Course Business

- Midterm assignment due **on Canvas** next Monday at 1:30 PM
  - Unsure if an article is suitable? Can run it by me
- Add-on packages to install for today:
  - `performance` (may have gotten this last week)
  - `emmeans`
- Lab materials on Canvas

Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means
- Lab

Unbalanced Factors

- Sometimes, we may have **differing numbers of observations** per level
- Possible reasons:
  - Some categories <u>naturally</u> more common
    - e.g., college majors
  - Categories may be equally common in the population, but we have <u>sampling error</u>
    - e.g., ended up 60% female participants, 40% male
  - Study was <u>designed</u> so that some conditions are more common
    - e.g., more "control" subjects than "intervention" subjects
  - We wanted equal numbers of observations, but lost some because of <u>errors</u> or <u>exclusion criteria</u>
    - e.g., data loss due to computer problems
    - Dropping subjects below a minimum level of performance

Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means
- Lab

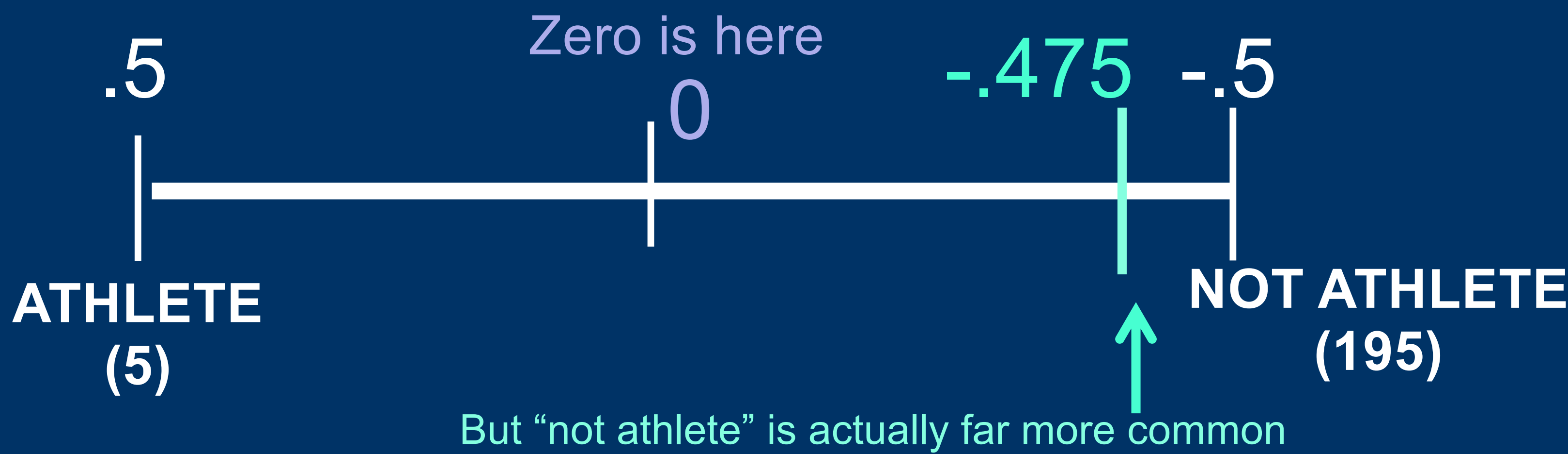
Weighted Coding

- "For the average student, does course size predict probability of graduation?"
  - Random sample of 200 Pitt undergrads
  - 5 are student athletes and 195 are not

- How can we make the intercept reflect the "average student"?
  - We could try to apply effects coding to the StudentAthlete variable by centering around the mean and getting `(0.5, -0.5)`, but…

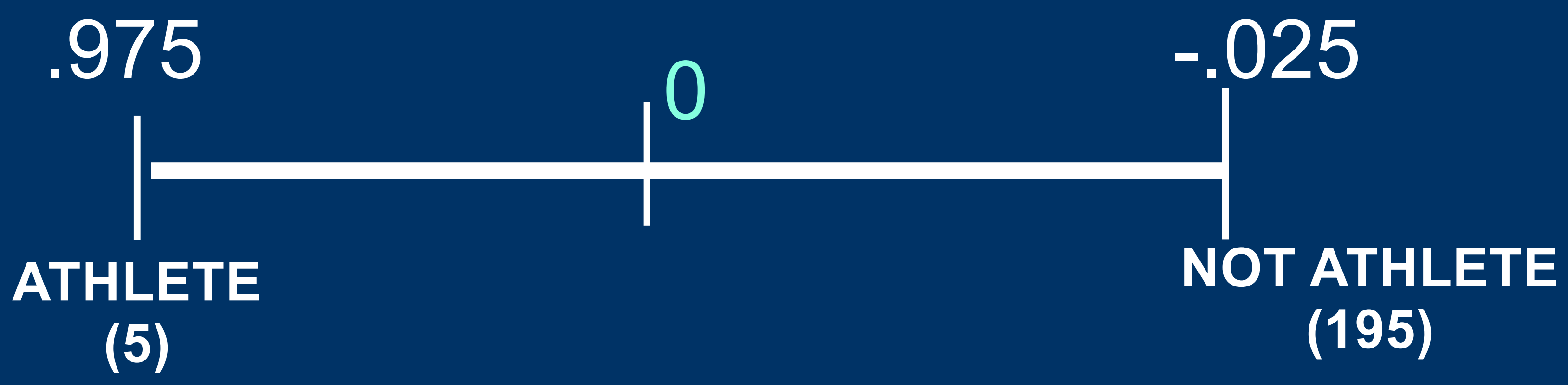
Weighted Coding

- An intercept at 0 would no longer correspond to the overall mean



- As a scale, this would be totally unbalanced
- To fix balance, we need to assign a heavier weight to Athlete

Weighted Coding



- Change codes so the mean *is* 0
  - `c(.975, -.025)`
- `contr.helmert.weighted()` function in my `psycholing` package will calculate this

Weighted Coding

- **Weighted coding:** Change the codes so that the mean is 0 again
  - Used when the imbalance reflects something *real*
  - Like Type II sums of squares

- "For the average student, does course size predict graduation rates?"
  - Average student is *not* a student athlete, and our answer to the question about an "average student" should reflect this!

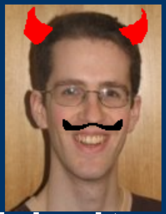
Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means
- Lab

Unweighted Coding

- Last week we looked at `aphasia.csv`:
  - Response times (`RT`) in a sentence verification task
  - Effect of `SubjectType` (aphasia vs. control)
  - Effect of `SentenceType` (active vs. passive)
  - And their interaction

Unweighted Coding

- Oops! Our experiment loaded up the wrong image for one of our Passive sentences ("Groceries")
  - It may have been sabotaged 
  - `UsableItem` column is `No` for this item
- First, can we remove this from our data?
- Some possibilities:
  - `aphasia %>% filter(UsableItem == 'Yes') -> aphasia`
  - `aphasia %>% filter(UsableItem != 'No') -> aphasia2`
  - etc.

Unweighted Coding

- Oops! Our experiment loaded up the wrong image for one of our Passive sentences ("Groceries")
- Now, there's an imbalance, but it's an <u>accident</u> and not meaningful
  - In fact, we'd like to get rid of it!

```
> summary(aphasia2)
    Subject              Item       SubjectType    SentenceType
 S1      : 31   Astronaut: 30   Aphasia:465    Active :480
 S10     : 31   Bear     : 30   Control:465    Passive:450
 S11     : 31   Boy      : 30
 S12     : 31   Breakfast: 30
 S13     : 31   Burglar  : 30
 S14     : 31   Cheese   : 30
 (Other):744   (Other)  :750
```

Unweighted Coding

- Oops! Our experiment loaded up the wrong image for one of our Passive sentences ("Groceries")

- Now, there's an imbalance, but it's an accident and not meaningful
  - In fact, we'd like to get rid of it!

- Retain the (-0.5, 0.5) codes
  - Weights the two conditions equally—because the imbalance isn't meaningful
  - Like Type III sums of squares
  - Probably what you want for factorial experiments

Unbalanced Factors: Summary

- Weighted coding: Change the codes so that the mean is 0
  - Use when the imbalance reflects something *real*
  - Can be done with `contr.helmert.weighted()`

Mean across each individual:

- Unweighted coding: Keep the codes as -0.5 and 0.5
  - Use when the imbalance is an *accident* that we want to eliminate

Mean of the two levels:

Mean of the active sentences

Mean of the passive sentences

- With balanced factors, these are identical

Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means

- Lab

Post-hoc Comparisons

- Maximal model for the aphasia data was:
  - ```
    model.Maximal <- lmer(RT ~
    1 + SentenceType * SubjectType +
    (1 + SentenceType|Subject) +
    (1 + SubjectType|Item),
    data = aphasia)
    ```
- This didn't converge:
  - boundary (singular) fit: see ?isSingular

```
Random effects:
 Groups   Name                Variance Std.Dev. Corr
 Item     (Intercept)         44860.91 211.804
          SubjectTypeAphasia     71.18   8.437  -1.00
 Subject  (Intercept)         43286.19 208.053
          SentenceTypePassive  1350.57  36.750   0.18
 Residual                      6861.01  82.831
Number of obs: 930, groups:  Item, 31; Subject, 30
```

Probably overparameterized
```

Post-hoc Comparisons

- Maximal model for the aphasia data was:
  - ```
    model.Maximal <- lmer(RT ~
    1 + SentenceType * SubjectType +
    (1 + SentenceType|Subject) +
    (1 + SubjectType|Item),
    data = aphasia)
    ```
- So, let's simplify:
  - ```
    model2<- lmer(RT ~
    1 + SentenceType * SubjectType +
    (1 + SentenceType|Subject) +
    (1|Item),
    data = aphasia)
    ```
- Doesn't seem to harm model fit—*p* > .20
  - `anova(model.Maximal, model2)`

	npar	AIC	BIC	logLik	deviance	Chisq	Df	Pr(>Chisq)
model2	9	11214	11257	-5597.9	11196			
model.Maximal	11	11216	11269	-5596.8	11194	2.2473	2	0.3251

Post-hoc Comparisons

- With treatment coding, we get estimates of simple-effects:

Intercept: RT for healthy controls, active sentences

```
Random effects:
 Groups   Name                  Variance Std.Dev. Corr
 Item     (Intercept)           43090    207.58
 Subject  (Intercept)           43286    208.05
          SentenceTypePassive    1348     36.72   0.18
 Residual                        6879     82.94
Number of obs: 930, groups:  Item, 31; Subject, 30

Fixed effects:
                                     Estimate Std. Error      df t value Pr(>|t|)
(Intercept)                           1716.01      74.88   56.54  22.916  < 2e-16 ***
SentenceTypePassive                    553.44      75.60   30.23   7.321 3.57e-08 ***
SubjectTypeAphasia                      84.52      76.35   28.00   1.107    0.278
SentenceTypePassive:SubjectTypeAphasia 188.30      17.27   28.00  10.904 1.38e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) SntnTP SbjcTA
SntncTypPss -0.465
SbjctTypAph -0.510 -0.011
SntncTP:STA -0.049 -0.114  0.095
```

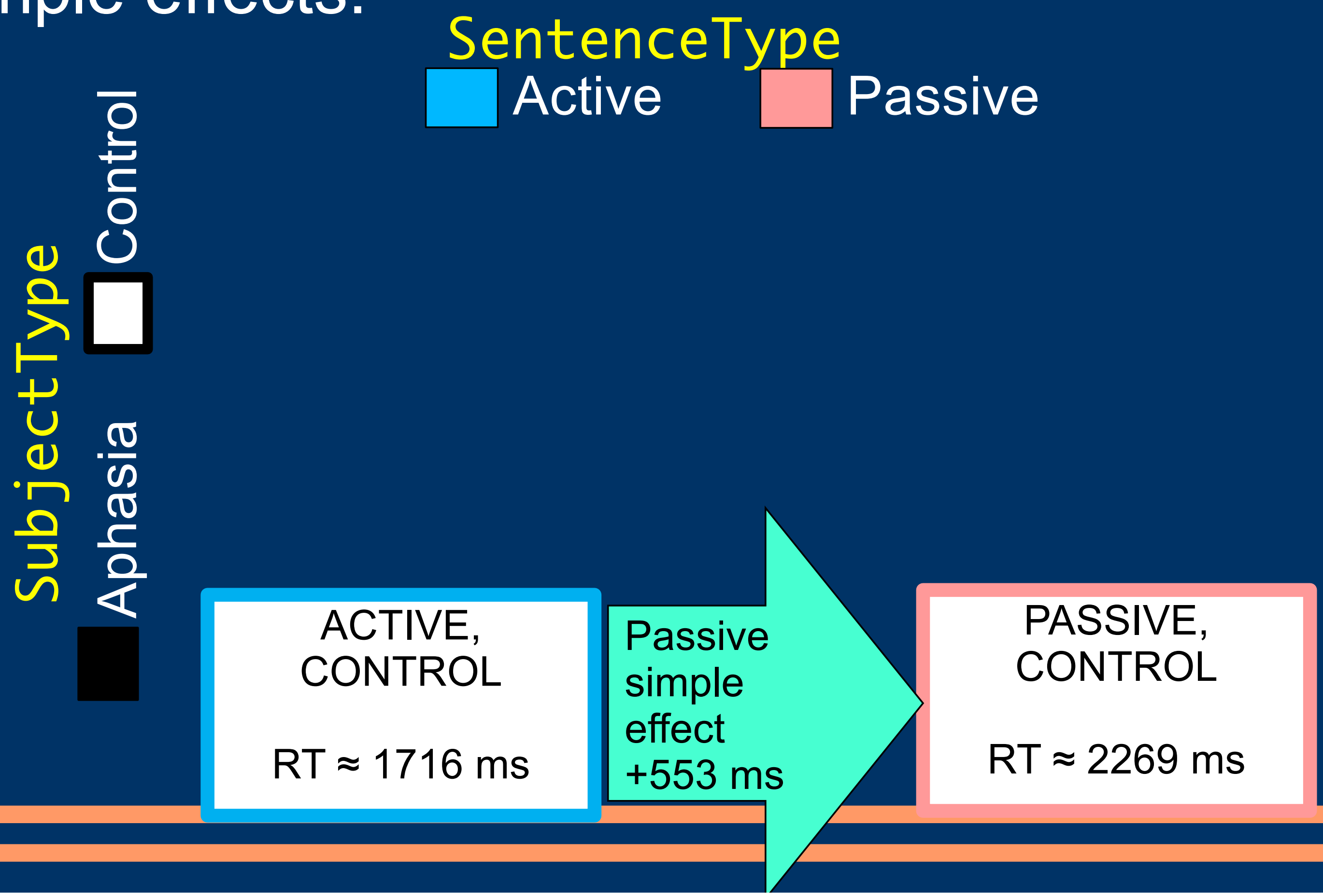
Significant RT difference for passive sentences (among healthy controls)

Not a significant RT difference for aphasics (among active sentences)

Significant special effect of aphasia + passive sentence

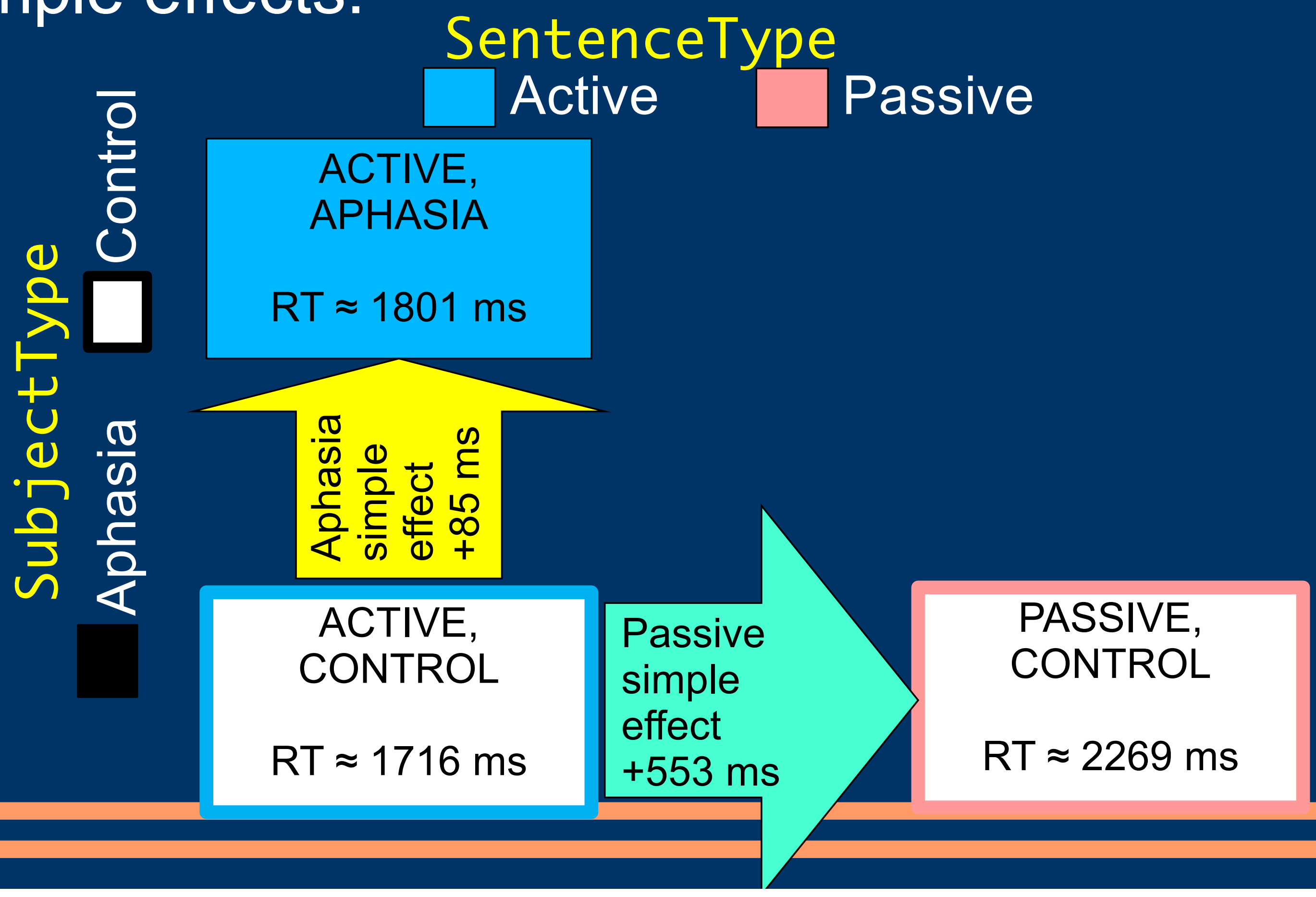
Post-hoc Comparisons

- The estimates from a model are enough to *fully describe* differences among conditions
- With simple effects:



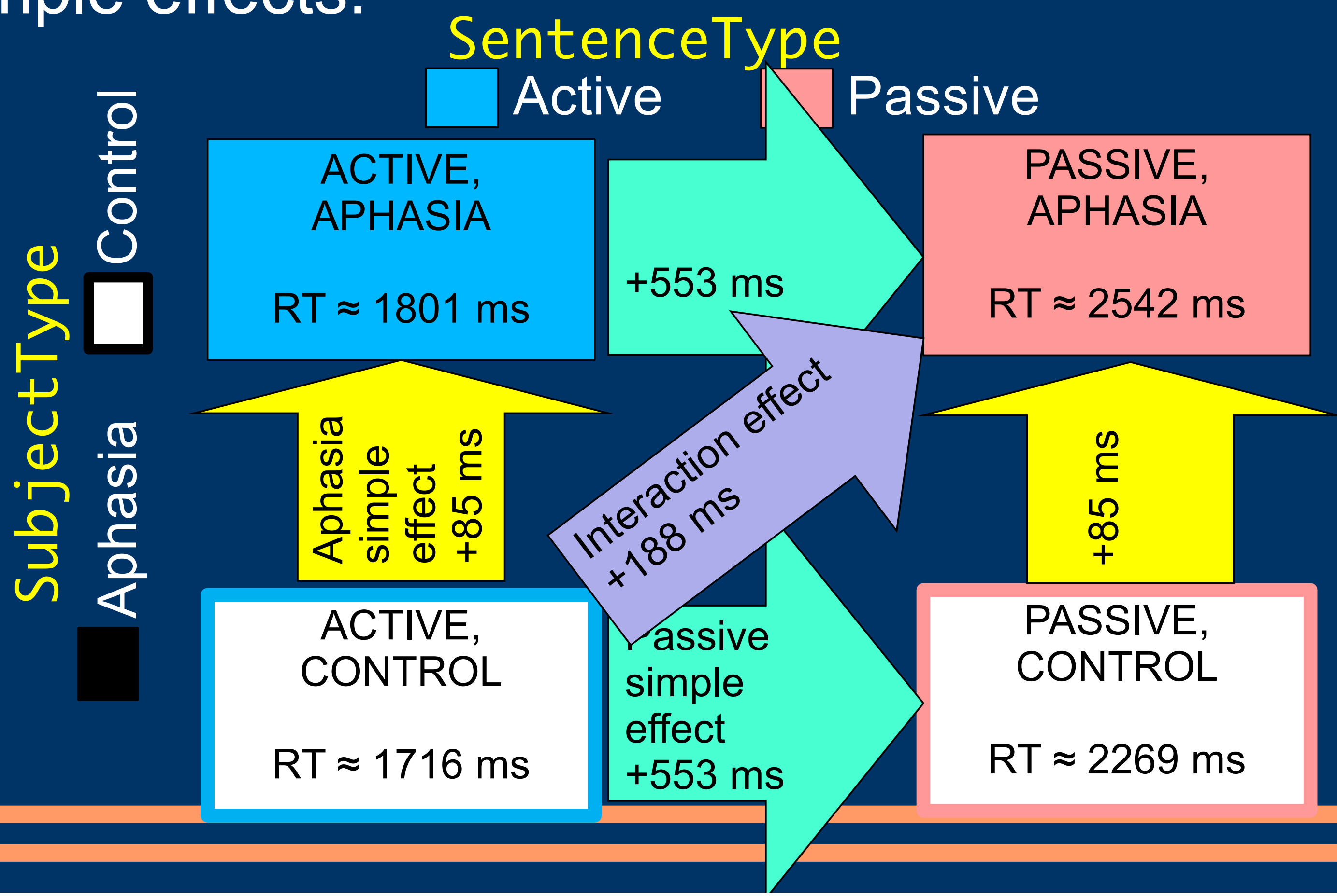
Post-hoc Comparisons

- The estimates from a model are enough to *fully describe* differences among conditions
- With simple effects:



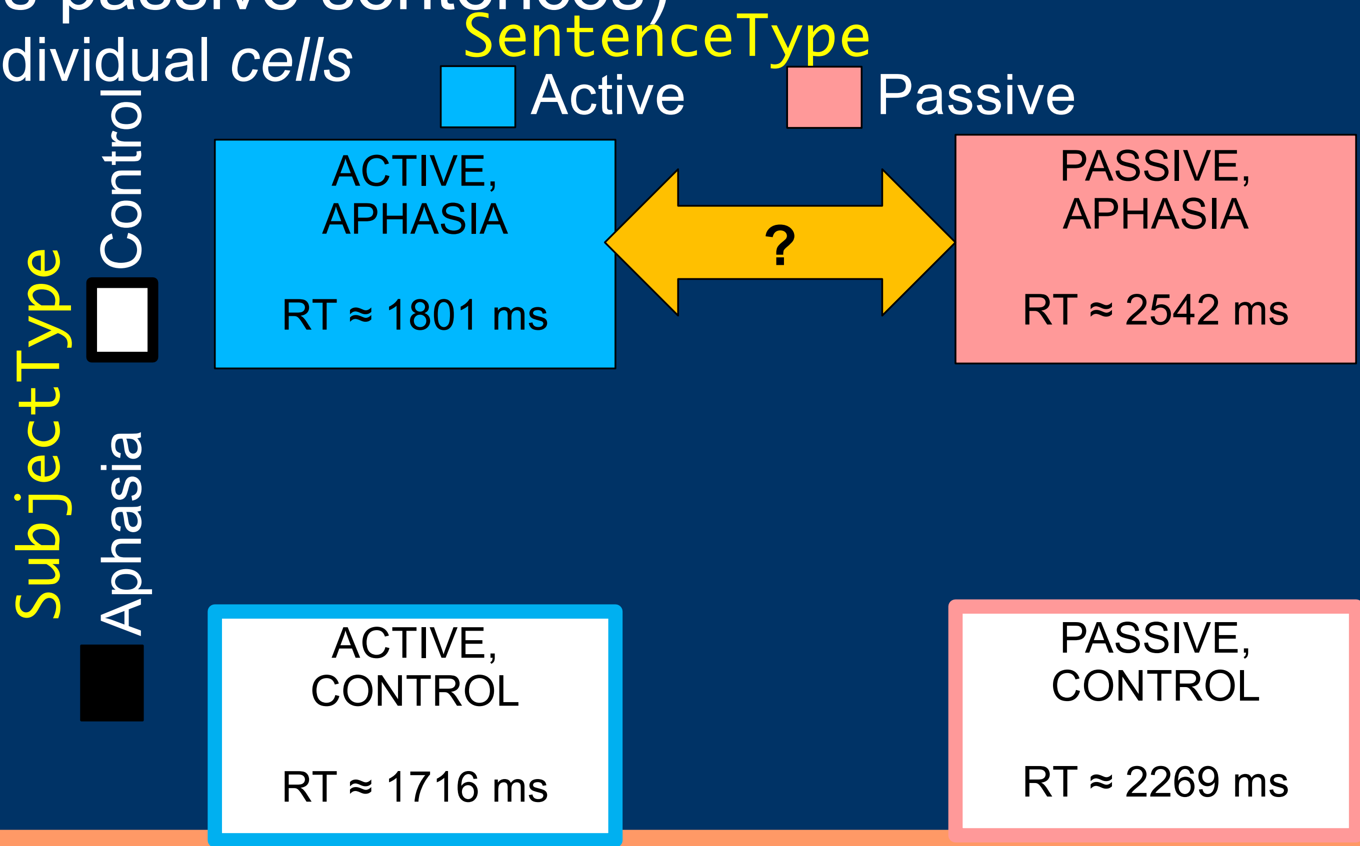
Post-hoc Comparisons

- The estimates from a model are enough to *fully describe* differences among conditions
- With simple effects:



Post-hoc Comparisons

- But, sometimes we want to compare individual combinations (e.g., people w/ aphasia seeing active vs passive sentences)
  - i.e., individual *cells*



Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means
- Lab

Post-hoc Comparisons: Tukey Test

- But, sometimes we want to compare individual combinations (e.g., people w/ aphasia seeing active vs passive sentences)
  - i.e., individual *cells*

Name of the *model* not the original dataframe

- `emmeans(Model.Maximal,pairwise~SentenceType*SubjectType)`
  - Requires `emmeans` package to be loaded
    - `library(emmeans)`
  - Uses Tukey test to *correct* for multiple comparisons so overall α still = .05
  - Which two cells *don't* significantly differ?

The independent variables (for now, all of them)

$contrasts

contrast	estimate	SE	df	t.ratio	p.value
Active Control - Passive Control	-553.4	77.1	29.9	-7.179	<.0001
Active Control - Active Aphasia	-84.5	76.4	28.0	-1.107	0.6885
Active Control - Passive Aphasia	-826.3	108.0	57.2	-7.650	<.0001
Passive Control - Active Aphasia	468.9	108.1	57.2	4.339	0.0003
Passive Control - Passive Aphasia	-272.8	79.9	28.0	-3.415	0.0100
Active Aphasia - Passive Aphasia	-741.7	74.1	29.9	-10.010	<.0001

Comparisons of each pair of cells

Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means
- Lab

Estimated Marginal Means

- `emmeans` also returns estimated means and std. errors for each cell of the design

- EMMs represent what the means of the different groups would look like *if* they didn't differ in other (fixed or random) variables

```
$emmeans
 SentenceType SubjectType emmean   SE   df lower.CL upper.CL
 Active       Control       1716 75.6 56.4     1565     1867
 Passive      Control       2269 78.6 56.3     2112     2427
 Active       Aphasia       1801 74.2 56.1     1652     1949
 Passive      Aphasia       2542 77.1 56.0     2388     2697
```

Estimated Marginal Means

- As an example, let's add `SentenceLength` as a covariate
  - Task was to read the sentence & judge whether it matches a picture, so length of sentence would plausibly affect time needed to do this
- *Not* perfectly balanced across conditions

	SentenceType	SubjectType	SentenceLength_M	RT_M
	<fct>	<fct>	<dbl>	<dbl>
1	Active	Control	10.2	1716.
2	Active	Aphasia	10.2	1801.
3	Passive	Control	9.53	2269.
4	Passive	Aphasia	9.53	2542.

Estimated Marginal Means

- New model & estimated marginal means:
  - `model.Length <- lmer(RT ~ 1 + SentenceType * SubjectType + SentenceLength + (SentenceType|Subject) + (1|Item), data=aphasia)`
  - `emmeans(model.Length, pairwise~SentenceType*SubjectType + SentenceLength)`

$emmeans

SentenceType	SubjectType	SentenceLength	emmean	SE	df	lower.CL	upper.CL
Active	Control	9.9	1676	54.3	28.7	1564	1787
Passive	Control	9.9	2313	56.8	28.7	2196	2429
Active	Aphasia	9.9	1760	54.3	28.7	1649	1871
Passive	Aphasia	9.9	2585	56.8	28.7	2469	2702

What we'd expect *if* the sentences were all of equal length

	SentenceType	SubjectType	SentenceLength_M	RT_M
	<fct>	<fct>	<dbl>	<dbl>
1	Active	Control	10.2	1716.
2	Active	Aphasia	10.2	1801.
3	Passive	Control	9.53	2269.
4	Passive	Aphasia	9.53	2542.

Our raw data (where the average sentence length differs slightly across conditions)

Estimated Marginal Means

- EMMs are a hypothetical
  - What the means of the different groups would look like *if* they didn't differ in this covariate
  - Like **unweighted** coding in the case of missingness

- Based on our statistical model, so if our model is wrong (e.g., we picked the wrong covariate), the adjusted means will be too
  - Unlike the raw sample means
  - Need to use some caution in interpreting them
  - Be clear what you are reporting

Estimated Marginal Means

- Also possible to *test* whether each of these estimated cell means significantly differs from 0
  - `ls_means(model.Length)`
  - Silly in case of RTs, but could be relevant for some other DVs (e.g., preference)

Least Squares Means table:

	Estimate	Std. Error	df	t value	lower	upper	Pr(>|t|)	
SentenceTypeActive	1717.838	38.670	29.4	44.423	1638.801	1796.876	< 2.2e-16	***
SentenceTypePassive	2448.983	40.440	29.4	60.559	2366.324	2531.641	< 2.2e-16	***
SubjectTypeControl	1994.074	55.080	28.4	36.204	1881.312	2106.837	< 2.2e-16	***
SubjectTypeAphasia	2172.747	55.080	28.4	39.447	2059.985	2285.509	< 2.2e-16	***
SentenceTypeActive:SubjectTypeControl	1675.578	54.338	28.7	30.836	1564.398	1786.757	< 2.2e-16	***
SentenceTypePassive:SubjectTypeControl	2312.571	56.833	28.7	40.690	2196.283	2428.860	< 2.2e-16	***
SentenceTypeActive:SubjectTypeAphasia	1760.099	54.338	28.7	32.392	1648.919	1871.279	< 2.2e-16	***
SentenceTypePassive:SubjectTypeAphasia	2585.394	56.833	28.7	45.491	2469.106	2701.683	< 2.2e-16	***

Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means
- Lab

Comparing Marginal Means

- `emmeans` can also test marginal means:
  - `emmeans(model2,pairwise~SubjectType)`

Now, include just *one* variable (for which we want marginal means)

- Effect of one variable *averaging over* the other
  - e.g., aphasic participants (averaging over all sentence types) vs. controls (averaging over all sentence types)
  - These are what *main effects* are testing

```
$emmeans
 SubjectType emmean   SE   df lower.CL upper.CL
 Control       1993 66.4 49.5     1859     2126
 Aphasia       2171 66.4 49.5     2038     2305

Results are averaged over the levels of: SentenceType
Degrees-of-freedom method: kenward-roger
Confidence level used: 0.95
```

Week 8.1: Post-Hoc Comparison

- Unbalanced Factors
  - Weighted Coding
  - Unweighted Coding
- Post-Hoc Comparisons
  - Tukey Test
  - Estimated Marginal Means
  - Comparing Marginal Means
- Lab