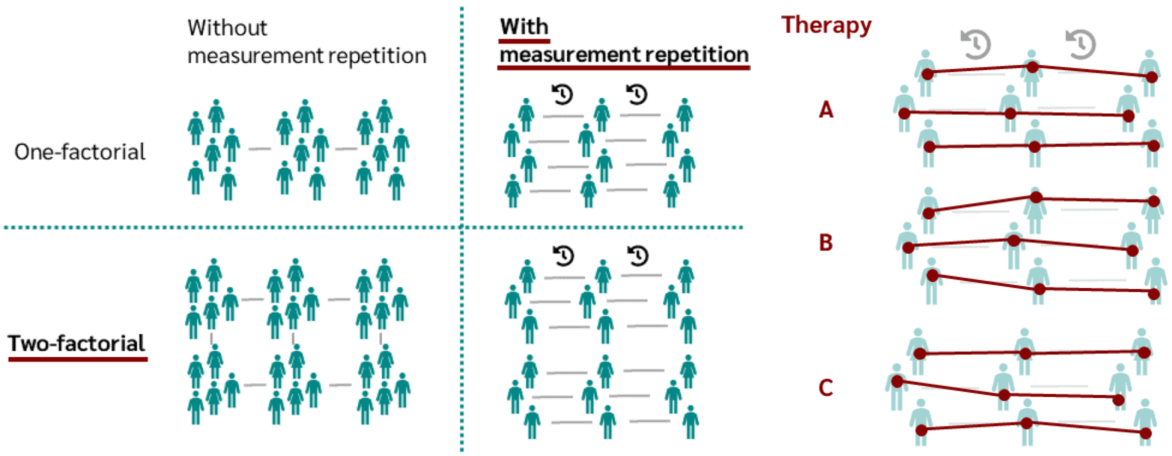
Comparing Centers of Several Independent Groups

EN5423 | Spring 2024

w14_several_independent_04.pdf
(Week 14)

Contents



1 Test Repeated Measures—The Extension of Matched-pair Tests

• Previously (week 09), tests for differences between ***matched pairs of observations were discussed***. Each pair of observations had one value in each of two groups, such as before versus after.

• The advantage of this design was that it blocks out the differences from one matched pair (row) to another and so removes unwanted noise. Such matching (or blocking) schemes can be extended ***to test differences among more than two groups***.

• One observation is available for each combination of factor (columns) and block (rows) to test for factor effects. This commonly used design is called ***repeated measures***, as well as a ***randomized complete block design***, and in the case of parametric assumptions it is called the ***two-way ANOVA without replication***.

• One example at the beginning of this chapter (Comparing Centers of Several Independent Groups)—detecting differences between ***three sampling*** or ***extraction methods*** used at numerous wells—illustrates this design.

• The factor tested is the sampling or extraction method, of which there are three types. The blocking effect is the well location; ***the well-to-well differences are to be blocked out***. One sample is analyzed for each sampling or extraction method at each well. With this design, observations, y_{ij}, are broken down into four contributions:

$$y_{ij} = \mu + \gamma_j + \delta_i + \epsilon_{ij}$$

where

y_{ij}	is the individual observation in block *i* and group *j*;
μ	is the overall mean or median (over all groups)
γ_j	is the *j*th group effect, $j = 1, 2, ..., k$
δ_i	is the *i*th block effect, $i = 1, 2, ..., n$
ϵ_{ij}	is the residual difference between the individual observation and the combined group and block effects.

• Median polish provides resistant estimates of group and block effects.

• ***It is an exploratory technique, not a hypothesis test***. Related graphical tools determine whether the ***two effects are additive or not***, and ***whether the ϵ_{ij} are normally distributed***, as assumed by an ANOVA. If not, a transformation should be employed to achieve additivity and normality before an ANOVA is performed.
₩
• The Friedman and median aligned ranks tests (later sections) are nonparametric alternatives for testing ***whether the median factor effect is significant in the presence of blocking***.

1.1 Median Polish

• Median polish (Hoaglin and others, 1983) is an iterative process that provides a resistant estimate of the overall median, n, as well as estimates α_j of the ***group effects*** and estimates β_i of the ***block effects***.

• The usefulness of median polish lies in its resistance to the effects of outliers. The process begins by subtracting the medians of each block (shown as the rows in Table 1) from the data, leaving the residuals.

• The median of these row medians is then computed as the first estimate of the overall median and subtracted from the row medians. The row medians are now the first estimates of the row effects.

• Then the median of each column is subtracted from the residual data and set aside. The median of the column medians is subtracted from the column medians and added to the previous estimate of the overall median. The column medians now become the first estimates of the column effects.

• The entire process is repeated a second time, producing an estimated overall median, m, row and column departures from the overall median (estimates α_j and β_i), and a table of residuals, ε_{ij}, estimating the ϵ_{ij} .

Table 1. Mercury concentrations, in micrograms per gram, in periphyton (Walpole and Myers, 1985).

Date	Site					
	1	2	3	4	5	6
1	0.45	3.24	1.33	2.04	3.93	5.93
2	0.10	0.10	0.99	4.31	9.92	6.49
3	0.25	0.25	1.65	3.13	7.39	4.43
4	0.09	0.06	0.92	3.66	7.88	6.24
5	0.15	0.16	2.17	3.50	8.82	5.39
6	0.17	0.39	4.30	2.91	5.50	4.29

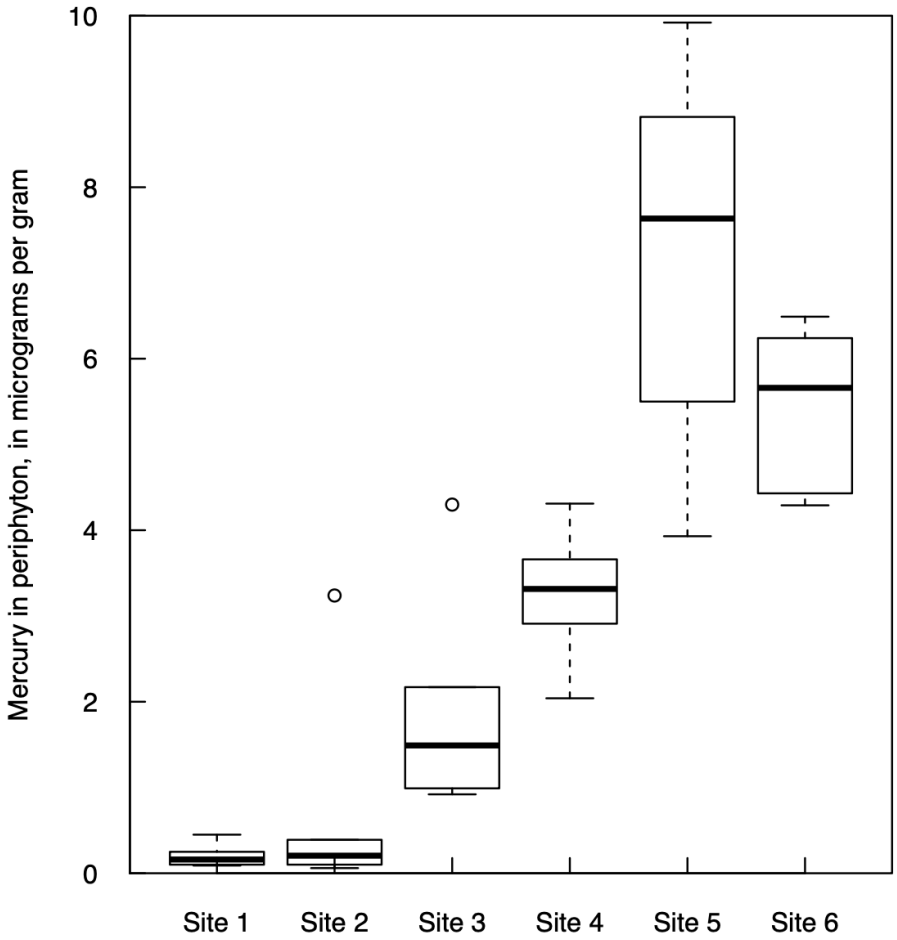


Figure 1. Boxplots showing mercury concentrations in periphyton along the South River, Virginia, from upstream (site 1) to downstream (site 6). Data from Walpole and Myers (1985).

Example 1. Mercury in periphyton—Median polish.

• Mercury concentrations were measured in periphyton at six sites along the South River, Virginia, above and below a large mercury spill (Walpole and Myers, 1985). Measurements were made on ***six different dates***. Of interest is ***whether the six sites differ in mercury concentration***.

• Is this a one-way ANOVA setup? No, because there may be differences among the six dates—the periphyton may not take up mercury ***as quickly during some seasons as others***.

• The dates are not randomly selected within each site but are the same for each site. ***Differences between the six sampling dates are unwanted noise that should be blocked out, hence date is a blocking effect***.

• The data are presented in **Table 1** and boxplots by site in **Figure 1**.

• Median polish will provide an estimate of the magnitude of row (block) and column effects, providing a magnitude of effects for the test results to follow.

• There appears to be a strong increase in mercury concentration going downstream from site 1 to site 6, reflecting an input of mercury along the way.

• The first step in computing median polish is to compute the median of each row (Date) and subtract it from that row's data. The residuals remain in the table (**Table 2**).

• Next, the median of the row medians (2.64) is computed as the first estimate of the overall median, m.

• This is subtracted from each of the row medians in **Table 3**. The median of each column (Site) is then computed and subtracted from that column's data (**Table 4**). The residuals from the subtractions remain in the table. Then the median of the column medians (−0.16) is subtracted from each of the column medians and added to the overall median. The result is shown in **Table 5**.

Table 2. Data from table 1 aligned by subtraction of row medians.

Date	Site						
	1	2	3	4	5	6	Row median (b_i)
1	−2.190	0.600	−1.310	−0.600	1.290	3.290	2.64
2	−2.550	−2.550	−1.660	1.660	7.270	3.840	2.65
3	−2.140	−2.140	−0.740	0.740	5.000	2.040	2.39
4	−2.200	−2.230	−1.370	1.370	5.590	3.950	2.29
5	−2.685	−2.675	−0.665	0.665	5.985	2.555	2.84
6	−3.430	−3.210	0.700	-0.690	1.900	0.690	3.60

Table 3. Data from table 2 after subtraction of the median of row medians. [-, no data]

Date	Site						
	1	2	3	4	5	6	Row median (b_i)
1	−2.19	0.60	−1.31	−0.60	1.29	3.29	0.00
2	−2.55	−2.55	−1.66	1.66	7.27	3.84	0.01
3	−2.14	−2.14	−0.74	0.74	5.00	2.04	−0.25
4	−2.20	−2.23	−1.37	1.37	5.59	3.95	−0.35
5	−2.69	−2.68	−0.67	0.67	5.99	2.56	0.20
6	−3.43	−3.21	0.70	-0.69	1.90	0.69	0.96
						Overall median	m=2.64

Table 4. Data from table 3 after subtractions of column medians from their respective column's data.

Date	Site						
	1	2	3	4	5	6	Row effect (b_i)
1	0.19	2.99	−0.29	−1.31	−4.01	0.37	0.00
2	−0.17	−0.16	−0.64	0.95	1.97	0.92	0.01
3	0.24	0.25	0.28	0.03	−0.30	−0.88	−0.25
4	0.18	0.16	−0.35	0.66	0.29	1.03	−0.35
5	−0.31	−0.29	0.35	−0.04	0.69	−0.36	0.20
6	−1.05	−0.82	1.72	−1.40	−3.40	−2.23	0.96
Column effect	−2.38	−2.39	−1.02	0.71	5.30	2.92	

Table 5. First polish of the periphyton data of Walpole and Myers (1985).

Date	Site						
	1	2	3	4	5	6	Row effect (b_i)
1	0.19	2.99	−0.29	−1.31	-4.01	0.37	0.00
2	−0.17	−0.16	−0.64	0.95	1.97	0.92	0.01
3	0.24	0.25	0.28	0.03	-0.30	−0.88	−0.25
4	0.18	0.16	−0.35	0.66	0.29	1.03	−0.35
5	−0.31	−0.29	0.35	−0.04	0.69	−0.36	0.20
6	−1.05	−0.82	1.72	−1.40	−3.40	−2.23	0.96
Column effect	−2.22	−2.23	−0.86	0.87	5.46	3.08	m=2.48

• The first polish of the data from Walpole and Myers (1985) is shown in Table 5. Two or more polishes are performed in order to produce more stable estimates of the overall median m, as well as row and column effects.

• For a second polish, the above process is repeated on the table of residuals from the first polish (Table 5). Median polish is accomplished in Python by provided Python script.

```
def median_polish(data, n_iter=100, tol=1e-10):
    """Performs median polish
    Args:
        data: pd data frame
        n_iter: maximum number of iterations
        tol: tolerance for convergence
    Returns:
        a dict with:
            'overall': overall median
            'row': row effects
            'column': column effects
            'residuals': residuals
    """
    data = data.copy()
    overall_effect = np.median(data)
    data -= overall_effect
    row_effects = np.zeros(data.shape[0])
    col_effects = np.zeros(data.shape[1])

    for _ in range(n_iter):
        row_medians = np.median(data, axis=1)
        row_effects += row_medians
        data -= row_medians[:, np.newaxis]

        col_medians = np.median(data, axis=0)
        col_effects += col_medians
        data -= col_medians

        new_overall_effect = overall_effect +
np.median(row_medians) + np.median(col_medians)
```

```
        if np.abs(new_overall_effect - overall_effect) < tol:
            break
        overall_effect = new_overall_effect

    residuals = data
    return {'overall': overall_effect, 'row': row_effects,
'column': col_effects, 'residuals': residuals}

# Read the data from the CSV file
data = pd.read_csv('Merc.csv')

# Convert DataFrame to NumPy array
data_array = data.values

# Perform median polish
result = median_polish(data_array)

# Extract the results
overall = result['overall']
row_effects = result['row']
col_effects = result['column']
residuals = result['residuals']

# Display the results
print(f"Overall: {overall}")
print("\nRow Effects:")
print(pd.Series(row_effects, index=data.index))
print("\nColumn Effects:")
print(pd.Series(col_effects, index=data.columns))
print("\nResiduals:")
print(pd.DataFrame(residuals, index=data.index,
columns=data.columns))
```

```
Overall: 2.527760416582459

Row Effects:
0   -0.399271
1   -0.044271
2   -0.505937
3   -0.369271
4   -0.365937
5   -0.653438
dtype: float64

Column Effects:
Site1   -2.384062
Site2   -2.379062
Site3   -1.077396
Site4    0.730937
Site5    5.052604
Site6    3.022604
dtype: float64
```

```
Residuals:
      Site1     Site2     Site3     Site4     Site5     Site6
0  0.213333  2.998333 -0.213333 -1.311667 -3.743333  0.286667
1 -0.491667 -0.496667 -0.908333  0.603333  1.891667  0.491667
2  0.120000  0.115000  0.213333 -0.115000 -0.176667 -1.106667
3 -0.176667 -0.211667 -0.653333  0.278333  0.176667  0.566667
4 -0.120000 -0.115000  0.593333  0.115000  1.113333 -0.286667
5  0.187500  0.402500  3.010833 -0.187500 -1.919167 -1.099167
```

The overall, row, and column effects are those after several polishes were computed. The median polish shows that:

1. The site (column) effects are ***large in comparison*** to the date (row) effects.
2. The site effects show a ***generally increasing pattern*** going downstream (Site 1 to Site 6), with the maximum at Site 5.
3. A large negative residual (−3.74) occurs at Site 5 on Date 1. This is a smaller concentration than expected for this site if the site effect was consistent across all dates.

• A boxplot of residuals, ε_{ij}, (fig. 2) provides a look at the distribution of errors after the factor and block effects have been removed; the figure shows that the residuals from median polish are relatively symmetric.

• This is true after subtracting medians, but may not be true when subtracting ***group means*** using ANOVA. ***Median polish is helpful in deciding whether to use an aligned-ranks test*** (see later section), where symmetry is assumed.

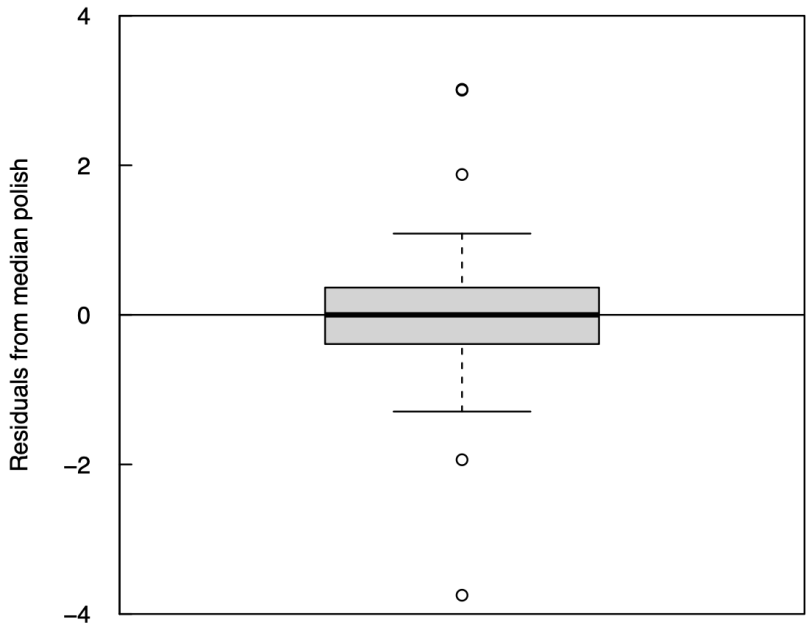


Figure 2. Residuals from the median polish of periphyton mercury data from Walpole and Myers (1985).

1.2 The Friedman Test

• The Friedman test is an extension of the sign test and reduces to the sign test when comparing only two treatment groups.

• Its advantages and disadvantages in comparison to analysis of variance are the same as those of the sign test compared to the *t*-test.

• When the residuals, ε_{ij}, can be considered normal with equal variance in each group, ANOVA will have more power. ***For the many situations where the residuals are not normal, the Friedman test will generally have greater power to detect differences between treatment groups and should be performed***.

• The Friedman test is especially useful for ordinal data—data that can be ranked but differences between observations cannot be computed, such as when comparing a <1 to a 5. The Friedman test is used to determine whether:

- H_0: The median values for all treatment groups are identical.
- H_A: At least one treatment group median is significantly different.

• As with the Kruskal-Wallis test, the test does not provide information on which medians are significantly different from others. That information must come from the associated multiple comparison test presented in the later section.

Example 2. Ordinial Data (Teaching methods) with Friedman Test

• We have three different types of teaching methods, and you want to determine which one is the most effective. You decide to ask a group of students to rank the effectiveness of each method after experiencing all three. The students provide their rankings, but because the data is ordinal, you cannot compute the differences between the rankings—only the order matters.

• Data:

We have 5 students who ranked the teaching methods (Method A, Method B, Method C) from 1 (least effective) to 3 (most effective). Here are the rankings provided by each student:

Student	Method A	Method B	Method C
1	2	3	1
2	1	2	3
3	3	1	2
4	2	1	3
5	3	2	1

• **Applying the Friedman Test:**

The Friedman test will help us determine if there are statistically significant differences in the rankings of the three teaching methods.

1. **Rank the methods** for each student:
   - Student 1: Method A = 2, Method B = 3, Method C = 1
   - Student 2: Method A = 1, Method B = 2, Method C = 3
   - Student 3: Method A = 3, Method B = 1, Method C = 2
   - Student 4: Method A = 2, Method B = 1, Method C = 3
   - Student 5: Method A = 3, Method B = 2, Method C = 1
2. **Perform the Friedman Test**:
   - The test will calculate the sum of the ranks for each method across all students.
   - It will then determine if the differences in the sums of ranks are greater than what would be expected by chance.
3. **Interpret the Results**:
   - If the test statistic is significant (i.e., the *p*-value is less than the chosen significance level, typically 0.05), we can conclude that at least one teaching method is ranked significantly different from the others.

```
# Example 2: Ordinial Data (Teaching methods) with Friedman Test
# Rankings provided by each student
rankings = np.array([
    [2, 3, 1],
    [1, 2, 3],
    [3, 1, 2],
    [2, 1, 3],
    [3, 2, 1]
])

# Perform the Friedman test
stat, p = friedmanchisquare(rankings[:, 0], rankings[:, 1],
rankings[:, 2])

print(f"Friedman test statistic: {stat}")
print(f"p-value: {p}")

if p < 0.05:
    print("There is a significant difference between the teaching
methods.")
else:
    print("There is no significant difference between the teaching
methods.")
Friedman test statistic: 0.4000000000000057
p-value: 0.8187307530779795
There is no significant difference between the teaching methods.
```

1.3 Computation of the Friedman Test

• Understanding the test statistic provides insight into how the Friedman test works. Data are ranked only within each block, not by making any cross-rankings between blocks. With *k* treatment groups (columns), rank the data within each of the *n* blocks (rows) from 1 to *k*, from smallest to largest.

• If the null hypothesis is true, the ranks within each row will vary randomly with no consistent pattern.

• Second, sum the ranks for each group (column). When the null hypothesis is true, the average rank for each group will be close to the overall average rank of $\frac{k+1}{2}$. When the alternative hypothesis is true, the average group rank will differ from one another and from the overall average rank.

• Third, compute the test statistic *Xf*, which squares the differences between the average group rank, R_j, and the overall rank to determine if the *k* groups differ in magnitude.

$$Xf = \frac{12n}{k(k+1)} \sum_{j=1}^{k} \left[\bar{R}_j - \frac{k+1}{2} \right]^2 \qquad \text{Eq. (1)}$$

• Iman and Davenport (1980) state that the exact test should be used for all cases where the number of treatment groups plus the number of blocks (k+n) is ≤ 9.

• For larger sample sizes, a large-sample approximation is sufficient. When observations are tied within a block, assign the average of their ranks to each. *Xf* must be corrected using equation 2 when ties within a block occur.

$$Xf = \frac{12n}{k(k+1) - \frac{1}{n(k-1)} \sum_{i=1}^{n} \sum_{j=1}^{k} \left(t_{ij}(j^3 - j) \right)} \sum_{j=1}^{k} \left[\bar{R}_j - \frac{k+1}{2} \right]^2 \qquad \text{Eq. (2)}$$

where t_{ij} equals the number of ties of extent j in row i. When ties occur, the large-sample approximation using a chi-squared distribution with k−1 degrees of freedom must be used.

Example 3. Mercury in periphyton—Friedman test

• Does the median mercury concentration in periphyton differ for the six sites along the South River of Virginia (fig. 1 and table 1)? There are sufficient columns and rows to employ the large-sample approximation, and because ties are present, the approximation is required.

```
data = pd.read_csv('Merc.csv')

# Extract the columns for the test
Hg = data.values

# Perform the Friedman test
stat, p_value = friedmanchisquare(Hg[:, 0], Hg[:, 1], Hg[:, 2], Hg[:,
3], Hg[:, 4], Hg[:, 5])
```

```
print(f"Friedman chi-squared = {stat:.3f}, df = {Hg.shape[1] - 1}, p-value = {p_value:.7f}")
```

The median mercury concentration differs significantly between the six sites.

1.4 Multiple Comparisons for the Friedman Test

• The decision of which groups' data differ from others can be determined using a multiple comparison test.

• The MCT associated with Friedman's test (Hollander and Wolfe, 1999) controls the family error rate using Bonferroni's adjustment, so it will have less power than previous MCTs using the BH adjustment.

• The test uses the difference in the mean group rank, rejecting the null hypothesis (H_0: No difference in mean rank) when differences are larger than expected. As with Dunn's MCT, Friedman ranks are joint ranks, so values for data in the (k−2) groups not being compared do affect the computation of each test.

• For this situation where there is only one observation per cell, an MCT using separate rankings for each pairwise comparison, such as a series of sign or signed-rank tests, would have little power unless there were many rows (blocks).

• For the mercury data in Table 1, a sign test between the first and fifth columns would have only six pairs of observations to use. Even though the fifth column has concentrations higher than the first column for all six pairs, the resulting p-value will not be below 0.05 as a result ***only of the small sample size***.

• For a two-factor analysis without replication, the joint Friedman ranks provide more information to determine group differences than would separate pairwise rankings.

Example 4. Mercury in periphyton—Pairwise Friedman comparison test

• The provided Python code scales the reported p-values to compare to the family error rate, whose default *αfamily* is 0.05.

```
def pairwise_friedman(data, alpha=0.05):
    k = data.shape[1]
    n = data.shape[0]
    comparisons = list(combinations(range(k), 2))
    ranks = data.rank(axis=1)
    mean_ranks = ranks.mean(axis=0)

    p_values = {}
    for i, j in comparisons:
        diff = np.abs(mean_ranks[i] - mean_ranks[j])
        se = np.sqrt(k * (k + 1) / (6 * n))
        z = diff / se
        p = 2 * (1 - norm.cdf(np.abs(z)))  # Two-tailed test
```

```
        p_values[(i, j)] = p

    # Bonferroni correction
    bonferroni_alpha = alpha / len(comparisons)
    reject = {comp: p < bonferroni_alpha for comp, p in
p_values.items()}

    return p_values, reject

from scipy.stats import norm

# Read the data from the CSV file
data = pd.read_csv('Merc.csv')

# Perform the Friedman test
stat, p_value = friedmanchisquare(*[data[col] for col in
data.columns])

print(f"Friedman chi-squared = {stat:.3f}, df = {data.shape[1] - 1},
p-value = {p_value:.7f}")



# Perform pairwise comparisons with Bonferroni correction
p_values, reject = pairwise_friedman(data)

# Display results
print("\nPairwise comparisons:")
for (i, j), p in p_values.items():
    print(f"Comparison {data.columns[i]} vs {data.columns[j]}: p-
value = {p:.7f}, reject H0 = {reject[(i, j)]}")
```

```
Friedman chi-squared = 25.577, df = 5, p-value = 0.0001078

Pairwise comparisons:
Comparison Site1 vs Site2: p-value = 0.5370940, reject H0 = False
Comparison Site1 vs Site3: p-value = 0.0896330, reject H0 = False
Comparison Site1 vs Site4: p-value = 0.0307536, reject H0 = False
Comparison Site1 vs Site5: p-value = 0.0000310, reject H0 = True
Comparison Site1 vs Site6: p-value = 0.0006871, reject H0 = True
Comparison Site2 vs Site3: p-value = 0.2800872, reject H0 = False
Comparison Site2 vs Site4: p-value = 0.1228226, reject H0 = False
Comparison Site2 vs Site5: p-value = 0.0003867, reject H0 = True
Comparison Site2 vs Site6: p-value = 0.0054786, reject H0 = False
Comparison Site3 vs Site4: p-value = 0.6434288, reject H0 = False
Comparison Site3 vs Site5: p-value = 0.0135547, reject H0 = False
Comparison Site3 vs Site6: p-value = 0.0896330, reject H0 = False
Comparison Site4 vs Site5: p-value = 0.0448623, reject H0 = False
Comparison Site4 vs Site6: p-value = 0.2170439, reject H0 = False
Comparison Site5 vs Site6: p-value = 0.4404007, reject H0 = False
```

For the periphyton mercury data from table 1, **Site 5** differs from Sites **1** and **2**, and **Site 6** differs from Site 1.

1.5 Aligned-ranks Test

• The Friedman test is the multi-treatment equivalent of the sign test. In week09, the signed-rank test was presented in addition to the sign test and was favored over the sign test when the differences between the two treatment groups ***were symmetric***.

• An extension to the signed-rank test for three or more treatment groups is the Aligned-Ranks Test (ART), one of several possible extensions—Quade's test (Conover, 1999) and Doksum's test (Hollander and Wolfe, 1999) are others.

• Groggel (1987) and Fawcett and Salter (1984) have shown that an aligned-rank method ***has substantial advantages in power over other signed rank extensions***. For more information on ART methods, see the textbook by Higgins (2003), as well as papers by Mansouri and others (2004), Richter and Payton (2005), and Wobbrock and others (2011).

• Discussion of the MCT for the two-way design without replication is found in Barefield and Mansouri (2001).

• Friedman's test computes ***within-block ranks***, avoiding the confusion produced by block-to-block differences. ART instead allows comparisons across blocks by first subtracting the within-block mean from all of the data within that block. This adds additional information and degrees of freedom to tests, increasing the power over the Friedman within-block only approach. Subtracting the block mean aligns the data across blocks to a common center. To compute aligned ranks, first subtract the ith block mean,

$$O_{ij} = \left(y_{ij} - \beta_i\right) \qquad \text{Eq. (3)}$$

where j = 1,2,...,k is the number of groups. Then the O_{ij} are jointly ranked from 1 to N, where $N = n \cdot k$ is the number of observations, forming aligned ranks, AR_{ij}.

• A one-way ANOVA or BDM test is then performed on the , AR_{ij}. Aligned ranks are equivalent to the ranking of magnitudes of row-to-row differences in the signed ranks test. To derive the benefits of these cross-block comparisons, a cost is incurred. The cost is an assumption that the residuals, ϵ_{ij}, from the ANOVA are symmetric.

• Symmetry can be evaluated by estimating the ***residuals using median polish***, or by computing them in the ANOVA process and plotting them on a boxplot, as in figure 1. The null and alternate hypotheses are identical to those of the Friedman test:

H_0: The median values for all groups are identical.
H_A: At least one group median is significantly different.

• Though a one-way ANOVA on the aligned ranks is computed, the correct F-test will differ from the one determined for the group effect by ANOVA software. Instead, the error degrees of freedom must be $(k-1)\cdot(n-1)$ because of blocking, not $k\cdot(n-1)$ as for a one-way ANOVA.

• Richter and Payton (2005) have shown that performing a BDM test on aligned ranks has greater power than using ANOVA on either the original data or ranks.

1.6 Multiple Comparisons for the Aligned-ranks Test

• Group multiple comparisons following ART take advantage of the data alignment by not requiring paired tests to be performed. A sequence of paired *t*-tests or paired Wilcoxon tests would be less powerful because there are few blocks (pairs) with which to conduct those tests. By using aligned ranks, the block effect is factored out and the relations across blocks can be utilized, increasing degrees of freedom and the power of the test.

• Tukey's tests on aligned ranks are the natural follow-up to ANOVA on aligned ranks to determine which group levels differ from others.

1.7 Two-factor ANOVA Without Replication

• The parametric alternative to a Friedman's test for a complete block design is a two-factor ANOVA with only one observation per factor-block combination. The first factor is the effect of interest, and the second factor is the block effect.

• The block effect is of no interest except to remove its masking of the factor effect, so no test for its presence is required. Because there is only one observation per cell, it is impossible to test for an interaction.

• The hypotheses are similar to those of the Friedman and ART tests, except that treatment group means, rather than medians, are being tested.

H_0: The treatment group means are identical, $\mu_1 = \mu_2 = \cdots = \mu_k$
H_A: At least one mean is significantly different.

The ANOVA model without replication is

$$y_{ij} = \mu + \gamma_j + \delta_i + \epsilon_{ij} \qquad \text{Eq. (4)}$$

where

y_{ij}	is the individual observation in block *i* and group *j*;
μ	is the overall mean or median (over all groups)
γ_j	is the *j*th group effect, *j* =1, 2, ..., *k*
δ_i	is the *i*th block effect, *i* =1, 2, ..., *n*
ϵ_{ij}	is the residual difference between the individual observation and the combined group and block effects.

It is assumed that the residuals, ϵ_{ij}, follow a normal distribution. ANOVA does not provide information on which means differ from others; that must come from a multiple comparison test.

1.8 Computation of Two-factor ANOVA Without Replication

• Sums of squares for factor, block, and error are computed using the following formulae (table 6). These are divided by their appropriate degrees of freedom to form mean squares. The factor F-test is the MSF divided by the MSE, to be compared to quantiles of the *F*-distribution for evaluation of its significance.

• We aren't interested in the block effect, so its test can be ignored. The general structure for a two-factor ANOVA table without replication is found in table 7. Reject the null hypothesis for the factor effect when the *F*-test with statistic MSF/MSE has a *p*-value less than the desired αα.

Table 6. Sums of squares definitions for two-factor ANOVA.
[SS, Sum of squares; SSF, SS for factor; SSB, SS for block; SSE, SS for error]

Sums of squares formula	Effect
$SSF = \frac{\sum^{k}\left[\sum^{n} y\right]^2}{n} - \frac{\left[\sum^{k}\sum^{n} y\right]^2}{kn}$	$\mu_j - \mu$
$SSB = \frac{\sum^{n}\left[\sum^{k} y\right]^2}{k} - \frac{\left[\sum^{k}\sum^{n} y\right]^2}{kn}$	$\mu_i - \mu$
$SSE = Total\ SS - SST - SSB$	$y_{ij} - \mu_i - \mu_j + \mu$
$Total\ SS = \sum^{k}\sum^{n} y^2 - \frac{\left[\sum^{k}\sum^{n} y\right]^2}{kn}$	$y_{ij} - \mu$

Table 7. Analysis of variance (ANOVA) table for two factors without replication. [df, degrees of freedom; SS, sums of squares; SSF, SS for factor; SSB, sum of squares for block; SSE, sum of squares for error; MS, mean square; MSF, mean square for factor; MSE, mean square for error; *F*, *F*-test statistic;-, not applicable]

Source	df	SS	MS	*F*	*p*-value
Factor/treatment	$k-1$	SSF	SSF/$(k-1)$	MSF/MSE	-
Block	$n-1$	SSB	SSB/$(n-1)$	-	-
Error	$(k-1)\cdot(n-1)$	SSE	SSE/$[(k-1)\cdot(n-1)]$	-	-

1.9 Parametric Multiple Comparisons for ANOVA Without Replication

Pairwise paired t-tests will take the blocking structure into account (the blocks form matched pairs) while comparing all pairs of group means. Because no alignment was performed, Tukey's test is not appropriate, as it doesn't take the blocking structure into account. The BH adjustment is used to minimize the false positive error rate.

2 Group Tests for Data with Nondetects

• The methods quickly described in this section extend those given for two groups in week08 and are described in far more detail in the textbook by Helsel (2012).

• The most convenient and powerful procedure is to recensor data so that all observations below the highest detection limit (HDL) are noted as <HDL. No alterations are needed if only one detection limit is present; then any of the nonparametric methods of this chapter can be computed with little loss of information.

• This method has far more power to detect differences than would substitution followed by a parametric test, as observations small enough to be below detection are certainly nearing the lower bound of zero, resulting in an overall skewed distributional shape. The ranks of all <HDLs will be tied, so software must include tie corrections (as does Python script) to obtain accurate *p*-values.

• For example, table 8 presents the mercury concentrations where concentrations below 0.20 have been censored as <0.20. These could have come from data measured with detection limits of 0.10, 0.15, and 0.20.

Table 8. Mercury concentrations, in micrograms per liter, in periphyton (Walpole and Myers, 1985), altered to have a detection limit of 0.20.

Date	Site					
	1	2	3	4	5	6
1	0.45	3.24	1.33	2.04	3.93	5.93
2	<0.20	<0.20	0.99	4.31	9.92	6.49
3	0.25	0.25	1.65	3.13	7.39	4.43
4	<0.20	<0.20	0.92	3.66	7.88	6.24
5	<0.20	<0.20	2.17	3.50	8.82	5.39
6	<0.20	0.39	4.30	2.91	5.50	4.29