Lesson 1.3. Types of Sums of Squares

ANOVA is a statistical process for analysing the amount of variance that is contributed to a sample by different factors. It was initially derived by R. A. Fisher in 1925, for the case of balanced data (equal numbers of observations for each level of a factor).

When data is unbalanced, there are different ways to calculate the sums of squares for ANOVA. There are at least 3 approaches, commonly called Type I, II and III sums of squares (this notation seems to have been introduced into the statistics world from the SAS package but is now widespread). Which type to use has led to an ongoing controversy in the field of statistics (see Heer for an overview). However, it essentially comes down to testing different hypotheses about the data.

Consider a model that includes two factors A and B; there are therefore two main effects, and an interaction, AB. The full model is represented by SS(A, B, AB).

Other models are represented similarly: SS(A, B) indicates the model with no interaction, SS(B, AB) indicates the model that does not account for effects from factor A, and so on.

The influence of particular factors (including interactions) can be tested by examining the differences between models. For example, to determine the presence of an interaction effect, an F-test of the models SS(A, B, AB) and the no-interaction model SS(A, B) would be carried out.

It is convenient to define incremental sums of squares to represent these differences. Let

$$
\begin{aligned}
SS(AB|A,B) &= SS(A,B,AB) - SS(A,B)\\
SS(A|B,AB) &= SS(A,B,AB) - SS(B,AB)\\
SS(B|A,AB) &= SS(A,B,AB) - SS(A,AB)\\
SS(A|B) &= SS(A,B) - SS(B)\\
SS(B|A) &= SS(A,B) - SS(A)
\end{aligned}
$$

The notation shows the incremental differences in sums of squares, for example $SS(AB|A,B)$ represents “the sum of squares for interaction after the main effects”, and $SS(A|B)$ is “the sum of squares for the A main effect after the B main effect and ignoring interactions”.

The different types of sums of squares then arise depending on the stage of model reduction at which they are carried out.

Type I SS (Sequential sum of squares)

This tests the main effect of factor A, followed by the main effect of factor B after the main effect of A, followed by the interaction effect AB after the main effects.

$$\begin{aligned} SS(A) &\text{ for factor A} \\ SS(B|A) &\text{ for factor B} \\ SS(AB|B,A) &\text{ for interaction AB} \end{aligned}$$

This tests the main effect of factor A, followed by the main effect of factor B after the main effect of A, followed by the interaction effect AB after the main effects. Because of the sequential nature and the fact that the two main factors are tested in a particular order, this type of sums of squares will give different results for unbalanced data depending on which main effect is considered first.

For unbalanced data, this approach tests for a difference in the weighted marginal means. In practical terms, this means that the results are dependent on the realized sample sizes, namely the proportions in the particular data set. In other words, it is testing the first factor without controlling for the other factor (for further discussion and a worked example, see Zahn)

Pros

1. Nice property: balanced or not, SS for all the effects add up to the total SS, a complete decomposition of the predicted sums of squares for the whole model. This is not generally true for any other type of sums of squares.

2. This approach may be appropriate for fully nested hierarchical models, for which there exists a natural ordering of the terms.

Cons

1. Order matters! Hypotheses depend on the order in which effects are specified. If you fit a 2-way ANOVA with two models, one with A then B, the other with B then A, not only can the type I SS for factor A be different under the two models, but there is NO certain way to predict whether the SS will go up or down when A comes second instead of first. This lack of invariance to order of entry into the model limits the usefulness of Type I sums of squares for testing hypotheses for certain designs.

2. Not appropriate for factorial designs

In R the *anova()* function produces Type I SS.

Example 1.3.1 (Balanced data)

Consider the following balanced data.

	Pesticide, A			
Variety, B	1	2	3	4
1	49	50	43	53
	39	55	38	48
2	55	67	53	85
	41	58	42	73
3	66	85	69	85
	68	92	62	99

```
balance.factorial <- read.csv("twofactorial_crd.csv")
out1 <- aov(Yield ~ Pesticide + Variety + Pesticide:Variety,
          data = balance.factorial)
anova(out1)
```

```
## Analysis of Variance Table
##
## Response: Yield
##                   Df Sum Sq Mean Sq F value    Pr(>F)
## Pesticide          3 2227.5  742.49 17.5563 0.0001098 ***
## Variety            2 3996.1 1998.04 47.2443 2.048e-06 ***
## Pesticide:Variety  6  456.9   76.15  1.8007 0.1816844
## Residuals         12  507.5   42.29
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
out2 <- aov(Yield ~ Variety + Pesticide +  Pesticide:Variety,
          data = balance.factorial)
anova(out2)
```

```
## Analysis of Variance Table
##
## Response: Yield
##                   Df Sum Sq Mean Sq F value    Pr(>F)
## Variety            2 3996.1 1998.04 47.2443 2.048e-06 ***
## Pesticide          3 2227.5  742.49 17.5563 0.0001098 ***
## Variety:Pesticide  6  456.9   76.15  1.8007 0.1816844
## Residuals         12  507.5   42.29
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example 1.3.2 (Unbalanced data)

Consider the following data.

Material Type	Temperature (°F)		
	15	70	125
1	130 155 74	34 40 80	 70 58
2	159 126	136 115	45
3	138 160	150 139	96

```r
library(tidyverse)
unbalance.factorial <- read.csv("example.csv")
unbalance.factorial <- unbalance.factorial %>%
  mutate(A = as.factor(Mat),
         B = as.factor(Temp))

out3 <- aov(Y ~ A + B + A:B,
           data = unbalance.factorial)
anova(out3)
```

```
## Analysis of Variance Table
##
## Response: Y
##           Df  Sum Sq Mean Sq F value   Pr(>F)
## A          2 10557.6  5278.8  8.1479 0.009558 **
## B          2 11086.8  5543.4  8.5564 0.008284 **
## A:B        4  5317.2  1329.3  2.0518 0.170293
## Residuals  9  5830.8   647.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
out4 <- aov(Y ~ B + A + A:B,
           data = unbalance.factorial)
anova(out4)
```

```
## Analysis of Variance Table
##
## Response: Y
##           Df  Sum Sq Mean Sq F value   Pr(>F)
## B          2 12049.2  6024.6  9.2991 0.006459 **
## A          2  9595.3  4797.6  7.4052 0.012550 *
## B:A        4  5317.2  1329.3  2.0518 0.170293
## Residuals  9  5830.8   647.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Sequential model comparisons

```
SS1.A <- anova(lm(Y ~ 1, data=unbalance.factorial),
               lm(Y ~ A, data=unbalance.factorial))
SS1.B <- anova(lm(Y ~ A, data=unbalance.factorial),
               lm(Y ~ A + B, data=unbalance.factorial))
SS1.AB <- anova(lm(Y ~ A + B, data=unbalance.factorial),
               lm(Y ~ A + B + A:B, data=unbalance.factorial))
print(c(SS1.A[2, "Sum of Sq"], SS1.B[2, "Sum of Sq"],SS1.AB[2, "Sum of Sq"]))
```

```
## [1] 10557.625 11086.821  5317.221
```

```
SST <- anova(lm(Y ~ 1, data=unbalance.factorial),
             lm(Y ~ A + B + A:B, data=unbalance.factorial))

#This is the Total Sum of Squares
SST[2, "Sum of Sq"]
```

```
## [1] 26961.67
```

```
#This calculates the sum of the individual effect sums of squares
sum(c(SS1.A[2, "Sum of Sq"], SS1.B[2, "Sum of Sq"],SS1.AB[2, "Sum of Sq"]))
```

```
## [1] 26961.67
```

These examples show two things:

1. For balanced data, the order of the effects in the model does not change the sums of squares of the main effects.

2. For unbalanced data, the order of the effects in the model changes the sums of squares of the main effects.

3. For unbalanced data, Type I SS is not recommended.

Type II SS (hierarchical or partially sequential)

This approach might be described as a conditional analysis. The Type II SS for a given term is defined as the reduction in the residual SS due to adding the term after all other terms have been included in the model except any terms that contain the effect being tested. In other words, main effects are not conditioned upon interaction terms that involve them. In the two-way crossed design, this amounts to fitting A given B, B given A and A×B given both A and B.

$$SS(A|B) \text{ for factor A}$$
$$SS(B|A) \text{ for factor B}$$

Note that no significant interaction is assumed (in other words, you should test for interaction first (SS(AB | A, B)) and only if AB is not significant, continue with the analysis for main effects).

Pros

1. conforms to the orinciple of marginality; that is, it tests main effects while controlling for other main effects, but not for the interaction term involving those effects
2. most powerful when there is no interaction
3. invariant to the order in which effects are entered into the model

Cons

1. assumes that the interaction effect is negligible. If there is significant interaction, the main effects reported by Type II can be misleading.
2. the total sum of squares is not equal to the sum of the effects sums of squares.

IN R we use the *Anova()* in the **car** package to generate Type II SS. This is illustrated in the following example.

Example 1.3.3

Let us use the same unbalanced data in *Example 1.3.2.

```
library(car)
Anova(out3, type="II")
```

```
## Anova Table (Type II tests)
##
## Response: Y
```

```
##             Sum Sq Df F value   Pr(>F)
## A          9595.3  2  7.4052 0.012550 *
## B         11086.8  2  8.5564 0.008284 **
## A:B        5317.2  4  2.0518 0.170293
## Residuals  5830.8  9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notice that these sums of squares are identical to those obtained using the Method of Fitting Constants in Lesson 1.2.

Model comparisons

```r
SS2.A <- anova(lm(Y ~ B, data=unbalance.factorial),
               lm(Y ~ A + B, data=unbalance.factorial))
SS2.B <- anova(lm(Y ~ A, data=unbalance.factorial),
               lm(Y ~ A + B, data=unbalance.factorial))
SS2.AB <- anova(lm(Y ~ A + B, data=unbalance.factorial),
               lm(Y ~ A + B + A:B, data=unbalance.factorial))
print(c(SS2.A[2, "Sum of Sq"], SS2.B[2, "Sum of Sq"],SS2.AB[2, "Sum of Sq"]))
```

```
## [1]  9595.268 11086.821  5317.221
```

```r
sum(c(SS2.A[2, "Sum of Sq"], SS2.B[2, "Sum of Sq"],SS2.AB[2, "Sum of Sq"]))
```

```
## [1] 25999.31
```

Type III SS (marginal or orthogonal)

This type tests for the presence of a main effect after the other main effect and interaction. This approach is therefore valid in the presence of significant interactions.

$$SS(A|B, AB) \text{ for factor A}$$
$$SS(B|A, AB) \text{ for factor B}$$

Pros

1. Type III is designed specifically to handle unequal sample sizes across groups. It treats each "cell" in your experiment as equally important, regardless of how many participants are in it.

2. It calculates the effect of Factor A after accounting for Factor B and the Interaction (A:B). This provides a "pure" test of the unique contribution of that factor to the model.

3. It tests the main effect of a factor by averaging the effects across the levels of the other factors. This is useful if you want an "overall" estimate that isn't biased by the fact that one group happened to have more data points than another.

Cons

1. Violation of the Principle of Marginality: Testing main effects in the presence of interaction is illogical.

2. Because Type III corrects for the interaction term, it can be less powerful than Type II if the interaction is actually zero. You are essentially "wasting" variance by correcting for a term that might just be noise.

Just like Type II, we will use the *Anova()* function in the **car** package to generate Type II sums of squares. But we must specify the contrasts option to obtain sensible results. This is illustrated as follows.

Example 1.3.4

Let us use the same unbalanced data.

```
Anova(lm(Y ~ A + B + A:B,
         data=unbalance.factorial,
         contrasts=list(A=contr.sum, B=contr.sum)),
      type="III")
```

```
## Anova Table (Type III tests)
##
## Response: Y
##             Sum Sq Df  F value    Pr(>F)
## (Intercept) 170111  1 262.5693 5.759e-08 ***
## A             7700  2   5.9428  0.022635 *
## B            11237  2   8.6725  0.007961 **
## A:B           5317  4   2.0518  0.170293
## Residuals     5831  9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Final takeaways

1. For balanced data the three types of sums of squares will give identical results.

2. For unbalanced data with no significant interaction, use Type II SS.
3. Caution must be practiced in using Type II SS because it violates the principle of marginality.
4. In modern statistics, many researchers are moving away from the "Type I/II/III" debate by using Estimated Marginal Means (the emmeans package in R). This allows you to look at group differences regardless of the SS type used, often providing a clearer picture of what's actually happening in your data.