***Learning Statistics w/ R - University of Adelaide***

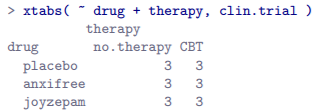
***Part V – Statistical Tools***

**16. Factorial ANOVA**

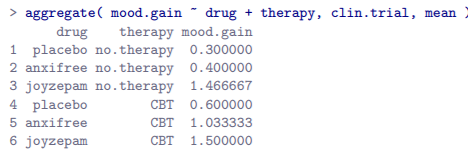
* Tools to compare 2 groups = most notably the **t-test** (Ch. 13) + then **analysis of variance (ANOVA)** as a method for comparing more than 2 groups (Ch. 14).
* Regression (Ch. 15) = somewhat different topic, but introduced a powerful new idea: *building statistical models that have MULTIPLE predictor variables used to explain a SINGLE outcome variable*
* A regression model could be used to predict # of errors a student makes in a reading comprehension test based on: 1) # of hours studied for the test + 2) their score on a standardized IQ test.
* Want to import this idea into the ANOVA framework
* Suppose we’re interested in using a reading comprehension test to measure student achievements in 3 different schools, + we also suspect girls + boys are developing at different rates (i.e. expected to have different performance on average).
* Each student is classified in 2 ways: gender + school 🡪 want to analyze scores in terms of both
* Tool for doing so = **factorial ANOVA** or **two-way ANOVA**, in contrast to the 1-way ANOVAs in Ch. 14

**16.1 Factorial ANOVA : Balanced designs, NO interactions**

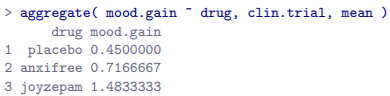
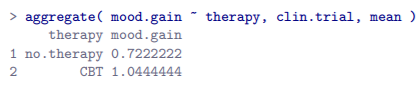
* W/ simple ANOVA, we assumed a fairly simple experimental design*: each person falls into 1 of several groups + we test if these groups have different means on some outcome variable*
* Broader class of experimental designs = **factorial design:** > 1 grouping variable.
* Ex: Effect of different drugs on mood experienced by each person
* When we did this, we found a significant effect of drug, but also ran an analysis to see if there was an effect of therapy + didn’t find one
* Something’s worrying about trying to run 2 separate analyses trying to predict the same outcome
* *Maybe there actually IS an effect of therapy on mood, but we couldn’t find it b/c it was being hidden by the effect of drug*
* Want to run a *single* analysis to include BOTH drug AND therapy as predictors.
* For this analysis, each person is **cross-classified** by drug given (factor w/ 3 levels) + therapy received (factor w/ 2 levels).
* Refer to this as a 3\*2 factorial design.
* Cross-tabulate drug by therapy using **xtabs()** + get the following table:



* Not only do we have participants corresponding to all possible combinations of the 2 factors, indicating our design is completely crossed, it turns out there are an equal # of people in each group = a **balanced design**.
* Balanced designs = simplest case.
* The story for *unbalanced* designs is quite tedious, so put it aside for the moment.
* Like one-way ANOVA, **factorial ANOVA** = a tool for testing certain types of hypotheses about *population means*.
* Sensible place to start = to be explicit about *what the hypotheses actually are.*
* However, before that point, it’s useful to have clean + simple notation to describe population means
* B/c observations are cross-classified in terms of 2 different factors, there are quite a lot of different means one might be interested in.
* Start by thinking about all different sample means we can calculate for this kind of design.
* Might be interested in this table of group means:



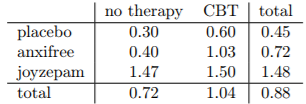
* See a cross-tabulation of group means for all possible combinations of the 2 factors (people w/ placebo + no therapy, people w/ placebo while getting CBT, etc.).
* However, can also construct tables that *ignore* 1 of the 2 factors.

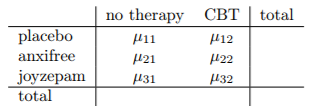
* If we can ignore 1 factor we can ignore both = calculating average mood gain across all 18 participants, regardless of drug or therapy



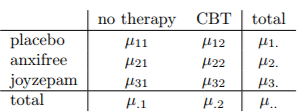
* At this point we have 12 different sample means to keep track of
* Organize all these numbers into a single table



* Each of these different means is a **sample statistic** = a quantity pertaining to specific observations made during a study.
* We *want* to make **inferences** about the corresponding **population parameters** = the TRUE means as they exist w/in some broader population.
* Those population means can also be organized into a similar table



* Each row = different level of Factor A (drug) + each column = level of Factor B (therapy)
* Let R = # of rows in the table + C = # of columns 🡺 refer to this as an **R\*C factorial ANOVA**.
* µ.rc = population mean associated w/ the r-th level of Factor A (row r) + the c-th level of Factor B (column c)
* Nice thing about subscript notation = it generalizes nicely
* If an experiment had involved a 3rd factor, we could just add a 3rd subscript.
* In principle, the notation extends to as many factors as you might care to include
* How to describe average mood gain across entire (hypothetical) population of people who might be given Joyzepam in an experiment like this, regardless of whether they were in CBT?
* Use **dot notation** to express this.
* In the case of Joyzepam: talking about a mean associated w/ the 3rd row in the table = averaging across 2 cell means (µ31 + µ32).
* Result = a **marginal mean** denoted µ3.
* Marginal mean for CBT = population mean associated w/ the 2nd column in the table, µ.2
* Grand mean = µ.. = mean obtained by averaging (**marginalizing**) over both.
* Technically, marginalizing != a regular mean, but = a **weighted average 🡪** *takes into account the frequency of the different events you’re averaging over.*
* However, w/ a balanced design, all cell frequencies are equal by definition, so the 2 are equivalent



* It is straightforward to formulate + express some hypotheses.
* Suppose goal = to find out 2 things
* 1) Does choice of drug have any effect on mood?
* 2) Does CBT have any effect on mood?
* These aren’t the only hypotheses we could formulate, but these are the 2 simplest to test
* 1st, if **drug** has no effect, we’d expect all row means, **µr.** to be identical = our null.
* On the other hand, if drug *does* matter, row means = different.
* Formally, we write down null + alternative in terms of the *equality of marginal means*:

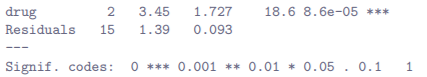


* These are the *exact same statistical hypotheses* formed w/ a one-way ANOVA but now w/ letters along w/ µ
* We’re talking about the same hypothesis, just the more-complicated ANOVA requires more careful notation due to the presence of multiple grouping variables
* Although hypothesis is identical, the *test* of that hypothesis is subtly different b/c we’re now acknowledging the existence of the SECOND grouping variable.
* 2nd hypothesis test is formulated the same way, but the null now corresponds to equality of *column* means:

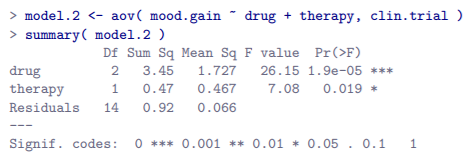


* These are basically the same as hypotheses testing in simpler one-way ANOVAs in Ch. 14.
* Expecting to see sums of squares (SS), mean squares (MS), degrees of freedom (df), + finally an F-statistic we can convert into a p-value
* If data you’re trying to analyze correspond to a balanced factorial design, running ANOVA is easy.
* Start by reproducing original analysis from Ch. 14 (single factor (drug) to predict outcome (mood)





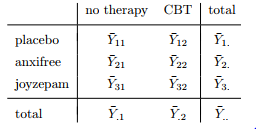
* Suppose we’re also curious to find out if **therapy** has a relationship to mood.
* All we have to do is extend the formula 🡺 specify mood.gain ~ drug + therapy



* Output = pretty simple to read 🡪 1st row = **between-group sum of squares (SS)** associated w/ **drug** along w/ a corresponding **between-group df**.
* It also calculates a **mean square value (MS),** **F-statistic**, a **p-value**.
* There is also a rows corresponding to **therapy** + to the **residuals** (**w/in groups variation**).
* Not only are all individual quantities familiar, the relationships between these different quantities has remained unchanged:
* Just like w/ one-way ANOVA, the **mean square** value = dividing SS by corresponding df regardless of talking about drug, therapy or residuals.
* Take it on faith that R calculates SS values correctly + try to verify all the rest of the #’s make sense.
* Note for drug, we divide 3.45 / 2, + end up w/ a **mean square** = 1.73 + for therapy, there’s only 1 dF, so 0.47 (SS) / 1 = 0.47 (MS).
* For F- statistics + p values, notice we have 2 of each: 1 corresponding to drug + 1 to therapy
* Regardless of factor, F-statistic = dividing MS value associated w/ the factor by the MS value associated w/ the residuals.
* A = 1st factor (drug) + R = residuals, then F-statistic associated w/ factor A = F.A is calculated as follows: **FA = MS.A / MS.R** + an equivalent formula exists for factor B (therapy).
* For **drugs**, take MS = 1.73 / residual MS 0.07 🡪 F-statistic = 26.15.
* Corresponding for therapy 🡪 divide 0.47 / 0.07 = 7.08 as the F-statistic.
* Last part of ANOVA table = calculation of p-values
* For each of our 2 factors, we’re trying to test the null that there is NO relationship between that factor + outcome
* To that end, we’ve (apparently) followed a similar strategy to one-way ANOVA + have calculated an F-statistic for each hypotheses.
* To convert these to p-values, must note the sampling distribution for the F-statistic under the null (*factor in question is irrelevant*) is an **F distribution** + that the 2 dF values are those corresponding to the factor + those corresponding to the residuals.
* For drug = F-distribution w/ 2 + 14 Df + for therapy, sampling distribution is F w/ 1 + 14 dF
* If we really wanted to, could calculate p-value using pf()



* This is indeed the p value reported in the ANOVA table above.
* ANOVA table for a more-complicated analysis should be read in much the same way as the ANOVA table for a simpler analysis
* In short, it’s telling us the factorial ANOVA for our 3\*2 design found a significant effect of drug (F(2,14) = 26.15, p < .001) as well as a significant effect of therapy (F(1,14) = 7.08, p < .02).
* More technically correct terminology 🡪 “*There are 2* ***main effects*** *of drug + therapy*”
* Probably seems redundant to refer to these as **main effects**, but it makes sense.
* Later on, we talk about the possibility of interactions between the 2 factors, + so we make a distinction between **main effects** + **interaction effects**.
* It’s genuinely true that factorial ANOVA = built in more or less the same way as a simpler one-way ANOVA model, but this feeling of familiarity starts to evaporate in the details.
* Remember, the hypothesis tests for the **main effects** (drug + therapy) are F-tests
* Assume for now we have only 2 predictor variables, Factor A + Factor B.
* If Y = outcome, use Y(r,c,i) = outcome associated w/ the i-th member of group (r,c) (level/row r for Factor A + level/column c for Factor B).
* Thus, if Y¯ = a sample mean, can use the same notation as before to refer to group means, marginal means + grand means:
* Y¯rc = sample mean associated w/ the r-th level of Factor A + the c-th level of Factor B,
* Y¯r. = marginal mean for the r-th level of Factor A
* Y¯.c = marginal mean for the c-th level of Factor B
* Y¯ .. = grand mean.
* In other words, our sample means can be organized into the same table as the population means.



* Drug has 3 levels + therapy has 2 levels, + so we’re trying to run a 3\*2 factorial ANOVA.
* Factor A (row factor) has R levels + Factor B (column factor) has C levels = R\*C factorial ANOVA.
* Now compute the sum of squares values for each of the 2 factors in a relatively familiar way.
* Factor A 🡪 between-group sum of squares = calculated by assessing the extent to which the (row) marginal means Y¯ 1. , Y¯ 2. etc., are different from the grand mean Y¯.. 🡺 calculate the sum of squared difference between Y¯i. values + Y¯.. values.
* Specifically, if there are N people in each group, calculate:



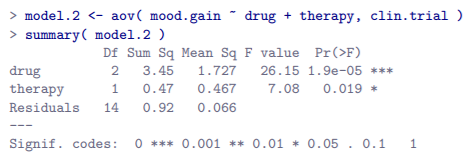
* As w/ one-way ANOVA, the most interesting part of this formula is



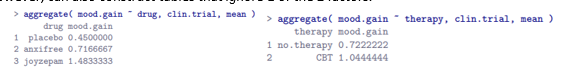
* This corresponds to the squared deviation associated w/ level r.
* Calculates this squared deviation for all R levels of the factor, adds them up, + then multiplies the result by N \* C.
* This last part = there are multiple cells in our design that have level r on Factor A + in fact, there are C of them, 1 corresponding to each possible level of Factor B
* There are 2 different cells in the design corresponding to anxifree: for people w/ no.therapy, + for CBT group.
* Not only that, w/in each of these cells there are N observations.
* So, to convert our SS value into a quantity that calculates the between-groups sum of squares on a *per observation* basis, we have to multiply by N \* C.
* The formula for factor B is of course the same thing, just w/ some subscripts shuffled around:



* Now that we have these formulas, we can check them against the R output from earlier



* We calculated all marginal means (row marginal means Y¯r. + column marginal means Y¯.c)



* And also calculated the grand mean



* Repeat those calculations,



* Calculate the sum of squares associated w/ the main effect of drug for the total of N = 3 people in each group, + C = 2 different types of therapy.
* Or, to put it another way, there are 3\*2 = 6 people who received any particular drug.



* Repeat the same kind of calculation for the main effect of therapy, again w/ N = 3 people in each group, but w/ R = 3 different drugs 🡺 3\*3 = 9 people who received CBT + 9 who received a placebo



* These SS values are analogous to the between-group sum of squares values from one-way ANOVA
* However, it’s *not a good idea to think of them as between-groups SS values anymore b/c we have 2 different grouping variables + it’s easy to get confused.*
* For an F test 🡪 also need to calculate **w/in-groups sum of squares**.
* Can refer to **w/in-groups SS** as the **residual sum of squares, SSres**.
* Easiest way to think about the residual SS values in *this* context = leftover variation in the outcome variable after taking into account the differences in the marginal means (after removing SS.A + SS.B)
* Start by calculating **total sum of squares** in the same way as for one-way ANOVA = take the difference between each observation Y(r, c, i) + the grand mean Y¯.., square the differences, + add them all up



* Triple summation = looks more complicated than it is.
* In the 1st 2 = summing across all levels of Factor A/possible rows, across all levels of Factor B/all possible columns c)
* Each r, c combination corresponds to a *single group*, + each group contains N people, so we have to sum across all those people (all i values) too.
* In other words, we’re summing across all observations in the data set/possible r, c, I combos
* At this point, we know the total variability of the outcome variable, SS.T, + know how much of that variability can be attributed to Factor A (SS.A) + to Factor B (SS.B)
* The **residual sum of squares** is thus defined to be the variability in Y that CAN’T be attributed to either of our 2 factors.



* Whole point of calling it a residual = it’s the *leftover* variation
* Common to refer to **SS.A + SS.B** as the variance attributable to the ANOVA model**, SS.M**,
* Often say **total sum of squares = the model sum of squares + the residual sum of squares**
* Not a surface similarity: ANOVA + regression are actually the same thing under the hood
* To get SSR, 1st calculate total sum of squares:



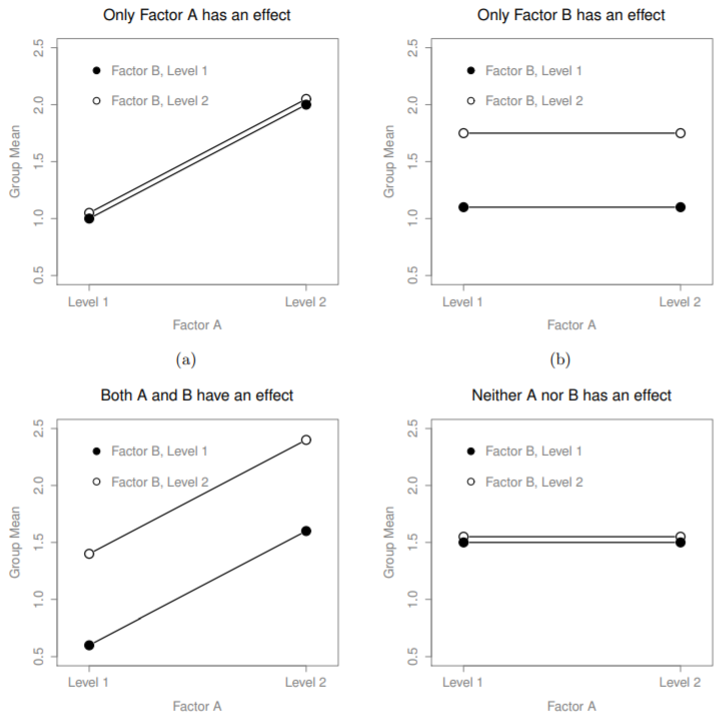
* Then we use it to calculate residual sum of squares (see its same as in ANOVA table)

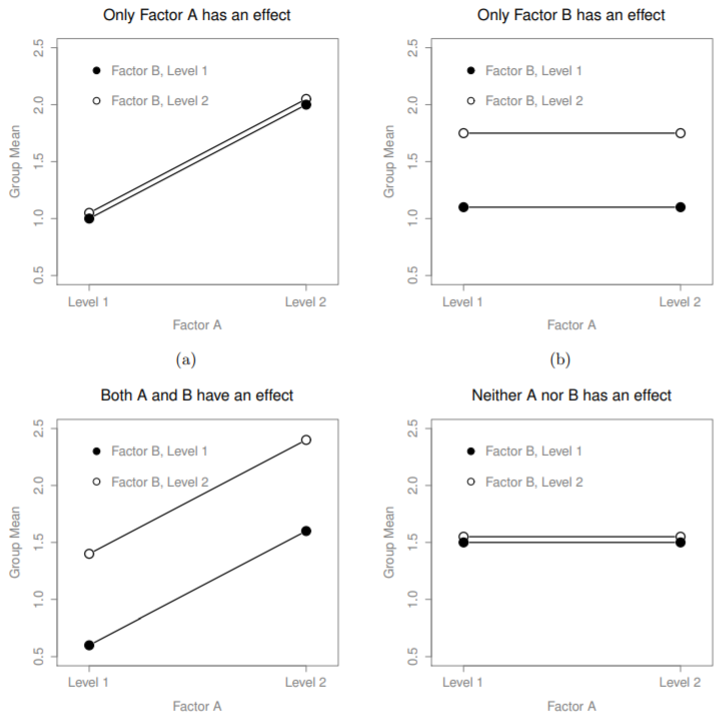


* The dF are calculated in much the same way as for one-way ANOVA.
* For any given factor, dF = # of levels minus 1 (R - 1 for row variable/Factor A, C - 1 for column variable/Factor B).
* So, for drug, df = 2, + for therapy df = 1.
* For drug, we observe 3 separate group means, which are constrained by 1 grand mean so therefore the dF = 2.
* For the residuals, the logic is similar, but not quite the same 🡪 total # of observations in our experiment = 18 + the constraints correspond to the 1 grand mean, the 2 additional group means drug introduces, + the 1 additional group mean therapy introduces, + so dF = 14.
* As a formula, this is  = 
* One-way ANOVA looked to see if there are any differences between drugs + a 2nd one-way ANOVA checked if there were any differences between therapies.
* Null + alternatives tested by the 1-way ANOVAs = identical to those tested by the factorial ANOVA
* Looking even more carefully at the ANOVA tables, the sum of squares associated w/ the factors are identical in the 2 different analyses (drug = 3.45 + therapy = 0.92), as are dF (drug = 2, therapy = 1)
* *But they don’t give the same answers*
* Most notably, the one-way ANOVA for therapy showed no significant effect (p-value = 0.21).
* However, main effect of therapy in the two-way ANOVA DID get a significant effect (p = .019)
* The 2 analyses are clearly not the same b/c of how the residuals are calculated.
* The whole idea behind an F-test = to compare the variability that can be attributed to a particular factor w/ the variability that CANNOT be accounted for (the residuals).
* A one-way ANOVA for therapy ignores the effect of drug, so **ANOVA dumps all the drug-induced variability into the residuals**
* This makes the data look more noisy than they really are, + the effect of therapy *which was correctly found to be significant in the two-way ANOVA* now becomes non-significant.
* If we ignore something that *actually matters* (drug) when trying to assess the contribution of something else (therapy), our analysis will be distorted.
* Perfectly okay to ignore variables that’re genuinely irrelevant to a phenomenon of interest
* Ex: color of the walls turned out to be non-significant in a three-way ANOVA (mood.gain ~ drug + therapy + wall.colour) = perfectly okay to disregard it
* **What you shouldn’t do is drop variables that actually make a difference**

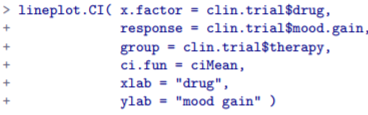
**16.2 Factorial ANOVA 2: balanced designs, interactions allowed**

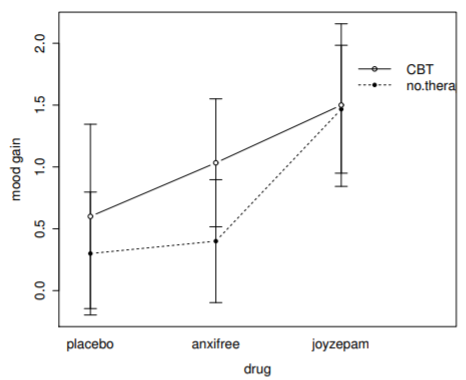
* The ANOVA model we’ve been talking about so far covers a range of different patterns we might observe in data.
* In a two-way ANOVA design, there are 4 possibilities:
* (a) only Factor A matters
* (b) only Factor B matters
* (c) both A + B matter
* (d) neither A nor B matters.



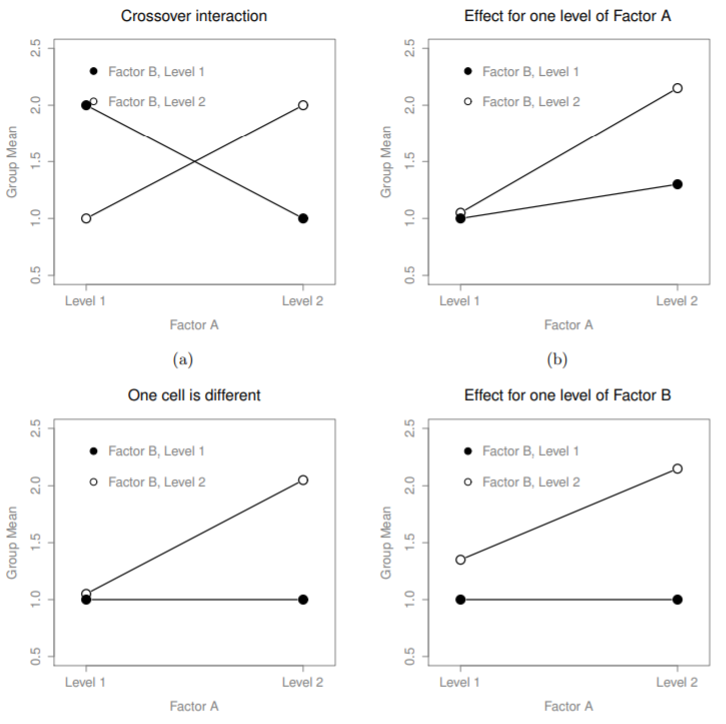


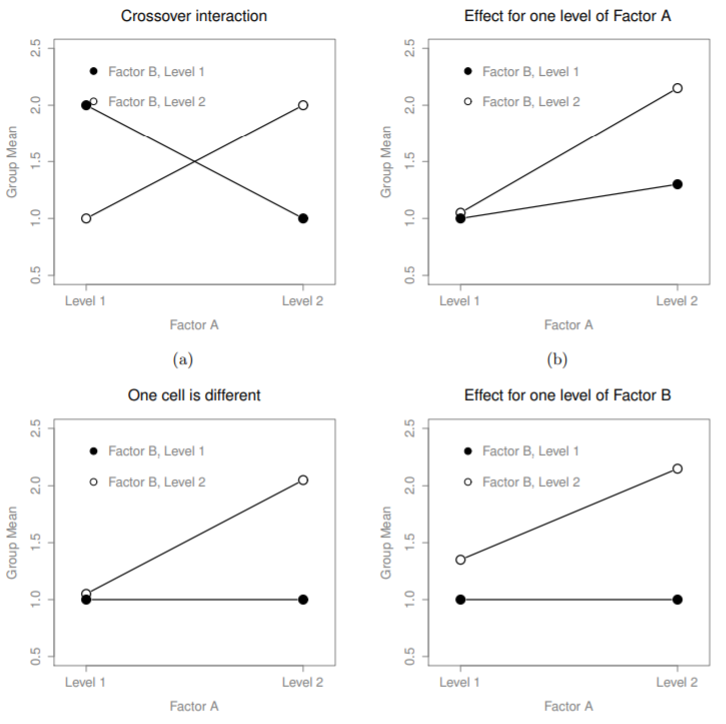
* These 4 patterns of data are all quite realistic: there are many data sets that produce exactly those patterns.
* However, they are not the whole story, + the ANOVA model we’ve been talking about so far is not sufficient to fully account for a table of group means.
* So far, we’ve talked about the idea that both drugs + therapy can influence mood, but no way of talking about the possibility of an *interaction* between the 2.
* An **interaction between A + B** is said to occur *whenever the effect of Factor A is different, depending on the level of Factor B*.
* Suppose operation of Anxifree + Joyzepam is governed quite different physiological mechanisms, + 1 consequence of this = Joyzepam has more or less the same effect on mood regardless of whether one is in therapy, Anxifree = much more effective when administered in conjunction w/ CBT
* *The ANOVA developed prior does not capture this idea*.
* To get some idea of whether an interaction is actually happening here, it helps to plot the various group means, say w/ **interaction.plot**(), which won’t draw error bars for you.
* To include error bars, use **lineplot.CI**() in the **sciplots** package
* don’t forget **ciMean**() is in **lsr** package, so you need to have lsr loaded





* Our main concern = the fact that the 2 lines aren’t parallel
* Effect of CBT (difference between solid + dotted lines) when drug = Joyzepam appears to be ~0, even smaller than the effect of CBT when a placebo is used
* However, w/ Anxifree, the effect of CBT is larger than w/ placebo
* Is this effect *real*, or is this just random variation due to chance?
* Our original ANOVA cannot answer this question, b/c we make no allowances for the idea interactions even exist
* Key idea = an **interaction effect**
* **mood.gain ~ drug + therapy + drug:therapy** = there’re only 2 factors involved in a model (drug + therapy), there are 3 distinct terms (drug, therapy + drug:therapy).
* In addition to the main effects of drug + therapy, we have a new component to the model, the **interaction term** drug:therapy
* Idea behind an interaction effect = *effect of Factor A is different depending on the level of Factor B*
* But what does that mean in terms of data?

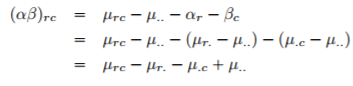




* Here are several patterns that, although different, all count as an interaction effect.
* It’s not entirely straightforward to translate this qualitative idea into something mathematical a statistician can work w/.
* As a consequence, the idea of an interaction effect being formalized in terms of null + alternative hypotheses is slightly difficult
* Basic idea = Be a little more explicit about our main effects.
* Consider main effect of Factor A (drug), originally formulated in terms of the null that the 2 marginal means **µr.** are all equal to each other.
* Obviously, if all these are equal to each other, they must also be equal to the grand mean µ.. as well,
* So we define the effect of Factor A at level r = the difference between the marginal mean µr. + the grand mean µ... 🡪 **αr** = µr. - µ..
* By definition, all αr values must sum to 0, for the same reason the average of the marginal means µr. must be = grand mean µ...
* Can similarly define main effect of Factor B at level i to be the difference between the column marginal mean **µ.c** + the grand mean **µ..** 🡪 βc = µ.c - µ..
* Once again, βc values must sum to 0.
* αr + βc values allows one to be precise about what it means to say that there is no interaction effect.
* If there is NO interaction at all, αr + βc will perfectly describe the group means µ(r, c).
* Specifically, it means µ(r, c) = µ.. + αr + βc
* That is, *there’s nothing special about the group means that you couldn’t predict perfectly by knowing all the marginal means* 🡺 that’s our null
* The alternative is that µ(r, c) != µ.. + αr + βc for *at* *least 1 group* (r, c)
* Statisticians usually define the specific interaction associated w/ group (r, c) to be some number, αβ(r, c), so the alternative is that **µ(r, c) = µ.. + αr + βc + αβ(r, c)** where *αβ(r, c) is non-zero* for at least 1 group.
* To calculate the **sum of squares for the interaction terms**, **SS.A:B**, 1st notice how the interaction effect is defined in terms of *the extent to which the actual group means differ from what you’d expect by just looking at the marginal means.*
* The previous refer to population parameters rather than sample statistics = don’t actually know what they are 🡪 can estimate them using sample means in place of population means.
* For Factor A, a good way to estimate the main effect at level r = difference between the sample marginal mean Y¯(r, c) + the sample grand mean Y¯..
* We’d use this as our estimate of the effect 🡪 
* Similarly, estimate of the main effect of Factor B at level c can be defined as follows: 
* Going back to the formulas used to describe the SS values for the 2 main effects, notice these effect terms are exactly the quantities we were squaring + summing

* What’s the analog of this for interaction terms? 🡪 1st rearrange the formula for the group means µ(r, c) under the alternative:



* Then substitute sample statistics in place of population means to get an estimate of the interaction effect for group (r, c),



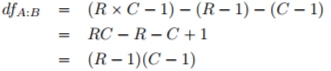
* Now all we have to do is sum all these estimates across all R levels of Factor A + all C levels of Factor B to get the formula for the **sum of squares associated w/ the interaction** as a whole:



* We multiply by N b/c there are N observations in each group, + we want SS values to reflect the variation *among observations* accounted for by the interaction, NOT the variation among *groups*.
* Now, w/ a formula for calculating SS.A:B, recognize that the interaction term is part of the model, so the total sum of squares associated w/ the model, SS.M = the sum of the 3 relevant SS values,
* The residual sum of squares SS.R is still “the leftover variation”, namely SST - SSM, but w/ the interaction term this becomes:



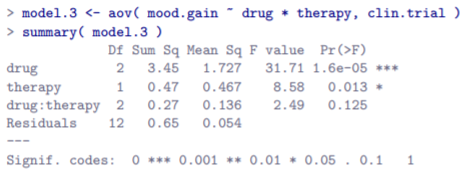
* As a consequence, SS.R will be smaller than in our original ANOVA that didn’t include interactions.
* Calculating interaction dF is, again, slightly trickier than corresponding calculation for main effects.
* To start, think about the ANOVA model as a whole.
* Once we include interaction effects in the model, we’re allowing every single group a unique mean, µ(r, c)
* For an R\*C factorial ANOVA, this means there are R\*C quantities of interest in the model, + only the 1 constraint: *all group means need to average out to the grand mean.*
* i.e. the model as a whole needs to have (R\*C) - 1 dF.
* But main effect of Factor A has R - 1 dF + main effect of Factor B has C - 1 dF, which means the dF associated w/ the interaction is



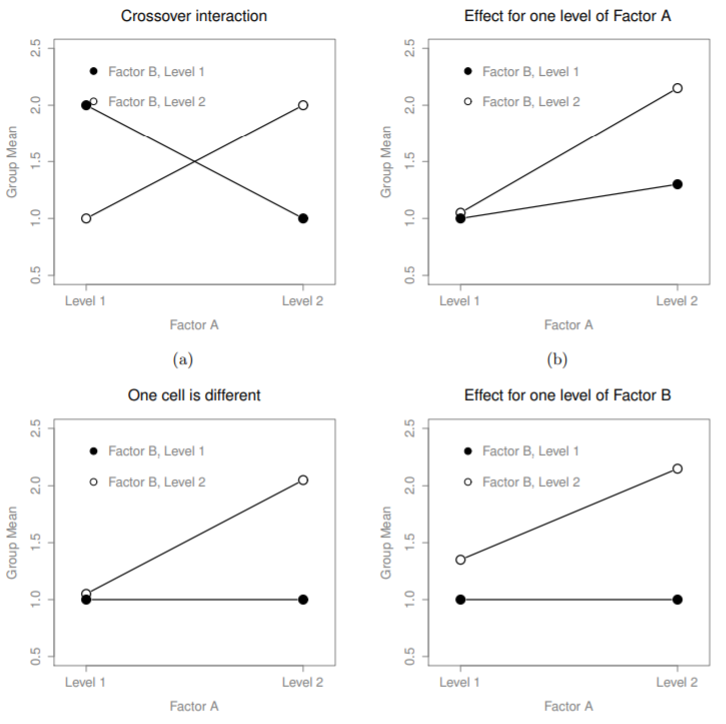
* This is the product of the dF associated w/ the row factor + the dF associated w/ the column factor
* B/c we’ve added interaction terms, which absorb some dF, there are fewer residual dF left over.
* Specifically, note if the model w/ interaction has a total of (R\*C) – 1 dF, + there are N observations in the data set constrained to satisfy 1 grand mean, then the residual dF now become N - (R\*C) – 1 + 1 = **N - (R\*C)**
* Adding interaction terms to the ANOVA model in R is straightforward 🡪 include the interaction term drug:therapy.



OR



* While we DO have a significant main effect of both drug (F(2,12) = 31.7, p < .001) + therapy (F(1,12) = 8.6, p = .013), *there is no significant interaction between the 2* (F(2,12) = 2.5, p = 0.125).
* There’s a couple of important things to consider when interpreting the results of factorial ANOVA
* 1st, the same issue w/ one-way ANOVA 🡪 if you obtain a significant main effect of (say) drug, it *doesn’t tell you anything about which drugs are different to one another*.
* To find that out, you need to run additional analyses (Sections 16.7 + 16.8)
* The same is true for interaction effects: knowing there’s a significant interaction doesn’t tell you anything about what kind of interaction exists.
* Again, you’ll need to run additional analyses.
* Secondly, there’s a very peculiar interpretation issue that arises when you *obtain a significant interaction effect but no corresponding main effect.*
* This happens sometimes 🡪 like in the crossover interaction



* In this case, neither main effect would be significant, but the *interaction effect* would be.
* This is difficult to interpret
* General advice = don’t pay much attention to the main effects when an interaction is present
* Although the tests of the main effects are perfectly valid from a mathematical point of view, when there is a significant interaction effect, the main effects rarely test interesting hypotheses.
* Recall that the null for a main effect = the marginal means are equal to each other, + recall that a marginal mean is formed by averaging across several different groups.
* But w/ a significant interaction effect, you know the groups that comprise the marginal mean aren’t homogeneous, so it’s not obvious why you’d even care about those marginal means.
* Suppose we had a 2\*2 design comparing 2 different treatments for phobias (systematic desensitization + flooding), + 2 different anxiety-reducing drugs (Anxifree + Joyzepam).
* Suppose we found Anxifree had no effect w/ desensitization, + Joyzepam had no effect w/ flooding, but both were effective for the other treatment.
* This is a classic crossover interaction + the ANOVA would show no main effect of drug, but a significant interaction.
* To say there’s “no main effect” means if we average over the 2 different treatments, the average effect of Anxifree + Joyzepam is the same.
* This is important b/c when treating someone for phobias, it is never the case a person can be treated using an *average* of flooding + desensitization 🡪 doesn’t make sense.
* For 1 treatment, 1 drug is effective; + for the other treatment, the other drug is effective = The **interaction** is the important thing + the main effect is irrelevant.
* This sort of thing happens a lot: the **main effect tests** = tests of marginal means, + when an interaction is present we often find ourselves not being interested in marginal means b/c they imply averaging over things that the interaction tells us shouldn’t be averaged
* **It’s not always the case that a main effect is meaningless when an interaction is present**.
* Often you can get a big main effect + a very small interaction, in which case you can still say things like “drug A is generally more effective than drug B”, b/c there was a big effect of drug A, but you’d need to modify it by adding that the difference in effectiveness was different for different psychological treatments.
* In any case, the main point = whenever you get a significant interaction, stop + think about what the main effect actually means in this context.
* Don’t automatically assume that the main effect is interesting.

**16.3 Effect size, estimated means, + confidence intervals**

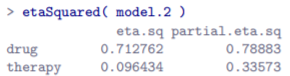
* The main thing you will probably want to calculate for a factorial ANOVA = **effect size** for each term in a model
* May also want to R to give you some estimates for the group means + associated CI’s.
* **Effect size** calculations for a factorial ANOVA = similar to those used in one-way ANOVA (Ch. 14.4)
* Specifically, we can use **η2 (eta-squared)** as simple way to measure how big the overall effect is for any particular term.
* η2 = dividing the sum of squares associated w/ a term by the total sum of squares.
* To determine size of main effect of Factor A:



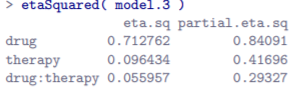
* This can be interpreted in much the same way as R2 in regression 🡪 tells you the proportion of variance in the outcome variable that can be accounted for by the main effect of Factor A.
* It therefore ranges from 0 (no effect at all) to 1 (accounts for all variability in the outcome)
* Moreover, the sum of all η2 values, taken across all terms in a model, will sum to the total R2 for the ANOVA model.
* If the ANOVA model fits *perfectly* (no w/in-groups variability at all), the η2 values sum to 1.
* Rarely, if ever, happens in real life
* A 2nd measure of effect = **partial** **η2** = **p.η2** 🡪 when measuring effect size for a particular term, you want to *deliberately ignore the other factor effects in the model*
* i.e. pretend the effect of all these other terms is zero
* Then we calculate what the η2 value *would have been*.
* To do so, remove the sum of squares associated w/ the other terms from the denominator.
* Ex: p.η2 for main effect of Factor A 🡪 denominator = sum of SS values for Factor A + the residuals



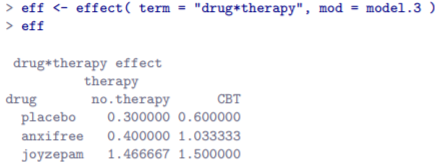
* This always gives a larger number than η2
* Once again, you get a number between 0 + 1, where 0 = no effect.
* However, it’s slightly trickier to interpret what a large partial η2 value means.
* Can’t actually compare the partial η2 values across terms
* Suppose there is no w/in-groups variability at all 🡪 If so, SSR = 0.
* This means every term has a partial η2 = 1
* **But that doesn’t mean all terms in the model are equally important, or that they are equally large**.
* All it means = all terms in the model have effect sizes large *relative to the residual variation*.
* It is NOT comparable across terms.
* Ex: 1st look at the effect sizes for the original ANOVA *w/out the interaction term*



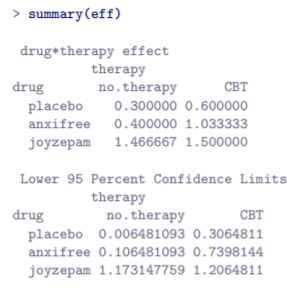
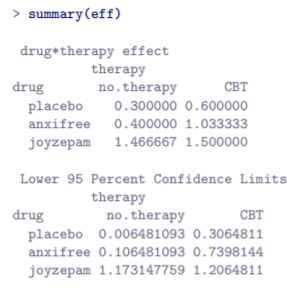
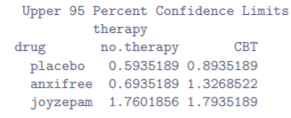
* Looking at η2 first 🡪 see that drug accounts for 71.3% of the variance in mood.gain, whereas therapy only accounts for 9.6%, leaving a total of 19.1% of the variation unaccounted for (residuals)
* Overall, this implies we have a very large effect of drug (too large for real-life, actually) + a modest effect of therapy.
* Looking at partial η2, b/c the effect of therapy isn’t all that large, controlling for it doesn’t make much of a difference = partial η2 for drug doesn’t increase very much
* In contrast, b/c the effect of drug was very large, controlling for it DOES make a big difference, + so when we calculate partial η2 for therapy, it rises up a good amount
* But what does these partial η2 values *actually mean*?
* General interpretation of partial η2 for the main effect of Factor A = as a statement about a hypothetical experiment in which *only* Factor A was being varied.
* partial η2 tells you how much *variance in the outcome variable you’d expect to see accounted for in that experiment w/ only Factor A being varied.*
* However, this interpretation (like many things associated w/ main effects) doesn’t make sense w/ a large + significant interaction effect.
* Speaking of interaction effects, calculate effect sizes for a model that includes the interaction term.



* η2 values for the main effects don’t change, but the *partial* η2 values do
* In many situations you will find yourself wanting to report **estimates of all group means** based on the results of an ANOVA, as well as CI’s associated w/ them.
* You can use **effect**() in the **effects** package to do this
* If the ANOVA you have is a **saturated model** (contains all possible main effects + all possible interaction effects), the estimates of the group means are identical to the sample means, though the CI’s will use a **pooled estimate of the standard errors**, rather than a separate one for each group

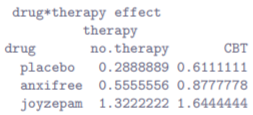


* Notice these are the same numbers we got when computing the sample means earlier (group.means computed using aggregate())
* 1 useful thing we can do using the effect variable **eff** is extract the CI’s using summary()

* The estimated mean mood gain for the placebo group w/ no therapy = 0.300, w/ a 95% CI from 0.006 to 0.594.
* \*\*\*Note: These are NOT the same CIs you’d get if you calculated them separately for each group, b/c of the fact that the ANOVA model assumes homoscedasticity + therefore uses a **pooled estimate of the standard deviation**.
* *When the model doesn’t contain the interaction term, the estimated group means will be different from the sample means.*
* Instead of reporting the sample mean, effect() will calculate the group means that would be expected on the basis of the marginal means (i.e., assuming no interaction).
* The estimate reported for the mean for level r on the (row) Factor A + level c on the (column) Factor B, **µ(r, c),** would be = **µ.. + αr + βc**
* If there are genuinely no interactions between the 2 factors, this is actually a better estimate of the population mean than the raw sample mean would be.
* Command to obtain these estimates = identical to the last, except w/ model.2 (no interaction term)





* \*\*\*R gives an error message letting you know the resulting estimates are based on the assumption that no interactions exist.
* Makes sense it would do this 🡪 W/ drug\*therapy as our input, we’re telling R we want it to output the **estimated group means** (rather than marginal means), but the actual input drug\*therapy might mean you want interactions included, or you might not.
* No *actual* ambiguity here, b/c the model itself either does or doesn’t have interactions, a warning just makes sure you’ve specified the actual model you care about.
* Can use **Effect**() to ignore this error message
* As before, we can obtain CIs w/ **summary( eff )** but the output looks the same as last time

**16.4 Assumption checking**

* As w/ one-way ANOVA, the key assumptions of factorial ANOVA are **homoscedasticity** (all groups have the same SD), **normality of the residuals**, + **independence of the observations**.
* The 1st 2 can be tested for, the 3rd is something you need to assess by asking if there are any special relationships between different observations.
* Additionally, if NOT using a **saturated model** (omitted the interaction terms), you’re also assuming omitted terms aren’t important.
* Can check this new 4th one by running an ANOVA w/ omitted terms included + see if significant
* Homoscedasticity + normality of the residuals checks are no to one-way ANOVA checks
* **Levene test for homoscedasticity** (14.7) 🡪 leveneTest() assumes a saturated model (included all of relevant terms), b/c the test is *primarily concerned w/ the w/in-group variance*
* Doesn’t make a lot of sense to calculate this any way other than w/ respect to the *full* model.
* 

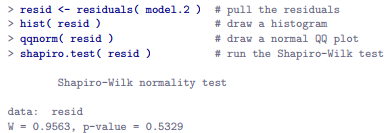


* Run Levene test w/ a saturated model





* The Levene test is non-significant = can safely assume homoscedasticity is not violated.
* Due to the way leveneTest() is implemented, however, if you use a formula like **mood.gain ~ drug + therapy + drug:therapy**, or input an ANOVA object based on a formula like this, you actually get the error message.
* Shouldn’t happen b/c this is a fully crossed model, however, there’s a quirky shortcut in the way leveneTest() checks whether a model is fully crossed that means it doesn’t recognize this as a fully crossed model.
* Essentially its checking that you used **\*** (ensures a model is fully crossed) + not “+” or “:” in the model formula.
* So if you’ve manually typed out all relevant terms for a fully crossed model, leveneTest() doesn’t detect it.
* As w/ one-way ANOVA, test for normality of residuals in a straightforward fashion (14.9)
* 1st use **residuals**() to extract the residuals from a model + examine them in a few different ways
* Generally a good idea to examine them *graphically* w/ + QQ plots (**qqnorm**())
* Formal test for normality of the residuals= **Shapiro-Wilk test** (**shapiro.test**())



* Non-significance of Shapiro-Wilk test = normality isn’t violated

**16.5 The F test as a model comparison**

* In the context of ANOVA, we’ve been referring to the **F-test** as a way of testing whether a particular term in the model (e.g., main effect of Factor A) is significant.
* This interpretation is perfectly valid, but not necessarily the most useful way to think about the test.
* In fact, it’s actually fairly limiting about what the F-test does.
* Suppose I want to see if there are any effects of *any kind* that involve **therapy** (don’t care if it’s a main or an interaction effect)
* 1 thing I could do is look at the output for **model.3** (DID see a main effect of therapy (p = .013), but did NOT see an interaction effect (p = .125))
* That’s *kind of* telling us what we want to know, but it’s not quite the same thing.
* What we *really* want = a single test that jointly checks main effect of therapy + the interaction effect
* Given the way I’ve been describing the ANOVA F-test up to this point, you’d be tempted to think that this isn’t possible.
* On the other hand, recall in regression (15.10), we were able to use F-tests to make comparisons between a wide variety of regression models.
* Something of that sort is possible w/ ANOVA
* \*\*\*Understand that the **F-test**, as used in both ANOVA + regression, is really a comparison of 2 statistical models.
* 1 of these models = the full model (alternative hypothesis), + the other model is a simpler model missing 1+ terms that the full model includes (null hypothesis).
* *The null model cannot contain any terms that are NOT in the full model*.
* Full model = model.3 🡪 contains main effect for therapy, main effect for drug, + the drug by therapy interaction term, drug:therapy
* The null model = model.1 🡪 contains only the main effect of drug.
* Say our full model can be written as a formula w/ several different terms, say **Y ~ A + B + C + D**.
* Our null model only contains *some subset* of these terms, say **Y ~ A + B**.
* Some of these terms might be main effect terms, others = interaction terms 🡪 really doesn’t matter
* Only thing that matters = want to treat some of these terms as the **starting point** (i.e. terms in the null model, A + B) + see if including the *other* terms (i.e., C + D) leads to significant improvement in model performance, over + above what could be achieved by a model that includes *only* A + B.
* In essence, we have null + alternatives that look like this:



* F-test is constructed from 2 kinds of quantity: **sums of squares (SS) + dF (df**), whichdefine a **mean square value (MS = SS/df),** + we obtain our F statistic by contrasting the MS associated w/ the thing we’re interested in (the model) w/ the MS value associated w/ everything else (the residuals).
* Want to figure out how to talk about the SS value associated w/ the difference between 2 models.
* Start w/ the fundamental rule used throughout regression 🡺 **SS.T = SS.M + SS.R** 🡺 total sums of squares (overall variability of the outcome variable) can be decomposed into 2 parts: variability associated w/ the model SS.M + the residual/leftover variability, SS.R.
* Useful to rearrange this equation 🡪say the SS value associated w/ a model is **SSM = SST - SSR**
* In our scenario, we have 2 models: the null (M0) + the full (M1):



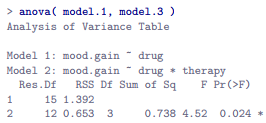
* Think about what we actually care about 🡪 *difference between the full model + the null model*
* So, to preserve the idea that we’re doing an analysis of the variance (ANOVA) in the outcome variable, define the SS associated w/ the difference to be equal to the difference in the SS values



* Now that we have our dF, we can calculate mean squares + F values in the usual way.
* Specifically 🡪 interested in the mean square *for the difference between models*, + the mean square *for the residuals associated w/ the full model* (M1), which are given by

* Finally, taking the ratio of these two gives us our F statistic: **F = MS∆ / MSR1**
* Null model here = model.1 🡪 stipulates there is a main effect of drug, but no other effects exist, expressed via the formula **mood.gain ~ drug**.
* The alternative model = model.3 🡪 stipulates there is a main effect of drug, a main effect of therapy, + an interaction 🡪 **mood.gain ~ drug + therapy + drug:therapy**
* Key = if we compare model.1 to model.3, we’re lumping the main effect of therapy + the interaction term together
* Input both models to anova() + it will run the exact F-test outlined above



* 1st , go back + look at the ANOVA tables for model.1 + model.3 to reassure that the RSS values printed in this table really do correspond to the residual sum of squares associated w/ these 2 models.



* Now, following the procedure above, the between model sum of squares = the difference between these 2 residual sum of squares values.



* Next, convert these SS values into MS values by dividing by the dF
* dF associated w/ the full-model residuals hasn’t changed from our original ANOVA for model.3 🡪 total sample size N, minus the total # of groups G relevant to the model.
* 18 people in the trial + 6 possible groups (2 therapies\*3 drugs) 🡪 dF.full = 12.
* dF for the null model are calculated similarly + the only difference = there are only 3 relevant groups (3 drugs), so dF.null = 15
* B/c the dF associated w/ the difference = the difference in the 2 dF, we have 15 - 12 = 3 dF.



* W/ our 2 MS values, we can divide one by the other + obtain an F-statistic



* Just as we had hoped, this turns = identical to the F-statistic anova() produced earlier

**16.6 ANOVA as a linear model**

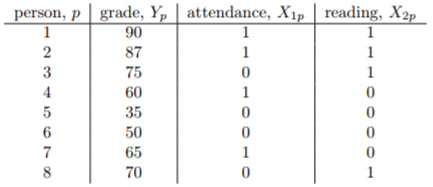
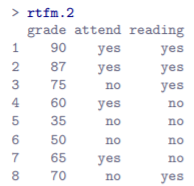
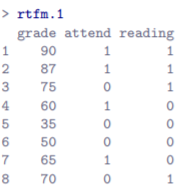
* 1 of the most important things to understand about ANOVA + regression = basically the same thing.
* On the surface, ANOVA is primarily concerned w/ testing for group differences, + regression is primarily concerned w/ understanding the correlations between variables
* But under the hood, the underlying mechanics of ANOVA + regression are awfully similar 🡪 both rely heavily on sums of squares (SS), make use of F tests, + so on.
* ANOVA + regression are both kinds of **linear models**.
* This is obvious in regression 🡺 equation used to define the relationship between predictors + outcomes = the equation for a straight line

 = 

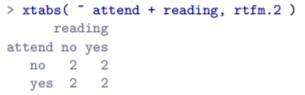
* If we ignore the residuals + just focus on the regression line itself, we get:



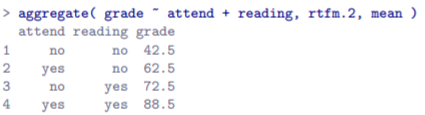
* where Yˆ p = value of Y the regression line predicts for person p, as opposed to the actually observed value Yp.
* *The thing that isn’t immediately obvious is we can write ANOVA as a linear model as well*.
* Simple example: rewriting a 2\*2 factorial ANOVA as a linear model
* Suppose our outcome variable = grade a student receives in class, a ratio-scale variable corresponding to a mark from 0%-100%.
* There are 2 binary predictors of interest = whether or not a student turned up to lectures (**attend**), + whether or not the student actually read the textbook (**reading**)
* Let Yp = the grade of the p-th student in the class 🡪 not that same as earlier, where we had notation Y(r, c)i to refer to the i-th person in the r-th group for predictor 1 (row factor) + the c-th group for predictor 2 (column factor).
* Extended notation = handy for describing how SS values are calculated, but a pain in this context
* Yp = visually simpler, but w/ the shortcoming that it *doesn’t keep track of the group memberships*
* If Y(0,0,3 ) = 35, you’d the 3rd didn’t attend lectures (attend = 0) nor read the textbook (reading = 0), + ended up failing the class (grade = 35).
* But if I tell you that Yp = 35, all you know is the p-th student didn’t get a good grade.
* To fix: introduce 2 new variables X1p + X2p to keep track of this info, where X1p = 0 🡺 attend = 0 + X2p = 0) 🡺 reading = 0

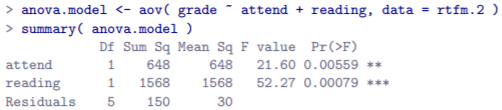
* This is exactly the format in which we’d expect to see our data
* Earlier I emphasized the importance of converting *nominal* scale variables such as attend + reading to *factors*, rather than encoding them as numeric (rtfm.2)
* But the goal here = to look at some of the math that underpins ANOVA
* Use xtabs() to confirm that this data set corresponds to a **balanced design**



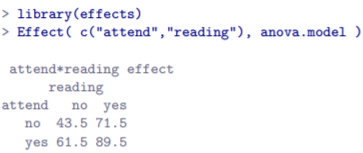
* If interested in calculating the mean grade for each cell, use aggregate()



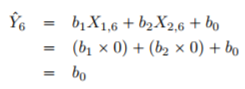
* Looking at this, one gets a strong impression that reading text + attending class both matter a lot.
* Now data is expressed in terms of 3 numeric variables: continuous variable Y + 2 binary variables X1 + X2
* Recognize our 2\*2 factorial ANOVA is exactly equivalent to the regression model used earlier to describe a 2-predictor regression model
* aov() is lm() in disguise
* The only difference 🡪 X1 + X2 are now *binary* variables (values can only be 0 or 1), whereas in a regression analysis we’d expect X1 + X2 be continuous.
* To prove this, running this as an ANOVA 1st w/ aov()



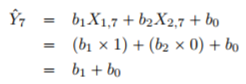
* We can see students obtained a higher grade if they attended class, (F(1,5) = 26.1, p < .0056) + if they read the textbook (F(1, 5) = 52.3, p < .0008).
* Make note of those p-values + those F statistics.



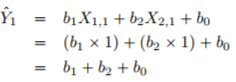
* Now think about the same analysis from a linear regression perspective.
* In rtfm.1 data, we have encoded attend + reading as numeric predictors.
* In this case, this is perfectly acceptable.
* There really is a sense in which a student who turns up to class (attend = 1) has in fact done more attendance than a student who does not (attend = 0).
* So, it’s not at all unreasonable to include it as a predictor in a regression model.
* It’s a little unusual, b/c the predictor only takes on 2 possible values, doesn’t violate any assumptions of linear regression, + is easy to interpret.
* If the regression coefficient for attend > 0, it means students that attend lectures get higher grades; if it’s < zero, students attending lectures get lower grades.
* The same is true for reading (intuitively obvious to everyone who has taken a few stats classes)
* Start by considering the 6th + 7th students in our data set (p = 6 + p = 7)
* Neither has read the text, so in both cases reading = 0, or “we observe X(2,6) = 0 + X(2,7) = 0
* However, student 7 did turn up to lectures (X(1,7) = 1) whereas student 6 did not (X(1,6) = 0)
* Insert these numbers into the general formula for our regression line
* Student 6



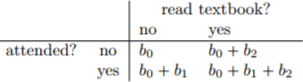
* Student 7



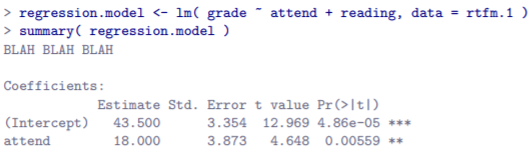
* So, we’re expecting student 6 will obtain a grade corresponding to the value of the intercept term b0 + student 7’s predicted grade = the intercept b0 + the coefficient associated w/ **attend**, b1.
* So, if b1 > 0, we expect students who turn up to lectures get higher grades than those who don’t
* If this coefficient is *negative*, we expect the opposite: students who turn up at class end up performing much worse.
* In fact, we can push this a little bit further
* Student 1 turned up to class (X(1,1) = 1) + read the text (X(2,1) = 1)

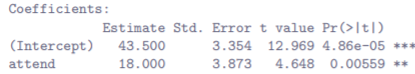


* So, if we assume attending class helps get a good grade (b1 > 0) + reading the text also helps get a good grade (b2 > 0), our expectation is student 1 will get a grade higher than student 6 + student 7
* So, we’re not surprised to learn the regression model predicts student 3 (read the book but didn’t attend lectures) will obtain a grade of b2 - b0



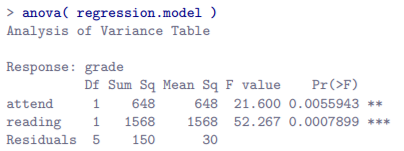
* As you can see, the intercept term b0 acts like a kind of baseline grade you’d expect from students who don’t take the time to attend class or read the textbook.
* **Similarly, b1 = boost expected if you come to class, + b2 = boost that comes from reading the text**
* **In fact, if this were an ANOVA, you might very well want to characterize b1 as the main effect of attendance, + b2 as the main effect of reading**
* *In fact, for a simple 2\*2 ANOVA, that’s exactly how it plays out.*
* Now actually run the regression using to convince ourselves this is really true.



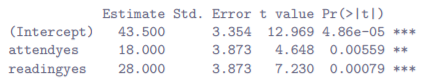




* Or use **coef(summary(regression.model))** to get the coefficients alone
* Notice the intercept term = 43.5 is close to the group mean = 42.5 observed for those 2 students who didn’t read the text or attend class.
* Moreover, it’s identical to the predicted group mean we pulled out of ANOVA using **Effects**()
* Secondly, notice we have the regression coefficient of b1 = 18.0 for attendance variable, suggesting those students that attended class scored 18% higher than those who didn’t.
* So, our expectation would be that students who turned up to class but didn’t read the textbook would obtain a grade of b0 + b1, which is 43.5 + 18.0 = 61.5
* Again, this is similar to the observed group mean 62.5 + identical to expected group mean pulled from our ANOVA.
* The same thing happens when we look at the students that read the textbook.
* We can push a little further in establishing the equivalence of our ANOVA + our regression.
* Look at the p-values associated w/ attend + reading in the regression output 🡪 identical to the ones encountered earlier when running ANOVA.
* Might seem surprising, since the regression model test calculates a t-statistic + ANOVA calculates an F-statistic.
* Remember (Ch. 9) there’s a relationship between the t-distribution + F-distribution = **If you have some quantity distributed according to a t-distribution w/ k degrees of freedom + square it, this new squared quantity follows an F-distribution whose dF = 1 + k**
* Can check this w/ respect to the t statistics in our regression model.
* For **attend**, we get t = 4.648 🡪 square this = 21.604, identical to the corresponding F-statistic in our ANOVA.
* 1 last thing to know 🡪 R understands that ANOVA + regression are both examples of linear models, + lets you extract the classic ANOVA table from a regression model using anova().

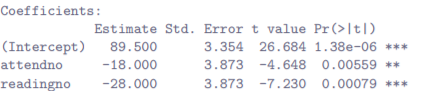


* What happens if we use **rtfm.2** to run the regression (coded **attend** + **reading** as factors rather than as numeric variables) 🡪 doesn’t 🡪 only differences are superficial

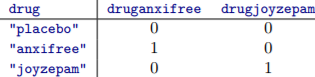


* The only thing that’s different is R labels the 2 variables differently
* Here, yes = 1 + no = 0.
* If yes = 0 + no = 1, all coefficients will go in the opposite direction b/c the effect of readingno would be referring to the consequences of NOT reading the textbook.
* Use **relevel**() to change which level of the reading variable is set to 0.
*  
* Notice R 1st prints out the level no + then yes
* W/ relevel(), R now lists yes before no = means R will now treat yes as the reference level (**baseline level**) when you include it in an ANOVA.





* This is a regression using the re-coded data w/ a few changes.
* attendno + readingno effects are both negative but w/ the same magnitude as before
* If you DON’T read the text, expect your grade to drop by 28% relative to someone who did
* t-statistics have reversed sign, + p-values remain the same, + the intercept has changed
* In our original regression, the **expected baseline** grade = student who didn’t attend class + didn’t read the textbook = 43.5
* However, w/ recoded variables, the baseline = student who HAS read the textbook + DID attend class = 89.5
* So that’s how we can view a 2\*2 ANOVA into a linear model + it’s easy to see how this generalizes to a 2\*2\*2 ANOVA or a 2\*2\*2\*2 ANOVA 🡺 same thing, just add a new binary variable for each factor.
* Its trickier w/ factors w/ > 2 levels 🡪 Consider the 3\*2 ANOVA w/ clin.trial data.
* How can we convert the 3-level drug factor into a numerical form appropriate for a regression?
* **Just realize a 3-level factor can be re-described as 2 binary variables.**
* Suppose we create a new binary variable druganxifree 🡪 whenever drug = anxifree, set druganxifree = 1 + otherwise, set druganxifree = 0.
* This variable sets up a **contrast**, in this case between anxifree + the other 2 drugs.
* By itself, the **druganxifree contrast** isn’t enough to fully capture all of the info in our drug variable.
* We need a 2nd binary contrast to distinguish between joyzepam + the placebo 🡪 drugjoyzepam, which = 1 if the drug = joyzepam, + 0 if it is not.
* Taken together, these 2 contrasts allows us to perfectly discriminate between all 3 possible drugs.

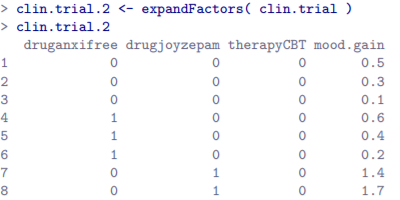


* Creating contrast variables manually is not too difficult to do using base R.

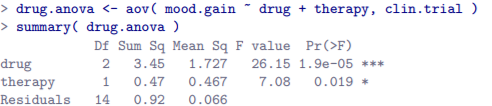




* Its tedious to do this over + over again for every single contrast you want to create.
* To make it easier, lsr package contains a simple function called **expandFactors**() to convert every factor in a data frame into a set of contrast variables
* Advanced users may want to look into **model.matrix**(), which produces similar output.
* Alternatively, use a command like **contr.treatment(3)[clin.trial$drug,].**



* Not as pretty as the original clin.trial data, but it’s definitely saying the same thing.
* We have now **recoded** our 3-level factor in terms of 2 binary variables, + we’ve already seen that ANOVA + regression behave the same way for binary variables.
* However, there are some additional complexities that arise in this case
* The equivalence between ANOVA + regression for non-binary factors
* Now we have 2 different versions of the same data set: a single 3-level factor, + 2 binary contrasts
* Want to demonstrate that our original 3\*2 factorial ANOVA is equivalent to a regression model applied to the contrast variables.
* Re-run the ANOVA + the regression

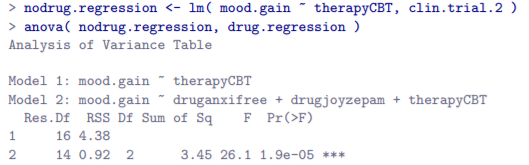


* 





* This ANOVA has the same output as last time, *but the regression does not*
* The regression output prints out the results for *each* of the 3 predictors separately
* On the one hand, we can see the p-value for **therapyCBT** is exactly the same as our original ANOVA, so we are reassured the regression model is doing the same thing as ANOVA.
* On the other hand, this regression model is testing the druganxifree + drugjoyzepam contrasts separately, as if they were 2 completely unrelated variables b/c lm() has no way of knowing these are actually the 2 different contrasts used to encode our 3-level drug factor.
* As far as it knows, these 2 are no more related than drugjoyzepam + therapyCBT.
* At this stage we’re not at all interested in determining whether these 2 contrasts are individually significant 🡪 just want to know if there’s an overall effect of drug.
* Want R to run some kind of **omnibus test** = one in which the 2 drug-related contrasts are lumped together for the purpose of the test = exactly the situation discussed in Section 16.5 + is precisely this situation the F-test is built to handle.
* All we need to do is specify our null 🡪 includes therapyCBT predictor + omit both drug variables



* Our F-statistic = 26.1, dF = 2 + 14, + the p-value = 0.000019, identical to what we obtained for the main effect of drug in our original ANOVA.
* **Once again, ANOVA + regression are essentially the same 🡪 both are linear models + the underlying statistical machinery for ANOVA is identical to the machinery used in regression.**
* Remember, **degrees of freedom =** the # of parameters that must be estimated in a model.
* For a regression model or ANOVA, the # of parameters = the # of regression coefficients (**b** values), including the intercept.
* Keeping in mind that any F-test is always a comparison between 2 models, the **1st dF = the difference in the # of parameters.**
* For example, the null model (mood.gain ~ therapyCBT) above has 2 parameters: a regression coefficient for **therapyCBT**, + a 2nd for the intercept.
* The alternative (mood.gain ~ druganxifree + drugjoyzepam + therapyCBT) has 4 parameters: a regression coefficient for each of the 3 contrasts, + 1 more for the intercept.
* So, the dF associated w/ the *difference* between these 2 models is dF1 = 4 - 2 = 2
* What if there doesn’t seem to be a null model?
* You might be thinking of the F-test that appears at the bottom of the regression output, originally described as a test of the regression model as a whole.
* However, that is still a comparison between 2 models:
* null = the trivial model that only includes an intercept, written as **outcome ~ 1**
* alternative = the *full* regression model.
* The null model in this case contains 1 regression coefficient, for the intercept term.
* The alternative model contains K - 1 regression coefficients (1 for each predictor variable, K) + 1 more for the intercept.
* So, the difference dF value you see in this F test is df1 = K + 1 - 1 = K.
* The 2ND dF value that appears in the F-test always refers to the dF *associated w/ the residuals*.
* It’s possible to think of this in terms of parameters too, but in a slightly counterintuitive way.
* Suppose the total # of observations across a study as a whole = N.
* If you wanted to *perfectly* describe each of these N values, you need to do so using N numbers.
* When you build a regression model, what you’re *really* doing is specifying *some* of the #’s needed to perfectly describe the data.
* If your model has K predictors + an intercept, you’ve specified K + 1 numbers.
* So, how many more #’s do you think are going to be needed to transform a K + 1 parameter regression model into a perfect re-description of the raw data?
* Well, since (K + 1) + (N – K – 1) = N, the answer would have to be N - K – 1
* In principle, imagine an absurdly complicated regression model that includes a parameter for every single data point + would, of course, provide a perfect description of the data.
* This model would contain N parameters in total, but we’re interested in the *difference* between the # of parameters required to describe this full model (i.e. N) + the # of parameters used by the simpler regression model you’re *actually interested in* (i.e., K + 1)
* So, the 2nd dF in the F test is df2 = N - K - 1, where K = # of predictors (in a regression model)/# of contrasts (in an ANOVA).
* In the example above, there are N = 18 observations + K + 1 = 4 regression coefficients associated w/ the ANOVA model, so the dF for the residuals is df2 = 18 - 4 = 14.
* 1 last thing to mention 🡪 previous example, we used **aov**() to run an ANOVA using clin.trial data, which codes drug as a single factor.
* We also used **lm**() to run a regression using clin.trial data in which we have 2 separate contrasts describing the drug.
* However, it’s also possible to use lm() on the original data.



* The fact that **drug** = 3-level factor does not matter 🡪 As long as **drug** has been declared as a factor, R automatically translates it into 2 binary contrast variables + performs the appropriate analysis
* After all, ANOVA + regression are both linear models + lm() = the function that handles linear models
* In fact, aov() doesn’t actually do very much of the work when you run an ANOVA using it
* Internally, R just passes all the hard work straight to lm().
* However, it is critical that your factor variables are declared as such.
* If **drug** were declared to be numeric, R would be happy to treat it as one.
* After all, it might be that **drug** refers to the # of drugs one has taken in the past, or something else that is genuinely numeric.
* R won’t second guess + assumes your factors are factors + your numbers are numbers.
* Don’t make the mistake of encoding factors as numbers, or R will run the wrong analysis.
* It’s your responsibility as the analyst to make sure you’re specifying the right model for your data.
* Kind of neat to run an ANOVA using lm()
* B/c you’ve called lm(),the **summary**() R pulls out is formatted like a regression.



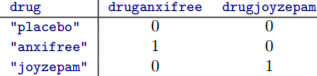
* However, b/c **drug** + **therapy** were both factors, **anova**() function actually *knows* which **contrasts** to group together for the purposes of running F-tests, so you can extract a classic ANOVA table.



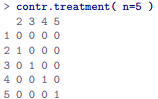
* *However, this behavior of* ***anova****() only occurs when the predictor variables are factors.*
* If we try a command like **anova(drug.regression),** the output will continue to treat druganxifree + drugjoyzepam as if they were 2 distinct binary factors.
* This is b/c in the drug.regression model, we *included all contrasts as raw variables, so R had no idea which ones belonged together.*
* However, w/ the drug.lm model, we gave R the *original factor variables*, so it DOES know which contrasts go together.
* The behavior of **anova**() reflects that.

**16.7 Different Ways to Specify Contrasts**

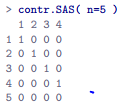
* Above, we had method for converting a factor into a collection of contrasts by specifying a set of binary variables defined in a table like this:



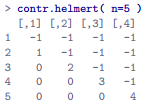
* Each row = a factor level, + each column = 1 of the contrasts.
* **Contrast Matrix =** table always has 1 more row than columns
* Lots of different ways to specify a contrast matrix (choice of contrasts doesn’t matter much for balanced designs)
* Treatment contrasts
* In the kind of contrasts described above, 1 level of the factor is *special* + acts as a kind of **baseline category** (placebo in our example), against which the other 2 are defined.
* The name for these kinds of contrasts = **treatment contrasts**.
* The name reflects the fact that these contrasts are quite natural + sensible when 1 category in your factor *really* *is special* b/c it *actually does represent a baseline*.
* That makes sense in a clinical trial 🡪 a placebo condition corresponds to a situation where you don’t give people any real drugs, + so it’s special.
* The other conditions are defined *in relation to the placebo* 🡪 replaced w/ anxifree or joyzepam
* R comes w/ a variety of functions that can generate different kinds of contrast matrices.
* Table shown above = a matrix of treatment contrasts for a factor w/ 3 levels.
* Matrix of treatment contrasts for a factor w/ 5 levels w/ **contr.treatment**()



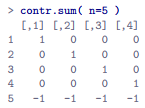
* By default, the 1st level of the factor is always treated as the baseline category (has all 0’s + doesn’t have an explicit contrast associated w/ it)
* Can use **relevel**() to change which category is the 1st level of the factor
* lsr contains a more general function called **permuteLevels**() that can shuffle them in any way you like
* There’s also a special function in R called **contr.SAS**() that generates a treatment contrast matrix in which the *last* category is treated as the baseline



* You can actually select any category you like as the baseline w/in **contr.treatment**() by specifying the **base** argument
* Helmert contrasts
* **Treatment contrasts** are useful for a lot of situations + are the default in R.
* However, they make most sense in the situation when there *really is* a baseline category + you want to assess all other groups in relation to that one.
* In other situations, however, no such baseline category exists + it may make more sense to compare each group to the *mean of the other groups.*
* **Helmert contrasts** generated by **contr.helmert**() can be useful w/ the idea to compare each group to the mean of the previous ones.
* That is, ***1st contrast = difference between group 2 + group 1, 2nd contrast = difference between group 3 + mean of groups 1 + 2, + so on.***
* This translates to a contrast matrix that looks like this:



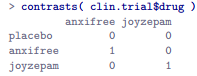
* 1 useful thing about **Helmert contrasts** is that **every contrast sums to 0** (all columns sum to 0).
* This has the consequence that, when we interpret the ANOVA as a regression, **the intercept term = the grand mean µ.. if using Helmert contrasts.**
* Compare this to **treatment contrasts, in which the intercept term = the group mean for the baseline category.**
* This property can be very useful in some situations.
* Here, *it doesn’t matter very much if you have a balanced design, which we’ve been assuming so far, but it will turn out to be important later w/ unbalanced designs in Section 16.10.*
* Contrasts become important if you want to understand *unbalanced* ANOVA
* Sum to zero contrasts
* These are used to construct pairwise comparisons between groups.
* Specifically, each contrast encodes the difference between 1 of the groups + a baseline category, which in this case corresponds to the *last group*:



* Much like Helmert contrasts, each column sums to 0 (intercept term = the grand mean when ANOVA is treated as a regression model)
* When interpreting these contrasts, the thing to recognize is = **each contrast is a pairwise comparison between group 5 + 1 of the other 4 groups**
* Specifically, contrast 1 = group 1 - group 5 comparison, contrast 2 = a group 2 - group 5 comparison, + so on.
* Every factor variable in R is associated w/ a contrast matrix.
* It has to be, otherwise R wouldn’t be able to run ANOVAs properly
* If you don’t specify one explicitly, R will implicitly specify one for you.
* W/ the clin.trial data, we didn’t specify any contrast matrix for either factor (check w/ **attr** to print out contrast attributes of factor variables)



* NULL = drug factor doesn’t have any attribute called “contrasts” for which it has any data.
* `There is no contrast matrix stored anywhere explicitly for this factor.
* Ask R to tell us what contrasts are set up for this factor:



* These are the same treatment contrast we set up manually in Section 16.6.
* R knew to set up treatment contrasts (even though we told it anything) b/c R has a hidden list of default options it looks up to resolve situations like this (print out all options by w/ **options**())
* There are a lot of options + we’re only interested in **contrasts** right now.

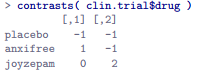


* This tells us the default contrasts for unordered factors (*nominal* scale variables) are **treatment contrasts**, + the default for ordered factors (*interval* scale variables) are **polynomial contrasts**.
* The key thing = **options**() also allows you to *reset* these defaults (only for the current session)





* Now see the default contrasts for unordered factors have changed.
* Now ask R for the contrasts associated w/ drug 🡪 gives a different answer b/c we changed default:

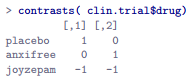


* In general, if changing the default settings for something in R, it’s a good idea to reset them to their original values once you’re done.



* Setting the contrasts for a single factor
* Suppose all you really want to do is change the contrasts associated w/ a *single* factor, + leave the defaults as they are.
* To do this, specifically assign the contrast matrix as an attribute of the factor via **contrasts**()
* Suppose I wanted to use sum to zero contrasts for **drug**, but keep the default treatment contrasts for everything else.





* Check that we have actually made a specific change to the factor itself by checking to see if it now has an attribute

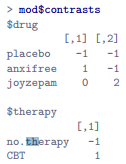


* This prints out the same output shown above, b/c the contrast has in fact been attached to the drug factor, + does not rely on the defaults.
* If you want to wipe the attribute + revert the defaults, use **contrasts( clin.trial$drug ) <- NULL**
* Setting the contrasts for a single analysis
* 1 last way of changing contrasts 🡪 find yourself wanting to change the contrasts only for 1 specific analysis
* aov() + lm() have a contrasts argument you can use.
* To change contrasts for 1 specific analysis, 1st set up a **list** variable that names the contrast types you want to use for each factor
* Technically, this list actually stores the functions themselves.
* R allows lists to contain functions, which is really neat for advanced purposes





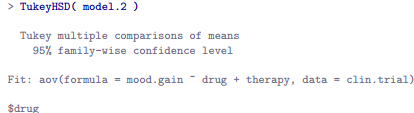
* If you try **summary(aov)**, you won’t see any difference in the output b/c choice of contrasts does not affect the outcome w/ a balanced design (won’t always be true later on).
* To check that it has actually worked, inspect the value of **mod$contrasts**

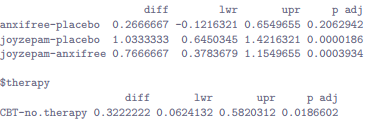


* See, *for the purposes of this 1 particular ANOVA*, we used Helmert contrasts for both variables
* If we did not specify the contrasts argument, we’d be looking at treatment contrasts here, b/c it would’ve reverted to whatever values **contrasts**() prints out for each of the factors.

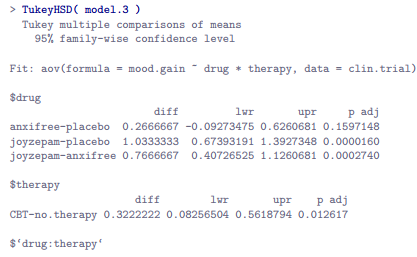
**16.8 Post hoc tests**

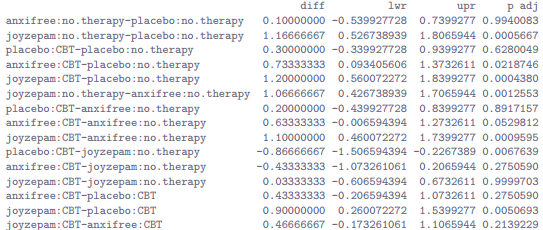
* Suppose you’ve done your ANOVA + it turns out that you obtained some significant effects.
* B/c **F-tests** = **omnibus tests** that really only test the null hypothesis = “there are no differences among groups”, obtaining a significant effect doesn’t tell you WHICH groups are different
* Ch. 14 solution was to run t-tests for all possible pairs of groups, making corrections for multiple comparisons (**Bonferroni, Holm**) to control the Type I error rate across all comparisons.
* Ch. 14 methods have the advantage of being relatively simple tools you can use in a lot of different situations when testing multiple hypotheses, but *they’re not necessarily the best choices if you’re interested in doing efficient post-hoc testing in an ANOVA context.*
* There are quite a lot of different methods for performing multiple comparisons in statistics literature (Hsu, 1996)
* That being said, there’s 1 tool to draw attention to 🡪 **Tukey’s Honestly Significant Difference (HSD)**
* Basic idea in Tukey’s HSD = examine all relevant pairwise comparisons between groups
* It’s only really appropriate to use Tukey’s HSD if it is *pairwise differences* you’re interested in.
* If, for instance, you’re interested to know if Group A is significantly different from the mean of Group B + Group C, you need to use a different tool (**Scheffe’s method =** more conservative)
* However, in most cases you probably are interested in pairwise group differences, so Tukey’s HSD is a pretty useful thing to know about.
* In model.2 (specified a main effect for both drug + therapy), we’d be interested in the following 4 comparisons:
* Difference in mood gain for people given Anxifree vs. people given the placebo
* Difference in mood gain for people given Joyzepam vs people given the placebo
* Difference in mood gain for people given Anxifree vs people given Joyzepam
* Difference in mood gain for people treated w/ CBT vs. people given no therapy.
* For any of these comparisons, we’re interested in the *true* difference between (*population*) group means
* **Tukey’s HSD** constructs *simultaneous CIs for all 4 of these comparisons.*
* **95% simultaneous CI** = there is a 95% probability that all these CIs contain the relevant true value.
* Moreover, can use these CIs to calculate **an adjusted p value** for any specific comparison.
* TukeyHSD() 🡪 input the model to run the post hoc tests for.
* If looking to run post hoc tests for model.2:





* The 1st comparison = Anxifree vs. placebo difference, + the 1st part of the output indicates the **observed difference in group means** = .27.
* The next 2 #’s indicate the 95 (simultaneous) CI for this comparison runs from -.12 to .65
* **B/c the CI for the difference *includes 0*, we CANNOT reject the null hypothesis that the 2 group means are identical, + so we’re not all that surprised to see that the adjusted p-value is .21**.
* In contrast, the observed difference in group means between Joyzepam + the placebo = 1.03, + the 95% CI runs from .64 to 1.42.
* **B/c the interval EXCLUDES 0, we see the result is significant p < .001.**
* What if a model includes **interaction terms**? 🡪 model.3 allowed for the possibility of an interaction between drug + therapy.
* If that’s the case, the number # of pairwise comparisons to consider starts to increase.
* Need to consider the 3 comparisons relevant to the main effect of drug + the 1 comparison relevant to the main effect of therapy.
* But, to consider the possibility of a *significant* interaction (+ try to find the group differences that underpin that significant interaction), we need to include comparisons such as
* Difference in mood gain for Anxifree + CBT vs. the placebo + CBT
* Difference in mood gain for Anxifree + no therapy vs. the placebo + no therapy
* Etc.
* There are quite a lot of these comparisons to consider.





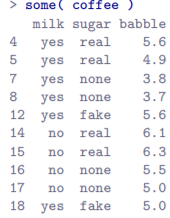
* Looks pretty similar to before, but w/ a lot more (19 total) comparisons made.

**16.9 The method of planned comparisons**

* **Method of planned comparisons** = important enough to deserve a quick discussion.
* In discussions of multiple comparisons (previous section + Ch. 14), we’ve been assuming the tests are *genuinely* post hoc.
* Maybe we thought the drugs would all have different effects on mood (i.e. hypothesized a main effect of drug), but didn’t have any specific hypothesis about HOW they would be different, nor have any real idea about WHICH pairwise comparisons would be worth looking at.
* If that is the case, we must resort to something like Tukey’s HSD to do pairwise comparisons.
* The situation is rather different if you *genuinely* did have *real, specific hypotheses* about *which* comparisons are of interest w/ no intention to look at any other comparisons besides the ones specified ahead of time.
* When this is true, + if you honestly + rigorously stick to your intentions to not run any other comparisons (even when data are showing significant effects for stuff you didn’t have a hypothesis test for), it doesn’t really make sense to run something like Tukey’s HSD, as it makes corrections for a whole bunch of comparisons you never cared about + never had any intention of looking at.
* Under those circumstances, you can safely run a (limited) number of hypothesis tests *w/out making an adjustment for multiple testing.*
* This situation = the **method of planned comparisons** + is sometimes used in clinical trials.

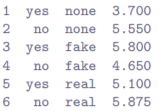
**16.10 Factorial ANOVA 3: Unbalanced Designs**

* **Factorial ANOVA** = very handy thing to know about + 1 of the standard tools used to analyze experimental data for many decades (can’t read more than 2 or 3 papers in psychology w/out running into an ANOVA in there somewhere)
* 1 huge difference between ANOVAs in a lot of real scientific articles + the ANOVA just described
* IRL = rarely lucky enough to have perfectly balanced designs.
* For 1 reason or another, it’s typical to end up w/ more observations in some cells than in others.
* **Unbalanced designs** need to be treated w/ a lot more care than balanced designs + the statistical theory that underpins them is a lot messier.
* Many active researchers don’t know there’s several different types of unbalanced ANOVAs + they produce quite different answers.
* Most people who report results of an unbalanced factorial ANOVA don’t actually give you enough details to reproduce the analysis
* May not even realize their statistical software package is making a whole lot of substantive data analysis decisions on their behalf
* **coffee.Rdata** contains a hypothetical data set that produces an unbalanced 3\*2 ANOVA.
* Suppose we’re interested in finding out if the tendency of people to babble when they have too much coffee is purely an effect of coffee itself or there’s some effect of added milk + sugar
* Suppose we took 18 people + gave them some coffee to drink w/ amount of coffee/caffeine held constant + varied whether or not milk was added (milk = binary factor w/ 2 levels, yes + no)
* Also varied *kind* of sugar involved 🡪 real, fake (i.e., artificial sweetener), or none = 3 level factor.
* Our outcome variable = continuous variable that presumably refers to some psychologically sensible measure of the extent to which someone is babbling (details don’t really matter for our purpose)
* To get a sense of what the data look like, use **some**() in the **car** package 🡪 randomly picks a few observations in the data frame to print out

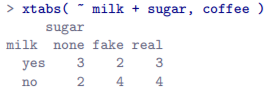


* If we use **aggregate**() function to quickly produce a table of means:





* We get a strong impression there are differences between the groups
* This is especially true when we compare these means to the SDs for the **babble** variable (calculate using aggregate() in much the same way)
* Across groups, this SD varies from .14 to .71, fairly small relative to the differences in group means
* This discrepancy in SDs might (+ should) make you wonder if we have a violation of homoscedasticity/homogeneity of variance assumption 🡪 check w/ **leveneTest**()
* So far, it’s looking like a straightforward factorial ANOVA, just like earlier.
* The problem arises when we check to see how many observations we have in each group:



* This violates 1 of our original assumptions 🡪 that the number of people in each group is the same.
* Standard ANOVA does not exist for unbalanced designs
* Unbalanced designs lead us to the somewhat unsettling discovery that there isn’t really any 1 thing we might refer to as a “standard ANOVA”.
* It turns out there are 3 main fundamentally different ways in which you might want to run an ANOVA in an unbalanced design.
* ANOVAs can vary in other ways besides the ones discussed in this book.
* There’s a difference between **fixed-effect models** (this book), in which levels of a factor are “fixed” by the experimenter (or the world), + **random-effect models**, in which levels are random samples from a larger population of possible levels
* Most researchers get by w/ a basic working knowledge of ANOVA that doesn’t go any further than this book.
* Just keep in mind this book is only the beginning of a very long story, not the whole story.
* W/ a *balanced* design, all 3 versions produce identical results, w/ the sums of squares, F-values, etc. all conforming to the formulas given at the start of the Chapter.
* However, when a design is *unbalanced* they DON’T give the same answers.
* Furthermore, they’re not all equally appropriate to *every* situation 🡪 some methods will be more appropriate to your situation than others.
* Given all this, it’s important to understand the *different* types of ANOVA + how they differ from 1 another.
* 1st kind of ANOVA = **Type I Sum of Squares**.
* Sum of squares part of the name was introduced by SAS + has become standard nomenclature, but it’s a bit misleading in some ways.
* Logic for referring to the different ANOVAs as different types of sum of squares is that may be due to the ANOVA tables produced have key difference in the SS values.
* dF don’t change + MS values are still defined as SS / dF, etc.
* However, the terminology hides the reason *why the SS values are different from one another.*
* More helpful to think of the 3 different kinds of ANOVA as 3 different hypothesis testing strategies
* These different strategies lead to different SS values, to be sure, but the strategy is the important thing here*, not the SS values themselves*.
* Recall (Section 16.5 + 16.6) **any particular F-test is best thought of as a comparison between 2 linear models.**
* So, **when looking at an ANOVA table, it helps to remember that each F-test corresponds to a pair of models that are being compared**.
* This naturally leads to the question of *which pair of models is being compared.*
* THAT is the fundamental difference between ANOVA Types I, II + III 🡪 each corresponds to a different way of choosing the model pairs for the tests.
* Type I SS
* Type I method is sometimes referred to as the **sequential sum of squares** b/c it involves a process of adding terms to the model 1 at a time.
* Suppose we want to run the full 3\*2 factorial coffee ANOVA, including interaction terms.
* The full model = **babble ~ sugar + milk + sugar:milk**, though often shortened to **sugar \* milk**
* The Type I strategy builds this model up sequentially, starting from the simplest possible model + gradually adding terms.
* Simplest possible model here = *neither milk nor sugar is assumed to have any effect on babbling*
* The only term included in such a model = the intercept 🡪 **babble ~ 1** = our initial null
* The next simplest model = only 1 of the 2 main effects is included.
* For coffee data, there are 2 different possible choices here, milk or sugar 1st
* The order actually turns out to matter, but for just make a choice arbitrarily + pick sugar.
* So, the 2nd model in our sequence of models = **babble ~ sugar**, + the alternative for our 1st test.
* We now have our 1st hypothesis test:



* THIS comparison forms our hypothesis test of the main effect of sugar.
* The next step in our model building = add the other main effect term
* Next model = **babble ~ sugar + milk**.
* The 2nd hypothesis test is then formed by comparing the following pair of models:

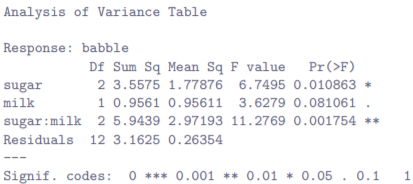


* THIS comparison forms our hypothesis test of the main effect of milk.
* In 1 sense, this approach is elegant 🡺 the alternative from the 1st test forms the null for the 2nd
* It is in this sense that the Type I method is *strictly sequential* 🡺 Every test builds directly on the results of the last one.
* However, in another sense it’s *inelegant*, b/c there’s a strong asymmetry between the 2 tests.
* The test of the main effect of sugar (1st test) completely ignores milk, whereas the test of the main effect of milk (2nd test) DOES take sugar into account.
* In any case, the 4th model = the full model = **babble ~ sugar + milk + sugar:milk**, + the corresponding hypothesis test is

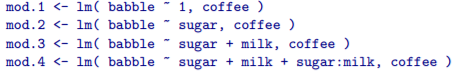


* **Type I SS =** the default hypothesis testing method used by anova() 🡪 just type in the same commands as always.
* Also, since ANOVA + regression are both linear models, use lm() to run the analyses:

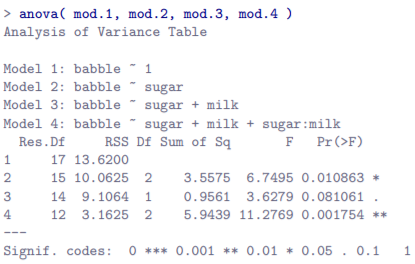




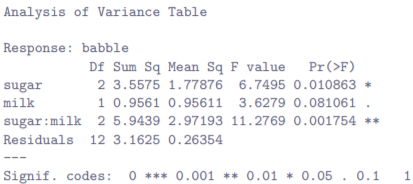
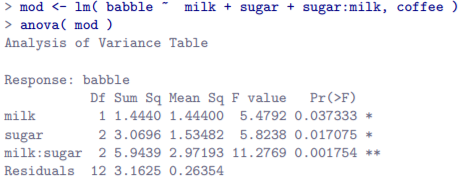
* Leaving aside for a moment the question of how this result should be interpreted, take note of the fact that our 3 p-values = .0109, .0811, + .0018 respectively.
* Next, let’s see if we can replicate the analysis using tools we’re a little more familiar w/.
* 1st, Fit all 4 models:



* To run the 1st hypothesis test comparing mod.1 to mod.2, use **anova(mod.1, mod.2)** in much the same way in Section 16.5
* Similarly, we can use **anova(mod.2, mod.3) + anova(mod.3, mod.4)** + to run the 2nd + 3rd hypothesis tests.
* Rather than run each of those commands separately, enter the full sequence of models like this:



* This output is rather more verbose than the last one, but it’s telling essentially the same story
* 1 thing that might seem a little opaque to some people = *why the residual dF in this output look different from one another* (i.e., ranging from 12 to 17) whereas in the original the residual dF is fixed at 12.
* R uses a residual dF of 12 in all cases (that’s why the p-values = the same in the 2 outputs, + it’s enough to verify **pf(6.7495, 2,12, lower.tail = FALSE))** gives the correct answer of p = .010863, for instance, whereas **pf(6.7495, 2,15, lower.tail=FALSE))** gives a p-value = .00812.
* It’s the residual dF in the *full* model (the last one) that matters here
* The big problem w/ using Type I sum of squares = it really DOES depend on the order in which you enter the variables.
* Yet, in many situations a researcher has no reason to prefer 1 ordering over another.
* Whether we add milk or sugar 1st feels as arbitrary as a data analysis question as it does as a coffee-making question.
* Look what happens when we change the ordering:

* The p-values for both main effect terms have changed fairly dramatically.
* Among other things, the effect of milk has become significant (avoid drawing any strong conclusions about this).
* Which of these 2 ANOVAs should one report? 🡪 It’s not immediately obvious.
* When you look at the hypothesis tests used to define the 1st main effect + the 2nd one, it’s clear they’re qualitatively different from one another.
* Initial example 🡪 the test for the main effect of sugar completely ignores milk, whereas the test of the *main effect of milk does take sugar into account*.
* As such, the Type I testing strategy really DOES treat the 1st main effect as if it had a kind of theoretical “primacy” over the second.
* There is very rarely, if ever, any theoretically primacy of this kind that would justify treating any 2 main effects asymmetrically.
* **The consequence of all = Type I tests are very rarely of much interest**
* The key principle in Type I SS = The hypothesis testing is sequential w/ terms added 1 at a time.
* However, it also implies main effects be added 1st (factors A, B, C, etc.), followed by 1st-order interaction terms (A:B + B:C), then 2nd-order interactions (A:B:C) + so on.
* W/in each block, you can specify whatever order you like.
* So, for instance, if we specified our model using a command like this



* Reordering terms *w/in each group* WILL change the ordering if we use **anova(mod)** to produce sequential hypothesis tests
* However, changing the order of terms *across blocks* has NO effect.



* If for some reason you really need an interaction term to be entered 1st, you have to do it the long way 🡪 creating each model manually using a separate **lm**() + using a command like **anova(mod.1, mod.2, mod.3, mod.4)** to force R to enter them in the order that you want.
* Type III SS 🡪 simple while Type II tests are trickier
* The basic idea behind Type III tests = regardless of which term you’re trying to evaluate, run the F-test in which the alternative = the full ANOVA model as specified by the user + the null just deletes that 1 term you’re testing.
* Our full model = **babble ~ sugar + milk + sugar:milk**, the test for a main effect of sugar would correspond to a comparison between the following 2 models



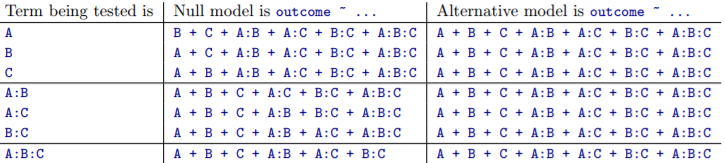
* Similarly, the main effect of milk is evaluated by testing the full model against a null model that removes the milk term



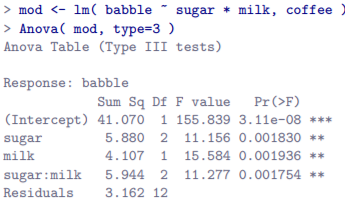
* Finally, the interaction term sugar:milk is evaluated in exactly the same way.



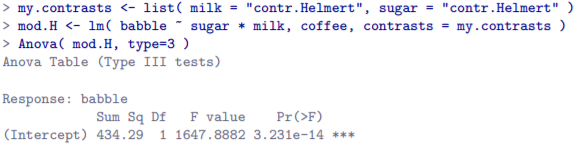
* The basic idea generalizes to higher-order ANOVAs.
* For instance, suppose we’re trying to run an ANOVA w/ 3 factors, A, B + C, + wanted to consider all possible main effects + all possible interactions, including the 3-way interaction A:B:C.

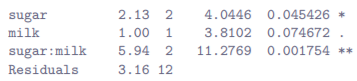


* As ugly as that table looks, it’s pretty simple
* In all cases, the alternative = the full model, which contains 3 main effect terms, 3 1st-order interactions, + 1 2nd-order interaction
* The null always contains 6 of these 7 terms + the missing 1 = the 1 whose significance we’re trying to test.
* At 1st pass, Type III tests seem like a nice idea.
* We’ve removed the asymmetry that caused us to have problems in Type I tests + are now treating all terms the same way, so the results of the hypothesis tests do not depend on the order in which we specify them.
* This is definitely a good thing, but there is a big problem when interpreting the results of the tests, especially for main effect terms
* Suppose it turns out the main effect of milk is not significant according to the Type III tests.
* This tells us **babble ~ sugar + sugar:milk** is a “better” model for the data than the full model.
* *But what does that even mean?*
* If the interaction term sugar:milk was *also* non-significant, we’d be tempted to conclude the data are telling us the only thing that matters is sugar.
* But suppose we have a significant interaction term, but a NON-significant main effect of milk.
* In this case, are we to assume there really is an effect of sugar, an interaction between milk + sugar, but no effect of milk?
* That seems crazy + the right answer simply must be it’s meaningless to talk about the main effect if the interaction is significant.
* Or, at the very least, rarely of interest
* In general, this seems to be what most statisticians advise us to do, + that’s the right advice.
* But if it *really* is meaningless to talk about non-significant main effects in the presence of a significant interaction, it’s not at all obvious why Type III tests should allow the null to rely on a model that includes the interaction but omits 1 of the main effects that make it up.
* When characterized in this fashion, the null hypotheses really don’t make much sense at all.
* Type III tests can be redeemed in some contexts, but 1st let’s actually compute a Type III ANOVA
* anova() in R does not directly support Type II tests or Type III tests.
* Technically, you can do it by creating the various models that form the null + alternative hypotheses for each test + *then* use anova() to compare the models to one another.
* I outlined the gist of how that would be done when talking about Type I tests, but speaking from first-hand experience I can tell you that it’s very tedious.
* In practice, anova() is only used to produce Type I tests or to compare specific models of particular interest (Section 16.5)
* If you want Type II or Type III, use **Anova**() in the **car** package w/ a **type** argument that you specify.
* So, our Type III tests are run as follows:

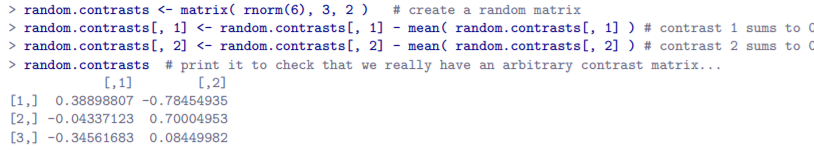


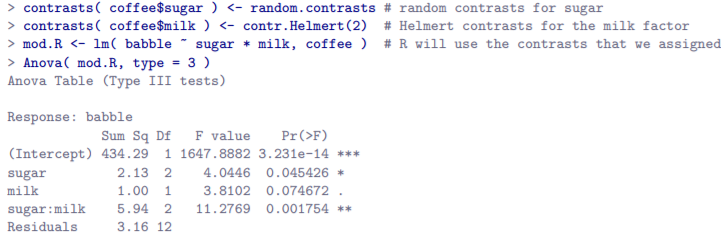
* sugar \* milk = shorthand way of referring to sugar + milk + sugar:milk.
* We can see our Type III tests are significant for all terms, even the intercept.
* Except, as usual, it’s not that simple.
* 1 of the perverse features of the Type III testing strategy = the results turn out to depend on the **contrasts** used to encode factors (Section 16.7)
* The results above are based on the R default, **treatment contrasts** 🡪 usually a very poor choice for Type III tests.
* Switch to **Helmert contrasts**:





* This isn’t good 🡪 milk’s p-value has changed from .002 to .0, a pretty substantial difference
* Hopefully this gives a sense of how important it is to take care when using Type III tests.
* If the p-values that come out of Type III analyses are so sensitive to choices of contrasts, does it mean Type III tests are essentially arbitrary + not to be trusted?
* To some extent that’s true, + Type II analyses avoid this arbitrariness entirely, but that’s too strong a conclusion
* Firstly, it’s important to recognize some choices of contrasts will *always* produce the same answers.
* Of particular importance = the fact that if the columns of our contrast matrix are all constrained to sum to 0, the Type III analysis will always give the same answers.
* This means you’ll get the same answers if you use **contr.Helmert** or **contr.sum** or **contr.poly**, but *different* answers for **contr.treatment** or **contr.SAS**.





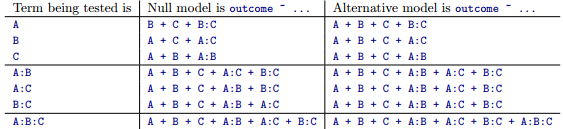
* Yep, same answers
* Type II sum of squares
* Type I + III tests are pretty straightforward
* Type I tests = gradually *adding* terms 1 at a time, whereas Type III tests = taking the full model + looking to see what happens when you *remove* each term.
* However, both have some serious flaws:
* Type I tests = dependent on the order in which you enter the terms
* Type III tests = dependent on how you code up contrasts.
* B/c of these flaws, neither is easy to interpret.
* Type II tests = harder to describe, but avoid both these problems 🡪 result = easier to interpret.
* Type II tests are broadly similar to Type III tests 🡺 start w/ a full model + test a particular term by deleting it from that model.
* However, Type II tests are based on the **marginality principle** which states you should not omit a lower order term from a model if there are any higher order ones that depend on it.
* Ex: If a model contains the interaction A:B (2nd-order term), it really ought to contain the main effects A + B (1st-order terms).
* If it contains a 3-way interaction term A:B:C, the model must include main effects A, B + C as well as the *simpler interactions* A:B, A:C + B:C.
* Type III tests routinely violate the marginality principle.
* Consider the test of main effect of A in the context of a 3-way ANOVA that includes all possible interaction terms.
* According to Type III tests, our null + alternative models are:



* Notice the null omits A, but includes A:B, A:C + A:B:C as part of the model.
* This, according to Type II tests, is NOT a good choice of a null.
* Instead, to test the null that A is not relevant to our outcome, specify the null = *the most complicated model that does not rely on A in any form, even as an interaction.*
* The alternative = this null model + a main effect term of A.
* This is a lot closer to what most people would intuitively think of as a main effect of A, + it yields the following as our Type II test of the main effect of A:



* Note, of course, this depends on the model the user specified.
* If the original ANOVA model doesn’t contain an interaction term for B:C, obviously it won’t appear in either the null or the alternative.
* But that’s true for Types I, II and III 🡪 They *never* include any terms *you* didn’t include, but they make different choices about how to construct tests for ones you DID include.
* For a sense of how Type II tests work, see full table of tests applied in a 3-way factorial ANOVA



* In the context of the 2-way ANOVA we’ve been using for coffee data, the hypothesis tests are even simpler
* Main effect of sugar = an F-test comparing these 2 models:



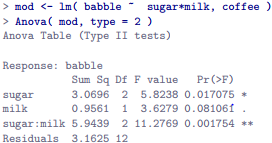
* Test for main effect of milk is:



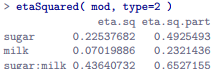
* Test for the interaction sugar:milk is:



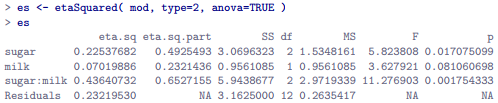
* To run the tests, use **Anova**() function, specifying **type=2**:



* Type II tests have some clear advantages over Type I + Type III tests 🡪 don’t depend on order in which you specify factors (Type I) nor on the contrasts used to specify your factors (Type III).
* Although opinions may differ on this last point (+ it will definitely depend on what you’re trying to do w/ your data), the hypothesis tests they specify are more likely to correspond to something you actually care about.
* As a consequence, it’s usually easier to interpret the results of a Type II test than Type I or Type III
* For this reason, tentative advice = If you can’t think of any obvious model comparisons that directly map onto your research questions but still want to run an ANOVA in an unbalanced design, Type II tests are probably a better choice than Type I or Type III
* Effect sizes (+ non-additive sums of squares)
* **etaSquared**() in **lsr** package computes η2 + partial η2 values for unbalanced designs + for different Types of tests 🡪 just indicate which type of tests you’re doing



* W/ an unbalanced design, there’s a bit of extra complexity involved.
* Expand the output from **etaSquared**() so it displays the full ANOVA table



* Remember back to early discussions of ANOVA 🡪 1 of the key ideas behind the sums of squares calculations is all the SS terms associated w/ the effects in the model + the residual SS is supposed to = the total sum of squares
* On top of that, the whole idea behind η2 is (b/c you’re dividing 1 of the SS terms by the total SS) an **η2 value** can be interpreted as a **proportion of variance accounted for by a particular term**
* Above, (including the η2 value associated w/ the residuals = proportion of variance in the outcome attributed to the residuals, rather than to 1 of the effects), you’d expect all η2 values to sum to 1.
* B/c the whole idea was that the variance in the outcome variable can be divided up into the variability attributable to the model, + the variability in the residuals.
* Yet these η2 values don’t sum to 1 and they generally don’t for type II and Type III tests.



* Some of the variability has gone missing 🡪 not being attributed to the model nor to the residuals
* From a mathematical perspective, it’s easy enough to see that the missing variance = a *consequence of the fact that in Types II + III, individual SS values are NOT obliged to the total sum of squares, + will only do so if you have balanced data.*
* Verify that this is the case using the ANOVA table.
* 1st, calculate total sum of squares directly from the raw data



* Next, read off all SS values from 1 of our Type I ANOVA tables + add them up.



* This gives us the same answer, just like it’s supposed to:
* However, when we do the same thing for the Type II ANOVA table (above), the SS values add up to slightly less than the total SS value:



* Once again, there’s a little bit of variance that has disappeared somewhere.
* This happens b/c w/ unbalanced designs, **factors become correlated w/ one another, + it becomes difficult to tell the difference between the effect of Factor A + the effect of Factor B**.
* In the extreme case, suppose we’d run a 2\*2 design in which the # of participants in each group had been as follows:



* Here we have a spectacularly unbalanced design: 100 people have milk + sugar, 100 people have no milk + no sugar, + that’s all 🡪 0 people w/ milk + no sugar, + 0 people w/ sugar but no milk.
* Now suppose that when we collected the data, it turned out there is a large (+ statistically significant) difference between the milk + sugar group + the no-milk + no-sugar group.
* Is this a main effect of sugar? A main effect of milk? Or an interaction? 🡺 ***It’s impossible to tell,*** b/c the presence of sugar has a **perfect association** w/ the presence of milk.
* Now suppose the design had been a *little* more balanced:



* This time around, it’s *technically* possible to distinguish between the effect of milk + effect of sugar, b/c we have a few people that have 1 but not the other.
* However, it will still be pretty difficult to do so b/c the association between sugar + milk is still extremely strong + there are so few observations in 2 of the groups.
* Again, we’re very likely to be in the situation where we know the predictor variables (milk + sugar) are related to the outcome (babbling), but don’t know if the nature of that relationship is a main effect of 1 predictor, the other predictor, or the interaction.
* **This uncertainty is the reason for the missing variance**.
* The missing variance = variation in the outcome variable clearly attributable to the predictors, but we don’t know WHICH of the effects in the model is responsible.
* When you calculate Type I SS, no variance ever goes missing 🡪 the sequential nature of Type I SS means the ANOVA automatically attributes this variance to whichever effects are entered 1st
* However, the Type II + Type III tests are more conservative 🡪 Variance that cannot be *clearly* attributed to a specific effect doesn’t get attributed to any of them, + it goes missing