

On Performance Characteristics of a Nonparametric Subset Selection Procedure with a Small Randomized Block Experimental Design

Gary C. McDonald, Sajidah Alsaeed

Department of Mathematics and Statistics, Oakland University, Rochester, MI, USA

Email: mcdonald@oakland.edu, salsaeed@oakland.edu

How to cite this paper: McDonald, G.C. and Alsaeed, S. (2024) On Performance Characteristics of a Nonparametric Subset Selection Procedure with a Small Randomized Block Experimental Design. *Applied Mathematics*, 15, 630-650.

<https://doi.org/10.4236/am.2024.159038>

Received: August 23, 2024

Accepted: September 22, 2024

Published: September 25, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This article addresses the issue of computing the constant required to implement a specific nonparametric subset selection procedure based on ranks of data arising in a statistical randomized block experimental design. A model of three populations and two blocks is used to compute the probability distribution of the relevant statistic, the maximum of the population rank sums minus the rank sum of the “best” population. Calculations are done for populations following a normal distribution, and for populations following a bi-uniform distribution. The least favorable configuration in these cases is shown to arise when all three populations follow identical distributions. The bi-uniform distribution leads to an asymptotic counterexample to the conjecture that the least favorable configuration, *i.e.*, that configuration minimizing the probability of a correct selection, occurs when all populations are identically distributed. These results are consistent with other large-scale simulation studies. All relevant computational R-codes are provided in appendices.

Keywords

Rank Statistics, Least Favorable Configuration, Probability of Correct Selection, Slippage Configuration, Equal Spaced Configuration, Counterexample Configuration

1. Introduction

Nonparametric (distribution-free) subset selection procedures have been developed by Gupta and McDonald [1] for a one-way Analysis of Variance type model,

and by McDonald [2] for a two-way randomized block type model. These procedures are based on random data assumed to follow a continuous probability distribution stochastically ordered by the parameter of interest (e.g., a location or a scale parameter). The subset selection procedure selects a subset of the populations with the goal of including within the chosen subset the population which is “best” (e.g., that characterized with the largest mean), with a prescribed probability no less than a user prescribed probability, P^* . Specifying such procedures requires determination of the parameter configuration that minimizes the probability of a correct selection and subsequent calculation of the constant required to implement the procedure so that the probabilistic inference is valid over the entire parameter space. This minimizing configuration is referred to as the “least favorable configuration” (LFC) and is the focus of this article for a randomized block nonparametric subset selection rule to be described. The foundations of the subset selection rule are taken from McDonald [2] and also presented in McDonald and Alsaeed [3].

Let π_1, \dots, π_k be $k (\geq 2)$ independent populations. Let X_{ij} , $j = 1, \dots, n$; $i = 1, \dots, k$, be independent samples of size n from the k populations. Assume the random variables X_{ij} have a continuous cumulative distribution function (CDF) $F_j(x; \theta_j)$, where θ_j 's belong to some interval Θ on the real line. Suppose $F_j(x; \theta)$ is a stochastically increasing family of distributions in θ , i.e., if $\theta_1 < \theta_2$, then $F_j(x; \theta_1)$ and $F_j(x; \theta_2)$ are distinct and $F_j(x; \theta_2) \leq F_j(x; \theta_1)$ for all x . Examples of such families of distributions are: 1) any location parameter family, i.e., $F_j(x; \theta) = F_j(x - \theta)$; 2) any scale parameter family, i.e., $F_j(x; \theta) = F_j(x/\theta)$, $\theta > 0$, $x > 0$; any family of distribution functions whose densities possess the monotone likelihood ratio property (see Lehmann [4]).

Let R_{ij} denote the rank of the observation x_{ij} among $x_{1j}, x_{2j}, \dots, x_{kj}$; i.e., if there are exactly r of the observations x_{mj} , $m = 1, \dots, k$ less than x_{ij} then $R_{ij} = r + 1$. These ranks are well-defined with probability one, since the random variables are assumed to have a continuous distribution, and take integer values from 1 to k inclusive. Now define the rank sums for the k populations as

$$T_i = \sum_{j=1}^n R_{ij}, \quad i = 1, \dots, k. \quad (1.1)$$

The quantities T_i will define the procedures for selecting a subset of the k populations. Letting $\theta_{[i]}$ denote the i^{th} smallest unknown parameter, it follows that

$$F_j(x; \theta_{[1]}) \geq F_j(x; \theta_{[2]}) \geq \dots \geq F_j(x; \theta_{[k]}), \quad \forall x. \quad (1.2)$$

The population whose associated random variables have the distribution $F_j(x; \theta_{[k]})$ is called the “best” population. In case several populations possess the largest parameter value $\theta_{[k]}$, one of these is tagged at random and called the best. A “Correct Selection” (CS) is said to occur if and only if the best population is included in the selected subset. In the subset selection formulation one wishes to select a subset such that the probability is at least equal to a preassigned constant P^* ($k^{-1} < P^* < 1$) that the selected subset includes the best population. Formally, for a given selection rule R ,

$$\inf_{\Omega} P(\text{CS} | \mathbf{R}) \geq P^*, \quad (1.3)$$

where

$$\Omega = \{\boldsymbol{\theta} = (\theta_1, \dots, \theta_k) : \theta_i \in \Theta, i = 1, \dots, k\}. \quad (1.4)$$

The choice of P^* is specified by the analyst and represents the confidence level that the resultant selected subset will contain the best population. The number of populations in the selected subset is a random variable and is a nondecreasing function of P^* .

In a similar fashion, the “worst” population can be defined as that population characterized by the probability distribution $F(x; \theta_{(1)})$. Selection procedures can analogously be defined with P^* requirements on the selected subset to contain the worst population. The assignment of “best” and “worst” is problem specific noted in the applications.

Four subset selection rules are considered for the analysis of state motor vehicle traffic fatality rates (MVTFRs) as given in McDonald [5]. In this application the populations are states and the blocks are years. Since low (high) fatality rates are good (bad), the “best” (“worst”) state is the one with the smallest (largest) mean fatality rate.

The two selection rules for choosing a subset containing the worst population are given by:

$$R_1: \text{Select } \pi_i \text{ iff } T_i \geq \max(T_j, j = 1, \dots, k) - b_1 \quad (1.5)$$

$$R_2: \text{Select } \pi_i \text{ iff } T_i > b_2.$$

Similarly, the two selection rules for choosing a subset containing the best population are given by:

$$R_3: \text{Select } \pi_i \text{ iff } T_i \leq \min(T_j, j = 1, \dots, k) + b_3$$

$$R_4: \text{Select } \pi_i \text{ iff } T_i < b_4.$$

The non-negative constants b_1 , b_3 , and b_4 are chosen as small as possible and b_2 is chosen as large as possible preserving the probability P^* goal. In cases considered here, these constants are calculated assuming the population parameters are equal and, thus, the distribution of the T statistics are distribution free. That is, the calculations required to implement the nonparametric selection rules do not depend on the particular statistical distribution of the populations. With many of the parametric subset selection procedures, the LFC is that configuration in which all the population parameters are equal (e.g., see Gupta and Panchapakesan [6]). As derived in McDonald [2] rules R_1 and R_3 are justified over a slippage space, Ω' , where all parameters θ_i are equal with the possible exception of $\theta_{[k]} (\theta_{(1)})$ in case of rule R_1 (R_3); and rules R_2 and R_4 are applicable over the entire parameter space, Ω . That is, the probability of a correct selection will be no less than P^* . If $k = 2$, the two selection rules R_1 and R_2 are equivalent, as are R_3 and R_4 , since $T_1 + T_2$ is a constant.

Rizvi and Woodworth [7] and McDonald [2] present a class of distributions for

which the LFC for selection rules R_1 and R_3 does not occur when all of the population (location) parameters are equal. This article will investigate the LFC for selection rule R_1 for two cases: normal populations differing in their mean values (Section 2), and the counterexample distributions also differing in location parameters (Section 3). The normal distribution is chosen as an example of a widely used distribution in practice that is unimodal and symmetric. The counterexample distribution (bi-uniform distribution) is chosen since it is the singular example of a family of distributions which does not possess the “usual” property for the LFC and hence limits an unqualified endorsement for the nonparametric subset selection procedures (R_1 and R_3) herein reviewed. The examples examined will be for $k = 3$ and $n = 2$ so as to facilitate exact calculations. A summary and conclusions are given in Section 4.

2. Normal Populations

We begin by deriving a general expression for $P_{123} = \Pr(X_1 \leq X_2 \leq X_3)$, where X_i follows a normal distribution with mean μ_i and standard deviation σ_i , $i = 1, 2, 3$, and the three variates are independent.

$$\begin{aligned} P_{123} &= \Pr\left[\frac{(X_1 - \mu_2)}{\sigma_2} \leq \frac{(X_2 - \mu_2)}{\sigma_2} \leq \frac{(X_3 - \mu_2)}{\sigma_2}\right] \\ &= \int_{-\infty}^{\infty} \Pr\left(\frac{X_1 - \mu_2}{\sigma_2} \leq x\right) \cdot \Pr\left(x \leq \frac{X_3 - \mu_2}{\sigma_2}\right) \varphi(x) dx \\ &= \int_{-\infty}^{\infty} \Phi\left(\frac{(\mu_2 - \mu_1 + x \cdot \sigma_2)}{\sigma_1}\right) \cdot \Phi\left(\frac{(\mu_3 - \mu_2 - x \cdot \sigma_2)}{\sigma_3}\right) \varphi(x) dx, \end{aligned} \quad (2.1)$$

where $\Phi(x)$ and $\varphi(x)$ are the cumulative distribution function (cdf) and probability density function (pdf), respectively, of a standard normal random variable (mean = 0, standard deviation = 1). While Equation (2.1) and **Appendix A** is structured for three independent normal distributions, they can be easily adjusted to handle other location-scale families of distributions such as the logistic, t-distributions, etc.

There are six permutations of the digits 1, 2, and 3 and hence the orderings of X_1 , X_2 , and X_3 . Order 1 is $(X_1 \leq X_2 \leq X_3)$; Order 2 is $(X_1 \leq X_3 \leq X_2)$; Order 3 is $(X_2 \leq X_1 \leq X_3)$; Order 4 is $(X_2 \leq X_3 \leq X_1)$; Order 5 is $(X_3 \leq X_1 \leq X_2)$; and Order 6 is $(X_3 \leq X_2 \leq X_1)$. The probabilities of these orders are designated by: Prob1, Prob2, ..., Prob6, respectively. These six probabilities are calculated with the R-code in **Appendix A** using (2.1) matching the specific population means and standard deviations with the appropriate permutation of the population indices (1, 2, 3). **Appendix A**, as herein stated, is set for the three means equal to 0 and the three standard deviations set equal to 1. These probabilities are the probabilities of the possible six orderings of the random draws from the three populations in each of the two blocks. There are thus 36 possible outcomes in the ordering of the data in a randomized block design for three populations and two blocks. The probability of permutation 1 for block one and permutation 2 for block two is, by block independence, Prob1 times Prob2. Similar calculations hold for the other thirty-five joint occurrences for the permutations. The probabilities of these 36 joint per-

mutation events are given as the `df1` data.frame output. The column under `Prob1` are the probabilities of the six joint occurrences of permutation 1 with each of the six permutations 1 through 6 in that order. Similarly, the same type probabilities occur under the columns `Prob2` through `Prob6`.

The `df1` data.frame provides the probabilities for the sample space of all possible rank orderings of a randomized block designed experiment with three normal populations and two blocks. The rank sums of the three populations can now be calculated along with the corresponding probabilities and from these, the probability mass function (pmf) and cumulative distribution function (cdf) of the statistic $S = T_{\max} - T_3$ can be determined, where $T_{\max} = \max(T_i, i = 1, 2, 3)$. Note that a CS, using R_1 , occurs when $T_3 \geq T_{\max} - b_1$, or $S \leq b_1$. Thus, the cdf of the statistic S is the relevant quantity linking the value of b_1 with the preassigned constant P^* .

To illustrate the appropriate computation, suppose both blocks 1 and 2 yield Order 1. Then $T_1 = 2$, $T_2 = 4$, and $T_3 = 6$, and so $S = 0$ with probability 0.5. The statistic S can assume only the values $d = 0, 1, 2, 3, 4$. The subsets of the 36 permutation configurations that result in values of 0, 1, 2, 3, 4 are given in **Appendix A** and, coupled with the probabilities of the configurations, yield the pmf of S (denoted by $\text{Pr}0, \dots, \text{Pr}4$) and the cdf of S (denoted by $\text{cumdf}0, \dots, \text{cumdf}4$) displayed in **Table 1**.

Table 1. Pmf and cdf for the statistic $S = T_{\max} - T_3$ with $\mu = c(0, 0, 0)$ and $\sigma = c(1, 1, 1)$.

d	$\text{Pr}(S = d)$	$\text{Pr}(S \leq d)$
0	0.5	0.5
1	0.11111	0.61111
2	0.16667	0.77778
3	0.16667	0.94444
4	0.05556	1

Implementing the subset selection rule R_1 given in (1.5) requires the calculation of b_1 so that (1.3) is satisfied, *i.e.*, that the probability of a correct selection is no less than a prescribed P^* for any underlying population parameter configurations. When the populations are normally distributed differing only in their mean values, the LFC is that with all mean values equal. This is shown for several specific cases. The first is the slippage configuration, where two populations have an equal mean value of 0, and the third has a mean value $\delta \geq 0$. The R-code in **Appendix A** is run with standard deviations equal to 1 and the mean values set at $c(0, 0, \delta)$ with $\delta = 0 (0.1) 5.2$. The values of the $\text{cumdf}0, \dots, \text{cumdf}4$ are entered into the R-code of **Appendix B** as $\text{Pr}0, \dots, \text{Pr}4$ which then generates **Figure 1** left sketch.

The second is the equi-spaced configuration, where the first population has mean value of 0, the second population has mean value δ , and the third mean value 2δ . As before, the R-code of **Appendix A** is run with standard deviations equal to 1 and the mean values set at $c(0, \delta, 2\delta)$ with $\delta = 0 (0.1) 4.2$. The values of

the cumcdf0, ..., cumcdf4 are entered into the R-code of **Appendix C** as Pr0, ..., Pr4 which then generates **Figure 1** right sketch. In both parameter configurations the $\Pr(\text{CS}|\delta)$ is a nondecreasing function of δ for $d = 0$ (1) 4 and is minimized at $\delta = 0$, *i.e.*, when the three populations have identical probability distributions.

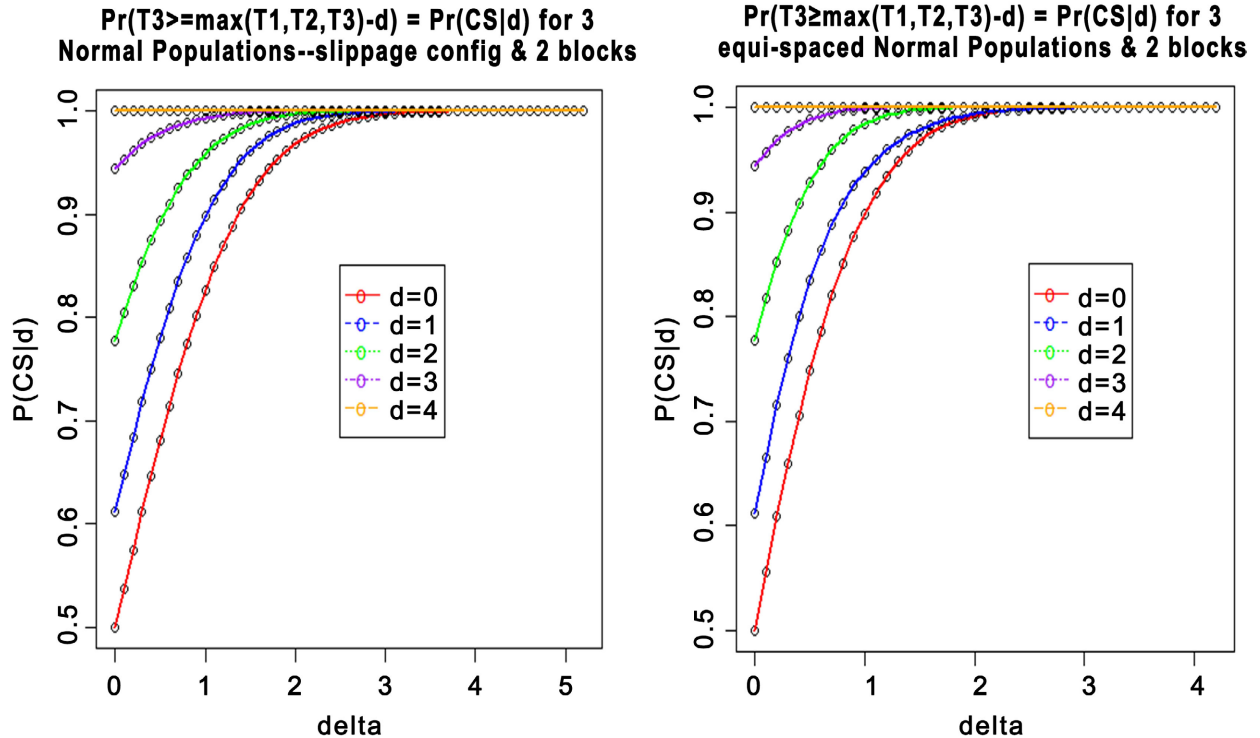


Figure 1. $\Pr(\text{CS})$ for three normal populations with equal variances and slippage configuration (left) and equal spaced configuration (right), $k=3, n=2$.

3. Bi-Uniform Distributions

McDonald [2] gives a class of continuous distribution functions for which the $\Pr(\text{CS})$ for nonparametric selection rule R_1 does not occur when the populations are identically distributed, and hence yields a counterexample to the hypothesis that the LFC occurs when the populations are identically distributed. For $0 < a < 1 < c$, let the cdf $F(x)$ be defined as

$$F(x) = \begin{cases} 0, & x \leq -c \\ (c+x)/2c, & -c < x \leq 0 \\ 1/2, & 0 < x \leq 1 \\ (a+x-1)/2a, & 1 < x \leq 1+a \\ 1, & 1+a < x \end{cases} \quad (3.1)$$

$F(x)$ is composed of two nonoverlapping uniform distributions: one on the interval $(-c, 0)$ and the other on the interval $(1, 1+a)$ and each scaled to have mass 0.5. The counterexample is constructed for $k \geq 3$, the population location parameters are equal with the exception of the two largest which are themselves equal, and n is asymptotically large. A sufficiently large value of “ c ” can be chosen to

establish the counterexample. The case with $n = 2$ is considered next.

The R-code of **Appendix D** generates the cdf of the statistic S based on a large sample of bi-uniform variates as defined in (3.1). Based on input values of c , a , δ_1 , δ_2 , and $nsim$. The values δ_1 , δ_2 are location parameters of the second and third population, and $nsim$ is the number of random draws from the three populations. The $nsim$ draws are structured in pairs so as to provide $nsim/2$ random block experimental designs with three populations and two blocks. The R-codes of **Appendix E** and **Appendix F** generate output similar to that of **Appendix B** and **Appendix C** respectively. The output is displayed in **Figure 2**. The cdf of the statistic S is based on $nsim = 200,000$ in **Appendix D**. The patterns in **Figure 2** are qualitatively the same as those in **Figure 1**. That is, in both parameter configurations the $\Pr(CS|d)$ is a nondecreasing function of δ and is minimized at $\delta = 0$, *i.e.*, when the three populations have identical probability distributions.

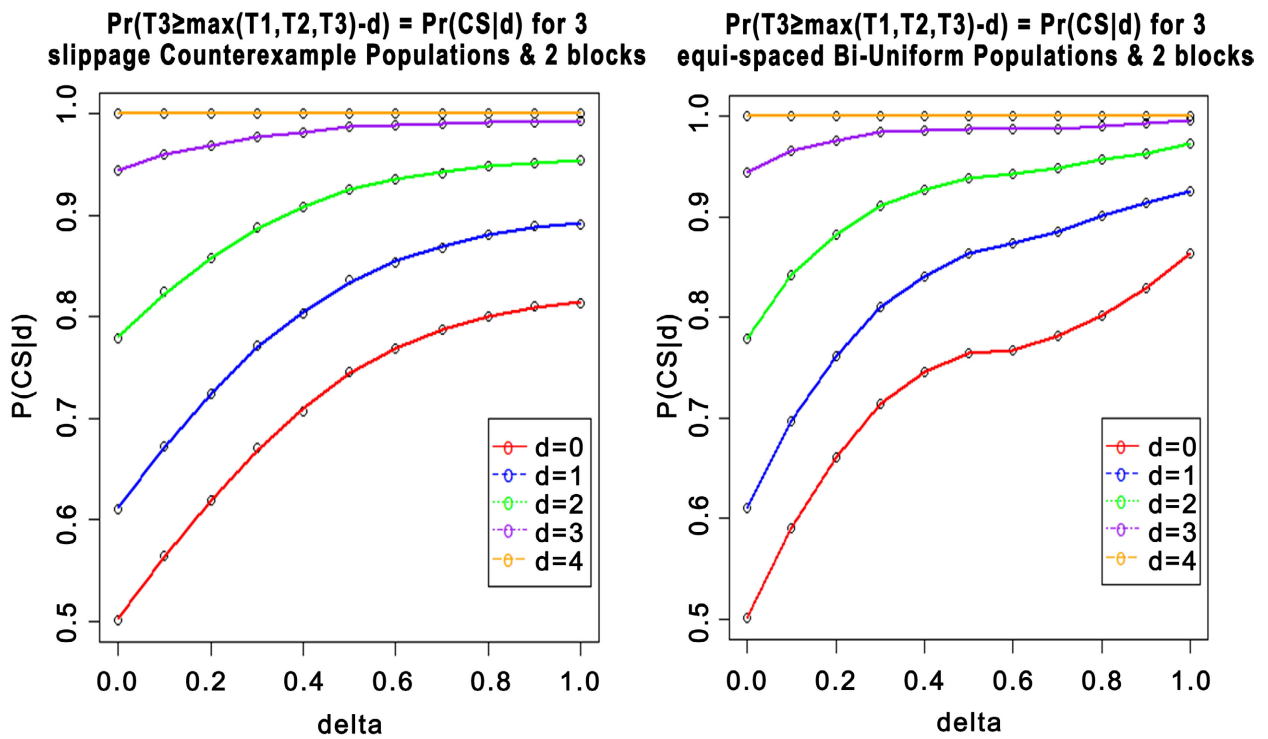


Figure 2. $\Pr(CS)$ for three bi-uniform populations with $c = a = 1$, slippage configuration (left) and equal spaced configuration (right), $k = 3$, $n = 2$.

4. Summary and Conclusions

The results given here in Sections 2 and 3 complement simulation studies investigating the LFC for the subset selection rule given in (1.5). Lorenzen and McDonald [8] executed a large-scale simulation study quantifying the selection probabilities for the subset rule R utilized here with three populations and eight blocks. They considered location parameters with the following distributions: normal, logistic, double exponential, Cauchy, and the exponential with a threshold parameter. As noted, the chosen distributions cover a wide spectrum of tail densities

including one non-symmetric density. They considered two distributions ordered by a scale parameter: the exponential distribution, and the gamma distribution. And, finally, they considered the counterexample distribution given in (3.1). In all cases, they considered three parameter configurations: the slippage, the reverse slippage (*i.e.*, two parameters are equal and larger than the third), and the equal spaced. In all cases investigated, they found that the LFC occurred when the distribution of the three populations were identical.

A critical inequality leading to the counterexample in McDonald [2] is that given by Rizvi and Woodward [7] in their Counterexample 3 Section. For any $\varepsilon > 0$, there exists $0.5 < P_\varepsilon < 1$ such that

$$A = A(P_\varepsilon, k) \leq (1 + \varepsilon) 2^{1/2} \Phi^{-1}(P_\varepsilon), \quad (3.2)$$

and A is given by

$$\int_{-\infty}^{\infty} \Phi^{k-1}(x + A) \varphi(x) dx = P_\varepsilon. \quad (3.3)$$

For $k = 3$, the value of ε for the counterexample can be shown to be (to 5 dp) 0.06066. Using the R-code in **Appendix G** yields the values of A and P_ε to be 5.0 and 0.99960 (to 5 dp), respectively, to establish the counterexample for large sample sizes (blocks) n . The value of P_ε is excessively large for most practical applications.

The exact small model results given here, along with larger model simulation results, support the use of identical population distributions for the LFC and computation of P^* for the lower bound of the $\text{Pr}(\text{CS})$.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Gupta, S.S. and McDonald, G.C. (1970) On Some Classes of Selection Procedures Based on Ranks. In: Puri, M.L., Ed., *Nonparametric Techniques in Statistical Inference*, Cambridge University Press, 491-513.
- [2] McDonald, G.C. (1972) Some Multiple Comparison Selection Procedures Based on Ranks. *Sankhya: The Indian Journal of Statistics Series A*, **34**, 53-64.
- [3] McDonald, G.C. and Alsaeed, S. (2024) Comparison of Block Design Nonparametric Subset Selection Rules Based on Alternative Scoring Rules. *Applied Mathematics*, **15**, 355-389. <https://doi.org/10.4236/am.2024.155022>
- [4] Lehmann, E.L. (1959) *Testing Statistical Hypotheses*. John Wiley & Sons, Inc.
- [5] McDonald, G.C. (2016) Applications of Subset Selection Procedures and Bayesian Ranking Methods in Analysis of Traffic Fatality Data. *WIREs Computational Statistics*, **8**, 222-237. <https://doi.org/10.1002/wics.1385>
- [6] Gupta, S.S. and Panchapakesan, S. (1979) *Multiple Decision Procedures*. John Wiley & Sons, Inc. <https://epubs.siam.org/doi/pdf/10.1137/1.9780898719161.fm>
- [7] Rizvi, M.H. and Woodward, G.G. (1970) On Selection Procedures Based on Ranks: Counterexamples Concerning Least Favorable Configurations. *Annals of Mathematical Statistics*, **41**, 1942-1951. <https://doi.org/10.1214/aoms/1177696695>

- [8] Lorenzen, T.J. and McDonald, G.C. (1984) A Nonparametric Analysis of Urban, Rural, and Interstate Traffic Fatality Rates. In: Santner, T.J. and Tamhane, A.C., Eds., *Design of Experiments-Ranking and Selection*, Marcel Dekker, 143-178.

Appendix A. Ordering Probabilities for Three Independent Normal Variates

```

#General Normal Ordering for k=3 and n=2
#Compute the Pr(X1<X2<X3) where Xi's are independent normal variates with
#means m[i] and standard deviations s[i].
#order 1 is (X1<X2<X3); order 2 is (X1<X3<X2); order 3 is (X2<X1<X3);
#order 4 is (X2<X3<X1); order 5 is (X3<X1<X2); order 6 is (X3<X2<X1).
#Input the values of the means and standard deviations in next line
m=c(0,0,0);s=c(1,1,1)
order<-NULL; error<-NULL
Prob1<-NULL;Prob2<-NULL;Prob3<-NULL;Prob4<-NULL;Prob5<-
NULL;Prob6<-NULL
integrand1<-function(x){
  ((pnorm((s[2]*x+m[2]-m[1])/s[1]))*(pnorm((-x*s[2]+m[3]-
m[2])/s[3]))*dnorm(x,mean=0,sd=1))
}
int1<-integrate(integrand1,lower=-Inf,upper=Inf)
order[1]<-int1$value
error[1]
error[1]<-int1$abs.error
integrand2<-function(x){
  ((pnorm((s[3]*x+m[3]-m[1])/s[1]))*(pnorm((-x*s[3]+m[2]-
m[3])/s[2]))*dnorm(x,mean=0,sd=1))
}
int2<-integrate(integrand2,lower=-Inf,upper=Inf)
order[2]<-int2$value
error[2]
error[2]<-int2$abs.error
integrand3<-function(x){
  ((pnorm((s[1]*x+m[1]-m[2])/s[2]))*(pnorm((-x*s[1]+m[3]-
m[1])/s[3]))*dnorm(x,mean=0,sd=1))
}
int3<-integrate(integrand3,lower=-Inf,upper=Inf)
order[3]<-int3$value
error[3]
error[3]<-int3$abs.error
integrand4<-function(x){
  ((pnorm((s[3]*x+m[3]-m[2])/s[2]))*(pnorm((-x*s[3]+m[1]-
m[3])/s[1]))*dnorm(x,mean=0,sd=1))
}
int4<-integrate(integrand4,lower=-Inf,upper=Inf)
order[4]<-int4$value
error[4]

```

```

error[4]<-int4$abs.error
integrand5<-function(x){
  ((pnorm((s[1]*x+m[1]-m[3])/s[3]))*(pnorm((-x*s[1]+m[2]-
m[1])/s[2]))*dnorm(x,mean=0,sd=1))
}
int5<-integrate(integrand5,lower=-Inf,upper=Inf)
order[5]<-int5$value
order[5]
error[5]<-int5$abs.error
integrand6<-function(x){
  ((pnorm((s[2]*x+m[2]-m[3])/s[3]))*(pnorm((-x*s[2]+m[1]-
m[2])/s[1]))*dnorm(x,mean=0,sd=1))
}
int6<-integrate(integrand6,lower=-Inf,upper=Inf)
order[6]<-int6$value
order[6]
error[6]<-int6$abs.error
df<-data.frame(order,error)
df
Total<-sum(order)
Total
#ProbX is the probability that order X & order[i] occur in 2 samples (blocks)
for (i in 1:6){
  Prob1[i]<-order[1]*order[i]
}
for (i in 1:6){
  Prob2[i]<-order[2]*order[i]
}
for (i in 1:6){
  Prob3[i]<-order[3]*order[i]
}
for (i in 1:6){
  Prob4[i]<-order[4]*order[i]
}
for (i in 1:6){
  Prob5[i]<-order[5]*order[i]
}
for (i in 1:6){
  Prob6[i]<-order[6]*order[i]
}
df1<-data.frame(Prob1,Prob2,Prob3,Prob4,Prob5,Prob6)
round(df1,5)
sum(Prob1,Prob2,Prob3,Prob4,Prob5,Prob6)

```

```

#PrX is the probability that Tmax-T3 = X for X=0,1,2,3,4
Pr0<-Prob1[1]+Prob1[2]+Prob1[3]+Prob1[4]+Prob1[6]+Prob2[1]+
  Prob2[3]+Prob2[4]+Prob3[1]+Prob3[2]+Prob3[3]+
  Prob3[4]+Prob3[5]+Prob4[1]+Prob4[2]+Prob4[3]+
  Prob5[3]+Prob6[1]
Pr1<-Prob1[5]+Prob3[6]+Prob5[1]+Prob6[3]
Pr2<-Prob2[2]+Prob2[6]+Prob4[4]+Prob4[5]+Prob5[4]+Prob6[2]
Pr3<-Prob2[5]+Prob4[6]+Prob5[2]+Prob5[6]+Prob6[4]+Prob6[5]
Pr4<-Prob5[5]+Prob6[6]
df2<-data.frame(Pr0,Pr1,Pr2,Pr3,Pr4)
round(df2,5)
#cumdfX is the cdf of Tmax-T3 for X=0,1,2,3,4
cumdf0<-Pr0
cumdf1<-cumdf0+Pr1
cumdf2<-cumdf1+Pr2
cumdf3<-cumdf2+Pr3
cumdf4<-cumdf3+Pr4
df3<-data.frame(cumdf0,cumdf1,cumdf2,cumdf3,cumdf4)
round(df3,5)

```

Appendix B. Pr(CS) for Three Independent Normal Populations with Slippage Configuration (Figure 1 Left Panel)

```

#norm_ord_computations using normal ordering.txt
#slippage parameter configurations
#mu=(0,0,delta)
require(splines)
delta<-seq(from=0,to=5.2,by=0.1)
delta
length(delta)
Pr0<-c(0.5,0.53756,0.57477,0.61131,0.64687,0.68115,0.71391,
  0.74494,0.77405,0.80114,0.82612,0.84895,0.86964,0.88822,
  0.90478,0.91942,0.93224,0.94338,0.95299,0.96121,0.96819,
  0.97407,0.97899,0.98308,0.98645,0.98922,0.99146,0.99328,
  0.99474,0.9959,0.99683,0.99756,0.99813,0.99857,0.99892,
  0.99918,0.99939,0.99954,0.99966,0.99975,0.99982,0.99987,
  0.9999,0.99993,0.99995,0.99996,0.99997,0.99998,0.99999,
  0.99999,0.99999,1,1)
length(Pr0)
plot(delta,Pr0)
fit0<-smooth.spline(delta,Pr0)
lines(fit0,lwd=2,col="red")
Pr1<-c(0.61111,0.64818,0.68394,0.71811,0.75042,0.78069,0.80877,

```



```

title("Pr(T3>=max(T1,T2,T3)-d) = Pr(CS|d)
for 3 Normal Populations--slippage config & 2 blocks")
points(delta,Pr1)
fit1<-smooth.spline(delta,Pr1)
lines(fit1,lwd=2,col="blue")
points(delta,Pr2)
fit2<-smooth.spline(delta,Pr2)
lines(fit2,lwd=2,col="green")
points(delta,Pr3)
fit3<-smooth.spline(delta,Pr3)
lines(fit3,lwd=2,col="purple")
points(delta,Pr4)
fit4<-smooth.spline(delta,Pr4)
lines(fit4,lwd=2,col="orange")
legend(2.5,0.85,legend=c("d = 0","d = 1","d = 2","d = 3","d = 4"),
col=c("red","blue","green","purple","orange"),pch=c("o","o","o","o","o"),
lty=c(1,2,3,4,5),ncol=1)

```

Appendix C. Pr(CS) for Three Independent Normal Populations with Equi-Spaced Configuration (Figure 1 Right Panel)

```

#norm_ord_computations using normal ordering.txt
#equi-spaced parameter configurations
#mu=(0,delta,2*delta)
require(splines)
delta<-seq(from=0,to=4.2,by=0.1)
delta
length(delta)
Pr0<-c(0.5,0.55563,0.60899,0.65921,0.70571,0.74812,0.78629,
0.82026,0.85017,0.87625,0.89875,0.91797,0.93421,0.94777,
0.95895,0.96806,0.97539,0.98122,0.98579,0.98933,0.99205,
0.99412,0.99568,0.99684,0.99771,0.99835,0.99881,0.99915,
0.9994,0.99958,0.9997,0.9998,0.99986,0.9999,0.99993,
0.99996,0.99997,0.99998,0.99999,0.99999,0.99999,1,1)
length(Pr0)
plot(delta,Pr0)
fit0<-smooth.spline(delta,Pr0)
lines(fit0,lwd=2,col="red")
Pr1<-c(0.61111,0.66562,0.71566,0.76058,0.80009,0.83419,0.8632,
0.88756,0.90786,0.92467,0.93856,0.95001,0.95946,0.96725,
0.97367,0.97895,0.98328,0.98681,0.98967,0.99197,0.99381,
0.99527,0.99641,0.9973,0.99799,0.99851,0.99891,0.99921,
0.99943,0.99959,0.99971,0.9998,0.99986,0.9999,0.99993,

```

```

0.99996,0.99997,0.99998,0.99999,0.99999,0.99999,1,1)
length(Pr1)
par(ask=TRUE)
plot(delta,Pr1)
fit1<-smooth.spline(delta,Pr1)
lines(fit1,lwd=2,col="blue")
Pr2<-c(0.77778,0.81764,0.85252,0.88246,0.90774,0.92868,0.94573,
0.95938,0.97011,0.97839,0.98466,0.98932,0.9927,0.99512,
0.9968,0.99795,0.99871,0.99921,0.99953,0.99972,0.99984,
0.99991,0.99995,0.99997,0.99999,0.99999,1,1,1,1,1,1,1,
1,1,1,1,1,1,1)
length(Pr2)
par(ask=TRUE)
plot(delta,Pr2)
fit2<-smooth.spline(delta,Pr2)
lines(fit2,lwd=2,col="green")
Pr3<-c(0.94444,0.95735,0.96797,0.97649,0.9832,0.98829,0.99207,
0.99478,0.99667,0.99794,0.99876,0.99928,0.9996,0.99978,
0.99989,0.99994,0.99997,0.99999,0.99999,1,1,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,1)
length(Pr3)
par(ask=TRUE)
plot(delta,Pr3)
fit3<-smooth.spline(delta,Pr3)
lines(fit3,lwd=2,col="purple")
Pr4<-c(rep(1,43))
length(Pr4)
par(ask=TRUE)
plot(delta,Pr4)
fit4<-smooth.spline(delta,Pr4)
lines(fit4,lwd=2,col="orange")
#####
par(ask=TRUE)
plot(delta,Pr0,ylab="P(CS|d)")
fit0<-smooth.spline(delta,Pr0)
lines(fit0,lwd=2,col="red")
title("Pr(T3>=max(T1,T2,T3)-d) = Pr(CS|d)
for 3 equi-spaced Normal Populations & 2 blocks")
points(delta,Pr1)
fit1<-smooth.spline(delta,Pr1)
lines(fit1,lwd=2,col="blue")
points(delta,Pr2)
fit2<-smooth.spline(delta,Pr2)

```

```

lines(fit2,lwd=2,col="green")
points(delta,Pr3)
fit3<-smooth.spline(delta,Pr3)
lines(fit3,lwd=2,col="purple")
points(delta,Pr4)
fit4<-smooth.spline(delta,Pr4)
lines(fit4,lwd=2,col="orange")
legend(2.5,0.85,legend=c("d = 0","d = 1","d = 2","d = 3","d = 4"),
col=c("red","blue","green","purple","orange"),pch=c("o","o","o","o","o"),
lty=c(1,2,3,4,5),ncol=1)

```

Appendix D. Ordering Probabilities for Three Independent Bi-Uniform Variates

```

#Bi-Uniform for nonparametric block design, k=3,n=2
#See McDonald (1972), Sankhya, Series A, Vol. 34, Part 1, pp.53-64
#The distribution is a mixture of two uniform densities
#Population 1: Uniform(-c,0) and uniform(1,1+a), c>-0 and a>-0
#Population 2: Uniform(-c+delta1,delta1) and uniform(1+delta1,1+a+delta1),
#c>0, a>0, delta1>=0
#Population 3: Uniform(-c+delta2,delta2) and uniform(1+delta2,1+a+delta2),
#c>0, a>0, delta2>=delta1
#set c, a, delta1, delta2 and number of random draws (nsim)
c<-5;a<-1;delta1<-0;delta2<-0;nsim<-200000
y1=NULL;y2=NULL;y3=NULL
z1=NULL;z2=NULL;z3=NULL
p1=NULL;p2=NULL;p3=NULL
for (i in 1:nsim){
y1[i]<-runif(1,-c,0)
z1[i]<-runif(1,1,1+a)
p1[i]<-rbinom(1,1,0.5)
x1[i]<-p1[i]*y1[i] + (1-p1[i])*z1[i]
}
for (j in 1:nsim){
y2[j]<-runif(1,-c+delta1,delta1)
z2[j]<-runif(1,1+delta1,1+a+delta1)
p2[j]<-rbinom(1,1,0.5)
x2[j]<-p2[j]*y2[j] + (1-p2[j])*z2[j]
}
for (k in 1:nsim){
y3[k]<-runif(1,-c+delta2,delta2)
z3[k]<-runif(1,1+delta2,1+a+delta2)
p3[k]<-rbinom(1,1,0.5)
x3[k]<-p3[k]*y3[k] + (1-p3[k])*z3[k]
}

```

```
#####  
df<-data.frame(x1,x2,x3)  
head(df,5)  
tail(df,5)  
avg<-c(mean(x1),mean(x2),mean(x3))  
std<-c(sd(x1),sd(x2),sd(x3))  
var<-std^2  
avg  
std  
#####  
r1=NULL;r2=NULL;rk1=NULL;rk2=NULL;add=NULL  
d=NULL  
n1<-floor((nsim+1)/2)  
for (i in 1:n1){  
j<-2*i-1  
r1<-c(x1[j],x2[j],x3[j])  
r2<-c(x1[j+1],x2[j+1],x3[j+1])  
rk1<-rank(r1)  
rk2<-rank(r2)  
add<-rk1+rk2  
d[i]<-max(add)-add[3]  
}  
#####  
pd<-table(d)  
pd  
sum(pd)  
den<-pd/sum(pd)  
den  
cdf<-cumsum(den)  
cdf  
summary<-data.frame(den,cdf)  
summary
```

Appendix E. Pr(CS) for Three Independent Bi-Uniform Populations with Slippage Configuration (Figure 2 Left Panel)

```
#Computations using Counterexample Distribution k=3,n=2  
#slippage parameter configurations  
#c=a=1; 200,000 simulations  
# theta=(0,0,delta)  
require(splines)  
delta<-seq(from=0,to=1.0,by=0.1)  
delta
```

```

length(delta)
###
Pr0<-c(0.50035,0.56392,0.61917,0.67020,0.70652,
0.74557,0.76886,0.78770,0.80021,0.81007,0.81374)
length(Pr0)
plot(delta,Pr0)
fit0<-smooth.spline(delta,Pr0)
lines(fit0,lwd=2,col="red")
Pr1<-c(0.61072,0.67262,0.72432,0.77131,0.80291,
0.83586,0.85383,0.86842,0.88060,0.88930,0.89132)
length(Pr1)
par(ask=TRUE)
plot(delta,Pr1)
fit1<-smooth.spline(delta,Pr1)
lines(fit1,lwd=2,col="blue")
Pr2<-c(0.77821,0.82438,0.85743,0.88818,0.90752,
0.92597,0.93527,0.94186,0.94859,0.95162,0.95389)
length(Pr2)
par(ask=TRUE)
plot(delta,Pr2)
fit2<-smooth.spline(delta,Pr2)
lines(fit2,lwd=2,col="green")
Pr3<-c(0.94404,0.95971,0.96915,0.97726,0.98219,
0.98690,0.98899,0.99011,0.99115,0.99182,0.99229)
length(Pr3)
par(ask=TRUE)
plot(delta,Pr3)
fit3<-smooth.spline(delta,Pr3)
lines(fit3,lwd=2,col="purple")
Pr4<-c(rep(1,11))
length(Pr4)
par(ask=TRUE)
plot(delta,Pr4)
fit4<-smooth.spline(delta,Pr4)
lines(fit4,lwd=2,col="orange")
#####
par(ask=TRUE)
plot(delta,Pr0,ylab="P(CS|d)",xlim=c(0,1),ylim=c(0.5,1))
fit0<-smooth.spline(delta,Pr0)
lines(fit0,lwd=2,col="red")
title("Pr(T3>=max(T1,T2,T3)-d) = Pr(CS|d)
for 3 slippage Counterexample Populations & 2 blocks")
points(delta,Pr1)

```

```

fit1<-smooth.spline(delta,Pr1)
lines(fit1,lwd=2,col="blue")
points(delta,Pr2)
fit2<-smooth.spline(delta,Pr2)
lines(fit2,lwd=2,col="green")
points(delta,Pr3)
fit3<-smooth.spline(delta,Pr3)
lines(fit3,lwd=2,col="purple")
points(delta,Pr4)
fit4<-smooth.spline(delta,Pr4)
lines(fit4,lwd=2,col="orange")
legend(0.8,0.7,legend=c("d = 0","d = 1","d = 2","d = 3","d = 4"),
col=c("red","blue","green","purple","orange"),pch=c("o","o","o","o","o"),
lty=c(1,2,3,4,5),ncol=1)

```

Appendix F. Pr(CS) for Three Independent Bi-Uniform Populations with Equi-Spaced Configuration (Figure 2 Right Panel)

```

#Computations using Counterexample Distribution k=3,n=2
#equi-spaced parameter configurations
#c=a=1; 200,000 simulations
# theta=(0,delta,2*delta)
require(splines)
delta<-seq(from=0,to=1.0,by=0.1)
delta
length(delta)
###
Pr0<-c(0.50035,0.59002,0.66110,0.71370,0.74518,
0.76439,0.76724,0.78088,0.80236,0.82969,0.86351)
length(Pr0)
plot(delta,Pr0)
fit0<-smooth.spline(delta,Pr0)
lines(fit0,lwd=2,col="red")
Pr1<-c(0.61072,0.69698,0.76194,0.81093,0.84024,
0.86286,0.87304,0.88545,0.90072,0.91396,0.92584)
length(Pr1)
par(ask=TRUE)
plot(delta,Pr1)
fit1<-smooth.spline(delta,Pr1)
lines(fit1,lwd=2,col="blue")
Pr2<-c(0.77821,0.84163,0.88157,0.91152,0.92619,
0.93778,0.94237,0.94811,0.95687,0.96347,0.97230)
length(Pr2)

```

```

par(ask=TRUE)
plot(delta,Pr2)
fit2<-smooth.spline(delta,Pr2)
lines(fit2,lwd=2,col="green")
Pr3<-c(0.94404,0.96501,0.97582,0.98380,0.98613,
0.98695,0.98671,0.98726,0.98959,0.99269,0.99564)
length(Pr3)
par(ask=TRUE)
plot(delta,Pr3)
fit3<-smooth.spline(delta,Pr3)
lines(fit3,lwd=2,col="purple")
Pr4<-c(rep(1,11))
length(Pr4)
par(ask=TRUE)
plot(delta,Pr4)
fit4<-smooth.spline(delta,Pr4)
lines(fit4,lwd=2,col="orange")
#####
par(ask=TRUE)
plot(delta,Pr0,ylab="P(CS|d)",xlim=c(0,1),ylim=c(0.5,1))
fit0<-smooth.spline(delta,Pr0)
lines(fit0,lwd=2,col="red")
title("Pr(T3>=max(T1,T2,T3)-d) = Pr(CS|d)
for 3 equi-spaced Bi-Uniform Populations & 2 blocks")
points(delta,Pr1)
fit1<-smooth.spline(delta,Pr1)
lines(fit1,lwd=2,col="blue")
points(delta,Pr2)
fit2<-smooth.spline(delta,Pr2)
lines(fit2,lwd=2,col="green")
points(delta,Pr3)
fit3<-smooth.spline(delta,Pr3)
lines(fit3,lwd=2,col="purple")
points(delta,Pr4)
fit4<-smooth.spline(delta,Pr4)
lines(fit4,lwd=2,col="orange")
legend(0.8,0.7,legend=c("d = 0","d = 1","d = 2","d = 3","d = 4"),
col=c("red","blue","green","purple","orange"),pch=c("o","o","o","o","o"),
lty=c(1,2,3,4,5),ncol=1)

```

Appendix G. Calculation of A and P_ϵ to Satisfy Equation (3.2)

```

#For a given k and epsilon, solves for A and P
#Specify k, epsilon, and an appropriate seq for A

```

```
k<-3;epsilon<-0.06066
integrand<-function(x,A=0){((pnorm(x+A))^(k-1))*dnorm(x)}
int<-function(A){integrate(integrand,lower=-Inf,upper=Inf,A=A)$value}
A = seq(0,10,by=0.5)
len<-length(A)
pstar<-sapply(A,int)
df<-data.frame(A,pstar)
ipstar<-qnorm(pstar)
bound<-ipstar*sqrt(2)*(1+epsilon)
diff<-bound-A
print("If diff>0, then A<bound and counterexample inequality holds")
df1<-data.frame(A,pstar,ipstar,bound,diff)
df1
plot(A,bound,main=paste("bound vs. A, k = ",k," , epsilon = ",epsilon,
"\nline is A = bound"), ylim=c(bound[1],bound[len]))
abline(a = 0,b = 1)
```