

PSY 503: Foundations of Statistical Methods in Psychological Science

More LM: Categorical Predictors

Jason Geller, Ph.D. (he/him/his)

Princeton University

Updated:2022-11-02

Outline

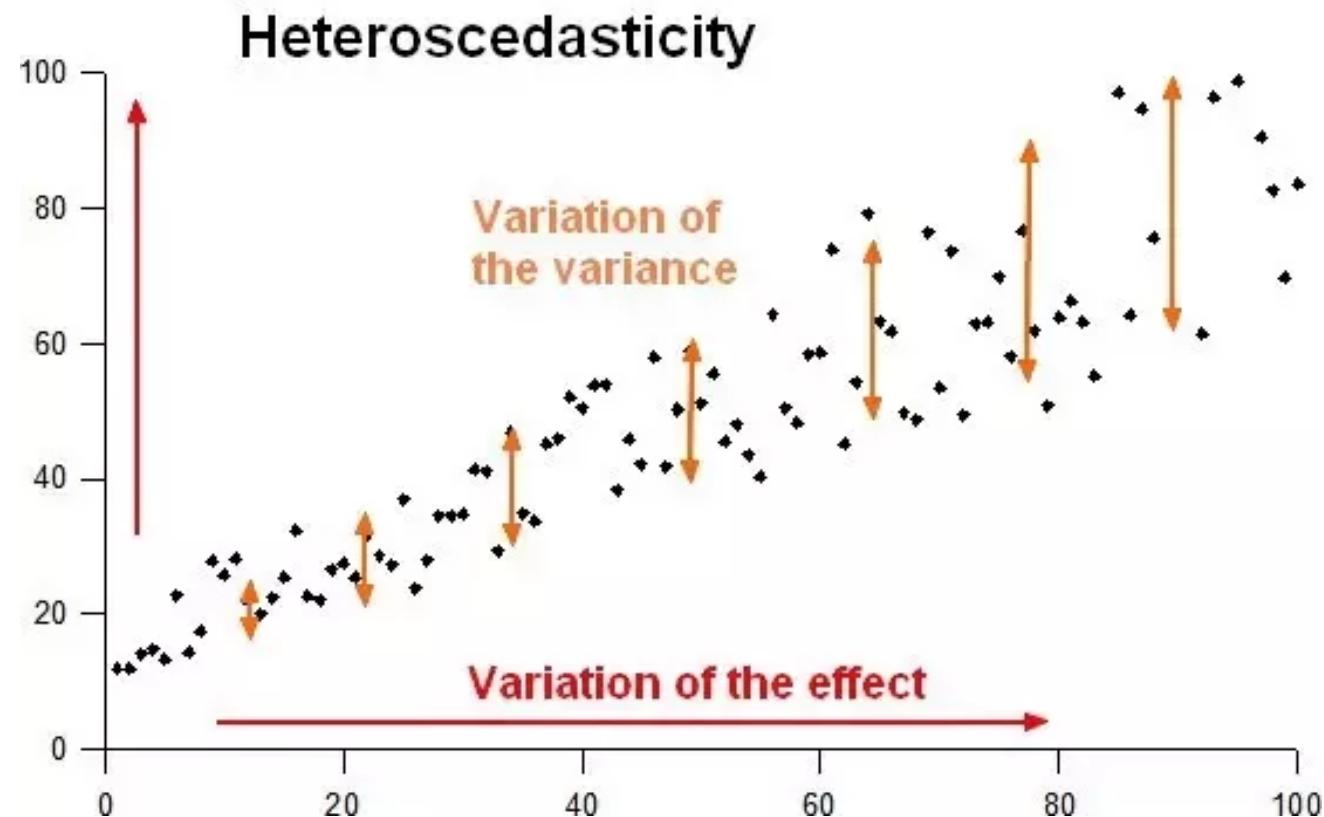
- Check-in questions
- Effect size and power in MR
- Categorical predictors
 - How linear modeling is related to t-tests/ANOVAs
 - Contrast coding for categorical predictors with two means
 - Contrast coding for categorical predictors with three means or more

Check-in Questions

- What the heck is heteroskedasticity?

Heteroskedasticity

- Non-constant error (residual) variance
 - Residuals and predictors are correlated



Heteroskedasticity

- Consequences:
 - Causes standard errors (SE) to be unreliable
 - Increased Type 1 and Type 2 Error
- Solution:
 - Easiest would be to use robust methods for the SE

```
library(performance)
model_parameters(model2) %>%
  flextable() %>%
  autofit()
```

Parameter	Coefficient	SE	CI	CI_low	CI_high
(Intercept)	55.16373592	3.88904610	0.95	47.5059719	62.8215189
PIL_total	-0.38157775	0.03384062	0.95	-0.4482119	-0.3149337
AUDIT_TOTAL_NEW	-0.09210892	0.09391092	0.95	-0.2770251	0.0928079
DAST_TOTAL_NEW	1.03415391	0.39871264	0.95	0.2490649	1.8231889

```
model_parameters(model2, vcov = "HC3") %>% #get corrected
flextable() %>%
  autofit()# robust params
```

Parameter	Coefficient	SE	CI	CI_low	CI_high
(Intercept)	55.16373592	4.20877887	0.95	46.87639900	63.45112985
PIL_total	-0.38157775	0.03597990	0.95	-0.45242431	-0.31073150
AUDIT_TOTAL_NEW	-0.09210892	0.08647293	0.95	-0.26237929	0.07815145
DAST_TOTAL_NEW	1.03415391	0.48095976	0.95	0.08711542	1.97918897

Check-in Questions

- What the heck is heteroskedasticity?
- Assumptions of simple linear model

Simple Linear Modeling Assumptions

- Normality of residuals
- Missingness
- Outliers

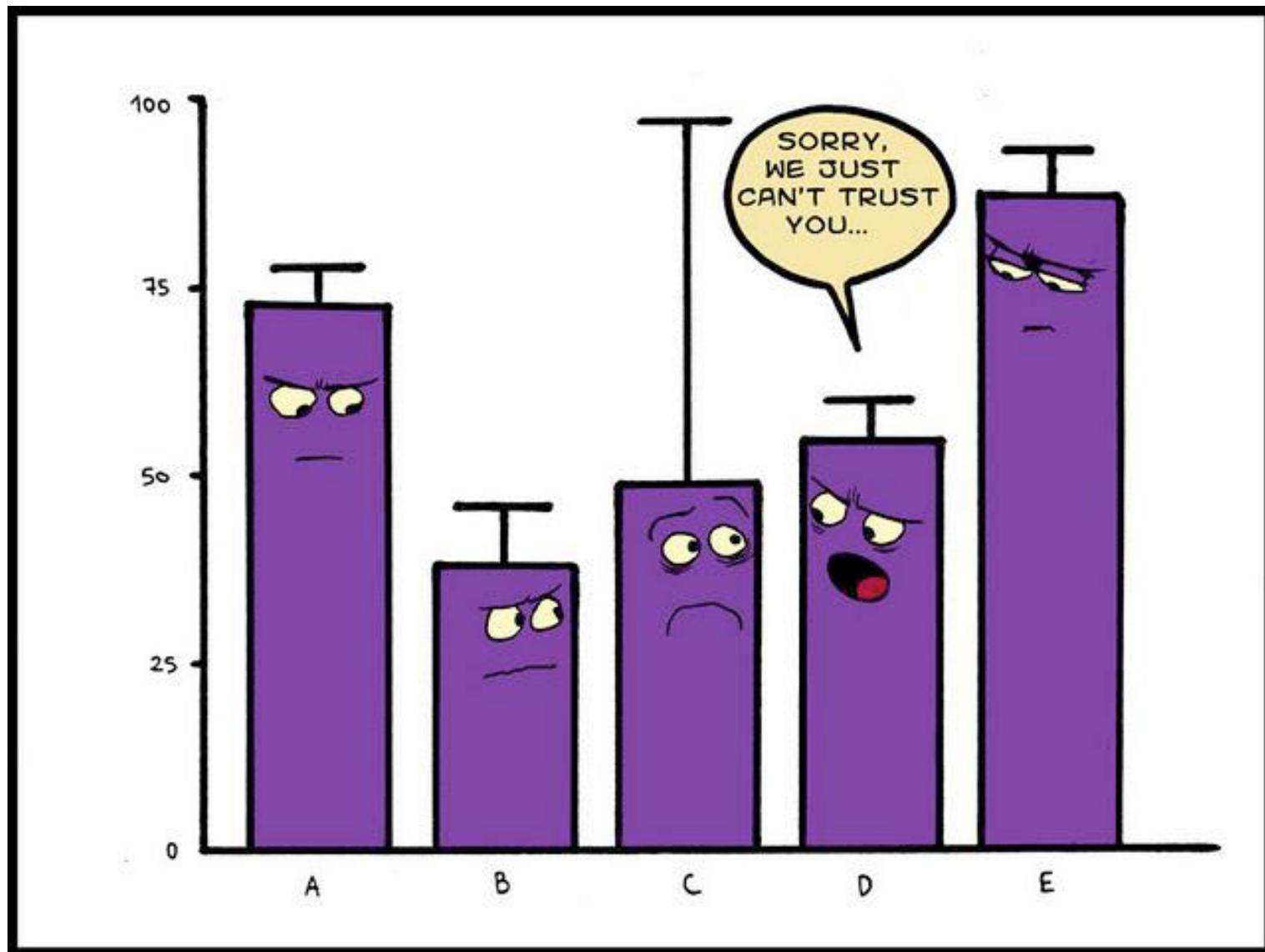
Burning Questions

- What the heck is heteroskedasticity?
- Assumptions of simple linear model
- Difference between ϵ and residual

Errors

- Error is basically an unobservable quantity in our model
 - We use residuals as a proxy

Outliers

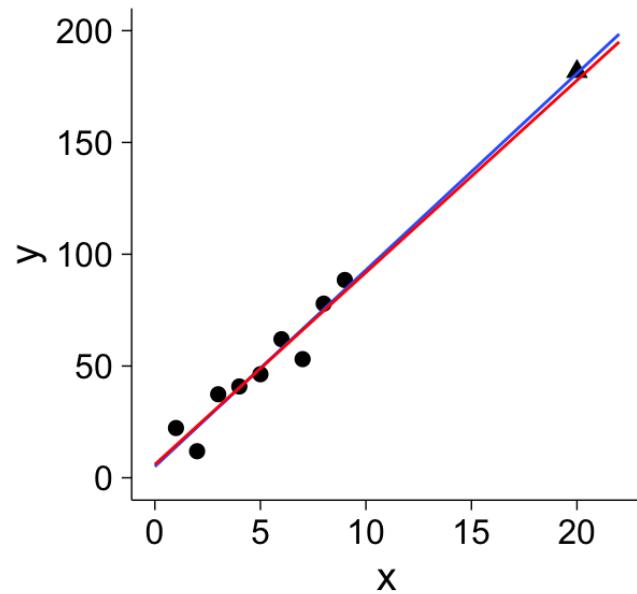


Outliers: Discrepancy

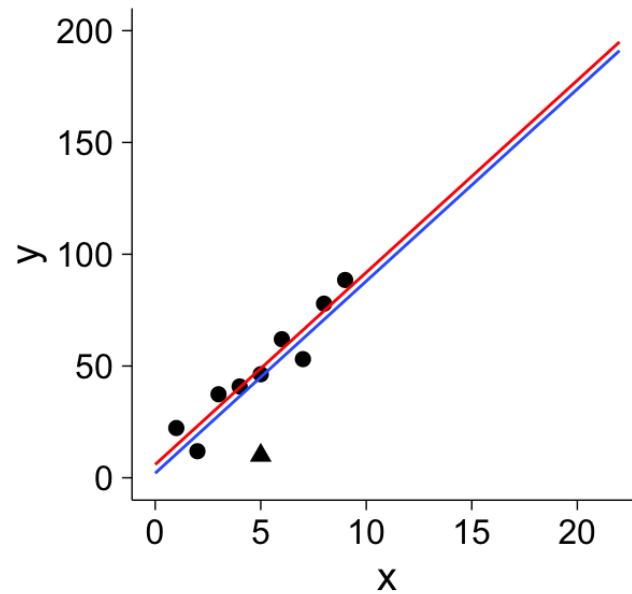
A data point that is unusual in the context of the least squares model

$$e_i^* = \frac{e_i}{S_{E(-i)} \sqrt{1 - h_i}}$$

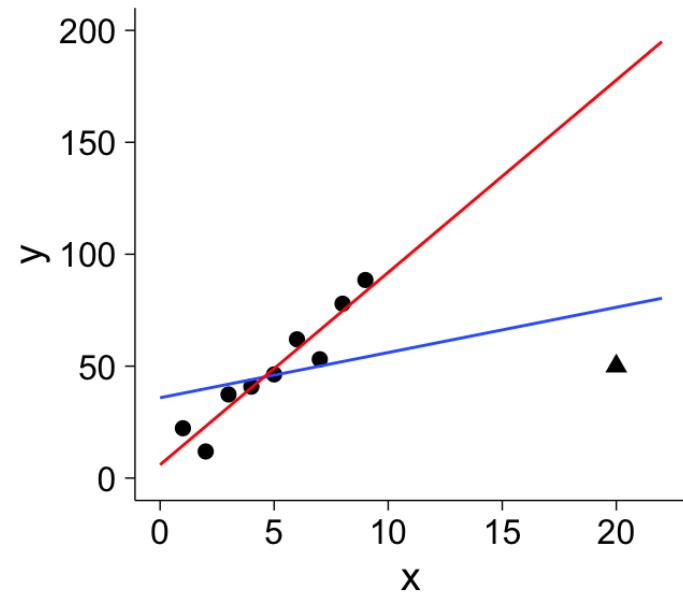
A High leverage



B High discrepancy



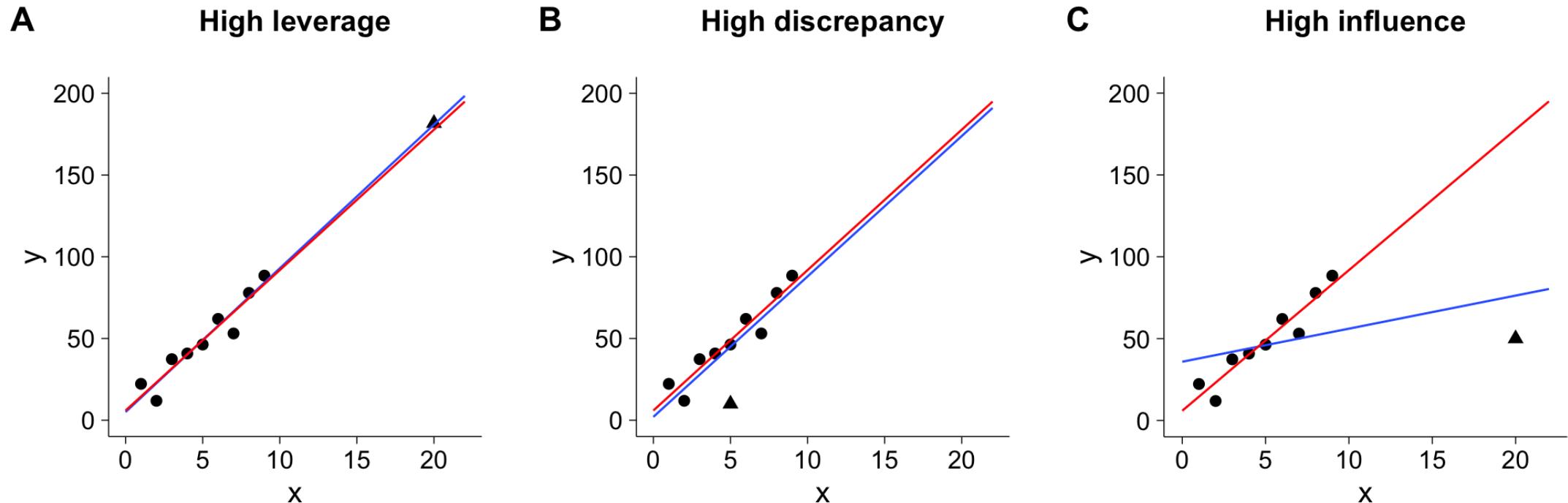
C High influence



Outliers: Leverage

Leverage measures how far a data point is from the mean

$$h_i = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{j=1}^n (x_j - \bar{x})^2} \quad \text{and} \quad \bar{h} = \frac{k}{n} \quad \text{and} \quad \frac{1}{n} \leq h_i \leq 1.0$$



Cook's Distance

- A measure of how much of an effect a single data point has on the whole model
- Often described as leverage + discrepancy (residuals)

$$D_i = \frac{e_i^2}{k \times \frac{1}{n} \sum e_i^2} \times \frac{h_i}{1 - h_i}$$

$$e_i^* = \frac{e_i}{S_{E(-i)} \sqrt{1 - h_i}}$$

- How do we calculate how much change is bad?

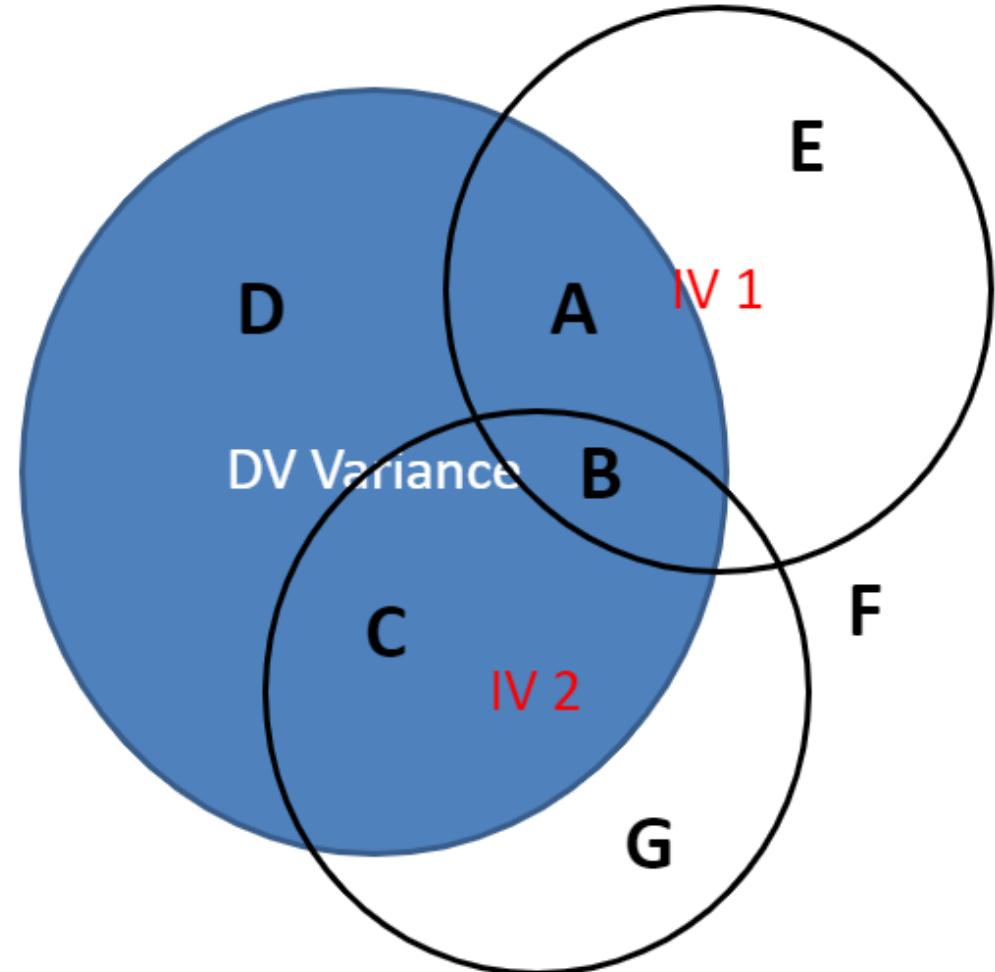
- $\frac{4}{N-K-1}$

Calculate in R

```
augment(model) # in broom package
```

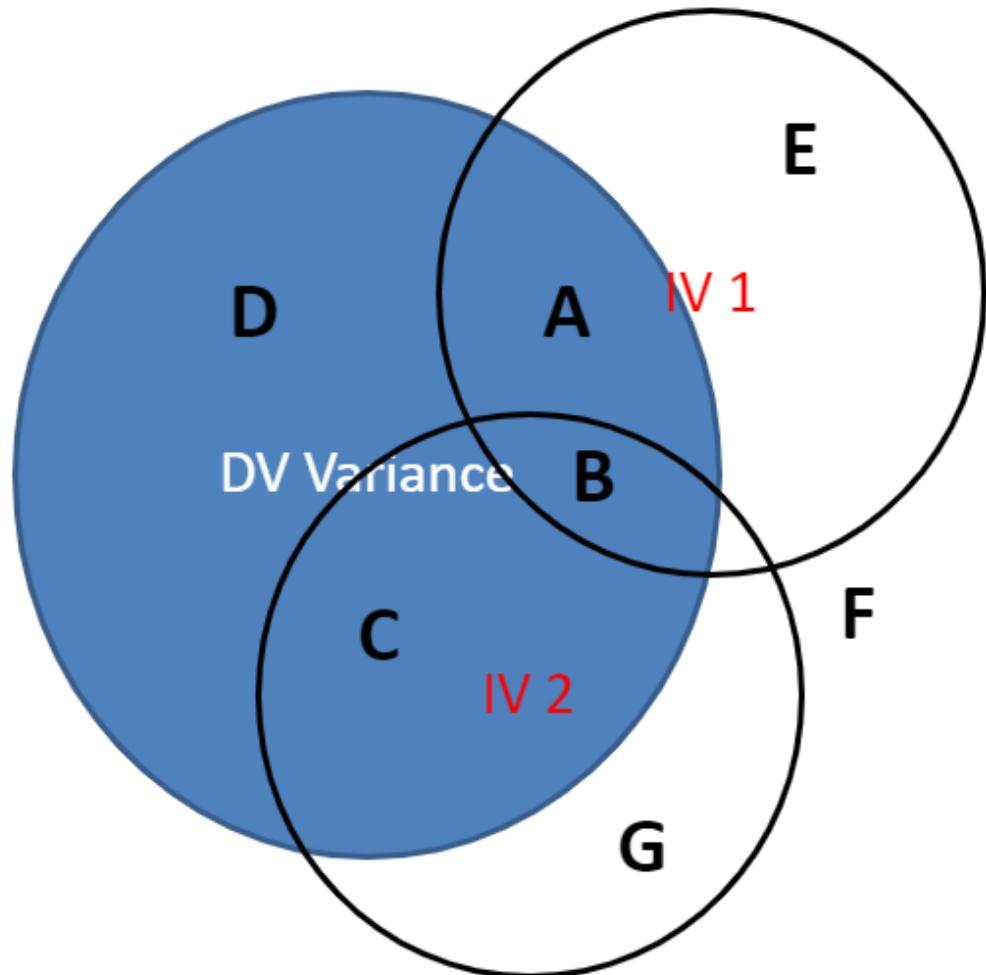
The Impact of Individual Predictors on the Model: Effect Size

- R is the multiple correlation
- R^2 is the multiple correlation squared
- All overlap in Y, used for overall model
- $A + B + C / (A + B + C + D)$



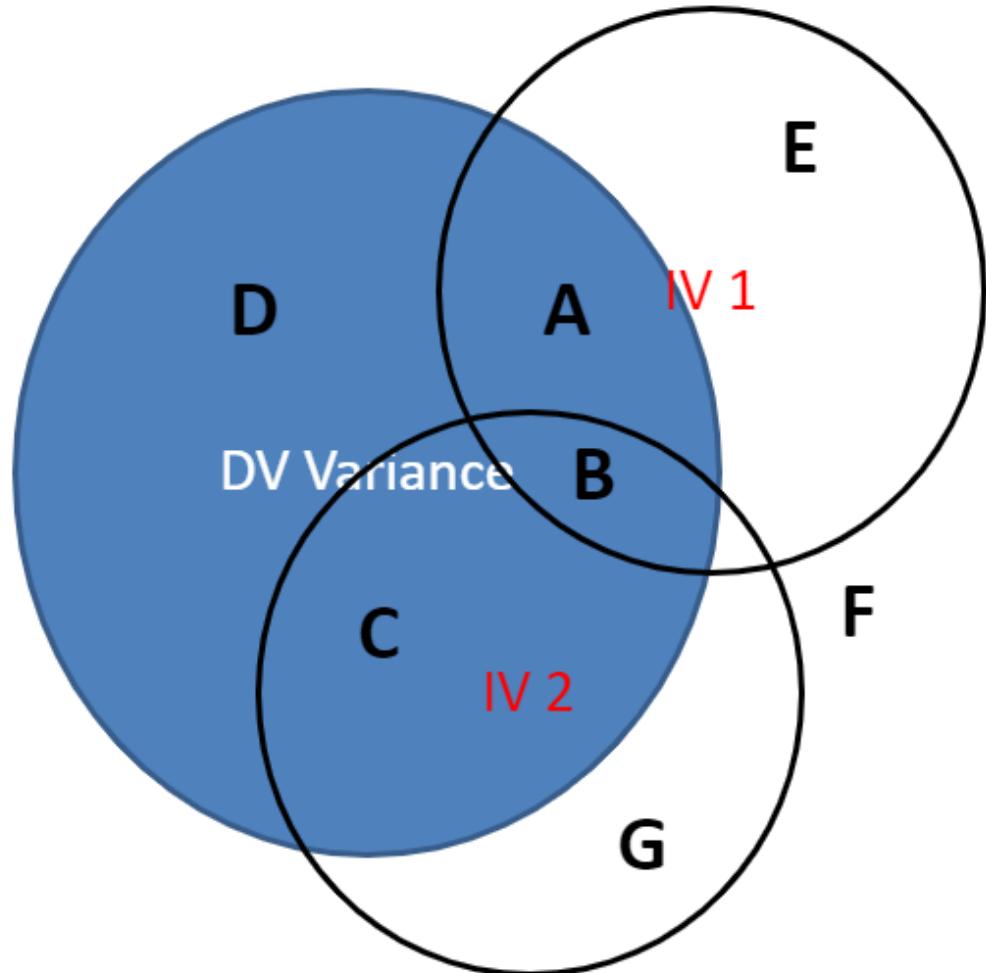
Effect Size

- sr is the semi-partial correlations
- Unique contribution of IV to R^2 for those IVs
- Increase in proportion of explained Y variance when X is added to the equation
- $A/(A + B + C + D)$



Effect Size

- r_p is the partial correlation
- Partial correlation asks how much of the Y variance, which is not estimated by the other IVs, is estimated by this variable.
- $A/(A + D)$
- Removes the shared variance of the control variable (Say X_2) from both Y and X_1
- $r_p > sr$



Partial Correlations

- We would add these to our other reports:

- Meaning: $b = -0.38$, 95% CI $[-0.45, -0.31]$, $t(262) = -11.28$, $p < .001$, ($pr^2 = .30$)`
- Alcohol: $b = -0.09$, 95% CI $[-0.28, 0.09]$, $t(262) = -0.98$, $p = .328$, ($pr^2 < .01$)`
- Drugs: $b = 1.03$, 95% CI $[0.25, 1.82]$, $t(262) = 2.59$, $p = .010$, ($pr^2 < .01$)`

```
library(ppcor)
partialis <- pcor(master)
partialis$estimate^2
#spcor.test (semi-partial)
```

Multiple Regression: Power

- We can use the `pwr` library to calculate the required sample size for any particular effect size
- First, we need to convert the R^2 value to f^2 , which is a different effect size, not the ANOVA F

```
library(pwr)
r2=glance(model2)
```

```
R2= r2$r.squared
f2 <- R2 / (1-R2)
R2
```

```
## [1] 0.3586693
```

```
f2
```

```
## [1] 0.559258
```

Multiple Regression: Power

- `u` is degrees of freedom for the model, first value in the F-statistic
- `v` is degrees of freedom for error, but we are trying to figure out sample size for each condition, so we leave this one blank.
- `f2` is the converted effect size.
- `sig.level` is our α value
- `power` is our power level
- The final sample size is $v + k + 1$ where k is the predictors

```
#f2 is cohen f squared
pwr.f2.test(u = model2$df[1],
             v = NULL, f2 = f2,
             sig.level = .05, power = .80)
```

Categorical Predictors

Modeling Categorical Variables

- Today's Dataset
 - Winter(2016)
 - Are smell words (e.g., *rancid*) rated as more negative/unpleasant than taste words (e.g., *sweet*)?
 - 1 to 9 rating scale

```
library(tidyverse)
senses<- read_csv("data/winter_2016_senses_valence.csv")

senses_filt <- senses %>%
  filter(Modality=="Taste" | Modality=="Smell")
```

Linear Model

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

- So far predictor variable has been continuous
- We can also use linear modeling for categorical variables

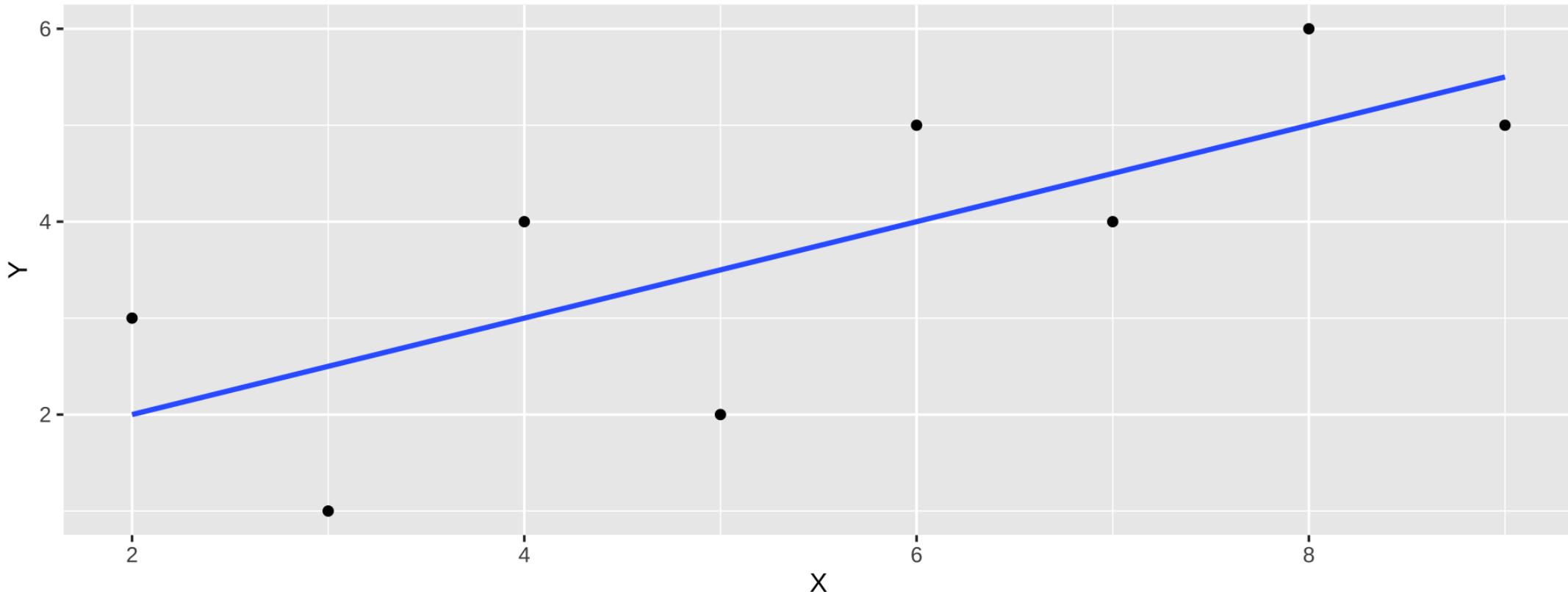
Categorical Variables

- Terminology
 - *Factor*: a variable with a fix set of categories
 - *Levels*: The individual categories within a factor
- In our dataset, what is the factor and what are its levels?

Linear Modeling and t-tests/ANOVAs

What do we do in linear modeling?

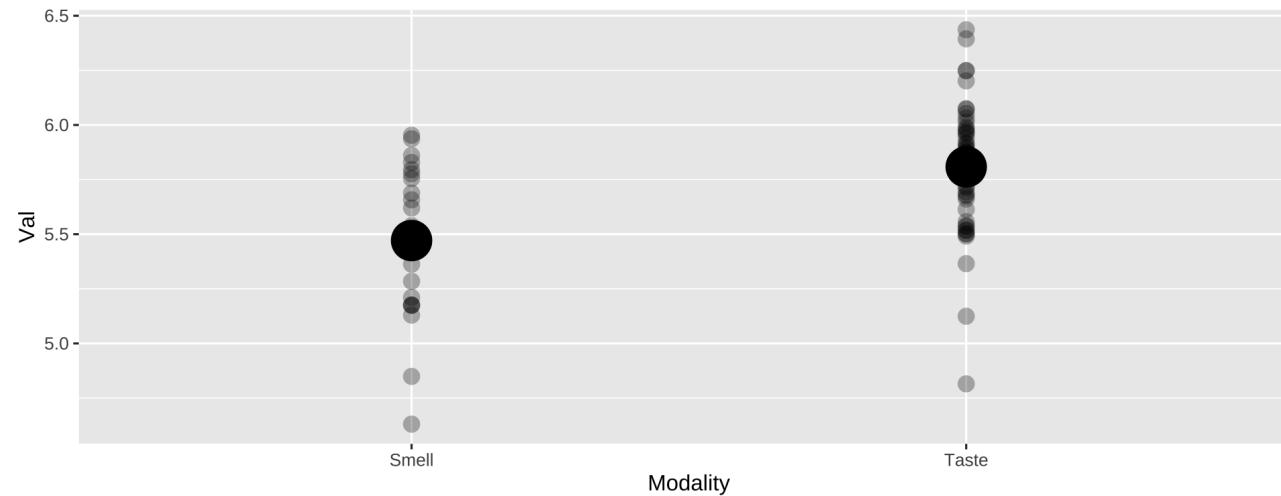
- Fit a line (least squares method)



Linear Modeling and t -tests/ANOVAs

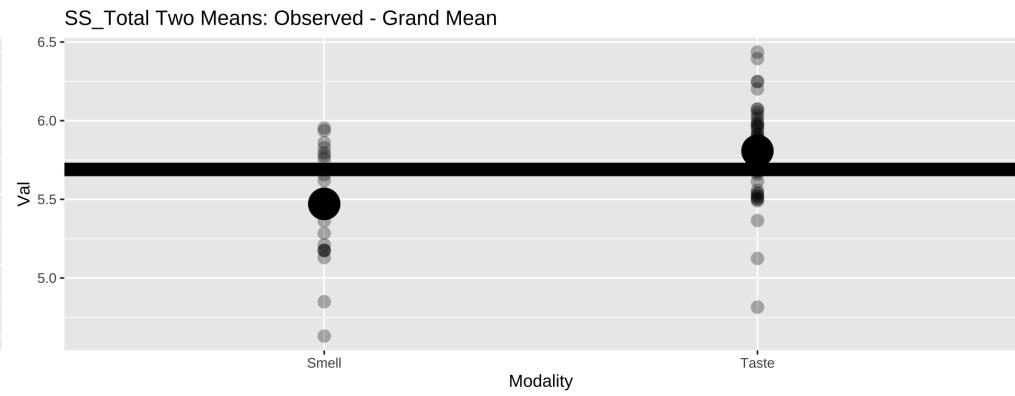
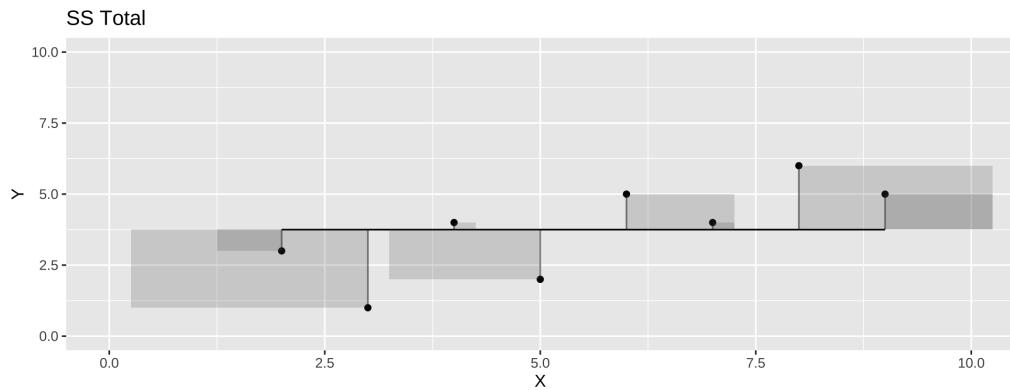
Within a t-test/ANOVA framework we want to know if means differ between groups

```
t.test() # test mean differences
```



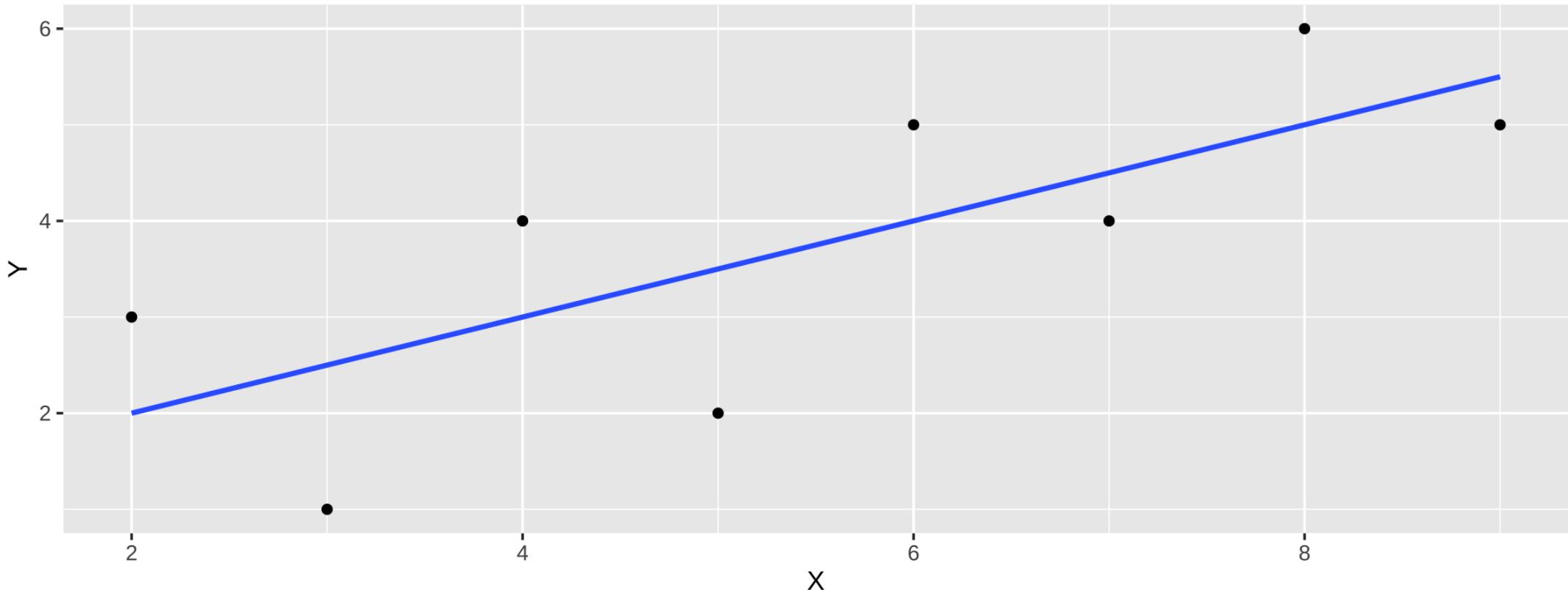
Linear Modeling and t-tests/ANOVAs

- First, calculate SS_{total}

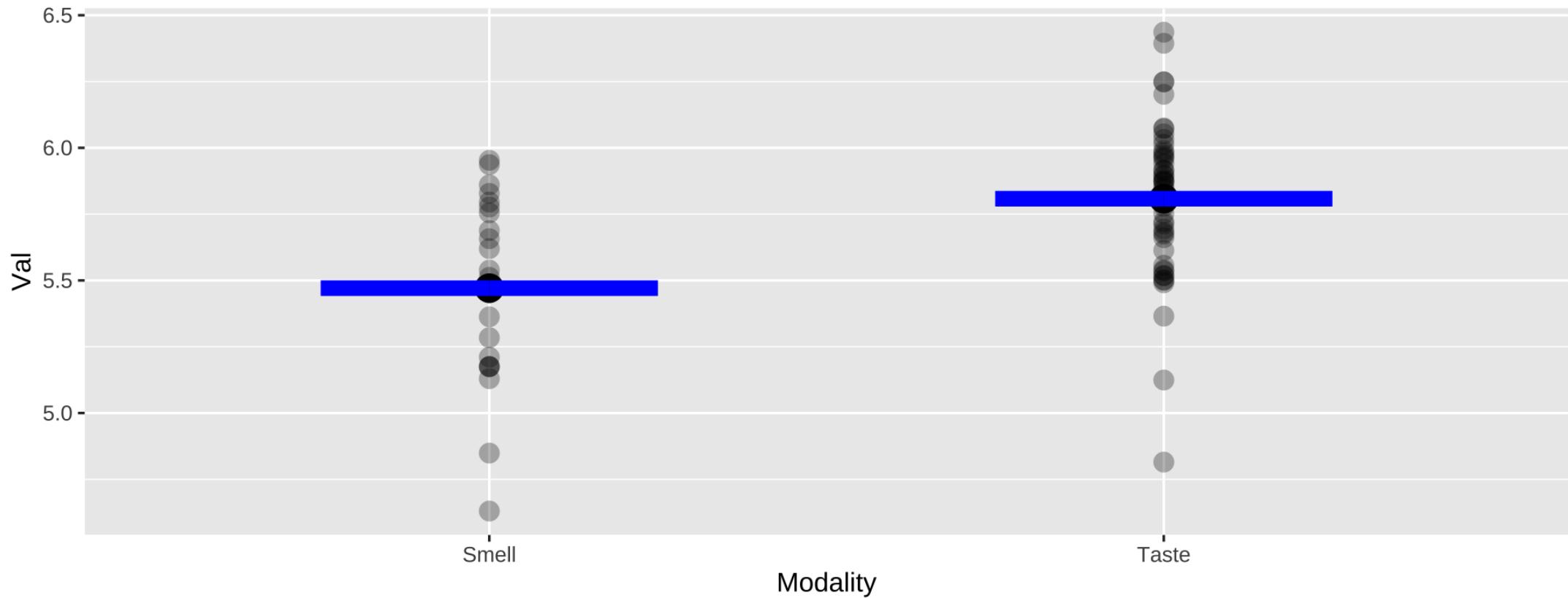


Fit a line

- Find line using least squares method
 - Mean is best line fit



Fit a line



Single Line

- We have two equations
 - $y = b_0$ = intercept(mean of smell)
 - $y = b_0$ = intercept(mean of touch)
- How do we get one linear equation?

Dummy Coding/Treatment Coding

Dummy Coding/Treatment Coding

- R's default system
 - 0's and 1's, with reference level at intercept
 - R does this automatically (0 = whatever comes first alphabetically)
 - Smell = 0
 - Taste = 1

$$\text{Val} = \alpha + \beta_1(X_i) + \epsilon$$

X_i = Indicator of group (0 or 1)

Dummy Coding/Treatment Coding

- Prediction for Smell $X_i = 0$

$$\text{Val} = \alpha + \beta_1(\text{Modality}_{\text{Taste}} 0) + \epsilon$$

$$\bar{Y} = \alpha(Smell_{mean})$$

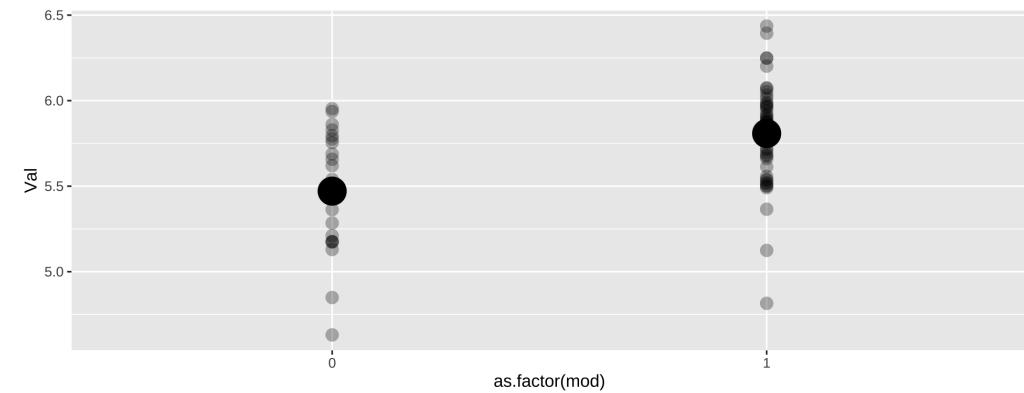
- Prediction for Taste $X_i = 1$

$$\text{Val} = \alpha + \beta_1(\text{Modality}_{\text{Taste}} 1) + \epsilon$$

$$\bar{Y}_{\text{taste}} = \alpha + \beta_1(\text{Modality}_{\text{Taste}} 1) + \epsilon$$

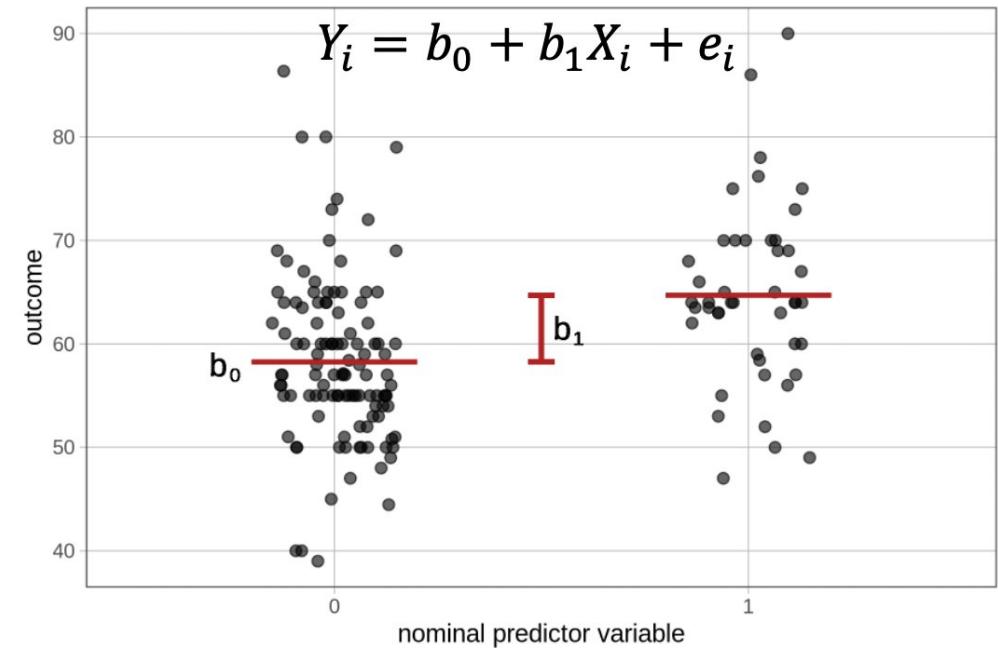
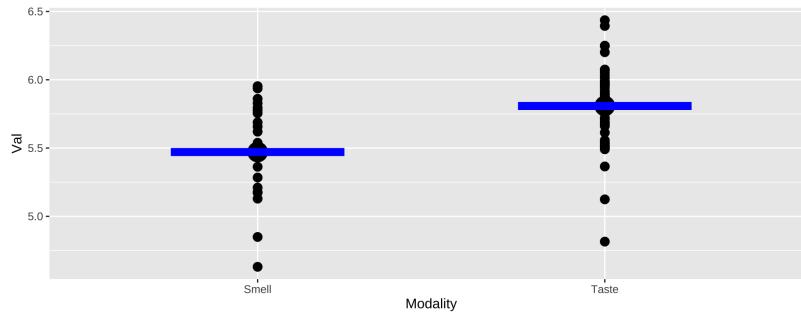
Dummy Coding By Hand

```
senses_dum <- senses_filt %>%  
  mutate(mod=ifelse(Modality=="Smell", 0, 1))
```



Categorical Contrast Coding

$$slope = \frac{\mu_{diff}}{run}$$



Dummy Coded Regression

```
lm(Val ~ Modality, data=senses_filt)
```

- Intercept: $\bar{Y} = b_0(\text{Smell}) = 5.47$
- Slope (\$b_1\$): Valance of taste words are .337 higher
 - One unit increase (going from 0 to 1; from Smell to Taste) associated with .337 increase in valance scores (mean difference)
- Adding intercept and slope together will give us mean valence of taste words

Change Reference Level

```
senses_dum <- senses_filt %>%
  mutate(mod=factor(Modality)) %>%
  mutate(mod1=relevel(mod, ref="Taste")) # relevel
contrasts(senses_dum$mod1)
```

```
##      Smell
## Taste    0
## Smell    1
```

```
contrasts(senses_dum$mod)
```

```
##      Taste
## Smell    0
## Taste    1
```

```
lm(Val ~ mod, data=senses_dum)
```

Effects Coding/Sum Coding

Effects Coding/Sum Coding

- So far the intercept at 0 has referred to a particular baseline or reference level
- Centering (subtracting mean from each value) changes the intercept to correspond to the overall mean
 - While mostly done for continuous variables, you can apply centering to categorical variables

Effects Coding/Sum Coding (.5, -.5)

- Mean of dummy coded variable is = .5
 - Subtract .5 from (0, 1) and we get $+0.5$ and -0.5

- Y intercept is now the grand mean

$$\frac{\mu_1 + \mu_2}{2}$$

- Slope is still the difference

```
senses_filt_sum <- senses_filt %>%
  mutate(mod_sum=as.factor(Modality), mod_sum_r = as.factor(Modality)) # add a new var to sum code

contrasts(senses_filt_sum$mod_sum) <- c(0.5, -0.5) # change 0 - 1 to +.5 and -.5
```

Effects Coding Results

```
# reg regression  
lm(Val ~ Modality, data=senses_filt)
```

```
# regression with sum coding (.5 - .5)  
lm(Val ~ mod_sum, data=senses_filt_sum)
```

- Intercept is grand mean (mean of means): $\hat{\beta}_0 = 5.64$
- Mean of Smell: $\hat{\beta}_0 + \hat{\beta}_1 \times .5 = 5.64 - 0.33(0.5) = 5.47$
- Mean of Taste: $\hat{\beta}_0 + \hat{\beta}_1 \times -.5 = 5.64 + (-0.33) * -0.5 = 5.80$

Default Sum Coding R behavior

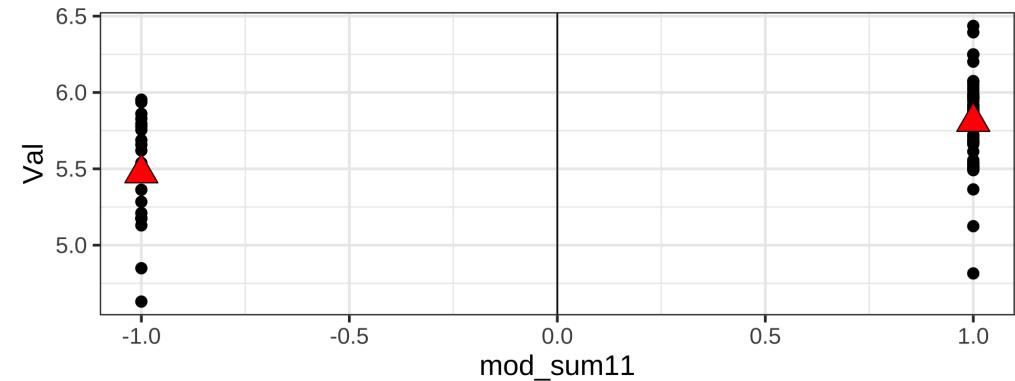
- +1 and -1

```
senses_filt_sum <- senses_filt_sum %>%  
  mutate(mod_sum55 = ifelse(mod_sum == "Taste", .5, -.5), mod_sum11 = ifelse(mod_sum == "Taste", 1, -1)) # add a new
```

Sum Coding (-1 + 1) Interpretation

What does this do to our interpretation?

- Intercept is now centered at 0 (grand mean)
- Slope rise is still the same (difference between categories) but:
 - Stepping from one category to another (the run) results in overall change of 2
 - Results are halved



Sum Coding (+1, -1) Model Results

```
lm(Val ~ mod_sum55, data=senses_filt_sum)
```

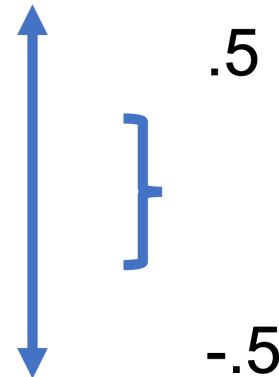
```
lm(Val ~ mod_sum11, data=senses_filt_sum)
```

--

- Intercept is grand mean: $\hat{\beta}_0 = 5.64$
- Mean of Smell:
- Mean of Taste:

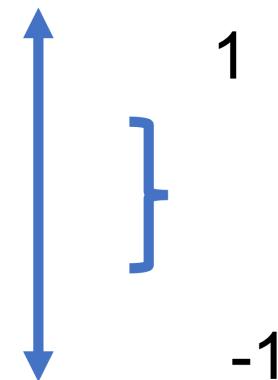
Why -0.5 and +0.5?

Contrast Code



1 unit change in contrast is the difference between smell and taste

Contrast Code



1 unit change in contrast is half the difference between smell and taste

The General Linear F-Test

- Can test overall influence for 2 or more levels of a factor
- We can think about the hypotheses for the overall test being:
 - H_0 : We cannot predict the dependent variable (over and above a restricted model (only an intercept))
 - H_1 : We can predict the dependent variable (over and above a model with only an intercept)

Restricted Model

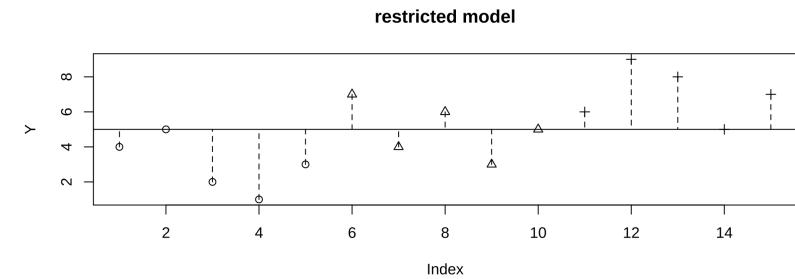
$$Y_{ij} = \mu + \epsilon$$

- Restricted model (Intercept-only): each score Y_{ij} is the result of a single population mean plus random error

$$SS_{error}(R) = \sum (y_i - \bar{y})^2 = SS_{total}$$

where:

y_i = observed value \bar{y} = mean value



Full Model

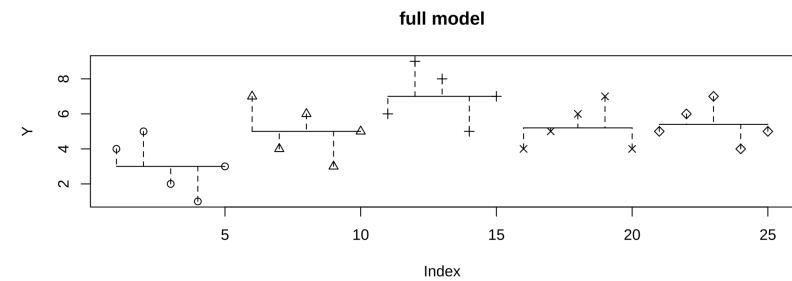
$$Y_{ij} = \mu_j + \epsilon$$

- Full model (all predictors/levels): each score Y_{ij} is the result of a different group mean plus random error

$$SSE(F) = \sum (y_{ij} - \hat{y}_{ij})^2 = SSE$$

where:

i = Person j = Group y_i = Observed value \hat{y} = Value estimated by regression line



F-ratio

- F-ratio is measure of signal to noise
- Tells us if overall model is significant fit to the data
 - H_0 : We cannot predict the dependent variable (over and above a model with only an intercept)
 - H_1 : We can predict the dependent variable (over and above a model with only an intercept)

$$df_R = N - 1$$

$$df_F = N - a$$

$$SSE(R) = SS_{total}$$

$$SSE(F) = SS_{error}$$

F-ratio

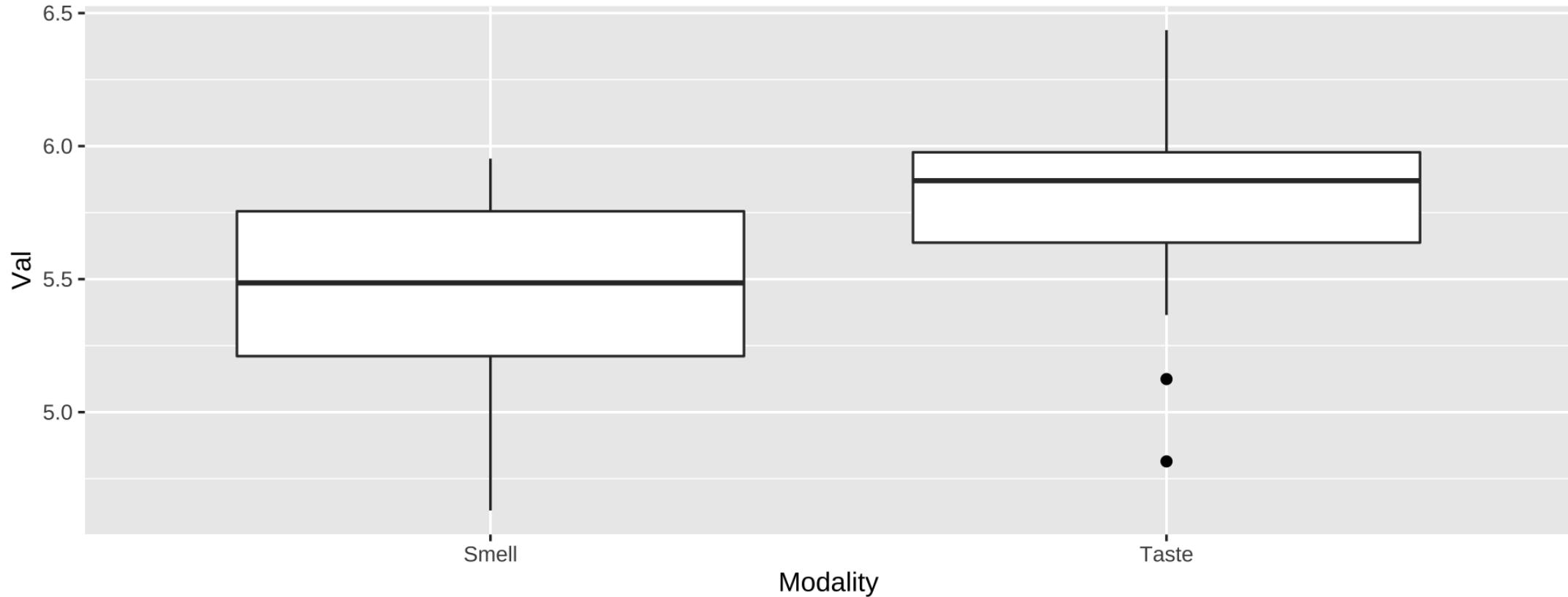
$$F = \frac{SS_R - SS_F/df_R - df_F(p-1)}{SS_F/df_F(N-p)} = \frac{MS_{model}}{MS_{error}}$$

- If Full = Restricted , then F=1
- If Full > Restricted, F > 1
- If Full < Restricted, F < 1

Degrees of freedom: F(a-1, n - a)

Plotting Categorical Effects

- Boxplot

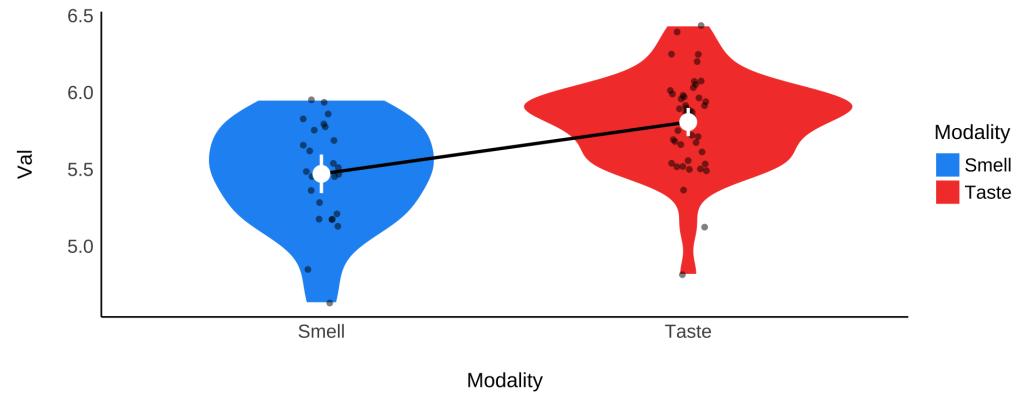


Violin Plots

```
mod=lm(Val ~ Modality, data=senses_filt)

means=modelbased::estimate_means(mod)

d=ggplot(senses_filt, aes(x = Modality, y = Val)) +
  # Add base data
  geom_violin(aes(fill = Modality), color = "white")
  geom_jitter2(width = 0.05, alpha = 0.5) +
  
  # Add pointrange and line from means
  geom_line(data = means, aes(y = Mean, group = 1),
  geom_pointrange(
    data = means,
    aes(y = Mean, ymin = CI_low, ymax = CI_high),
    size = 1,
    color = "white"
  ) +
  # Improve colors
  scale_fill_material() +
  theme_modern()
```



Activity

Mental Health and Drug Use

- CESD: Depressions scores
- unemp: 1=employed 0=unemployed

```
d <- read_csv("https://raw.githubusercontent.com/ASKurz/Applied-Longitudinal-Data-Analysis-with-brms-and-the-ti")
```

Activity

1. Change `unemp` variable to a factor with categorical labels
2. Dummy code the unemployment variable
3. Contrast code the unemployment variable
4. Run `lm` on the dummy coded variable
5. Interpret the output
6. Use output to extract the means for each group only using the output
7. Plot the results

Linear Models with Multiple Levels

- So far we have only been looking at two levels
 - We easily can extend linear modeling approach to multiple levels
 - Let's go back to our sense data
 - Before filtering it down to 2 senses it had 5 senses!

Treatment/Dummy Coding: Multilevel Factors

```
lm(Val~Modality, data=senses) %>%  
  tidy()
```

term	estimate	std.error	statistic	p.value
(Intercept)	5.58	0.0189	295	0
ModalitySmell	-0.109	0.0564	-1.93	0.0549
ModalitySound	-0.174	0.0376	-4.64	4.66e-06
ModalityTaste	0.228	0.0431	5.3	1.96e-07
ModalityTouch	-0.0452	0.0374	-1.21	0.227

- What is going on here? There are only 4 levels, but we actually have 5 levels.

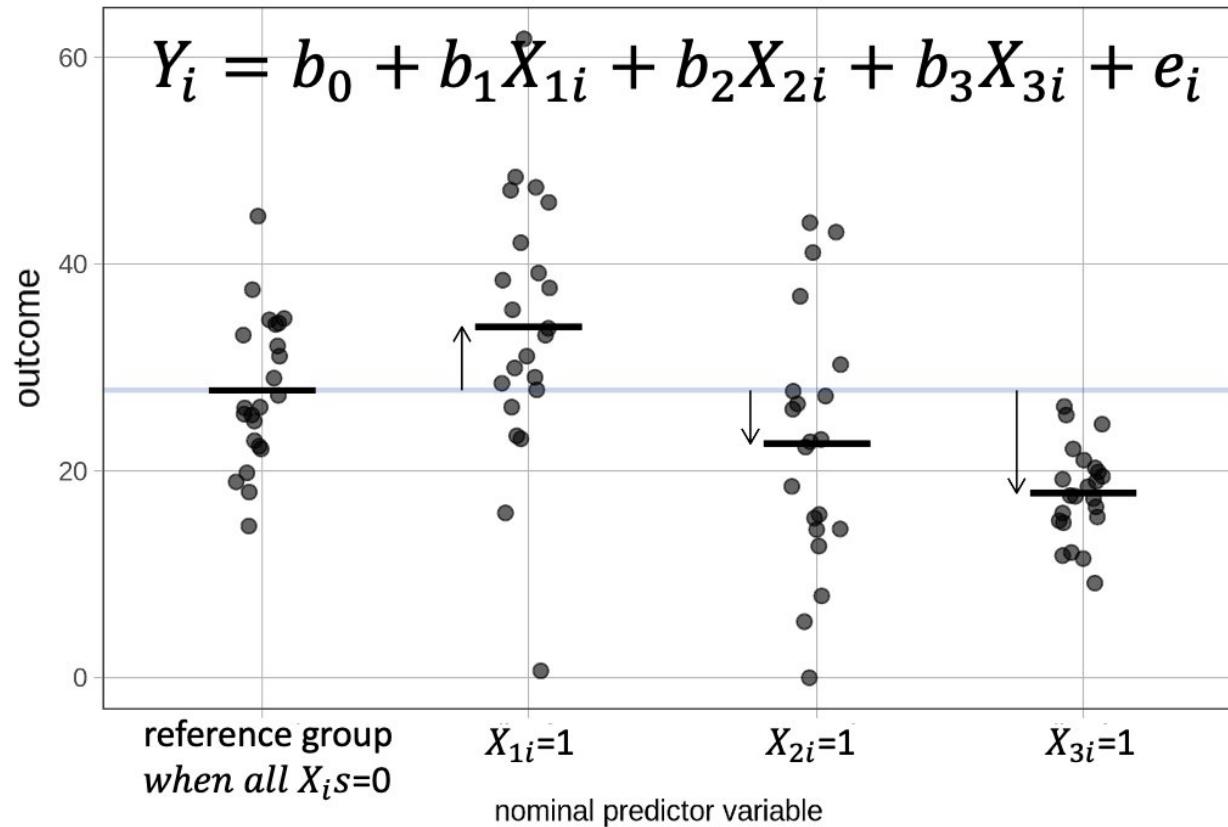
Dummy Coding Extension

1. Create one fewer dummy codes than levels (K
(number of levels)-1)
2. Choose one of your levels as baseline and
assign all zeros for this level across each
dummy code
3. For first dummy code, assign 1 to first group
and 0s for rest of levels
4. For the second dummy code, assign 1 to
second group and 0s for rest of levels
5. For third dummy code, assign 1 to third group
and 0s for rest of levels
6. For fourth dummy code, assign 1 to fourth
group and 0s for rest of levels

2	3	4	5
0	0	0	0
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Linear Equation

$$\text{Val} = \alpha + \beta_1(\text{Modality}_{\text{Smell}}) + \beta_2(\text{Modality}_{\text{Sound}}) + \beta_3(\text{Modality}_{\text{Taste}}) + \beta_4(\text{Modality}_{\text{Touch}}) + \epsilon$$



Hello Again Sums of Squares

The General Linear F-Test

- We can think about the hypotheses for the overall test being:

$$H_0 : b_1 = 0$$

$$H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5$$

$$H_1 : b_1 \neq b_2 \neq b_3 \neq b_4 \neq b_5$$

$$H_1 : \mu_1 \neq \mu_2 \neq \mu_3 \neq \mu_4 \neq \mu_5$$

- *Analysis of Variance (ANOVA)*

Restricted Model

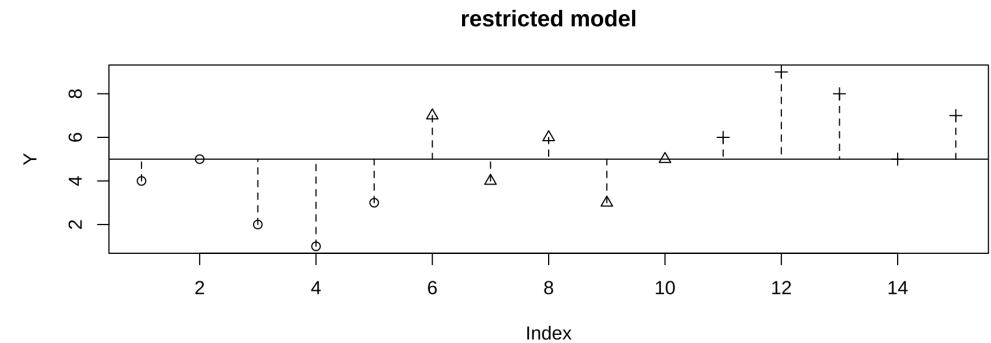
$$Y_{ij} = \mu + \epsilon$$

- Restricted model (Intercept-only): each score Y_{ij} is the result of a single population mean plus random error

$$SS_{error}(R) = \sum (y_i - \bar{y})^2 = SS_{total}$$

where:

y_i = observed value \bar{y} = mean value



Full Model

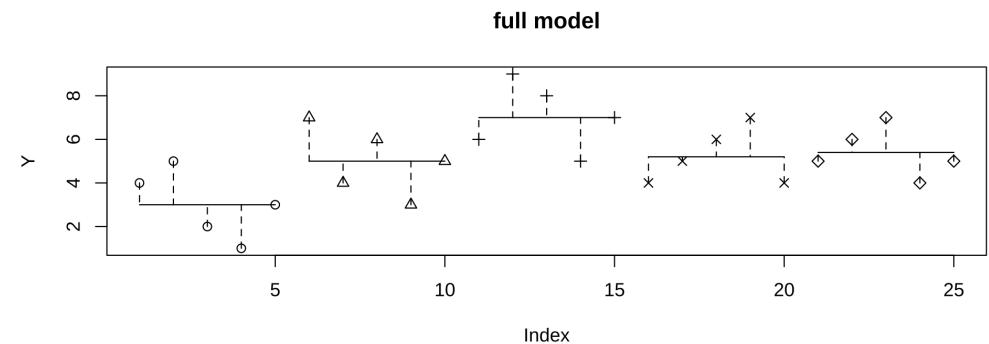
$$Y_{ij} = \mu_j + \epsilon$$

- Full model (all predictors/levels): each score Y_{ij} is the result of a different group mean plus random error

$$SSE(F) = \sum (y_{ij} - \hat{y}_{ij})^2 = SSE$$

where:

i = Person j = Group y_i = Observed value \hat{y} = Value estimated by regression line



F-ratio

- F-ratio is measure of signal to noise

$$df_R = N - 1$$

$$df_F = N - p$$

$$SSE(R) = SS_{total}$$

$$SSE(F) = SS_{error}$$

F-ratio

$$F = \frac{SS_R - SS_F/df_R - df_F(p-1)}{SS_F/df_F(N-p)} = \frac{MS_{model}}{MS_{error}}$$

- If Full = Restricted , then F=1
- If Full > Restricted, F > 1
- If Full < Restricted, F < 1

Degrees of freedom: F(p-1, n - p)

ANOVA Table

```
lm(Val~Modality, data=senses) %>%  
  parameters::model_parameters()
```

Parameter	Coefficient	SE	CI	CI_low	CI_high	t	df_error	p
(Intercept)	5.58	0.0189	0.95	5.54	5.62	295	400	0
ModalitySmell	-0.109	0.0564	0.95	-0.22	0.00229	-1.93	400	0.0549
ModalitySound	-0.174	0.0376	0.95	-0.248	-0.101	-4.64	400	4.66e-06
ModalityTaste	0.228	0.0431	0.95	0.144	0.313	5.3	400	1.96e-07
ModalityTouch	-0.0452	0.0374	0.95	-0.119	0.0282	-1.21	400	0.227

ANOVA Table

```
aov1<-aov(Val~ 1, data=senses)
aov2 <- aov(Val~Modality, data=senses)
#anova(aov1, aov2) compare two models
aov(Val~Modality, data=senses) %>%
  parameters::model_parameters()
```

Parameter	Sum_Squares	df	Mean_Square	F	p
Modality	4.81	4	1.2	17	6.62e-13
Residuals	28.3	400	0.0707		

Model Comparison Approach vs Traditional Approach to ANOVA

- Traditional formulation of ANOVA asks the same question in a different way:
 - Is the variability between groups (variance due to differences between groups) greater than expected on the basis of the within-group variability (the variability within a group) observed, and random sampling of group members?
- Both use sum of squares
- Both use F-statistic
- $F = \frac{MSR}{MSE}$ (Same Mean Squared Error on ANOVA table outputs)

Post-Hoc Comparisons

```
aov_em=aov(Val~Modality, data=senses) # fit ANOVA  
  
aov_em %>%  
parameters::model_parameters(.) # print out ANOVA table
```

Parameter	Sum_Squares	df	Mean_Square	F	p
Modality	4.81	4	1.2	17	6.62e-13
Residuals	28.3	400	0.0707		

- The Modality factor is significant. Now what?

Post-Hoc Comparisons

- The best package ever created: [emmeans](#)
- Allows one to extract marginal means for the model and also test comparisons of interest

Pairwise Tests

- Get means and pairwise comparisons

```
# get pairwise tests between all groups
as.data.frame(emmeans::emmeans(aov_em, specs = "Modality"))
flextable()
```

Modality	emmean	SE	df	lower.CL	upper.CL
Sight	5.579663	0.01889440	400	5.542518	5.616808
Smell	5.471012	0.05317357	400	5.366477	5.575546
Sound	5.405193	0.03248092	400	5.341338	5.469047
Taste	5.808124	0.03878081	400	5.731884	5.884364
Touch	5.534435	0.03224121	400	5.471052	5.597818

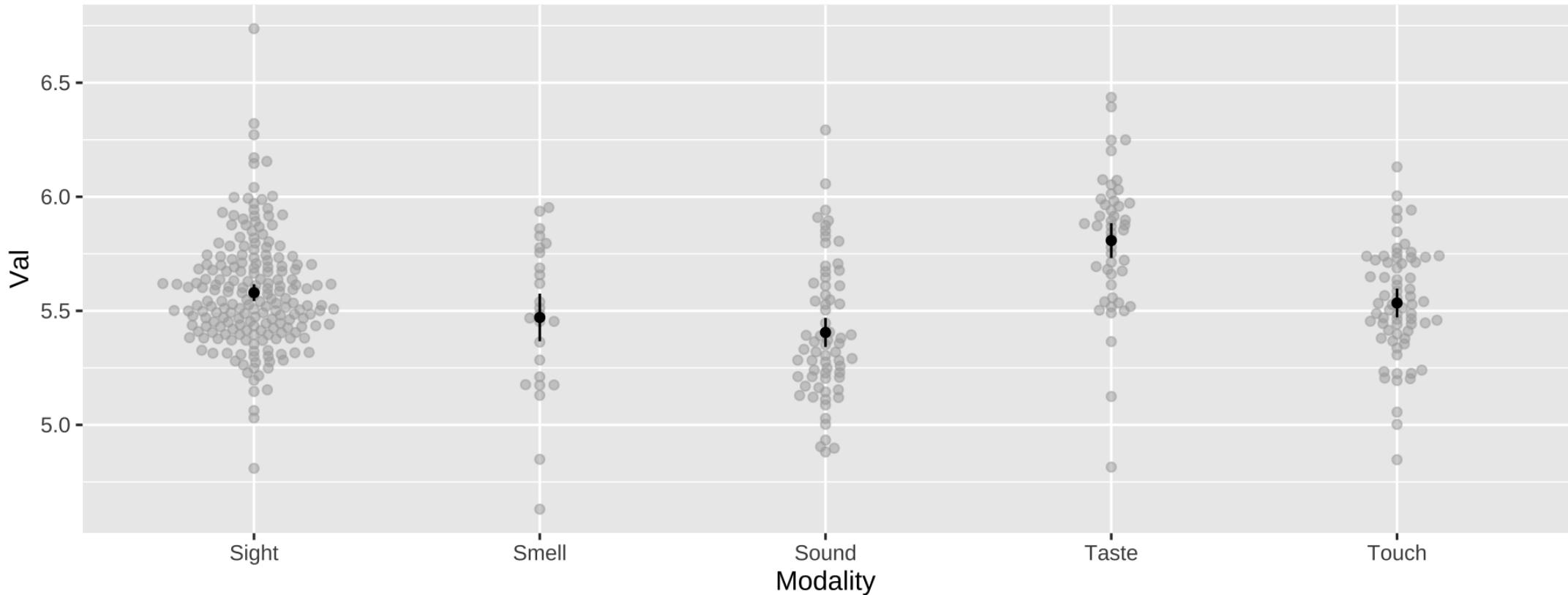
contrast	estimate	SE	df	t.ratio	p.v
Sight - Smell	0.10865148	0.05643072	400	1.925396	0.305501164
Sight - Sound	0.17447036	0.03757671	400	4.643046	0.0000456942
Sight - Taste	-0.22846083	0.04313872	400	-5.295957	0.0000019441
Sight - Touch	0.04522812	0.03736969	400	1.210289	0.745433731
Smell - Sound	0.06581888	0.06230922	400	1.056327	0.8286561493
Smell - Taste	-0.33711231	0.06581321	400	-5.122259	0.000004659
Smell - Touch	-0.06342336	0.06218459	400	-1.019921	0.8461335445
Sound - Taste	-0.40293120	0.05058618	400	-7.965243	0.000000000000000

afex ANOVA package

```
library(afex)

one_fit <- aov_ez("Word", "Val", data = senses, between = c("Modality"))

afex_plot(one_fit, x="Modality")
```



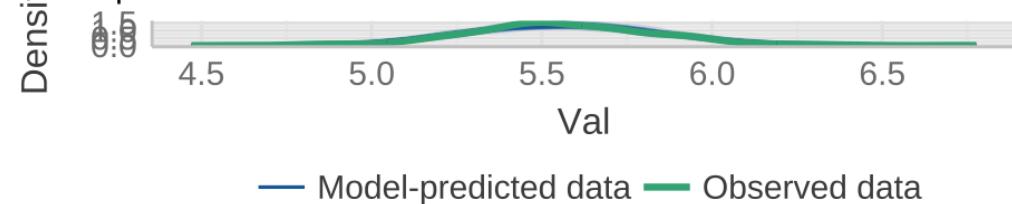
Assumptions

- Within linear modeling framework, do normal assumptions checks

```
check_model()
```

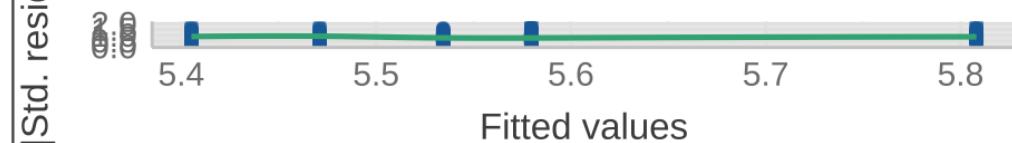
Posterior Predictive Check

Model-predicted lines should resemble observed data line



Homogeneity of Variance

Reference line should be flat and horizontal



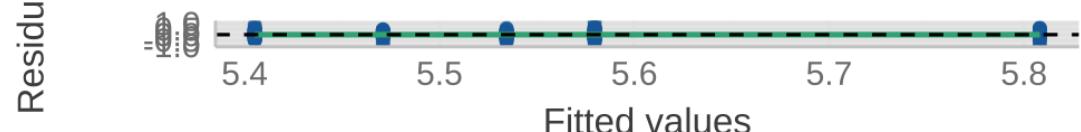
Normality of Residuals

Dots should fall along the line



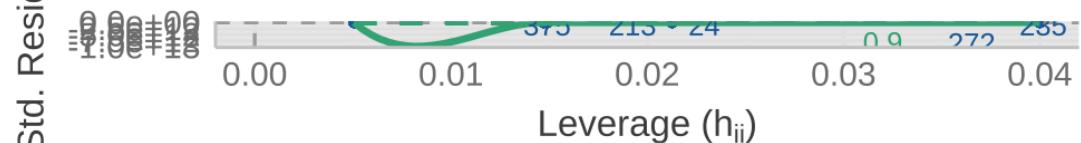
Linearity

Reference line should be flat and horizontal



Influential Observations

Points should be inside the contour lines



Effect Sizes: eta

- η^2

Interpretation: % of variance explained

$$\eta^2 = \frac{SS_{model}}{SS_{total}}$$

- η^2 cannot easily be compared between studies, because the total variability in a study (SS_{total}) depends on the design of a study, and increases when additional variables are manipulated
- .01: Small
- .06: Medium
- .14: Large

Effect Sizes:eta

- η_p^2

Interpretation: % of variance explained for one effect (partiaing out others)

$$\eta_p^2 = \frac{SS_{model}}{SS_{model} + SS_{error}}$$

Effect Sizes: eta

- η_g^2 - Generalized eta-squared

$$\frac{SS_{model}}{SS_{model} + SS_{subject} + SS_{error}}$$

Less Biased Effect Size: Omega

- ω^2

$$\omega^2 = \frac{SS_{model} - df_{model} \cdot MS_{error}}{SS_{total} + MS_{error}}$$

- ω_p^2

$$\frac{df_{model} \times (MS_{model} - MS_{error})}{SS_{model} + (N - df_{model}) \times MS_{error}}$$

- .01: Small
- .06: Medium
- .14: Large

Effect Size: Cohen's f

$$\text{Cohen's } f_p = \sqrt{\frac{\eta_p^2}{1 - \eta_p^2}} = \sqrt{\frac{SS_{effect}}{SS_{error}}}$$

- .14: Small
- .39: Medium
- .59: Large

<https://imaging.mrc-cbu.cam.ac.uk/statswiki/FAQ/effectSize>

Calculate ANOVA Effect Size in R

```
cohens_f(aov_em)
```

Parameter	Cohens_f	CI	CI_low	CI_high
Modality	0.413	0.95	0.316	Inf

```
eta_squared(aov_em)
```

Parameter	Eta2	CI	CI_low	CI_high
Modality	0.146	0.95	0.0909	1

```
omega_squared(aov_em)
```

Parameter	Omega2	CI	CI_low	CI_high
Modality	0.137	0.95	0.0832	1

Power

- We can also run power analyses for omnibus tests (e.g., number of participants needed to find a sufficiently powered main effect or interaction)
 - May not sufficiently power one for the smallest desired effect size of interest
 - Recommendation: Perform on smallest desired effect size (e.g., mean comparison while controlling for multiple corrections)
- Often complex tests cannot be performed analytically and you must use numerical methods
- Same approach we have already done!

Power Analysis: ANOVA

- You are planning a reaction-time study involving three groups ($k = 3$)
- Pilot research & data from literature suggest effect size is medium $f = .39$
- Suppose you want a power of 0.9
- How many subjects do you need in each sample group?

Power Analysis: ANOVA

```
library(pwr)

pwr.anova.test(k=3,n=NULL,f=.39,sig.level=0.05,power=0.9)

##          Balanced one-way analysis of variance power calculation
##
##          k = 3
##          n = 28.75626
##          f = 0.39
##          sig.level = 0.05
##          power = 0.9
##
## NOTE: n is number in each group
```

```
#k = groups
#n= sample size
#es = cohen's f
```

Superpower

- Same RT study
- Pilot research & data from literature suggest population means might be 400, 450 and 500 ms with a sample within-group standard deviation of 100 ms
- Suppose you want a power of 0.9
- How many subjects do you need in each sample group?

Non-Parametric

- Kruskal Wallis Test
 - Can be used if assumptions are not met
 - Extension of Mann-Whitney test
- Welch's F test (W-test)

```
```{r}
kruskal.test(Val ~ Modality, data=senses)
```
...
##  
##  Kruskal-Wallis rank sum test  
##  
## data: Val by Modality  
## Kruskal-Wallis chi-squared = 57.815, df = 4, p-value = 8.544e-12
```
library(onewaytests)
welch.test(Val ~ Modality, data = senses)

##
Welch's Heteroscedastic F Test (alpha = 0.05)

data : Val and Modality

statistic : 13.01914
num df : 4
denom df : 101.1231
p-value : 1.374323e-08

Result : Difference is statistically significant.

```