

Week 8 - GLM - ANOVA part 1

Andrew Stewart

Andrew.Stewart@manchester.ac.uk



@ajstewart_lang



<https://github.com/ajstewartlang>

Week	Topic
1	Introduction, Open Science, and Power
2	Introduction to R
3	Data Wrangling and Visualisation
4	General Linear Model - Regression
5	General Linear Model - Regression
6	No Timetabled Lecture - Reading Week
7	Consolidation Lab
8	General Linear Model - ANOVA
9	General Linear Model - ANOVA
10	Tidy Thursday Data Wrangling & Visualisation Challenge
11	Reproducing your Computational Environment using Binder
12	Dynamic, Reproducible Presentations Using xaringan

Semester 1 Assignments

Data wrangling and visualisation – Due December 5th

ANOVA/ANCOVA – Due January 17th

- We're going to have our first look at the Analysis of Variance (ANOVA).
- This week we'll look at ANOVA for within-subjects, between-subjects and mixed designs.
- ANOVA is an important statistical test and (in various forms) is used widely across many areas of psychology.

Assessment

- The second assessment will be on the ANOVA lectures. It will again require you to conduct an ANOVA and to produce a report using R Markdown.
- The assessment question will be of a similar type to the ones we'll look at in the lab classes over the next couple of weeks.

Reporting ANOVA

- Say what type of ANOVA it was, say what factors you had (and with labels for each level).
- Report the results of main effects first, then interactions.
- Report F values, exact p -values and effect size values.
- Remember to interpret interactions further - such as with contrasts or pairwise comparisons.
- When you have main effects, say which direction the effect goes.
- Avoid sillies - e.g., mixing up $<$ and $>$ or saying $p = .000$

Why ANOVA, why not t-tests?

- So, t-tests are fine if we're just comparing two means.
- In the real world of psychology, we often have more than two conditions.
- How could we analyse our data ?

- One possibility could be that we do multiple t-tests – but there's a problem with that.
- With one t-test, at $p < 0.05$ alpha level there is a 5% chance of falsely rejecting our null hypothesis (type I error).
- If we have three conditions, then we have three pairs of means to compare (condition 1 vs condition 2, condition 2 vs condition 3 and condition 1 vs condition 3).

- For each test, there is 0.95 probability of not having a type I error.
- But when we do three tests the probability is $0.95 \times 0.95 \times 0.95$ which equals 0.857.
- So that means there is a 14.3% chance of us falsely rejecting the null hypothesis $(1 - 0.857) \times 100 = 14.3$

The familywise error rate

- This is known as the familywise error rate.

$$\text{familywise error} = 1 - (0.95)^n$$

- If we had 5 conditions, and hence 10 t-tests to conduct, our error rate would be 0.4 – which means there is a 40% chance of having made at least one Type I error (i.e., thinking we have an effect when none is present).

Similarities between t-tests and the ANOVA

- t-tests tell us whether or not two samples have the same mean.
- ANOVA tells us whether two or more samples have the same mean.
- As the t-test produced the t-statistic, the ANOVA gives us an F-statistic or F-ratio which compares the amount of systematic variance with the amount of unsystematic variance.

- ANOVA can tell us that there is a difference between means – so for three samples it tells us that $\overline{X}_1 = \overline{X}_2 = \overline{X}_3$ is not true.
- But it doesn't tell us where the difference is.
- It doesn't tell us whether \overline{X}_1 differs from both \overline{X}_2 and \overline{X}_3 or whether \overline{X}_2 differs from \overline{X}_3 but not \overline{X}_1 etc.

ANOVA

- Imagine we're interested in the impact of caffeine consumption on an individual's motor performance.
- It's a between-subjects design with 3 conditions:
 - low amount of caffeine (single espresso)
 - large amount of caffeine (double espresso)
 - placebo group (water)

- We conduct an ANOVA and find a significant F-ratio.
- What does it mean?
- The single espresso people could have performed better from the double espresso and water group.
- Or maybe they performed the same as the water group but better than the double espresso group.
- Or maybe (unexpectedly) they performed worse than both the double espresso and water groups.
- To know what is the case we need to do planned contrasts (similar to 1 tailed tests) or post hoc tests (similar to 2 tailed tests).

- We know that at least one of our means differs from at least one of our other means but (so far) we don't know where that difference lies.....
- Luckily things easy for us as we can conduct what are known as post hoc tests. These will tell us which means differ from which other means (and allow us to begin to tell a story....)

Post hocs tests

- Work by doing pairwise comparisons on all the different combinations of experimental groups.....
- They control for the familywise error rate though to get round that problem.
- Bonferroni method divides our critical p value (0.05) by the number of tests. If we are conducting ten tests, then for each test the critical p is 0.005 – but this increases our chances of a Type II error – missing an effect when it's there.

When deciding which post hoc test to use :

Does it control the Type I error rate ?

Does it control the Type II error rate ?

Is it reliable when ANOVA assumptions have been violated ?

LSD, Bonferroni, and Tukey tests.

- The least significant differences test (LSD) doesn't control the Type I error and is like doing multiple t-tests on the data (but only if the ANOVA is significant).
- Bonferroni and Tukey both control for Type I errors but are conservative. Bonferroni works by dividing the critical alpha level by the number of tests conducted.
- Tukey is less conservative than Bonferroni.

The Packages

```
library(tidyverse) #load the tidyverse packages
```

```
library(afex) #load afex for running ANOVA
```

```
library(emmeans) #load emmeans for running pairwise comparisons
```

ANOVA

We have 45 participants, a between participants condition with 3 levels (Water vs. Single Espresso vs. Double Espresso), and Ability as our DV measured on a continuous scale.

```
cond <- read_csv("data_files/cond.csv")
```

```
cond$Condition <- as.factor(cond$Condition)
```

```

> cond
# A tibble: 45 x 3
  Participant Condition Ability
      <dbl>    <fct>      <dbl>
1         1      Water      4.82
2         2      Water      5.41
3         3      Water      5.73
4         4      Water      4.36
5         5      Water      5.47
6         6      Water      5.50
7         7      Water      5.07
8         8      Water      5.08
9         9      Water      5.07
10        10      Water      4.94
# ... with 35 more rows

```

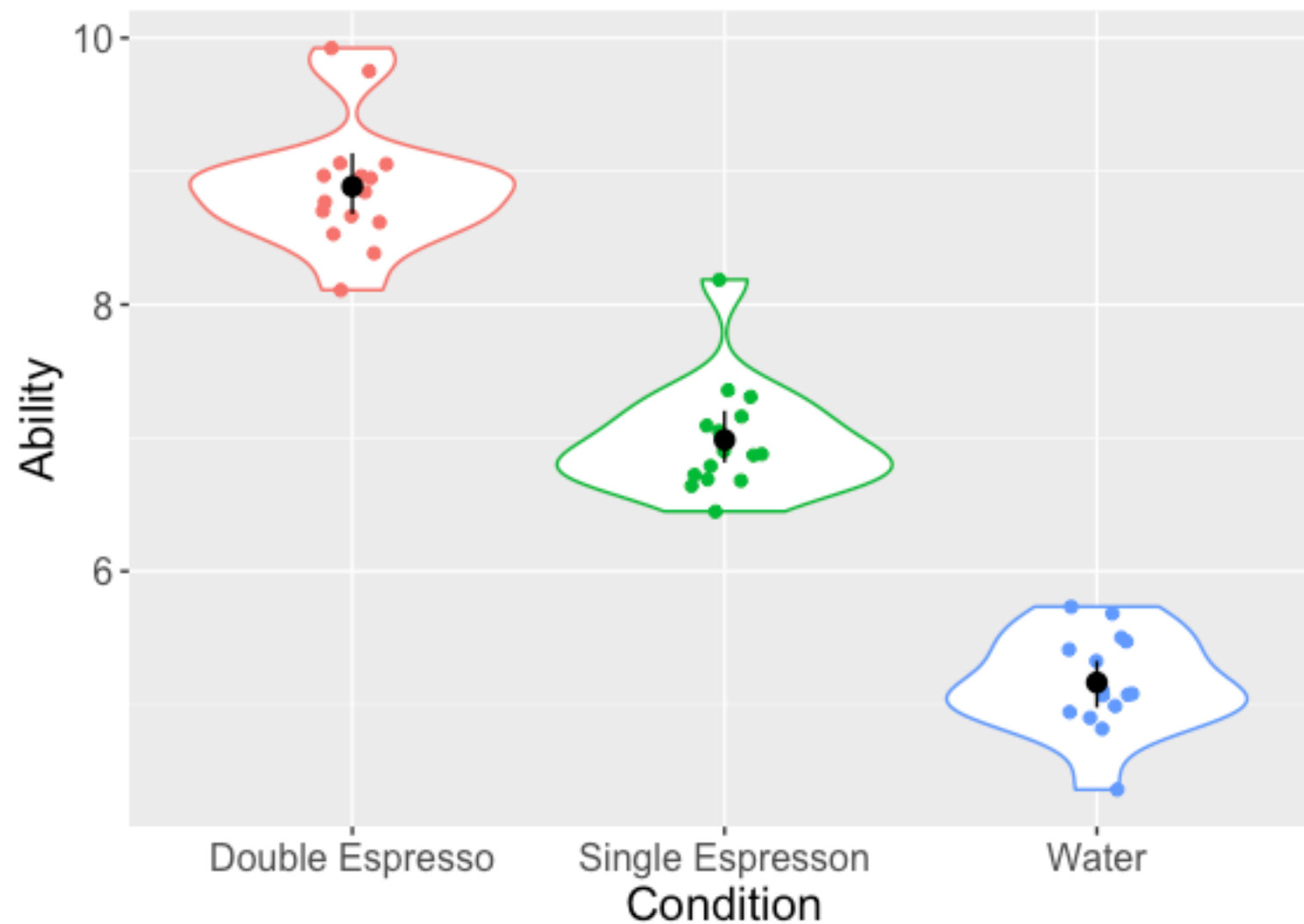
We have three columns - Participant number, Condition, and Ability. Condition is our IV, and Ability our DV. Note, our data are in tidy or long format with one observation per row.

Generating Descriptives and Visualising the Data

```
cond %>%  
  group_by(Condition) %>%  
  summarise(mean = mean(Ability), sd = sd(Ability))
```

```
# A tibble: 3 x 3  
  Condition      mean      sd  
  <fct>      <dbl> <dbl>  
1 Double Espresso  8.89 0.467  
2 Single Espresso  6.99 0.419  
3 Water           5.17 0.362
```

```
cond %>%
  ggplot(aes(x = Condition, y = Ability, colour = Condition)) +
  geom_violin() +
  geom_jitter(width = .1) +
  guides(colour = FALSE) +
  stat_summary(fun.data = "mean_cl_boot", colour = "black") +
  theme(text = element_text(size = 15))
```



```
library(afex)
```

```
model <- aov_4(Ability ~ Condition + (1 | Participant),  
data = cond)
```

This is our DV

This is our IV

This is our random effect

```
> summary(model)
```

```
Anova Table (Type 3 tests)
```

```
Response: Ability
```

	num	Df	den	Df	MSE	F	ges	Pr(>F)	
Condition	2		42		0.17484	297.05	0.93397	< 2.2e-16	***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

To determine what's driving the effect we can use the `emmeans::emmeans()` to run pairwise comparisons (note, default is Tukey correction).

```
> emmeans(model, pairwise ~ Condition)
```

```
$emmeans
```

Condition	emmean	SE	df	lower.CL	upper.CL
Double Espresso	8.89	0.108	42	8.67	9.10
Single Espresso	6.99	0.108	42	6.77	7.20
Water	5.17	0.108	42	4.95	5.38

```
Confidence level used: 0.95
```

```
$contrasts
```

contrast	estimate	SE	df	t.ratio	p.value
Double Espresso - Single Espresso	1.90	0.153	42	12.453	<.0001
Double Espresso - Water	3.72	0.153	42	24.372	<.0001
Single Espresso - Water	1.82	0.153	42	11.920	<.0001

```
P value adjustment: tukey method for comparing a family of 3 estimates
```


Measure of Effect Size

- The effect size is measured by η_G^2 which stands for generalised effect size or generalised eta squared (η_G^2).
- For designs with more than one factor it can be a useful indicator of how much variance in the dependent variable can be explained by each factor (plus any interactions between factors).

```
> summary(model)
Anova Table (Type 3 tests)
```

```
Response: Ability
```

	num	Df	den	Df	MSE	F	ges	Pr(>F)	
Condition		2		42	0.17484	297.05	0.93397	< 2.2e-16	***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So, to make sense of our output

- We found a significant effect of Beverage type ($F(2,42) = 297.05$, $p < .001$, generalised $\eta^2 = .93$). Tukey comparisons revealed that the Water group performed significantly worse than the Single Espresso Group ($p < .001$), that the Water group performed significantly worse than the Double Espresso Group ($p < .001$), and that the Single Espresso Group performed significantly worse than the Double Espresso Group ($p < .001$).
- In other words, drinking some coffee improves motor performance relative to drinking water, and drinking a lot of coffee improves motor performance even more.

ANOVA for repeated measures designs

- Let's imagine we have an experiment where we asked 32 participants to memorise words of differing levels of spelling complexity - Very Easy, Easy, Hard, and Very Hard.
- They were presented with these words in an initial exposure phase. After a 30 minute break we tested them by asking them to write down all the words. We scored them as number correct for each condition.
- We want to know whether there is a difference in the number of words they remembered for each level of spelling complexity.

```
rm_data <- read_csv("data_files/rm_data.csv")
rm_data$Condition <- as.factor(rm_data$Condition)
rm_data
```

```
# A tibble: 128 x 3
```

	Participant	Condition	Score
	<dbl>	<fct>	<dbl>
1	1	Very Easy	80
2	2	Very Easy	86
3	3	Very Easy	89
4	4	Very Easy	75
5	5	Very Easy	86
6	6	Very Easy	87
7	7	Very Easy	82
8	8	Very Easy	82
9	9	Very Easy	82
10	10	Very Easy	81

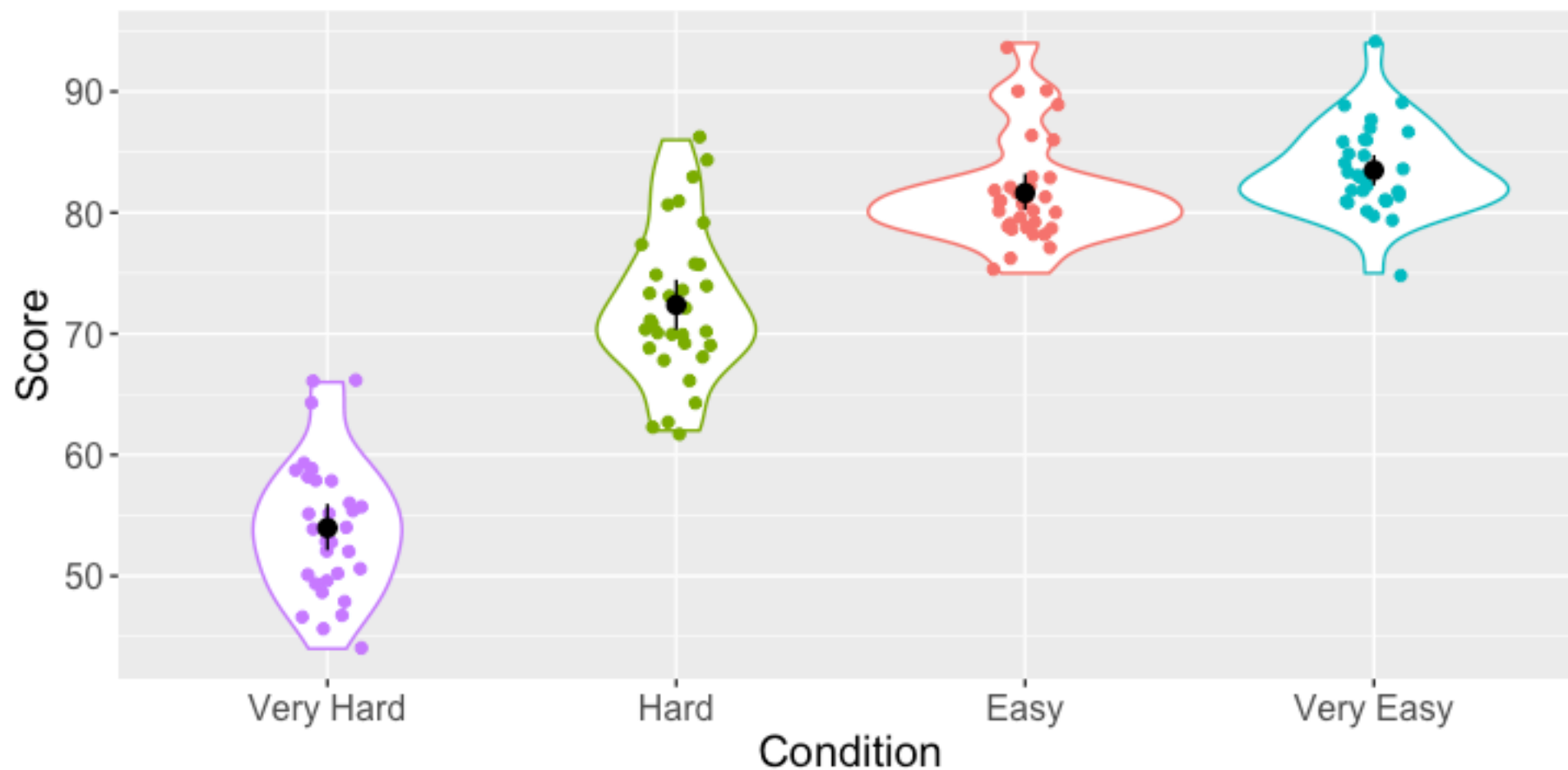
```
# ... with 118 more rows
```

Generating Descriptives and Visualising the Data

```
rm_data %>%  
  group_by(Condition) %>%  
  summarise(mean = mean(Score), sd = sd (Score))
```

```
# A tibble: 4 x 3  
  Condition    mean    sd  
  <fct>      <dbl> <dbl>  
1 Easy       81.6   4.28  
2 Hard       72.4   6.24  
3 Very Easy  83.5   3.62  
4 Very Hard  54.0   5.50
```

```
rm_data %>%
  ggplot(aes(x = fct_reorder(Condition, Score), y = Score, colour =
Condition)) +
  geom_violin() +
  geom_jitter(width = .1) +
  guides(colour = FALSE) +
  stat_summary(fun.data = "mean_cl_boot", colour = "black") +
  theme(text = element_text(size = 15)) +
  labs(x = "Condition")
```



This is the our ANOVA model - we have a significant effect of Condition.

```
> model <- aov_4(Score ~ Condition + (1 + Condition | Participant), data = rm_data)
> summary(model)
```

Univariate Type III Repeated-Measures ANOVA Assuming Sphericity

	SS	num	Df	Error	SS	den	Df	F	Pr(>F)	
(Intercept)	679632		1	936.49		31	22497.36	< 2.2e-16	***	
Condition	17509		3	2179.48		93	249.04	< 2.2e-16	***	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Mauchly Tests for Sphericity

	Test statistic	p-value
Condition	0.90603	0.71042

Greenhouse-Geisser and Huynh-Feldt Corrections
for Departure from Sphericity

	GG eps	Pr(>F[GG])
Condition	0.9401	< 2.2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	HF eps	Pr(>F[HF])
Condition	1.043895	2.615157e-44

```

> anova(model)
Anova Table (Type 3 tests)

Response: Score
          num Df den Df      MSE      F      ges      Pr(>F)
Condition  2.8203   87.43 24.928 249.04 0.84892 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

The effect size is measured by ges which stands for generalised effect size (η_G^2) - this is the recommended effect size measure for repeated measures designs (Bakeman, 2005). We get this by using the `anova()` function on our model. Note the dfs in this output are always corrected as if there is a violation of sphericity - to be conservative (and to avoid Type I errors) we might be better off to always choose these corrected dfs.

Where does the difference lie?

```
> emmeans(model, pairwise ~ Condition, adjust = "Bonferroni")
```

```
$emmeans
```

Condition	emmean	SE	df	lower.CL	upper.CL
Easy	81.6	0.886	122	79.9	83.4
Hard	72.4	0.886	122	70.6	74.1
Very.Easy	83.5	0.886	122	81.7	85.3
Very.Hard	54.0	0.886	122	52.2	55.7

Warning: EMMs are biased unless design is perfectly balanced
Confidence level used: 0.95

```
$contrasts
```

contrast	estimate	SE	df	t.ratio	p.value
Easy - Hard	9.25	1.21	93	7.643	<.0001
Easy - Very.Easy	-1.88	1.21	93	-1.549	0.7483
Easy - Very.Hard	27.66	1.21	93	22.852	<.0001
Hard - Very.Easy	-11.12	1.21	93	-9.192	<.0001
Hard - Very.Hard	18.41	1.21	93	15.209	<.0001
Very.Easy - Very.Hard	29.53	1.21	93	24.401	<.0001

P value adjustment: bonferroni method for 6 tests

So far we have looked at ANOVA for designs when we have one factor which is between subjects (i.e., each participant appears in one condition), and for designs when we have one factor that is repeated measures (each participant appears in all conditions). These are examples of 1-way ANOVA.

Now we're going to look at factorial ANOVA - this is for cases where we have more than one factor and we might be interested in how the factors interact with each other. If we have two factors, we have a 2-way ANOVA, three factors a 3-way ANOVA etc.

- Imagine we have 2 factors. Factor 1 with two levels, Factor 2 with three. Our analysis might reveal a main effect of Factor 1 (i.e., a difference between the two levels), a main effect of Factor 2 (i.e., a difference between the three levels) or an interaction between the two.....

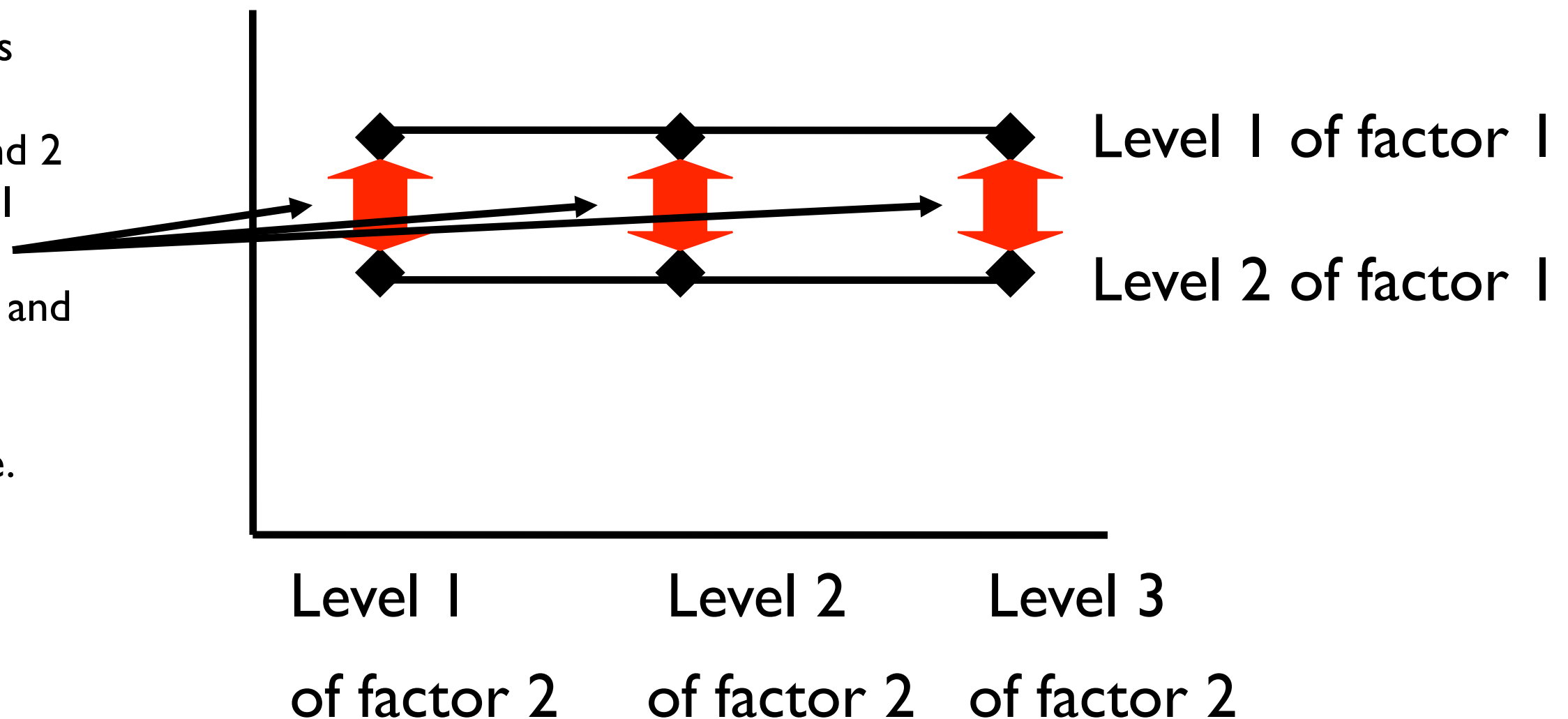
- This is a 2x3 ANOVA

Corresponds to
Factor 1 – it has
two levels.

Corresponds to Factor 2
– it has three levels.

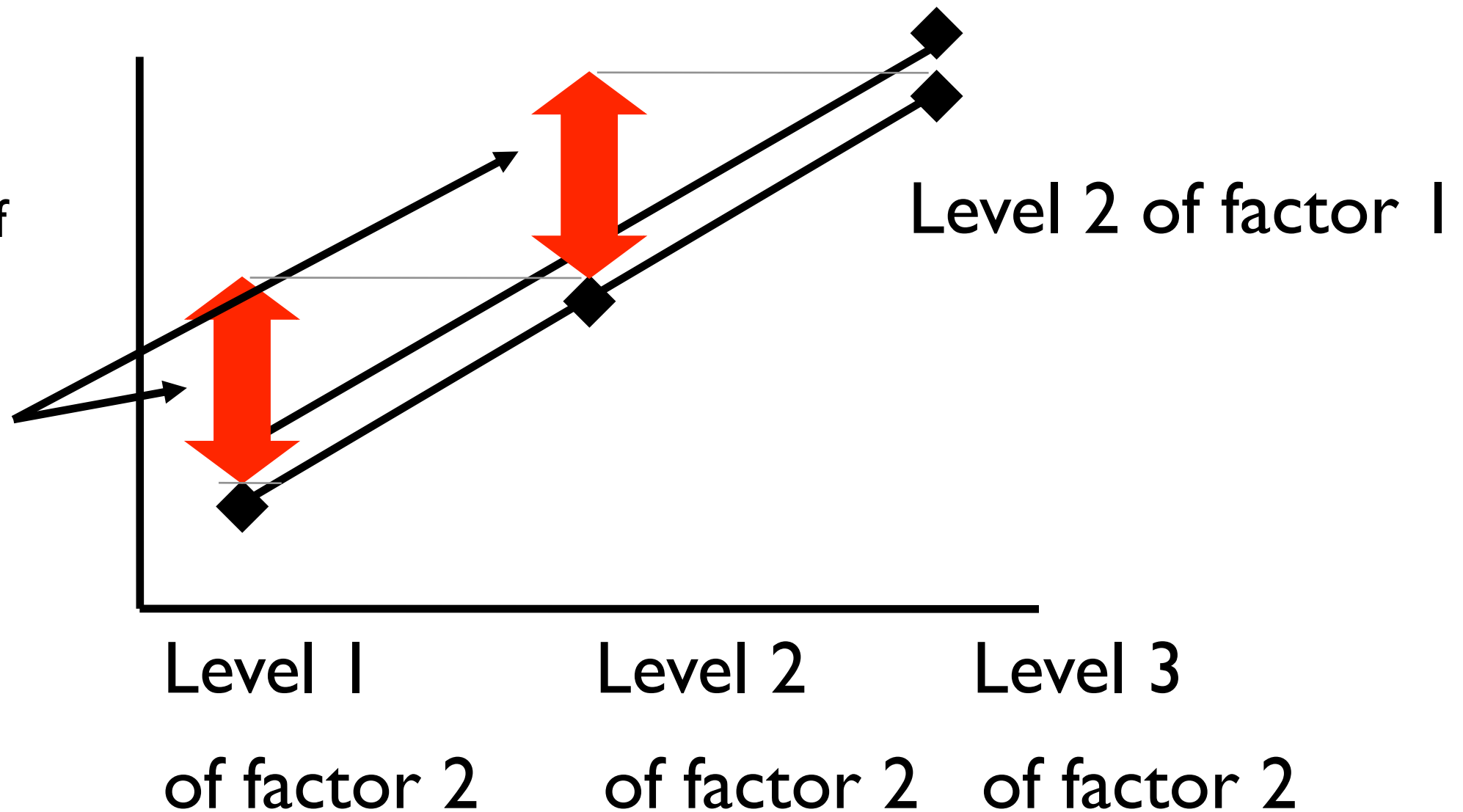
Main effect of Factor 1, no main effect of Factor 2 and no interaction

The differences between levels 1 and 2 of Factor 1 are all significant and are of the same magnitude.



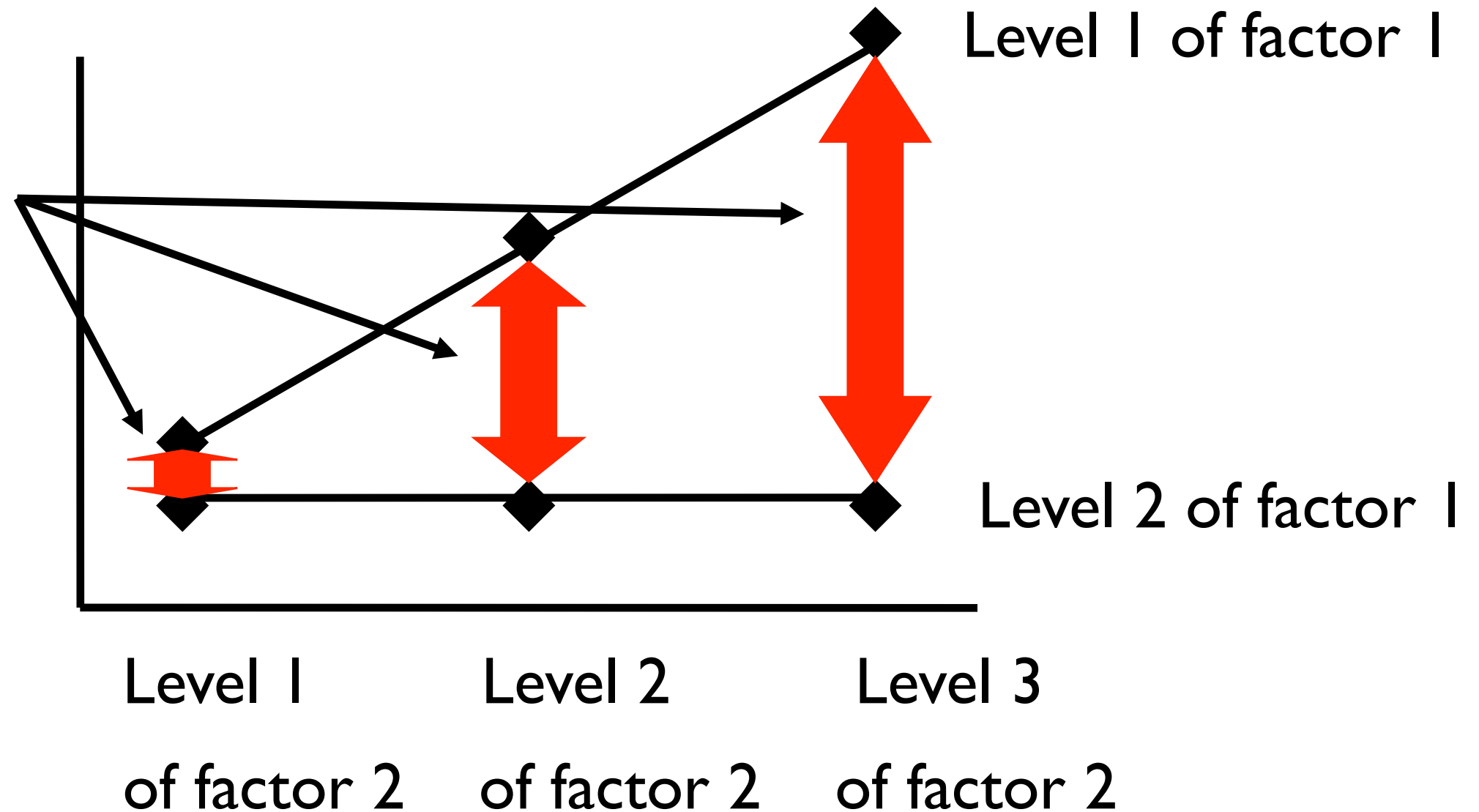
No main effect of Factor 1, main effect of Factor 2 and no interaction

The differences between levels 1 & 2 and 2 & 3 of Factor 2 are all significant and are of the same magnitude. There are no significant differences between levels 1 and 2 of Factor 1.



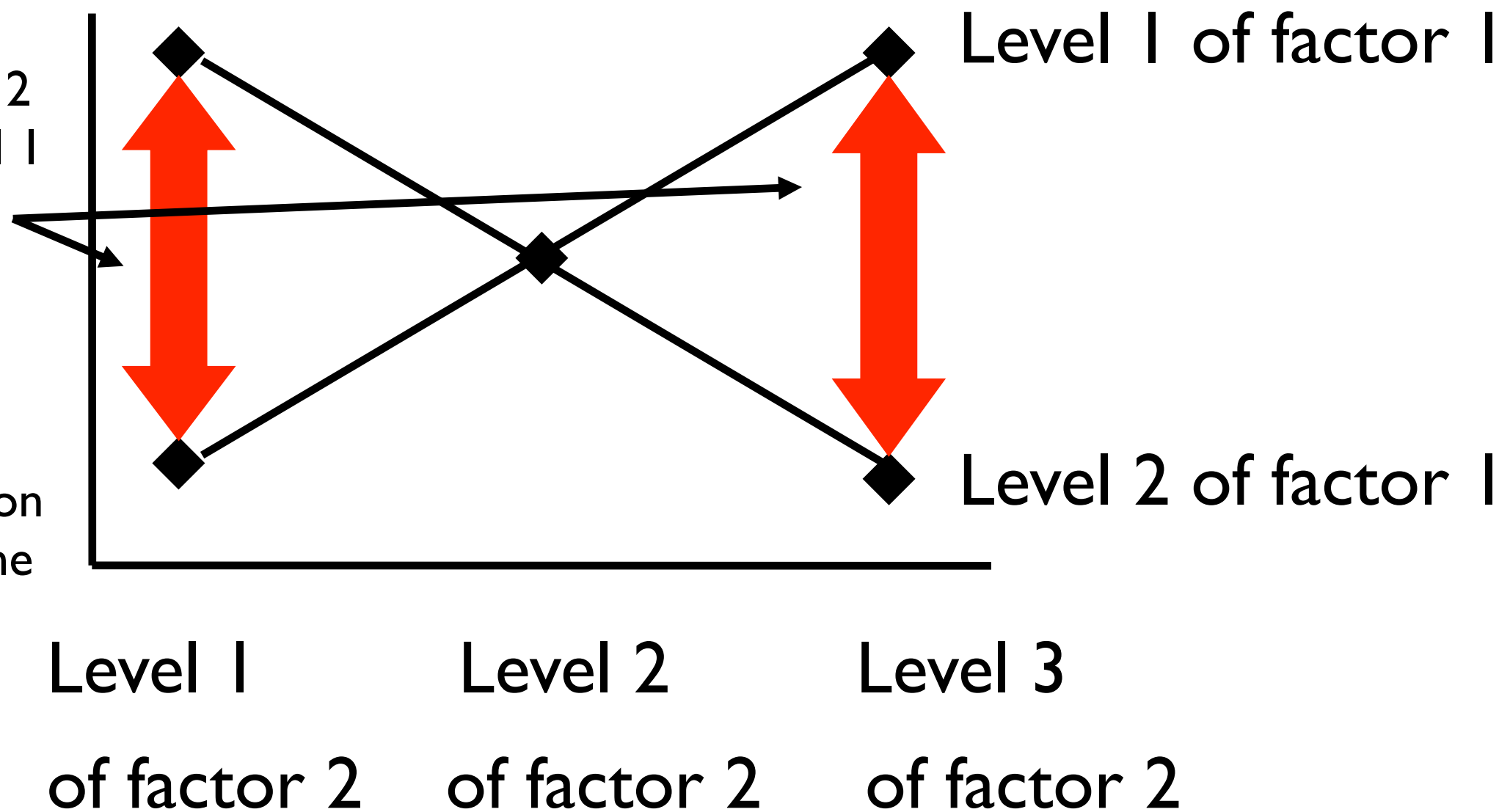
Main effect of Factor 1, main effect of Factor 2 and an interaction

The differences between the two levels of factor 1 change as a function of factor 2.



No main effect of Factor 1, no main effect of Factor 2 but an interaction

The difference between levels 1 & 2 of Factor 1 at Level 1 of Factor 2 is different from the same difference at Levels 2 and 3 of Factor 2. This is a crossover interaction as the polarity of the difference flips.



Factorial ANOVA

- Imagine the case where we're interested in the effect of positive vs. negative contexts on how quickly (in milliseconds) people respond to positive vs negative sentences. We think there might be a priming effect (i.e., people are quicker to respond to positive sentences after positive contexts vs. after negative contexts - and vice versa).
- So, we have two factors, each with two levels. This is what's known as a full factorial design where every subject participates in every condition.


```
fact_data <- read_csv("data_files/fact_data.csv")
fact_data$Sentence <- as.factor(fact_data$Sentence)
fact_data$Context <- as.factor(fact_data$Context)
```

```
fact_data
```

```
# A tibble: 1,680 x 5
```

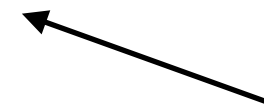
	Subject	Item	RT	Sentence	Context
	<dbl>	<dbl>	<dbl>	<fct>	<fct>
1	1	3	1270	Positive	Negative
2	1	7	739	Positive	Negative
3	1	11	982	Positive	Negative
4	1	15	1291	Positive	Negative
5	1	19	1734	Positive	Negative
6	1	23	1757	Positive	Negative
7	1	27	1052	Positive	Negative
8	2	4	1706	Positive	Negative
9	2	8	533	Positive	Negative
10	2	12	1009	Positive	Negative

```
# ... with 1,670 more rows
```

Generating Descriptives and Visualising the Data

```
fact_data %>%  
  group_by(Context, Sentence) %>%  
  summarise(mean = mean(RT), sd = sd(RT))
```

```
# A tibble: 4 x 4  
# Groups:   Context [2]  
  Context Sentence mean    sd  
  <fct>     <fct>   <dbl> <dbl>  
1 Negative Negative 1474.  729.  
2 Negative Positive  NA    NA  
3 Positive Negative  NA    NA  
4 Positive Positive 1579.  841.
```

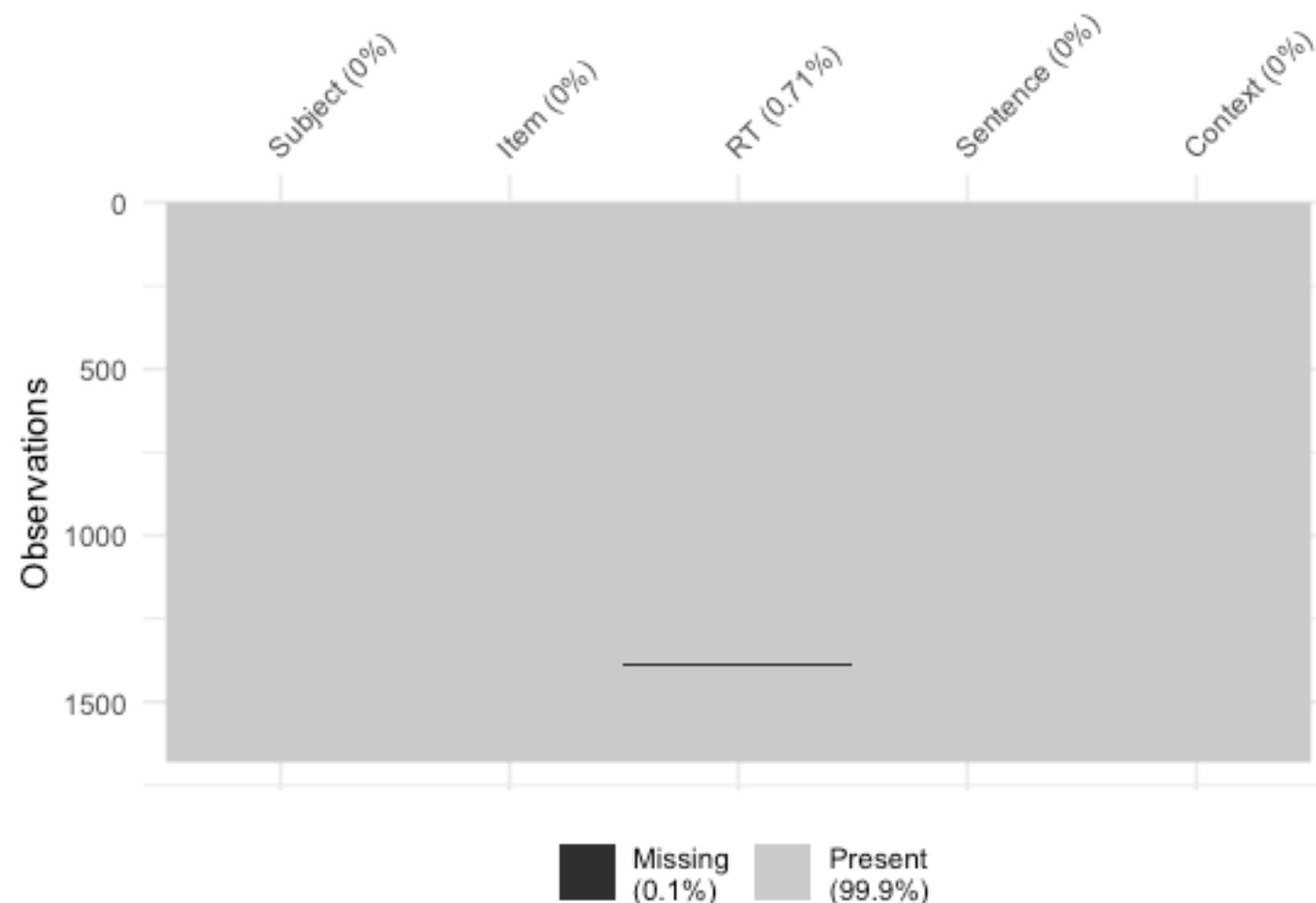


What's going on here?

Let's visualise the whole dataset...

We can use a function in a package without loading it into our library (although we still need to install it) using `package_name::function_name` like this:

```
visdat::vis_miss(fact_data)
```



Let's ignore the missing data (NAs) - one way:

```
fact_data %>%  
  filter(!is.na(RT)) %>%  
  group_by(Context, Sentence) %>%  
  summarise(mean = mean(RT), sd = sd(RT))
```

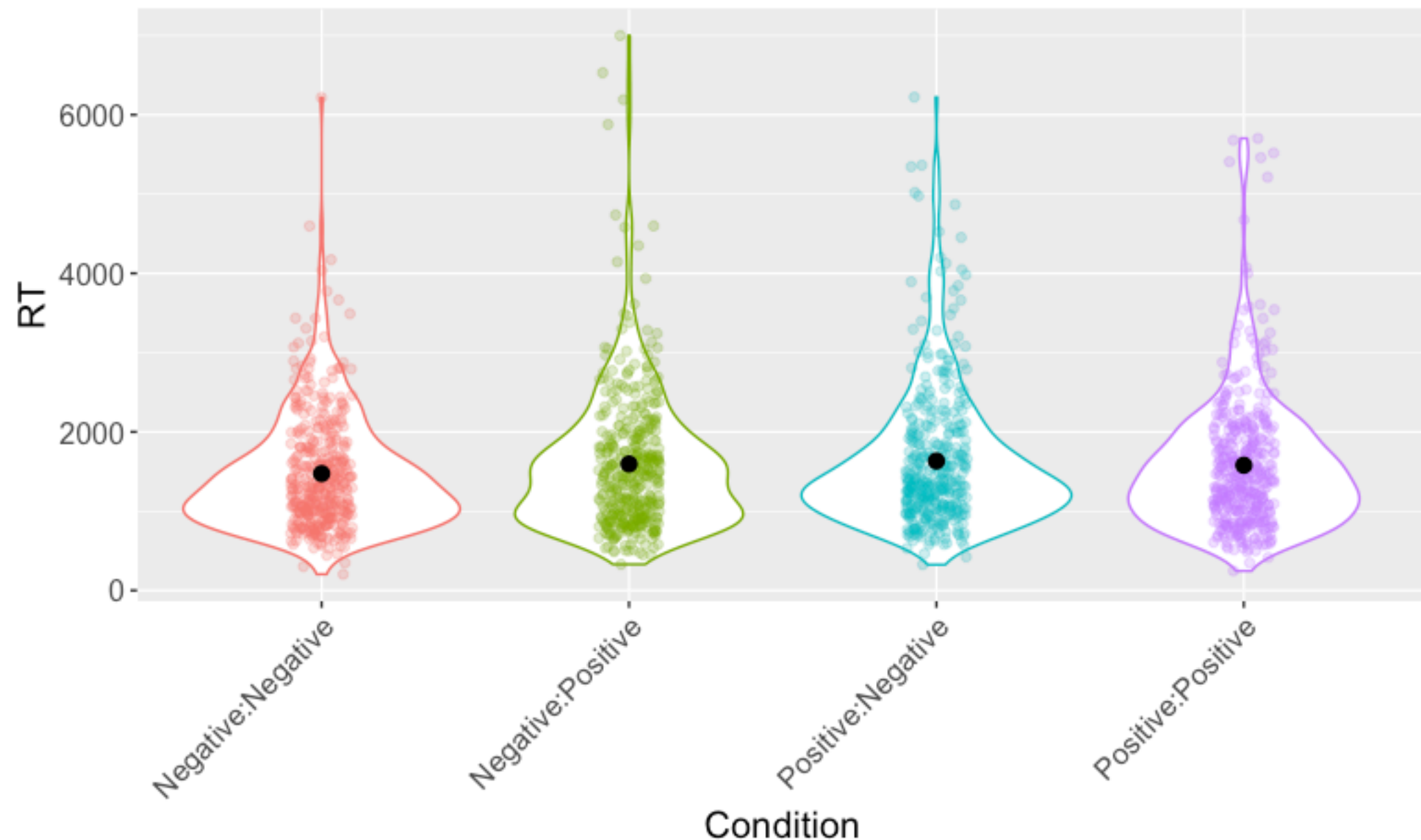
```
# A tibble: 4 x 4  
# Groups:   Context [2]  
  Context Sentence mean    sd  
  <fct>      <fct>   <dbl> <dbl>  
1 Negative Negative 1474.  729.  
2 Negative Positive 1595.  887.  
3 Positive Negative 1633.  877.  
4 Positive Positive 1579.  841.
```

or another way...

```
fact_data %>%  
  group_by(Context, Sentence) %>%  
  summarise(mean = mean(RT, na.rm = TRUE),  
            sd = sd(RT, na.rm = TRUE))
```

```
# A tibble: 4 x 4  
# Groups:   Context [2]  
  Context Sentence mean    sd  
  <fct>      <fct>   <dbl> <dbl>  
1 Negative Negative 1474.  729.  
2 Negative Positive 1595.  887.  
3 Positive Negative 1633.  877.  
4 Positive Positive 1579.  841.
```

```
fact_data %>%
  ggplot(aes(x = Context:Sentence, y = RT, colour = Context:Sentence)) +
  geom_violin() +
  geom_jitter(width = .1, alpha = .25) +
  guides(colour = FALSE) +
  stat_summary(fun.data = "mean_cl_boot", colour = "black") +
  theme(text = element_text(size = 15), axis.text.x = element_text(angle =
45, hjust = 1)) +
  labs(x = "Condition")
```



By Subjects

```
model_subjects <- aov_4(RT ~ Context * Sentence + (1 + Context  
* Sentence | Subject), data = fact_data, na.rm = TRUE)
```

- Syntax corresponds to RT being predicted by the two factors (Context * Sentence) corresponds to two main effects plus the interaction) plus the random effect by Subjects using the datafile called `fact_data`. By setting `na.rm` to be `TRUE`, we are telling the analysis to ignore individual trials where there might be missing data - effectively this calculates the condition means over the data that is present (and ignores trial where it is missing).
- `aov_4` aggregates over the grouping term in the random effect. Simply change to `(1 + Context * Sentence | Item)` for by-item (i.e., F2) analysis. This requires the data to contain the individual observations (not aggregated as means).

```
> anova(model_subjects)
Anova Table (Type 3 tests)
```

```
Response: RT
```

	num	Df	den	Df	MSE	F	ges	Pr(>F)
Context		1		59	90195	3.1767	0.0060231	0.07984 .
Sentence		1		59	124547	0.6283	0.0016524	0.43114
Context:Sentence		1		59	93889	4.5967	0.0090449	0.03616 *

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- The output contains the main effect of Sentence, the main effect of Context, and the interaction between the two. Associated with each are the dfs, the Mean Squared Error, the F ratio, the generalized eta-squared, and p-value. Note, you can ask for partial eta-squared as effect size measure too.

By Items

```
> model_items <- aov_4(RT ~ Context * Sentence + (1 + Context * Sentence | Item),  
data = DV, na.rm = TRUE)
```

```
> anova(model_items_
```

Anova Table (Type 3 tests)

Response: RT

	num	Df	den	Df	MSE	F	ges	Pr(>F)
Context	1		27	39844	4.0013	0.0080150	0.05561	.
Sentence	1		27	203164	0.1221	0.0012553	0.72951	
Context:Sentence	1		27	40168	5.7687	0.0116070	0.02346	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

- With the same datafile and just by changing *one* word in the analysis code.

Interpreting Interactions

We can build the model as before and pass the model to the function `emmeans` (remember to load the `emmeans` package) and ask for pairwise comparisons with no correction - we need to work out the Bonferroni corrected values ourselves...

```
> emmeans(model_subjects, pairwise ~ Context * Sentence, adjust = "none")
```

```
$emmeans
```

Context	Sentence	emmean	SE	df	lower.CL	upper.CL
Negative	Negative	1474	57.8	138	1360	1588
Positive	Negative	1628	57.8	138	1514	1742
Negative	Positive	1595	57.8	138	1481	1709
Positive	Positive	1579	57.8	138	1465	1693

Warning: EMMs are biased unless design is perfectly balanced

Confidence level used: 0.95

```
$contrasts
```

contrast	estimate	SE	df	t.ratio	p.value
Negative, Negative - Positive, Negative	-153.9	55.4	118	-2.779	0.0064
Negative, Negative - Negative, Positive	-120.9	60.3	116	-2.004	0.0474
Negative, Negative - Positive, Positive	-105.2	59.8	115	-1.759	0.0813
Positive, Negative - Negative, Positive	33.0	59.8	115	0.551	0.5824
Positive, Negative - Positive, Positive	48.7	60.3	116	0.807	0.4213
Negative, Positive - Positive, Positive	15.7	55.4	118	0.284	0.7772

Results

We conducted a 2 (Context: Positive vs. Negative) x 2 (Sentence: Positive vs. Negative) repeated measures ANOVA to investigate the influence of context valence on reaction times to words of the same or different valence. The ANOVA revealed no effect of Sentence ($F < 1$), no effect of Context ($F(1, 59) = 3.18, p = .080, \eta_G^2 = .006$), but an interaction between Sentence and Context ($F(1, 59) = 4.60, p = .036, \eta_G^2 = .009$).

The interaction was interpreted by conducting Bonferroni-corrected pairwise comparisons. These comparisons revealed that the interaction was driven by Negative sentences being processed faster in Negative vs. Positive contexts (1,474 ms. vs. 1,628 ms., $t(118) = 2.78, p = .012$) while Positive sentences were read equivalently in Negative vs. Positive contexts (1,595 ms. vs. 1,579 ms., $t(118) = .284, p = 1$).

Now for the lab...