Supplementary material of manuscript:

"Exact Permutation and Bootstrap Distribution of Generalized Pairwise Comparison statistics"

Test Code Validation

William N. Anderson and Johan Verbeeck

February 22, 2023

This document contains validation for the Permutation and two-sample bootstrap algorithm, in the Generalized Paired Comparison context.

- The *input* data to the analysis functions are a previously computed U matrix, and a vector of trial arms.
- The *output* is the expected win sums W_T and W_C and their variance from a permutation or two-sample bootstrap sample. These values are also the mean and variance of the win sums if all possible permutation or bootstrap samples were taken, each occurring once. In this context, a permutation sample is formed by permuting the trial arm assignments among the patients, while preserving the trial arm counts. The bootstrap samples are taken from the two trial arms separately.
- The matrix U is an $n \times n$ skew matrix with arbitrary real values. In these examples the rows and columns of the skew matrix are labeled with the trial arms; that labeling is for the reader's convenience, but the labels are not used in the computational algorithm.
- The trial arm vector has the value 1 for Treatment and 0 for Control. These are numeric values.
- The code will also work in the $\{1,0,-1\}$ case, with no special parameter choices. (Special code for the $\{1,0,-1\}$ case would be somewhat faster, but such code is not considered here.)

Two different versions of the algorithm are used in this file:

- The full permutation or bootstrap algorithm, which means that all possible permutations or bootstrap samples of the trial arm assignments are explicitly generated and the win sums evaluated. In this context, only permutations preserving the trial arm counts are considered. The bootstrapping is separate by trial arm.
  - By definition this is the fundamental algorithm. Every other computation is trying to more efficiently produce the results of this computation.

  - The results of the practical algorithm must agree exactly (to within roundoff error) with the full algorithm. However, the full algorithm can be run only for very small data sets. In the examples below we go up to 20 x 20 matrices.

- The practical algorithm developed in (Anderson and Verbeeck 2023). This is an $O(n^2)$ algorithm in time and space.

Two other algorithms can be also considered. These are not examined in this document. Details are available from the authors.

- The conceptual algorithm, upon which the practical algorithm is based. This algorithm follows the manuscript closely, and each edge pair is explicitly examined. This is an $O(n^4)$ algorithm, because the number of edges is $O(n^2)$, and hence the number of edge pairs is $O(n^4)$. Use of this algorithm would allow somewhat larger data sets, but the $O(n^4)$ behavior still limits use to reasonably small data sets.

- The randomized algorithm, in which a predetermined number of randomized permutations or bootstrap samples is evaluated, and the means and variances of the win sums are computed. This is an $O(B \times n^2)$ algorithm, where B is the number of permutation or bootstrap samples. The results of this algorithm should be close to those of the practical algorithm, but there is the inevitable sampling error. Convergence to the expected values is agonizingly slow.

The examples below are of two types

- Minimal examples to illustrate the various cases presented in the manuscript.

- Some small examples that combine various cases. Only very small examples can be presented, because of the exponential nature of the full permutation algorithm.

- Each example is is presented twice: once with a $\{1,0,-1\}$ U matrix, and once with a more general skew matrix.

For each example below, the following items are presented:

- The observed win sums W_T (Test) and W_C (Control)

- The expected win sums from the practical algorithm, and the mean win sums from the full algorithm.

  - These must agree.

  - We note that in the permutation context, the expected numbers are the same for W_T and W_C, and generally will not be the same as the observed win sums.

- The variance of the pair (W_T, W_C) from the practical algorithm, and from the full algorithm.

  - These must agree.

Examples – Error checking

Some data error checks are built into the practical algorithm. These are tested below for just that algorithm.

- In each case the algorithm must generate an error message for the specific example, but allow execution of the remaining examples to continue.

Win matrix non-zero on diagonal

```
##
##  The matrix

##      [,1] [,2]
## [1,]    1   -1
## [2,]    1    0

##
##  The trial arms

## [1] 0 1

## Error in checkmatrixandarms(winmatrix, trialarms) : Win matrix not skew
```

off diagonal entries are not skew

```
##
##  The matrix

##      [,1] [,2]
## [1,]    0   -1
## [2,]    2    0

##
##  The trial arms

## [1] 0 1

## Error in checkmatrixandarms(winmatrix, trialarms) : Win matrix not skew
```

dimensions incompatible

```
##
##  The matrix

##      [,1] [,2]
## [1,]    0   -2
## [2,]    2    0

##
##  The trial arms

## [1] 0 1 1

## Error in checkmatrixandarms(winmatrix, trialarms) :
##   Incompatible dimensions in win matrix and trial arms
```

trial arms other than 0, 1

```
##
##  The matrix

##      [,1] [,2]
## [1,]    0   -2
## [2,]    2    0

##
##  The trial arms
```

```
## [1] 1 2

## Error in checkmatrixandarms(winmatrix, trialarms) : Invalid trial arms
```

S1 Permutation

S1.1 Example manuscript

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0  -2   0   0   1
## T_2   2   0   3   0  -5
## C_3   0  -3   0   4   0
## C_4   0   0  -4   0  -1
## C_5  -1   5   0   1   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               4               5
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             4.8             4.8

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            6.96           -1.54
## Control Win Sum        -1.54            5.56
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             4.8             4.8
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            6.96           -1.54
## Control Win Sum        -1.54            5.56
```

S1.2 Other Examples

The first set of examples examines separately all the cases in the algorithm. The cases in the first set are separate cases in the bootstrap algorithm; some of those cases are combined in the Permutation test scenario, but both examples are shown in this file.

- For each original example, one or more similar examples are created by varying the values of the skew matrices. Only integer values are used, to aid in reading the matrices. The algorithm makes no use of the fact that the values are integers.

Tiny Example 1: One edge (case 1)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   0   1
## T_2   0   0   0   0   0
## C_3   0   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1                0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.3             0.3

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.21           -0.09
## Control Win Sum        -0.09            0.21
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.3             0.3
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.21           -0.09
## Control Win Sum        -0.09            0.21
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   0   3
## T_2   0   0   0   0   0
## C_3   0   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -3   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               3                0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.9             0.9

## $Variance
##                 Test Win Sum Control Win Sum
```

```
## Test Win Sum            1.89           -0.81
## Control Win Sum        -0.81            1.89
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.9             0.9
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            1.89           -0.81
## Control Win Sum        -0.81            1.89
```

Tiny Example 3: Two edges with common tail (case 3)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   1   0   1
## T_2   0   0   0   0   0
## C_3  -1   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.64           -0.36
## Control Win Sum        -0.36            0.44
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.64           -0.36
## Control Win Sum        -0.36            0.44
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   4   0   2
## T_2   0   0   0   0   0
## C_3  -4   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -2   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               6                0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             1.8             1.8

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            5.96           -3.24
## Control Win Sum        -3.24            4.36
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             1.8             1.8
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            5.96           -3.24
## Control Win Sum        -3.24            4.36
```

The same example, but add one more edge at the same vertex

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   4   3   2
## T_2   0   0   0   0   0
## C_3  -4   0   0   0   0
## C_4  -3   0   0   0   0
## C_5  -2   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               9                0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             2.7             2.7

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           11.81           -7.29
## Control Win Sum        -7.29            6.61
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             2.7             2.7
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           11.81           -7.29
## Control Win Sum        -7.29            6.61
```

Tiny Example 4: Two edges with common head (case 2)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   0   0
## T_2   0   0   0   1   0
## T_3   0   0   0   1   0
## C_4   0  -1  -1   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.64           -0.36
## Control Win Sum        -0.36            0.44
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.64           -0.36
## Control Win Sum        -0.36            0.44
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   0   0
## T_2   0   0   0   5   0
## T_3   0   0   0   7   0
## C_4   0  -5  -7   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              12               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             3.6             3.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           23.24          -12.96
## Control Win Sum       -12.96           16.24
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             3.6             3.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           23.24          -12.96
## Control Win Sum       -12.96           16.24
```

Tiny Example 5: Two edges with common head (case 2)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -1  -1
## T_2   0   0   0   0   0
## C_3   0   0   0   0   0
## C_4   1   0   0   0   0
## C_5   1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.36
## Control Win Sum        -0.36            0.64
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.36
## Control Win Sum        -0.36            0.64
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0 -11 -13
## T_2   0   0   0   0   0
## C_3   0   0   0   0   0
## C_4  11   0   0   0   0
## C_5  13   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               5               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             3.6             3.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           16.24          -12.96
## Control Win Sum       -12.96           23.24
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             3.6             3.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           16.24          -12.96
## Control Win Sum       -12.96           23.24
```

Tiny Example 6: Two edges with common tail (case 3)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0  -1   0
## T_2   0   0   0  -1   0
## T_3   0   0   0   0   0
## C_4   1   1   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.36
## Control Win Sum        -0.36            0.64
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.36
## Control Win Sum        -0.36            0.64
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0 -11   0
```

```
## T_2   0   0   0 -17   0
## T_3   0   0   0   0   0
## C_4  11  17   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0              28
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             8.4             8.4

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           89.84          -70.56
## Control Win Sum       -70.56          127.24
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             8.4             8.4
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           89.84          -70.56
## Control Win Sum       -70.56          127.24
```

Tiny Example 7: Tail of one edge is head of another (cases 4 and 5)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   1  -1
## T_2   0   0   0   0   0
## T_3   0   0   0   0   0
## C_4  -1   0   0   0   0
## C_5   1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.24           -0.06
## Control Win Sum        -0.06            0.24
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
```

```
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.24           -0.06
## Control Win Sum        -0.06            0.24
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   3  -7
## T_2   0   0   0   0   0
## T_3   0   0   0   0   0
## C_4  -3   0   0   0   0
## C_5   7   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               3               7
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               3               3

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             8.4            -2.7
## Control Win Sum         -2.7             8.4
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               3               3
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             8.4            -2.7
## Control Win Sum         -2.7             8.4
```

The same example, but with more edges at the vertex

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   8   0   3  -7
## T_2  -8   0   0   0   0
## T_3   0   0   0   0  -9
## C_4  -3   0   0   0   0
## C_5   7   0   9   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               3              16
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             8.1             8.1
```

```
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           32.49          -22.71
## Control Win Sum       -22.71           49.89
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             8.1             8.1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           32.49          -22.71
## Control Win Sum       -22.71           49.89
```

Tiny Example 8: Tail of one edge is head of another (cases 4 and 5)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   1   0
## T_2   0   0   0  -1   0
## T_3   0   0   0   0   0
## C_4  -1   1   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.24           -0.06
## Control Win Sum        -0.06            0.24
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.24           -0.06
## Control Win Sum        -0.06            0.24
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0  12   0
## T_2   0   0   0 -11   0
## T_3   0   0   0   0   0
## C_4 -12  11   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              12               11
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             6.9              6.9

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           31.89           -8.01
## Control Win Sum        -8.01           31.89
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             6.9              6.9
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           31.89           -8.01
## Control Win Sum        -8.01           31.89
```

Tiny Example 9: Disjoint edges (case 5)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   1   0
## T_2   0   0   0   0   1
## C_3   0   0   0   0   0
## C_4  -1   0   0   0   0
## C_5   0  -1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2                0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6              0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.16
## Control Win Sum        -0.16            0.44
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6              0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.16
## Control Win Sum        -0.16            0.44
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   3   0
## T_2   0   0   0   0   5
## C_3   0   0   0   0   0
## C_4  -3   0   0   0   0
## C_5   0  -5   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               8               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             2.4             2.4

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            7.44           -2.76
## Control Win Sum        -2.76            7.44
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             2.4             2.4
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            7.44           -2.76
## Control Win Sum        -2.76            7.44
```

Tiny Example 10: Disjoint edges (case 5)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -1   0
## T_2   0   0   0   0  -1
## C_3   0   0   0   0   0
## C_4   1   0   0   0   0
## C_5   0   1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.16
## Control Win Sum        -0.16            0.44
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.16
## Control Win Sum        -0.16            0.44
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -2   0
## T_2   0   0   0   0  -1
## C_3   0   0   0   0   0
## C_4   2   0   0   0   0
## C_5   0   1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               3
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.9             0.9

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            1.09           -0.41
## Control Win Sum        -0.41            1.09
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.9             0.9
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            1.09           -0.41
## Control Win Sum        -0.41            1.09
```

Tiny Example 11: Disjoint edges (case 5)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -1   0
## T_2   0   0   0   0   1
## C_3   0   0   0   0   0
## C_4   1   0   0   0   0
## C_5   0  -1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.6             0.6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.16
## Control Win Sum        -0.16            0.44
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.6             0.6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.44           -0.16
## Control Win Sum        -0.16            0.44
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -6   0
## T_2   0   0   0   0   5
## C_3   0   0   0   0   0
## C_4   6   0   0   0   0
## C_5   0  -5   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               5               6
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             3.3             3.3

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           13.41           -4.89
## Control Win Sum        -4.89           13.41
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             3.3             3.3
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           13.41           -4.89
## Control Win Sum        -4.89           13.41
```

Tiny Example 12: Three pairwise Disjoint edges

```
##     T_1 T_2 C_3 C_4 C_5 T_6
## T_1   0   0   0  -1   0   0
```

```
## T_2   0   0   0   0   1   0
## C_3   0   0   0   0   0   1
## C_4   1   0   0   0   0   0
## C_5   0  -1   0   0   0   0
## T_6   0   0  -1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             0.9             0.9

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.69           -0.21
## Control Win Sum        -0.21            0.69
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             0.9             0.9
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.69           -0.21
## Control Win Sum        -0.21            0.69
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5 T_6
## T_1   0   0   0  -4   0   0
## T_2   0   0   0   0   3   0
## C_3   0   0   0   0   0   2
## C_4   4   0   0   0   0   0
## C_5   0  -3   0   0   0   0
## T_6   0   0  -2   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               3               6
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             2.7             2.7

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            6.61           -2.09
## Control Win Sum        -2.09            6.61
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             2.7             2.7
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            6.61           -2.09
## Control Win Sum        -2.09            6.61
```

Examples – Second set

The second set of examples examines some larger win matrices. For the full permutation analysis it is not realistic to go larger than 20 by 20.

The manuscript example, but with a $\{1,0,-1\}$ *skew matrix*

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0  -1   0   0   1
## T_2   1   0   1   0  -1
## C_3   0  -1   0   1   0
## C_4   0   0  -1   0  -1
## C_5  -1   1   0   1   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##             1.8             1.8

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.76           -0.24
## Control Win Sum        -0.24            0.56
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##             1.8             1.8
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            0.76           -0.24
## Control Win Sum        -0.24            0.56
```

Another example

```
##     T_1 T_2 C_3 C_4 C_5 C_6
## T_1   0   1   0   0   1   0
## T_2  -1   0   1   0   0  -1
## C_3   0  -1   0   1   1   0
## C_4   0   0  -1   0   1   0
```

```
## C_5  -1   0  -1  -1   0   0
## C_6   0   1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##        1.866667        1.866667

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.0488889      -0.8177778
## Control Win Sum   -0.8177778       1.5822222
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##          1.8667          1.8667
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.0488889      -0.8177778
## Control Win Sum   -0.8177778       1.5822222
```

The same example, but with a more general skew matrix

```
##     C_1 T_2 C_3 T_4 C_5 T_6
## C_1   0   1   0   0   2   0
## T_2  -1   0   3   0   0  -4
## C_3   0  -3   0   5   6   0
## T_4   0   0  -5   0   7   0
## C_5  -2   0  -6  -7   0   0
## T_6   0   4   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               5               4
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##        7.466667        7.466667

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum        21.98222       -15.61778
## Control Win Sum    -15.61778        32.64889
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##          7.4667          7.4667
##
```

```
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum         21.98222        -15.61778
## Control Win Sum     -15.61778         32.64889
```

Larger example

In this example all possible pair configurations actually occur

```
##      C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8 C_9 T_10 T_11 T_12 T_13 T_14 T_15 T_16 T_17 T_18 T_19 T_20
## C_1    0   1   1   0   0   0   1   0  -1    0   -1   -1    0    0    0    0    1    0   -1    0
## C_2   -1   0   0   0   1   1  -1   0  -1    0   -1    0    0   -1    0    0    0    0   -1    1
## C_3   -1   0   0   0   0   0   1  -1   0   -1   -1    0   -1    0   -1    0    1   -1    0    0
## C_4    0   0   0   0   0   0   1  -1   0   -1   -1    0   -1   -1    0    1    0    0   -1    0
## C_5    0  -1   0   0   0   0   1  -1  -1    0   -1    0    0   -1    0   -1    0   -1   -1    0
## C_6    0  -1   0   0   0   0   1  -1  -1   -1    0    0    0    0   -1   -1    0    0    0   -1
## C_7   -1   1  -1  -1  -1  -1   0  -1  -1   -1    0    0    0    0   -1    0   -1    0    0   -1
## C_8    0   0   1   1   1   1   1   0  -1   -1    0    0    0    0   -1    0   -1   -1    0   -1
## C_9    1   1   0   0   1   1   1   1   0   -1    1    0    1    0   -1    0    0    0   -1   -1
## T_10   0   0   1   1   0   1   1   1   1    0    1    0    0    0   -1    1    0   -1   -1   -1
## T_11   1   1   1   1   1   0   0   0  -1   -1    0    1   -1    0    0    0    0    0    0    0
## T_12   1   0   0   0   0   0   0   0   0    0   -1    0    1    0    0    0    0    0    0    0
## T_13   0   0   1   1   0   0   0   0  -1    0    1   -1    0    1   -1    0    0    0    0    0
## T_14   0   1   0   1   1   0   0   0   0    0    0    0   -1    0    1    0    0    0    0    0
## T_15   0   0   1   0   0   1   1   1   1    1    0    0    1   -1    0    1   -1    0    0    0
## T_16   0   0   0  -1   1   1   0   0   0   -1    0    0    0    0   -1    0   -1    0    0    0
## T_17  -1   0  -1   0   0   0   1   1   0    0    0    0    0    0    1    1    0    1    0    0
## T_18   0   0   1   0   1   0   0   1   0    1    0    0    0    0    0    0   -1    0    1    0
## T_19   1   1   0   1   1   0   0   0   1    1    0    0    0    0    0    0    0   -1    0    1
## T_20   0  -1   0   0   0   1   1   1   1    1    0    0    0    0    0    0    0    0   -1    0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              38               6
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##        21.62368        21.62368

## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum         29.30467        -23.06112
## Control Win Sum     -23.06112         28.26257
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##          21.624          21.624
##
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum         29.30467        -23.06112
## Control Win Sum     -23.06112         28.26257
```

Larger example with different trial arms

```
##      C_1 T_2 C_3 T_4 C_5 T_6 C_7 T_8 C_9 T_10 C_11 T_12 C_13 T_14 C_15 T_16 C_17 T_18 T_19 T_20
## C_1    0   1   1   0   0   0   1   0  -1    0   -1   -1    0    0    0    0    1    0   -1    0
## T_2   -1   0   0   0   1   1  -1   0  -1    0   -1    0    0   -1    0    0    0    0   -1    1
## C_3   -1   0   0   0   0   0   1  -1   0   -1   -1    0   -1    0   -1    0    1   -1    0    0
## T_4    0   0   0   0   0   0   1  -1   0   -1   -1    0   -1   -1    0    1    0    0   -1    0
## C_5    0  -1   0   0   0   0   1  -1  -1    0   -1    0    0   -1    0   -1    0   -1   -1    0
```

```
## T_6    0  -1   0   0   0   0   1  -1  -1   -1    0    0    0    0   -1   -1    0    0    0   -1
## C_7   -1   1  -1  -1  -1  -1   0  -1  -1   -1    0    0    0    0   -1    0   -1    0    0   -1
## T_8    0   0   1   1   1   1   1   0  -1   -1    0    0    0    0   -1    0   -1   -1    0   -1
## C_9    1   1   0   0   1   1   1   1   0   -1    1    0    1    0   -1    0    0    0   -1   -1
## T_10   0   0   1   1   0   1   1   1   1    0    1    0    0    0   -1    1    0   -1   -1   -1
## C_11   1   1   1   1   1   0   0   0  -1   -1    0    1   -1    0    0    0    0    0    0    0
## T_12   1   0   0   0   0   0   0   0   0    0   -1    0    1    0    0    0    0    0    0    0
## C_13   0   0   1   1   0   0   0   0  -1    0    1   -1    0    1   -1    0    0    0    0    0
## T_14   0   1   0   1   1   0   0   0   0    0    0    0   -1    0    1    0    0    0    0    0
## C_15   0   0   1   0   0   1   1   1   1    1    0    0    1   -1    0    1   -1    0    0    0
## T_16   0   0   0  -1   1   1   0   0   0   -1    0    0    0    0   -1    0   -1    0    0    0
## C_17  -1   0  -1   0   0   0   1   1   0    0    0    0    0    0    1    1    0    1    0    0
## T_18   0   0   1   0   1   0   0   1   0    1    0    0    0    0    0    0   -1    0    1    0
## T_19   1   1   0   1   1   0   0   0   1    1    0    0    0    0    0    0    0   -1    0    1
## T_20   0  -1   0   0   0   1   1   1   1    1    0    0    0    0    0    0    0    0   -1    0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              22               17
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##        21.62368        21.62368

## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum         29.30467       -23.06112
## Control Win Sum    -23.06112        28.26257
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##          21.624           21.624
##
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum         29.30467       -23.06112
## Control Win Sum    -23.06112        28.26257
```

Another one

```
##      C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8 C_9 T_10 T_11 T_12 T_13 T_14 T_15 T_16 T_17 T_18 T_19 T_20
## C_1    0   1   1   0   0   0   2   0  -3    0   -4   -5    0    0    0    0    6    0   -7    0
## C_2   -1   0   0   0   8   9  -9   0  -1    0   -1    0    0   -1    0    0    0    0   -1    1
## C_3   -1   0   0   0   0   0   1  -1   0   -1   -1    0   -1    0   -1    0    1   -1    0    0
## C_4    0   0   0   0   0   0   1  -1   0   -1   -1    0   -1   -1    0    1    0    0   -1    0
## C_5    0  -8   0   0   0   0   1  -1  -1    0   -1    0    0   -1    0   -1    0   -1   -1    0
## C_6    0  -9   0   0   0   0   1  -1  -1   -1    0    0    0    0   -1   -1    0    0    0   -1
## C_7   -2   9  -1  -1  -1  -1   0  -1  -1   -1    0    0    0    0   -7    0   -2    0    0   -3
## C_8    0   0   1   1   1   1   1   0  -4   -5    0    0    0    0   -5    0   -6   -6    0   -7
## C_9    3   1   0   0   1   1   1   4   0   -8    9    0    8    0   -7    0    0    0   -6   -5
## T_10   0   0   1   1   0   1   1   5   8    0    4    0    0    0   -3    2    0   -1   -1   -1
## T_11   4   1   1   1   1   0   0   0  -9   -4    0    1   -1    0    0    0    0    0    0    0
## T_12   5   0   0   0   0   0   0   0   0    0   -1    0    1    0    0    0    0    0    0    0
## T_13   0   0   1   1   0   0   0   0  -8    0    1   -1    0    1   -1    0    0    0    0    0
## T_14   0   1   0   1   1   0   0   0   0    0    0    0   -1    0    1    0    0    0    0    0
## T_15   0   0   1   0   0   1   7   5   7    3    0    0    1   -1    0    1   -1    0    0    0
## T_16   0   0   0  -1   1   1   0   0   0   -2    0    0    0    0   -1    0   -1    0    0    0
## T_17  -6   0  -1   0   0   0   2   6   0    0    0    0    0    0    1    1    0    1    0    0
## T_18   0   0   1   0   1   0   0   6   0    1    0    0    0    0    0    0   -1    0    1    0
## T_19   7   1   0   1   1   0   0   0   6    1    0    0    0    0    0    0    0   -1    0    1
## T_20   0  -1   0   0   0   1   3   7   5    1    0    0    0    0    0    0    0    0   -1    0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              60               39
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##        53.66842        53.66842

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum        246.4046       -152.7507
## Control Win Sum    -152.7507        240.6151
```

Full Permutation Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##          53.668          53.668
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum        246.4046       -152.7507
## Control Win Sum    -152.7507        240.6151
```

S2 (Two-sample) Bootstrap

S2.1 Example manuscript

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0  -2   0   0   1
## T_2   2   0   3   0  -5
## C_3   0  -3   0   4   0
## C_4   0   0  -4   0  -1
## C_5  -1   5   0   1   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               4               5
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               4               5

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum      11.0000000      -0.8333333
## Control Win Sum   -0.8333333      37.5000000
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               4               5
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum      11.0000000      -0.8333333
## Control Win Sum   -0.8333333      37.5000000
```

S2.2 Other Examples

Examples – first set

The first set of examples examines separately all the cases in the algorithm.

- For each original example, one or more similar examples are created by varying the values of the skew matrices. Only integer values are used, to aid in reading the matrices. The algorithm makes no use of the fact that the values are integers.

Tiny Example 1: One treatment edge (case 1)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   0   1
## T_2   0   0   0   0   0
```

```
## C_3   0   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               1               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             1.5               0
## Control Win Sum          0.0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               1               0
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             1.5               0
## Control Win Sum          0.0               0
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   0   3
## T_2   0   0   0   0   0
## C_3   0   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -3   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               3               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               3               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            13.5               0
## Control Win Sum          0.0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               3               0
##
```

```
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            13.5               0
## Control Win Sum          0.0               0
```

Tiny Example 2: One control edge (case 2)

```
##     C_1 C_2 T_3 T_4 T_5
## C_1   0   0   0   0   1
## C_2   0   0   0   0   0
## T_3   0   0   0   0   0
## T_4   0   0   0   0   0
## T_5  -1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0               1

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0             0.0
## Control Win Sum            0             1.5
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0               1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0             0.0
## Control Win Sum            0             1.5
```

The same example, but with a more general skew matrix

```
##     C_1 C_2 T_3 T_4 T_5
## C_1   0   0   0   0   3
## C_2   0   0   0   0   0
## T_3   0   0   0   0   0
## T_4   0   0   0   0   0
## T_5  -3   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               3
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0               3
```

```
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum                0             0.0
## Control Win Sum             0            13.5
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0               3
##
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum                0             0.0
## Control Win Sum             0            13.5
```

Tiny Example 3: Two treatment edges meet at a treatment vertex (case 3)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   1   0   1
## T_2   0   0   0   0   0
## C_3  -1   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               2               0

## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum                3               0
## Control Win Sum             0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               2               0
##
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum                3               0
## Control Win Sum             0               0
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   4   0   2
## T_2   0   0   0   0   0
## C_3  -4   0   0   0   0
## C_4   0   0   0   0   0
## C_5  -2   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               6               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               6               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum              30               0
## Control Win Sum            0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               6               0
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum              30               0
## Control Win Sum            0               0
```

Tinny Example 3: Three treatment edges meet at a treatment vertex (case 3)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   1   1   1
## T_2   0   0   0   0   0
## C_3  -1   0   0   0   0
## C_4  -1   0   0   0   0
## C_5  -1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               3               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               3               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             4.5               0
## Control Win Sum          0.0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               3               0
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             4.5               0
## Control Win Sum          0.0               0
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   4   3   2
## T_2   0   0   0   0   0
## C_3  -4   0   0   0   0
## C_4  -3   0   0   0   0
## C_5  -2   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               9               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               9               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            43.5               0
## Control Win Sum          0.0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               9               0
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            43.5               0
## Control Win Sum          0.0               0
```

Tiny Example 4: Two treatment edges meet at a control vertex (case 4)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   0   0
## T_2   0   0   0   1   0
## T_3   0   0   0   1   0
## C_4   0  -1  -1   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               2               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               3               0
## Control Win Sum            0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               2               0
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               3               0
## Control Win Sum            0               0
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   0   0
## T_2   0   0   0   5   0
## T_3   0   0   0   7   0
## C_4   0  -5  -7   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              12               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##              12               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             111               0
## Control Win Sum            0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##              12               0
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             111               0
## Control Win Sum            0               0
```

Tiny Example 5: Two control edges meet at a treatment vertex (case 5)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -1  -1
## T_2   0   0   0   0   0
## C_3   0   0   0   0   0
## C_4   1   0   0   0   0
## C_5   1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0               2

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0               3
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0               2
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0               3
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0 -11 -13
## T_2   0   0   0   0   0
## C_3   0   0   0   0   0
## C_4  11   0   0   0   0
## C_5  13   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0              24
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0              24

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0             435
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0              24
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0             435
```

Tiny Example 6: Two control edges meet at a control vertex (case 6)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0  -1   0
```

```
## T_2   0   0   0  -1   0
## T_3   0   0   0   0   0
## C_4   1   1   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0               2

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0               3
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0               2
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0               3
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0 -11   0
## T_2   0   0   0 -17   0
## T_3   0   0   0   0   0
## C_4  11  17   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0               2

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0               3
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0               2
```

```
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0               0
## Control Win Sum            0               3
```

Tiny Example 7: Treatment and control edges meet at a treatment vertex (cases 7 and 7a)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   1  -1
## T_2   0   0   0   0   0
## T_3   0   0   0   0   0
## C_4  -1   0   0   0   0
## C_5   1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               1               1

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.5000000      -0.1666667
## Control Win Sum   -0.1666667       1.5000000
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               1               1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.5000000      -0.1666667
## Control Win Sum   -0.1666667       1.5000000
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   3  -7
## T_2   0   0   0   0   0
## T_3   0   0   0   0   0
## C_4  -3   0   0   0   0
## C_5   7   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               3               7
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               3               7
```

```
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            13.5            -3.5
## Control Win Sum         -3.5            73.5
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               3               7
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            13.5            -3.5
## Control Win Sum         -3.5            73.5
```

Tiny Example 8: Treatment and control edges meet at a treatment vertex (cases 8 and 8a)

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0   1   0
## T_2   0   0   0  -1   0
## T_3   0   0   0   0   0
## C_4  -1   1   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               1               1

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             1.5             0.0
## Control Win Sum          0.0             1.5
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               1               1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             1.5             0.0
## Control Win Sum          0.0             1.5
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 C_4 C_5
## T_1   0   0   0  12   0
## T_2   0   0   0 -11   0
## T_3   0   0   0   0   0
## C_4 -12  11   0   0   0
## C_5   0   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              12               11
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##              12               11

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             216             0.0
## Control Win Sum            0           181.5
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##              12               11
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             216             0.0
## Control Win Sum            0           181.5
```

Tiny Example 9: Disjoint treatment edges (case 9)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   1   0
## T_2   0   0   0   0   1
## C_3   0   0   0   0   0
## C_4  -1   0   0   0   0
## C_5   0  -1   0   0   0
```

The same example, but with a more general skew matrix

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2                0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               2                0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum        1.666667                0
## Control Win Sum     0.000000                0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               2                0
##
## $Variance
##                 Test Win Sum Control Win Sum
```

```
## Test Win Sum        1.666667               0
## Control Win Sum     0.000000               0

##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0   3   0
## T_2   0   0   0   0   5
## C_3   0   0   0   0   0
## C_4  -3   0   0   0   0
## C_5   0  -5   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               8               0
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               8               0

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum              31               0
## Control Win Sum            0               0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               8               0
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum              31               0
## Control Win Sum            0               0
```

Tiny Example 10: Disjoint control edges (case 10)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -1   0
## T_2   0   0   0   0  -1
## C_3   0   0   0   0   0
## C_4   1   0   0   0   0
## C_5   0   1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               2
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0               2

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum               0        0.000000
## Control Win Sum            0        1.666667
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0               2
##
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum                0        0.000000
## Control Win Sum             0        1.666667
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -2   0
## T_2   0   0   0   0  -1
## C_3   0   0   0   0   0
## C_4   2   0   0   0   0
## C_5   0   1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               0               3
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               0               3

## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum                0        0.000000
## Control Win Sum             0        4.833333
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               0               3
##
## $Variance
##                  Test Win Sum Control Win Sum
## Test Win Sum                0        0.000000
## Control Win Sum             0        4.833333
```

Tiny Example 11: Disjoint treatment and control edges (case 11)

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -1   0
## T_2   0   0   0   0   1
## C_3   0   0   0   0   0
## C_4   1   0   0   0   0
## C_5   0  -1   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               1               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               1               1

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.5000000      -0.6666667
## Control Win Sum   -0.6666667       1.5000000
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               1               1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.5000000      -0.6666667
## Control Win Sum   -0.6666667       1.5000000
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0   0   0  -6   0
## T_2   0   0   0   0   5
## C_3   0   0   0   0   0
## C_4   6   0   0   0   0
## C_5   0  -5   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               5               6
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               5               6

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            37.5             -20
## Control Win Sum        -20.0              54
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               5               6
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum            37.5             -20
## Control Win Sum        -20.0              54
```

Examples – Second set

The second set of examples examines some larger win matrices. As a practical matter, matrices larger than 8×8 would take too much time with the full bootstrap algorithm.

The manuscript example, but with a $\{1,0,-1\}$ skew matrix

```
##     T_1 T_2 C_3 C_4 C_5
## T_1   0  -1   0   0   1
## T_2   1   0   1   0  -1
## C_3   0  -1   0   1   0
## C_4   0   0  -1   0  -1
## C_5  -1   1   0   1   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               2               1

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.6666667      -0.1666667
## Control Win Sum   -0.1666667       1.5000000
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               2               1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum       1.6666667      -0.1666667
## Control Win Sum   -0.1666667       1.5000000
```

Another example

```
##     T_1 T_2 C_3 C_4 C_5 C_6
## T_1   0   1   0   0   1   0
## T_2  -1   0   1   0   0  -1
## C_3   0  -1   0   1   1   0
## C_4   0   0  -1   0   1   0
## C_5  -1   0  -1  -1   0   0
## C_6   0   1   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               2               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               2               1
```

```
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             2.0          -0.500
## Control Win Sum         -0.5           1.625
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               2               1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum             2.0          -0.500
## Control Win Sum         -0.5           1.625
```

The same example, but with a more general skew matrix

```
##     C_1 T_2 C_3 T_4 C_5 T_6
## C_1   0   1   0   0   2   0
## T_2  -1   0   3   0   0  -4
## C_3   0  -3   0   5   6   0
## T_4   0   0  -5   0   7   0
## C_5  -2   0  -6  -7   0   0
## T_6   0   4   0   0   0   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               5               4
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               5               4

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum          13.625            -3.5
## Control Win Sum       -3.500            26.0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               5               4
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum          13.625            -3.5
## Control Win Sum       -3.500            26.0
```

A third example

```
##     T_1 T_2 T_3 T_4 C_5 C_6 C_7 C_8
## T_1   0   1   0   0   1   0   0   0
## T_2  -1   0   1  -1   0   1   0   0
## T_3   0  -1   0   1   1   1   1   0
## T_4   0   1  -1   0   0   0  -1   1
## C_5  -1   0  -1   0   0   0   1   1
```

```
## C_6   0  -1  -1   0   0   0   0   1
## C_7   0   0  -1   1  -1   0   0  -1
## C_8   0   0   0  -1  -1  -1   1   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##               6               1
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##               6               1

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           6.750         -1.1250
## Control Win Sum       -1.125          2.0625
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##               6               1
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum           6.750         -1.1250
## Control Win Sum       -1.125          2.0625
```

The same example, but with a more general skew matrix

```
##     T_1 T_2 T_3 T_4 C_5 C_6 C_7 C_8
## T_1   0   1   0   0   2   0   0   0
## T_2  -1   0   3  -5   0   4   0   0
## T_3   0  -3   0   6   7   8   9   0
## T_4   0   5  -6   0   0   0  -8   7
## C_5  -2   0  -7   0   0   0   6   5
## C_6   0  -4  -8   0   0   0   0   5
## C_7   0   0  -9   8  -6   0   0  -4
## C_8   0   0   0  -7  -5  -5   4   0
```

Observed Wins

```
##    Test Win Sum Control Win Sum
##              37               8
```

Practical Algorithm

```
## $Expected
##    Test Win Sum Control Win Sum
##              37               8

## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum        414.0625           -33.5
## Control Win Sum     -33.5000           132.0
```

Full Bootstrap Algorithm

```
## $Means
##    Test Win Sum Control Win Sum
##              37               8
##
## $Variance
##                 Test Win Sum Control Win Sum
## Test Win Sum        414.0625           -33.5
## Control Win Sum     -33.5000           132.0
```