-Mann-Whitney test
  p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
  #median test
  p.med[i] <- median.test.two_sided(Y1, Y2)
}
# Monte-Carlo approximation of the power
# for alpha = .05
results <- data.frame(n = n,
  t_test_power = mean(p.t < 0.05),
  wmw_test_power = mean(p.wmw < 0.05),
  med_test_power = mean(p.med < 0.05)
)
```

Question 3: Power tests for different distributions



At a sample size of 20, we observe that the Wilcoxon-Mann-Whitney test and our test based on the medians have a similar performance when comparing the power of the tests. When the data follows a normal distribution, a logistic distribution or a t distribution with 5 degrees of freedom, there is little that separates the three tests in terms of power with a sample size of 20. For the Laplace distribution, Exponential distribution and t distribution with 3 degrees of freedom, the permutation t-test has a substantially lower power with a sample size of 20. One notable observation is for the power of the uniform distribution. It seems that the test based on medians is not well suited for this distribution. However, for the uniform distribution, we want to refrain from making conclusions as the location shift model does not hold.

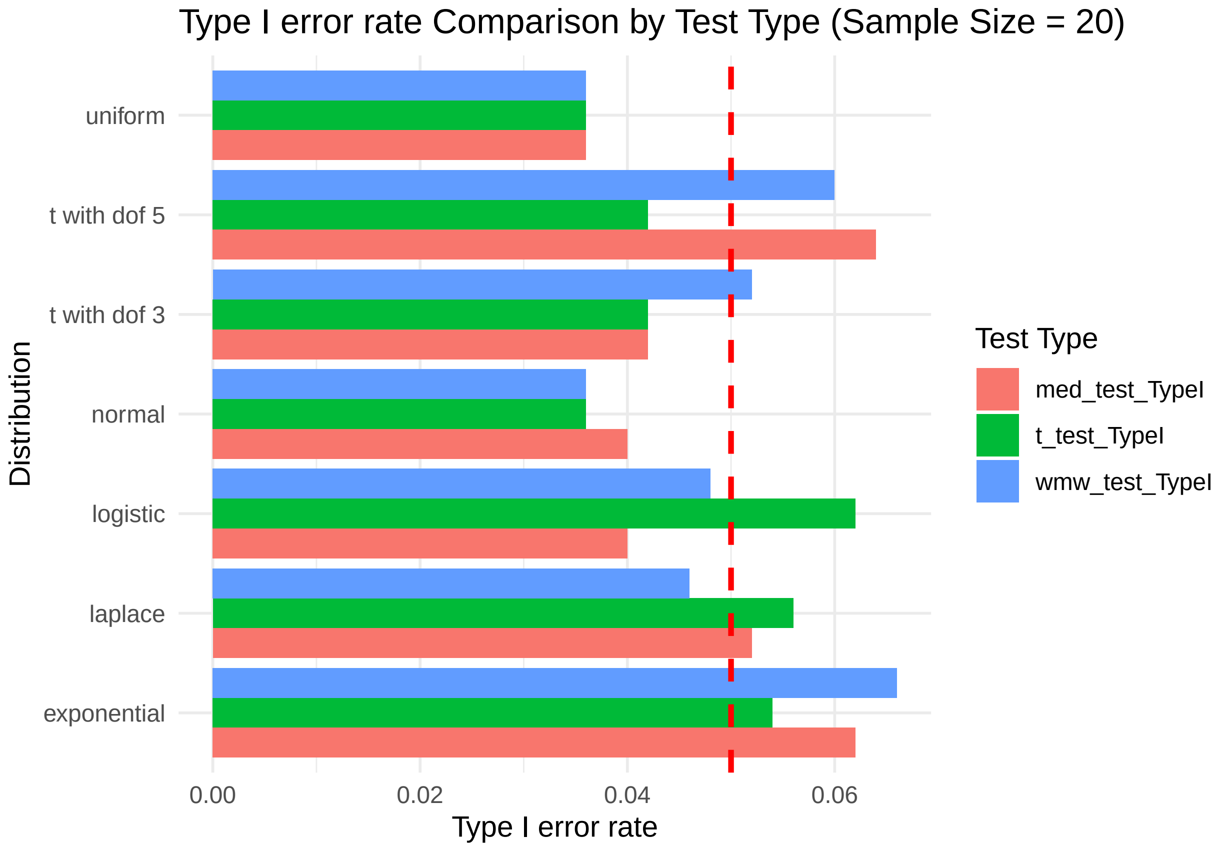
Interestingly, if we increase the sample size to 100, the Wilcoxon-Mann-Whitney test and the permutation t-test significantly outperform the median test for the normal distribution. This effect is not seen for other distributions.

Question 4: Type I error rate of tests for different distributions

A type I error is defined as the error where H_0 is rejected, given that H_0 is in fact true. The probability of making a type I error rate is controlled by the researcher at the significance level. In other words, $P(\text{Reject } H_0 \mid H_a \text{ is true}) = \alpha$. We can verify whether the Type I error rate is controlled by simulating under the null hypothesis that both groups are drawn from the same distribution:

$$H_0 : F_1 = F_2$$

After drawing two groups from the same distribution, thereby working under the assumption that H_0 holds, we can calculate the Type I error rate as the proportion of times H_0 is rejected in a simulation study. We perform this simulation study for the same distributions and the same sample sizes as the power simulation.



We can see that the type I error rate is for all tests approximately controlled at $\alpha = 0.05$. We see that the test based on medians is less conservative than the Wilcoxon-Mann-Whitney test and the permutation t test, except for the case of the uniform distribution. In the case of the logistic and exponential distribution the type I error rate for the test based on medians is slightly bigger than the desired 0.05.

Question 5: Recommendation on the use of the median test

Throughout this report, we observed that the (permutation) median test is a valid and powerful alternative to the Wilcoxon-Mann-Whitney test and the permutation t test.

Especially compared to the permutation t-test, we observed that the median test is more powerful for more heavy-tailed distributions, such as the t distribution at 3 degrees of freedom and the laplace distribution. This should not be surprising: the test statistic in the t-test is based on the difference in sample averages. The sample average, however, is not robust to outliers, leading to more variability in the test-statistic. Furthermore, for heavy-tailed distributions, the probability of randomly drawing outliers is higher. As a consequence, the power of the test will intuitively be lower for the same sample size compared to a test statistic based on the difference in sample medians for a heavy tailed distribution. This is because the median is robust to outliers.

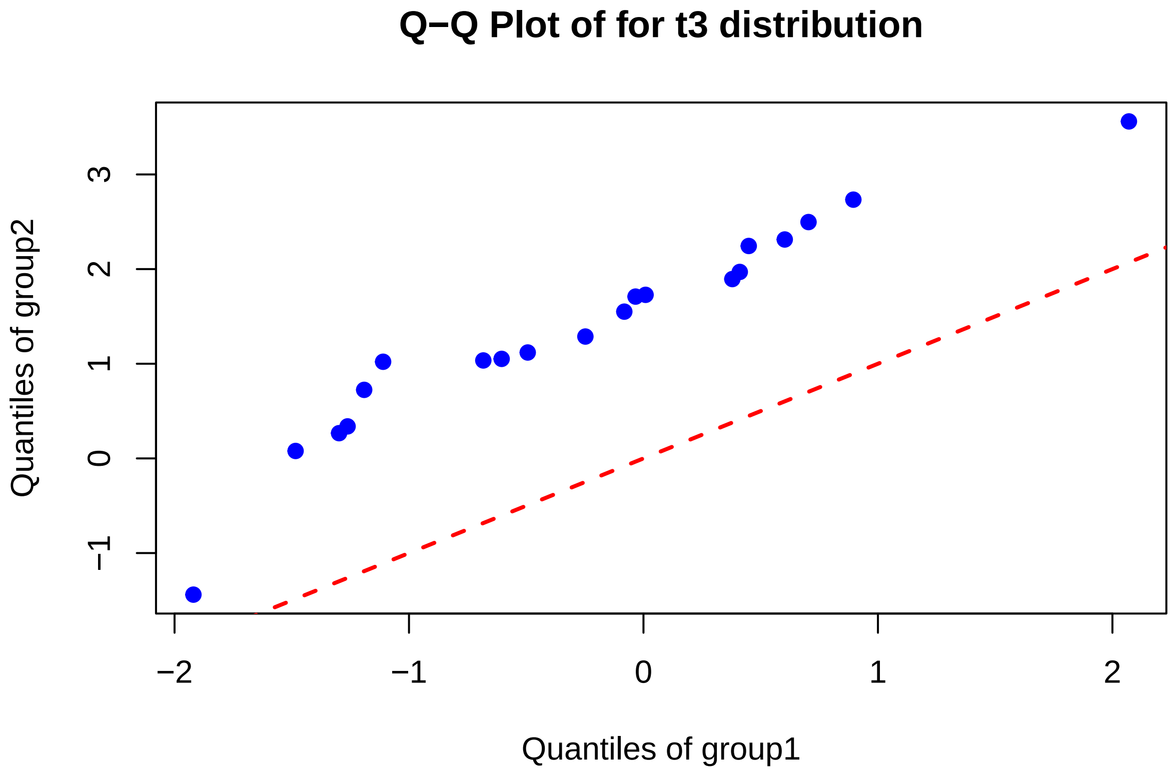
The same benefit described above applies to the Wilcoxon-Mann-Whitney test. However, the Wilcoxon-Mann-Whitney test may not provide good results in the case of many ties. Therefore, the median test based on a permutation strategy is recommended if the data consists of many ties in a heavy tailed distribution.

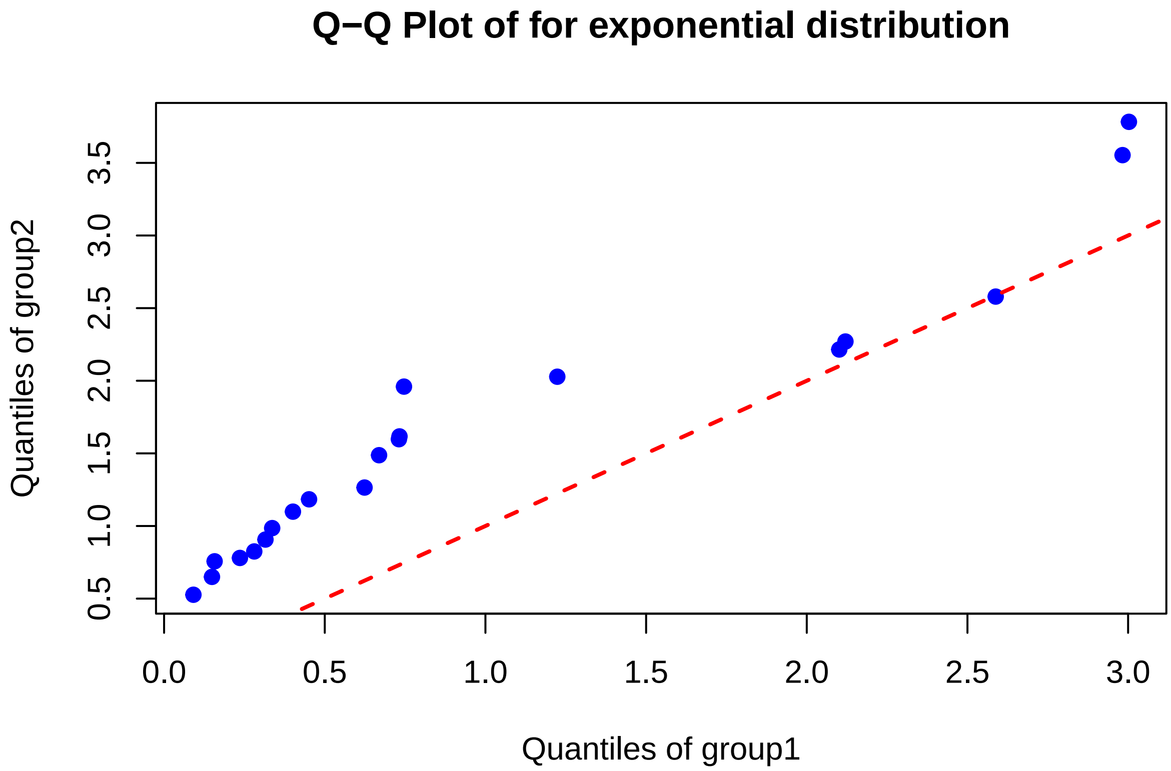
Question 6: Division of work

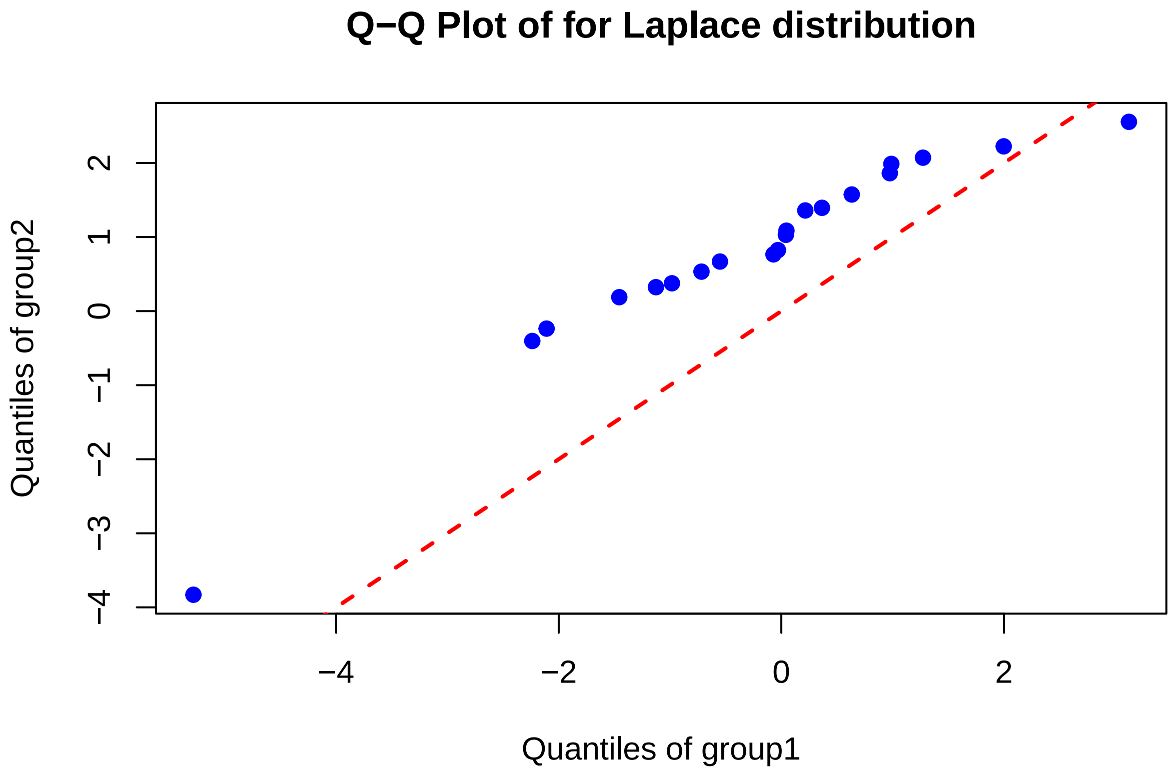
We separately constructed the median test to make sure that it is as correct as possible, since it is the cornerstone of this homework. We then discussed and merged our solutions into one. Jonas wrote the code for the simulations per distribution, Toon extended them to include all sample sizes. The interpretation of the results was always a joint effort. For every code written by the other person, we both verified its correctness.

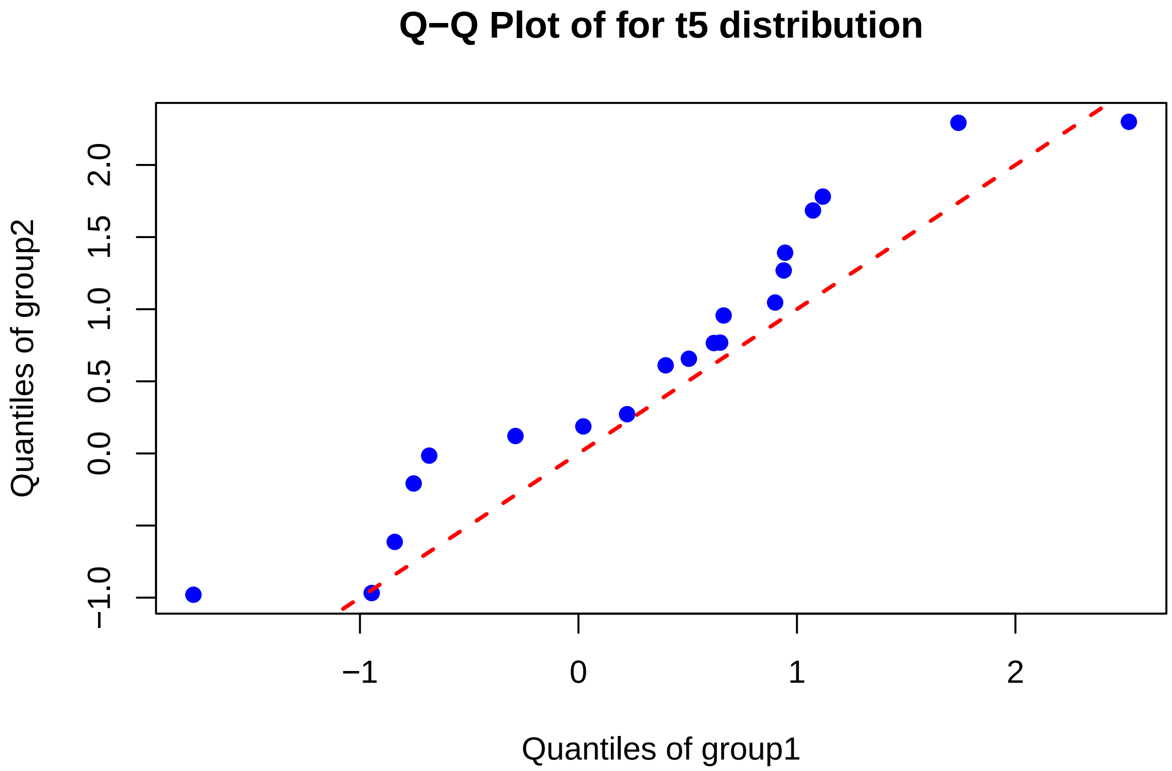
Appendix 1

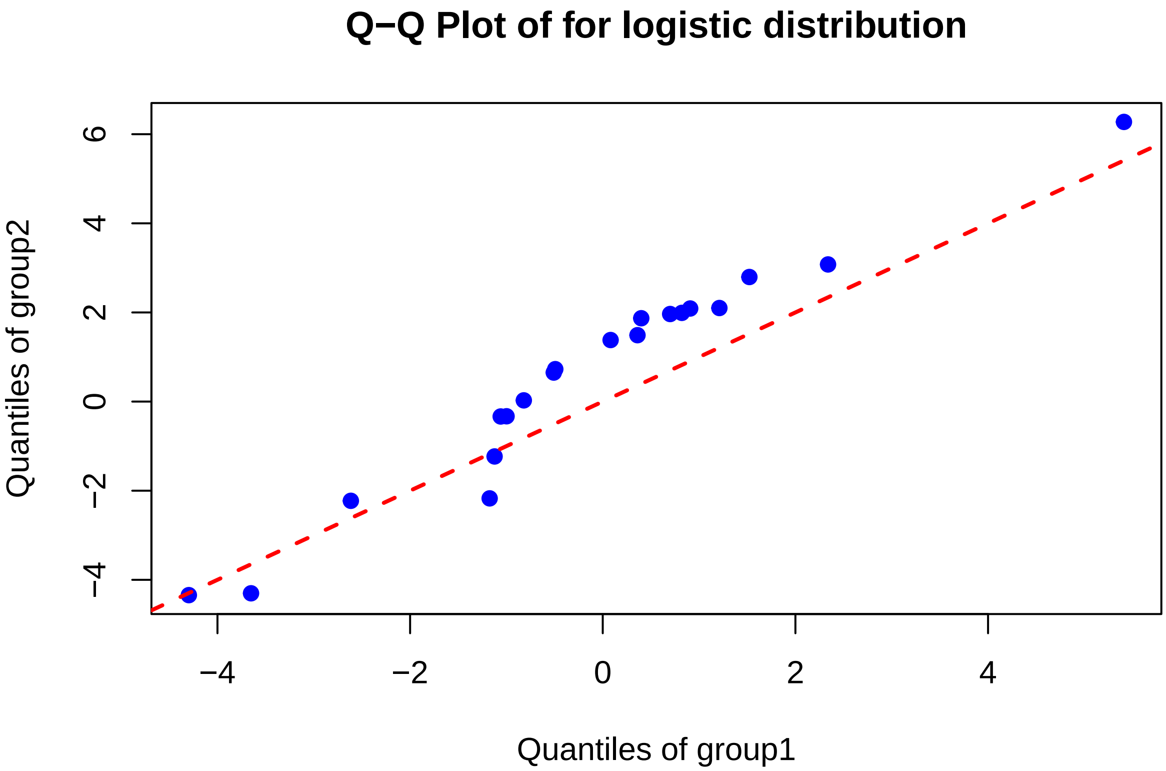
In this appendix, we present the Q-Q Plots of the first and second group for each considered distribution to verify whether the location-shift model holds.

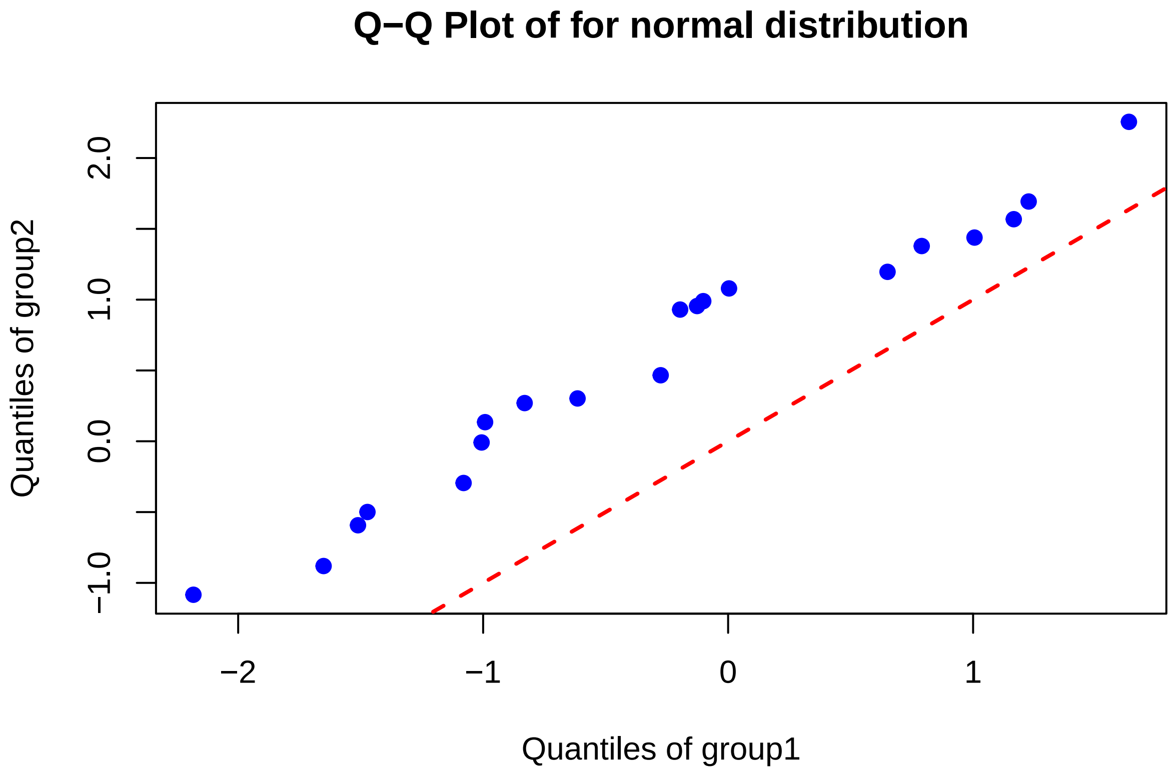


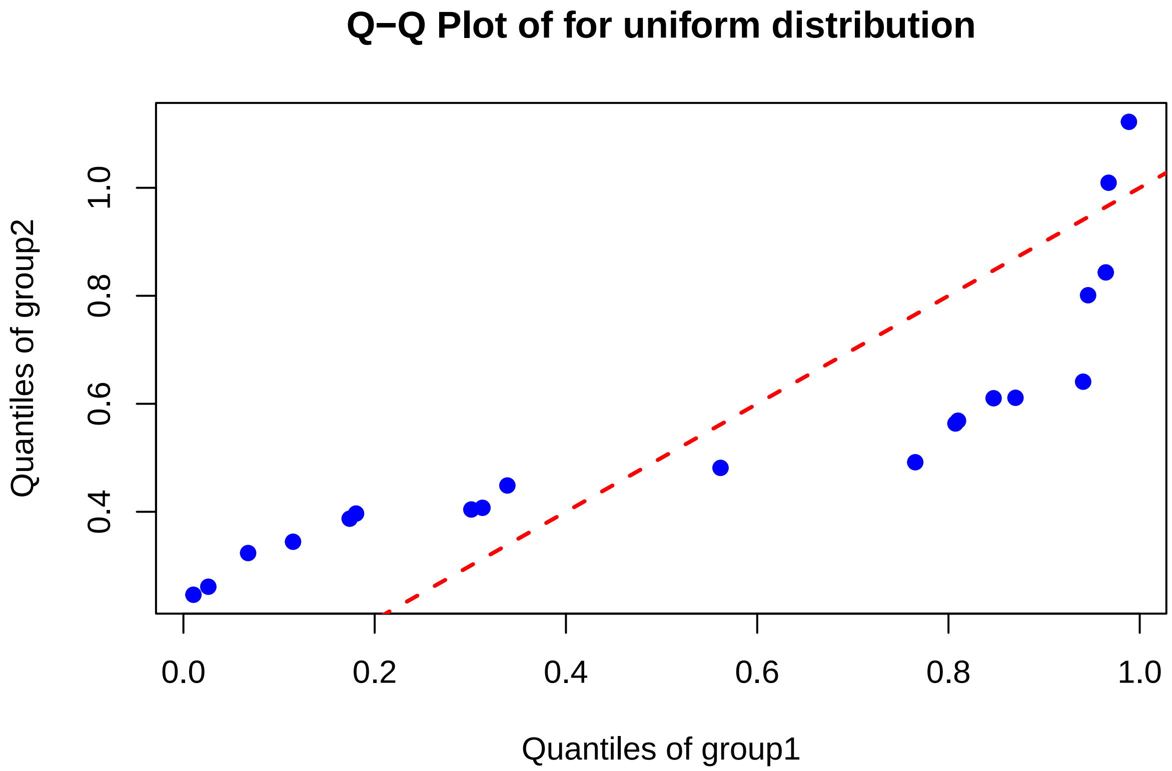

Q−Q Plot of for exponential distribution
Quantiles of group2
0.5
1.0
1.5
2.0
2.5
3.0
3.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Quantiles of group1


Q−Q Plot of for Laplace distribution
Quantiles of group2
2
1
0
−1
−2
−3
−4
−4
−2
0
2
Quantiles of group1


Q−Q Plot of for t5 distribution
Quantiles of group2
−1.0
0.0
0.5
1.0
1.5
2.0
−1
0
1
2
Quantiles of group1


Q−Q Plot of for logistic distribution
Quantiles of group2
6
4
2
0
−2
−4
−4
−2
0
2
4
Quantiles of group1


Q−Q Plot of for normal distribution
Quantiles of group2
Quantiles of group1
2.0
1.0
0.0
−1.0
−2
−1
0
1



Appendix 2

In this appendix, we present the results of the power simulation.

```
##    sample_size t_test_power wmw_test_power med_test_power distribution
## 1            5        0.094          0.070          0.104      uniform
## 2           10        0.160          0.148          0.134      uniform
## 3           20        0.308          0.288          0.182      uniform
## 4           50        0.706          0.666          0.364      uniform
## 5          100        0.954          0.922          0.558      uniform
## 6            5        0.146          0.114          0.152  exponential
## 7           10        0.244          0.296          0.264  exponential
## 8           20        0.402          0.598          0.478  exponential
## 9           50        0.722          0.934          0.784  exponential
## 10         100        0.942          0.998          0.962  exponential
## 11           5        0.156          0.118          0.164 t with dof 3
## 12          10        0.258          0.286          0.310 t with dof 3
## 13          20        0.454          0.580          0.534 t with dof 3
## 14          50        0.756          0.918          0.898 t with dof 3
## 15         100        0.926          0.996          0.994 t with dof 3
## 16           5        0.130          0.092          0.116 t with dof 5
## 17          10        0.210          0.188          0.208 t with dof 5
## 18          20        0.366          0.408          0.394 t with dof 5
## 19          50        0.694          0.772          0.748 t with dof 5
## 20         100        0.928          0.970          0.948 t with dof 5
## 21           5        0.132          0.096          0.116      laplace
## 22          10        0.264          0.270          0.294      laplace
## 23          20        0.364          0.468          0.468      laplace
## 24          50        0.710          0.832          0.880      laplace
## 25         100        0.912          0.980          0.992      laplace
## 26           5        0.140          0.106          0.120     logistic
## 27          10        0.210          0.200          0.216     logistic
## 28          20        0.346          0.362          0.348     logistic
## 29          50        0.672          0.700          0.664     logistic
## 30         100        0.946          0.952          0.916     logistic
## 31           5        0.082          0.058          0.074       normal
## 32          10        0.192          0.182          0.194       normal
## 33          20        0.354          0.340          0.286       normal
## 34          50        0.666          0.664          0.562       normal
## 35         100        0.948          0.936          0.854       normal
```

Appendix 3

In this appendix, we present the results of the Type I simulation.

```
##    sample_size t_test_TypeI wmw_test_TypeI med_test_TypeI distribution
## 1            5        0.042          0.038          0.046      uniform
## 2           10        0.042          0.036          0.036      uniform
## 3           20        0.036          0.036          0.036      uniform
## 4           50        0.054          0.056          0.042      uniform
## 5          100        0.036          0.042          0.032      uniform
## 6            5        0.040          0.026          0.042  exponential
## 7           10        0.036          0.048          0.048  exponential
## 8           20        0.054          0.066          0.062  exponential
## 9           50        0.050          0.046          0.054  exponential
## 10         100        0.054          0.058          0.068  exponential
## 11           5        0.062          0.046          0.060 t with dof 3
## 12          10        0.056          0.038          0.046 t with dof 3
## 13          20        0.042          0.052          0.042 t with dof 3
## 14          50        0.062          0.034          0.050 t with dof 3
## 15         100        0.058          0.050          0.056 t with dof 3
## 16           5        0.036          0.024          0.038 t with dof 5
## 17          10        0.048          0.048          0.046 t with dof 5
## 18          20        0.042          0.060          0.064 t with dof 5
## 19          50        0.056          0.058          0.048 t with dof 5
## 20         100        0.038          0.044          0.046 t with dof 5
## 21           5        0.038          0.024          0.040      laplace
## 22          10        0.046          0.028          0.038      laplace
## 23          20        0.056          0.046          0.052      laplace
## 24          50        0.048          0.042          0.040      laplace
## 25         100        0.062          0.072          0.074      laplace
## 26           5        0.052          0.044          0.058     logistic
## 27          10        0.042          0.034          0.042     logistic
## 28          20        0.062          0.048          0.040     logistic
## 29          50        0.054          0.054          0.050     logistic
## 30         100        0.050          0.072          0.068     logistic
## 31           5        0.040          0.032          0.042       normal
## 32          10        0.056          0.036          0.052       normal
## 33          20        0.036          0.036          0.040       normal
## 34          50        0.056          0.058          0.066       normal
## 35         100        0.048          0.042          0.046       normal
```

Appendix 4

In this appendix, we present the code used for the power simulation.

```
#----------------------------------------------------------------------------------

# set-up for Power simulation

nsim <- 500
p.t <- p.wmw <- p.med <- numeric(nsim) #empty vector of p values for each test

sample_sizes <- c(5, 10, 20, 50, 100) # sample sizes for which we do the test.
n_sample_sizes <- length(sample_sizes)

#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for t3:

set.seed(1234) #For reproductibility
delta <- sqrt(3)/2 #delta = sqrt(variance)/2

results_t3 <- data.frame(
  sample_size = sample_sizes,
  t_test_power = numeric(n_sample_sizes),
  wmw_test_power = numeric(n_sample_sizes),
  med_test_power = numeric(n_sample_sizes)
)

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rt(n, 3)
    Y2 <- rt(n, 3) + delta
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    #Wilcoxon-Mann-Whitney test
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    #median test
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  # Monte-Carlo approximation of the power
  # for alpha = .05
  results_t3$t_test_power[j] <- mean(p.t < 0.05)
  results_t3$wmw_test_power[j] <- mean(p.wmw < 0.05)
  results_t3$med_test_power[j] <- mean(p.med < 0.05)
```

```r
}



#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for Exponential distribution

set.seed(10234) #For reproductibility

results_exp <- data.frame(
  sample_size = sample_sizes,
  t_test_power = numeric(n_sample_sizes),
  wmw_test_power = numeric(n_sample_sizes),
  med_test_power = numeric(n_sample_sizes)
)

#variance = 1/rate**2

delta <- sqrt(1)/2 #delta = sqrt(variance)/2
for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for (i in 1:nsim) {
    Y1 <- rexp(n, rate = 1)
    Y2 <- rexp(n, rate = 1)+delta
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)

    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2, exact=TRUE)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_exp$t_test_power[j] <- mean(p.t < 0.05)
  results_exp$wmw_test_power[j] <- mean(p.wmw < 0.05)
  results_exp$med_test_power[j] <- mean(p.med < 0.05)
}




#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for Laplace distribution

set.seed(100234) #For reproductibility

results_lap <- data.frame(
  sample_size = sample_sizes,
  t_test_power = numeric(n_sample_sizes),
  wmw_test_power = numeric(n_sample_sizes),
  med_test_power = numeric(n_sample_sizes)
)
```

```r
# variance = 2*sigma**2
delta <- sqrt(2)/2 #delta = sqrt(variance)/2

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim){
    Y1 <- rlaplace(n, mu = 0, sigma = 1)
    Y2 <- rlaplace(n, mu = delta, sigma = 1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_lap$t_test_power[j] <- mean(p.t < 0.05)
  results_lap$wmw_test_power[j] <- mean(p.wmw < 0.05)
  results_lap$med_test_power[j] <- mean(p.med < 0.05)
}

#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for t5 distribution

set.seed(1000234) #For reproductibility

results_t5 <- data.frame(
  sample_size = sample_sizes,
  t_test_power = numeric(n_sample_sizes),
  wmw_test_power = numeric(n_sample_sizes),
  med_test_power = numeric(n_sample_sizes)
)

#variance t-distribution = dof/(dof-2)
#with dof the degrees of freedom
delta <- sqrt(5/3)/2 #delta = sqrt(variance)/2

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rt(n, 5)
    Y2 <- rt(n, 5) + delta
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_t5$t_test_power[j] <- mean(p.t < 0.05)
  results_t5$wmw_test_power[j] <- mean(p.wmw < 0.05)
  results_t5$med_test_power[j] <- mean(p.med < 0.05)
}

#-----------------------------------------------------------------------------------

# Sample simulation study of Q2 for Logistic distribution

set.seed(10000234) #For reproductibility

results_log <- data.frame(
  sample_size = sample_sizes,
  t_test_power = numeric(n_sample_sizes),
  wmw_test_power = numeric(n_sample_sizes),
  med_test_power = numeric(n_sample_sizes)
)

#variance = scale**2 * pi**2 / 3
delta <- sqrt(pi**2/3)/2 #delta = sqrt(variance)/2
for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rlogis(n, location=0, scale=1)
    Y2 <- rlogis(n, location = delta, scale=1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_log$t_test_power[j] <- mean(p.t < 0.05)
  results_log$wmw_test_power[j] <- mean(p.wmw < 0.05)
  results_log$med_test_power[j] <- mean(p.med < 0.05)
}



#-----------------------------------------------------------------------------------

# Sample simulation study of Q2 for normal distribution

set.seed(100000234) #For reproductibility

results_norm <- data.frame(
  sample_size = sample_sizes,
  t_test_power = numeric(n_sample_sizes),
  wmw_test_power = numeric(n_sample_sizes),
  med_test_power = numeric(n_sample_sizes)
)
#variance = sqrt(sigma)
```

```r
delta <- sqrt(1)/2 #delta = sqrt(variance)/2
for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rnorm(n, 0, 1)
    Y2 <- rnorm(n, delta, 1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_norm$t_test_power[j] <- mean(p.t < 0.05)
  results_norm$wmw_test_power[j] <- mean(p.wmw < 0.05)
  results_norm$med_test_power[j] <- mean(p.med < 0.05)
}

#-----------------------------------------------------------------------------------

# Sample simulation study of Q2 for uniform distribution

set.seed(1000000234) #For reproductibility

results_unif <- data.frame(
  sample_size = sample_sizes,
  t_test_power = numeric(n_sample_sizes),
  wmw_test_power = numeric(n_sample_sizes),
  med_test_power = numeric(n_sample_sizes)
)
#variance = 1/12 * (max-min)**2
delta <- sqrt(1/12)/2 #delta = sqrt(variance)/2
for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]

  for(i in 1:nsim)
  {
    Y1 <- runif(n, min=0, max=1)
    Y2 <- runif(n, min=0, max=1)+delta
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_unif$t_test_power[j] <- mean(p.t < 0.05)
  results_unif$wmw_test_power[j] <- mean(p.wmw < 0.05)
  results_unif$med_test_power[j] <- mean(p.med < 0.05)
}
```

```
# Add distribution column
results_unif$distribution <- "uniform"
results_exp$distribution <- "exponential"
results_t3$distribution <- "t with dof 3"
results_t5$distribution <- "t with dof 5"
results_lap$distribution <- "laplace"
results_log$distribution <- "logistic"
results_norm$distribution <- "normal"

# Combine all datasets into one
results_Power <- rbind(
  results_unif,
  results_exp,
  results_t3,
  results_t5,
  results_lap,
  results_log,
  results_norm
)
```

Appendix 5

In this appendix, we present the code used for the Type I error rate simulation.

```
# set-up for Type I error rate simulation

nsim <- 500
p.t <- p.wmw <- p.med <- numeric(nsim) #empty vector of p values for each test

sample_sizes <- c(5, 10, 20, 50, 100) # sample sizes for which we do the test.
n_sample_sizes <- length(sample_sizes)

#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for t3:

set.seed(1234) #For reproductibility

results_t3 <- data.frame(
  sample_size = sample_sizes,
  t_test_TypeI = numeric(n_sample_sizes),
  wmw_test_TypeI = numeric(n_sample_sizes),
  med_test_TypeI = numeric(n_sample_sizes)
)

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rt(n, 3)
    Y2 <- rt(n, 3)
    X <- factor(c(rep("A",n),rep("B",n)))
```

```r
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    #Wilcoxon-Mann-Whitney test
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    #median test
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  # Monte-Carlo approximation of the power
  # for alpha = .05
  results_t3$t_test_TypeI[j] <- mean(p.t < 0.05)
  results_t3$wmw_test_TypeI[j] <- mean(p.wmw < 0.05)
  results_t3$med_test_TypeI[j] <- mean(p.med < 0.05)
}



#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for Exponential distribution

set.seed(10234) #For reproductibility

results_exp <- data.frame(
  sample_size = sample_sizes,
  t_test_TypeI = numeric(n_sample_sizes),
  wmw_test_TypeI = numeric(n_sample_sizes),
  med_test_TypeI = numeric(n_sample_sizes)
)

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for (i in 1:nsim) {
    Y1 <- rexp(n, rate = 1)
    Y2 <- rexp(n, rate = 1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)

    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2, exact=TRUE)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_exp$t_test_TypeI[j] <- mean(p.t < 0.05)
  results_exp$wmw_test_TypeI[j] <- mean(p.wmw < 0.05)
  results_exp$med_test_TypeI[j] <- mean(p.med < 0.05)
}




#----------------------------------------------------------------------------------
```

```r
# Sample simulation study of Q2 for Laplace distribution

set.seed(100234) #For reproductibility

results_lap <- data.frame(
  sample_size = sample_sizes,
  t_test_TypeI = numeric(n_sample_sizes),
  wmw_test_TypeI = numeric(n_sample_sizes),
  med_test_TypeI = numeric(n_sample_sizes)
)


for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim){
    Y1 <- rlaplace(n, mu = 0, sigma = 1)
    Y2 <- rlaplace(n, mu = 0, sigma = 1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                  distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_lap$t_test_TypeI[j] <- mean(p.t < 0.05)
  results_lap$wmw_test_TypeI[j] <- mean(p.wmw < 0.05)
  results_lap$med_test_TypeI[j] <- mean(p.med < 0.05)
}

#------------------------------------------------------------------------------------

# Sample simulation study of Q2 for t5 distribution

set.seed(1000234) #For reproductibility

results_t5 <- data.frame(
  sample_size = sample_sizes,
  t_test_TypeI = numeric(n_sample_sizes),
  wmw_test_TypeI = numeric(n_sample_sizes),
  med_test_TypeI = numeric(n_sample_sizes)
)


for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rt(n, 5)
    Y2 <- rt(n, 5)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_t5$t_test_TypeI[j] <- mean(p.t < 0.05)
  results_t5$wmw_test_TypeI[j] <- mean(p.wmw < 0.05)
  results_t5$med_test_TypeI[j] <- mean(p.med < 0.05)
}

#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for Logistic distribution

set.seed(10000234) #For reproductibility

results_log <- data.frame(
  sample_size = sample_sizes,
  t_test_TypeI = numeric(n_sample_sizes),
  wmw_test_TypeI = numeric(n_sample_sizes),
  med_test_TypeI = numeric(n_sample_sizes)
)

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rlogis(n, location=0, scale=1)
    Y2 <- rlogis(n, location=0, scale=1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_log$t_test_TypeI[j] <- mean(p.t < 0.05)
  results_log$wmw_test_TypeI[j] <- mean(p.wmw < 0.05)
  results_log$med_test_TypeI[j] <- mean(p.med < 0.05)
}



#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for normal distribution

set.seed(100000234) #For reproductibility

results_norm <- data.frame(
  sample_size = sample_sizes,
  t_test_TypeI = numeric(n_sample_sizes),
  wmw_test_TypeI = numeric(n_sample_sizes),
  med_test_TypeI = numeric(n_sample_sizes)
)

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]
  for(i in 1:nsim)
  {
    Y1 <- rnorm(n, 0, 1)
    Y2 <- rnorm(n, 0, 1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_norm$t_test_TypeI[j] <- mean(p.t < 0.05)
  results_norm$wmw_test_TypeI[j] <- mean(p.wmw < 0.05)
  results_norm$med_test_TypeI[j] <- mean(p.med < 0.05)
}

#----------------------------------------------------------------------------------

# Sample simulation study of Q2 for uniform distribution

set.seed(1000000234) #For reproductibility

results_unif <- data.frame(
  sample_size = sample_sizes,
  t_test_TypeI = numeric(n_sample_sizes),
  wmw_test_TypeI = numeric(n_sample_sizes),
  med_test_TypeI = numeric(n_sample_sizes)
)

for (j in 1:length(sample_sizes)) {
  n <- sample_sizes[j]

  for(i in 1:nsim)
  {
    Y1 <- runif(n, min=0, max=1)
    Y2 <- runif(n, min=0, max=1)
    X <- factor(c(rep("A",n),rep("B",n)))
    Y <- c(Y1, Y2)
    #permutation t-test
    p.t[i] <- pvalue(oneway_test(Y ~ X,
                                 distribution=approximate(nresample=1000)))
    p.wmw[i] <- wilcox.test(Y1,Y2)$p.value
    p.med[i] <- median.test.two_sided(Y1, Y2)
  }
  results_unif$t_test_TypeI[j] <- mean(p.t < 0.05)
  results_unif$wmw_test_TypeI[j] <- mean(p.wmw < 0.05)
  results_unif$med_test_TypeI[j] <- mean(p.med < 0.05)
}



# Add distribution variable
results_unif$distribution <- "uniform"
results_exp$distribution <- "exponential"
results_t3$distribution <- "t with dof 3"
results_t5$distribution <- "t with dof 5"
results_lap$distribution <- "laplace"
results_log$distribution <- "logistic"
results_norm$distribution <- "normal"

results_TypeI <- rbind(
  results_unif,
  results_exp,
  results_t3,
  results_t5,
  results_lap,
  results_log,
  results_norm
)
```