Exact Distribution of a Two-Sample Nonparametric Test for Comparing Hazard Rates

William Breslin

Abstract

A two-sample nonparametric test for comparing the hazard rate functions of two populations was proposed in Kochar (1979) and Kochar (1981). The exact null distribution of the test statistic does not have a known closed-form distribution function. In this paper, the exact null distribution of this test statistic is obtained using the statistical software R for small sample sizes. Furthermore, an R package is developed to implement the hazard rate test. The normal approximation is discussed for sufficiently large sample sizes ($m \geqslant 8$ and $n \geqslant 8$) as an alternative to finding the exact null distribution due to being computationally expensive to obtain. The power of this test is compared via simulation to the Wilcoxon rank-sum test and the log-rank test. The results from the simulation suggest that the hazard rate test is slightly more powerful than the rank-sum test, but less powerful than the log-rank test under optimal conditions for the log-rank test.

Introduction

Let X and Y be two non-negative independent random variables of continuous type with probability density functions f_X and f_Y, cumulative distribution functions F_X and F_Y, and survival functions $\bar{F}_X = 1 - F_X$ and $\bar{F}_Y = 1 - F_Y$ respectively. Let $r_X = f_X/\bar{F}_X$ and $r_Y = f_Y/\bar{F}_Y$ be their hazard rates. Based on independent random samples $X_1, \ldots, X_m$ on X and $Y_1, \ldots, Y_n$ on Y, Kochar (1979) considered the problem of testing the null hypothesis

$$H_0 : r_X(t) = r_Y(t), \quad \forall t \geqslant 0 \tag{1}$$

against the alternative hypothesis

$$H_A : r_X(t) \leqslant r_Y(t), \quad \forall t \geqslant 0 \tag{2}$$

with a strict inequality on a set of non-zero probability. The kernel of the proposed test statistic is

$$\phi(X_{i1}, X_{i2}, Y_{j1}, Y_{j2}) = \begin{cases} 1 & \text{if } yyxx \text{ or } xyyx \\ 0 & \text{if } xyxy \text{ or } yxyx \\ -1 & \text{if } xxyy \text{ or } yxxy \end{cases} \tag{3}$$

and is computed by choosing two values from the X-sample and two values from the Y-sample. These four values are then ranked, and depending on the sequencing of the ranks either a 1, 0, or -1 is assigned to ϕ. For example, the sequence $yyxx$ represents the case when both of the y values are smaller than both of the x values. In this case, ϕ is equal to 1. The test proposed in Kochar (1979) is to reject the null hypothesis H_0 against the alternative H_A for large values of the U-statistic,

$$W = \left\{ \binom{n}{2} \binom{m}{2} \right\}^{-1} \sum \phi(X_{i1}, X_{i2}, Y_{j1}, Y_{j2}) \tag{4}$$

The distribution of this test statistic does not have a known closed-form distribution; however, the procedure for computing this test statistic is conceptually relatively simple. The primary objective of this paper is to provide the means for computing the null distribution of this test statistic. Additionally, the 'HazardRateTest' R package was created for applying this test to applicable data-sets.

Mathematical Theory

First, a few definitions. A random variable X is said to be greater than the random variable Y in *stochastic ordering* if and only if the survival function of X is greater than the survival function of Y.

$$X \geqslant_{st} Y \iff \bar{F}_X(t) \geqslant \bar{F}_Y(t), \quad \forall t \geqslant 0 \tag{5}$$

Similarly, a random variable X is said to be greater than the random variable Y in *hazard rate ordering* if the hazard rate function of X is smaller than the hazard rate function of Y.

$$X \geqslant_{hr} Y \iff r_X(t) \leqslant r_Y(t) \quad \forall t \geqslant 0 \tag{6}$$

For both of these definitions, the inequality must be strict on a set of non-zero probability. Note that the distribution of a random variable is uniquely determined by its hazard rate function, and vice-versa. That means that the null hypothesis is equivalent to:

$$H_0 : F_X(t) = F_Y(t), \quad \forall t \geqslant 0 \tag{7}$$

This is the null hypothesis for several nonparametric tests, such as the Wilcoxon rank-sum test and the log-rank test. Even though this test uses hazard rate functions to compare the two samples, it is still testing whether or not the two samples came from populations with the same distribution. The main difference between the proposed hazard rate test and the rank-sum and log-rank tests is that the alternative hypotheses are not the same. The alternative hypothesis for the hazard rate test is that one of the random variables is greater than the other in hazard rate ordering, whereas the alternative for the rank-sum and log-rank tests is that one of the random variables is greater than the other in stochastic ordering.

$$H_A : X \geqslant_{hr} Y \tag{8}$$

An important fact that will be shown is that hazard rate ordering implies stochastic ordering, but the converse is not necessarily true. The following lemma will be useful for proving this.

Lemma: $X \geqslant_{hr} Y$ if and only if $\bar{F}_X(t)/\bar{F}_Y(t)$ is a non-decreasing function of t.

The proof below involves writing the hazard rate function in terms of the log of the survival function, and using the fact that the log function is monotonic.

Proof of Lemma:

$$\begin{aligned} X \geqslant_{hr} Y &\iff r_X(t) \leqslant r_Y(t) \\ &\iff -\frac{d}{dt}\log\left[\bar{F}_X(t)\right] \leqslant -\frac{d}{dt}\log\left[\bar{F}_Y(t)\right] \\ &\iff \frac{d}{dt}\log\left[\bar{F}_X(t)/\bar{F}_Y(t)\right] \geqslant 0 \\ &\iff \frac{d}{dt}\bar{F}_X(t)/\bar{F}_Y(t) \geqslant 0 \\ &\iff \bar{F}_X(t)/\bar{F}_Y(t) \text{ is a non-decreasing in } t \end{aligned}$$

This lemma is used to prove that hazard rate ordering implies stochastic ordering.

Theorem: If $X \geqslant_{hr} Y$, then $X \geqslant_{st} Y$.

This proof is fairly straightforward. By the above lemma, it can easily be seen that $\bar{F}_X(t)/\bar{F}_Y(t) \geqslant 1$, since $\bar{F}_X(0)/\bar{F}_Y(0) = 1$ and since $\bar{F}_X(t)/\bar{F}_Y(t)$ is non-decreasing. This fact will be used to prove the above theorem, and is the point of failure for the converse of this theorem.

Proof of Theorem:

$$\begin{aligned} X \geqslant_{hr} Y &\iff \bar{F}_X(t)/\bar{F}_Y(t) \text{ is non-decreasing} \\ &\implies \bar{F}_X(t)/\bar{F}_Y(t) \geqslant 1 \text{ for every } t > 0 \\ &\iff \bar{F}_X(t) \geqslant \bar{F}_Y(t) \text{ for every } t > 0 \\ &\iff X \geqslant_{st} Y \end{aligned}$$

This shows that hazard rate ordering is a stronger condition than stochastic ordering. This is beneficial because if sufficient evidence is found to reject the null hypothesis for the alternative of hazard rate ordering, then sufficient evidence has also been found to reject the null hypothesis for the alternative of stochastic ordering. As a consequence, it should be more likely to correctly reject the null hypothesis for this test, thus theoretically increasing the power of this test relative to the stochastic ordering alternative.

The alternative hypothesis for this test, that $X \geqslant_{hr} Y$, is true if and only if for every $s \geqslant t \geqslant 0$:

$$\delta(s,t) = \bar{F}_X(s)\bar{F}_Y(t) - \bar{F}_X(t)\bar{F}_Y(s) \geqslant 0 \tag{9}$$

This equation is the basis from which the test statistic will be constructed. Let $s = \max\{x,y\}$ and $t = \min\{x,y\}$. For this choice of s and t, it will always be true that $s \geqslant t \geqslant 0$, assuming that the variables are non-negative. Now define $\eta(F_X, F_Y)$, as shown in the equation below.

$$\eta(F_X,F_Y) = \mathbb{E}\Big[\delta(\max\{X,Y\},\min\{X,Y\})\Big] = \iint_{0\leqslant x\leqslant y} \delta(s,t)\Big[dF_X(x)dF_Y(y) + dF_X(y)dF_Y(x)\Big] \tag{10}$$

Evaluating the integral results in the following probability expression.

$$P(Y_1 \leqslant Y_2 \leqslant X_1 \leqslant X_2) + P(X_1 \leqslant Y_1 \leqslant Y_2 \leqslant X_2) - P(X_1 \leqslant X_2 \leqslant Y_1 \leqslant Y_2) - P(Y_1 \leqslant X_1 \leqslant X_2 \leqslant Y_2) \tag{11}$$

The X and Y sequences defined in the kernel are obtained from the probability expressions above. The first two probability expressions suggest that $X \geqslant_{hr} Y$ whereas the second two suggest $X \leqslant_{hr} Y$. The test statistic for this test is simply counting the number of sequences that suggest a hazard rate ordering in the direction of the alternative hypothesis, and subtracting from that the number of sequences suggesting a hazard rate ordering in the direction opposite of the alternative hypothesis. The kernel ϕ leads to an unbiased estimator of $\eta(F_X, F_Y)$.

Exact Null Distribution

Consider two data-sets X and Y of size m and n respectively. This test is a permutation test, so consider all possible permutations of the combined data-set, and then separate the permutations back into two samples of size m and n. For every distinct permutation obtained this way, the test statistic W is computed, and the frequency distribution of W is the null distribution for this test.

$$\text{Number of distinct permutations} = \binom{n+m}{n} \tag{12}$$

To compute the test statistic for a given permutation, all possible combinations of two X and two Y must be compared to the kernel sequences.

$$\text{Number of sequences} = \binom{n}{2}\binom{m}{2} \tag{13}$$

Now to find the null distribution computationally. The most difficult part of computing the test statistic is comparing the sequences of X and Y values. Table 1 shows expressions that are equivalent to kernel sequences. While not as concise as the original conditions, they are easier to implement.

Table 1: Equivalent kernel sequence conditions

Sequence	Equivalent Condition
$yyxx$	$\min\{x_1, x_2\} > \max\{y_1, y_2\}$
$xyyx$	$\max\{x_1, x_2\} > \max\{y_1, y_2\}$ and $\min\{x_1, x_2\} < \min\{y_1, y_2\}$
$xxyy$	$\min\{y_1, y_2\} > \max\{x_1, x_2\}$
$yxxy$	$\max\{y_1, y_2\} > \max\{x_1, x_2\}$ and $\min\{y_1, y_2\} < \min\{x_1, x_2\}$

The code for finding the null distribution is broken up into two pieces. The first piece is a function that computes the summation of the kernel $\phi(\cdot)$ for a given permutation. The second piece finds the exact null distribution for given sample sizes m and n.

```r
kernel = function(x.pairs,y.pairs){
  w=0
  for (i in 1:nrow(x.pairs)){
    for (j in 1:nrow(y.pairs)){
      x1 = x.pairs[i,1]; x2 = x.pairs[i,2]
      y1 = y.pairs[j,1]; y2 = y.pairs[j,2]
      if (min(x1,x2) > max(y1,y2)){
        w = w+1}
      if (max(x1,x2) > max(y1,y2) & min(x1,x2) < min(y1,y2)){
        w = w+1}
      if (min(y1,y2) > max(x1,x2)){
        w = w-1}
      if (max(y1,y2) > max(x1,x2) & min(y1,y2) < min(x1,x2)){
        w = w-1}}}
  return(w)}
```

```r
null.dist = function(m,n){
  N = m+n
  Ranks = 1:N
  x.data = t(combn(Ranks,m)); y.data=c()
  for (i in 1:nrow(x.data)){
    if (i == 1){
      y.data = t(as.data.frame(setdiff(Ranks,x.data[1,])))}
    else{
      temp = setdiff(Ranks, x.data[i,])
      y.data = rbind(y.data,temp)}
    x.pairs = t(combn(x.data[i,],2))
    y.pairs = t(combn(y.data[i,],2))
    if (i == 1){
      w = kernel(x.pairs,y.pairs)/(choose(n,2)*choose(m,2))}
    else{
      temp = kernel(x.pairs,y.pairs)/(choose(n,2)*choose(m,2))
      w = c(w,temp)}}
  support = sort(unique(w))
  for (i in support){
    if (i == min(support)){
      freq = sum(w==i)
      prob = freq/length(w)}
    else{
      freq = c(freq, sum(w==i))
      prob = c(prob, sum(w==i)/length(w))}}
  table = cbind(support,freq,prob)
  colnames(table)=c("W","Occurrences","Probability")
  return(table)}
```

The R output below is the result of applying the null.dist(·) function to selected small sample sizes.

```
null.dist(m=3, n=3)
```

```
##               W Occurrences Probability
## [1,] -1.0000000           2        0.10
## [2,] -0.5555556           3        0.15
## [3,] -0.3333333           2        0.10
## [4,] -0.1111111           3        0.15
## [5,]  0.1111111           3        0.15
## [6,]  0.3333333           2        0.10
## [7,]  0.5555556           3        0.15
## [8,]  1.0000000           2        0.10
```

```
null.dist(m=3, n=4)
```

```
##                W Occurrences Probability
##  [1,] -1.0000000           2  0.05714286
##  [2,] -0.6666667           4  0.11428571
##  [3,] -0.5555556           1  0.02857143
##  [4,] -0.3333333           4  0.11428571
##  [5,] -0.2222222           2  0.05714286
##  [6,] -0.1111111           1  0.02857143
##  [7,]  0.0000000           6  0.17142857
##  [8,]  0.1111111           2  0.05714286
##  [9,]  0.2222222           1  0.02857143
## [10,]  0.3333333           5  0.14285714
## [11,]  0.4444444           1  0.02857143
## [12,]  0.5555556           1  0.02857143
## [13,]  0.6666667           3  0.08571429
## [14,]  1.0000000           2  0.05714286
```

R Package: HazardRateTest

The ‘HazardRateTest’ package contains the code for finding the exact null distribution of this test statistic (as shown in the previous section) along with code for applying this test to data-sets that meet the requirements of this test. This package is hosted on Github, and is not in the CRAN repositories. To install the package from Github, the ‘devtools’ package is required. The following command can be used to install the ‘HazardRateTest’ package using ‘devtools’.

```r
devtools::install_github("wbbreslin/HazardRateTest")
```

This package has two main functions:

```r
null.dist(m,n)
hazard.test(x,y)
```

The null.dist function is used to find the null distribution for sample sizes n and m, whereas the hazard.test function applies the hazard rate test to two samples x and y.

The code for this package is open-source, and can be found on Github using the following link:

https://github.com/wbbreslin/HazardRateTest

The Normal Approximation

The null distribution is symmetric when $n = m$, and approximately symmetric otherwise. As a result, the expected value of W under H_0 is approximately 0. It was shown in Kochar (1979) that the null variance of W is given by the following expression, where $N = n + m$.

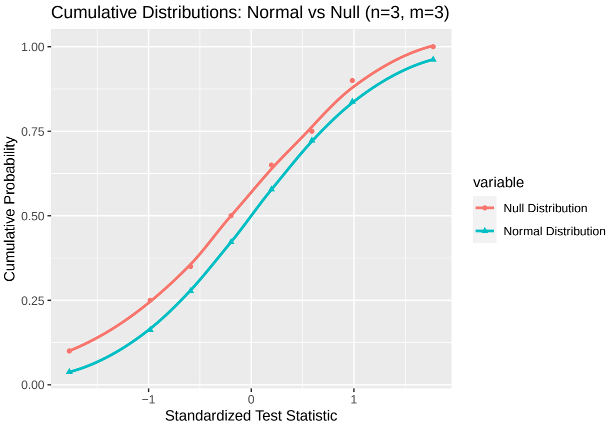
$$Var_0(W) = \frac{16mnN - (11m^2 + 11n^2 + 6mn) - 3N + 8}{210\binom{n}{2}\binom{m}{2}} \tag{14}$$

Also from Kochar (1979), it is known that as the sample sizes increase, the distribution of Z approaches a standard normal distribution, where Z is defined as:

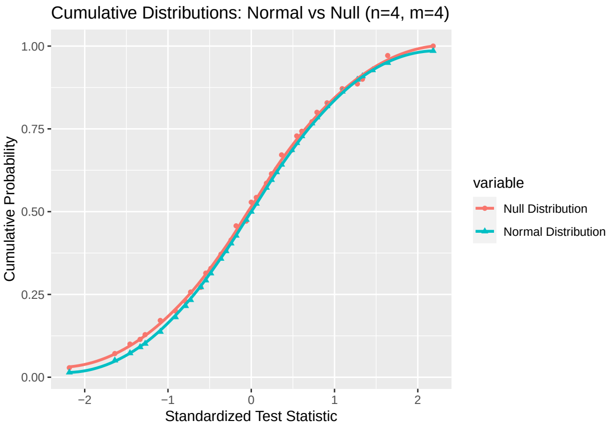
$$Z = \frac{W - \mathbb{E}_0(W)}{\sqrt{\mathrm{Var}_0(W)}} \tag{15}$$

When working with larger samples, it is convenient to compute the p-value using the normal approximation due to the computational expense required to find the exact null distribution of W. In this section, it will be shown that the normal approximation is sufficient even for relatively small values of m and n.

The cumulative distribution of the standardized null distribution is compared visually to the cumulative distribution of a normal distribution for select sample sizes, starting with $n = m = 3$. As can be seen in the following graph, the normal approximation works very poorly for $n = m = 3$.



Now for $n = m = 4$. This graph shows a much closer fit; however, there are inaccuracies in the tails of the distribution. Accurate approximation of the tails is extremely important, considering that the critical values for hypothesis testing are in the tails of the distribution.



For $n = m = 5$ and greater, the normal approximation is almost indistinguishable from the actual null distribution based on a visual inspection of the graphs. Rather than looking at the graphs, the critical values are compared for common values of α. Comparing critical values illustrates the quality of the normal approximation more accurately than a visual inspection of the graphs. The table below shows the critical values for a two-sided hypothesis test for $\alpha = 0.05$ and $\alpha = 0.01$. As the null distribution is discrete, the critical values do not result in tail probabilities exactly equal to α, only approximately equal.

Table 2: Normal approximation critical values for hazard rate test

	Critical Value: $\alpha = 0.05$	Critical Value: $\alpha = 0.01$
Null Dist. ($n = m = 5$)	2.02	N/A
Null Dist. ($n = m = 6$)	2.02	2.84
Null Dist. ($n = m = 7$)	1.99	2.84
Null Dist. ($n = m = 8$)	1.96	2.58
Normal Dist.	**1.96**	**2.58**

For $n = m = 8$, the critical values of the normal approximation match the critical values obtained from a standard normal distribution. Therefore, the normal approximation of the hazard rate test can be used when both samples have at least 8 observations. For smaller samples, the exact null distribution should be used.

Power Comparisons

The power of this test was compared to the Wilcoxon rank-sum test and the log-rank test. The power was compared through simulation using an exponential distribution, for which the log-rank test is the locally most powerful rank based test. The following steps describe the simulation method used to estimate the power of these tests.

1. Take a random sample of size m from an exponential distribution with mean $\mu = 1$
2. Take a random sample of size n from an exponential distribution with mean $\mu > 1$
3. Compute the two-tailed p-values for each test given these samples
4. Repeat this process 1000 times
5. Find the proportion of times that we reject the null hypothesis at $\alpha = 0.05$ for each test

Prior to conducting the simulations, two packages need to be loaded: the 'survival' package and the 'HazardRateTest' package.

```
library(HazardRateTest)
library(survival)
```

To ensure replicability, a seed was set prior to running the simulations.

```
set.seed(314)
```

The code for the simulation was put into a function, so the sample sizes and the mean specified for the alternative hypothesis could easily be adjusted.

```
power.simulation = function(alt, n, m){
  Hazard.Reject = Wilcox.Reject = Logrank.Reject = 0
  itr = 1000

  for (i in 1:itr){
    x = rexp(n=n, rate=1)
    y = rexp(n=m, rate=1/alt)
    H = hazard.test(x,y)
    Hazard.p = H$p.value
    if (Hazard.p <= 0.05){Hazard.Reject = Hazard.Reject+1}
    W = wilcox.test(x,y)
    Wilcox.p = W$p.value
    if (Wilcox.p <= 0.05){Wilcox.Reject = Wilcox.Reject+1}
    failure = c(x,y)
    N = length(x)+length(y)
    censor = rep(1,N)
    group = c(rep(1, length(x)),rep(2, length(y)))
    logrank = survdiff(Surv(failure, censor) ~ group)
    chi.sq = logrank$chisq
    Logrank.p = 1-pchisq(chi.sq,length(logrank$n)-1)
    if (Logrank.p <= 0.05){Logrank.Reject = Logrank.Reject+1}
  }

  Hazard.Power = Hazard.Reject/itr
  Wilcox.Power = Wilcox.Reject/itr
  Logrank.Power = Logrank.Reject/itr
  out = rbind(Hazard.Power, Wilcox.Power, Logrank.Power)
  print(out)
}
```

To run a simulation, specify the alternative mean, and the two sample sizes. The code below runs a simulation with alternative $\mu = 1.2$ on sample sizes $n = m = 10$.

```
power.simulation(alt=1.2, n=10, m=10)
```

```
##                [,1]
## Hazard.Power  0.080
## Wilcox.Power  0.071
## Logrank.Power 0.099
```

A total of 20 simulations were conducted, with 5 different alternative means and 4 different pairs of sample sizes. The first set of simulations with $n = m = 10$ are shown in the following table.

Table 3: Power simulation results for $n = m = 10$

n	m	μ	Hazard	Wilcoxon	Log-Rank
10	10	1.2	0.080	0.071	0.099
10	10	1.4	0.101	0.093	0.127
10	10	1.6	0.127	0.112	0.171
10	10	1.8	0.178	0.174	0.249
10	10	2.0	0.238	0.224	0.315

In all five cases, the hazard rate test had greater power than the Wilcoxon test, but less power than the log-rank test, as expected.

The second set of simulations is for unequal sample sizes $n = 10$ and $m = 15$.

Table 4: Power simulation results for $n = 10$ and $m = 15$

n	m	μ	Hazard	Wilcoxon	Log-Rank
10	15	1.2	0.068	0.058	0.090
10	15	1.4	0.076	0.081	0.114
10	15	1.6	0.165	0.174	0.235
10	15	1.8	0.243	0.212	0.320
10	15	2.0	0.283	0.282	0.389

The hazard rate test only outperformed the Wilcoxon test in three of the five simulations. The log-rank test still had the greatest power in all of the simulations. Also worth noting, the simulations had lower power for $\mu = 1.2$ and $\mu = 1.4$, despite increasing the sample size from $m = 10$ to $m = 15$. For all other alternative μ, the power did increase with the sample size, as to be expected.

The third set of simulations is for equal sample sizes $n = m = 15$.

Table 5: Power simulation results for $n = m = 15$

n	m	μ	Hazard	Wilcoxon	Log-Rank
15	15	1.2	0.070	0.064	0.085
15	15	1.4	0.141	0.121	0.183
15	15	1.6	0.200	0.178	0.247
15	15	1.8	0.235	0.225	0.309
15	15	2.0	0.353	0.327	0.431

In all five cases, the hazard rate test had greater power than the Wilcoxon test, but less power than the log-rank test, as expected. With the exception of the $\mu = 1.2$ alternative, the power did increase along with the sample size, suggesting that the result of the $n = m = 10$ and $\mu = 1.2$ simulation was an outlier with unusually high power.

The final set of simulations is for unequal sample sizes $n = 15$ and $m = 20$.

Table 6: Power simulation results for $n = 15$ and $m = 20$

n	m	μ	Hazard	Wilcoxon	Log-Rank
15	20	1.2	0.059	0.061	0.095
15	20	1.4	0.145	0.126	0.172
15	20	1.6	0.195	0.168	0.244
15	20	1.8	0.296	0.267	0.385
15	20	2.0	0.411	0.386	0.506

Overall, the log-rank test outperformed the others 100% of the time. The log-rank test is the locally most powerful rank-based test for the exponential distribution, so those results were to be expected. The hazard rate test outperformed the Wilcoxon test in 17 of the 20 simulations (85%).

The simulations support the claim that the hazard rate test is more powerful than the Wilcoxon test, at least for an exponential distribution. Further simulations could be conducted for other distributions, provided that they do not violate the assumptions of the hazard rate test (most notably continuity and non-negativity). Other viable distributions to test include the log-normal distribution, gamma distribution, and Weibull distribution.

Conclusions

The primary objective was to find the null distribution of the test statistic based on the concept of hazard rate ordering, developed by Professor Subhash Kochar. The code included within this paper is sufficient for finding the exact distribution when the sample sizes are small. For sample sizes starting at $m = 8$ and $n = 8$, the normal approximation gives satisfactory results that are nearly identical to those obtained using the exact distribution. Therefore, finding the exact distribution for larger sample sizes is not necessary.

The power analysis showed that this test is more powerful than the Wilcoxon rank-sum test. This increase in power comes at the cost of computational complexity of the test statistic, limiting the applicability of this test to small sample sizes. Due to the computational complexity, it is not recommended to use the hazard.test function from the ‘HazardRateTest’ package on samples much larger than $n = m = 100$. Even at $n = m = 100$, the code may take a few minutes to run.

Acknowledgments

This project was done under the direction of Professor Subhash Kochar, whom I want to thank for his guidance and assistance.

References

Kochar, Subhash. 1979. “Distribution-Free Comparison of Two Probability Distributions with Reference to Their Hazard Rates.” *Biometrika* 66 (3): 437–41.

———. 1981. “A New Distribution-Free Test for the Equality of Two Failure Rates.” *Biometrika* 68 (2): 423–26.