Journal of the Nigerian Association of Mathematical Physics
Volume **16 (May, 2010), pp 579 - 584**

A method for generating permutation distribution of ranks in a k-sample experiment.

**F. Ewere and S.M. Ogbonmwan*
Department of Mathematics, University of Benin, Benin City, Nigeria.
*Corresponding author: **e-mail:** frewere2@yahoo.com Tel.+2348058504893

Abstract

A method for generating exhaustive permutation distribution of the ranks in a k-sample experiment is presented. This provides a methodology for constructing exact test of significance of a rank statistic.The proposed method is linked to the partition of integers and in a combinatorial sense the distribution of the ranks is obtained via its generating function.The formulas are defined recursively to speed up computations using the computer algebra system Mathematica.

Key words: Partitions, generating functions, combinatorics, permutation test, exact tests, computer algebra, k-sample, rank statistics.

1.0 Introduction

Constructing the exact distribution of a rank statistic is a very vital aspect of inferential statistics as it ensures that the probability of making a type I error is exactly α. But, a major challenge has been the availability of computational formulas for generating the associated permutation sample spaces required to conduct the exact tests especially when the sample sizes are not small.

[27] showed that the permutation approach is the only possible technique of constructing exact tests of significance for a general class of problems. The null distribution of statistics obtained through the unconditional exact permutation approach in which row and column totals are allowed to vary with each permutation turns out to be the most reliable, see [1], [10] and [19]. The unconditional exact permutation approach is very much unlike the conditional exact permutation approach of fixing the row and column totals, see [4], [13] and [19]. Other approaches to the unconditional exact permutation exist in the literature. For a detailed discussion on Monte Carlo methods, see [8], [9], [12] and [20]. The Bayesian and the Likelihood approaches can be found in [5] and [29]. All these approaches only give approximate results. The purpose of this paper therefore is to provide a method for generating the permutation sample spaces in a *k*-sample experiment. Exact procedures are the best and should always be applied whenever practically possible, see [10] and [16]. Permutation tests provide exact results especially when complete enumeration is possible, see [22].

This paper is organized as follows. In section 2, a brief discussion on generating functions is provided, and in section 3, the partitions of integers is presented. Section 4 gives the concept of combinatorics and in section 5, an efficient method for handling some combinatorial problems is introduced. In section 6, the Mathematica procedures of the proposed method is shown and section 7 gives the conclusion of the paper.

Throughout this paper, command statements appear in a separate font and are written in terms of Mathematica 6.0 code. All calculations were performed on an intel Pentium M computer with a processor speed of $1.73GH_Z$.

2.0 Generating Functions

A generating function is a polynomial in one or more variables (in expanded form) whose exponents are real numbers and coefficients are the numbers being sought. Generating functions are widely used in probability theory, see [6], [11], [15] and [26]. Generating functions provide a simple and elegant way to describe probability or frequency distributions of discrete statistics and in particular, permutation distributions. They are also a computational tool.

Many efficient algorithms, including those described as fast Fourier transform methods, network methods and multiple shift methods are different implementations of the recursions needed to evaluate generating functions efficiently, see [3]. Usually polynomials have integer exponents only. Since Mathematica works well with this kind of "generalized polynomials", they are used instead.

Generating functions are often used as analytical results in literature. In this paper, it will be shown how these generating functions are easily implemented in Mathematica for computing the permutation distribution of ranks in a k-sample experiment.

3.0 Partitions of Integers

Given an integer n, it is possible to represent it as the sum of one or more positive integers a_i, that is $n = x_1 + x_2 + ... + x_m$. This representation is called a partition if the order of the x_i is of no consequence. Thus, two partitions of an integer n are distinct if they differ with respect to the x_i they contain. For example, there are seven distinct partitions of the integer 5:
5, 4 + 1, 3 + 2, 3 + 1 + 1, 2 + 2 + 1, 2 + 1 + 1 + 1, 1 + 1 + 1 + 1 + 1.

The partitions of an integer have been the subject of extensive study for over 300 years, since Leibnitz asked Bernoulli if he had investigated $P(n)$, the number of partitions of an integer n. Details of the history and the state of the art as of 1920 can be found in [7]. Additional details and later results can be found in most combinatorics texts; in particular, see [11], [17] and [25]. The interest in this work is partly motivated by the important role played by partitions in many problems of combinatorics and algebra. For computational purposes one is often interested in generating all the partitions of an integer, or sometimes just those satisfying various restrictive conditions. Several such algorithms, dealing with both the unrestricted and restricted cases have appeared in the literature. For the unrestricted cases, see [2], [18], [21] and [23]. In the case of the restricted partitions, see [2], [24] and [30].

Generating functions were first applied to partitions by Euler. This technique can reduce the difficulty of otherwise complex problems. We use generating functions because they can be manipulated much more easily than combinatorial quantities. Euler invented a generating function which gives rise to a recurrence equation in $P(n)$ given as

$$P(n) = \sum_{k=1}^{n} (-1)^{k+1} \left[P\left(n - \tfrac{1}{2}k(3k-1)\right) + P\left(n - \tfrac{1}{2}k(3k+1)\right) \right]$$

(3.1)

Other recurrence equations include

$$P(2n+1) = P(n) + \sum_{k=1}^{\infty} \left[P\left(n - 4k^2 - 3k\right) + P\left(n - 4k^2 + 3k\right) \right] - \sum_{k=1}^{\infty} (-1)^k \left[P\left(2n+1-3k^2+k\right) + P\left(2n+1-3k^2+1\right) \right.$$

(3.2)

$$P(n) = \tfrac{1}{n} \sum \sigma_1 (n-k) P(k)$$

(3.3)

Where $\sigma_1(n)$ is the divisor function. For these recurrence equations, see [28].
The partition numbers $P(n)$ are given by the generating function

$$\frac{1}{(q)_\infty} = \sum_{n=0}^{\infty} P(n)q^n = 1 + q + 2q^2 + 3q^3 + 5q^4 + \Lambda \qquad (3.4)$$

see Hirschhorn (1999). $P(n,k)$ denotes the number of ways of writing n as a sum of exactly k terms. Hence, $P(5,3) = 2$, since the partition of 5 of length 3 are $\{3,1,1\}$ and $\{2,2,1\}$. $P(n,k)$ can be computed from the recurrence relation

$$P(n,k) = P(n-1,k-1) + P(n-k,k) \qquad (3.5)$$

With $P(n,k) = 0$ for $k > n, P(n,n) = 1$ and $P(n,0) = 0$, see [28].

4.0 Combinatorics

A large class of problems in Combinatorial Mathematics is concerned with computing the number of ways in which some well-defined operation can be performed. The notions of combinations and permutations are the simplest and yet

most fundamental concepts in the study of the theory of enumeration. Other enumerative techniques include generating functions, recurrence relations, the principle of inclusion and exclusion, Polya's theory of counting. For detailed discussion of these concepts, see [17].

The crucial point in nonparametric test theory is the fact that all possible arrangement of the ranks of the observed values are equally likely. The sufficient condition for a permutation test to be exact and unbiased against shifts in direction of higher values is the exchangeability of the observations in the combined sample, see [10] and [22] noted that when exchangeability may be assumed in the null hypothesis H_0, reference null distributions of permutation tests always exist, because, at least in principle, they are obtained by considering all permutations of available data.To calculate the distribution density $\Pr(\{X = x\})$ of a statistic X based on ranks, it is therefore only necessary to obtain the number of cases satisfying the condition $X = x$. The combinatorial problems below give an essential idea in achieving this.
Suppose there are n observations which are ranked 1, 2, 3,…, n. In how many different ways can one divide these n observations among k samples such that the ith sample T_i contains n_i observations and the sum of the ranks of these n_i observations in sample T_i is r_i with $n = n_1 + n_2 + ... + n_k$ and $r = r_1 + r_2 + ... + r_k = \frac{1}{2}n(n+1)$? Let the number be:

$$P[nlist, rlist] := P[\{n_1, n_2, ..., n_k\}, \{r_1, r_2 ..., r_k\}] \qquad (4.1)$$

We can calculate this number $P[nlist, rlist]$ by counting the relevant partitions. There are $\frac{(n_1 + n_2 + ... + n_k)!}{n_1!n_2!...,n_k!}$ possible permutations of the n variates of the k samples of sizes $n_{i,} i = 1,2,...,k$ which are equally likely with probability $\left(\frac{n!}{n_1!n_2!...,n_k!}\right)^{-1}$. Consider an experiment of two samples with three observations in the first sample and two observations in the second sample. The total number of distinct arrangements is $\frac{5!}{3!2!} = 10$. Clearly from table 1, $P[\{3,2\},\{8,7\}] = 2$ and $P[\{3,2\},\{11,4\}] = 1$. If n and k are not as small as in this example, this method of enumeration fails because of the large number of permutation sample spaces.

Suppose again that there are n observations numbered in such a way that d_1 of these observations are ranked m_1, d_2 have ranks $m_2,...,$ and d_l have ranks m_l so that $n = d_1 + d_2 + ... + d_l$. We can find the different ways that is possible to divide these n observations among k samples such that the ith sample contains n_i observations and the sum of the ranks of these n_i observations is r_i with $n = n_1 + n_2 + ... + n_k = d_1 + d_2 + ... + d_l$ and $r = r_1 + r_2 + ... + r_k = d_1 m_1 + d_2 m_2 + ... + d_k m_k$

Again, let this number be represented by:

$$Q[dlist, mlist, nlist, rlist] := Q[\{d_1, d_2, ..., d_l\}, \{m_1, m_2, ..., m_k\}, \{n_1, n_2, ..., n_k\}, \{r_1, r_2, ..., r_k\}]$$

(4.2)

Clearly, (4.2) is a generalization of (4.1), since for $l = n,$ we have

$$Q[\{1,1,...,1\}, \{1,2,...,n\}, \{n_1, n_2, ..., n_k\}, \{r_1, r_2, ..., r_k\}] = P[\{n_1, n_2, ..., n_k\}, \{r_1, r_2, ..., r_k\}]$$

(4.3)

If we let $d_1 = 2$ of the $n = 5$ observations in table 1 be ranked $m_1 = 1$. Suppose these are the observations with the ranks 1, 2. Also let the remaining $d_2 = 3$ observations be ranked $m_2 = 2,$ then, all possible arrangements of the ranks follows immediately from table 1 and this is given in table 2. However, when ties occur in ranking, it is customary to assign average ranks. It is evident from table 2 that $Q[\{2,3\}, \{1,2\}, \{3,2\}, \{5,3\}] = 6$ and

$Q[\{2,3\}, \{1,2\}, \{3,2\}, \{4,4\}] = 3$. Again, if n and k are not as small as in this example, the method fails.

5. An efficient method

To overcome the problems associated with the method of enumeration in section 4, the t concept of generating functions is introduced. The aim here is to find the generating functions for the numbers $P[nlist, rlist]$ and

$Q[dlist, mlist, nlist, rlist]$ respectively. Let $x[i]$ be a variable governing the number of observations in the ith sample and $y[i]$ be a variable governing the sum of the ranks of the observations in the ith sample. With this remark, the generating functions for the numbers $P[nlist, rlist]$ and $Q[\{dlist, mlist, nlist, rlist\}]$ respectively are:

$$p[n,k] = \prod_{j=1}^{n} \left(\sum_{i=1}^{k} x[i] y[i]^j \right)$$

(5.1)

$$q[dlist, mlist, k] = \prod_{j=1}^{l} \left(\sum_{i=1}^{k} x[i] y[i]^{m_j} \right)^{d_j}$$

(5.2)

Obviously, the numbers $P[nlist, rlist]$ and $Q[dlist, mlist, nlist, rlist]$ are the coefficients of $\prod_{i=1}^{k} x[i]^{n_i} y[i]^{r_i}$ of the polynomial $p[n,k]$ and $q[dlist, mlist, k]$. Hence, we get the numbers $P[nlist, rlist]$ and $Q[dlist, mlist, nlist, rlist]$ by selecting the coefficients of $\prod_{i=1}^{k} x[i]^{n_i} y[i]^{r_i}$. The generating functions in (4.1), (4.2), (5.1) and (5.2) are implemented in Mathematica 6.0 and the procedures are given in section 6. With this algorithm, it is possible to solve the combinatorial problems posed in section 4 and other problems which are very difficult to handle manually.

6.0 Mathematica procedures

This section contains the Mathematica procedures for calculating the numbers $P[nlist, rlist]$ and $Q[dlist, mlist, nlist, rlist]$.

6.1 Mathematica Commands

```
xlist[k_]:=Table[x[i],{i,1,k}];
ylist[k_]:=Table[y[i],{i,1,k}];
p[1,k_]:=xlist[k].ylist[k];
p[n_,k_]:=p[n,k]=p[n-1,k]*xlist[k].ylist[k]^n
q[{d_},{m_},k_]:=(xlist[k].ylist[k]^m)^d;
q[dlist_,mlist_,k_]:=q[dlist,mlist,k]= q[{First[dlist]},{First[mlist]},k]
q[Rest[dlist],Rest[mlist],k]

P[nlist_,rlist_]:=
Module[{n,r,k},
n=Plus@@nlist;
r=Plus@@rlist;
k=Length[nlist];
If[r□n (n+1)/2, Fold[Coefficient,Expand[p[n,k]],
Union[xlist[k]^nlist,ylist[k]^rlist]],
Print["r □ n (n+1)/2"]]]

Q[dlist_,mlist_,nlist_,rlist_]:= Module[{d,n,r,k}, d=Plus@@dlist;
n=Plus@@nlist; r=Plus@@rlist;
k=Length[nlist];
If[n□d && r□dlist.mlist, Fold[Coefficient,Expand[q[dlist,mlist,k]],
Union[xlist[k]^nlist,ylist[k]^rlist]],
Print["n □ d1+...+dl or r □ d1 m1+...+dl ml"]]]
```

Numerical examples

Expand[p[3,3]]=

$$x[1]^{3} y[1]^{6} + x[1]^{2} x[2]\, y[1]^{5} y[2] + x[1]^{2} x[2]\, y[1]^{4} y[2]^{2} + x[1]^{2} x[2]\, y[1]^{3} y[2]^{3} + x[1]\, x[2]^{2} y[1]^{3} y[2]^{3} + x[1]\, x[2]^{2} y[1]^{2} y[2]^{4} + x[1]\, x[2]^{2} y[1]\, y[2]^{5} + x[2]^{3} y[2]^{6} +$$
$$x[1]^{2} x[3]\, y[1]^{5} y[3] + x[1]\, x[2]\, x[3]\, y[1]^{3} y[2]^{2} y[3] + x[1]\, x[2]\, x[3]\, y[1]^{2} y[2]^{3} y[3] + x[2]^{2} x[3]\, y[2]^{5} y[3] + x[1]^{2} x[3]\, y[1]^{4} y[3]^{2} + x[1]\, x[2]\, x[3]\, y[1]^{3} y[2]\, y[3]^{2} +$$
$$x[1]\, x[2]\, x[3]\, y[1]\, y[2]^{3} y[3]^{2} + x[2]^{2} x[3]\, y[2]^{4} y[3]^{2} + x[1]^{2} x[3]\, y[1]^{3} y[3]^{3} + x[1]\, x[3]^{2} y[1]^{3} y[3]^{3} + x[1]\, x[2]\, x[3]\, y[1]^{2} y[2]\, y[3]^{3} + x[1]\, x[2]\, x[3]\, y[1]\, y[2]^{2} y[3]^{3} +$$
$$x[2]^{2} x[3]\, y[2]^{3} y[3]^{3} + x[2]\, x[3]^{2} y[2]^{3} y[3]^{3} + x[1]\, x[3]^{2} y[1]^{2} y[3]^{4} + x[2]\, x[3]^{2} y[2]^{2} y[3]^{4} + x[1]\, x[3]^{2} y[1]\, y[3]^{5} + x[2]\, x[3]^{2} y[2]\, y[3]^{5} + x[3]^{3} y[3]^{6}$$

Expand[q[{3,2},{1,2},3]]=

$$x[1]^{5} y[1]^{7} + 3x[1]^{4} x[2]\, y[1]^{6} y[2] + 2x[1]^{4} x[2]\, y[1]^{5} y[2]^{2} + 3x[1]^{3} x[2]^{2} y[1]^{5} y[2]^{2} + 6x[1]^{3} x[2]^{2} y[1]^{4} y[2]^{3} + x[1]^{2} x[2]^{3} y[1]^{4} y[2]^{3} + x[1]^{3} x[2]^{2} y[1]^{3} y[2]^{4} +$$
$$6x[1]^{2} x[2]^{3} y[1]^{3} y[2]^{4} + 3x[1]^{2} x[2]^{3} y[1]^{2} y[2]^{5} + 2x[1]\, x[2]^{4} y[1]^{2} y[2]^{5} + 3x[1]\, x[2]^{4} y[1]\, y[2]^{6} + x[2]^{5} y[2]^{7} + 3x[1]^{4} x[3]\, y[1]^{6} y[3] + 6x[1]^{3} x[2]\, x[3]\, y[1]^{5} y[2]\, y[3] +$$
$$6x[1]^{3} x[2]\, x[3]\, y[1]^{4} y[2]^{2} y[3] + 3x[1]^{2} x[2]^{2} x[3]\, y[1]^{4} y[2]^{2} y[3] + 12x[1]^{2} x[2]^{2} x[3]\, y[1]^{3} y[2]^{3} y[3] + 3x[1]^{2} x[2]^{2} x[3]\, y[1]^{2} y[2]^{4} y[3] + 6x[1]\, x[2]^{3} x[3]\, y[1]^{2} y[2]^{4} y[3] +$$
$$6x[1]\, x[2]^{3} x[3]\, y[1]\, y[2]^{5} y[3] + 3x[2]^{4} x[3]\, y[2]^{6} y[3] + 2x[1]^{4} x[3]\, y[1]^{5} y[3]^{2} + 3x[1]^{3} x[3]^{2} y[1]^{5} y[3]^{2} + 6x[1]^{3} x[2]\, x[3]\, y[1]^{4} y[2]\, y[3]^{2} + 3x[1]^{2} x[2]\, x[3]^{2} y[1]^{4} y[2]\, y[3]^{2} +$$
$$2x[1]^{3} x[2]\, x[3]\, y[1]^{3} y[2]^{2} y[3]^{2} + 6x[1]^{2} x[2]^{2} x[3]\, y[1]^{3} y[2]^{2} y[3]^{2} + 6x[1]^{2} x[2]\, x[3]^{2} y[1]^{3} y[2]^{2} y[3]^{2} + 6x[1]^{2} x[2]^{2} x[3]\, y[1]^{2} y[2]^{3} y[3]^{2} + 2x[1]\, x[2]^{3} x[3]\, y[1]^{2} y[2]^{3} y[3]^{2} +$$
$$6x[1]\, x[2]^{2} x[3]^{2} y[1]^{2} y[2]^{3} y[3]^{2} + 6x[1]\, x[2]^{3} x[3]\, y[1]\, y[2]^{4} y[3]^{2} + 3x[1]\, x[2]^{2} x[3]^{2} y[1]\, y[2]^{4} y[3]^{2} + 2x[2]^{4} x[3]\, y[2]^{5} y[3]^{2} + 3x[2]^{3} x[3]^{2} y[2]^{5} y[3]^{2} + 6x[1]^{3} x[3]^{2} y[1]^{4} y[3]^{3} +$$
$$x[1]^{2} x[3]^{3} y[1]^{4} y[3]^{3} + 12x[1]^{2} x[2]\, x[3]^{2} y[1]^{3} y[2]\, y[3]^{3} + 6x[1]^{2} x[2]\, x[3]^{2} y[1]^{2} y[2]^{2} y[3]^{3} + 6x[1]\, x[2]^{2} x[3]^{2} y[1]^{2} y[2]^{2} y[3]^{3} + 2x[1]\, x[2]\, x[3]^{3} y[1]^{2} y[2]^{2} y[3]^{3} +$$
$$12x[1]\, x[2]^{2} x[3]^{2} y[1]\, y[2]^{3} y[3]^{3} + 6x[2]^{3} x[3]^{2} y[2]^{4} y[3]^{3} + x[2]^{2} x[3]^{3} y[2]^{4} y[3]^{3} + x[1]^{3} x[3]^{2} y[1]^{3} y[3]^{4} + 6x[1]^{2} x[3]^{3} y[1]^{3} y[3]^{4} + 3x[1]^{2} x[2]\, x[3]^{2} y[1]^{2} y[2]\, y[3]^{4} +$$
$$6x[1]\, x[2]\, x[3]^{3} y[1]^{2} y[2]\, y[3]^{4} + 3x[1]\, x[2]^{2} x[3]^{2} y[1]\, y[2]^{2} y[3]^{4} + 6x[1]\, x[2]\, x[3]^{3} y[1]\, y[2]^{2} y[3]^{4} + x[2]^{3} x[3]^{2} y[2]^{3} y[3]^{4} + 6x[2]^{2} x[3]^{3} y[2]^{3} y[3]^{4} + 3x[1]^{2} x[3]^{3} y[1]^{2} y[3]^{5} +$$
$$2x[1]\, x[3]^{4} y[1]^{2} y[3]^{5} + 6x[1]\, x[2]\, x[3]^{3} y[1]\, y[2]\, y[3]^{5} + 3x[2]^{2} x[3]^{3} y[2]^{2} y[3]^{5} + 2x[2]\, x[3]^{4} y[2]^{2} y[3]^{5} + 3x[1]\, x[3]^{4} y[1]\, y[3]^{6} + 3x[2]\, x[3]^{4} y[2]\, y[3]^{6} + x[3]^{5} y[3]^{7}$$

P[{8,8},{70,66}]= 515

Q[{3,4,5,3},{1,2,3,4},{6,4,5},{16,10,12}]= 26355
Other results can be obtained in a similar fashion.

7. Conclusion

This paper provides a method for generating the permutation distribution of the ranks of a k-sample experiment. The algorithm discussed in this article enables us to handle the combinatorial problems posed in section 4 and other problems that are very difficult to calculate manually. Therefore, it is possible to manage with larger sample sizes within reasonable time. For example, it takes only 0.015 seconds to compute P[{8, 8},{70, 66}]
and 0.219 seconds to calculate Q[{3, 4, 5, 3},{1, 2, 3, 4}, {6, 4, 5},{16, 10, 12}]. These calculations require 12,870 and 630,630 distinct arrangements of the ranks respectively.

Finally, the proposed method show promises of being applied to the computation of exact distribution of rank statistics. This is the next challenge.

Table 1. Permutation of the ranks of a 2 sample experiment

T_1	r_1	T_2	r_2
1,2,3	6	4,5	9
1,2,4	7	3,5	8
1,3,4	8	2,5	7
2,3,4	9	1,5	6
1,2,5	8	3,4	7
1,3,5	9	2,4	6
2,3,5	10	1,4	5
1,4,5	10	2,3	5
2,4,5	11	1,3	4
3,4,5	12	1,2	3

Table 2. Permutation of the ranks of a 2 sample experiment

T_1	r_1	T_2	r_2
1, 1, 2	4	2, 2	4
1, 1, 2	4	2, 2	4
1, 2, 2	5	1, 2	3
1, 2, 2	5	1, 2	3
1, 1, 2	4	2, 2	4
1, 2, 2	5	1, 2	3
1, 2, 2	5	1, 2	3
1, 2, 2	5	1, 2	3
1, 2, 2	5	1, 2	3
2, 2, 2	6	1, 1	2

References

[1] Agresti, A (1992). A survey of exact inference for contingency tables, Statistical Science, 7, 131-177.

[2] Andrews, G. E. (1976). The theory of Partitions, Addison-Wesley, Reading, Ma.

[3] Baglivo, J., Pagano, M and Spino C. (1993). Symbolic computation of permutation distributions. Proceedings of the Statistical Computing section of the American Statistical Association, 218-223.

[4] Bagui, S., and Bagui, S. (2004). An Algorithm and Code for computing exact Critical values for the Kruskal-Wallis nonparametric one way ANOVA. Journal of Modern Applied Statistical Methods,3, 498-503.

[5] Bayarri, M. J. and Berger, J. O. (2004). The interplay of Bayesian and frequentist analysis. Statistical Science, 19, 58-80.
[6] Davenport, W. B. (1970). Probability and Random Processes. New York: Mc Graw Hill.
[7] Dickson, L. E. (1971). History of the theory of numbers, Vol. II, Diophantine Analysis, Chelsea Publishing Co., New York.
[8] Efron, B. (1979). Bootstrap methods: another look at the jackknife. The Annals of Statistics, 7, 1-26.
[9] Efron, B. and Tibshirani, R.J. (1993). An introduction to the bootstrap. NY: Chapman and Hall.
[10] Good, P. (2000). Permutation Tests: A practical Guide to Resampling methods for testing hypothesis, 2nd ed. Springer-Verlag, New York.
[11] Hall, M, Jr. (1967). Combinatorial Theory, Blaisdell, Waltham, Mass.
[12] Hall, P. and Tajvidi, N. (2002). Permutation tests for equality of distributions in high dimensional settings. Biometrika, 89, 359-374.
[13] Headrick, T. C. (2003). An algorithm for generating exact critical values for the Kruskal-Wallis One-Way ANOVA. Journal of Modern Applied Statistical Methods, 2, 268-271
[14] Hirschhorn, M. D. (1999).Another short proof of Ramanujan's Mod 5 Partition Congruences and more. American Mathematical Monthly, 106, 580-583.
[15] Johnson, N. L., and Kotz, S. (1977). Urn Models and their applications. New York: John Wiley & Sons.
[16] Lehmann, E. L. (1986). Testing statistical hypothesis, 2nd ed. New York: Wiley.
[17] Liu, C. L. (1968). Introduction to Combinatorial Mathematics. McGraw Hill, Inc.
[18] McKay, J. K. S. (1965). Map of partitions into integers, Alg. 264, Comm. ACM, 8, 493.
[19] Odiase, J. I. and Ogbonmwan, S.M. (2005). An algorithm for generating unconditional exact permutation distribution for a two-sample experiment. Journal of Modern Applied Statistical Methods, 4, 319-332.
[20] Opdyke, J. D. (2003). Fast permutation tests that maximize power under conventional Monte Carlo sampling for pairwise and multiple comparisons. Journal of Modern Applied Statistical Methods, 2, 27-49.
[21] Page, E. S and Wilson, L. B. (1979). An Introduction to Computational Combinatorics, Cambridge Univ. Press.
[22] Pesarin, F. (2001). Multivariate Permutation tests. New York: John Wiley and Sons.
[23] Reingold, E. M., Nievergelt, J. and Deo, N. (1977). Combinatorial Algorithms, Prentice Hall, Englewood Cliffs, New Jersey.
[24] Riha, W. and James, K. R. (1976). Efficient algorithms for doubly and multiply restricted partitions, Algorithm 29, Computing, 16, 163-168.
[25] Riordan, J. (1958). An introduction to Combinatorial Analysis, John Wiley.
[26] Rohatgi, V. K. (1976). An introduction to probability Theory and Mathematical Statistics. New York: John Wiley and Sons.
[27] Scheffe, H. (1943). Statistical Inference in the nonparametric case. The Annals of Mathematical Statistics, 14, 305-332.
[28] Skiena, S. (1990). Implementing Discrete Mathematics, Combinatorics and Graph theory with Mathematica. Reading, Ma: Addison-Wesley.
[29] Spiegelhalter, D. J. (2004). Incorporating Bayesian ideas into health care evaluation. Statistical Science, 19, 156-174.
[30] White, J. S. (1970). Restricted partition generator, Alg. 374, Comm. ACM, 13, 120.
[31] Wolfram Research, Inc. (2007). Mathematica 6.0, Champaign Illinois.