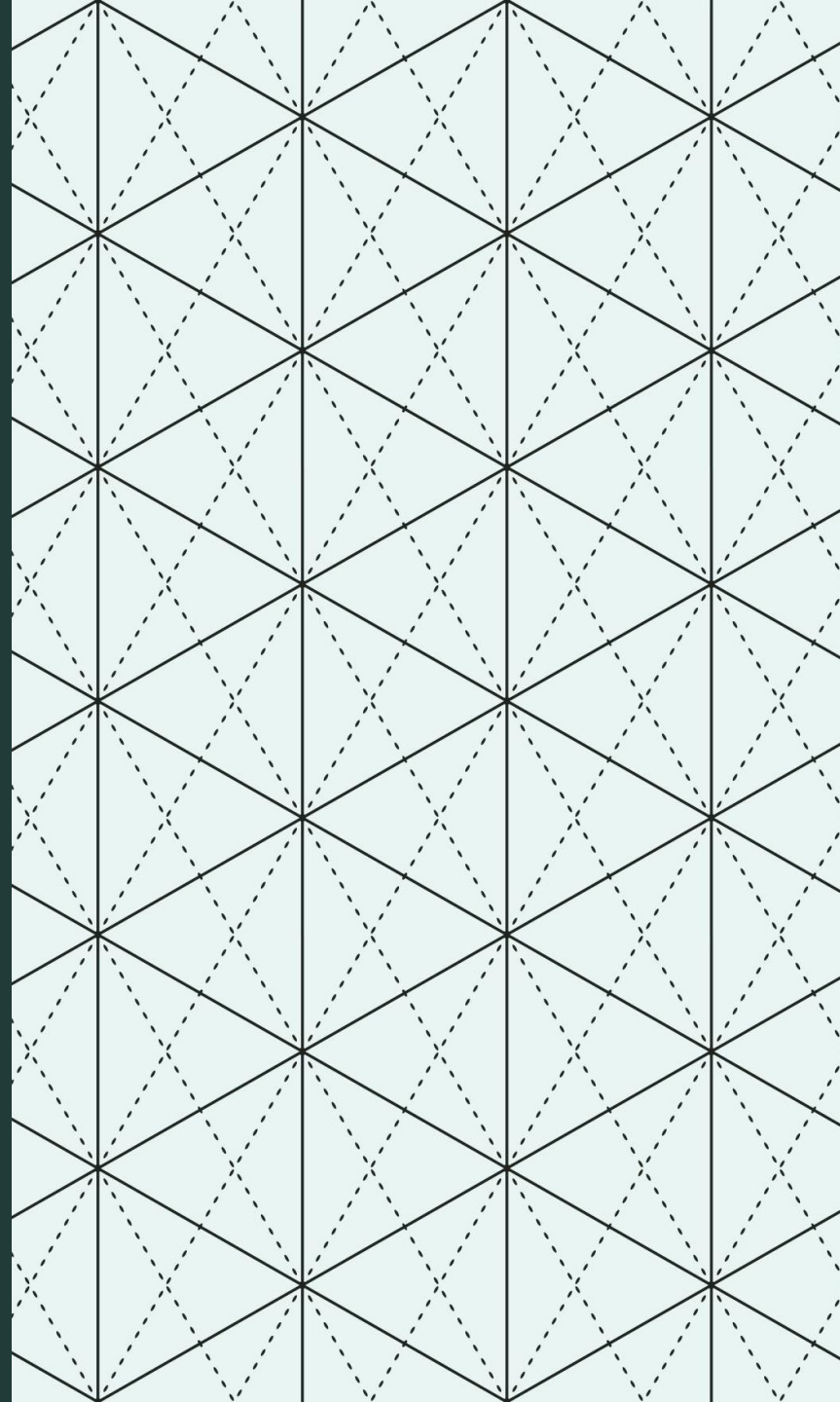
WELCOME TO PSY 653 LAB!

MODULE 08: RANDOM COEFFICIENTS MODEL



OBJECTIVES

1. Quick overview of random coefficient modeling
2. Dataset description
3. Load in data
4. Learn to pivot data from a wide format to long format
5. Visualize data
6. Model data

RANDOM COEFFICIENT MODELING

- The first step to understanding multilevel modeling
- Random coefficient modeling utilizes Maximum Likelihood (ML) Estimation
  - This is in contrast to Ordinary Least squares (OLS) which we have been using previously

DEFINING **FIXED** AND **RANDOM** EFFECTS

These definitions are a little different in multilevel analyses than in ANOVAs.

Fixed effect = does not vary over subjects of groups – average value of slope or intercept (i.e., *what is the estimate of the effect across all of the groups?*)

Random effect = might vary across subjects or groups – intercepts and slopes might be calculated for each group or each subject to see if they vary meaningfully (i.e., *how much does the estimate for the effect vary across the groups?*)

CLASS EXAMPLE

A research team is interested if student performance in coding skills increased over time during a coding class. During each week of the program, all participants completed a coding challenge. Each challenge had a set of coding skills that had to be employed to solve the challenge, but each challenge focused on solving some substantive problem (e.g., mapping social networks of users on an online forum, developing an algorithm to recommend new music based on a user's Spotify history, etc.). Each student's performance on the challenge was graded by the research team using a valid and reliable rubric able to detect growth in skills over time.

This dataset was provided by Kim Henry, PhD.

CREATE A NEW R-PROJECT AND R-NOTEBOOK!

Download the “mlm_grow.csv” file from Canvas and save it into your R-project file

Load libraries

```{r}

install.packages("lme4")
install.packages("lmerTest")

library(lme4)
library(lmerTest)
library(psych)
library(olsrr)
library(tidyverse)
```

IMPORT DATA

- **kid_id:** The student's ID number.
- **team_id**: The team ID of the student.
- **txcond**: The treatment condition, 0 = individual-focused condition, 1 = team-focused condition.
- **perf_0 - perf_6:** Performance on the coding challenge each week of the program, starting at baseline (before the program began (perf_0) and ending at week 6 (perf_6). The score ranges from 0 to 10.

```
# read in data
```{r}
grow_wide <- read_csv("mlm_grow.csv")
```
```

```
-- Column specification --------------
--------------------------------------
cols(
  kid_id = col_double(),
  team_id = col_double(),
  txcond = col_double(),
  perf_0 = col_double(),
  perf_1 = col_double(),
  perf_2 = col_double(),
  perf_3 = col_double(),
  perf_4 = col_double(),
  perf_5 = col_double(),
  perf_6 = col_double()
)
```

THE DATA IS WIDE, AND WE WANT TO CONVERT IT TO LONG

Filter

	kid_id	team_id	txcond	perf_0	perf_1	perf_2	perf_3	perf_4	perf_5	perf_6
1	5	1	1	6.332146	6.507988	6.888969	5.563810	6.152750	5.984029	4.932417
2	10	2	1	4.531640	5.602623	6.851863	6.076729	5.586186	6.255608	8.042116
3	15	3	1	4.902634	6.072946	4.830770	6.623872	7.695781	8.171289	9.632302
4	20	4	1	5.164947	6.640575	5.485294	6.583001	6.021457	5.097809	5.640439
5	25	5	1	5.130913	4.438954	5.311976	6.173483	3.726906	6.246301	4.886199
6	30	6	1	6.014382	2.939461	3.540898	6.626733	6.748707	6.872268	7.002733
7	35	7	1	5.067151	2.286552	3.113482	3.752372	5.940044	4.139538	4.478529
8	40	8	1	6.423918	5.729861	4.878145	7.154612	5.338027	7.261612	6.980742
9	45	9	1	2.316031	3.078097	3.149725	3.258672	2.403630	2.930873	3.377603
10	50	10	1	5.932377	5.773619	5.927979	5.790073	6.985271	7.120951	9.056178

Filter

	kid_id	team_id	txcond	week	perform
1	5	1	1	0	6.332146
2	5	1	1	1	6.507988
3	5	1	1	2	6.888969
4	5	1	1	3	5.563810
5	5	1	1	4	6.152750
6	5	1	1	5	5.984029
7	5	1	1	6	4.932417
8	10	2	1	0	4.531640
9	10	2	1	1	5.602623
10	10	2	1	2	6.851863

PIVOT_LONGER(): TURNS WIDE DATASETS TO LONG

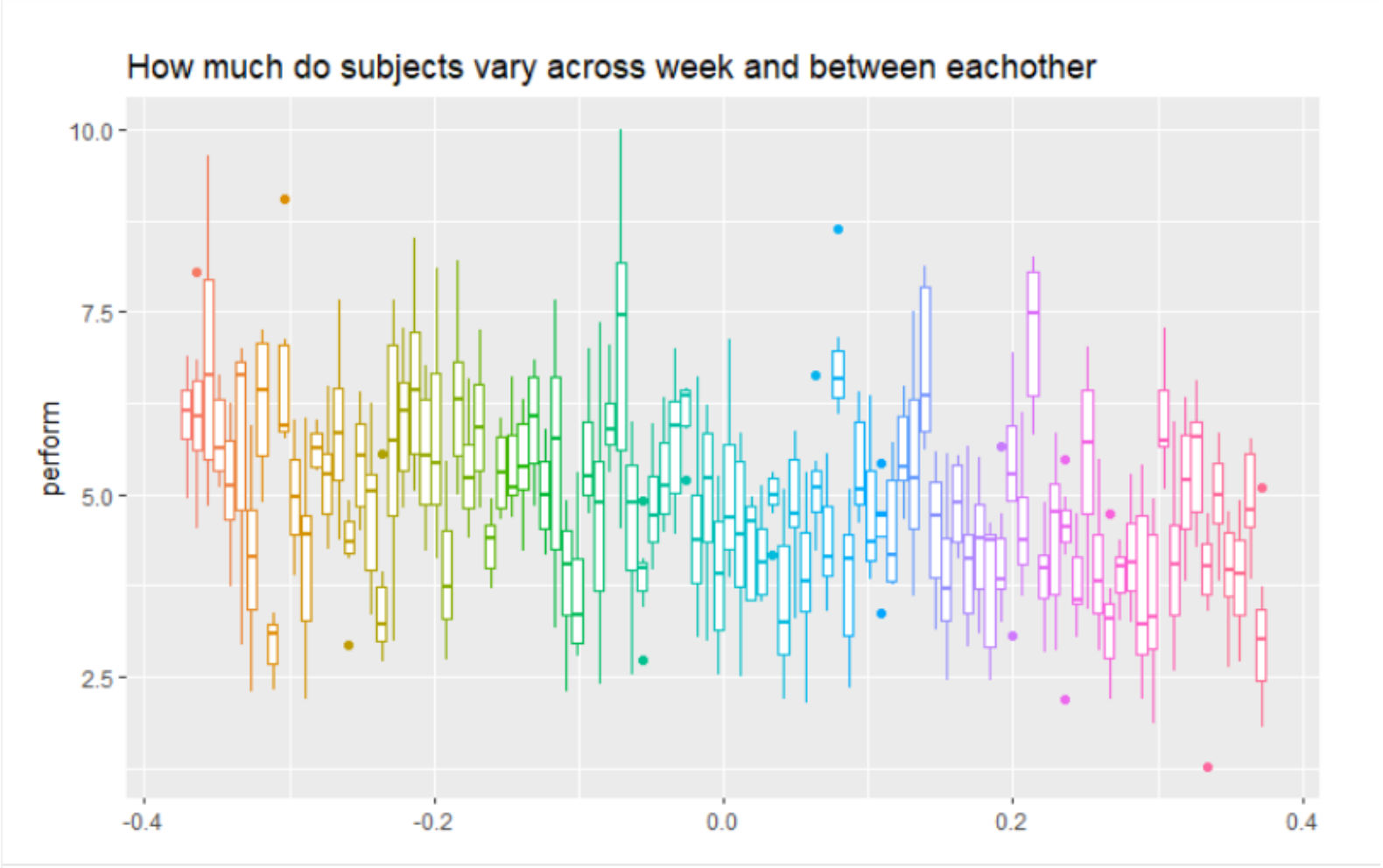
Learn to pivot tables
We will turn this dataset from wide to long

```{r}
grow_long <- pivot_longer(grow_wide,
                          cols = perf_0:perf_6,
                          names_to = "week",
                          names_prefix = "perf_",
                          values_to = "perform")

# Turn week into a numeric variable
grow_long <- mutate(grow_long, week = as.numeric(week))

```

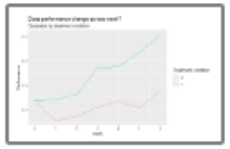
```
# Learn to pivot tables
We will turn this dataset from wide to long
```{r}
grow_long <- pivot_longer(grow_wide,
                          cols = perf_0:perf_6,
                          names_to = "week",
                          names_prefix = "perf_",
                          values_to = "perform")

# Turn week into a numeric variable
grow_long <- mutate(grow_long, week = as.numeric(week))

```
```

Converts week to a numeric variable (Otherwise it will be read as a character variable)

- **pivot_longer**: function
- **grow_wide**: dataset name
- **cols = perf_0:perf_6**: columns to convert to long (A colon indicates take all the variables inbetween)
- **names_to: "week"**: what to name the column that the variable names go to.
- **names_prefix = "perf_"**: remove the prefix of "perf_" from the names column
- **values_to = "perform"**: what to name the values column

```
# Describe data
```{r}
describe(grow_long)
```
```

	vars <dbl>	n <dbl>	mean <dbl>	sd <dbl>	median <dbl>	trimmed <dbl>	mad <dbl>	min <dbl>	max <dbl>
kid_id	1	700	252.5	144.43	252.50	252.50	185.32	5.00	500
team_id	2	700	50.5	28.89	50.50	50.50	37.06	1.00	100
txcond	3	700	0.5	0.50	0.50	0.50	0.74	0.00	1
week	4	700	3.0	2.00	3.00	3.00	2.97	0.00	6
perform	5	700	4.9	1.33	4.88	4.87	1.32	1.27	10

5 rows | 1-10 of 13 columns

```
# Plot the change in performance
```{r}
ggplot(grow_long, aes(y = perform, color = factor(kid_id))) +
  geom_boxplot() +
  theme(legend.position = "none") +
  labs(title = "How much do subjects vary across week and between eachother")
```
```

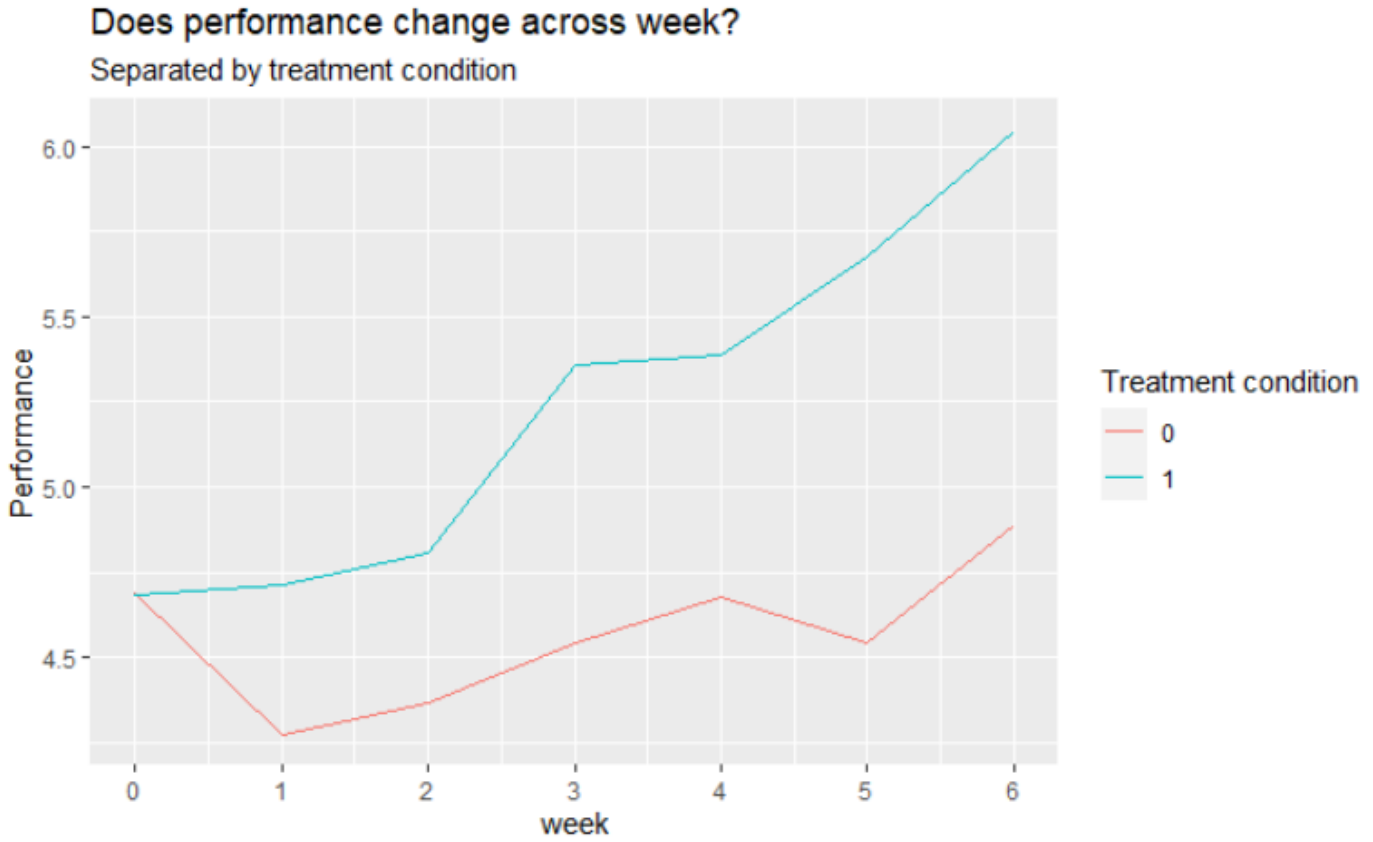


VISUALIZE THE AMOUNT OF VARIATION BETWEEN PARTICIPANTS

VISUALIZE THE DATA -- BY TREATMENT GROUP

```r
grouped_weektx <- group_by(grow_long, week, txcond)
mean_perftx <- summarize(grouped_weektx, perform = mean(perform, na.rm = TRUE))
mean_perftx <- ungroup(mean_perftx)


ggplot(mean_perftx, aes(x = week, y = perform, color = as.factor(txcond))) +
  geom_line() +
  scale_x_continuous(limits = c(0,6), breaks = seq(0,6, by = 1)) +
  labs(title = "Does performance change across week?", subtitle = "Separated by treatment condition
       y = "Performance", color = "Treatment condition")
```





The lmer() function

Similar to a lm, the dependent variable is listed, then a tilde. Since, this is an unconditional model, there are no predictors, but we include a 1 to denote the intercept. This is called the fixed effects part of the model and will provide us with the mean means across the groups.

lmer is the function used to specify a of multilevel model (it stands for linear mixed effects regression).

```
mod1 = lmer(score ~ 1 + (1|team_id), REML = FALSE, data = teams)
summary(mod1)
```

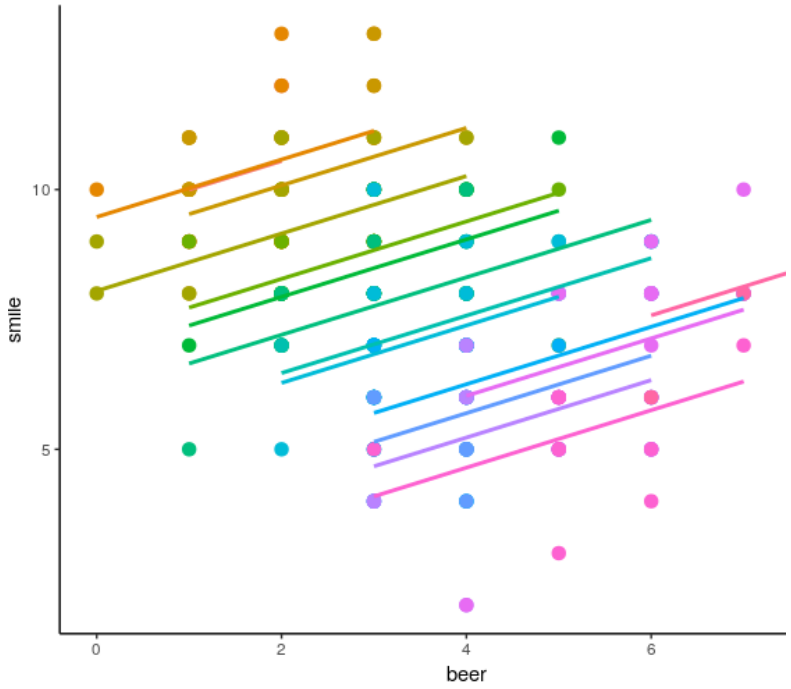
After the fixed effects, we provide the random effects. Here we list

the effects that we want to denote as random. In this case it is just the intercept (1), which will capture the between group variability. The bar (|) and then team_id denotes the Level 2 grouping variable.

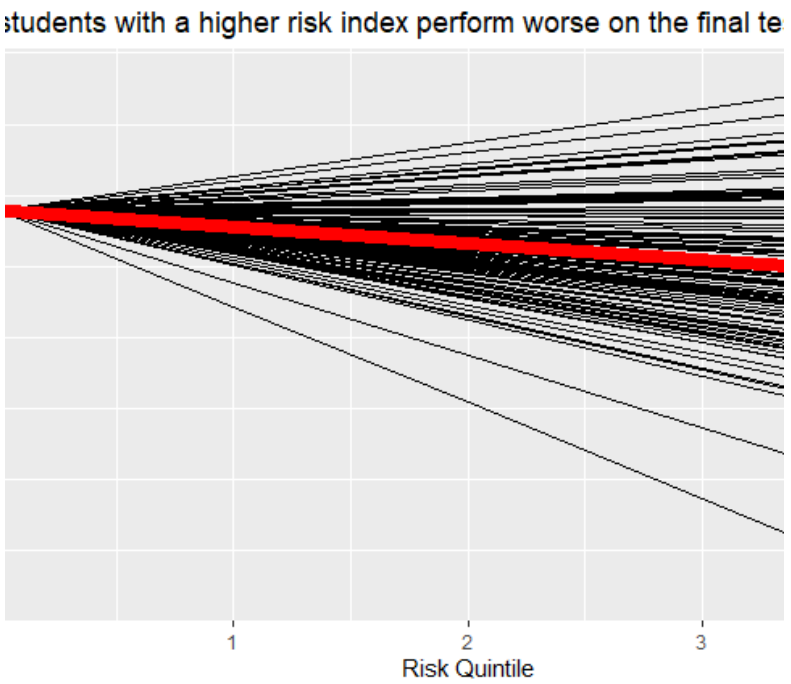
© Kim Henry

RANDOM INTERCEPT, RANDOM SLOPE

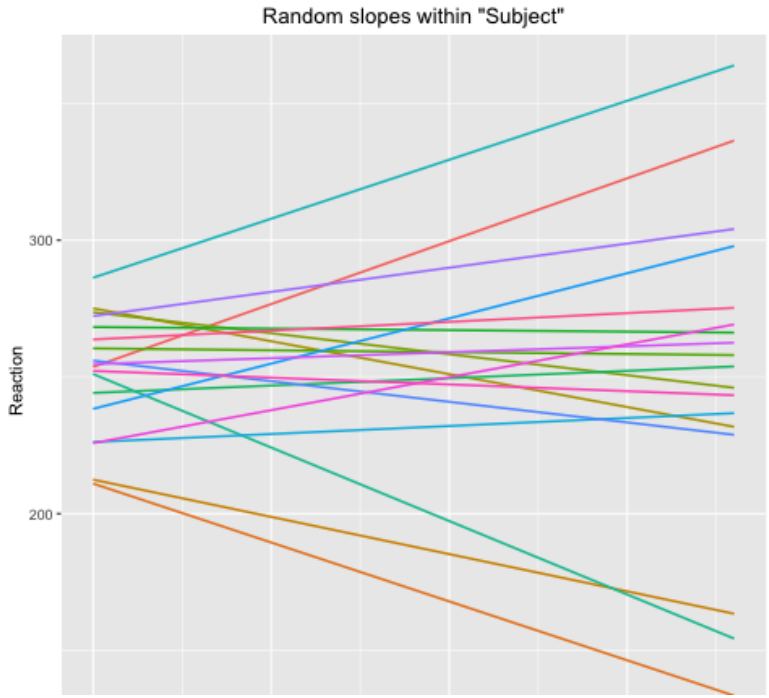
RANDOM INTERCEPT MODEL



RANDOM SLOPE MODEL



RANDOM INTERCEPT **AND** SLOPE MODEL



MODEL 1: RANDOM INTERCEPT ONLY

```r
growmod1 <- lmer(perform ~  (1|kid_id), data = grow_long, REML = FALSE)
summary(growmod1)
```

```
Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: perform ~ (1 | kid_id)
   Data: grow_long

     AIC      BIC   logLik deviance df.resid
  2169.3   2182.9  -1081.6   2163.3      697

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.6458 -0.6426 -0.0098  0.6005  3.2779

Random effects:
 Groups   Name        Variance Std.Dev.
 kid_id   (Intercept) 0.7918   0.8899
 Residual             0.9821   0.9910
Number of obs: 700, groups:  kid_id, 100

Fixed effects:
            Estimate Std. Error        df t value Pr(>|t|)
(Intercept)   4.90455    0.09655  99.99994    50.8   <2e-16 ***
---
Signif. codes:  0 â€˜***â€™ 0.001 â€˜**â€™ 0.01 â€˜*â€™ 0.05 â€˜.â€™ 0.1 â€˜ â€™ 1
```

MODEL 1: RANDOM INTERCEPT ONLY

```r
growmod1 <- lmer(perform ~  (1|kid_id), data = grow_long, REML = FALSE)
summary(growmod1)
```

```
Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: perform ~ (1 | kid_id)
   Data: grow_long

     AIC      BIC   logLik deviance df.resid
  2169.3   2182.9  -1081.6   2163.3      697

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.6458 -0.6426 -0.0098  0.6005  3.2779

Random effects:
 Groups   Name        Variance Std.Dev.
 kid_id   (Intercept) 0.7918   0.8899
 Residual             0.9821   0.9910
Number of obs: 700, groups:  kid_id, 100

Fixed effects:
            Estimate Std. Error       df t value Pr(>|t|)
(Intercept)  4.90455    0.09655 99.99994    50.8   <2e-16 ***
---
Signif. codes:  0 â€˜***â€™ 0.001 â€˜**â€™ 0.01 â€˜*â€™ 0.05 â€˜.â€™ 0.1 â€˜ â€™ 1
```

On average, kid intercepts vary by .8899 std. deviations

In the absence of fixed effects, this is the Grand Mean

MODEL 2: RANDOM INTERCEPT AND FIXED SLOPE

```{r}

growmod2 <- lmer(perform ~  txcond + (1|kid_id), data = grow_long, REML = FALSE)
summary(growmod2)
```

```
Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: perform ~ txcond + (1 | kid_id)
   Data: grow_long

     AIC      BIC   logLik deviance df.resid
  2158.4   2176.6  -1075.2   2150.4      696

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.6875 -0.6269 -0.0379  0.6080  3.2653

Random effects:
 Groups   Name        Variance Std.Dev.
 kid_id   (Intercept) 0.6796   0.8244
 Residual             0.9821   0.9910
Number of obs: 700, groups:  kid_id, 100

Fixed effects:
            Estimate Std. Error       df t value Pr(>|t|)
(Intercept)   4.5696     0.1281 100.0000  35.684  < 2e-16 ***
txcond        0.6700     0.1811 100.0000   3.699 0.000353 ***
---
Signif. codes:  0 â€˜***â€™ 0.001 â€˜**â€™ 0.01 â€˜*â€™ 0.05 â€˜.â€™ 0.1 â€˜ â€™ 1

Correlation of Fixed Effects:
       (Intr)
txcond -0.707
```

MODEL 2: RANDOM INTERCEPT AND FIXED SLOPE

```
```{r}

growmod2 <- lmer(perform ~  txcond + (1|kid_id), data = grow_long, REML = FALSE)
summary(growmod2)
```
```

```
Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: perform ~ txcond + (1 | kid_id)
   Data: grow_long

     AIC      BIC   logLik deviance df.resid
  2158.4   2176.6  -1075.2   2150.4      696

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.6875 -0.6269 -0.0379  0.6080  3.2653

Random effects:
 Groups   Name        Variance Std.Dev.
 kid_id   (Intercept) 0.6796   0.8244
 Residual             0.9821   0.9910
Number of obs: 700, groups:  kid_id, 100

Fixed effects:
            Estimate Std. Error       df t value Pr(>|t|)
(Intercept)   4.5696     0.1281 100.0000  35.684  < 2e-16 ***
txcond        0.6700     0.1811 100.0000   3.699 0.000353 ***
---
Signif. codes:  0 â€˜***â€™ 0.001 â€˜**â€™ 0.01 â€˜*â€™ 0.05 â€˜.â€™ 0.1 â€˜ â€™ 1

Correlation of Fixed Effects:
       (Intr)
txcond -0.707
```

On average, while incorporating txcond, the intercept varies by .8244 std. deviations

Average intercept for kids in txcond = 0

Average change of moving from txcond = 0 to txcond = 1

MODEL 3: RANDOM INTERCEPT, RANDOM SLOPE, AND FIXED SLOPE

```r
growmod3 <- lmer(perform ~  txcond + (1 + week|kid_id), data = grow_long, REML = FALSE)
summary(growmod3)
```

```
Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's method [lmerModLmerTest
]
Formula: perform ~ txcond + (1 + week | kid_id)
   Data: grow_long

     AIC      BIC   logLik deviance df.resid
  2066.3   2093.6  -1027.2   2054.3      694

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2148 -0.5633  0.0296  0.6327  3.0298

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 kid_id   (Intercept) 0.46116  0.6791
          week        0.05779  0.2404   -0.18
 Residual             0.71247  0.8441
Number of obs: 700, groups:  kid_id, 100

Fixed effects:
            Estimate Std. Error       df t value Pr(>|t|)
(Intercept)   4.4797     0.1147 100.0007  39.054   <2e-16 ***
txcond        0.3519     0.1622 100.0007   2.169   0.0324 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
       (Intr)
txcond -0.707
```

MODEL 3: RANDOM INTERCEPT, RANDOM SLOPE, AND FIXED SLOPE

```{r}
growmod3 <- lmer(perform ~  txcond + (1 + week|kid_id), data = grow_long, REML = FALSE)
summary(growmod3)
```

```
Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's method [
]
Formula: perform ~ txcond + (1 + week | kid_id)
   Data: grow_long

     AIC      BIC   logLik deviance df.resid
  2066.3   2093.6  -1027.2   2054.3      694

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2148 -0.5633  0.0296  0.6327  3.0298

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 kid_id   (Intercept) 0.46116  0.6791
          week        0.05779  0.2404   -0.18
 Residual             0.71247  0.8441
Number of obs: 700, groups:  kid_id, 100

Fixed effects:
            Estimate Std. Error       df t value Pr(>|t|)
(Intercept)   4.4797     0.1147 100.0007  39.054   <2e-16 ***
txcond        0.3519     0.1622 100.0007   2.169   0.0324 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
       (Intr)
txcond -0.707
```

On average, while incorporating txcond, and week the intercept varies by .6791 std. deviations

On average, slopes differ by an average of .2404 std. deviations

Average intercept for kids in txcond = 0

Average change of moving from txcond = 0 to txcond = 1

COMPARE MODELS

```r
anova(growmod1, growmod2, growmod3)
```

```
Data: grow_long
Models:
growmod1: perform ~ (1 | kid_id)
growmod2: perform ~ txcond + (1 | kid_id)
growmod3: perform ~ txcond + (1 + week | kid_id)
          npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
growmod1     3 2169.3 2182.9 -1081.6   2163.3
growmod2     4 2158.4 2176.6 -1075.2   2150.4 12.827  1  0.0003416 ***
growmod3     6 2066.3 2093.6 -1027.2   2054.3 96.111  2  < 2.2e-16 ***
---
Signif. codes:  0 â€˜***â€™ 0.001 â€˜**â€™ 0.01 â€˜*â€™ 0.05 â€˜.â€™ 0.1 â€˜ â€™ 1
```