Interactions

Categorical predictors

One of the benefits of using regression (instead of partial correlations) is that it can handle both continuous and categorical predictors and allows for using both in the same model.

Categorical predictors with more than two levels are broken up into several smaller variables. In doing so, we take variables that don't have any inherent numerical value to them (i.e., nominal and ordinal variables) and ascribe meaningful numbers that allow for us to calculate meaningful statistics.

You can choose just about any numbers to represent your categorical variable. However, there are several commonly used methods that result in very useful statistics.

Dummy coding

In dummy coding, one group is selected to be a reference group. From your single nominal variable with K levels, $K - 1$ dummy code variables are created; for each new dummy code variable, one of the non-reference groups is assigned 1; all other groups are assigned 0.

Occupation	D1	D2
Engineer	0	0
Teacher	1	0
Doctor	0	1

The dummy codes are entered as IV's in the regression equation.

Person	Occupation	D1	D2
Billy	Engineer	0	0
Susan	Teacher	1	0
Michael	Teacher	1	0
Molly	Engineer	0	0
Katie	Doctor	0	1

Example

Solomon's paradox describes the tendency for people to reason more wisely about other people's problems compared to their own. One potential explanation for this paradox is that people tend to view other people's problems from a more psychologically distant perspective, whereas they view their own problems from a psychologically immersed perspective. To test this possibility, researchers asked romantically-involved participants to think about a situation in which their partner cheated on them (self condition) or a friend's partner cheated on their friend (other condition). Participants were also instructed to take a first-person perspective (immersed condition) by using pronouns such as I and me, or a third-person perspective (distanced condition) by using pronouns such as he and her.

```
solomon <- read.csv(here::here("R", "solomon.csv"))
```

Grossmann, I., & Kross, E. (2014). Exploring Solomon's paradox: Self-distancing eliminates self-other asymmetry in wise reasoning about close relationships in younger and older adults. *Psychological Science, 25*, 1571-1580.

```r
psych::describe(solomon[,c("ID", "CONDITION", "WISDOM")], fast = T)
```

```
##           vars   n  mean    sd   min    max  range   se
## ID           1 120 64.46 40.98  1.00 168.00 167.00 3.74
## CONDITION    2 120  2.46  1.12  1.00   4.00   3.00 0.10
## WISDOM       3 115  0.01  0.99 -2.52   1.79   4.31 0.09
```

```r
library(knitr)
library(kableExtra)
library(tidyverse)
head(solomon) %>%
  select(ID, CONDITION,
         WISDOM) %>%
  kable() %>% kable_styling()
```

ID	CONDITION	WISDOM
1	3	-0.2758939
6	4	0.4294921
8	4	-0.0278587
9	4	0.5327150
10	2	0.6229979
12	2	-1.9957813

```r
solomon = solomon %>%
  mutate(dummy_2 = ifelse(CONDITION == 2, 1, 0),
         dummy_3 = ifelse(CONDITION == 3, 1, 0),
         dummy_4 = ifelse(CONDITION == 4, 1, 0))
solomon %>%
  select(ID, CONDITION, WISDOM,
         matches("dummy")) %>%
  kable() %>% kable_styling()
```

ID	CONDITION	WISDOM	dummy_2	dummy_3	dummy_4
1	3	-0.2758939	0	1	0
6	4	0.4294921	0	0	1
8	4	-0.0278587	0	0	1
9	4	0.5327150	0	0	1
10	2	0.6229979	1	0	0
12	2	-1.9957813	1	0	0
14	3	-1.1514699	0	1	0
18	2	-0.6912011	1	0	0
21	2	0.0053117	1	0	0

```r
mod.1 = lm(WISDOM ~ dummy_2 + dummy_3 + dummy_4, data = solomon)
summary(mod.1)
```

```
##
## Call:
## lm(formula = WISDOM ~ dummy_2 + dummy_3 + dummy_4, data = solomon)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6809 -0.4209  0.0473  0.6694  2.3499
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.5593     0.1686  -3.317 0.001232 **
## dummy_2       0.6814     0.2497   2.729 0.007390 **
## dummy_3       0.7541     0.2348   3.211 0.001729 **
## dummy_4       0.8938     0.2524   3.541 0.000583 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9389 on 111 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.1262,    Adjusted R-squared:  0.1026
## F-statistic: 5.343 on 3 and 111 DF,  p-value: 0.001783
```

Interpreting coefficients

When working with dummy codes, the intercept can be interpreted as the mean of the reference group.

$$\begin{aligned}
\hat{Y} &= b_0 + b_1 D_2 + b_2 D_3 + b_3 D_2 \\
\hat{Y} &= b_0 + b_1(0) + b_2(0) + b_3(0) \\
\hat{Y} &= b_0 \\
\hat{Y} &= \bar{Y}_{\text{Reference}}
\end{aligned}$$

What do each of the slope coefficients mean?

From this equation, we can get the mean of every single group.

```
newdata = data.frame(dummy_2 = c(0,1,0,0),
                     dummy_3 = c(0,0,1,0),
                     dummy_4 = c(0,0,0,1))
predict(mod.1, newdata = newdata, se.fit = T)
```

```
## $fit
##          1          2          3          4
## -0.5593042  0.1220847  0.1948435  0.3344884
##
## $se.fit
##         1         2         3         4
## 0.1686358 0.1841382 0.1634457 0.1877848
##
## $df
## [1] 111
##
## $residual.scale
## [1] 0.9389242
```

From this equation, we can get the mean of every single group.

```
solomon %>%
  mutate_at("CONDITION", ~as.factor(.)) %>%
  group_by(CONDITION) %>%
  drop_na() %>%
  summarize(meanWisdom = mean(WISDOM))
```

```
## # A tibble: 4 × 2
##   CONDITION meanWisdom
##   <fct>          <dbl>
## 1 1             -0.559
## 2 2              0.122
## 3 3              0.195
## 4 4              0.334
```

And the test of the coefficient represents the significance test of each group to the reference. This is an independent-samples *t*-test.

The test of the intercept is the one-sample *t*-test comparing the intercept to 0.

```
summary(mod.1)$coef
```

```
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -0.5593042  0.1686358 -3.316641 0.0012319438
## dummy_2      0.6813889  0.2496896  2.728944 0.0073896074
## dummy_3      0.7541477  0.2348458  3.211247 0.0017291997
## dummy_4      0.8937927  0.2523909  3.541303 0.0005832526
```

What if you wanted to compare groups 2 and 3?

```r
solomon = solomon %>%
  mutate(dummy_1 = ifelse(CONDITION == 1, 1, 0),
         dummy_3 = ifelse(CONDITION == 3, 1, 0),
         dummy_4 = ifelse(CONDITION == 4, 1, 0))
mod.2 = lm(WISDOM ~ dummy_1 + dummy_3 + dummy_4, data = solomon)
summary(mod.2)
```

```
##
## Call:
## lm(formula = WISDOM ~ dummy_1 + dummy_3 + dummy_4, data = solomon)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6809 -0.4209  0.0473  0.6694  2.3499
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.12208    0.18414   0.663  0.50870
## dummy_1     -0.68139    0.24969  -2.729  0.00739 **
## dummy_3      0.07276    0.24621   0.296  0.76816
## dummy_4      0.21240    0.26300   0.808  0.42104
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9389 on 111 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.1262,    Adjusted R-squared:  0.1026
```

Time savers

Show in R!

```r
mod.3 = lm(WISDOM ~ CONDITION, data = solomon)
summary(mod.3)
```

```
## 
## Call:
## lm(formula = WISDOM ~ CONDITION, data = solomon)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.64827 -0.55096  0.09494  0.72958  2.20076 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -0.69638    0.21399  -3.254 0.001500 ** 
## CONDITION    0.28621    0.07956   3.598 0.000478 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.943 on 113 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.1028,	Adjusted R-squared:  0.09482 
## F-statistic: 12.94 on 1 and 113 DF,  p-value: 0.000478
```

Omnibus test

```
summary(mod.1)
```

```
##
## Call:
## lm(formula = WISDOM ~ dummy_2 + dummy_3 + dummy_4, data = solomon)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6809 -0.4209  0.0473  0.6694  2.3499
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.5593     0.1686  -3.317 0.001232 **
## dummy_2       0.6814     0.2497   2.729 0.007390 **
## dummy_3       0.7541     0.2348   3.211 0.001729 **
## dummy_4       0.8938     0.2524   3.541 0.000583 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9389 on 111 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.1262,    Adjusted R-squared:  0.1026
## F-statistic: 5.343 on 3 and 111 DF,  p-value: 0.001783
```

Omnibus test

```
summary(mod.2)
```

```
##
## Call:
## lm(formula = WISDOM ~ dummy_1 + dummy_3 + dummy_4, data = solomon)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6809 -0.4209  0.0473  0.6694  2.3499
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.12208    0.18414   0.663  0.50870
## dummy_1     -0.68139    0.24969  -2.729  0.00739 **
## dummy_3      0.07276    0.24621   0.296  0.76816
## dummy_4      0.21240    0.26300   0.808  0.42104
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9389 on 111 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.1262,    Adjusted R-squared:  0.1026
## F-statistic: 5.343 on 3 and 111 DF,  p-value: 0.001783
```

Omnibus test

```
summary(mod.3)
```

```
##
## Call:
## lm(formula = WISDOM ~ CONDITION, data = solomon)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.64827 -0.55096  0.09494  0.72958  2.20076
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.69638    0.21399  -3.254 0.001500 **
## CONDITION    0.28621    0.07956   3.598 0.000478 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.943 on 113 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.1028,    Adjusted R-squared:  0.09482
## F-statistic: 12.94 on 1 and 113 DF,  p-value: 0.000478
```

NA

Omnibus test

```
anova(mod.3)
```

```
## Analysis of Variance Table
##
## Response: WISDOM
##            Df  Sum Sq Mean Sq F value   Pr(>F)
## CONDITION   1  11.508 11.5082  12.942 0.000478 ***
## Residuals 113 100.478  0.8892
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What are interactions?

When we have two variables, A and B, in a regression model, we are testing whether these variables have **additive effects** on our outcome, Y. That is, the effect of A on Y is constant over all values of B.

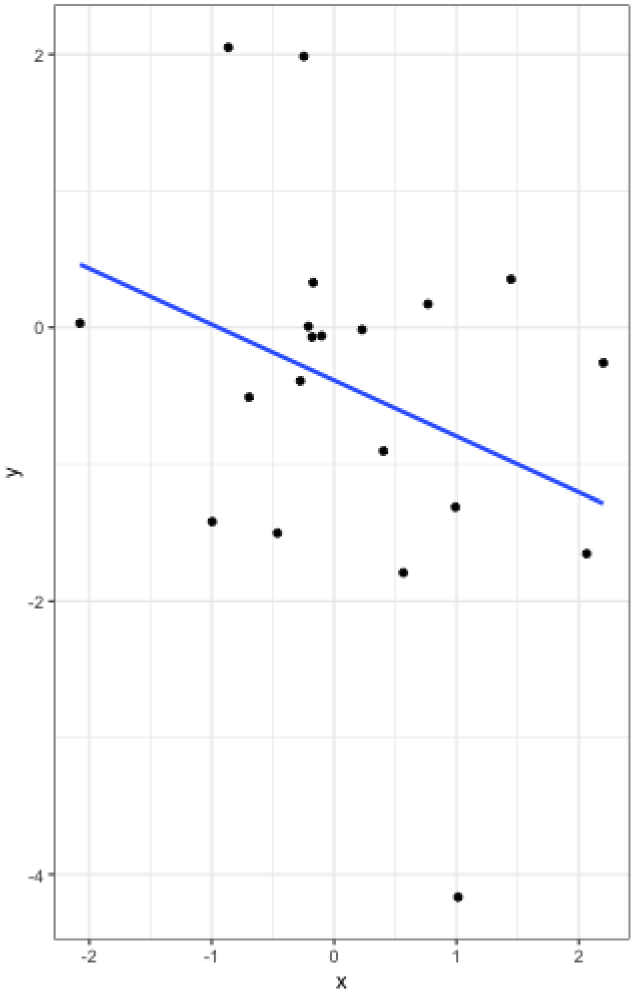
- Example: Drinking coffee and hours of sleep have additive effects on alertness; no matter how any hours I slept the previous night, drinking one cup of coffee will make me .5 SD more awake than not drinking coffee.

What are interactions?

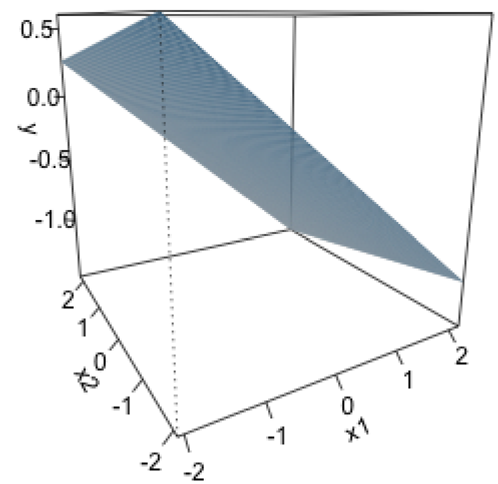
However, we may hypothesize that two variables have **joint effects**, or interact with each other. In this case, the effect of A on Y changes as a function of B.

- Example: Chronic stress has a negative impact on health but only for individuals who receive little or no social support; for individuals with high social support, chronic stress has no impact on health.
- This is also referred to as **moderation.**

Univariate regression



Multivariate regression



Multivariate regression with an interaction

Example

Let's use data about stress. We have an outcome (Stress) that we are interested in predicting from trait Anxiety and levels of Social Support. We can ignore the `group` status for the time being.

```
library(here)
stress.data = read.csv(here("R/stress.csv"))
library(psych)
describe(stress.data)
```

```
##          vars   n   mean     sd median trimmed    mad  min    max  range  sk
## id          1 118 488.65 295.95 462.50  485.76 372.13 2.00 986.00 984.00  0.
## Anxiety     2 118   7.61   2.49   7.75    7.67   2.26 0.70  14.64  13.94 -0.
## Stress      3 118   5.18   1.88   5.27    5.17   1.65 0.62  10.32   9.71  0.
## Support     4 118   8.73   3.28   8.52    8.66   3.16 0.02  17.34  17.32  0.
## group*      5 118   1.53   0.50   2.00    1.53   0.00 1.00   2.00   1.00 -0.
##          kurtosis    se
## id          -1.29 27.24
## Anxiety      0.28  0.23
## Stress       0.22  0.17
## Support      0.19  0.30
## group*      -2.01  0.05
```

In R

```
i.model1 = lm(Stress ~ Anxiety + Support + Anxiety:Support,
              data = stress.data)
i.model2 = lm(Stress ~ Anxiety*Support, data = stress.data)
```

Both methods of specifying the interaction above will work in R. Using the `*` tells R to create both the main effects and the interaction effect. Note, however that the following code *gives you the wrong results*:

```
imodel_bad = lm(Stress ~ Anxiety:Support,
                data = stress.data)
# This does not create main effects.
# It is VERY WRONG
# Don't do this
```

```r
i.model1 = lm(Stress ~ Anxiety*Support, data = stress.data)
summary(i.model1)
```

```
## 
## Call:
## lm(formula = Stress ~ Anxiety * Support, data = stress.data)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -3.8163 -1.0783  0.0373  0.9200  3.6109 
## 
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)    
## (Intercept)     -2.73966    1.12101  -2.444  0.01606 *  
## Anxiety          0.61561    0.13010   4.732 6.44e-06 ***
## Support          0.66697    0.09547   6.986 2.02e-10 ***
## Anxiety:Support -0.04174    0.01309  -3.188  0.00185 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.462 on 114 degrees of freedom
## Multiple R-squared:  0.4084,	Adjusted R-squared:  0.3928 
## F-statistic: 26.23 on 3 and 114 DF,  p-value: 5.645e-13
```

Conceptual interpretation

$$\hat{Y} = b_0 + b_1X + b_2Z + b_3XZ$$

You can interpret the interaction term in the same way you normally interpret a slope coefficient -- this is the effect of the interaction controlling for other variables in the model.

You can also interpret the intercept the same way as before (the expected value of Y when all predictors are 0).

But here, b_1 is the effect of X on Y when Z is equal to 0.

Conceptual interpretation

$$\hat{Y} = b_0 + b_1X + b_2Z + b_3XZ$$

Lower-order terms change depending on the values of the higher-order terms. The value of b_1 and b_2 will change depending on the value of b_3.

- These values represent "conditional effects" (because the value is conditional on the level of the other variable). In many cases, the value and significance test with these terms is either meaningless (if Z is never equal to 0) or unhelpful, as these values and significance change across the data.

Higher-order terms are those terms that represent interactions. b_3 is a higher-order term.

- This value represents how much the slope of X changes for every 1-unit increase in Z AND how much the slope of Z changes for everyone 1-unit increase in X.

Conceptual interpretation

Higher-order interaction terms represent:

- the change in the slope of X as a function of Z
- the degree of curvature in the regression plane
- the linear effect of the product of independent variables

```
stress.data$AxS = stress.data$Anxiety*stress.data$Support
head(stress.data[,c("Anxiety", "Support", "AxS")])
```

```
##    Anxiety Support       AxS
## 1 10.18520  6.1602  62.74287
## 2  5.58873  8.9069  49.77826
## 3  6.58500 10.5433  69.42763
## 4  8.95430 11.4605 102.62076
## 5  7.59910  5.5516  42.18716
## 6  8.15600  7.5117  61.26543
```

```r
summary(lm(Stress ~ Anxiety + Support + AxS, data = stress.data))
```

```
##
## Call:
## lm(formula = Stress ~ Anxiety + Support + AxS, data = stress.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8163 -1.0783  0.0373  0.9200  3.6109
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.73966    1.12101  -2.444  0.01606 *
## Anxiety      0.61561    0.13010   4.732 6.44e-06 ***
## Support      0.66697    0.09547   6.986 2.02e-10 ***
## AxS         -0.04174    0.01309  -3.188  0.00185 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.462 on 114 degrees of freedom
## Multiple R-squared:  0.4084,    Adjusted R-squared:  0.3928
## F-statistic: 26.23 on 3 and 114 DF,  p-value: 5.645e-13
```

```r
summary(lm(Stress ~ Anxiety*Support, data = stress.data))
```

```
##
## Call:
## lm(formula = Stress ~ Anxiety * Support, data = stress.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8163 -1.0783  0.0373  0.9200  3.6109
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -2.73966    1.12101  -2.444  0.01606 *
## Anxiety          0.61561    0.13010   4.732 6.44e-06 ***
## Support          0.66697    0.09547   6.986 2.02e-10 ***
## Anxiety:Support -0.04174    0.01309  -3.188  0.00185 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.462 on 114 degrees of freedom
## Multiple R-squared:  0.4084,    Adjusted R-squared:  0.3928
## F-statistic: 26.23 on 3 and 114 DF,  p-value: 5.645e-13
```

They're the same!!

Conditional effects and simple slopes

The regression line estimated in this model is quite difficult to interpret on its own. A good strategy is to decompose the regression equation into **simple slopes**, which are determined by calculating the conditional effects at a specific level of the moderating variable.

- Simple slope: the equation for Y on X at differnt levels of Z; but also refers to only the coefficient for X in this equation
- Conditional effect: the slope coefficients in the full regression model which can change. These are the lower-order terms associated with a variable. E.g., X has a conditional effect on Y.

Which variable is the "predictor" (X) and which is the "moderator" (Z)?

Getting Simple Slopes

The conditional nature of these effects is easiest to see by "plugging in" different values for one of your variables. Return to the regression equation estimated in our stress data:

$$\hat{Stress} = -2.74 + 0.62(\text{Anx}) + 0.67(\text{Sup}) + -0.04(\text{Anx} \times \text{Sup})$$

Set Support to 5

$$\begin{aligned} \hat{Stress} &= -2.74 + 0.62(\text{Anx}) + 0.67(5) + -0.04(\text{Anx} \times 5) \\ &= -2.74 + 0.62(\text{Anx}) + 3.35 + -0.2(\text{Anx}) \\ &= 0.61 + 0.42(\text{Anx}) \end{aligned}$$

Set Support to 10

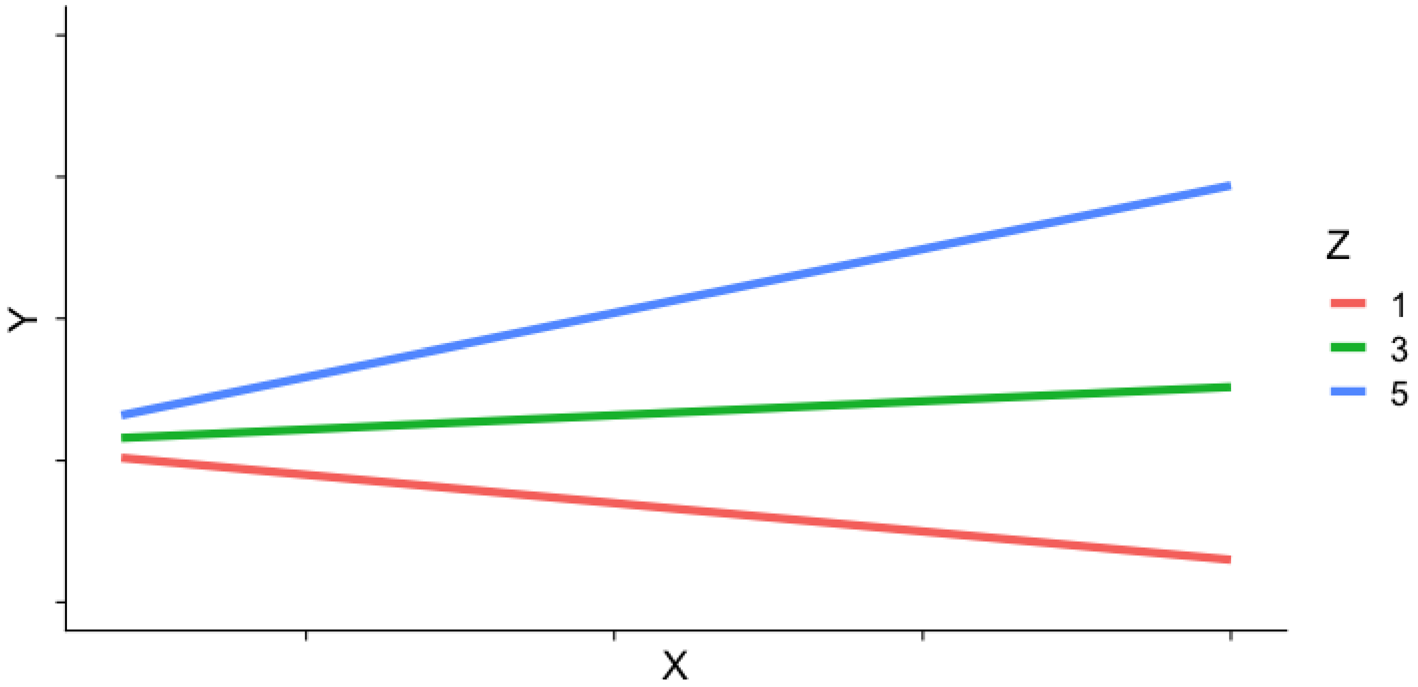
$$\begin{aligned} \hat{Stress} &= -2.74 + 0.62(\text{Anx}) + 0.67(10) + -0.04(\text{Anx} \times 10) \\ &= -2.74 + 0.62(\text{Anx}) + 6.7 + -0.4(\text{Anx}) \\ &= 3.96 + 0.22(\text{Anx}) \end{aligned}$$

Interaction shapes

Often we graph the simple slopes as a way to understand the interaction. Interpreting the shape of an interaction can be done using the numbers alone, but it requires a lot of calculation and mental rotation. For those reasons, consider it a requirement that you graph interactions in order to interpret them.

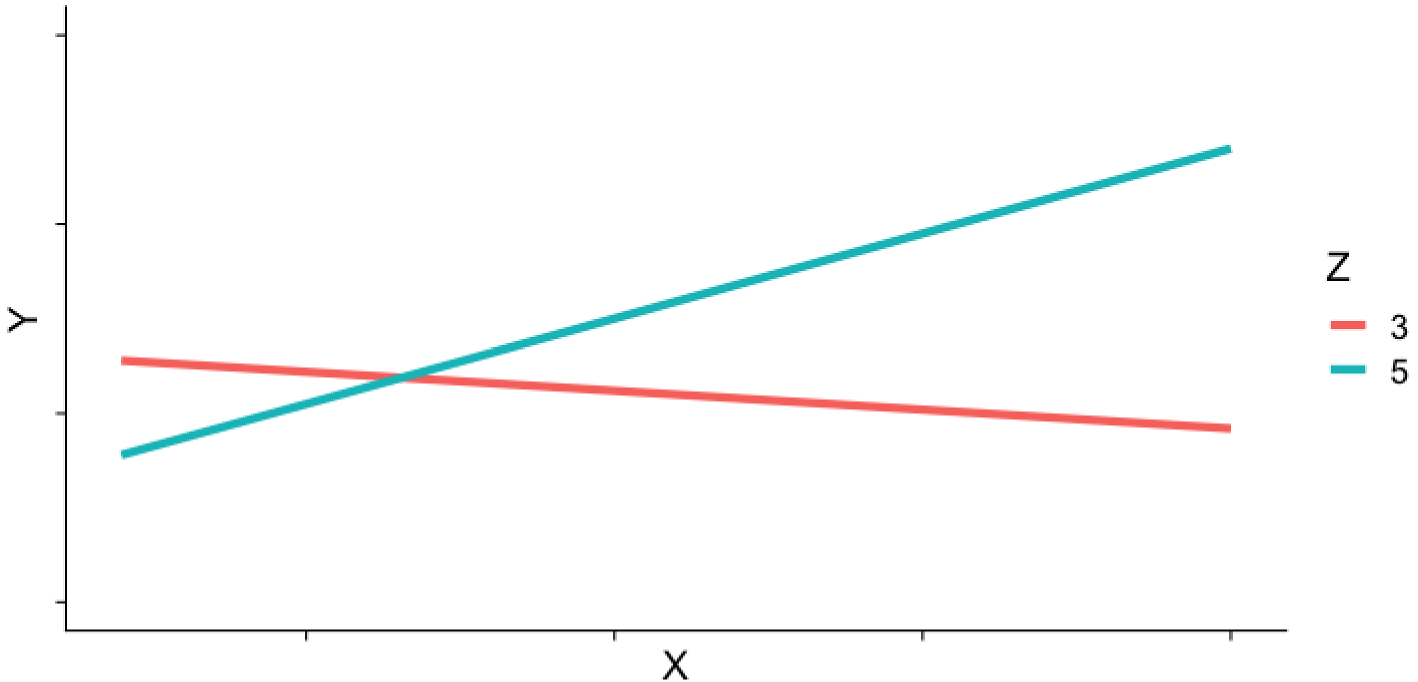
Interaction shapes

Ordinal interactions

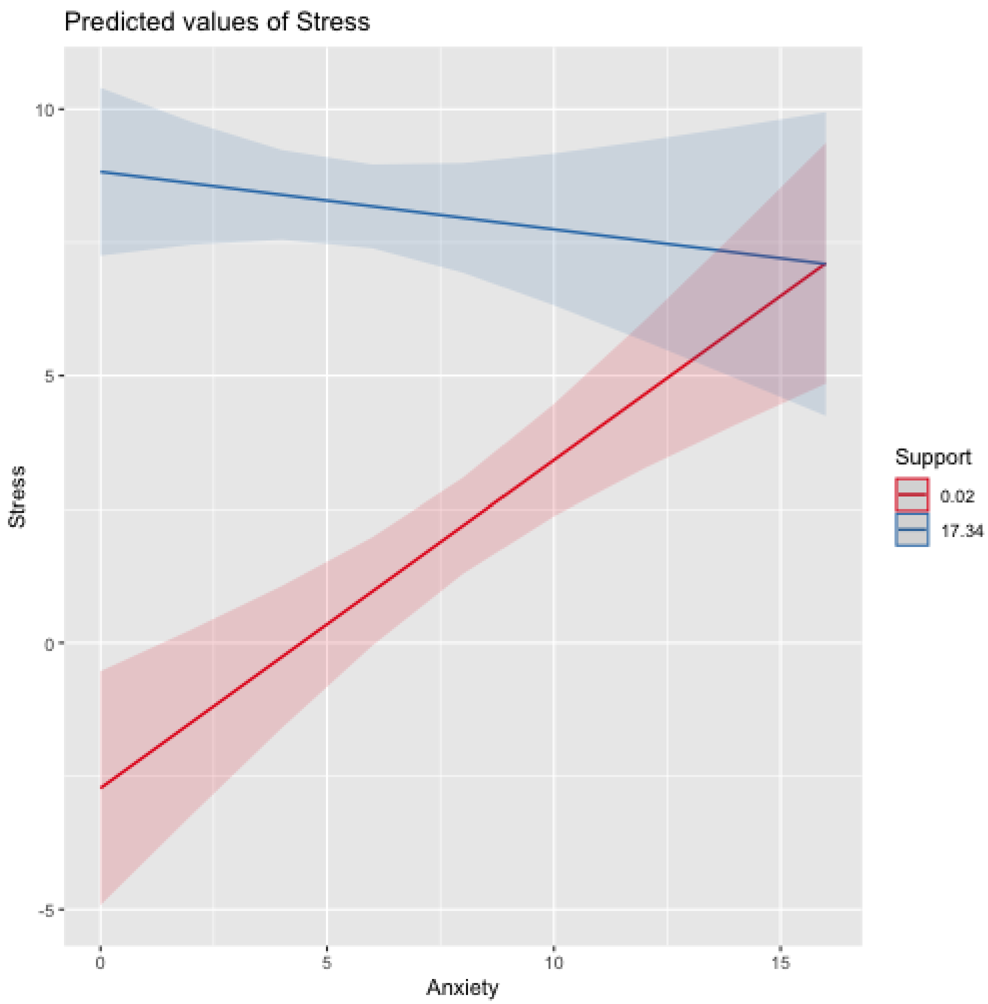


Interaction shapes

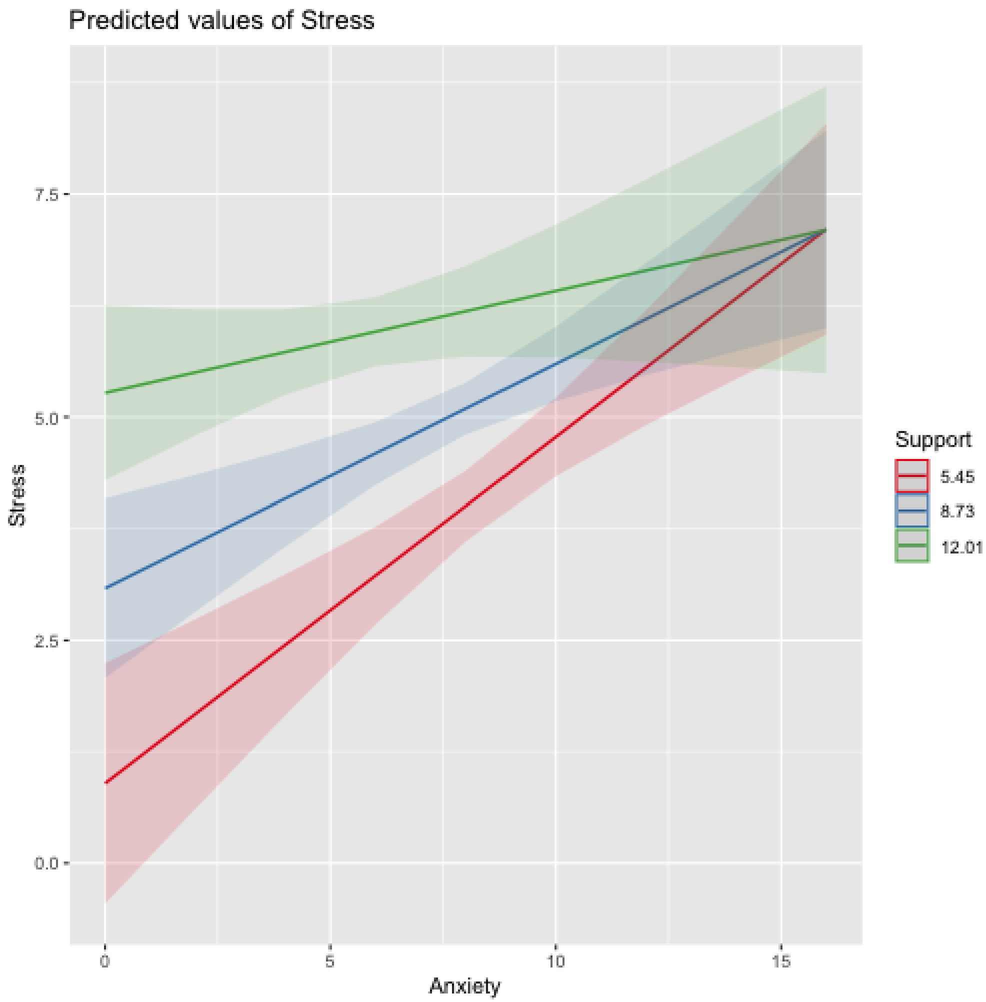
Cross-over (disordinal) interactions



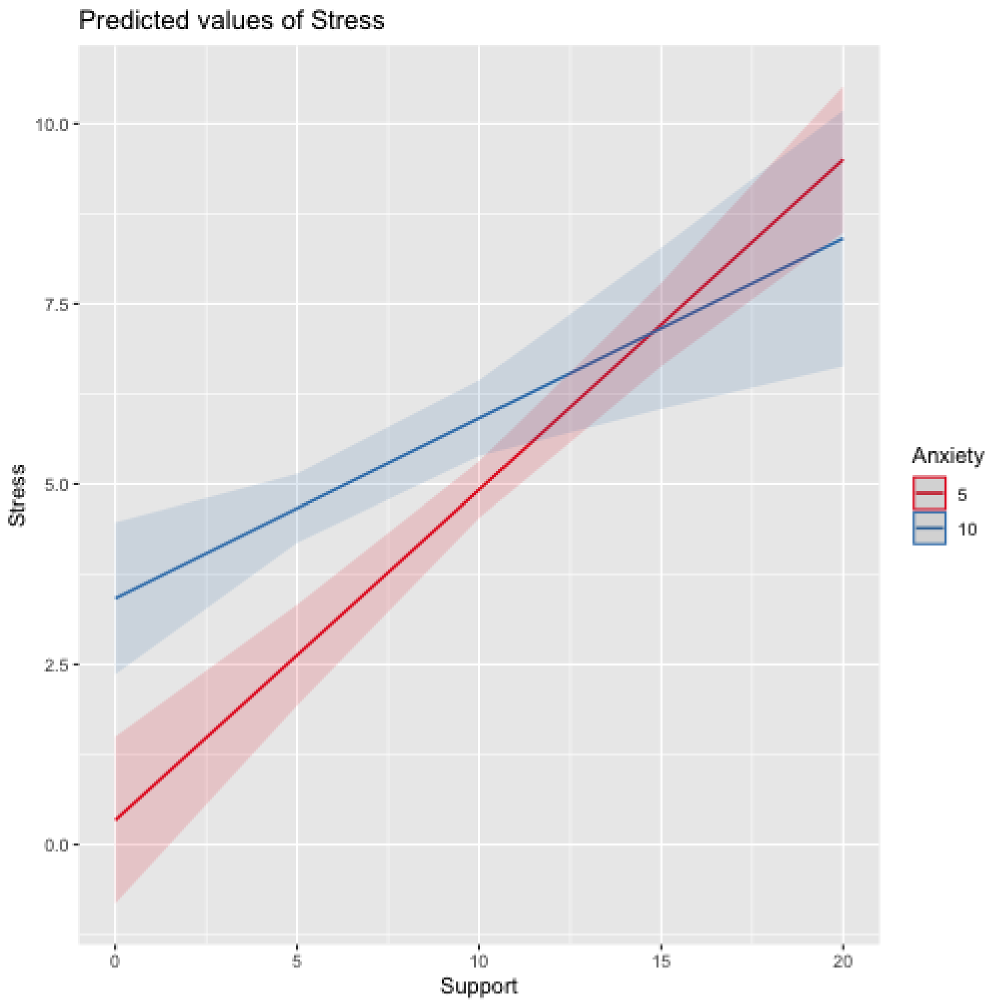
```r
library(sjPlot)
plot_model(imodel, type = "int")
```



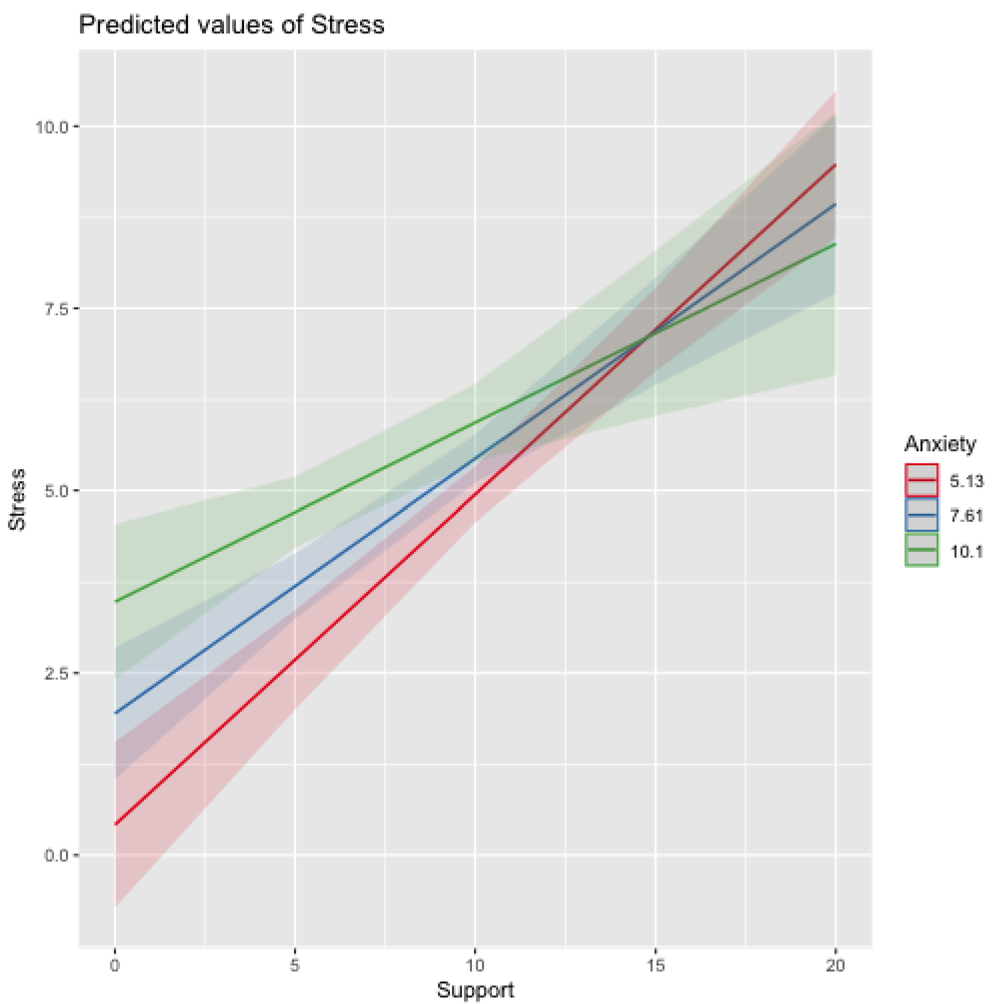
```r
plot_model(imodel, type = "int", mdrt.values = "meansd")
```



```
plot_model(imodel, type = "pred", terms = c("Support", "Anxiety[5,10]"))
```



```
plot_model(imodel, type = "pred", terms = c("Support", "Anxiety"), mdrt.
```



Simple slopes - Significance tests

$$\hat{Stress} = -2.74 + 0.62(\text{Anx}) + 0.67(\text{Sup}) + -0.04(\text{Anx} \times \text{Sup})$$

We want to know whether anxiety is a significant predictor of stress at different levels of support.

```r
library(reghelper)
simple_slopes(imodel, levels = list(Support = c(4,6,8,10,12)))
```

```
##   Anxiety Support Test Estimate Std. Error t value  df  Pr(>|t|) Sig.
## 1  sstest       4        0.4486     0.0886  5.0617 114 1.610e-06  ***
## 2  sstest       6        0.3652     0.0733  4.9791 114 2.289e-06  ***
## 3  sstest       8        0.2817     0.0654  4.3095 114 3.488e-05  ***
## 4  sstest      10        0.1982     0.0674  2.9424 114  0.003946   **
## 5  sstest      12        0.1147     0.0786  1.4600 114  0.147036
```

If you don't list levels, then this function will test simple slopes at the mean and 1 SD above and below the mean.

Simple slopes - Significance tests

What if you want to compare slopes to each other? How would we test this?

The test of the interaction coefficient is equivalent to the test of the difference in slopes at levels of Z separated by 1 unit.

```
coef(summary(imodel))
```

```
##                     Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)      -2.73966246 1.12100519 -2.443934 1.606052e-02
## Anxiety           0.61561220 0.13010161  4.731780 6.435373e-06
## Support           0.66696689 0.09547464  6.985802 2.017698e-10
## Anxiety:Support -0.04174076 0.01309328 -3.187954 1.849736e-03
```

Centering

The regression equation built using the raw data is not only diffiuclt to interpret, but often the terms displayed are not relevant to the hypotheses we're interested.

- b_0 is the expected value when all predictors are 0, but this may never happen in real life
- b_1 is the slope of X when Z is equal to 0, but this may not ever happen either.

Centering your variables by subracting the mean from all values can improve the interpretation of your results.

- Remember, a linear transformation does not change associations (correlations) between variables. In this case, it only changes the interpretation for some coefficients

Centering

```
stress.data = stress.data %>%
  mutate(Anxiety.c = Anxiety - mean(Anxiety),
         Support.c = Support - mean(Support))
head(stress.data[,c("Anxiety", "Anxiety.c", "Support", "Support.c")])
```

```
##    Anxiety   Anxiety.c Support  Support.c
## 1 10.18520  2.57086873  6.1602 -2.5697997
## 2  5.58873 -2.02560127  8.9069  0.1769003
## 3  6.58500 -1.02933127 10.5433  1.8133003
## 4  8.95430  1.33996873 11.4605  2.7305003
## 5  7.59910 -0.01523127  5.5516 -3.1783997
## 6  8.15600  0.54166873  7.5117 -1.2182997
```

DO NOT CENTER YOUR DEPENDENT VARIABLE (Y; STRESS)

```r
summary(lm(Stress ~ Anxiety.c*Support.c, data = stress.data))
```

```
##
## Call:
## lm(formula = Stress ~ Anxiety.c * Support.c, data = stress.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8163 -1.0783  0.0373  0.9200  3.6109
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.99580    0.14647  34.108  < 2e-16 ***
## Anxiety.c            0.25122    0.06489   3.872 0.000181 ***
## Support.c            0.34914    0.05238   6.666 9.82e-10 ***
## Anxiety.c:Support.c -0.04174    0.01309  -3.188 0.001850 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.462 on 114 degrees of freedom
## Multiple R-squared:  0.4084,    Adjusted R-squared:  0.3928
## F-statistic: 26.23 on 3 and 114 DF,  p-value: 5.645e-13
```

```r
summary(imodel)
```

```
##
## Call:
## lm(formula = Stress ~ Anxiety * Support, data = stress.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8163 -1.0783  0.0373  0.9200  3.6109
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -2.73966    1.12101  -2.444  0.01606 *
## Anxiety          0.61561    0.13010   4.732 6.44e-06 ***
## Support          0.66697    0.09547   6.986 2.02e-10 ***
## Anxiety:Support -0.04174    0.01309  -3.188  0.00185 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.462 on 114 degrees of freedom
## Multiple R-squared:  0.4084,    Adjusted R-squared:  0.3928
## F-statistic: 26.23 on 3 and 114 DF,  p-value: 5.645e-13
```

What changed? What stayed the same?

Standardized regression equation

So far, we've only discussed the unstandardized regression equation. If you're interested in getting the standardized regression equation, you can follow the same procedure of standardizing your variables first and then entering them into your linear model.

An important note: You must take the product of the Z-scores, not the Z-score of the products to get the correct regression model.

This is OK

```
Y ~ z(X) + z(Z) + z(X)*z(Z)

Y ~ z(X)*z(Z)
```

This is not OK

```
Y ~ z(X) + z(Z) + z(X*Z)
```

Extensions of Interactions

Interactions are all over the place and we can extend these concetps out:

- Mixing continuous & categorical variables. *"does the slop of x & y change between group 1 and group 2?"*
- *ANOVAs are regressions*
- Polynomials are also interactions

Mixing categorical and continuous

Consider the case where D is a variable representing two groups. In a univariate regression, how do we interpret the coefficient for D?

$$\hat{Y} = b_0 + b_1 D$$

b_0 is the mean of the reference group, and D represents the difference in means between the two groups.

Interpreting slopes

Extending this to the multivariate case, where X is continuous and D is a dummy code representing two groups.

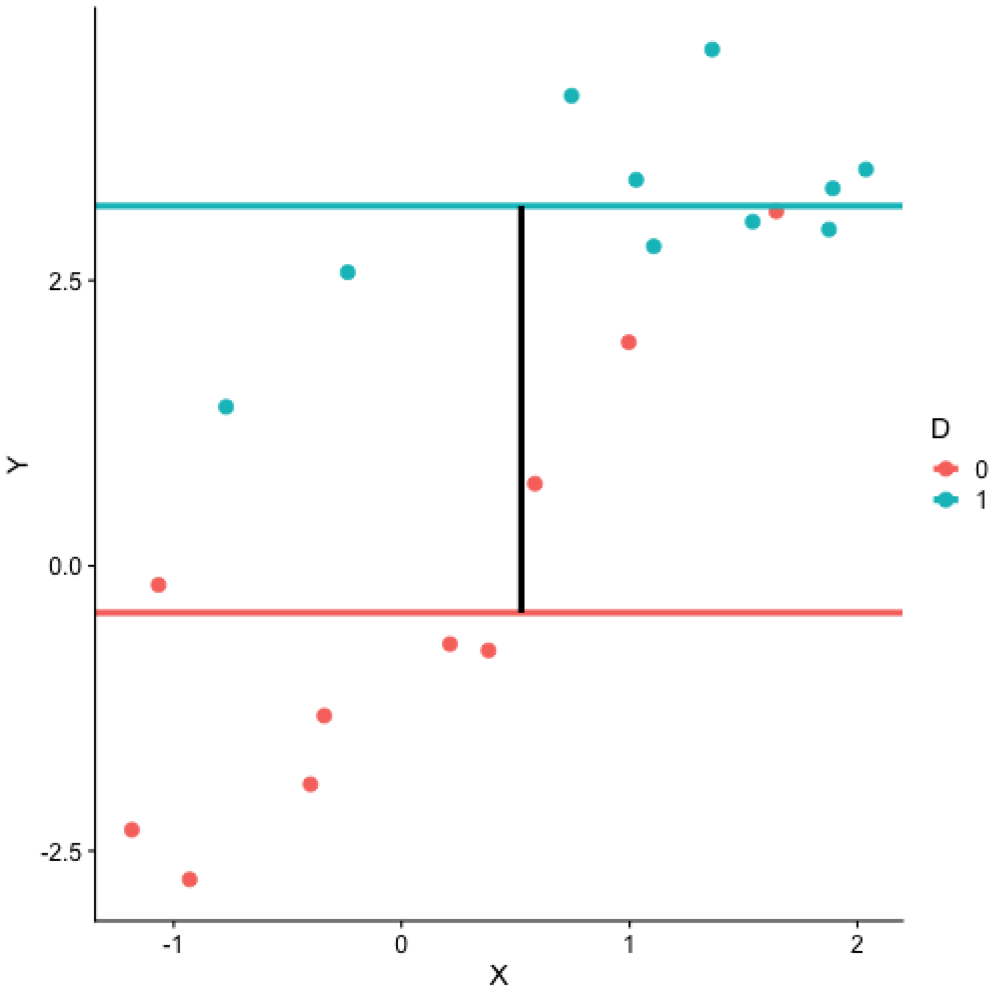
$$\hat{Y} = b_0 + b_1 D + b_2 X$$

How do we interpret b_1?

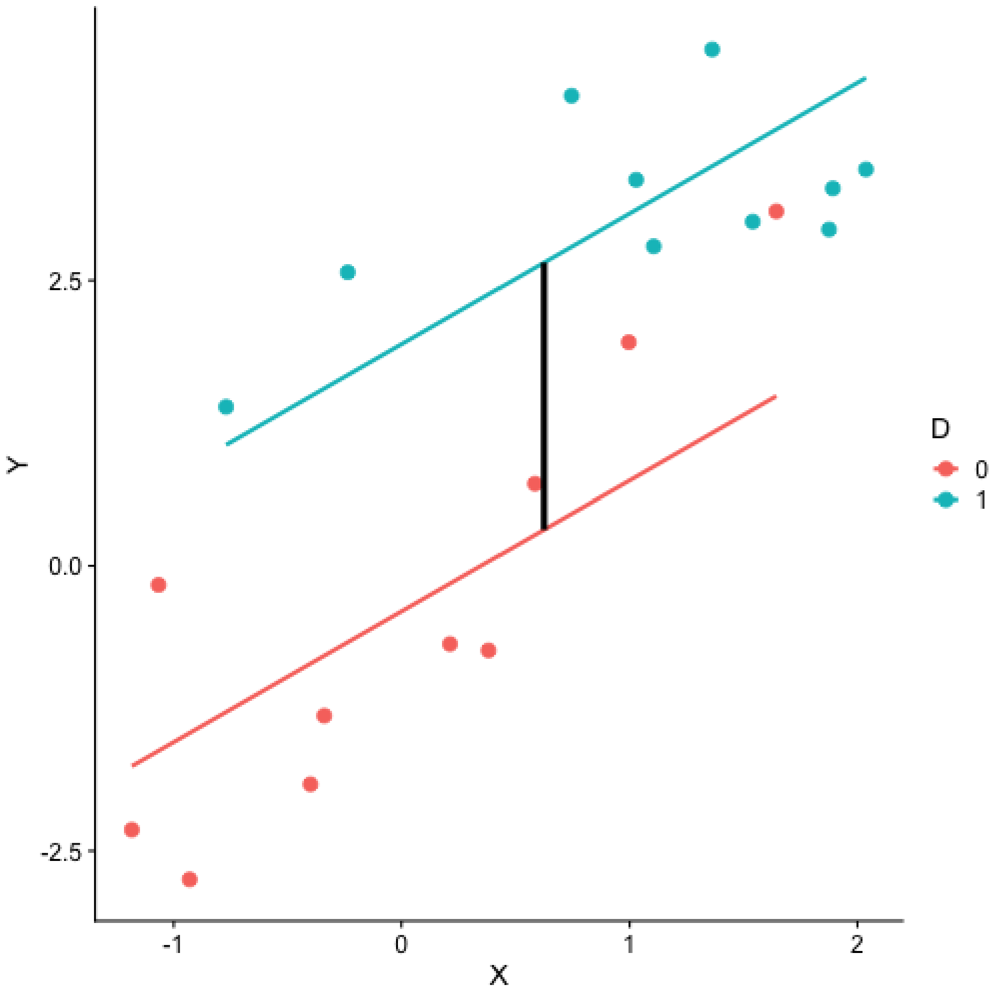
b_1 is the difference in means between the two groups *if the two groups have the same average level of X* or holding X constant.

This, by the way, is ANCOVA.

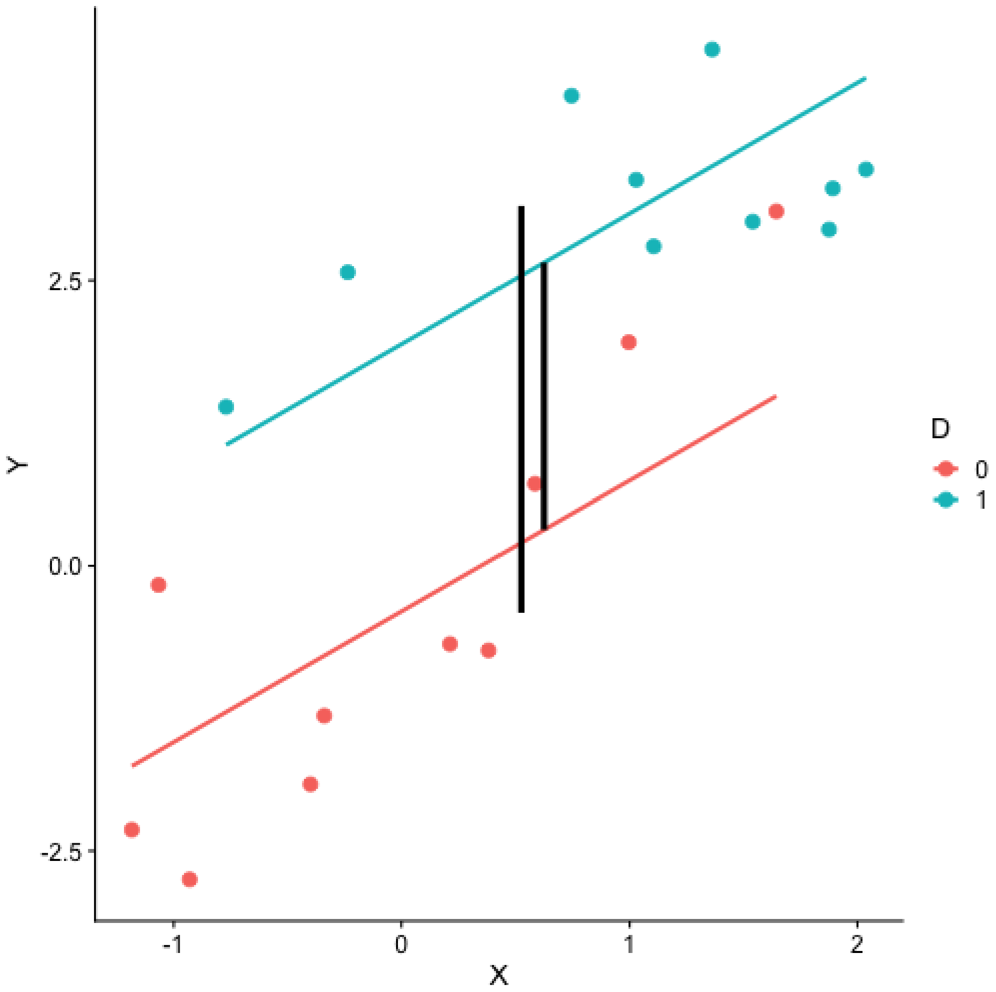
Visualizing



Visualizing



Visualizing



Interactions

Now extend this example to include joint effects, not just additive effects:

$$\hat{Y} = b_0 + b_1 D + b_2 X + b_3 DX$$

How do we interpret b_1?

b_1 is the difference in means between the two groups *when X is 0*.

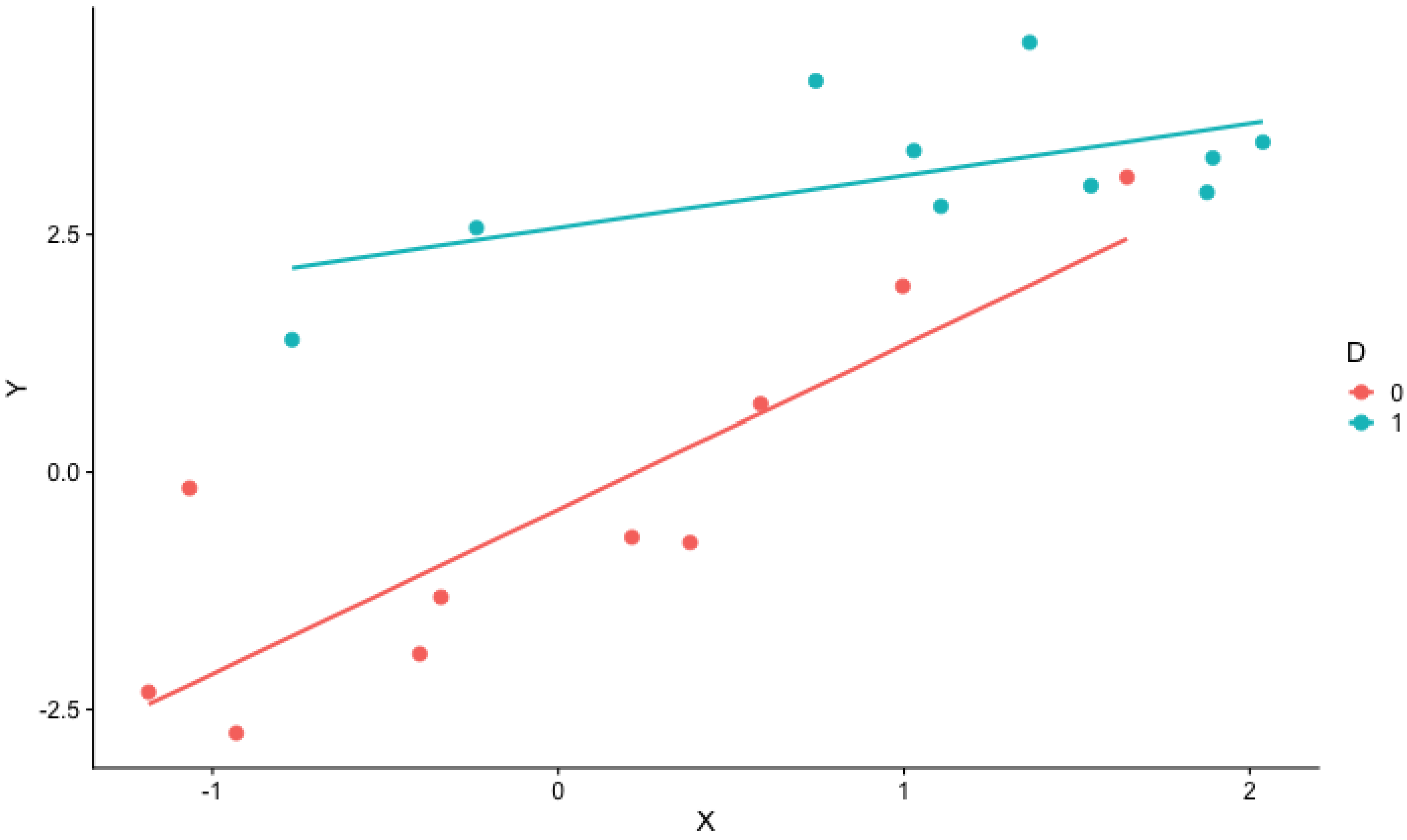
What is the interpretation of b_2?

b_2 is the slope of X among the reference group.

What is the interpretation of b_3?

b_3 is the difference in slopes between the reference group and the other group.

Visualizing



Polynomial Regression

Polynomial regression (nonlinear) is most often a form of hierearchical regression that systematically tests a series of higher order functions for a single variable.

$$\textbf{Linear: } \hat{Y} = b_0 + b_1 X$$

$$\textbf{Quadtratic: } \hat{Y} = b_0 + b_1 X + b_2 X^2$$

$$\textbf{Cubic: } \hat{Y} = b_0 + b_1 X + b_2 X^2 + b_3 X^3$$

You need 16x the sample size to detect an interaction as you need for a main effect of the same size