ST 437/537: Applied Multivariate and Longitudinal Data Analysis

Comparing mean vectors from multiple independent populations: Part I

Arnab Maity
NCSU Department of Statistics
SAS Hall 5240 919-515-1937 amaity[at]ncsu.edu

Introduction

In the lecture about **[multiple treatments comparison] (Lecture05_MultiTrt.html)**, we considered the situation where each subject/item was administered all the p treatments. In this lecture, we will discuss inference involving two or more mean vectors from *independent* populations. In this case, each subject only receives one treatment (i.e., belongs to only one group/population). We first discuss the situation where there are two independent populations and then the case with more than two independent populations.

Two independent populations

Suppose we observe data from two independent populations.

> ***Population 1****: normal with mean vector $\boldsymbol{\mu}_1$ and covariance matrix $\boldsymbol{\Sigma}_1$ are unknown. We observe a random sample of size m, $\{\boldsymbol{X}_1, ..., \boldsymbol{X}_m\}$, where each $\boldsymbol{X}_i$ is a $p \times 1$ vector.*

> ***Population 2****: normal with mean vector $\boldsymbol{\mu}_2$ and covariance matrix $\boldsymbol{\Sigma}_2$ are unknown. We observe a random sample of size n, $\{\boldsymbol{Y}_1, ..., \boldsymbol{Y}_n\}$, where each $\boldsymbol{Y}_i$ is a $p \times 1$ vector.*

We want to answer the following questions:

- How to test $H_0: \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 = \delta_0$ vs. $H_0: \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 \neq \delta_0$ for some known vector δ_0?
- What are the posible values of $\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2$?

Notice that unlike the paired design discussed in the previous lecture, the sample sizes, n and m, can be different.

As an illustrative example, let us consider the **[lizard data] (data/T6-7.DAT)** shown in Table 6.7 in the Johnson and Wichern textbook. The dataset contains measurements of snout-vent length (`SVL`) and mass (`Mass`) of $m = 20$ lizards from the Cnemidophorous (C) genus and $n = 40$ lizards from the Sceloporus (S) genus. Johnson and Wichern suggest taking a natural logarithm of the variables before performing further analysis.

```
# read data as a matrix
dat <- as.matrix( read.table("data/T6-7.dat", header = F) )
colnames(dat) <- c("Mass", "SVL", "Genus")
#show the first and last few rows
head(dat)
```

```
##         Mass  SVL Genus
## [1,]  7.513 74.0     0
## [2,]  5.032 69.5     0
## [3,]  5.867 72.0     0
## [4,] 11.088 80.0     0
## [5,]  2.419 56.0     0
## [6,] 13.610 94.0     0
```

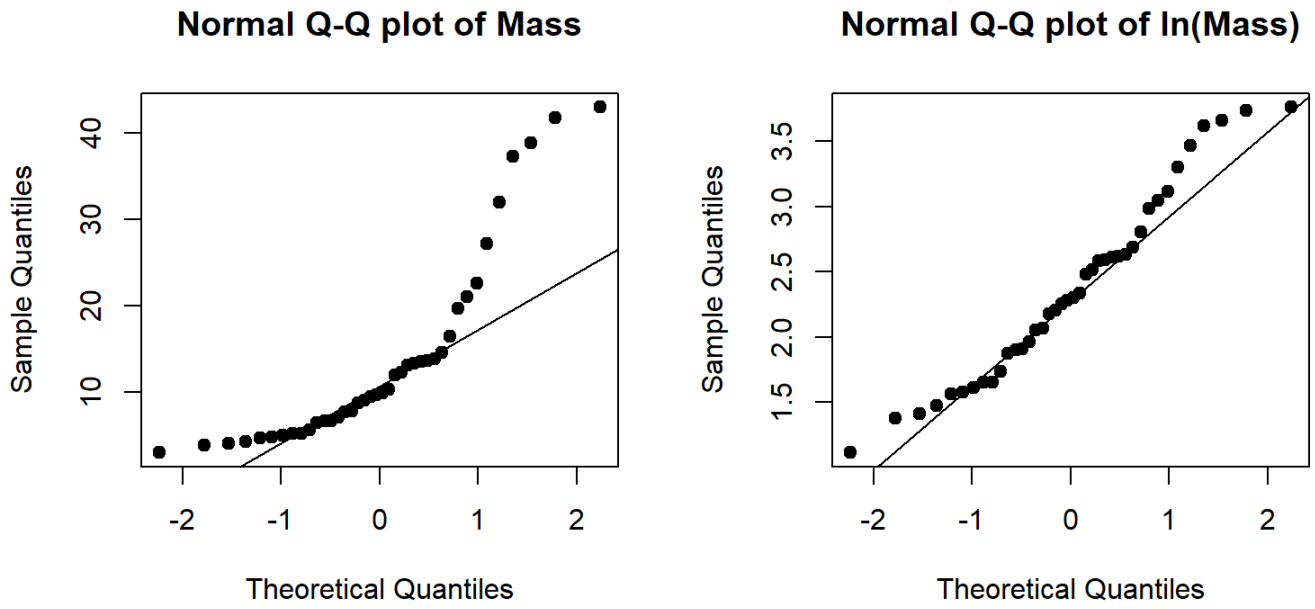
```
tail(dat)
```

```
##          Mass  SVL Genus
## [55,] 13.700 82.5     1
## [56,] 10.350 74.0     1
## [57,]  7.900 68.5     1
## [58,]  9.103 70.0     1
## [59,] 13.216 77.5     1
## [60,]  9.787 70.0     1
```

In this dataset, `genus` is coded as 0 (C) and 1 (S).

Let us view normal Q-Q plot of `Mass` and `ln(Mass)` .

```
# genera
genus <- dat[, 3]

# Normal QQ plots before and after taking logarithm
# We only show the Mass variable for genera = 1 (S)
par(mfrow = c(1,2))
mass.S <- dat[genus==1, 1]
# without log
qqnorm(mass.S, pch=19, main = "Normal Q-Q plot of Mass")
qqline(mass.S)
# with log
qqnorm(log(mass.S), pch=19, main = "Normal Q-Q plot of ln(Mass)")
qqline(log(mass.S))
```



We can see that the Q-Q plot `ln(Mass)` is close to linear, and thus assumption of normality is reasonable.

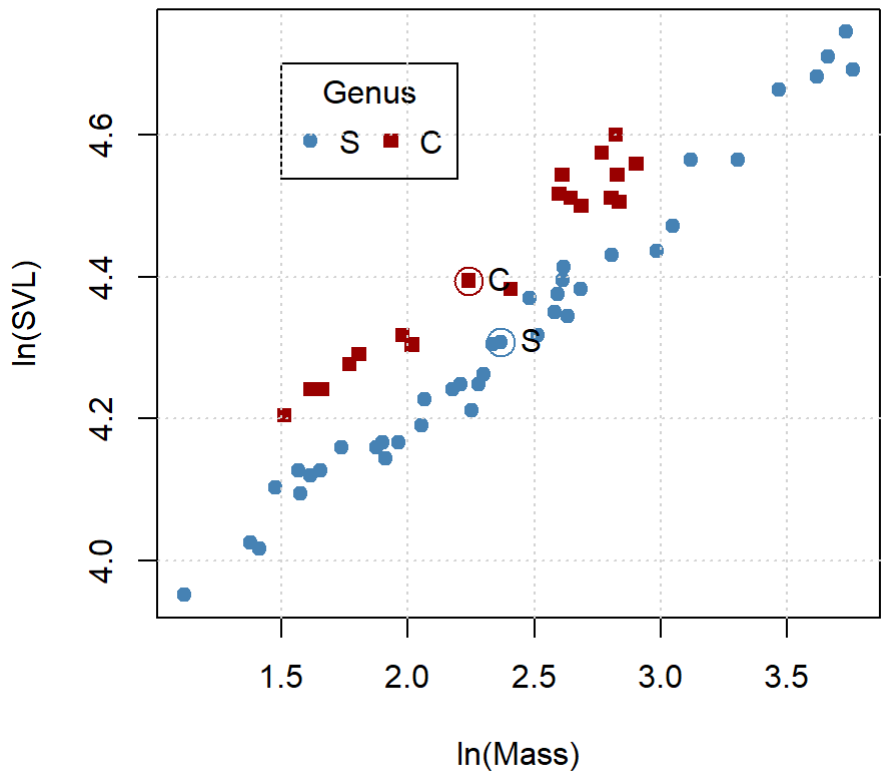
In this example, we have

$$\boldsymbol{X}_i = \begin{pmatrix} ln(Mass) \\ ln(SVL) \end{pmatrix} \text{ for the } i\text{-th lizard in C genus} \quad i = 1, \ldots, m;$$

$$\boldsymbol{Y}_i = \begin{pmatrix} ln(Mass) \\ ln(SVL) \end{pmatrix} \text{ for the } i\text{-th lizard in S genus} \quad i = 1, \ldots, n.$$

A plot of the data is shown below. The sample mean vectors are shown as well using circled points.

```
# log of the first two columns (Mass, SVL)
lndata <- cbind(log(dat[, 1:2]))
colnames(lndata) <- c("ln(Mass)", "ln(SVL)")
genus <- dat[, 3]

# scatterplot
plot(lndata[ genus==1,], pch=19, col = "steelblue") # genus S
points(lndata[ genus==0,], col = "#990000", pch=15) # genus C
legend(1.5, 4.7, legend = c("S","C"), col = c("steelblue", "#990000"), pch = c(19, 15
), title = "Genus", horiz = T)
grid()

# sample means
xbar.C <- colMeans(lndata[genus==0,])
xbar.S <- colMeans(lndata[genus==1,])

# add the means to the plot
points(xbar.C[1], xbar.C[2], pch=15, col="#990000")
points(xbar.C[1], xbar.C[2], cex=2, col="#990000")
text(xbar.C[1], xbar.C[2], labels = "C", pos=4)

points(xbar.S[1], xbar.S[2], pch=19, col="steelblue")
points(xbar.S[1], xbar.S[2], cex=2, col="steelblue")
text(xbar.S[1], xbar.S[2], labels = "S", pos=4)
```



We are interested in testing whether $\mu_1 - \mu_2$ is zero or not.

Univariate inference: two sample *t*-test

Let us first review the univariate case. Suppose we observe

$$\text{Sample 1: } X_1, \ldots, X_m \sim N(\mu_1, \sigma^2)$$

$$\text{Sample 2: } Y_1, \ldots, Y_n \sim N(\mu_2, \sigma^2)$$

Note that we are making the following assumptions:

- both populations are normal
- both populations have the same variance σ^2
- both samples are mutually independent

We are interested in making inference on $\mu_1 - \mu_2$.

Hypothesis testing: *To test the hypothesis $H_0: \mu_1 - \mu_2 = \delta_0$ vs. $H_a: \mu_1 - \mu_2 \neq \delta_0$, we use a **two-sample t-test.** The test statistic is*

$$t = \frac{(\bar{X} - \bar{Y}) - \delta_0}{\sqrt{S_p^2(\frac{1}{m} + \frac{1}{n})}} \text{ where } S_p^2 = \frac{(m-1)S_1^2 + (n-1)S_2^2}{m+n-2}.$$

Here S_1^2 and S_2^2 are sample variances for the X- and Y- samples, respectively. We reject H_0 if observed value of $|t|$ exceeds $t_{m+n-2}(\alpha/2)$ for a given significance level α.

The quantity S_p^2 is called the *pooled variance*; it is an estimator of σ^2 obtained by combining two samples together.

In R, we can use the `t.test()` function. **Read the documentation for `t.test()` for details**. Below we first estimate the difference $\mu_1 - \mu_2$ for each variable, and then perform the tests.

```
# disply the sample means of the two genera
# sample means
xbar.C <- colMeans(lndata[genus==0,])
xbar.S <- colMeans(lndata[genus==1,])
# difference between the means
difference <- xbar.C - xbar.S
rbind(xbar.C, xbar.S, difference)
```

```
##              ln(Mass)    ln(SVL)
## xbar.C      2.2399185 4.39442660
## xbar.S      2.3681404 4.30809127
## difference -0.1282218 0.08633533
```

```
# Testing for difference in mean for ln(Mass)
a <- t.test(lndata[genus==0, 1], lndata[genus==1, 1], var.equal = T)
a
```

```
##
##  Two Sample t-test
##
## data:  lndata[genus == 0, 1] and lndata[genus == 1, 1]
## t = -0.69299, df = 58, p-value = 0.4911
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4985922  0.2421485
## sample estimates:
## mean of x mean of y
##  2.239919  2.368140
```

```
# Testing for difference in mean for ln(SVL)
b <- t.test(lndata[genus==0, 2], lndata[genus==1, 2], var.equal = T)
b
```

```
##
##  Two Sample t-test
##
## data:  lndata[genus == 0, 2] and lndata[genus == 1, 2]
## t = 1.6364, df = 58, p-value = 0.1072
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.01927613  0.19194678
## sample estimates:
## mean of x mean of y
##  4.394427  4.308091
```

The `var.equal = T` argument specifies that both populations have the same variance.

> *We can relax the equal variance assumption, and obtain an approximate t-test by specifying `var.equal = F`.*

Looking at the p-values, we see that the two-sample t-tests do not detect any difference between means of the variables.

The `t.test()` function also provides 95% confidence interval for $\mu_1 - \mu_2$. We can set the confidence level using the argument `conf.level`.

```
int <- rbind(a$conf.int, b$conf.int)
rownames(int) <- c("Difference between mean ln(Mass) [C - S] ",
                   "Difference between mean ln(SVL)  [C - S] ")
int
```

```
##                                                    [,1]      [,2]
## Difference between mean ln(Mass) [C - S]  -0.49859222 0.2421485
## Difference between mean ln(SVL)  [C - S]  -0.01927613 0.1919468
```

Notice that the intervals above are one-at-a-time intervals. While *individually* they each have 95% confidence level, their joint confidence level is less.

Multivariate inference: Two-sample Hotelling's T^2 test

The multivariate procedures rely on the direct extension of the two-sample t-test to higher dimensions.

Suppose we observe

$$\text{Sample 1: } \boldsymbol{X}_1, \ldots, \boldsymbol{X}_m \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$$

$$\text{Sample 2: } \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n \sim N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$$

Note that we are making the following assumptions:

- both populations are multivariate normal;
- both populations have the same variance-covariance matrix $\boldsymbol{\Sigma}$,
- the two samples are mutually independent.

Hypothesis testing: *To test the hypothesis $H_0: \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 = \delta_0$ vs. $H_a: \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 \neq \delta_0$, we use a* ***two-sample Hotelling's T^2 test.*** *The test statistic is*

$$T^2 = \frac{m+n-p-1}{(m+n-2)p}\{(\bar{\boldsymbol{X}} - \bar{\boldsymbol{Y}}) - \delta_0\}^T \left\{ \boldsymbol{S}_{pool}\left(\frac{1}{m} + \frac{1}{n}\right)\right\}^{-1}\{(\bar{\boldsymbol{X}} - \bar{\boldsymbol{Y}}) - \delta_0\},$$

where

$$\boldsymbol{S}_{pool} = \frac{(m-1)\boldsymbol{S}_1 + (n-1)\boldsymbol{S}_2}{m+n-2}.$$

Here $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are sample covariance matrices for the X- and Y- samples, respectively. We reject H_0 if observed value of T^2 exceeds $F_{p, m+n-p-1}(\alpha)$ for a given significance level α.

The quantity $\boldsymbol{S}_p$ is called the *pooled covariance matrix*; it is an estimator of $\boldsymbol{\Sigma}$ obtained by combining two samples together.

The two-sample Hotelling's T^2 test can be performed using the `HotellingsT2()` function in the `ICSNP` library. Another library that also implements this test is `Hotelling`.

```
library(ICSNP)

## Separate the dataset
# Genus C
X <- lndata[genus == 0, ]
# Genus S
Y <- lndata[genus == 1, ]

# Two-sample Hotelling's T2 (Equal covariance matrix)
# By default tests for zero mean difference (mu2 - mu1 = 0)
# For other H0 values, use the mu argument in the function
HotellingsT2(X, Y)
```

```
##
##  Hotelling's two sample T2-test
##
## data:  X and Y
## T.2 = 112.13, df1 = 2, df2 = 57, p-value < 2.2e-16
## alternative hypothesis: true location difference is not equal to c(0,0)
```

The bivariate analysis strongly suggests that there is a size difference between the two groups of lizard.

Some remarks:

- If the sample size is moderate to large, Hotelling's T^2 is still quite robust even if there are slight departures from normality and/or a few outliers are present.
- If the two populations have unequal covariance matrices, we can still perform an approximate T^2 test; see Johnson and Wichern, pp 291–296.
- In presence of large sample size, we can obtain an approximate χ^2 test even if the populations are not normal and have different covariance matrices. Such a test is available in `HotellingsT2()` with the argument `test = "chi"`.
- We can also perform the Hotelling's T^2 using the `manova()` function in base R using the argument `test="Hotelling-Lawley"`, as shown below.

```
# Test
res <- manova(lndata ~ genus)
summary(res, test="Hotelling-Lawley")
```

```
##           Df Hotelling-Lawley approx F num Df den Df    Pr(>F)
## genus      1           3.9343   112.13      2     57 < 2.2e-16 ***
## Residuals 58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The `approx F` column gives the test statistic and the `Pr(>F)` column gives the p-value. The argument `lndata ~ genus` is a formula, where the left side contains the data matrix (`lndata`; each column is one variable, each row is one subject), the right side contains the group indicators (`genus`).

Bonferroni intervals

We can construct Bonferroni intervals for each elements of $\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2$ using a similar method discussed in previous lectures. Specifically, we will construct t-intervals but with $1 - \alpha/p$ confidence level instead of $1 - \alpha$, and using the pooled covariance estimator $\boldsymbol{S}_p$, where p is the number of variables. The intervals are shown below.

$$\text{For } \mu_{1k} - \mu_{2k}\text{: } (\bar{x}_k - \bar{y}_k) \pm t_{m+n-2}\left(\frac{\alpha}{2p}\right)\sqrt{\left(\frac{1}{m} + \frac{1}{n}\right)\boldsymbol{S}_{pool,ii}}$$

where $\boldsymbol{S}_{pool,ii}$ is the i-th diagonal entry of $\boldsymbol{S}_{pool}$.

In our example, $p = 2$. For 95% Bonferroni intervals (i.e., $\alpha = 0.05$), we need to use $1 - \alpha/2 = 1 - 0.05/2 = 0.975$ confidence level for the intervals.

```
alpha <- 0.05 # old significance level
p <- 2 # number of intervals/variables

# Genus C
X <- lndata[genus == 0, ]
# Genus S
Y <- lndata[genus == 1, ]

# mean vectors
xbar.C <- colMeans(X)
xbar.S <- colMeans(Y)


# covariances of each group
S.one <- cov(X)
S.two <- cov(Y)

# sample sizes
m <- nrow(X)
n <- nrow(Y)

# pooled covariance
S.pool = ((m-1)*S.one + (n-1)*S.two) / (m + n - 2)

# Bonferroni intervals
lower <- (xbar.C - xbar.S) - qt(alpha/(2*p), df = m+n-2, lower.tail = F)*sqrt(diag(S.
pool)*(1/m + 1/n))
upper <- (xbar.C - xbar.S) + qt(alpha/(2*p), df = m+n-2, lower.tail = F)*sqrt(diag(S.
pool)*(1/m + 1/n))

int.bonf <- cbind(lower, upper)
int.bonf
```

```
##                lower     upper
## ln(Mass) -0.55398282 0.2975392
## ln(SVL)  -0.03507081 0.2077415
```

Comparing Means from More Than Two Populations

Let us now consider the situation where we observe data from multiple independent populations. Let us consider the egyptian skull data (`skulls` dataset in `HSAUR` library). The dataset contains observations on 4 variables

- `mb` : maximum breaths of the skull,
- `bh` : basibregmatic heights of the skull,

- `bl` : basialiveolar length of the skull,
- `nh` : nasal heights of the skull,

on 150 skulls from 5 `epoch` (c4000BC c3300BC, c1850BC, c200BC, and cAD150).

```
# snapshot of the data
library(HSAUR3)
head(skulls)
```

```
##     epoch  mb  bh  bl nh
## 1 c4000BC 131 138  89 49
## 2 c4000BC 125 131  92 48
## 3 c4000BC 131 132  99 50
## 4 c4000BC 119 132  96 44
## 5 c4000BC 136 143 100 54
## 6 c4000BC 138 137  89 56
```

```
tail(skulls)
```

```
##      epoch  mb  bh  bl nh
## 145 cAD150 132 127  97 52
## 146 cAD150 137 125  85 57
## 147 cAD150 129 128  81 52
## 148 cAD150 140 135 103 48
## 149 cAD150 147 129  87 48
## 150 cAD150 136 133  97 51
```

```
# number of skulls in each epoch
table(skulls$epoch)
```

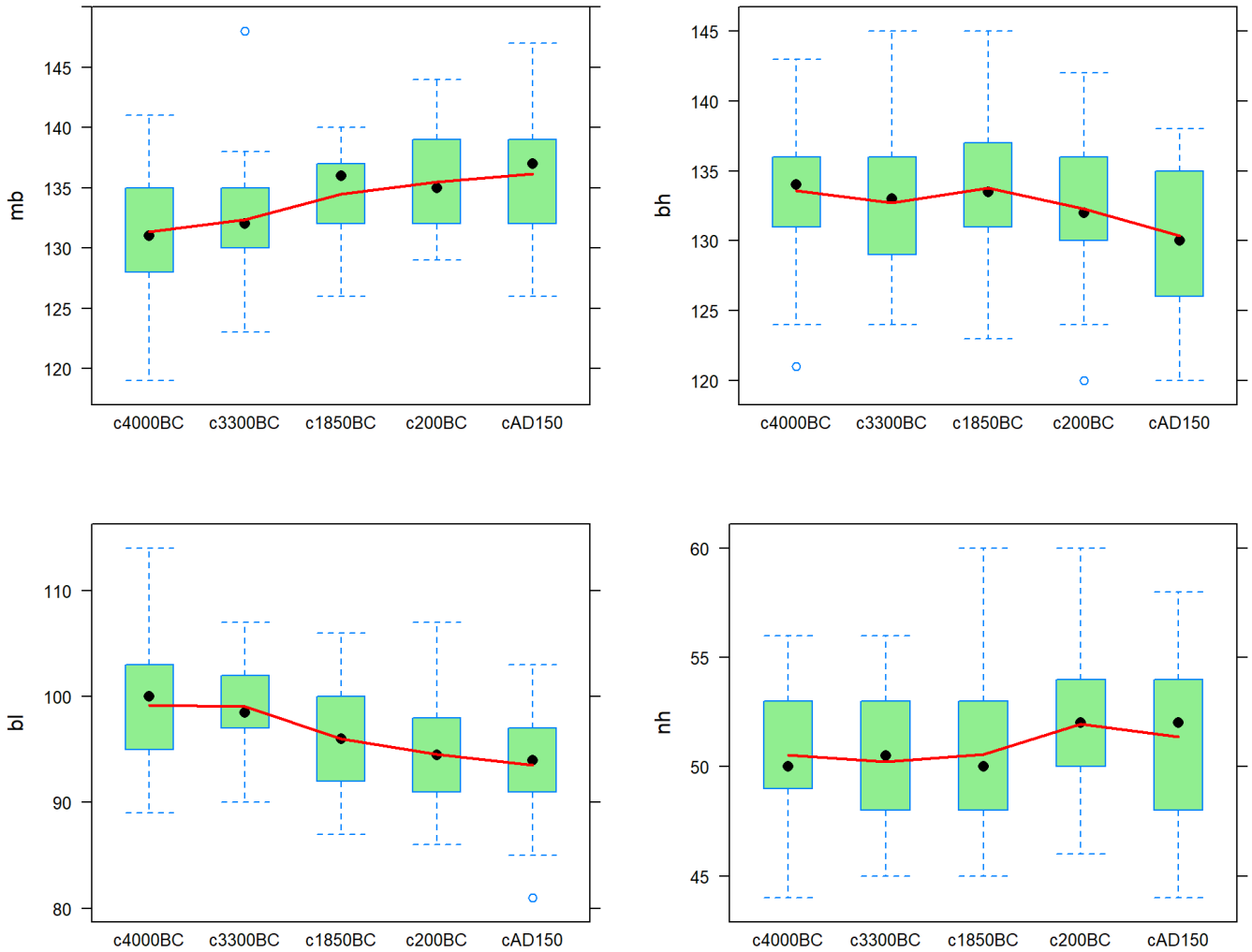
```
##
## c4000BC c3300BC c1850BC  c200BC  cAD150
##      30      30      30      30      30
```

Thus there are five groups (five epochs), and we want to compare the mean of (mb, bh, bl, nh) accross the five groups.

Let us compute sample means, as well as create boxplots to view means accross the groups.

```
dat <- as.matrix(skulls[, -1])
epoch <- skulls$epoch

# group means
aggregate(dat, by = list(epoch), FUN = mean)
```

```
##   Group.1       mb       bh       bl       nh
## 1 c4000BC 131.3667 133.6000 99.16667 50.53333
## 2 c3300BC 132.3667 132.7000 99.06667 50.23333
## 3 c1850BC 134.4667 133.8000 96.03333 50.56667
## 4  c200BC 135.5000 132.3000 94.53333 51.96667
## 5  cAD150 136.1667 130.3333 93.50000 51.36667
```



In general, the variable that creates the group is called a **factor**. The different values a factor takes are called **levels**. In this case there is only one factor, `epoch`, with five levels (the years). Thus we will employ an one-way ANOVA model to test whether the mean vectors for different groups are equal or not.

Univariate one-way ANOVA

Let us review the one-way ANOVA in the univariate situation. Assume that we observe samples from g groups:

> *Sample 1:* $X_{11}, \ldots, X_{1n_1}$ *from a* $N(\mu_1, \sigma^2)$ *population*

Sample 2: $X_{21}, \ldots, X_{2n_2}$ *from a* $N(\mu_2, \sigma^2)$ *population*

$\vdots$

Sample g*:* $X_{g1}, \ldots, X_{gn_g}$ *from a* $N(\mu_g, \sigma^2)$ *population*

Here the notation X_{ij} denotes the response for the ***j*-th individual** in the ***i*-th group**.

The main assumptions are

- each population is normal
- groups/populations has different means but **same variance σ^2**;
- the samples are mutually independent

We want to test $H_0 : \mu_1 = \ldots = \mu_g$ vs. $H_a :$ at least two means are different.

We write the ANOVA model as

$$\underbrace{X_{ij}}_{\text{Response for } j\text{-th individual in } i\text{-th group}} = \underbrace{\mu_i}_{i\text{-th group mean}} + \underbrace{e_{ij}}_{N(0,\sigma^2) \text{ errors}}$$

$$= \underbrace{\mu}_{\text{overall mean}} + \underbrace{\alpha_i}_{i\text{-th group effect}} + e_{ij}.$$

The i-th group effect is $\alpha_i =$ (i-th group mean - overall mean). We often put the constraint that $\sum_{i=1}^{g} n_i \alpha_i = 0$. In this formulation, testing for $H_0 : \mu_1 = \ldots = \mu_g$ is equivalent to testing $H_0 : \alpha_1 = \ldots = \alpha_g = 0$.

Motivated by the decomposition above, we can also decompose the observed data as follows:

$$\underbrace{x_{ij}}_{\text{Observation}} = \underbrace{\bar{x}}_{\text{overall sample mean}} + \underbrace{(\bar{x}_i - \bar{x})}_{\text{estimated group effect}} + \underbrace{(\bar{x}_{ij} - \bar{x}_i)}_{\text{residual}}.$$

Here $\bar{x}_i$ denotes the sample mean of the i-th sample, and $\bar{x}$ is the overall mean of all the observations. Thus the estimated parameters are

$$\hat{\mu} = \bar{x}, \quad \text{and} \quad \hat{\alpha}_i = \bar{x}_i - \bar{x}.$$

Note that if $H_0: \alpha_1 = \ldots = \alpha_g = 0$, that is if each $\alpha_i = 0$, then each $\hat{\alpha}_i = \bar{x}_i - \bar{x}$ should be close to zero as well. If some of the group effects, $\bar{x}_i - \bar{x}$, are large, they we should reject H_0. To assess the group effect, we look at the sum of squares:

$$\text{Treatment sum of squares (SSTr)} = \sum_{i=1}^{g} n_i(\bar{x}_i - \bar{x})^2,$$

$$\text{Error sum of squares (SSE)} = \sum_{i=1}^{g}\sum_{j=1}^{n_i} (x_{ij} - \bar{x}_i)^2.$$

The ANOVA F test rejects H_0 if

$$\frac{SSTr/(g-1)}{SSE/(\sum_i n_i - g)} > F_{g-1,\, \sum_i n_i - g}(\alpha).$$

The quantity in the numerator $SSTr/(g-1)$ is called the **Mean square for treatments (MSTr)**. If the null hypothesis is true (i.e., all the group means are same) then the estimates of the individual population means (i.e., the individual sample means) should be really close to the common overall mean. Thus, under H_0, $MSTr$ would have a small value. This quantity measures how similar or different the samples are in terms of their mean. This is a measure of between sample variation.

The quantity in the denominator $SSE/(\sum_i n_i - g)$ is called the **Mean square for error (MSE)**. This quantifies the variation within samples.

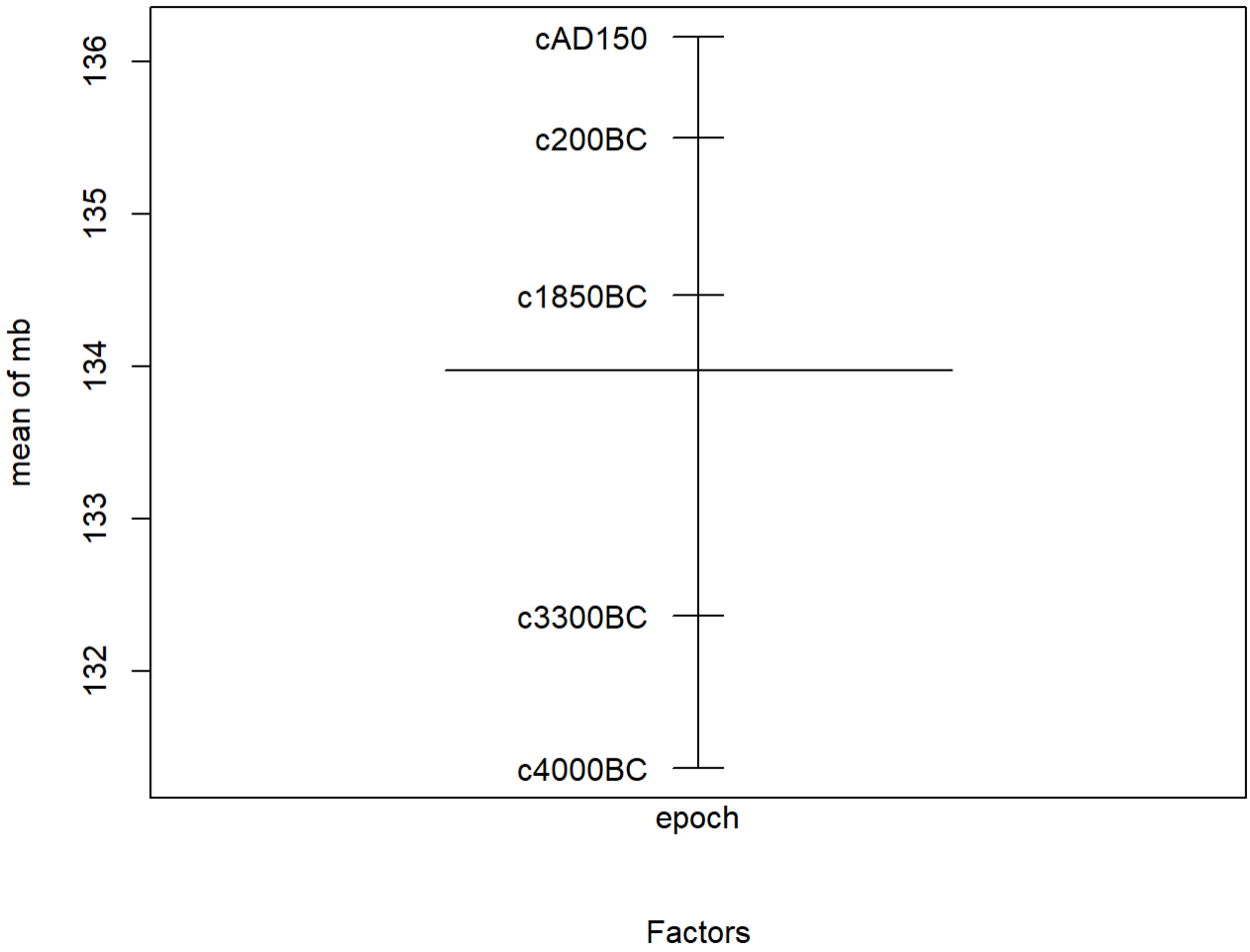
Thus the test statistic above compares between sample variation to the within sample variation.

Let us consider the `skulls` data and consider only one variable, maximum breaths of the skull (`mb`). Let us perform an one-way ANOVA. The R function is `aov`.

```
# extract `mb` and the group information
mb <- dat[, 1]
epoch <- skulls$epoch
```

The `plot.design()` function provides a simple visual of means for each level of the factor (i.e., different groups).

```
plot.design(mb ~ epoch)
```



The wider line in the middle represents the overall mean of the data accross all groups. One-way anova can be performed by the function `aov()`.

```
# one-way anova
result <- aov(mb ~ epoch)

# ANOVA F-test
summary(result)
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## epoch         4  502.8  125.71   5.955 0.000183 ***
## Residuals   145 3061.1   21.11
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The table shown above is the ANOVA table. The first column shows the `degrees of freedoms`, that is, the values $g-1$ and $(\sum_i n_i - g)$. The second column `Sum Sq` shows the values of $SSTr$ and SSE. The third column `Mean Sq` shows the values of $MSTr$ and MSE (computed by dividing the `Sum Sq` column by the `Df` column). The `F value` is then computed by dividing $MSTr$ by MSE.

The small p-value indicates the group means are not equal as we reject H_0.

Further analysis

In ANOVA, when the computed value of the F test statistic in not significant, we usually terminate the analysis since we can not identify any difference among the population means. However, when the null hypothesis is rejected (that is, we detect that there are some differences) then one might want to know which of the means are different from each other. A method for carrying out this further analysis is called a multiple comparisons procedure.

In our data example, we rejected H_0. This test, however, does not termine exactly which two means are different. We can look at the means for each group as follows.

```
# Estimated treatment effects
aggregate(x = mb, by = list(epoch), FUN = mean)
```

```
##   Group.1        x
## 1 c4000BC 131.3667
## 2 c3300BC 132.3667
## 3 c1850BC 134.4667
## 4  c200BC 135.5000
## 5  cAD150 136.1667
```

We can estimate treatment contrasts $\mu_i - \mu_{i'}, i \neq i^{'}$ by using Bonferroni intervals. Specifically, we can compute t-intervals but adjusting the confidence level for multiple comparisons.

The function `emmeans()` in the `emmeans` library estimates all treatment effects

```
library(lsmeans)
# Estimate all the group means
lsm <- emmeans(result, "epoch")
lsm
```

```
##  epoch   emmean    SE  df lower.CL upper.CL
##  c4000BC    131 0.839 145      130      133
##  c3300BC    132 0.839 145      131      134
##  c1850BC    134 0.839 145      133      136
##  c200BC     136 0.839 145      134      137
##  cAD150     136 0.839 145      135      138
##
## Confidence level used: 0.95
```

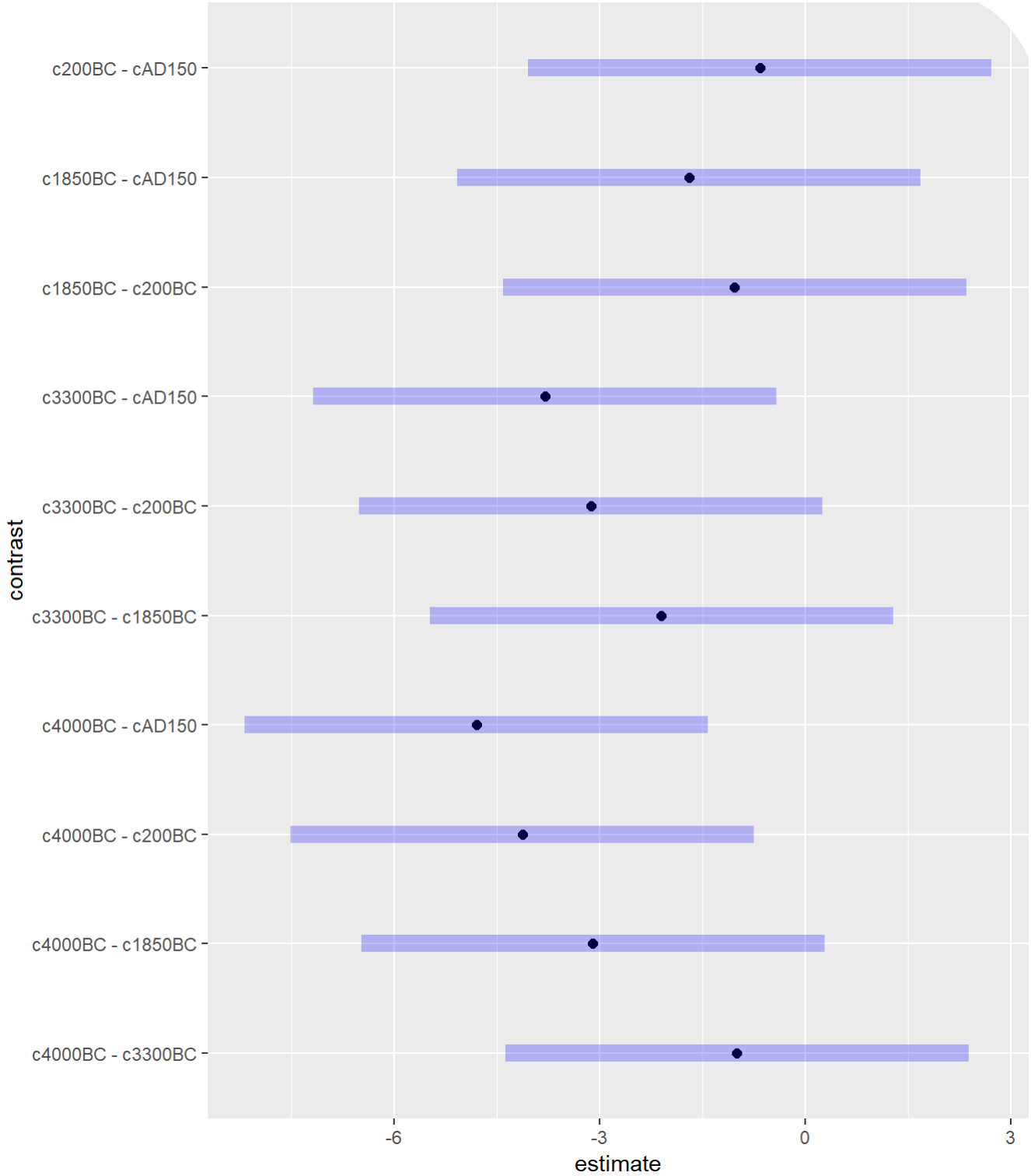
We need to estimate the pairwise treatment contrasts; however, we need to adjust for multiple comparisons. Specifically, since we have $g = 5$ groups, we have $g(g-1)/2 = 10$ pairwise differences. Thus instead of taking $(1-\alpha)$ as the confidence

level, we need to set it to $1 - \alpha/(g(g-1)/2) = 1 - \alpha/10$. The following code does so.

```
# estimate the contrasts
cont <- contrast(lsm, "pairwise")

# adjust alpha
g <- 5 # number of groups
alpha <- 0.05 # old confidence level
newalpha = alpha/(g*(g-1)/2) # adjusted confidence level

# obtain the intervals
intervals <- confint(cont, level=1-newalpha, adj="none")
intervals
```

```
##  contrast          estimate   SE  df lower.CL upper.CL
##  c4000BC - c3300BC   -1.000 1.19 145    -4.38    2.382
##  c4000BC - c1850BC   -3.100 1.19 145    -6.48    0.282
##  c4000BC - c200BC    -4.133 1.19 145    -7.52   -0.752
##  c4000BC - cAD150    -4.800 1.19 145    -8.18   -1.418
##  c3300BC - c1850BC   -2.100 1.19 145    -5.48    1.282
##  c3300BC - c200BC    -3.133 1.19 145    -6.52    0.248
##  c3300BC - cAD150    -3.800 1.19 145    -7.18   -0.418
##  c1850BC - c200BC    -1.033 1.19 145    -4.42    2.348
##  c1850BC - cAD150    -1.700 1.19 145    -5.08    1.682
##  c200BC - cAD150     -0.667 1.19 145    -4.05    2.715
##
## Confidence level used: 0.995
```

```
plot(intervals)
```



The intervals that do not contain zero indicate significant treatment contrasts.

Multivariate ANOVA (MANOVA) with one factor

Assume that we observe samples from g groups:

> *Sample 1:* $\boldsymbol{X}_{11}, \ldots, \boldsymbol{X}_{1n_1}$ *from a* $N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$ *population*

Sample 2: $\boldsymbol{X}_{21}, \ldots, \boldsymbol{X}_{2n_2}$ from a $N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$ population

$\vdots$

Sample g: $\boldsymbol{X}_{g1}, \ldots, \boldsymbol{X}_{gn_g}$ from a $N(\boldsymbol{\mu}_g, \boldsymbol{\Sigma})$ population

The main assumptions are

- each population is multivariate normal
- each group/population has different means but **same covariance matrix $\boldsymbol{\Sigma}$**;
- the samples are mutually independent

We want to test $H_0: \boldsymbol{\mu}_1 = \ldots = \boldsymbol{\mu}_g$ vs. $H_a:$ at least two means are different.

Similar to the univariate case, we write the multivariate response as

$$\underbrace{\boldsymbol{x}_{ij}}_{\text{Observation}} = \underbrace{\bar{\boldsymbol{x}}}_{\text{overall sample mean}} + \underbrace{(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}})}_{\text{estimated group effect}} + \underbrace{(\bar{\boldsymbol{x}}_{ij} - \bar{\boldsymbol{x}}_i)}_{\text{residual}}.$$

Here each term is a vector, thus we need a mutivariate analog of the univariate sum of squares, the matrix of sum of squares and cross-products:

$$\text{Treatment (Between) sum of squares and cross-products: } \boldsymbol{B} = \sum_{i=1}^{g} n_g (\boldsymbol{x}_i - \bar{\boldsymbol{x}})(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T,$$

$$\text{Error (Within) sum of squares and cross-products: } \boldsymbol{W} = \sum_{i=1}^{g} \sum_{j=1}^{n_i} (\boldsymbol{x}_{ij} - \bar{\boldsymbol{x}}_i)(\boldsymbol{x}_{ij} - \bar{\boldsymbol{x}}_i)^T.$$

We use the Wilks' lambda Test statistic:

$$\Lambda^* = \frac{|\boldsymbol{W}|}{|\boldsymbol{B} + \boldsymbol{W}|}.$$

This statistic can be thought of as ratio of generalized variances. We reject the null if Λ^* is small.

Remarks:

- The Wilks' lambda statistics is related to the likelihood ratio statistics. The Wilks' lambda statistics is a multivariate generalization of the univariate F-distribution, generalizing the F-distribution in the same way that the Hotelling's T^2 distribution generalizes Student's t-distribution.
- There are other forms of test statistics: Pillai's statistic, the Lawley-Hotelling statistic, and Roy's largest root statistic. These statistics can be written as particular functions (of the eigenvalues) of $\boldsymbol{W}^{-1}\boldsymbol{B}$.
- The exact distribution of Λ^* can be derived in special cases for p and g; for other cases a modification of Λ^* is used along with large sample approximations.
- All the above mentioned test statistics have similar large sample distributions. In the absence of normality, Pillai's trace seems to be robust among these tests.

Now let us revisit our data example, but with all four variables. Recall that there are five groups (five epochs), and we want to compare the mean of (mb, bh, bl, nh) accross the five groups. The R function `manova()` is used together for this purpose.

```
# Matrix of the responses
# each column contains one response variable
# each row is one subject
dat <- as.matrix(skulls[, -1])

# grouping factor
epoch <- skulls$epoch

# manova
out <- manova(dat ~ epoch)
```

We called `manova()` with the formula `dat ~ epoch`. Thus each row in `dat` should correspond to the same row of `epoch`.

```
# Wilk's lambda test
summary( out, test = "Wilks")
```

```
##            Df   Wilks approx F num Df den Df   Pr(>F)
## epoch       4 0.66359   3.9009     16 434.45 7.01e-07 ***
## Residuals 145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The very small p-value suggests that we reject H_0 and conclude that the group means are different. Other tests can be performed with `test` argument with "Pillai", "Hotelling-Lawley" and "Roy" as options. See the help page for `summary.manova()` for details.

Summary of individual ANOVA (univariate) results can be obtained from the same output.

```
summary.aov(out)
```

```
##  Response mb :
##              Df  Sum Sq Mean Sq F value    Pr(>F)
## epoch         4  502.83 125.707  5.9546 0.0001826 ***
## Residuals   145 3061.07  21.111
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response bh :
##              Df Sum Sq Mean Sq F value  Pr(>F)
## epoch         4  229.9  57.477  2.4474 0.04897 *
## Residuals   145 3405.3  23.485
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response bl :
##              Df Sum Sq Mean Sq F value    Pr(>F)
## epoch         4  803.3 200.823  8.3057 4.636e-06 ***
## Residuals   145 3506.0  24.179
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response nh :
##              Df Sum Sq Mean Sq F value Pr(>F)
## epoch         4   61.2  15.300   1.507 0.2032
## Residuals   145 1472.1  10.153
```

We see that the p-values for the `mb` and `bl` are very small; this indicates large difference between group means for these variables. The other two variables, `bh` and `nh`, do not show such significant results.

We can also look at the pairwise multivariate tests between different epochs. For example, we can look at the differences between epoch `c4000BC` with the other epochs. The `subset` argument can be used to choose which epochs to compare.

```
# Difference between "c4000BC", "c3300BC"
summary(manova(dat ~ epoch, data = skulls, subset = epoch %in% c("c4000BC", "c3300BC"
)))
```

```
##           Df   Pillai approx F num Df den Df Pr(>F)
## epoch      1 0.027674  0.39135      4     55 0.8139
## Residuals 58
```

```
# Difference between "c4000BC", "c1850BC"
summary(manova(dat ~ epoch, data = skulls, subset = epoch %in% c("c4000BC", "c1850BC"
)))
```

```
##           Df  Pillai approx F num Df den Df  Pr(>F)
## epoch      1 0.18757   3.1744      4     55 0.02035 *
## Residuals 58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Difference between "c4000BC", "c200BC"
summary(manova(dat ~ epoch, data = skulls, subset = epoch %in% c("c4000BC", "c200BC"
)))
```

```
##           Df  Pillai approx F num Df den Df    Pr(>F)
## epoch      1 0.30297   5.9766      4     55 0.0004564 ***
## Residuals 58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Difference between "c4000BC", "cAD150"
summary(manova(dat ~ epoch, data = skulls, subset = epoch %in% c("c4000BC", "cAD150"
)))
```

```
##           Df  Pillai approx F num Df den Df    Pr(>F)
## epoch      1 0.36182   7.7956      4     55 4.736e-05 ***
## Residuals 58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ideally, since we have 5 groups, we would perform all $C_2^5 = 10$ tests, and compare the p-values with $\alpha/4 = 0.05/10 = 0.005$ rather than simply to 0.05. The last two tests shown above provide significant p-values. It is evident that as the epochs become further separated in time, the means become more and more different from each other.

Another R function:

The `Manova()` function in the `car` library can also be used (and sometimes preferred for its detailed output) to perform manova.

```
library(car)
```

```
## Loading required package: carData
```

```
# First fit a linear regression
lmres <- lm(dat ~ epoch)

# Call Manova
summary( Manova(lmres) )
```

```
##
## Type II MANOVA Tests:
##
## Sum of squares and products for error:
##             mb          bh         bl        nh
## mb 3061.066667    5.333333   11.46667  291.3000
## bh    5.333333 3405.266667  754.00000  412.5333
## bl   11.466667  754.000000 3505.96667  164.3333
## nh  291.300000  412.533333  164.33333 1472.1333
##
## ----------------------------------------
##
## Term: epoch
##
## Sum of squares and products for the hypothesis:
##           mb         bh        bl         nh
## mb  502.8267 -228.14667 -626.6267  135.43333
## bh -228.1467  229.90667  292.2800  -66.06667
## bl -626.6267  292.28000  803.2933 -180.73333
## nh  135.4333  -66.06667 -180.7333   61.20000
##
## Multivariate Tests: epoch
##                  Df test stat  approx F num Df   den Df     Pr(>F)
## Pillai            4 0.3533056  3.512037     16 580.0000 4.6753e-06 ***
## Wilks             4 0.6635858  3.900928     16 434.4548 7.0102e-07 ***
## Hotelling-Lawley  4 0.4818191  4.230974     16 562.0000 8.2782e-08 ***
## Roy               4 0.4250954 15.409707      4 145.0000 1.5883e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In the output shown above,

- the top block gives $\boldsymbol{W}$
- the middle block gives $\boldsymbol{B}$
- the bottom block shows test results using four methods.

Univariate ANOVA results can be extracted as below.

```
summary( Anova(lmres),
         univariate=TRUE,
         multivariate=FALSE )
```

```
##
##  Type II Sums of Squares
##            df      mb      bh      bl     nh
## epoch       4  502.83  229.91  803.29   61.2
## residuals 145 3061.07 3405.27 3505.97 1472.1
##
##  F-tests
##         mb   bh   bl   nh
## epoch 5.95 2.45 8.31 1.51
##
##  p-values
##       mb         bh         bl         nh
## epoch 0.00018263 0.04896988 4.6364e-06 0.20317875
```

Pair-wise comparison

Once we have rejected the null hypothesis of equal means, our next step is to find out which components are different across groups. We can again use the Bonferroni approach. Specifically, we can create t-intervals for each pair-wise difference between groups *for each variable*. Thus, if we have g groups and p variables, we would adjust for $\frac{pg(g-1)}{2}$ comparisons.

In R, we can use the function `emmeans()` in the `emmeans` library.

```
library(emmeans)

# number of variables
p <- 4

# Create a list to store the results
pair.lst <- vector("list", p)

# name the list according to variabls (for convenience)
names(pair.lst) <- colnames(dat)

# run emmeans for each variable to estimate the group means etc
for(j in 1:p){
  wts <- rep(0, p)
  wts[j] <- 1
  pair.lst[[j]] <- emmeans(out, "epoch", weights=wts)
}

pair.lst
```

```
## $mb
##  epoch   emmean    SE  df lower.CL upper.CL
##  c4000BC    131 0.839 145      130      133
##  c3300BC    132 0.839 145      131      134
##  c1850BC    134 0.839 145      133      136
##  c200BC     136 0.839 145      134      137
##  cAD150     136 0.839 145      135      138
##
## Results are averaged over the levels of: rep.meas
## Confidence level used: 0.95
##
## $bh
##  epoch   emmean    SE  df lower.CL upper.CL
##  c4000BC    134 0.885 145      132      135
##  c3300BC    133 0.885 145      131      134
##  c1850BC    134 0.885 145      132      136
##  c200BC     132 0.885 145      131      134
##  cAD150     130 0.885 145      129      132
##
## Results are averaged over the levels of: rep.meas
## Confidence level used: 0.95
##
## $bl
##  epoch   emmean    SE  df lower.CL upper.CL
##  c4000BC   99.2 0.898 145     97.4    100.9
##  c3300BC   99.1 0.898 145     97.3    100.8
##  c1850BC   96.0 0.898 145     94.3     97.8
##  c200BC    94.5 0.898 145     92.8     96.3
##  cAD150    93.5 0.898 145     91.7     95.3
##
## Results are averaged over the levels of: rep.meas
## Confidence level used: 0.95
##
## $nh
##  epoch   emmean    SE  df lower.CL upper.CL
##  c4000BC   50.5 0.582 145     49.4     51.7
##  c3300BC   50.2 0.582 145     49.1     51.4
##  c1850BC   50.6 0.582 145     49.4     51.7
##  c200BC    52.0 0.582 145     50.8     53.1
##  cAD150    51.4 0.582 145     50.2     52.5
##
## Results are averaged over the levels of: rep.meas
## Confidence level used: 0.95
```

First we adjust α accordingly due to multiple comparisons.

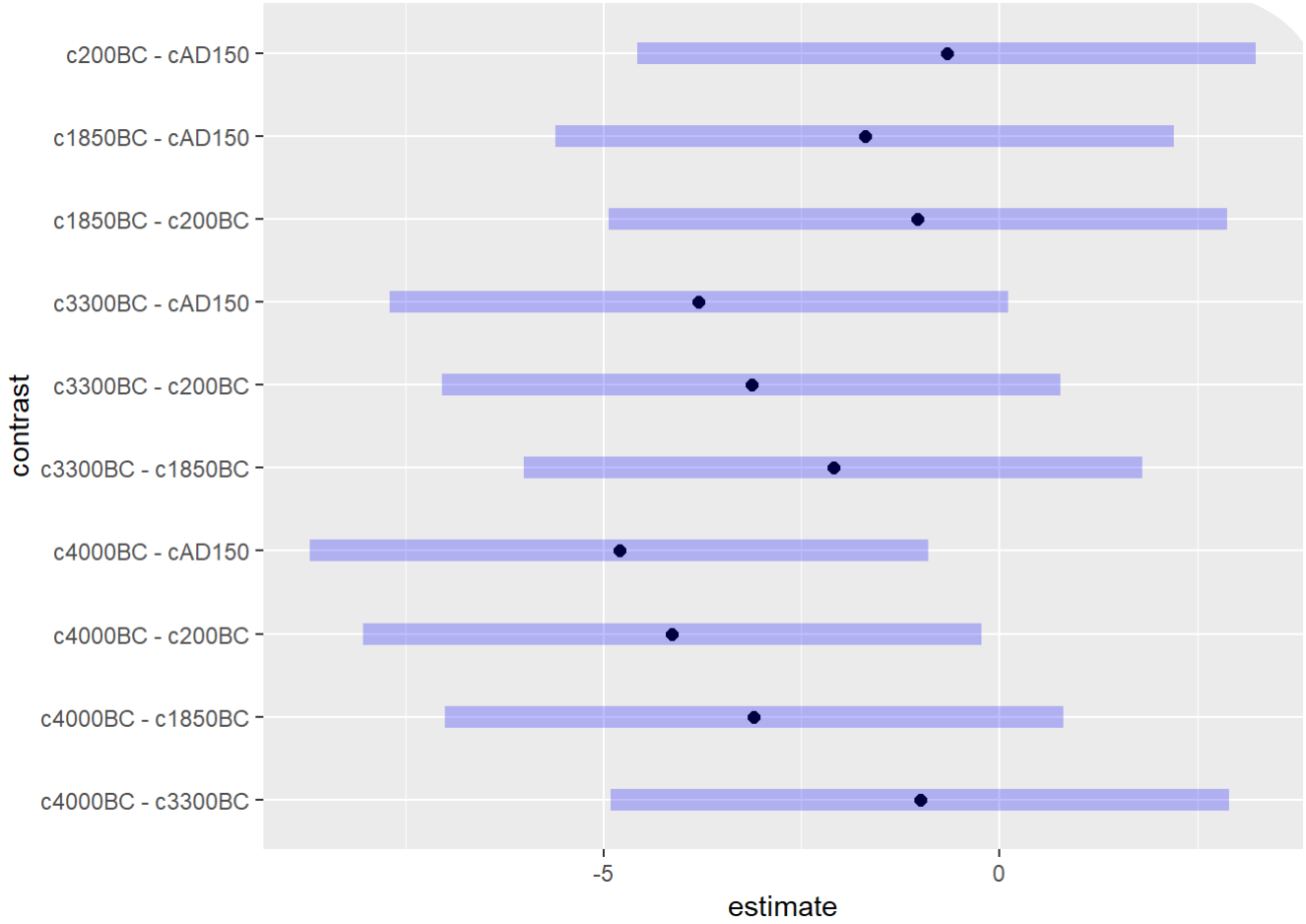
```
# number of groups
g <- 5

# old significance level
alpha <- 0.05

# number of comparison
nc <- p * g * (g-1) / 2

# new significance level
alphanew <- 0.05 / nc
```

A plot of pair-wise differences for the `mb` variable.

```
# obtain the contrasts first
cont <- contrast(pair.lst$mb, "pairwise")
# pair-wise differences for `mb`
bb <- confint(cont, level=1-alphanew, adj="none")
bb
```

```
##  contrast          estimate   SE  df lower.CL upper.CL
##  c4000BC - c3300BC   -1.000 1.19 145    -4.91    2.905
##  c4000BC - c1850BC   -3.100 1.19 145    -7.01    0.805
##  c4000BC - c200BC    -4.133 1.19 145    -8.04   -0.228
##  c4000BC - cAD150    -4.800 1.19 145    -8.71   -0.895
##  c3300BC - c1850BC   -2.100 1.19 145    -6.01    1.805
##  c3300BC - c200BC    -3.133 1.19 145    -7.04    0.772
##  c3300BC - cAD150    -3.800 1.19 145    -7.71    0.105
##  c1850BC - c200BC    -1.033 1.19 145    -4.94    2.872
##  c1850BC - cAD150    -1.700 1.19 145    -5.61    2.205
##  c200BC - cAD150     -0.667 1.19 145    -4.57    3.239
##
## Results are averaged over the levels of: rep.meas
## Confidence level used: 0.99875
```

```
plot(bb)
```



The intervals not containing zero indicate a significant difference between the corresponding group means. It seems, for the `mb` variable, means between epochs `c4000BC` and `c200BC`, and between `c4000BC` and `cAD150` are significantly different from zero.

Main page: **ST 437/537: Applied Multivariate and Longitudinal Data Analysis (https://maityst537.wordpress.ncsu.edu/)**