Nonparametric Statistical Analysis of FMRI Data

B. Douglas Ward
Biophysics Research Institute
Medical College of Wisconsin
email: ward@mcw.edu

July 28, 1997

Abstract

Parametric statistical analysis programs such as 3dttest and 3dANOVA assume that the underlying populations (of voxel intensities) have a normal (or near normal) distribution. There are two reasons why one might prefer to use a nonparametric statistical analysis: 1) The population in question may differ significantly from the normal distribution. 2) Nonparametric statistical analysis techniques are usually less sensitive to the presence of "outliers", i.e., they are more robust. Therefore, to provide the user with this option, the current distribution of *AFNI* includes four nonparametric statistical analysis programs: 3dMannWhitney, 3dWilcoxon, 3dKruskalWallis, and 3dFriedman. This set of programs is intended to provide the capability to perform nonparametric statistical analysis of FMRI data, roughly corresponding to the present capability to perform parametric statistical analysis.

Section 1 describes Program 3dMannWhitney, for comparison of two treatments (two samples). This program performs the Wilcoxon-Mann-Whitney rank-sum test on two groups of *AFNI* 3d datasets, voxel-by-voxel, to determine if the two samples are from the same population. Program output includes an estimate of the treatment effect, as well as the normalized Wilcoxon rank-sum statistic, for each voxel.

Section 2 describes Program 3dWilcoxon, for the paired comparison of two treatments. This program performs the Wilcoxon signed-rank test for pairs of *AFNI* 3d datasets. Output includes an estimate for the treatment effect, and the normalized Wilcoxon signed-rank statistic, for each voxel.

Section 3 describes Program 3dKruskalWallis, for comparing multiple treatments. This program performs the Kruskal-Wallis test to determine if any of k treatments (k groups of AFNI 3d datasets) are statistically different, on a voxel-by-voxel basis. Output includes the index of the best (highest ranking) treatment, as well as the Kruskal-Wallis chi-square statistic, for each voxel.

Section 4 describes Program 3dFriedman, which compares blocked multiple treatments. This program performs the Friedman test for randomized block designs, on a voxel-by-voxel basis. Output includes the index of the best (highest ranking) treatment, as well as the Friedman chi-square statistic, for each voxel.

Contents

1 Program 3dMannWhitney

1.1 Purpose

Program 3dMannWhitney was developed for nonparametric comparison of two treatments, or two samples. This program performs the Wilcoxon-Mann-Whitney rank-sum test on two groups of *AFNI* 3d datasets, voxel-by-voxel, to determine if the two samples are from the same population. Output includes an estimate for the treatment effect, as well as the normalized Wilcoxon rank-sum statistic, for each voxel.

The Wilcoxon-Mann-Whitney rank-sum test is the nonparametric counterpart of the (unpaired data) t-test. As such, program 3dMannWhitney roughly corresponds to program 3dttest, which may be used to compare two samples, assuming the underlying populations are normally distributed.

1.2 Theory

This section contains a very brief summary of material that can be found in references 1-3.

1.2.1 Wilcoxon-Mann-Whitney rank-sum test

Suppose that we have two independent samples, X_1, X_2, …, X_m, and Y_1, Y_2, …, Y_n, and we wish to test whether these samples are from the same population. That is, we wish to test the null hypothesis:

H_o : Y observations and X observations are from the same population

against the alternative hypothesis:

H_a : Y observations tend to be either smaller or larger than the X observations.

First, replace each observation by its rank within the combined list of $m + n$ observations: (We will assume for the moment that no ties are present among the observations.)

$$\begin{aligned} X \text{ ranks} &: Q_1, Q_2, \ldots, Q_m, \\ Y \text{ ranks} &: R_1, R_2, \ldots, R_n. \end{aligned}$$

Now, summing the ranks of the Y observations:

$$W_y = R_1 + \cdots + R_n$$

The expected value for the W_y statistic, under the null hypothesis, is given by:

$$E(W_y) = \frac{1}{2}n(m + n + 1)$$

and the variance is given by:

$$Var(W_y) = \frac{1}{12}mn(m+n+1)$$

Now, if the observed value of W_y is either much smaller or much larger than $E(W_y)$, we have reason to reject the null hypothesis.

In order to calculate p-values for significance of the results, we can make use of the fact that, under the null hypothesis, W_y has an asymptotic normal distribution:

$$\frac{W_y - E(W_y)}{\sqrt{Var(W_y)}} \xrightarrow{d} N(0,1) \quad \text{as} \quad m, n \to \infty.$$

For a discussion of the accuracy of the normal approximation, see reference 1. In general, the accuracy of the approximation is very good for m and n at least 10, and is good for smaller values as long as neither m nor n is too small. However, the approximation becomes less accurate (in a relative sense) as the probability being estimated approaches zero (i.e., in the tails of the distribution). This is a problem for FMRI data when the Bonferroni method is used to maintain the overall significance level for simultaneous inferences involving millions of voxels. Since the Bonferroni method sets the individual voxel probability threshold equal to the desired α-level divided by the total number of voxels, these probabilities can be quite small. And, incidentally, this destroys the statistical power of the test. Therefore, it is strongly recommended that an alternative to the Bonferroni method be used. One alternative is to restrict attention to small regions of interest. Another alternative is to use minimum cluster size thresholding, instead of probability thresholding, to achieve the desired overall α significance level. (See documentation for program AlphaSim.)

So far, we have only considered the case when no ties exist among the data points. When ties are present, the ranks are replaced by the average of the ranks (i.e., the "midranks") of the tied observations:

$$W_y^* = R_1^* + \cdots + R_n^*$$

The expected value of W_y^* is the same as that for W_y :

$$E(W_y^*) = \frac{1}{2}n(m+n+1)$$

However, the variance of W_y^* is reduced by the presence of ties:

$$Var(W_y^*) = \frac{1}{12}mn(m+n+1) - \frac{mn\sum_{i=1}^{e}(d_i^3 - d_i)}{12(m+n)(m+n-1)}$$

where e is the number of distinct values in the combined $m+n$ observations, and d_i is the multiplicity of the ith value. There is still asymptotic normality for W_y^* (provided that the proportion of ties is not too large):

$$Z^* \equiv \frac{W_y^* - E(W_y^*)}{\sqrt{Var(W_y^*)}} \xrightarrow{d} N(0,1)$$

which allows us to calculate the (approximate) p-values for the significance of the differences between the two samples.

Program 3dMannWhitney calculates the Wilcoxon rank-sum statistic W_y^*, along with the expected value $E(W_y^*)$ and the variance $Var(W_y^*)$, for each voxel, and places the normalized statistic Z^* into the second sub-brick of the output *AFNI* "fizt" dataset. Therefore, when using Program afni to view the dataset, the 2nd sub-brick can be used as a threshold, so that only voxels having the user specified statistical significance level will light-up.

1.2.2 Estimation of treatment effect

To estimate the treatment effect, we will assume that the populations from which the samples are drawn have the same shape, but are offset in location:

$$Y = X + \Delta$$

where Δ is the difference in location between the two populations, i.e., Δ is the amount by which the treatment has shifted the response. It may be seen that if X_i is an observation from the first sample, and if Y_j is an observation from the second sample, then $Y_j - X_i$ is an estimate of Δ. Furthermore, each of the mn differences provides an estimate of Δ :

$$Y_1 - X_1, Y_1 - X_2, \ldots, Y_n - X_m$$

We will take as our estimate of Δ the median of these mn differences, as represented by the formula:

$$\hat{\Delta} = med\,(Y_j - X_i)$$

Program 3dMannWhitney calculates this estimate of the population shift parameter for each voxel, and places these estimates in the first sub-brick of an *AFNI* "fizt" dataset. Therefore, when using Program afni to view the dataset, the color coding of the voxels which light-up corresponds to the magnitude of the treatment effect.

1.3 Usage

The command line format for program 3dMannWhitney is as follows:

```
3dMannWhitney \
-dset 1 filename \
        ⋮
-dset 1 filename \
-dset 2 filename \
        ⋮
-dset 2 filename \
[-workmem mega] \
[-voxel num] \
-out prefixname
```

The different command line options are explained below.

1.4 Options

-dset i filename

The -dset command is used to specify the filenames of the *AFNI* 3d datasets to be used as input. The integer i indicates whether the dataset is a member of the first sample ($i = 1$) or the second sample ($i = 2$). It is *not* necessary that the two samples contain equal numbers of datasets.

-workmem mega

The optional -workmem command specifies the number of megabytes of RAM to use for the statistical workspace. The default value is 12. The program will run faster if this value is set higher.

-voxel num

The optional -voxel command is used to send additional output to the screen. The program will display the intermediate calculations of the Wilcoxon-Mann-Whitney rank-sum test for voxel number *num* only.

-out prefixname

The -out command is used to specify the prefix name of the output file to contain the results of the analysis. As indicated below, the output file is an AFNI "fizt" 3d dataset, whose first sub-brick contains the estimated treatment effect $\hat{\Delta}$, and whose second sub-brick contains the normalized Wilcoxon-Mann-Whitney rank-sum statistic Z^*.

$$\textit{AFNI}\text{ “fizt” dataset}\quad \left\{ \begin{array}{l} \boxed{\hat{\Delta} = med\,(Y_j - X_i)} \\ \\ \boxed{Z^* = \dfrac{W_y^* - E(W_y^*)}{\sqrt{Var(W_y^*)}}} \end{array} \right.$$

1.5 Examples

Example 1.

A researcher wishes to study the differences in verbal stimulus neural activation between left-handed and right-handed people. FMRI images were obtained for a group of 10 left-handed subjects and a group of 12 right-handed subjects. The following sequence of commands is used to conduct a nonparametric test for differences in neural activation between the two populations.

Batch Command File for Program 3dMannWhitney

```
3dMannWhitney \
-dset 1 subj101+tlrc \
-dset 1 subj102+tlrc  \
-dset 1 subj103+tlrc  \
                ⋮
-dset 1 subj110+tlrc \
-dset 2 subj201+tlrc \
                ⋮
-dset 2 subj210+tlrc  \
-dset 2 subj211+tlrc  \
-dset 2 subj212+tlrc  \
-workmem 12  \
-voxel 2321701  \
-out verbal.out
```

■

The above -dset commands specify that files subj101+tlrc, ..., subj110+tlrc (.BRIK and .HEAD) contain data for the first sample, and files subj201+tlrc, ..., subj212+tlrc (.BRIK and .HEAD) contain the data for the second sample. The -workmem command specifies that 12 megabytes of memory are to be used for performing the calculations. The -voxel command indicates that the results of the calculations are to be written to the screen for voxel #2321701. Finally, the -out command directs that the program output be written to file verbal.out+tlrc (.BRIK and .HEAD).

Screen Output from Program 3dMannWhitney

```
Last revision: 8 July 1997
Data set dimensions:    nx = 161    ny = 191    nz = 151    nxyz = 4643401
num_pieces = 33    piece_size = 142987
piece = 0
        ⋮
piece = 16

Results for voxel #2321701 :
  X data:
    104.0   223.0   241.0   421.0   375.0   779.0   995.0   963.0   895.0   421.0

  Y data:
    635.0    94.0   103.0    71.0   510.0    23.0    10.0   421.0    71.0   486.0
    541.0   326.0

 X ranks:
      7.0     8.0     9.0    13.0    11.0    19.0    22.0    21.0    20.0    13.0

 Y ranks:
     18.0     5.0     6.0     3.5    16.0     2.0     1.0    13.0     3.5    15.0
     17.0    10.0
Wy = 110.000000
E(Wy) = 138.000000
Var(Wy) = 229.350649
Z = -1.848877

Ordered differences:
 −985.0  −972.0  −953.0  −940.0  −924.0  −924.0  −901.0  −892.0  −892.0  −892.0
 −885.0  −872.0  −869.0  −860.0  −824.0  −824.0  −801.0  −792.0  −769.0  −756.0
 −708.0  −708.0  −685.0  −676.0  −669.0  −637.0  −574.0  −569.0  −542.0  −509.0
 −485.0  −477.0  −474.0  −454.0  −453.0  −453.0  −422.0  −411.0  −411.0  −409.0
 −398.0  −398.0  −385.0  −365.0  −360.0  −358.0  −354.0  −352.0  −350.0  −350.0
 −350.0  −350.0  −328.0  −327.0  −327.0  −318.0  −318.0  −304.0  −304.0  −293.0
 −281.0  −272.0  −269.0  −260.0  −238.0  −231.0  −218.0  −213.0  −200.0  −170.0
 −170.0  −152.0  −152.0  −147.0  −144.0  −138.0  −129.0  −120.0   −95.0   −95.0
  −94.0   −81.0   −49.0   −33.0   −33.0   −10.0    −1.0     0.0     0.0    46.0
   65.0    65.0    85.0    89.0    89.0   103.0   111.0   120.0   120.0   135.0
  166.0   180.0   198.0   214.0   214.0   222.0   245.0   260.0   263.0   269.0
  287.0   300.0   317.0   318.0   382.0   394.0   406.0   412.0   437.0   531.0
Delta hat = -287.000000

piece = 17
      ⋮
piece = 32
— Writing AFNI 'fizt' dataset into ./verbal.out+tlrc.HEAD
```

■

As requested by the -voxel command, the program writes to the screen the results of the calculations performed for voxel #2321701. The 10 input data values for voxel #2321701 for the X sample, and the 12 observations for voxel #2321701 for the Y sample, are listed first. Next, the (mid)ranks of the individual data values within the combined X and Y samples are listed. Note that since there are some tied observations (one value appears twice, and one value appears 3 times), the midranks contain fractional values. The Wilcoxon midrank-sum statistic W_y^*, which is just the sum of the midranks of the Y observations, is:

$$\begin{aligned} W_y^* &= R_1^* + \cdots + R_n^* \\ &= 18 + 5 + 6 + 3.5 + 16 + 2 + 1 + 13 + 3.5 + 15 + 17 + 10 \\ &= 110 \end{aligned}$$

This is followed by the expected value and variance, $E(W_y^*)$ and $Var(W_y^*)$:

$$\begin{aligned} E(W_y^*) &= \frac{1}{2} n(m+n+1) \\ &= \frac{1}{2} \cdot 12(10+12+1) \\ &= 138 \end{aligned}$$

$$\begin{aligned} Var(W_y^*) &= \frac{1}{12} mn(m+n+1) - \frac{mn \sum_{i=1}^{e}(d_i^3 - d_i)}{12(m+n)(m+n-1)} \\ &= \frac{1}{12} \cdot 10 \cdot 12(10+12+1) - \frac{10 \cdot 12 \cdot [(2^3-2)+(3^3-3)]}{12(10+12)(10+12-1)} \\ &= 229.35 \end{aligned}$$

Next, the normalized Wilcoxon statistic Z^* is given by:

$$\begin{aligned} Z^* &= \frac{W_y^* - E(W_y^*)}{\sqrt{Var(W_y^*)}} \\ &= \frac{110 - 138}{\sqrt{229.35}} \\ &= -1.849 \end{aligned}$$

This is followed by a list of all $mn = 120$ differences $Y_i - X_j$. The estimate of the treatment effect is given by the median of these 120 differences, which in this case is $\hat{\Delta} = \frac{-293 + (-281)}{2} = -287.0$. Thus, for this voxel, it is estimated that the Y population is shifted 287 units in the negative direction relative to the X population.

2 Program 3dWilcoxon

2.1 Purpose

Program 3dWilcoxon was developed for nonparametric paired comparison of two treatments. This program performs the Wilcoxon signed-rank test for paired *AFNI* 3d datasets. Output includes the estimate for the treatment effect, and the normalized Wilcoxon signed-rank statistic, for each voxel.

Unlike the Wilcoxon-Mann-Whitney rank-sum test, which makes no assumption about the distribution of the populations, the Wilcoxon signed-rank test assumes that the population (differences between pairs of observations) has a symmetric distribution.

The Wilcoxon signed-rank test is the nonparametric counterpart of the paired data t-test. As such, program 3dWilcoxon roughly corresponds to program 3dttest, which may be used to compare paired samples, assuming that the underlying populations are normally distributed.

2.2 Theory

This section contains a very brief summary of material that can be found in references 1-3.

2.2.1 Wilcoxon signed-rank test

Suppose that we have two samples (perhaps corresponding to two different treatments) of n data points:

$$\begin{array}{cccc} X_1, & X_2, & \ldots, & X_n \\ Y_1, & Y_2, & \ldots, & Y_n \end{array}$$

Further, suppose that there is a natural pairing in the data, i.e., X_1 is paired with Y_1, X_2 is paired with Y_2, etc. This might occur if the sub-index represents different people, and the X and Y samples represent two different tests that are given to each of the n subjects. One might expect that there is a large natural variation from subject to subject, in addition to the difference between the two tests. In this case, it would be disadvantageous to use the Wilcoxon-Mann-Whitney rank-sum test to test for a difference between the underlying populations, since this test does not take into account this subject-to-subject variation.

A better approach in this case is to use the Wilcoxon signed-rank test. This test is performed on the differences between the pairs of data:

$$D_i = Y_i - X_i, \quad i = 1, \ldots, n.$$

The absolute values of these differences are then ranked, but the signs of the differences (+, 0, -) are attached to the ranks by multiplying the rank by +1, 0, or -1, respectively. The following table illustrates this.

X :	16	18	73	57	30	81
Y :	4	47	23	85	45	57
$\|D\|$:	12	29	50	28	15	24
signed rank :	-1	$+5$	-6	$+4$	$+2$	-3

The Wilcoxon signed-rank statistic is formed by taking the sum of the *positive* ranks:

$$W_+ = R_1 + \cdots + R_k$$

where k out of the n ranks are positive. We are assuming, for the moment, that there are no ties among the differences, and that none of the differences is zero. In the above example,

$$W_+ = 5 + 4 + 2 = 11$$

It is obvious that a large value for W_+ would tend to indicate that the Y values are larger than the X values. Under the null hypothesis that the differences are symmetrically distributed about zero, the expected value and the variance of the W_+ statistic are given by:

$$E(W_+) = \frac{n(n+1)}{4}$$

$$Var(W_+) = \frac{n(n+1)(2n+1)}{24}$$

If the observed value for W_+ is close to $E(W_+)$, then we do not have reason to reject the null hypothesis. On the other hand, if W_+ is either much larger or much smaller than $E(W_+)$, then we do have reason to reject the null hypothesis.

To calculate the p-value corresponding to a set of data, we can make use of the fact that the W_+ statistic is asymptotically normal:

$$\frac{W_+ - E(W_+)}{\sqrt{Var(W_+)}} \xrightarrow{d} N(0,1) \quad \text{as} \quad n \to \infty$$

For a discussion of the accuracy of the normal approximation, see reference 1. In general, the accuracy of the approximation is very good for $n \geq 20$ and α not too small. However, the approximation becomes less accurate (in a relative sense) as the probability being estimated approaches zero (i.e., in the tails of the distribution). This is a problem for FMRI data when the Bonferroni method is used to maintain the overall significance level for simultaneous inferences involving millions of voxels. Since the Bonferroni method sets the individual voxel probability threshold equal to the desired α-level divided by the total number of voxels, these probabilities can be quite small. And, incidentally, this destroys the statistical power of the test. Therefore, it is strongly recommended that an alternative to the Bonferroni method be used. One alternative is to restrict attention to small regions of interest. Another alternative is to use minimum cluster size thresholding, instead of probability thresholding, to achieve the desired overall α significance level. (See documentation for program AlphaSim.)

Until now, we have assumed that there are no ties among the differences, and that no difference is equal to zero. When there are ties among the differences, or when a difference is equal to zero, we use the sum of the positive signed midranks:

$$W_+^* = R_1^* + \cdots + R_k^*$$

For example, if we have

X :	31	29	23	45	74	38
Y :	66	35	20	45	68	73
$\lvert D\rvert$:	35	6	3	0	6	35
signed midrank :	5.5	3.5	-2	0	-3.5	5.5

then the sum of the positive signed midranks is

$$W_+^* = 5.5 + 3.5 + 5.5 = 14.5$$

The expected value for W_+^*, under the null hypothesis, is given by:

$$E(W_+^*) = \frac{n(n+1) - d_0(d_0+1)}{4}$$

where d_0 is the number of zero differences. The formula for the variance of W_+^* is

$$\begin{aligned} Var(W_+^*) &= \frac{1}{24}\left[n(n+1)(2n+1) - d_0(d_0+1)(2d_0+1)\right] \\ &\quad -\frac{1}{48}\sum_{i=1}^{e} d_i(d_i-1)(d_i+1) \end{aligned}$$

where d_0 is again the number of zero differences, and the d_i are the multiplicities of the absolute values of the nonzero differences. We again have a normal approximation for the W_+^* statistic:

$$Z^* \equiv \frac{W_+^* - E(W_+^*)}{\sqrt{Var(W_+^*)}} \xrightarrow{d} N(0,1)$$

Program 3dWilcoxon calculates the Wilcoxon signed-rank statistic W_+^*, along with the expectation $E(W_+^*)$ and variance $Var(W_+^*)$, for each voxel, and places the normalized estimate Z^* in the second sub-brick of the output *AFNI* "fizt" dataset. Therefore, when using Program afni to view the dataset, the 2nd sub-brick can be used as a threshold, so that only voxels having the user specified statistical significance level will light-up.

2.2.2 Estimation of treatment effect

Estimation of the treatment effect is also accomplished by examining the differences D_i between the pairs of data points. In fact, any one difference D_i would provide an estimate of the median of the difference between the X and Y populations. A better estimate is provided by the median of all the differences, $\tilde{\Delta} = med\{D_1, D_2, \ldots, D_n\}$. However, in conjunction with the Wilcoxon signed-rank statistic, the usual procedure is to take the median of the averages of all pairs of differences (referred to as the Walsh averages):

$$\hat{\Delta} = \underset{i \le j}{med}\left\{\frac{1}{2}(D_i + D_j)\right\}$$

Program 3dWilcoxon calculates the median of the $\frac{1}{2}n(n+1)$ Walsh averages for each voxel, and places this estimate in the first sub-brick of the output *AFNI* "fizt" dataset. Therefore, when using Program afni to view the dataset, the color coding of the voxels which light-up corresponds to the magnitude of the treatment effect.

2.3 Usage

The command line format for program 3dWilcoxon is as follows:

```
3dWilcoxon \
-dset 1 filename \
        ⋮
-dset 1 filename \
-dset 2 filename \
        ⋮
-dset 2 filename \
[-workmem mega] \
[-voxel num] \
-out prefixname
```

The different command line options are explained below.

2.4 Options

-dset i filename

The -dset command is used to specify the filenames of the *AFNI* 3d datasets to be used as input. The integer i indicates whether the dataset is a member of the first sample $(i = 1)$ or the second sample $(i = 2)$. Of course, the number of datasets entered for the first sample *must* equal the number of datasets entered for the second sample. Further, it is assumed that the datasets are paired in the order in which they are entered, i.e., the first dataset entered for the first sample is paired with the first dataset entered for the second sample, etc.

-workmem mega

The optional -workmem command specifies the number of megabytes of RAM to use for the statistical workspace. The default value is 12. The program will run faster if this value is set higher.

-voxel num

The optional -voxel command is used to send additional output to the screen. The program displays the intermediate results of the Wilcoxon signed-rank test for voxel number *num* only.

-out prefixname

The -out command is used to specify the prefix name of the output file to contain the results of the analysis. As indicated below, the output file is an AFNI "fizt" 3d dataset, whose first sub-brick contains the estimated treatment effect $\hat{\Delta}$, and whose second sub-brick contains the normalized Wilcoxon signed-rank statistic Z^*.

$$\textit{AFNI}\ \text{``fizt'' dataset}\quad \left\{ \begin{array}{l} \boxed{\hat{\Delta} = \underset{i\le j}{med}\left\{\frac{1}{2}\left(D_i+D_j\right)\right\}} \\[2ex] \boxed{Z^* = \frac{W_+^* - E(W_+^*)}{\sqrt{Var(W_+^*)}}} \end{array}\right.$$

2.5 Examples

Example 1.

A researcher wishes to study differences in neural activation due to differences in language acquistion. The study was designed using subjects fluent in two different languages. Since there is large subject-to-subject variation, a paired comparison nonparametric test should be performed. Twelve subjects were used. The first set of FMRI data represents test results for each subject's primary language, and the second set of results are for the subject's secondary language. The commands necessary to perform the Wilcoxon signed-rank test for paired comparisons are presented below.

Batch Command File for Program 3dWilcoxon

```
3dWilcoxon  \
-dset 1 subj101+tlrc \
-dset 1 subj102+tlrc \
-dset 1 subj103+tlrc \
                ⋮
-dset 1 subj112+tlrc \
-dset 2 subj201+tlrc \
                ⋮
-dset 2 subj210+tlrc  \
-dset 2 subj211+tlrc  \
-dset 2 subj212+tlrc  \
-voxel 2321701  \
-out language.out \
```

■

The above -dset commands specify that files subj101+tlrc and subj201+tlrc (.BRIK and .HEAD) contain data for the first pair, ..., and files subj112+tlrc and subj212+tlrc (.BRIK and .HEAD) contain the data for the last pair. The -voxel command indicates that the results of the calculations are to be written to the screen for voxel #2321701. Finally, the -out command directs that the program output be written to file language.out+tlrc (.BRIK and .HEAD). The screen output is listed below.

Screen Output for Program 3dWilcoxon

```
Program 3dWilcoxon

Last revision: 8 July 1997
Data set dimensions:   nx = 161   ny = 191   nz = 151   nxyz = 4643401
num_pieces = 36   piece_size = 131072
piece = 0
        ⋮
piece = 17

Results for voxel #2321701 :

         X data:
           713.0    701.0  497.0  499.0  708.0    557.0    276.0    907.0  251.0    890.0
           992.0    651.0

         Y data:
            11.0    326.0  914.0  874.0  708.0     75.0     32.0    633.0  449.0    716.0
           380.0    648.0

          Y - X:
          −702.0   −375.0  417.0  375.0    0.0   −482.0   −244.0   −274.0  198.0   −174.0
          −612.0     −3.0

  Signed Ranks:
           −12.0     −7.5    9.0    7.5    0.0    −10.0     −5.0     −6.0    4.0     −3.0
           −11.0     −2.0

W+ = 20.500000
E(W+) = 38.500000
Var(W+) = 162.125000
Z = -1.413668

Ordered Walsh averages:
```

```
−702.0  −657.0  −612.0  −592.0  −547.0  −538.5  −493.5  −488.0  −482.0  −473.0
−443.0  −438.0  −428.5  −428.0  −393.0  −378.0  −375.0  −363.0  −352.5  −351.0
−328.0  −324.5  −309.5  −307.5  −306.0  −274.5  −274.0  −259.0  −252.0  −244.0
−242.5  −241.0  −224.0  −209.0  −207.0  −189.0  −187.5  −174.0  −163.5  −142.5
−142.0  −138.5  −137.0  −123.5  −122.0  −118.5   −97.5   −88.5   −88.5   −87.0
 −53.5   −38.0   −32.5   −23.0    −3.0    −1.5     0.0     0.0    12.0    21.0
  50.5    65.5    71.5    86.5    97.5    99.0   100.5   121.5   186.0   187.5
 198.0   207.0   208.5   286.5   307.5   375.0   396.0   417.0
Delta hat = -153.000000

piece = 18
      ⋮
piece = 35
— Writing AFNI 'fizt' dataset into ./language.out+tlrc.HEAD
```

■

The Wilcoxon signed-rank statistic W_+^* is found by summing the *positive* ranks:

$$\begin{aligned} W_+^* &= R_1^* + \cdots + R_k^* \\ &= 9 + 7.5 + 4 \\ &= 20.5 \end{aligned}$$

The expected value of W_+^* is calculated (noting that there is 1 zero difference):

$$\begin{aligned} E(W_+^*) &= \frac{n(n+1) - d_0(d_0+1)}{4} \\ &= \frac{12(12+1) - 1(1+1)}{4} \\ &= 38.5 \end{aligned}$$

and the variance (noting there is one tied absolute difference of multiplicity 2):

$$\begin{aligned} Var(W_+^*) &= \frac{1}{24}\left[n(n+1)(2n+1) - d_0(d_0+1)(2d_0+1)\right] \\ &\quad - \frac{1}{48}\sum_{i=1}^{e} d_i(d_i-1)(d_i+1) \\ &= \frac{1}{24}\left[12(12+1)(2\cdot 12+1) - 1(1+1)(2\cdot 1+1)\right] \\ &\quad - \frac{1}{48}\left[2(2-1)(2+1)\right] \\ &= 162.125 \end{aligned}$$

The normalized Wilcoxon signed-rank statistic is therefore

$$Z^* \equiv \frac{W_+^* - E(W_+^*)}{\sqrt{Var(W_+^*)}}$$

$$= \frac{20.5 - 38.5}{\sqrt{162.125}} = -1.4137$$

The $\frac{1}{2}(12)(13) = 78$ Walsh averages are listed for voxel #2321701. The treatment effect is estimated by taking the median of the set of Walsh averages, which in this case is:

$$\begin{aligned} \hat{\Delta} &= \underset{i \leq j}{med} \left\{ \frac{1}{2} (D_i + D_j) \right\} \\ &= \frac{-163.5 + (-142.5)}{2} = -153.0 \end{aligned}$$

Thus, the secondary language seems to produce 153 units less activation than the primary language, in this voxel.

3 Program 3dKruskalWallis

3.1 Purpose

Program 3dKruskalWallis was developed for comparing multiple treatments. This program performs the Kruskal-Wallis test for whether any of s treatments are different. Output includes the index of the best (highest ranking) treatment, as well as the Kruskal-Wallis chi-square statistic, for each voxel.

The Kruskal-Wallis test is the nonparametric counterpart of the one-way ANOVA. As such, program 3dKruskalWallis roughly corresponds to program 3dANOVA, which may be used to compare multiple treatments, assuming that the underlying populations are normally distributed with equal variances.

3.2 Theory

This section contains a very brief summary of material that can be found in references 1-3.

3.2.1 Kruskal-Wallis test

If more than two treatments are being compared, then the Kruskal-Wallis test is appropriate (we are assuming that there is no blocking of the data). So, suppose that we are trying to determine is there are any differences among s treatments. Let treatment i have n_i observations, so that the total number of observations is given by:

$$N = n_1 + n_2 + \cdots + n_s.$$

For example, suppose that we are comparing the results of $s = 4$ different tests, which are given to randomly selected subjects. The sample sizes for the 4 tests are $n_1 = 3$, $n_2 =$

5, $n_3 = 5$, and $n_4 = 4$. The measured results are listed below:

Y_1	Y_2	Y_3	Y_4
2.25	1.00	0.31	0.23
0.55	0.21	0.77	−1.27
−1.20	−0.31	1.30	0.11
	0.35	−1.32	−2.27
	0.63	−0.67	

The first step is to rank each of the observations within the entire set of $N = n_1 + n_2 + n_3 + n_4 = 17$ data points.

	R_1	R_2	R_3	R_4
	17	15	10	9
	12	8	14	3
	4	6	16	7
		11	2	1
		13	5	
$\frac{R_{i.}}{n_i}$	11.0	10.6	9.4	5.0

As indicated above, the ranks are summed within each treatment:

$$R_{i.} = R_{i1} + \cdots R_{in_i}$$

and the average rank for a treatment is obtained by dividing the rank sum by the number of observations within that treatment:

$$\frac{R_{i.}}{n_i} = \text{average rank within treatment } i$$

Since the sum of all ranks for all treatments is equal to $\frac{N(N+1)}{2}$, the average of all ranks is just $\frac{N+1}{2}$. So, if the null hypothesis of no treatment effect is correct, then we would expect the average rank within each treatment to be close to $\frac{N+1}{2}$. Therefore, if the null hypothesis is correct, then $\frac{R_{i.}}{n_i} - \frac{N+1}{2}$ should be close to zero, for $i = 1, 2, \ldots, s$. The Kruskal-Wallis K statistic is actually defined as follows:

$$\begin{aligned} K &= \frac{12}{N(N+1)} \sum_{i=1}^{s} n_i \left(\frac{R_{i.}}{n_i} - \frac{N+1}{2} \right)^2 \\ &= \frac{12}{N(N+1)} \sum_{i=1}^{s} \frac{R_{i.}^2}{n_i} - 3(N+1) \end{aligned}$$

It may be seen that a large value for K is evidence against the null hypothesis, whereas a small value for K tends to support the null hypothesis.

For large sample sizes, the K statistic has an approximate chi-square distribution with $s-1$ degrees of freedom:

$$K \xrightarrow{d} \chi^2(s-1)$$

For a discussion of the accuracy of the chi-square approximation, see reference 1. In general, the accuracy of the approximation is adequate if α is not too small, and either $s = 3$ and all $n_i > 5$, or $s > 3$ and all $n_i > 4$. However, the approximation becomes less accurate (in a relative sense) as the probability being estimated approaches zero (i.e., in the tails of the distribution). This is a problem for FMRI data when the Bonferroni method is used to maintain the overall significance level for simultaneous inferences involving millions of voxels. Since the Bonferroni method sets the individual voxel probability threshold equal to the desired α-level divided by the total number of voxels, these probabilities can be quite small. And, incidentally, this destroys the statistical power of the test. Therefore, it is strongly recommended that an alternative to the Bonferroni method be used. One alternative is to restrict attention to small regions of interest. Another alternative is to use minimum cluster size thresholding, instead of probability thresholding, to achieve the desired overall α significance level. (See documentation for program AlphaSim.)

The above assumes that there are no ties in the data. When ties are present, the midranks are used:

$$R_{i.}^* = R_{i1}^* + \cdots R_{in_i}^*$$

This requires the following modification of the Kruskal-Wallis statistic:

$$K^* = \frac{[12/N(N+1)]\sum_{i=1}^{s} R_{i.}^{*2}/n_i - 3(N+1)}{1 - \sum_{i=1}^{e}(d_i^3 - d_i)/(N^3 - N)}$$

where e is the number of distinct values in the entire set of data, and d_i is the multiplicity of the ith data value. Again, we have the asymptotic distribution of the Kruskal-Wallis statistic:

$$K^* \xrightarrow{d} \chi^2(s-1)$$

Program 3dKruskalWallis calculates the K^* statistic for each voxel, and places these statistics in the second sub-brick of the output *AFNI* "fict" dataset. Therefore, when using Program afni to view the dataset, the 2nd sub-brick can be used as a threshold, so that only voxels having the user specified statistical significance level will light-up.

3.3 Usage

The command line format for program 3dKruskalWallis is as follows:

```
3dKruskalWallis \
-levels s \
-dset 1 filename \
        ⋮
-dset 1 filename \
        ⋮
-dset s filename \
        ⋮
-dset s filename \
[-workmem mega] \
[-voxel num] \
-out prefixname
```

The different command line options are explained below.

3.4 Options

-levels s

The mandatory -levels command is used to indicate the number of different treatments (factor levels) to be considered. The number of levels s must satisfy: $2 \leq s \leq 100$. Note: the -levels command must appear prior to any -dset command.

-dset i filename

The -dset command is used to specify the filenames of the *AFNI* 3d datasets to be used as input to program 3dKruskalWallis. The integer i indicates which treatment was applied to that particular dataset ($i = 1, \ldots, s$). It is *not* necessary that different treatments have the same number of datasets.

-workmem mega

The optional -workmem command specifies the number of megabytes of RAM to use for the statistical workspace. The default value is 12. The program will run faster if this value is set higher.

-voxel num

The optional -voxel command is used to send additional output to the screen. The program displays the intermediate results of the Kruskal-Wallis test for voxel number *num* only.

-out prefixname

The -out command is used to specify the prefix name of the output file to contain the results of the analysis. As indicated below, the output file is an AFNI "fict" 3d dataset, whose first sub-brick contains the index number for the treatment which has the greatest effect, and whose second sub-brick contains the Kruskal-Wallis statistic K^*.

$$\textit{AFNI}\ \text{“fict” dataset}\quad \left\{ \begin{array}{l} \boxed{I = i, \text{ where } \dfrac{R^*_{i.}}{n_i} = \max\limits_{j=1,\ldots,s}\left\{\dfrac{R^*_{j.}}{n_j}\right\}} \\ \\ \boxed{K^* = \dfrac{[12/N(N+1)]\sum_{i=1}^{s} R^{*2}_{i.}/n_i - 3(N+1)}{1-\sum_{i=1}^{e} (d_i^3 - d_i)/(N^3-N)}} \end{array}\right.$$

3.5 Examples

Example 1.

Five different drugs, labeled A, B, C, D, and E, are to be tested for differences in the resulting neural activation. Twenty seven subjects are assigned, at random, to receive one of the 5 drugs. Five each are administered drugs A, B, and C, while drugs D and E are each administered to 6 subjects. To analyze the FMRI data using the nonparametric Kruskal-Wallis test to determine if there is any difference in neural activation due to differences among the drugs, the following commands were used:

Batch Command File for Program 3dKruskalWallis

```
3dKruskalWallis \
-levels 5 \
-dset 1 a01+tlrc \
-dset 1 a02+tlrc \
-dset 1 a03+tlrc \
           ⋮
-dset 5 e04+tlrc \
-dset 5 e05+tlrc \
-dset 5 e06+tlrc \
-voxel 2321701 \
-out drug.out
```

■

The -levels command indicates that there are 5 treatments. The input filenames are then listed on the following lines, each preceded by -dset k, where k = (drug) treatment index number. Output is to be sent to file drug.out+tlrc (.BRIK and .HEAD).

The screen output generated for voxel #2321701 is listed below.

Screen Output from Program 3dKruskalWallis

```
Program 3dKruskalWallis

Last revision: 08 July 1997
Data set dimensions:    nx = 161    ny = 191    nz = 151    nxyz = 4643401
num_pieces = 40    piece_size = 116508
piece = 0
        ⋮
piece = 19

Results for voxel #2321701 :

  Y1 data:  131.0  106.0  120.0  145.0  174.0
  Y2 data:  138.0  119.0  119.0  131.0  139.0
  Y3 data:  191.0  188.0  151.0  129.0  167.0
  Y4 data:  106.0  148.0  103.0  125.0  143.0  168.0
  Y5 data:  118.0  145.0  105.0  117.0  178.0  111.0

 Y1 ranks:   13.5    3.5   10.0   18.5   24.0
 Y2 ranks:   15.0    8.5    8.5   13.5   16.0
 Y3 ranks:   27.0   26.0   21.0   12.0   22.0
 Y4 ranks:    3.5   20.0    1.0   11.0   17.0   23.0
 Y5 ranks:    7.0   18.5    2.0    6.0   25.0    5.0

 Y1:   Rank sum = 69.5     Rank average = 13.9
 Y2:   Rank sum = 61.5     Rank average = 12.3
 Y3:   Rank sum = 108.0    Rank average = 21.6
 Y4:   Rank sum = 75.5     Rank average = 12.6
 Y5:   Rank sum = 63.5     Rank average = 10.6

K = 6.124674

piece = 20
        ⋮
piece = 39
— Writing AFNI 'fict' dataset into ./drug.out+tlrc.HEAD
```

■

The Kruskal-Wallis K^* statistic is easily verified using the above rank sums (and the fact that there are 4 values of multiplicity 2):

$$K^* \quad = \quad \frac{[12/N(N+1)] \sum_{i=1}^{s} R_{i.}^{*2}/n_i - 3(N+1)}{1 - \sum_{i=1}^{e} (d_i^3 - d_i)/(N^3 - N)}$$

$$= \frac{\frac{12}{27(28)}\left[\frac{69.5^2}{5}+\frac{61.5^2}{5}+\frac{108.0^2}{5}+\frac{75.5^2}{6}+\frac{63.5^2}{6}\right]-3(28)}{1-4(2^3-2)/(27^3-27)}$$
$$= 6.125$$

The probability of obtaining $K^* \geq 6.125$, where K^* has the $\chi^2(4)$ distribution is $p \approx 0.19$. Thus, for this voxel, the p-value is approximately 0.19.

4 Program 3dFriedman

4.1 Purpose

Program 3dFriedman compares blocked multiple treatments. This program performs the nonparametric Friedman test for randomized complete block design experiments, on a voxel-by-voxel basis. Output includes the index of the best (highest ranking) treatment, as well as the Friedman chi-square statistic, for each voxel.

The Friedman test is the nonparametric counterpart of the mixed effects two-way ANOVA. As such, program 3dFriedman roughly corresponds to program 3dANOVA2, which may be used to compare blocked multiple treatments, assuming that the underlying populations are normally distributed with equal variances.

4.2 Theory

This section contains a very brief summary of material that can be found in references 1-3.

4.2.1 The Friedman test

Here, we consider the case where there are multiple treatments, and there is blocking of the data. An example of this sort of experiment is presented below. This might arise if 5 different subjects were each subjected to 4 different tests. In this case, the blocking would be by the individual subject, since there might be large subject-to-subject variation.

	Y_1	Y_2	Y_3	Y_4
Block 1	866	414	977	419
Block 2	541	681	421	521
Block 3	414	941	205	222
Block 4	942	683	479	982
Block 5	995	882	291	374

The Friedman test seeks to remove this variation between blocks by ranking the data within blocks only, as illustrated below. We assume for now that no ties exist within a block.

	R_1	R_2	R_3	R_4
Block 1	3	1	4	2
Block 2	3	4	1	2
Block 3	3	4	1	2
Block 4	3	2	1	4
Block 5	4	3	1	2
$\frac{R_{i.}}{n}$	3.2	2.8	1.6	2.4

The sum of the ranks is computed for each treatment.

$$R_{i.} = R_{i1} + \cdots R_{in}$$

So, the average rank for an individual treatment is $\frac{R_{i.}}{n}$.

If there are s treatments and n blocks, then the sum of all ranks for all treatments is $n\left(\frac{s(s+1)}{2}\right)$, hence the average rank for an individual observation is $\frac{s+1}{2}$. So, if the null hypothesis of no treatment effect is correct, then we would expect the average rank within each treatment to be close to $\frac{s+1}{2}$. Therefore, if the null hypothesis is correct, then $\frac{R_{i.}}{n} - \frac{s+1}{2}$ should be close to zero, for $i = 1, 2, \ldots, s$. The Friedman Q statistic is actually defined as follows:

$$\begin{aligned} Q &= \frac{12n}{s(s+1)} \sum_{i=1}^{s} \left(\frac{R_{i.}}{n_i} - \frac{s+1}{2}\right)^2 \\ &= \frac{12}{ns(s+1)} \sum_{i=1}^{s} R_{i.}^2 - 3n(s+1) \end{aligned}$$

It may be seen that a large value for Q is evidence against the null hypothesis, whereas a small value for Q tends to support the null hypothesis.

For large sample sizes, the Q statistic has an approximate chi-square distribution with $s-1$ degrees of freedom.

$$Q \xrightarrow{d} \chi^2(s-1)$$

For a discussion of the accuracy of the chi-square approximation, see reference 1. In general, the accuracy of the approximation is adequate if α is not too small, and $sn \geq 30$. However, the approximation becomes less accurate (in a relative sense) as the probability being estimated approaches zero (i.e., in the tails of the distribution). This is a problem for FMRI data when the Bonferroni method is used to maintain the overall significance level for simultaneous inferences involving millions of voxels. Since the Bonferroni method sets the individual voxel probability threshold equal to the desired α-level divided by the total number of voxels, these probabilities can be quite small. And, incidentally, this destroys the statistical power of the test. Therefore, it is strongly recommended that an alternative to the Bonferroni method be used. One alternative is to restrict attention to small regions of interest. Another alternative is to use minimum cluster size thresholding, instead of probability thresholding, to achieve the desired overall α significance level. (See documentation for program AlphaSim.)

The above assumes that there are no ties in the data. When ties are present, the midranks are used:

$$R^*_{i.} = R^*_{i1} + \cdots R^*_{in}$$

This requires the following modification of the Friedman statistic:

$$Q^* = \frac{[12/ns(s+1)]\sum_{i=1}^{s} R^{*2}_{i.} - 3n(s+1)}{1 - \sum_{j=1}^{n}\sum_{i=1}^{e_j}(d^3_{ij} - d_{ij})/ns(s^2-1)}$$

where e_j is the number of distinct values in the jth block, and d_{ij} is the multiplicity of the ith data value within the jth block. Again, we have the asymptotic distribution of the Friedman statistic:

$$Q^* \xrightarrow{d} \chi^2(s-1)$$

Program 3dFriedman calculates the Q^* statistic for each voxel, and places these statistics in the second sub-brick of the output *AFNI* "fict" dataset. Therefore, when using Program afni to view the dataset, the 2nd sub-brick can be used as a threshold, so that only voxels having the user specified statistical significance level will light-up.

4.3 Usage

The command line format for program 3dFriedman is as follows:

```
3dFriedman \
-levels s \
-dset 1 filename \
       ⋮
-dset 1 filename \
       ⋮
-dset s filename \
       ⋮
-dset s filename \
[-workmem mega] \
[-voxel num] \
-out prefixname
```

The different command line options are explained below.

4.4 Options

-levels s

The mandatory -levels command is used to indicate the number of different treatments (factor levels) to be considered. The number of levels s must satisfy: $2 \leq s \leq 100$. Note: the -levels command must appear prior to any -dset command.

-dset i filename

The -dset command is used to specify the filenames of the *AFNI* 3d datasets to be used as input to program 3dFriedman. The integer i indicates which treatment was applied to that particular dataset ($i = 1, \ldots, s$). All treatments *must* have the same number n of datasets. Further, it is assumed that the datasets are blocked in the order in which they are entered, i.e., the first dataset entered for each treatment is in block 1, the second dataset entered for each treatment is in block 2, etc.

-workmem mega

The optional -workmem command specifies the number of megabytes of RAM to use for the statistical workspace. The default value is 12. The program will run faster if this value is set higher.

-voxel num

The optional -voxel command is used to send additional output to the screen. The program displays the intermediate results of the Friedman test for voxel number *num* only.

-out prefixname

The -out command is used to specify the prefix name of the output file to contain the results of the analysis. As indicated below, the output file is an AFNI "fict" 3d dataset, whose first sub-brick contains the index number for the treatment which has the greatest effect, and whose second sub-brick contains the Friedman statistic Q^*.

$$\textit{AFNI}\ \text{“fict” dataset}\quad \begin{cases} \boxed{I = i, \text{ where } \frac{R^*_{i.}}{n} = \max_{j=1,\ldots,s} \left\{ \frac{R^*_{j.}}{n} \right\}} \\ \\ \boxed{Q^* = \frac{[12/ns(s+1)] \sum_{i=1}^{s} R^{*2}_{i.} - 3n(s+1)}{1 - \sum_{j=1}^{n} \sum_{i=1}^{e_j} (d^3_{ij} - d_{ij})/ns(s^2-1)}} \end{cases}$$

4.5 Examples

Example 1.

Three different tests are to be compared for differences in neural activation. Fifteen subjects are each given the three tests, in randomized order. To reduce the variation in the results, blocking is used (with an individual subject constituting a block). The following commands are used to perform the nonparametric Friedman test to determine if there is a statistically significant difference among the tests (at each voxel).

Batch Command File for Program 3dFriedman

```
3dFriedman  \
-levels 3  \
-dset 1 testa.subj01+tlrc  \
-dset 1 testa.subj02+tlrc  \
-dset 1 testa.subj03+tlrc  \
                 ⋮
-dset 3 testc.subj13+tlrc  \
-dset 3 testc.subj14+tlrc  \
-dset 3 testc.subj15+tlrc  \
-voxel 2321701  \
-out tests.out
```

■

The -levels command indicates that there are 3 treatments (tests). The input filenames are then listed on the following lines, each preceded by -dset k, where k = test number. Output is to be sent to file tests.out+tlrc (.BRIK and .HEAD).

The screen output generated for voxel #2321701 is listed below.

Screen Output from Program 3dFriedman

```
Program 3dFriedman

Last revision: 08 July 1997
Data set dimensions:   nx = 161   ny = 191   nz = 151   nxyz = 4643401
num_pieces = 67   piece_size = 69905
piece = 0
        ⋮
piece = 33

Results for voxel #2321701 :

 Y1 data:  3.0  8.0  1.0  2.0  8.0  5.0  1.0  1.0  7.0  8.0  7.0  5.0  0.0  9.0  6.0
 Y2 data:  1.0  3.0  6.0  0.0  9.0  1.0  6.0  1.0  1.0  0.0  7.0  3.0  5.0  3.0  3.0
 Y3 data:  4.0  2.0  5.0  6.0  4.0  5.0  9.0  8.0  7.0  0.0  2.0  9.0  3.0  9.0  9.0

Y1 ranks:  2.0  3.0  1.0  2.0  2.0  2.5  1.0  1.5  2.5  3.0  2.5  2.0  1.0  2.5  2.0
Y2 ranks:  1.0  2.0  3.0  1.0  3.0  1.0  2.0  1.5  1.0  1.5  2.5  1.0  3.0  1.0  1.0
Y3 ranks:  3.0  1.0  2.0  3.0  1.0  2.5  3.0  3.0  2.5  1.5  1.0  3.0  2.0  2.5  3.0
```

```
  Y1:   Rank sum = 30.5   Rank average = 2.0
  Y2:   Rank sum = 25.5   Rank average = 1.7
  Y3:   Rank sum = 34.0   Rank average = 2.3

Q = 2.703704

piece = 34
        ⋮
piece = 66
— Writing AFNI 'fict' dataset into ./tests.out+tlrc.HEAD
```

■

The Friedman Q^* statistic is easily verified using the above rank sums (and the fact that 6 blocks have a tie of multiplicity 2):

$$
\begin{aligned}
Q^* &= \frac{[12/ns(s+1)]\sum_{i=1}^{s} R_{i.}^{*2} - 3n(s+1)}{1 - \sum_{j=1}^{n}\sum_{i=1}^{e_j}(d_{ij}^3 - d_{ij})/ns(s^2-1)} \\
&= \frac{[12/(15\cdot 3\cdot 4)]\,(30.5^2 + 25.5^2 + 34.0^2) - 3\cdot 15(4)}{1 - [6(2^3-2)]\,/\,(15\cdot 3(3^2-1))} \\
&= 2.4333/\frac{9}{10} = 2.7037
\end{aligned}
$$

The probability of obtaining $Q^* \geq 2.7037$, where Q^* has the $\chi^2(2)$ distribution, is $p \approx 0.26$. Thus, for this voxel, the p-value is 0.26.

5 References

1. E. L. Lehmann, *Nonparametrics: Statistical Methods Based on Ranks.* Oakland, CA: McGraw-Hill (1975).

2. R. G. Miller Jr., *Beyond ANOVA, Basics of Applied Statistics.* New York: John Wiley & Sons (1986).

3. G. E. Noether, *Introduction to Statistics: The Nonparametric Way.* New York: Springer-Verlag (1991).