

Parametric bootstrap procedures for three-factor ANOVA and multiple comparison procedures with unequal group variances

Sarah Alver & Guoyi Zhang

To cite this article: Sarah Alver & Guoyi Zhang (2022): Parametric bootstrap procedures for three-factor ANOVA and multiple comparison procedures with unequal group variances, Communications in Statistics - Simulation and Computation, DOI: [10.1080/03610918.2022.2104316](https://doi.org/10.1080/03610918.2022.2104316)

To link to this article: <https://doi.org/10.1080/03610918.2022.2104316>



Published online: 03 Aug 2022.



Submit your article to this journal [↗](#)



Article views: 132



View related articles [↗](#)



View Crossmark data [↗](#)



Parametric bootstrap procedures for three-factor ANOVA and multiple comparison procedures with unequal group variances

Sarah Alver and Guoyi Zhang

Department of Mathematics and Statistics, University of New Mexico, Albuquerque, New Mexico, USA

ABSTRACT

The issue of unmet equal variance assumption in multi-factor ANOVA has been addressed in the literature with several methods, and parametric bootstrap (PB) has been found in the one-way and two-way cases to outperform other methods. We extend previously developed PB procedures for one- and two-way ANOVA, and illustrate with a three-way ANOVA model with unequal group variances (heteANOVA model). We develop a framework for working with these models, analogous to usual multi-factor ANOVA procedures, where F-tests and Tukey's simultaneous multiple comparison procedures are replaced by PB procedures. Using simulation, we compare these methods to F-tests for each step in model selection, as well as to Tukey's test for multiple comparison procedures (MCP). The results of our simulations indicate that the PB methods outperform F-tests and Tukey's test in terms of Type I error when data are unbalanced.

ARTICLE HISTORY

Received 2 August 2021
Accepted 14 July 2022

KEYWORDS

ANOVA; HeteANOVA;
Multiple comparison;
Parametric bootstrap;
Simulations; Tukey's test;
Unequal variance

1. Introduction

Consider the three-factor ANOVA problem of abc normal populations with unequal population variances σ_{ijk}^2 , $i = 1, \dots, a$, $j = 1, \dots, b$, $k = 1, \dots, c$, and let $Y_{ijk1}, Y_{ijk2}, \dots, Y_{ijkn_i}$ be the observations from each group. The full ANOVA model, hereafter called heteANOVA model, is

$$Y_{ijkm} = G + A_i + B_j + C_k + AB_{ij} + AC_{ik} + BC_{jk} + ABC_{ijk} + e_{ijkm}$$

where $e_{ijkm} \sim N(0, \sigma_{ijk}^2)$. The usual F-tests for main and interaction effects in these models assume equal group variances, and can be smaller or larger in size than the nominal level when this assumption is violated (Weerahandi 1995; Bao and Ananda 2001; Scheffe 1959). Weerahandi (Weerahandi 1995) showed examples of this: when there was no particular relationship between sample size and group variance, the p-value for the conventional F-test was too large (type II error); however, they provide an additional example where sample sizes were negatively correlated with the group variances, and the p-value of the conventional F-test was shown to be too small in this case. Transformed data, such as the log or square root of observed values, may in some cases meet the equal variance assumption. However, this method does not always work and can make the results more difficult to interpret. Other approaches such as the generalized F-test have been proposed (Weerahandi 1995; Ananda and Weerahandi 1997) for one-way and two-way models, but may not perform well with larger numbers of treatment levels (Xu et al. 2013).

As described in (Christensen 2016) section 4.3, caution is needed when making practical decisions based on differences in means between groups with unequal variances. For example, if a

lower value of a response is desired, such as blood pressure, a treatment group with a smaller mean and smaller variance may have a smaller probability of achieving the desired outcome than a treatment group with a larger mean and also larger variance. Thus, additional consideration of implications for the practical issue being studied is warranted. Nevertheless, the problem of unequal variance does arise in practice, so methods of dealing with the problem are desirable.

The parametric bootstrap (PB) approach has been shown to work well for one-way and two-way heteANOVA models, including cases with unbalanced data (Xu et al. 2013; Krishnamoorthy, Lu, and Mathew 2007; Zhang 2015a; Zhang 2015b). This work generalizes the approach to a three-factor model and uses simulations to compare the performance of the PB method with the usual F-tests.

Another problem in ANOVA models is multiple comparison procedures (MCP's): pairwise simultaneous comparisons of all factor levels. The PB approach has been shown to work well for MCPs in one-way and two-way heteANOVA cases (Zhang 2015a; Zhang 2015b). We again generalize this to the three-factor case, and use simulations to compare the performance of the PB methods to Tukey's test.

This paper is organized as follows. In [Sec. 2](#), we describe the overall PB method and show relationships between PB methods and conventional F-tests, as well as develop an overall procedure for analyzing data under these models, analogous to conventional methods. In [Sec. 3](#), we illustrate the procedure for a three-way ANOVA model and compare performance of the PB tests with that of the usual F-tests for each term in the model. [Section 4](#) illustrates MCP using PB. [Section 5](#) includes discussion of our results, limitations and areas for further research. R code (R Core Team 2020) is given in the [Appendix](#) for the algorithms developed.

2. General PB method for ANOVA models

The overall process for analyzing multi-factor data using PB methods is similar to the usual ANOVA approach, such as in Christensen 2016 (Christensen 2016) and Kutner et al. 2005 (Kutner et al. 2005) and is shown in [Figure 1](#). For the PB method, a PB test rather than an F-test is used at each step of testing to determine the terms to be included in the final model, and PB tests rather than traditional MCP's are used to examine factor level means. In usual ANOVA models where the equal variance assumption is met, for testing $H_0 : Par = 0$, where the parameter of interest (Par) is a main effects term or an interaction term, our usual F-test statistic, or general linear test (Kutner et al. 2005), takes the form

$$\frac{(SSE(R) - SSE(F))/(dfE(R) - dfE(F))}{MSE(F)},$$

where $SSE(R)$ indicates the sum of square for error (SSE) from the reduced model, $SSE(F)$ indicates SSE from the full model, dfE indicates the degrees of freedom for error for the respective models, and $MSE(F)$ indicates the mean squared error (MSE) from the full model.

In the following sections, we develop PB algorithms for use at each level of testing that are analogous to the F-test (general linear test for a three-way ANOVA model). Algorithm 1 will be used for testing the three-way interaction term, Algorithm 2 for the two-way interaction terms, and Algorithm 3 for testing main effects when no interaction terms have significant effects. Algorithm 4 is used when only one two-way interaction term is significant and we want to test the remaining main effect term that is not involved in the significant interaction. Algorithms 5 and 6 will pertain to MCP's. Algorithms 1-4 are the same at each step other than the design matrix specific to the reduced model being tested. For each of algorithms 1-4, the test statistic is based on the standardized sum of squares for the term under investigation, that is, a function of the numerator of the F-test shown above. As discussed in (Christensen 2018), this can be written in matrix form as:

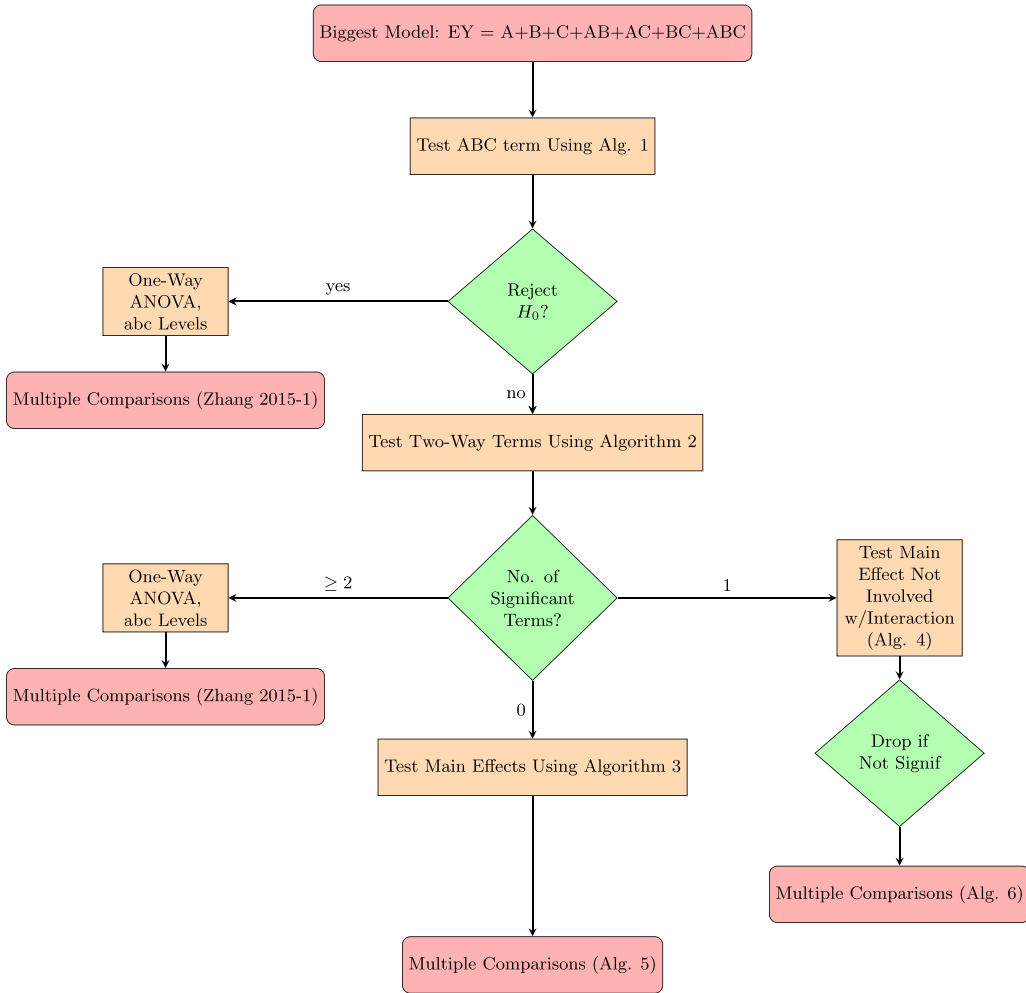


Figure 1. Overall process: three-way ANOVA using parametric bootstrap.

$$S_I = Y'(A - A_0)' \Sigma_*^{-1} (A - A_0) Y = SSE(R) - SSE(F)$$

For our application, $\Sigma_* = \text{diag}(\sigma_{111}^2, \dots, \sigma_{111}^2, \sigma_{112}^2, \dots, \sigma_{abc}^2)$, (i.e., each σ_{ijk}^2 is repeated n_{ijk} times along the diagonal). For equal variances, Σ_* reduces to $\sigma^2 I$ where I is an $n \times n$ identity matrix. In S_I above, $A = X(X'\Sigma_*^{-1}X)^{-1}X'\Sigma_*^{-1}$ and $A_0 = X_0(X_0'\Sigma_*^{-1}X_0)^{-1}X_0'\Sigma_*^{-1}$ are the projection operators onto the column spaces of the design matrices X and X_0 for the full and reduced models respectively, where X' indicates the transpose of a matrix X . If variances are known, $S_I \sim \chi^2(r(X) - r(X_0))$, as shown in linear models texts such as (Christensen 2018). We discuss this idea more specifically to each parameter of interest in the following sections. In general, variances are unknown, so the true distribution of this test statistic is also unknown. When group variances are equal (but unknown), the usual F-test statistic follows an F distribution since it is equivalent to:

$$\frac{\frac{Y'(A-A_0)Y}{\sigma^2} / [r(X) - r(X_0)]}{\frac{[N-r(X)]MSE(F)}{\sigma^2} / [N - r(X)]} = \frac{\chi_{r(X)-r(X_0)}^2 / [r(X) - r(X_0)]}{\chi_{N-r(X)}^2 / [N - r(X)]} \sim F_{[r(X)-r(X_0), N-r(X)]},$$

where $r(X)$ refers to the rank of the X matrix and N is the total number of observations for all groups. When variances are equal so that $\Sigma_* = \sigma^2 I$, A and A_0 reduce to $X(X'X)^{-1}X'$ and

$X_0(X_0'X_0)^{-1}X_0'$, respectively, so in this case, $Y'(A - A_0)' \Sigma_*^{-1}(A - A_0)Y = \frac{Y'(A - A_0)Y}{\sigma^2}$. In the above F-statistic equation, the σ^2 cancel since they are equal, so unknown σ^2 is not a problem. However, as we see in our simulation results, the pooled variance estimate that we use for the MSE will not be accurate for all groups and can lead to test statistics being too large or too small, and thus decisions to reject or not reject hypotheses can be too liberal or too conservative, similar to results illustrated by Weerahandi (Weerahandi 1995).

The X and X_0 matrices above are the design matrices corresponding to a Y vector with all responses. The PB method here uses design matrices corresponding to the vector of group means, e.g., for a three-way ANOVA model with $a=3$, $b=2$ and $c=2$, $\bar{Y} = (\bar{y}_{111}, \bar{y}_{112}, \bar{y}_{121}, \dots, \bar{y}_{322})$, where $\bar{y}_{ijk} = \sum_{m=1}^{n_{ijk}} y_{ijkm} / n_{ijk}$. It can be shown that $S_I = Y'(A - A_0)' \Sigma_*^{-1}(A - A_0)Y = SSE(R) - SSE(F) = \bar{Y}' \Sigma^{-1} \bar{Y} - \bar{Y}' \Sigma^{-1} X_* (X_*' \Sigma^{-1} X_*)^{-1} X_*' \Sigma^{-1} \bar{Y}$, where $\Sigma = \text{diag}(\sigma_{111}^2/n_{111}, \sigma_{112}^2/n_{112}, \dots, \sigma_{abc}^2/n_{abc})$ and X_* is a matrix of indicators corresponding to each group mean, discussed further for each parameter in the upcoming sections.

For a three-factor ANOVA model, if σ_{ijk}^2 's are known, $\Sigma = \text{diag}(\sigma_{111}^2/n_{111}, \sigma_{112}^2/n_{112}, \dots, \sigma_{abc}^2/n_{abc})$, and the null hypothesis $H_0 : Par = 0$ is true (under the null hypothesis, the χ^2 non-centrality parameter is 0), then a natural test statistic for testing H_0 is S_I , the standardized sum of squares for the term being tested, which as discussed above, follows a χ^2 distribution with $r(X) - r(X_0)$ degrees of freedom. In general, variances are unknown, so we replace S_I with the test statistic $\tilde{S}_I = \bar{Y}' S^{-1} \bar{Y} - \bar{Y}' S^{-1} X_* (X_*' S^{-1} X_*)^{-1} X_*' S^{-1} \bar{Y}$, where $S = \text{diag}(s_{111}^2/n_{111}, s_{112}^2/n_{112}, \dots, s_{abc}^2/n_{abc})$, and $s_{ijk}^2 = \frac{1}{n_{ijk}-1} \sum_{m=1}^{n_{ijk}} (y_{ijkm} - \bar{y}_{ijk})^2$.

In this case, since the variances are unequal and unknown, the test statistic no longer follows a known distribution. The overall idea of a PB approach to this problem is to simulate a distribution for \tilde{S}_I under the null hypothesis.

Each of Algorithms 1-4 follows the same procedure for testing each null hypothesis $H_0 : Par = 0$, with Par the applicable parameter. This procedure involves (1) calculate the test statistic \tilde{S}_I above, (2) simulate a distribution for \tilde{S}_I under H_0 , and (3) calculate a Monte Carlo estimate of a p-value: the proportion of our simulated null distribution that is at least as extreme as our test statistic. This p-value can be used in the typical manner to reject or not reject the null hypothesis pertaining to the model term (parameter) we are investigating. In each algorithm 1-4, the X_* matrix in \tilde{S}_I changes to reflect each reduced model; otherwise these algorithms are the same at each step.

For multiple comparisons of levels of a factor, Algorithms 5 and 6 are analogous to Tukey's test, but again, Tukey's test is intended for cases where the equal variance assumption is met and group sizes are equal. The Tukey-Kramer procedure does allow for different sample sizes (Kutner et al. 2005), and the documentation for the 'TukeyHSD' procedure in R (R Core Team 2020) states that the results are valid for mildly unbalanced data. When the equal variance assumption is met, Tukey's test statistic can be compared to the studentized range distribution, but if not, the test statistic no longer follows a standard distribution for comparison, so we simulate a distribution using the PB method. Figure 1 depicts the overall procedure for a three-factor hetaANOVA problem using these PB algorithms.

3. Illustration of PB for three-factor ANOVA

Consider the three factor ANOVA full model, with all interactions and main effects:

$$y_{ijkm} = G + A_i + B_j + C_k + [AB]_{ij} + [AC]_{ik} + [BC]_{jk} + [ABC]_{ijk} + e_{ijkm}, \quad (1)$$

where G indicates the grand mean, A , B , and C indicate main effects, $[AB]$, $[AC]$, and $[BC]$ indicate two-way interaction terms, and $[ABC]$ indicates the three-way interaction term. Also, we

assume e_{ijkm} independent $\sim N(0, \sigma_{ijk}^2)$, and for identifiability, we assume the constraints $\sum_i w_i A_i = 0$, $\sum_j v_j B_j = 0$, $\sum_k u_k C_k = 0, \dots$, $\sum_i w_i [ABC]_{ijk} = 0$, $\sum_j v_j [ABC]_{ijk} = 0$, $\sum_k u_k [ABC]_{ijk} = 0$, where the w 's, v 's, and u 's are non-negative weights, not all zero. Define the vector of means, $\bar{Y} = (\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{121}, \bar{y}_{122}, \dots, \bar{y}_{abc})'$, indicating the sample means of the observations from each factor level and combination of factor levels. Define the vector of sample variances for each combination of factor levels to be $s_{ijk}^2 = (s_{111}^2, s_{112}^2, \dots, s_{abc}^2)'$, and the matrix

$$S_{abc \times abc} = \text{diag}(s_{111}^2/n_{111}, s_{112}^2/n_{112}, \dots, s_{abc}^2/n_{abc})$$

Following the procedure in Figure 1, we test each term in the model (1), from highest order to lowest order. Reduced models for each stage of testing are shown below. For each term in the model, if σ_{ijk}^2 's are known, $\Sigma = \text{diag}(\sigma_{111}^2/n_{111}, \sigma_{112}^2/n_{112}, \dots, \sigma_{abc}^2/n_{abc})$, and the null hypothesis is true (under the null hypothesis, the χ^2 non-centrality parameter is 0), then a natural test statistic for testing H_0 is the standardized sum of squares for the term being tested (and higher order terms):

$\bar{Y}'\Sigma^{-1}\bar{Y} - \bar{Y}'\Sigma^{-1}X_*(X_*'\Sigma^{-1}X_*)^{-1}X_*'\Sigma^{-1}\bar{Y} \sim \chi_{abc-r(X_*)}^2$, where X_* refers to e.g., X_{ABC} for the three-way interaction term, X_{BC} for the BC interaction term, and X_C for the main effects for factor C as described below.

The matrix X_* consists of a column of 1's for the grand mean and (0, 1) indicators for membership in each factor level and combination of factor levels. Note that this matrix is indicating the levels for the group means, not each observation, so it should not be confused with the design matrix for the full data, which could include replications. X_* can be expressed using Kronecker products. Let J_n indicate a column vector of n 1's, and I_n indicate an $n \times n$ identity matrix. Then, for example,

$$X_{ABC} = ([J_{abc}, I_a \otimes J_b, J_a \otimes (I_b \otimes J_c), J_a \otimes (J_b \otimes I_c)], [J_{ab} \otimes J_c], [I_a \otimes (J_b \otimes I_c)], [J_a \otimes I_{bc}])$$

In general, variances are unknown, so we replace Σ with S to form the test statistic introduced earlier: $\tilde{S}_I = \bar{Y}'S^{-1}\bar{Y} - \bar{Y}'S^{-1}X_*(X_*'S^{-1}X_*)^{-1}X_*'S^{-1}\bar{Y}$.

The test statistic \tilde{S}_I is location invariant (Xu et al. 2013), so without loss of generality, take $E(Y) = \mathbf{0}$. The PB variable can then be developed as follows. For a given $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc}; s_{111}^2, s_{112}^2, \dots, s_{abc}^2)$, $\bar{y}_{Bijk} \sim N(0, s_{ijk}^2/n_{ijk})$, and $S_{Bijk}^2 \sim \left(\frac{s_{ijk}^2}{n_{ijk}-1}\right)\chi_{n_{ijk}-1}^2$, $i = 1, \dots, a$, $j = 1, \dots, b$, $k = 1, \dots, c$.

Let $\bar{Y}_B = (\bar{y}_{B111}, \bar{y}_{B112}, \dots, \bar{y}_{Babc})'$ and $S_B = \text{diag}(s_{B111}^2/n_{111}, s_{B112}^2/n_{112}, \dots, s_{Babc}^2/n_{abc})$.

Then we can construct the PB pivot variable based on the test statistic \tilde{S}_I , replacing \bar{Y} with \bar{Y}_B and S with S_B :

$\tilde{S}_{IB} = \bar{Y}'_B S_B^{-1} \bar{Y}_B - \bar{Y}'_B S_B^{-1} X_* (X_*' S_B^{-1} X_*)^{-1} X_*' S_B^{-1} \bar{Y}_B$. For a given level α , there is evidence that the main effects or interaction effects exist when $P(\tilde{S}_{IB} > \tilde{s}_I) < \alpha$, where \tilde{s}_I is an observed value of \tilde{S}_I . This probability can be estimated by Algorithms 1-4 depending on the term being tested. Note that while model parameter estimates depend on the chosen weights, the tests considered here do not. This is discussed for the two-way case in (Xu et al. 2013) and proofs are given by Arnold (Steven F, 1981). These ideas can be extended to the three-way case. As discussed by Arnold (Steven F, 1981), when testing the main effects and two-way interactions, we are performing not quite a test for e.g., $A_i = 0$, but a test for $A_i + [AB]_{ij} + [AC]_{ik} + [ABC]_{ijk} = 0$, that is, the main effect plus the higher order terms involving it, which does not involve the weights. Additionally, we do not suggest testing a main effect term if the interaction terms involving it are found significantly different from zero. Further, the tests presented here are based on the differences in sums of squares for error between models, not the parameter estimates themselves (see e.g. (Searle 1971), Sec. 5.2). The specific tests of a main or two-way term plus the higher order terms that involve it are shown in the null and alternative hypotheses for each Algorithm 1-4.

3.1. Testing three-way interaction

For the three-way interaction term, consider model (1) and:

$$H_{0ABC} : [ABC]_{ijk} = 0 \text{ for } i = 1, \dots, a, j = 1, \dots, b, k = 1, \dots, c \text{ vs}$$

$$H_{\alpha ABC} : [ABC]_{ijk} \neq 0 \text{ for some } i, j, k.$$

If σ_{ijk}^2 's are known, as discussed previously, a natural test statistic for testing H_0 is the standardized sum of squares for the three way interaction, a function of $(\bar{Y} - \hat{G} - \hat{A} - \hat{B} - \hat{C} - \widehat{AB} - \widehat{AC} - \widehat{BC})$, where the terms $\hat{G}, \dots, \widehat{BC}$ are the parameter estimates from fitting all terms from model (1) other than the ABC term:

$$\bar{Y}'\Sigma^{-1}\bar{Y} - \bar{Y}'\Sigma^{-1}X_{ABC}(X'_{ABC}\Sigma^{-1}X_{ABC})^{-1}X'_{ABC}\Sigma^{-1}\bar{Y} \sim \chi_{abc-r(X_{abc})}^2 \quad (1.1)$$

In general, variances are unknown, so we replace (1.1) with the following test statistic:

$$\tilde{S}_I = \bar{Y}'S^{-1}\bar{Y} - \bar{Y}'S^{-1}X_{ABC}(X'_{ABC}S^{-1}X_{ABC})^{-1}X'_{ABC}S^{-1}\bar{Y}. \quad (1.2)$$

This test statistic is location invariant (Xu et al. 2013), so without loss of generality, take $E(\bar{Y}) = \mathbf{0}$. We construct the PB pivot variable based on test statistic (1.2), replacing \bar{Y} with \bar{Y}_B and S with S_B :

$$\tilde{S}_{IB} = \bar{Y}'_B S_B^{-1} \bar{Y}_B - \bar{Y}'_B S_B^{-1} X_{ABC} (X'_{ABC} S_B^{-1} X_{ABC})^{-1} X'_{ABC} S_B^{-1} \bar{Y}_B \quad (1.3)$$

For a given level α , the test rejects H_{0ABC} when $P(\tilde{S}_{IB} > \tilde{s}_I) < \alpha$, where \tilde{s}_I is an observed value of \tilde{S}_I in (1.2). This probability can be estimated by Algorithm 1.

Algorithm 1:

For a given $(n_{111}, n_{112}, \dots, n_{abc})$, $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc})$, and $(s_{111}^2, s_{112}^2, \dots, s_{abc}^2)$, compute $\tilde{S}_I = \bar{Y}'S^{-1}\bar{Y} - \bar{Y}'S^{-1}X_{ABC}(X'_{ABC}S^{-1}X_{ABC})^{-1}X'_{ABC}S^{-1}\bar{Y}$ and call it \tilde{s}_I .

For $l = 1, \dots, L$:

Generate $\bar{y}_{Bijk} \sim N(0, s_{ijk}^2/n_{ijk})$, and

$$S_{Bijk}^2 \sim \left(\frac{s_{ijk}^2}{n_{ijk} - 1} \right) \chi_{n_{ijk}-1}^2, \quad i = 1, \dots, a, j = 1, \dots, b, k = 1, \dots, c,$$

Compute $\tilde{S}_{IB} = \bar{Y}'_B S_B^{-1} \bar{Y}_B - \bar{Y}'_B S_B^{-1} X_{ABC} (X'_{ABC} S_B^{-1} X_{ABC})^{-1} X'_{ABC} S_B^{-1} \bar{Y}_B$,

If $\tilde{S}_{IB} > \tilde{s}_I$, set $Q_l = 1$,

(end loop)

$\frac{1}{L} \sum_{l=1}^L Q_l$ is a Monte Carlo estimate of the p-value $P(\tilde{S}_{IB} > \tilde{s}_I)$.

3.2. Testing two-way interaction terms

For the two-way interaction terms, if we do not reject H_0 for the ABC interaction term, we drop this term and consider the model:

$$y_{ijkm} = G + A_i + B_j + C_k + [AB]_{ij} + [AC]_{ik} + [BC]_{jk} + e_{ijkm}. \quad (2)$$

Note that if the three-way interaction term $[ABC]_{ijk}$ is equal to zero for all i, j, k , this model (2) is equivalent to the full model (1). Additionally, if we do not reject H_0 for the ABC interaction term, the term would not be significantly different from zero, but weak/non-significant effects could be present. As discussed by Xu et al (Xu et al. 2013), when the three-way interaction is present, each two-way effect alone, for example BC , cannot reflect the effects of B and C because it depends on the level of the ABC interaction. So rather than testing $H_{0BC} : [BC]_{jk} = 0$, we are

actually testing $H_{0BC} : [BC]_{jk} + [ABC]_{ijk} = 0$ as shown below. In testing the two-way interaction term $[BC]$, the sum of squares for the BC and ABC interaction will be a function of $(\bar{Y} - \hat{G} - \hat{A} - \hat{B} - \hat{C} - \widehat{AB} - \widehat{AC})$, where the terms $\hat{G}, \dots, \widehat{AC}$ are the parameter estimates from fitting all terms from model (2) other than the BC term, i.e., from fitting the reduced model:

$$y_{ijkm} = G + A_i + B_j + C_k + [AB]_{ij} + [AC]_{ik} + e_{ijkm} \quad (3)$$

Similarly to the three-way interaction case, a natural test statistic for testing

$H_{0BC} : [BC]_{jk} + [ABC]_{ijk} = 0$ for $i = 1, \dots, a, j = 1, \dots, b, k = 1, \dots, c$ vs $H_{\alpha BC} : [BC]_{jk} + [ABC]_{ijk} \neq 0$ for some i, j, k

is the standardized sum of squares for the BC and ABC interaction term:

$$\bar{Y}'\Sigma^{-1}\bar{Y} - \bar{Y}'\Sigma^{-1}X_{BC}(X'_{BC}\Sigma^{-1}X_{BC})^{-1}X'_{BC}\Sigma^{-1}\bar{Y} \sim \chi^2_{abc-r(X_{BC})}, \text{ where}$$

$$X_{BC} = ([J_{abc}, I_a \otimes J_{bc}, J_a \otimes (I_b \otimes J_c), J_a \otimes (J_b \otimes I_c)], [I_{ab} \otimes J_c], [I_a \otimes (J_b \otimes I_c)])$$

For unknown Σ , the test statistic will be:

$$\tilde{S}_I = \bar{Y}'S^{-1}\bar{Y} - \bar{Y}'S^{-1}X_{BC}(X'_{BC}S^{-1}X_{BC})^{-1}X'_{BC}S^{-1}\bar{Y} \quad (1.4)$$

The test statistic (1.4) is analogous to the general linear test of the reduced model (3) above, vs the biggest model (1). The PB pivot variable for H_{0BC} is constructed based on test statistic (1.4), replacing \bar{Y} with \bar{Y}_B and S with S_B :

$$\tilde{S}_{IB} = \bar{Y}'_B S_B^{-1} \bar{Y}_B - \bar{Y}'_B S_B^{-1} X_{BC} (X'_{BC} S_B^{-1} X_{BC})^{-1} X'_{BC} S_B^{-1} \bar{Y}_B \quad (1.5)$$

For a given level α , the test rejects H_{0BC} when $P(\tilde{S}_{IB} > \tilde{s}_I) < \alpha$, where \tilde{s}_I is an observed value of \tilde{S}_I in (1.4). This probability can be estimated by Algorithm 2. Algorithm 2 should be used three times to test each two-way interaction term and is similar for each term. The X -matrix in 1.4 and 1.5 should be replaced to reflect the term under testing as follows:

$$\begin{aligned} X_{AC} &= [J_{abc}, I_a \otimes J_{bc}, J_a \otimes (I_b \otimes J_c), J_a \otimes (J_b \otimes I_c), I_{ab} \otimes J_c, J_a \otimes I_{bc}]. \\ X_{AB} &= [J_{abc}, I_a \otimes J_{bc}, J_a \otimes (I_b \otimes J_c), J_a \otimes (J_b \otimes I_c), I_a \otimes (J_b \otimes I_c), J_a \otimes I_{bc}]. \end{aligned}$$

Algorithm 2 is identical to Algorithm 1 except that X_{BC} , X_{AC} or X_{AB} replaces X_{ABC} in the calculation of \tilde{S}_I and \tilde{S}_{IB} .

3.3. Testing main effects w/no significant interaction terms

If we do not reject H_0 for any of the interaction terms, we drop these terms and consider the model

$$y_{ijkm} = G + A_i + B_j + C_k + e_{ijkm}. \quad (4)$$

In testing the main effect term C , the sum of squares for C and the interactions will be a function of $(\bar{Y} - \hat{G} - \hat{A} - \hat{B})$, where the terms \hat{G} , \hat{A} and \hat{B} are the parameter estimates from fitting all terms from model (4) other than the C term, i.e., from fitting the reduced model:

$$y_{ijm} = G + A_i + B_j + e_{ijm} \quad (5)$$

A natural test statistic for testing

$H_{0C} : C_k + [AB]_{ij} + [AC]_{ik} + [BC]_{jk} + [ABC]_{ijk} = 0$ for $i = 1, \dots, a, j = 1, \dots, b, k = 1, \dots, c$ vs $H_{\alpha C} : C_k + [AB]_{ij} + [AC]_{ik} + [BC]_{jk} + [ABC]_{ijk} \neq 0$ for some i, j, k

is the standardized sum of squares for C and the interaction terms: $\bar{Y}'\Sigma^{-1}\bar{Y} - \bar{Y}'\Sigma^{-1}X_C(X'_{C}\Sigma^{-1}X_C)^{-1}X'_{C}\Sigma^{-1}\bar{Y} \sim \chi^2_{abc-r(X_C)}$. Again note that this is not quite a test for $C_k = 0$ alone, but a test for C_k and the interaction terms, which does not involve the weights.

For unknown Σ , the test statistic will be:

$$\tilde{S}_I = \bar{Y}'S^{-1}\bar{Y} - \bar{Y}'S^{-1}X_C(X'_CS^{-1}X_C)^{-1}X'_CS^{-1}\bar{Y}, \quad (1.6)$$

where $X_C = [J_{abc}, I_a \otimes J_{bc}, J_a \otimes (I_b \otimes J_c)]$. (1.6)

The PB pivot variable for H_{0C} is constructed based on the test statistic (1.6), replacing \bar{Y} with \bar{Y}_B and S with S_B :

$$\tilde{S}_{IB} = \bar{Y}'_B S_B^{-1} \bar{Y}_B - \bar{Y}'_B S_B^{-1} X_C (X'_C S_B^{-1} X_C)^{-1} X'_C S_B^{-1} \bar{Y}_B \quad (1.7)$$

For a given level α , the test rejects H_{0BC} when $P(\tilde{S}_{IB} > \tilde{s}_I) < \alpha$, where \tilde{s}_I is an observed value of \tilde{S}_I in (1.6). This probability can be estimated by Algorithm 3. Algorithm 3 should be used three times to test each main effect term and is similar for each term. The X-matrix in 1.6 and 1.7 should be replaced to reflect the term under testing as follows:

$$\begin{aligned} X_A &= J_{abc}, J_a \otimes (I_b \otimes J_c), J_a \otimes (J_b \otimes I_c) \\ X_B &= J_{abc}, I_a \otimes J_{bc}, J_a \otimes (J_b \otimes I_c) \end{aligned}$$

Algorithm 3 is identical to Algorithm 1 except that we use X_A , X_B or X_C in place of X_{ABC} in the calculation of \tilde{S}_I and \tilde{S}_{IB} .

3.4. Testing one main effect in presence of one significant Two-Way interaction

If we do not reject H_0 for two of the interaction terms, but do reject for one of them, say AB , we drop the non-significant terms and consider the model

$$y_{ijkm} = G + A_i + B_j + C_k + [AB]_{ij} + e_{ijkm} \quad (6)$$

which would be equivalent to model (1) if all interaction terms other than AB are zero. Again we are performing a test for C and the higher order interaction terms that involve it together, which does not involve the weights, rather than C alone, after already finding the higher order terms not significantly different from zero.

In testing the main effect term C when the AB interaction term is not significantly different from 0, the sum of squares for C and the remaining interactions will be a function of $(\bar{Y} - \hat{G} - \hat{A} - \hat{B} - \widehat{AB})$, where the terms \hat{G} , \hat{A} , \hat{B} and \widehat{AB} are the parameter estimates from fitting all terms from model (6) other than the C term, i.e., from fitting the reduced model:

$$y_{ijkm} = G + A_i + B_j + [AB]_{ij} + e_{ijkm} \quad (7)$$

Similarly to the previous cases, a natural test statistic for testing

$H_{0C*} : C_k + [AC]_{ik} + [BC]_{jk} + [ABC]_{ijk} = 0$ for $i = 1, \dots, a, j = 1, \dots, b, k = 1, \dots, c$ vs $H_{\alpha C*} : C_k + [AC]_{ik} + [BC]_{jk} + [ABC]_{ijk} \neq 0$ for some i, j, k

is the standardized sum of squares for C and the interaction terms other than AB : $\bar{Y}'\Sigma^{-1}\bar{Y} - \bar{Y}'\Sigma^{-1}X_{C*}(X'_{C*}\Sigma^{-1}X_{C*})^{-1}X'_{C*}\Sigma^{-1}\bar{Y} \sim \chi^2_{abc-r(X_{C*})}$, where $X_{C*} = [J_{abc}, I_a \otimes J_{bc}, J_a \otimes (I_b \otimes J_c), I_{ab} \otimes J_c]$

For unknown Σ , the test statistic will be:

$$\tilde{S}_I = \bar{Y}'S^{-1}\bar{Y} - \bar{Y}'S^{-1}X_{C*}(X'_{C*}S^{-1}X_{C*})^{-1}X'_{C*}S^{-1}\bar{Y} \quad (1.8)$$

Similarly to the previous terms, for H_{0C*} , we can construct the PB pivot variable based on test statistic (1.8), replacing \bar{Y} with \bar{Y}_B and S with S_B :

$$\tilde{S}_{IB} = \bar{Y}'_B S_B^{-1} \bar{Y}_B - \bar{Y}'_B S_B^{-1} X_{C*} (X'_{C*} S_B^{-1} X_{C*})^{-1} X'_{C*} S_B^{-1} \bar{Y}_B \quad (1.9)$$

For a given level α , the test rejects H_{0C*} when $P(\tilde{S}_{IB} > \tilde{s}_I) < \alpha$, where \tilde{s}_I is an observed value of \tilde{S}_I in (1.8). This probability can be estimated by Algorithm 4. Algorithm 4 could be used for

to test any main effect term that is not involved in an interaction. To do so, the X-matrix in 1.8 and 1.9 should be replaced to reflect the term being tested, as follows:

$X_{A^*} = [J_{abc}, J_a \otimes (I_b \otimes J_c), J_a \otimes (J_b \otimes I_c), J_a \otimes I_{bc}]$, where the reduced model is $y_{ijkm} = G + B_j + C_k + [BC]_{jk} + e_{ijkm}$;

$X_{B^*} = [J_{abc}, I_a \otimes J_b, J_a \otimes (J_b \otimes I_c), I_a \otimes (J_b \otimes I_c)]$, where the reduced model is $y_{ijkm} = G + A_i + C_k + [AC]_{ik} + e_{ijkm}$.

Algorithm 4 is identical to Algorithm 1 except that we use X_{A^*} , X_{B^*} or X_{C^*} in place of X_{ABC} in the calculation of \tilde{S}_I and \tilde{S}_{IB} .

3.5. Simulations for testing interaction and main effects terms

For each term being tested, we again consider model (1) and reduced models shown in the previous corresponding sections. For each simulation, datasets were generated under the reduced model with $e_{ijkm} \sim N(0, \sigma_{ijk}^2)$, $i = 1, \dots, a$, $j = 1, \dots, b$, $k = 1, \dots, c$, $G = 0$, and to meet the constraints $\sum_{i=1}^a A_i = 0$, $\sum_{j=1}^b B_j = 0$, $\sum_{k=1}^c C_k = 0$, $\sum_{j=1}^b AB_{ij} = 0$, $\sum_{k=1}^c AC_{ik} = 0$, and $\sum_{k=1}^c BC_{jk} = 0$. The sample mean and sample variance vectors $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc})$, and $(s_{111}^2, s_{112}^2, \dots, s_{abc}^2)$ were calculated from each simulated dataset. The simulation was performed with:

(1) $a = b = c = 2$ to form 8 combinations;

(2) population standard deviation $\sigma_i = (\sigma_{111}, \sigma_{112}, \dots, \sigma_{222})$:

$\sigma^2_1 = (1, 1, 1, 1, 1, 1, 1, 1)$, $\sigma^2_2 = (0.1, 0.1, 0.1, 0.1, 0.5, 0.5, 0.5, 0.5)$, $\sigma^2_3 = (1, 1, 1, 1, 0.5, 0.5, 0.5, 0.5)$, $\sigma^2_4 = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 1)$, $\sigma^2_5 = (0.1, 0.3, 0.9, 0.4, 0.7, 0.5, 0.6, 1)$, $\sigma^2_6 = (0.01, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 1)$;

(3) significance level $\alpha = 0.05$ and $\alpha = 0.1$;

(4) group sizes $\mathbf{n}_i = (n_{111}, n_{112}, \dots, n_{222})$: $\mathbf{n}_1 = (5, 5, 5, 5, 5, 5, 5, 5)$, $\mathbf{n}_2 = (10, 10, 10, 10, 10, 10, 10, 10)$, $\mathbf{n}_3 = (3, 3, 4, 5, 4, 5, 6, 6)$, $\mathbf{n}_4 = (4, 6, 8, 12, 14, 16, 18, 20)$. For a given sample size and population variance configuration, we generated 2500 datasets, calculated the observed vectors $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc})$, and $(s_{111}^2, s_{112}^2, \dots, s_{abc}^2)$ from the datasets, and used 5000 PB runs to estimate the p-value using Algorithms 1-4 as indicated. The p-value for the F-test (general linear test discussed in Sec. 2) for each term was also calculated for each simulated dataset using the ‘lm’ function in R (R Core Team 2020). The tests were considered to reject if the p-value was less than α , and the proportions rejected out of the 2500 datasets were calculated for both the algorithm and the F-test, and shown in Tables 1–4.

For simulations for the three-way interaction, datasets were generated under the reduced model (2). Algorithm 1 was used to calculate the simulated p-value for the PB test, and the F-test comparing the reduced model with model (1) was calculated as described above; results shown in Table 1. Similarly: for the BC interaction term, model (3) was the reduced model and Algorithm 2 was used - results shown in Table 2; for the main effect C, model (5) was the reduced model and Algorithm 3 was used - results shown in Table 3; and for simulations of testing one main effect when one two-way term is significant, model (7) was the reduced model and Algorithm 4 was used - results shown in Table 4. We see from these tables that the F-test does not work well for some cases, but the PB test is robust; simulation results are discussed further in Sec. 5.

4. Multiple comparisons

For our three-way ANOVA illustration, if the highest order term (i.e., the three-factor interaction term) is found to have a significant effect, or if two or more of the two-factor interaction terms are found to be significant, we can approach the problem as a one-way ANOVA problem with abc levels, and then perform multiple comparisons of factor level means. Approaching this

Table 1. Simulation results for testing ABC interaction. Numbers in the table are simulated p-values. We consider four different sizes and six different variance vectors: $\mathbf{n}_1 = (5, 5, 5, 5, 5, 5, 5, 5)$; $\mathbf{n}_2 = (10, 10, 10, 10, 10, 10, 10, 10)$; $\mathbf{n}_3 = (3, 3, 4, 5, 4, 5, 6, 6)$; $\mathbf{n}_4 = (4, 6, 8, 12, 14, 16, 18, 20)$; $\sigma^2_1 = (1, 1, 1, 1, 1, 1, 1, 1)$; $\sigma^2_2 = (0.1, 0.1, 0.1, 0.1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_3 = (1, 1, 1, 1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_4 = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 1)$; $\sigma^2_5 = (0.1, 0.3, 0.9, 0.4, 0.7, 0.5, 0.6, 1)$; $\sigma^2_6 = (0.01, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 1)$, and the two different α levels shown.

		$\alpha = 0.05$		$\alpha = 0.1$	
σ^2_1		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0576	0.0528	0.1104	0.1080
	\mathbf{n}_2	0.0504	0.0492	0.1096	0.1092
	\mathbf{n}_3	0.0556	0.0536	0.1068	0.0988
	\mathbf{n}_4	0.0528	0.0476	0.1056	0.1020
σ^2_2		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0412	0.0348	0.1088	0.1012
	\mathbf{n}_2	0.0568	0.0556	0.0936	0.0944
	\mathbf{n}_3	0.0248	0.0412	0.0572	0.0848
	\mathbf{n}_4	0.0088	0.0528	0.0256	0.0972
σ^2_3		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0492	0.0468	0.1060	0.1008
	\mathbf{n}_2	0.0504	0.0500	0.0992	0.0984
	\mathbf{n}_3	0.0688	0.0456	0.1252	0.0960
	\mathbf{n}_4	0.1060	0.0544	0.1400	0.0940
σ^2_4		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0556	0.0500	0.1028	0.0968
	\mathbf{n}_2	0.0504	0.0496	0.1044	0.1032
	\mathbf{n}_3	0.0232	0.0416	0.0580	0.0928
	\mathbf{n}_4	0.0100	0.0636	0.0252	0.1000
σ^2_5		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0552	0.0504	0.0900	0.0832
	\mathbf{n}_2	0.0560	0.0548	0.0960	0.0940
	\mathbf{n}_3	0.0340	0.0460	0.0816	0.1032
	\mathbf{n}_4	0.0176	0.0484	0.0464	0.1056
σ^2_6		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0696	0.0576	0.1180	0.1036
	\mathbf{n}_2	0.0564	0.0500	0.1192	0.1124
	\mathbf{n}_3	0.0244	0.0452	0.0472	0.0904
	\mathbf{n}_4	0.0060	0.0404	0.0160	0.1048

problem using PB methods is described in detail by Zhang (2015b), which performs all pairwise comparisons of factor level means analogously to Tukey’s test, but uses PB methods to allow for unequal variances. If there are no significant interaction terms but some main effects are found to be significant, all pairwise comparisons of the factor level means of the significant main effects may be of interest.

4.1. Multiple comparisons for main effects only

Consider simultaneous comparisons of the factor A level means when no interactions are present, i.e., in model (4). An estimator of the factor A level means, similar to the estimator described in (Zhang 2015b) is a weighted average of the corresponding cell means:

$$\bar{Y}_{i\dots} = \frac{\sum_j \sum_k v_{jk} \bar{Y}_{ijk}}{\sum_j \sum_k v_{jk}}, \tag{1.10}$$

where $v_{jk} = \frac{\sum_l n_{ljk}}{N}$, with N the total number of observations.

Table 2. Simulation results for testing BC + ABC interaction. Numbers in the table are simulated p-values. We consider four different sizes and six different variance vectors: $\mathbf{n}_1 = (5, 5, 5, 5, 5, 5, 5, 5)$; $\mathbf{n}_2 = (10, 10, 10, 10, 10, 10, 10, 10)$; $\mathbf{n}_3 = (3, 3, 4, 5, 4, 5, 6, 6)$; $\mathbf{n}_4 = (4, 6, 8, 12, 14, 16, 18, 20)$; $\sigma^2_1 = (1, 1, 1, 1, 1, 1, 1, 1)$; $\sigma^2_2 = (0.1, 0.1, 0.1, 0.1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_3 = (1, 1, 1, 1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_4 = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 1)$; $\sigma^2_5 = (0.1, 0.3, 0.9, 0.4, 0.7, 0.5, 0.6, 1)$; $\sigma^2_6 = (0.01, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 1)$, and the two different α levels shown.

		$\alpha = 0.05$		$\alpha = 0.1$	
σ^2_1		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0504	0.0440	0.1040	0.0864
	\mathbf{n}_2	0.0528	0.0496	0.1044	0.1052
	\mathbf{n}_3	0.0500	0.0452	0.1040	0.0920
	\mathbf{n}_4	0.0528	0.0512	0.1008	0.0920
σ^2_2		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0784	0.0464	0.1112	0.0936
	\mathbf{n}_2	0.0688	0.0456	0.1152	0.0948
	\mathbf{n}_3	0.0496	0.0452	0.0864	0.0832
	\mathbf{n}_4	0.0344	0.0492	0.0644	0.0900
σ^2_3		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0568	0.0416	0.0980	0.0908
	\mathbf{n}_2	0.0544	0.0536	0.1020	0.0956
	\mathbf{n}_3	0.0652	0.0424	0.1288	0.0932
	\mathbf{n}_4	0.0812	0.0468	0.1392	0.0996
σ^2_4		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0624	0.0480	0.1168	0.0960
	\mathbf{n}_2	0.0640	0.0424	0.0980	0.0908
	\mathbf{n}_3	0.0388	0.0468	0.0720	0.0852
	\mathbf{n}_4	0.0320	0.0420	0.0612	0.0948
σ^2_5		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0536	0.0392	0.0984	0.0844
	\mathbf{n}_2	0.0456	0.0420	0.1048	0.1040
	\mathbf{n}_3	0.0432	0.0460	0.0852	0.0972
	\mathbf{n}_4	0.0280	0.0464	0.0604	0.0976
σ^2_6		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0816	0.0500	0.1340	0.1008
	\mathbf{n}_2	0.0784	0.0496	0.1160	0.0996
	\mathbf{n}_3	0.0336	0.0448	0.0720	0.0932
	\mathbf{n}_4	0.0188	0.0512	0.0360	0.0972

The variance of these estimators is found to be $V(\bar{Y}_{i\dots}) = \frac{1}{(\sum_{j,k} v_{jk})^2} \sum_j \sum_k v_{jk}^2 \frac{\sigma_{ijk}^2}{n_{ijk}}$ with the estimated variance

$$\hat{V}(\bar{Y}_{i\dots}) = \frac{1}{(\sum_{j,k} v_{jk})^2} \sum_j \sum_k v_{jk}^2 \frac{s_{ijk}^2}{n_{ijk}}. \tag{1.11}$$

Similarly to Tukey’s test, a test statistic for testing $H_0 : A_i = A_r$ is

$$q_{ir}^A = \frac{|\bar{Y}_{i\dots} - \bar{Y}_{r\dots}|}{\sqrt{\hat{V}(\bar{Y}_{i\dots}) + \hat{V}(\bar{Y}_{r\dots})}}$$

Since we have unequal variances and possibly also unbalanced data, the studentized range distribution typically used for Tukey’s test is inappropriate. Thus, we use the PB method to simulate a distribution for the test statistic and for the confidence interval $\bar{y}_{i\dots} - \bar{y}_{r\dots} \pm q_\alpha^A \sqrt{(\hat{V}(\bar{Y}_{i\dots}) + \hat{V}(\bar{Y}_{r\dots}))}$, where q_α^A is the $1 - \alpha$ percentile of the simulated distribution of q . The PB pivot variable for this procedure is based on the test statistic q_{ir}^A , and can be developed as follows.

Table 3. Simulation results for testing main effect C and interactions. Numbers in the table are simulated p-values. We consider four different sizes and six different variance vectors: $\mathbf{n}_1 = (5, 5, 5, 5, 5, 5, 5, 5)$; $\mathbf{n}_2 = (10, 10, 10, 10, 10, 10, 10, 10)$; $\mathbf{n}_3 = (3, 3, 4, 5, 4, 5, 6, 6)$; $\mathbf{n}_4 = (4, 6, 8, 12, 14, 16, 18, 20)$; $\sigma^2_1 = (1, 1, 1, 1, 1, 1, 1, 1)$; $\sigma^2_2 = (0.1, 0.1, 0.1, 0.1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_3 = (1, 1, 1, 1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_4 = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 1)$; $\sigma^2_5 = (0.1, 0.3, 0.9, 0.4, 0.7, 0.5, 0.6, 1)$; $\sigma^2_6 = (0.01, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 1)$, and the two different α levels shown.

		$\alpha = 0.05$		$\alpha = 0.1$	
σ^2_1		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0484	0.0404	0.0996	0.0920
	\mathbf{n}_2	0.0468	0.0460	0.1056	0.1016
	\mathbf{n}_3	0.0400	0.0376	0.1040	0.0900
	\mathbf{n}_4	0.0520	0.0524	0.1048	0.1084
σ^2_2		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0672	0.0420	0.1232	0.0984
	\mathbf{n}_2	0.0652	0.0480	0.1092	0.0872
	\mathbf{n}_3	0.0496	0.0400	0.0780	0.0876
	\mathbf{n}_4	0.0264	0.0508	0.0524	0.0956
σ^2_3		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0560	0.0424	0.1084	0.0936
	\mathbf{n}_2	0.0644	0.0540	0.1160	0.0972
	\mathbf{n}_3	0.0836	0.0400	0.1252	0.0872
	\mathbf{n}_4	0.0988	0.0460	0.1704	0.1008
σ^2_4		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0660	0.0432	0.1144	0.0820
	\mathbf{n}_2	0.0708	0.0480	0.1312	0.1084
	\mathbf{n}_3	0.0424	0.0440	0.0620	0.0792
	\mathbf{n}_4	0.0260	0.0532	0.0492	0.0988
σ^2_5		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.0656	0.0388	0.1100	0.0880
	\mathbf{n}_2	0.0580	0.0460	0.1188	0.0964
	\mathbf{n}_3	0.0440	0.0460	0.1008	0.0920
	\mathbf{n}_4	0.0352	0.0460	0.0688	0.1036
σ^2_6		F-test	Algorithm	F-test	Algorithm
	\mathbf{n}_1	0.1152	0.0524	0.1612	0.0968
	\mathbf{n}_2	0.1096	0.0504	0.1540	0.0988
	\mathbf{n}_3	0.0584	0.0488	0.0984	0.0968
	\mathbf{n}_4	0.0316	0.0460	0.0572	0.0896

For a given $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc}, s^2_{111}, s^2_{112}, \dots, s^2_{abc})$, $\bar{Y}_{Bijk} \sim N(0, s^2_{ijk}/n_{ijk})$, and $s^2_{Bijk} \sim \frac{s^2_{ijk}}{n_{ijk}-1} \chi^2_{(n_{ijk}-1)}$. In Algorithm 5 below, these variables are simulated. Then, $\bar{Y}_{Bi\dots}$ and $\bar{Y}_{B'i\dots}$ can be calculated from \bar{Y}_{Bijk} using (1.10), and the variances $V(\bar{Y}_{Bi\dots})$ and $V(\bar{Y}_{B'i\dots})$ are as in (1.11) with s^2_{Bijk} taking the place of s^2_{ijk} . Thus, the PB pivot variable is

$$q^A_{Bii'} = \frac{|\bar{Y}_{Bi\dots} - \bar{Y}_{B'i\dots}|}{\sqrt{\hat{V}(\bar{Y}_{Bi\dots}) + \hat{V}(\bar{Y}_{B'i\dots})}}. \tag{1.12}$$

Algorithm 5

For a given $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc})$, $(s^2_{111}, s^2_{112}, \dots, s^2_{abc})$, and $(n_{111}, n_{112}, \dots, n_{abc})$:
 For $l = 1, \dots, L$

Generate $\bar{Y}_{Bijk} \sim N(0, s^2_{ijk}/n_{ijk})$ and $s^2_{Bijk} \sim \frac{s^2_{ijk}}{n_{ijk}-1} \chi^2_{n_{ijk}-1}$

Compute $q^A_{Bii'}$ using (1.12) for $i = 1, \dots, a - 1, i' = i + 1, \dots, a$

Find $q_l = \max(q^A_{Bii'})$

(end loop)

q^A_x is the $1 - \alpha$ percentile of the simulated distribution of q .

Table 4. Simulation results for testing main effect C when AB interaction present. Numbers in the table are simulated p-values. We consider four different sizes and six different variance vectors: $n_1 = (5, 5, 5, 5, 5, 5, 5, 5)$; $n_2 = (10, 10, 10, 10, 10, 10, 10, 10)$; $n_3 = (3, 3, 4, 5, 4, 5, 6, 6)$; $n_4 = (4, 6, 8, 12, 14, 16, 18, 20)$; $\sigma^2_1 = (1, 1, 1, 1, 1, 1, 1, 1)$; $\sigma^2_2 = (0.1, 0.1, 0.1, 0.1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_3 = (1, 1, 1, 1, 0.5, 0.5, 0.5, 0.5)$; $\sigma^2_4 = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 1)$; $\sigma^2_5 = (0.1, 0.3, 0.9, 0.4, 0.7, 0.5, 0.6, 1)$; $\sigma^2_6 = (0.01, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 1)$, and the two different α levels shown.

		$\alpha = 0.05$		$\alpha = 0.1$	
σ^2_1		F-test	Algorithm	F-test	Algorithm
	n_1	0.0480	0.0404	0.1096	0.0888
	n_2	0.0472	0.0468	0.0952	0.0972
	n_3	0.0512	0.0412	0.0908	0.0812
	n_4	0.0428	0.0404	0.1036	0.0940
σ^2_2		F-test	Algorithm	F-test	Algorithm
	n_1	0.0704	0.0396	0.1164	0.0832
	n_2	0.0668	0.0504	0.1204	0.0956
	n_3	0.0576	0.0344	0.0940	0.0756
	n_4	0.0384	0.0568	0.0680	0.0948
σ^2_3		F-test	Algorithm	F-test	Algorithm
	n_1	0.0540	0.0472	0.1100	0.0908
	n_2	0.0592	0.0528	0.1076	0.1032
	n_3	0.0824	0.0376	0.1292	0.0768
	n_4	0.0888	0.0472	0.1484	0.0904
σ^2_4		F-test	Algorithm	F-test	Algorithm
	n_1	0.0600	0.0400	0.1168	0.0908
	n_2	0.0668	0.0424	0.1160	0.0972
	n_3	0.0420	0.0428	0.0732	0.0840
	n_4	0.0284	0.0472	0.0580	0.0956
σ^2_5		F-test	Algorithm	F-test	Algorithm
	n_1	0.0612	0.0408	0.1068	0.0916
	n_2	0.0620	0.0520	0.1196	0.1080
	n_3	0.0504	0.0392	0.0944	0.0888
	n_4	0.0376	0.0436	0.0736	0.1016
σ^2_6		F-test	Algorithm	F-test	Algorithm
	n_1	0.1108	0.0452	0.1496	0.0988
	n_2	0.0916	0.0572	0.1524	0.1064
	n_3	0.0696	0.0500	0.1052	0.0948
	n_4	0.0336	0.0444	0.0512	0.0984

The procedure for simultaneous comparisons of the factor B or C level means, when no interactions are present, is analogous to Algorithm 5.

4.2. Multiple comparisons for two-way interaction term

Consider simultaneous comparisons of the levels of the AB interaction term in model (6). An estimator of the AB level means is a weighted average of the corresponding cell means, similar to the weights described in (Zhang 2015b):

$$\bar{Y}_{ij..} = \sum_k v_k \bar{Y}_{ijk}, \tag{1.13}$$

where $v_k = \frac{\sum_{i,j} n_{ijk}}{N}$, with N the total number of observations.

The variance of these estimators is found to be $V(\bar{Y}_{ij..}) = \sum_k v_k^2 \frac{\sigma_{ijk}^2}{n_{ijk}}$ with the estimated variance $\hat{V}(\bar{Y}_{ij..}) = \sum_k v_k^2 \frac{s_{ijk}^2}{n_{ijk}}$. Similarly to Tukey’s test, a test statistic for testing $H_0 : AB_{ij} = AB_{i'j'}$ is

$$q_{ij'i'j'}^{AB} = \frac{|\bar{Y}_{ij..} - \bar{Y}_{i'j'..}|}{\sqrt{\hat{V}(\bar{Y}_{ij..}) + \hat{V}(\bar{Y}_{i'j'..})}} \tag{1.14}$$

Since the variances are unequal and the data possibly unbalanced, the studentized range distribution typically used for Tukey's test is inappropriate. Thus, we use the PB method to simulate a distribution for the test statistic and for the confidence interval $\bar{y}_{ij..} - \bar{y}_{i'j'..} \pm q_{\alpha}^{AB} \sqrt{(\hat{V}(\bar{Y}_{ij..}) + \hat{V}(\bar{Y}_{i'j'..}))}$, where q_{α}^{AB} is the $1 - \alpha$ percentile of the simulated distribution of q . The PB pivot variable for this procedure is based on the test statistic $q_{ij'i'j'}^{AB}$, and can be developed as follows.

For a given $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc}, s_{111}^2, s_{112}^2, \dots, s_{abc}^2)$, $\bar{Y}_{Bijk} \sim N(0, s_{ijk}^2/n_{ijk})$, and $s_{Bijk}^2 \sim \frac{s_{ijk}^2}{n_{ijk}-1} \chi_{(n_{ijk}-1)}^2$. In Algorithm 6 below, these variables are simulated. Then, $\bar{Y}_{Bij..}$ and $\bar{Y}_{B'i'j'..}$ can be calculated from \bar{Y}_{Bijk} using (1.15), and the variances $V(\bar{Y}_{Bij..})$ and $V(\bar{Y}_{B'i'j'..})$ are as above with s_{Bijk}^2 taking the place of s_{ijk}^2 . Thus, our PB pivot variable is:

$$q_{Bij'i'j'}^{AB} = \frac{|\bar{Y}_{Bij..} - \bar{Y}_{B'i'j'..}|}{\sqrt{\hat{V}(\bar{Y}_{Bij..}) + \hat{V}(\bar{Y}_{B'i'j'..})}}. \quad (1.15)$$

Algorithm 6

For a given $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc})$, $(s_{111}^2, s_{112}^2, \dots, s_{abc}^2)$, and $(n_{111}, n_{112}, \dots, n_{abc})$:

For $l = 1, \dots, L$

Generate $\bar{Y}_{Bijk} \sim N(0, s_{ijk}^2/n_{ijk})$ and $s_{Bijk}^2 \sim \frac{s_{ijk}^2}{n_{ijk}-1} \chi_{n_{ijk}-1}^2$

Compute $q_{Bij'i'j'}^{AB}$ using (1.15) for all pairs $(ij, i'j')$ where $ij \neq i'j'$.

Take q_l to be the max of the $q_{Bij'i'j'}^{AB}$ for the l^{th} run.

(end loop)

q_{α}^{AB} is the $1 - \alpha$ percentile of the simulated distribution of q . Reject $H_0 : AB_{ij} = AB_{i'j'}$ if the test statistic (1.14) is greater than q_{α}^{AB} .

4.3. MCP simulations

Datasets were generated under model (4) for simulating MCP for levels of Factor A and under model (6) for MCP for levels of the AB interaction term, assuming $EY = 0$ for all factor levels (such that $H_0 : A_i = A_{i'}$ is true or $H_0 : AB_{ij} = AB_{i'j'}$ is true, respectively). For both simulations, the sample mean and sample variance vectors $(\bar{y}_{111}, \bar{y}_{112}, \dots, \bar{y}_{abc})$, and $(s_{111}^2, s_{112}^2, \dots, s_{abc}^2)$ were calculated from each simulated dataset. The simulations were performed with $a = 3$, $b = 2$, $c = 4$ to form 24 combinations, and the population variances and sample size scenarios as:

$$\begin{aligned} \sigma_1^2 &= (1, 1, \dots, 1), \sigma_2^2 = (0.1, 0.1, \dots, 0.1, 0.5, 0.5, \dots, 0.5), \sigma_3^2 = (1, 1, \dots, 1, 0.5, 0.5, \dots, 0.5), \\ \sigma_4^2 &= (0.1, 0.1, 0.1, 0.2, 0.2, 0.2, 0.3, 0.3, 0.3, 0.4, 0.4, 0.4, 0.5, 0.5, 0.5, 0.6, 0.6, 0.6, \\ &0.7, 0.7, 0.7, 1, 1, 1), \sigma_5^2 = (0.1, 0.1, 0.1, 0.3, 0.3, 0.3, 0.9, 0.9, 0.9, 0.4, 0.4, 0.4, 0.7, \\ &0.7, 0.7, 0.5, 0.5, 0.5, 0.6, 0.6, 0.6, 1, 1, 1), \sigma_6^2 = (0.01, 0.01, 0.01, 0.1, 0.1, 0.1, \\ &0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 1, 1, 1), \\ \text{and } n_1 &= (5, 5, \dots, 5), n_2 = (10, 10, \dots, 10), n_3 = (3, 3, 3, 3, 3, 3, 4, 4, 4, 5, 5, \\ &5, 4, 4, 4, 5, 5, 5, 6, 6, 6, 6, 6, 6), n_4 = (4, 4, 4, 6, 6, 6, 8, 8, 8, 12, 12, 12, 14, \\ &14, 14, 16, 16, 16, 18, 18, 18, 20, 20, 20). \end{aligned}$$

Table 5. Results of simulations for testing multiple comparisons for factor A. Numbers in the table are simulated p-values. We consider four different sizes and six different variance vectors as shown in Section 4.3, with the two different α levels shown.

		$\alpha = 0.05$		$\alpha = 0.1$	
σ^2_1		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0564	0.0544	0.0968	0.0972
	n_2	0.0448	0.0432	0.0972	0.1012
	n_3	0.0584	0.0512	0.0892	0.0864
	n_4	0.0488	0.0524	0.1016	0.1040
σ^2_2		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0564	0.0444	0.1028	0.0900
	n_2	0.0564	0.0508	0.1008	0.0948
	n_3	0.0332	0.0496	0.0616	0.0936
	n_4	0.0240	0.0584	0.0428	0.0860
σ^2_3		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0476	0.0412	0.0944	0.0880
	n_2	0.0544	0.0476	0.0996	0.0972
	n_3	0.0680	0.0484	0.1220	0.0916
	n_4	0.0792	0.0472	0.1512	0.1088
σ^2_4		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0580	0.0480	0.1008	0.0952
	n_2	0.0564	0.0476	0.1008	0.0984
	n_3	0.0340	0.0536	0.0732	0.0976
	n_4	0.0224	0.0472	0.0520	0.1004
σ^2_5		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0524	0.0496	0.1004	0.0980
	n_2	0.0496	0.0536	0.0980	0.0968
	n_3	0.0448	0.0520	0.0800	0.0940
	n_4	0.0292	0.0444	0.0656	0.0956
σ^2_6		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0736	0.0544	0.1100	0.0912
	n_2	0.0612	0.0508	0.1084	0.0956
	n_3	0.0276	0.0480	0.0616	0.0968
	n_4	0.0180	0.0508	0.0368	0.1000

Each scenario was simulated for $\alpha = 0.05$ and $\alpha = 0.1$. For the factor A MCP simulation, Tukey's test was also performed on factor A for each dataset using the 'TukeyHSD' function in R, and on factor AB for the simulation for the AB term MCP. The smallest p-value for Tukey's test was checked and the test was considered to reject if this p-value was less than α . For the algorithms, the $1 - \alpha$ percentile was taken from the simulated PB distribution, and the test was considered to reject if the test statistic for the simulated dataset was greater than this percentile. The proportions rejected out of the 2500 datasets were calculated for both the algorithm and Tukey's test, and shown in Table 5 for the factor A MCP simulation and Table 6 for the AB simulation.

5. Discussion

As shown in Table 1, under equal variances and equal sample sizes, the F-test and the algorithm perform similarly, with overall p-values near the nominal level. In particular, with equal variances for all groups (σ^2_1), both tests are near the nominal level for all simulated sample sizes. However, for the other simulated (unequal) variances, the F-test begins to over-reject or under-reject the null hypothesis for those sample sizes with unbalanced data (n_3 and n_4).

For σ^2_2 , σ^2_4 , σ^2_5 , and σ^2_6 , the F-test rejects the null hypothesis less often than would be expected when we have unbalanced data, indicating the F-statistic is artificially small due to the pooled variance estimate being artificially large. This is particularly true for n_4 , where the largest group size has the largest variance (recall that calculating an estimate of pooled variance involves weighting each sample variance by the sample sizes of the respective groups). On the other hand,

Table 6. Results of simulations for testing multiple comparisons for levels of AB. Numbers in the table are simulated p-values. We consider four different sizes and six different variance vectors as shown in Section 4.3, with the two different α levels shown.

		$\alpha = 0.05$		$\alpha = 0.1$	
σ^2_1		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0476	0.0368	0.0972	0.0888
	n_2	0.0524	0.0452	0.0940	0.0932
	n_3	0.0456	0.0432	0.1008	0.0784
	n_4	0.0452	0.0444	0.1092	0.1044
σ^2_2		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0768	0.0420	0.1376	0.0920
	n_2	0.0892	0.0500	0.1472	0.1020
	n_3	0.0552	0.0408	0.0988	0.0888
	n_4	0.0416	0.0508	0.0732	0.0944
σ^2_3		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0456	0.0372	0.1012	0.0900
	n_2	0.0592	0.0504	0.1004	0.0920
	n_3	0.0696	0.0376	0.1308	0.0804
	n_4	0.1028	0.0420	0.1624	0.0900
σ^2_4		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0792	0.0460	0.1192	0.0936
	n_2	0.0724	0.0456	0.1228	0.0820
	n_3	0.0484	0.0388	0.0852	0.0896
	n_4	0.0236	0.0488	0.0540	0.0936
σ^2_5		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.0660	0.0468	0.1232	0.0980
	n_2	0.0736	0.0560	0.1052	0.0952
	n_3	0.0516	0.0456	0.0844	0.0796
	n_4	0.0368	0.0500	0.0752	0.0976
σ^2_6		Tukey	Algorithm	Tukey	Algorithm
	n_1	0.1148	0.0400	0.1500	0.0856
	n_2	0.0936	0.0464	0.1680	0.0996
	n_3	0.0572	0.0384	0.0852	0.1012
	n_4	0.0368	0.0448	0.0564	0.0920

for σ^2_3 , the F-test rejects the null hypothesis more often than expected when we also have (pronounced) unbalanced data. In this case, the F-statistic is artificially large due to the pooled variance estimate being artificially small. Again, this is true for n_4 , where the largest group size has a smaller variance. This is not as pronounced for n_3 ; although this group has unequal sample sizes they are closer together than for n_4 . These trends appear to be true both for $\alpha = 0.05$ and $\alpha = 0.1$. In these cases where the F-test is either too conservative or too liberal, the algorithm still appears to give satisfactory results, rejecting the null hypothesis with a proportion close to the nominal level α .

The simulation results for testing the BC (and ABC) interaction were similar to those for testing the ABC interaction. Again, under equal variances and equal sample sizes, the F-test and the algorithm perform similarly, with overall p-values near the nominal level. However, for the other simulated (unequal) variances, the F-test begins to over-reject or under-reject the null hypothesis for those sample sizes with unbalanced data (n_3 and n_4). For σ^2_2 , σ^2_4 , σ^2_5 , and σ^2_6 , the F-test rejects the null hypothesis less often than would be expected when we have unbalanced data. Also similar to the results for testing the three-way interaction, for σ^2_3 , the F-test rejects the null hypothesis more often than expected when we also have (pronounced) unbalanced data, both for $\alpha = 0.05$ and $\alpha = 0.1$. Again, in all simulated cases, the proportion rejected using the algorithm was fairly close to the nominal level. The results shown in Tables 3 and 4, with the F-test rejecting more or less often than the nominal level in cases with both unequal variances and unbalanced data, while the algorithm performs satisfactorily in each case, are similar to results of the

simulations shown in Tables 1 and 2. While we illustrated this method with a three-way ANOVA model, the test statistic \tilde{S}_I for each term in the model takes on the same form. Thus, we expect higher-way models to follow the same pattern, though interpretation becomes more complicated with additional factors.

Table 5 shows the results for comparing our MCP PB method (Algorithm 5 - pairwise comparisons of the levels of factor A) to Tukey's test. As with the other simulations, the simulated p-values are near the specified α level for both methods when we have homoscedasticity and balanced data (σ_1^2 and n_1 or n_2). However, in cases with both unequal variances and unbalanced data, the simulated p-values for the algorithm are generally near the specified α level, whereas those for Tukey's test tend to be too conservative. An exception to this is with σ_3^2 and n_4 , where Tukey's test rejected H_0 more often than the nominal level. Similarly to our comparisons between Algorithms 1-4 and the F-test, for σ_3^2 and n_4 , smaller variances correspond to larger sample sizes, so the pooled variance estimate used for Tukey's test becomes artificially small, and thus the test statistic artificially large. Table 6 shows the results for comparing our MCP PB method (Algorithm 6 - pairwise comparisons of the levels of the AB interaction term) to the analogous version of Tukey's test. These results are very similar to the results shown in Table 5, the multiple comparisons of the levels of A.

In this research, we looked at the three-factor heteANOVA problem with unbalanced data, including MCP's analogous to Tukey's test from a parametric bootstrap view and proposed applicable PB tests. Simulation results show that traditional tests and the PB tests give acceptable results under the equal variance assumption. Additionally, when data are balanced, the classical F-tests and MCP's perform satisfactorily in most heteroscedastic cases. However, for heteANOVA problems when the equal variance assumption is violated and data are unbalanced, the traditional tests no longer provide reasonable nominal levels, while the proposed PB methods works well and is easy to implement.

Some limitations of the proposed PB methods are that they still require the normality assumption, so if a particular dataset violates both the normality and homoscedasticity assumptions, a transformation may still be needed. Additionally, as discussed in the introduction and in (Christensen 2016), we may need to exercise caution when making practical decisions based on differences in means between groups with unequal variances, carefully considering implications for the practical issue being studied. In this study, we only examined two levels for each factor in our simulations for Algorithms 1-4, for simplicity, so further simulations with additional levels may be warranted. Despite these limitations, the proposed PB tests provide viable methods for dealing with multi-factor heteANOVA problems and MCP. Further areas for research may include extending the procedures to more complicated models, such as additional factors/levels or more complex designed experiments.

Disclosure statement

No potential conflict of interest was reported by the authors.

Appendix-R code for Algorithms 1-6

```
#Algorithm 1
alg.ABC <- function(ns, ybars, s2, a, b, c, L){
  S <- diag(s2/ns) ##make S matrix

  ##make terms for X matrix
  J.abc <- rep(1, a*b*c)
  I.a <- diag(a)
```

```

I.b <- diag(b)
I.c <- diag(c)
J.bc <- rep(1, b*c)
J.a <- rep(1, a)
J.b <- rep(1, b)
J.c <- rep(1, c)
I.ab <- diag(a*b)
I.bc <- diag(b*c)
X <- as.matrix(cbind(
J.abc, kronecker(I.a, J.bc), kronecker(J.a, kronecker(I.b, J.c)), kronecker(J.
a, kronecker(J.b, I.c)),
kronecker(I.ab, J.c), kronecker(I.a, kronecker(J.b, I.c)), kronecker(J.a, I.
bc)))

#test statistic
library(MASS)
SI <- t(ybars)%%solve(S)%%ybars -
t(ybars)%%solve(S)%%X%%ginv(t(X)%%solve(S)%%X)%%t(X)%%solve
(S)%%ybars

##Q, counts how many times test stat is less than PB pivot variable
Q <- NULL
for(j in 1:L) {
ybar.B <- NULL
S2B <-NULL
for(i in 1:length(ybars)) {
ybar.B[i] <- rnorm(1, mean=0, sd=sqrt(s2/ns)[i]) ##create bootstrap mean vector
S2B[i] <- rchisq(1, df=(ns[i]-1)) * s2[i]/(ns[i]-1) ##create bootstrap var-
iances vector
}
SB <- diag(S2B/ns)

##PB variable:
SIB <- t(ybar.B)%%solve(SB)%%ybar.B -
t(ybar.B)%%solve(SB)%%X%%ginv(t(X)%%solve(SB)%%X)%%t(X)%%solve
(SB)%%ybar.B
Q[j] <- ifelse(SIB>SI, 1, 0)
}
return(sum(Q)/length(Q)) ##p-value
}

#####
#Algorithm 2
alg.BC <- function(ns, ybars, s2, a, b, c, L){
S <- diag(s2/ns) ##make S matrix

##make terms for X matrix
J.abc <- rep(1, a*b*c)
I.a <- diag(a)
I.b <- diag(b)
I.c <- diag(c)
J.bc <- rep(1, b*c)
J.a <- rep(1, a)
J.b <- rep(1, b)

```

```

J.c <- rep(1,c)
I.ab <- diag(a*b)
I.bc <- diag(b*c)

X <- as.matrix(cbind(
J.abc, kronecker(I.a, J.bc), kronecker(J.a, kronecker(I.b, J.c)), kronecker(J.
a, kronecker(J.b, I.c)),
kronecker(I.ab, J.c), kronecker(I.a, kronecker(J.b, I.c))))

#test statistic
library(MASS)
SI <- t(ybars)%%solve(S)%%ybars -
t(ybars)%%solve(S)%%X%%ginv(t(X)%%solve(S)%%X)%%t(X)%%solve
(S)%%ybars

##Q, counts how many times test stat is less than PB pivot variable
Q <- NULL
for(j in 1:L) {
ybar.B <- NULL
S2B <-NULL
for (i in 1:length(ybars)) {
ybar.B[i] <- rnorm(1, mean=0, sd=sqrt(s2/ns)[i]) ##create bootstrap mean vector
S2B[i] <- rchisq(1, df=(ns[i]-1)) * s2[i]/(ns[i]-1) ##create bootstrap var-
iances vector
}
SB <- diag(S2B/ns)

##PB variable:
SIB <- t(ybar.B)%%solve(SB)%%ybar.B -
t(ybar.B)%%solve(SB)%%X%%ginv(t(X)%%solve(SB)%%X)%%t(X)%%solve
(SB)%%ybar.B
Q[j] <- ifelse(SIB>SI, 1, 0)
}
return(sum(Q)/length(Q)) ##p-value
}

#####
#Algorithm 3
alg.C <- function(ns, ybars, s2, a, b, c, L){
S <- diag(s2/ns) ##make S matrix

##make terms for X matrix
J.abc <- rep(1, a*b*c)
I.a <- diag(a)
I.b <- diag(b)
I.c <- diag(c)
J.bc <- rep(1, b*c)
J.a <- rep(1, a)
J.b <- rep(1,b)
J.c <- rep(1,c)
I.ab <- diag(a*b)
I.bc <- diag(b*c)

```

```
X <- as.matrix(cbind(J.abc, kronecker(I.a, J.bc), kronecker(J.a, kronecker(I.
b, J.c))))
```

```
#test statistic
```

```
library(MASS)
```

```
SI <- t(ybars)%*%solve(S)%*%ybars -
```

```
t(ybars)%*%solve(S)%*%X%*%ginv(t(X)%*%solve(S)%*%X)%*%t(X)%*%solve
(S)%*%ybars
```

```
##Q, counts how many times test stat is less than PB pivot variable
```

```
Q <- NULL
```

```
for(j in 1:L) {
```

```
  ybar.B <- NULL
```

```
  S2B <-NULL
```

```
  for (i in 1:length(ybars)) {
```

```
    ybar.B[i] <- rnorm(1, mean=0, sd=sqrt(s2/ns)[i]) ##create bootstrap mean vector
```

```
    S2B[i] <- rchisq(1, df=(ns[i]-1)) * s2[i]/(ns[i]-1) ##create bootstrap var-
    iances vector
```

```
  }
```

```
  SB <- diag(S2B/ns)
```

```
##PB variable:
```

```
SIB <- t(ybar.B)%*%solve(SB)%*%ybar.B -
```

```
t(ybar.B)%*%solve(SB)%*%X%*%ginv(t(X)%*%solve(SB)%*%X)%*%t(X)%*%solve
(SB)%*%ybar.B
```

```
Q[j] <- ifelse(SIB > SI, 1, 0)
```

```
}
```

```
return(sum(Q)/length(Q)) ##p-value
```

```
}
```

```
#####
```

```
#Algorithm 4
```

```
alg.C.AB <- function(ns, ybars, s2, a, b, c, L){
```

```
  S <- diag(s2/ns) ##make S matrix
```

```
##make terms for X matrix
```

```
J.abc <- rep(1, a*b*c)
```

```
I.a <- diag(a)
```

```
I.b <- diag(b)
```

```
I.c <- diag(c)
```

```
J.bc <- rep(1, b*c)
```

```
J.a <- rep(1, a)
```

```
J.b <- rep(1,b)
```

```
J.c <- rep(1,c)
```

```
I.ab <- diag(a*b)
```

```
I.bc <- diag(b*c)
```

```
X <- as.matrix(cbind(J.abc, kronecker(I.a, J.bc), kronecker(J.a, kronecker(I.
b, J.c)), kronecker(I.ab, J.c)) )
```

```
#test statistic
```

```
library(MASS)
```

```
SI <- t(ybars)%*%solve(S)%*%ybars -
t(ybars)%*%solve(S)%*%X%*%ginv(t(X)%*%solve(S)%*%X)%*%t(X)%*%solve
(S)%*%ybars
```

```
##Q, counts how many times test stat is less than PB pivot variable
```

```
Q <- NULL
```

```
for(j in 1:L) {
```

```
  ybar.B <- NULL
```

```
  S2B <-NULL
```

```
  for (i in 1:length(ybars)) {
```

```
    ybar.B[i] <- rnorm(1, mean=0, sd=sqrt(s2/ns)[i]) ##createbootstrapmeanvector
```

```
    S2B[i] <- rchisq(1, df=(ns[i]-1)) * s2[i]/(ns[i]-1) ##create bootstrap var-
```

```
    iances vector
```

```
  }
```

```
  SB <- diag(S2B/ns)
```

```
##PB variable:
```

```
SIB <- t(ybar.B)%*%solve(SB)%*%ybar.B -
```

```
t(ybar.B)%*%solve(SB)%*%X%*%ginv(t(X)%*%solve(SB)%*%X)%*%t(X)%*%solve
(SB)%*%ybar.B
```

```
Q[j] <- ifelse(SIB>SI, 1, 0)
```

```
}
```

```
return(sum(Q)/length(Q)) ##p-value
```

```
}
```

```
#####
```

```
#Algorithm 5
```

```
Q.test.dist <- function(L=5000, ns, means, s2, alpha=0.05, a, b, c){
```

```
##Calculate weights for actual test stat and the PB pivot variable
```

```
library(plyr)
```

```
ns.ind <- arrange(expand.grid(A=1:a, B=1:b, C=1:c), A,B)
```

```
n.grp <- array(0, c(a,b,c))
```

```
for(i in 1:a){
```

```
  for(j in 1:b){
```

```
    for(k in 1:c)
```

```
      n.grp[i,j,k] = ns[which(ns.ind$A==i & ns.ind$B==j & ns.ind$C==k)]
```

```
  }
```

```
}
```

```
v.weight <- matrix(0, b, c)
```

```
for(j in 1:b){
```

```
  for(k in 1:c){
```

```
    v.weight[j,k] <- sum(n.grp[,j,k])
```

```
  }
```

```
}
```

```
vjk <- as.vector(t(v.weight/sum(ns))) ##the weights in order of the j,k index
```

```
#calculate factor level estimated means (using the weights) for the test
statistic
```

```
ybari <- rep(0, a)
```

```
ni <- rep(0, a)
```

```
var.YA <- rep(0, a)
```

```
ni[1] <- sum(ns[1:(b*c)])
```

```
ybari[1] <- sum(vjk*means[1:(b*c)])
```

```

var.YA[1] <- sum(vjk^2 * (s2/ns)[1:(b*c)])
for(i in 2:a){
ybari[i] <- sum(vjk*means[(b*c*(i-1)+1):(i*b*c)])
ni[i] <- sum(ns[(b*c*(i-1)+1):(i*b*c)])
var.YA[i] <- sum(vjk^2 * (s2/ns)[(b*c*(i-1)+1):(i*b*c)])
}

Qtest.mat <- matrix(0,a,a)
#we just fill in upper triangular part
for (r in 1: (a -1))
for (s in (r +1):(a)) {
Qtest.mat[r,s]<- abs(ybari[r] - ybari[s])/sqrt(var.YA[r] + var.YA[s])
}
  Q.test <- max(Qtest.mat)

##calculate the parts of the PB pivot variable
Q <- rep(0, L)
for(i in 1:L) {##calculate the bootstrap means and sample variances

y.B <- rep(0, length(means))
s2.B <- rep(0, length(s2))

for (j in 1:length(means)) {
y.B[j]<- rnorm(1, 0, sqrt(s2[j]/ns[j]))
s2.B[j] <- rchisq(1, df=(ns[j]-1))*s2[j]/(ns[j]-1)
}##end the j loop

#now Q will be the PB analogy of the Q.test above. we use the same ni's
yB.bari <- rep(0,a)
var.YBA <- rep(0, a)
yB.bari[1] <- sum(vjk*y.B[1:(b*c)])
var.YBA[1] <- sum(vjk^2 * (s2.B/ns)[1:(b*c)])
for(m in 2:a){
yB.bari[m] <- sum(vjk*y.B[(b*c*(m-1)+1):(m*b*c)])
var.YBA[m] <- sum(vjk^2 * (s2.B/ns)[(b*c*(m-1)+1):(m*b*c)])
} #end m loop

Qmat <- matrix(0,a,a)
#we just fill in upper triangular part
for (r in 1: (a -1))
for (s in (r +1):a) {
Qmat[r,s]<- abs(yB.bari[r] - yB.bari[s])/sqrt(var.YBA[r] + var.YBA[s])
}

  Q[i] <- max(Qmat)
} #end i loop that has L reps
Q.crit <-quantile(Q, 1-alpha)
list(Q.crit=Q.crit, Q.test=Q.test)
}

#####
#Algorithm 6
Q.ABmc <- function(L = 5000, ns, means, s2, alpha = 0.05, a, b, c){

```

```

##get the ns, means and s2 in an array so we can identify the indices
library(plyr)
ns.ind <- arrange(expand.grid(A = 1:a, B = 1:b, C = 1:c), A,B)
n.grp <- array(0, c(a,b,c))
s2.grp <- array(0, c(a,b,c))
means.grp <- array(0, c(a,b,c))
for(i in 1:a){
  for(j in 1:b){
    for(k in 1:c){
      n.grp[i,j,k] = ns[which(ns.ind$A==i & ns.ind$B==j & ns.ind$C==k)]
      s2.grp[i,j,k] = s2[which(ns.ind$A==i & ns.ind$B==j & ns.ind$C==k)]
      means.grp[i,j,k] = means[which(ns.ind$A==i & ns.ind$B==j & ns.ind$C==k)]
    }
  }
}

##Calculate weights vk for actual test stat and the PB pivot variable
vk <- rep(0, c)
for(k in 1:c){
  vk[k] <- sum(n.grp[, ,k])
}
v.wt.k <- vk/sum(ns) ##the weights in order of the k index
#calculate estimated means (using the weights) for each level of AB for the test
statistic
ybarij <- matrix(0, a, b)
var.YAB <- matrix(0, a, b)
for(i in 1:a){
  for(j in 1:b){
    ybarij[i,j] <- sum(v.wt.k*means.grp[i,j,])
    var.YAB[i,j] <- sum(v.wt.k^2 * s2.grp[i,j,]/n.grp[i,j,])
  }
}

ybarijVect <- as.vector(ybarij)
var.YABvect <- as.vector(var.YAB)

Qtest.mat <- matrix(0,a*b,a*b)
#we just fill in upper triangular part
for (r in 1: ((a*b) -1))
for (s in (r+1):(a*b)){
Qtest.mat[r,s]<- abs(ybarijVect[r] - ybarijVect[s])/sqrt(var.YABvect[r] +
var.YABvect[s])
}

Q.test <- max(Qtest.mat)

##calculate the parts of the PB pivot variable
Q <- rep(0, L)
for(l in 1:L) {##calculate the bootstrap means and sample variances
  y.B <- rep(0, length(means))
  s2.B <- rep(0, length(s2))

  for (j in 1:length(means)) {
    y.B[j]<- rnorm(1, 0, sqrt(s2[j]/ns[j]))
    s2.B[j] <- rchisq(1, df=(ns[j]-1))*s2[j]/(ns[j]-1)
  }
}

```



```

}#end the j loop

#put the bootstrap means and s2's in indexed arrays
s2B.grp <- array(0, c(a,b,c))
meansB.grp <- array(0, c(a,b,c))
for(i in 1:a){
  for(j in 1:b){
    for(k in 1:c){
      s2B.grp[i,j,k] = s2.B[which(ns.ind$A==i & ns.ind$B==j & ns.ind$C==k)]
      meansB.grp[i,j,k] = y.B[which(ns.ind$A==i & ns.ind$B==j & ns.ind$C==k)]
      n.grp[i,j,k] = ns[which(ns.ind$A==i & ns.ind$B==j & ns.ind$C==k)]
    }
  }
}
#now Q will be the PB analogy of the Q.test above, use same weights
yB.barij <- matrix(0, a, b)
varB.YAB <- matrix(0, a, b)

for(i in 1:a){
  for(j in 1:b){
    yB.barij[i,j] <- sum(v.wt.k*meansB.grp[i,j,])
    varB.YAB[i,j] <- sum(v.wt.k^2 * s2B.grp[i,j,]/n.grp[i,j,])
  }
}

yB.barijVect <- as.vector(yB.barij)
varB.YABvect <- as.vector(varB.YAB)

Qmat <- matrix(0, a*b, a*b)
#we just fill in upper triangular part
for (r in 1: ((a*b) -1))
for (s in (r+1):(a*b)){
Qmat[r,s]<- abs(yB.barijVect[r] - yB.barijVect[s])/sqrt(varB.YABvect[r] +
varB.YABvect[s])
}

Q[1] <- max(Qmat)
} #end l loop that has L reps

Q.crit <-quantile(Q, 1-alpha)
list(Q.crit=Q.crit, Q.test=Q.test)
}

```

References

- Ananda, M. M., and S. Weerahandi. 1997. Two-way ANOVA with unequal cell frequencies and unequal variances. *Statistica Sinica* 7 (3):631–46.
- Bao, P., and M. M. Ananda. 2001. Performance of two-way ANOVA procedures when cell frequencies and variances are unequal. *Communications in Statistics - Simulation and Computation* 30 (4):805–29. doi:10.1081/SAC-100107782.
- Christensen, R. 2016. *Analysis of variance, design, and regression: Linear modeling for unbalanced data*. 2nd ed. Boca Raton, FL: CRC Press.
- Christensen, R. 2018. *Plane answers to complex questions: The theory of linear models*. 5th ed. Cham, Switzerland: Springer.

- Krishnamoorthy, K., F. Lu, and T. Mathew. 2007. A parametric bootstrap approach for ANOVA with unequal variances: Fixed and random models. *Computational Statistics and Data Analysis* 51 (12):5731–42. doi:[10.1016/j.csda.2006.09.039](https://doi.org/10.1016/j.csda.2006.09.039).
- Kutner, M. H., C. J. Nachtsheim, J. Neter, and W. Li. 2005. *Applied linear statistical models*. 5th ed. New York, NY: McGraw-Hill Irwin.
- R Core Team. 2020. *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing.
- Scheffe, H. 1959. *The analysis of variance*. New York: John Wiley and Sons.
- Searle, S. R. 1971. *Linear models*. New York: John Wiley and Sons.
- Steven, F. A. 1981. *The theory of linear models and multivariate analysis*. New York: John Wiley and Sons.
- Weerahandi, S. 1995. ANOVA under unequal error variances. *Biometrics* 51 (2):589–99. doi:[10.2307/2532947](https://doi.org/10.2307/2532947).
- Xu, L.-W., F.-Q. Yang, A. Abula, and S. Qin. 2013. A parametric bootstrap approach for two-way ANOVA in presence of possible interactions with unequal variances. *Journal of Multivariate Analysis* 115:172–80. doi:[10.1016/j.jmva.2012.10.008](https://doi.org/10.1016/j.jmva.2012.10.008).
- Zhang, G. 2015a. A parametric bootstrap approach for one-way ANOVA under unequal variances with unbalanced data. *Communications in Statistics - Simulation and Computation* 44 (4):827–32. doi:[10.1080/03610918.2013.794288](https://doi.org/10.1080/03610918.2013.794288).
- Zhang, G. 2015b. Simultaneous confidence intervals for pairwise multiple comparisons in a two-way unbalanced design with unequal variances. *Journal of Statistical Computation and Simulation* 85 (13):2727–35. doi:[10.1080/00949655.2014.935735](https://doi.org/10.1080/00949655.2014.935735).