Conditional Models

Emorie D Beck

9/7/2017

Contents

Workspace

Packages

```
library(psych)
library(sjPlot)
library(broom)
library(lme4)
library(MuMIn)
library(merTools)
library(reghelper)
library(stargazer)
library(lsmeans)
library(multcompView)
library(plyr)
library(tidyverse)
```

Background

Basic Syntex

From last week:

- **Level 1:** $Y_{ij} = \beta_{0j} + \varepsilon ij$
- **Level 2:** $\beta_{0j} = \gamma_{00} + U_{0j}$

Sample Data

The National Longitudinal Study of Youths 1979 Child and Young Adult Sample (NLSYCYA) is a longitudinal study conducted by the National Bureau of Labor Statistics. The sample includes the children of the original 1979 sample. Here, we are going to use a subset of the more than 11,000 variables available that include the following.

Item Name	Description	Time-Varying?
PROC_CID	Participant ID	No
Dem_DOB	Year of Date of Birth	No
groups	Jail, Community Service, None	No
DemPWeight	Weight Percentile at age 10	No
age	Age of participant	Yes
Year	Year of Survey	Yes
age0	Age of participant (centered)	Yes
SensSeek	Sensation-Seeking Composite	Yes
CESD	CESD Depression Composite	Yes

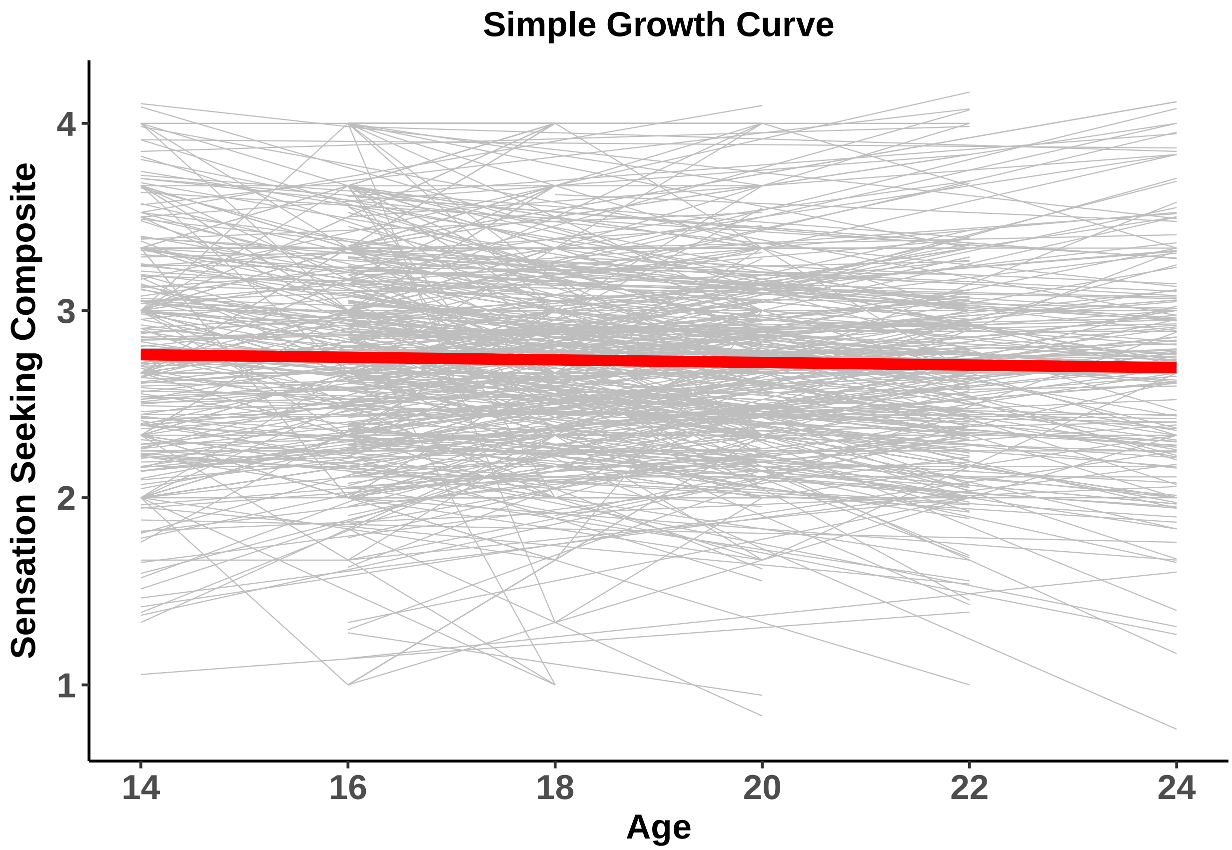
```r
data_path <- "https://github.com/longitudinal-data/1-descriptives-and-graphs-emoriebeck/raw/master/Condi
load(url(paste(data_path, "sample.RData", sep = "/")))

head(sample_dat)
```

```
## # A tibble: 6 x 8
##   PROC_CID   age  year  age0   groups      CESD SensSeek DemPweight
##      <dbl> <dbl> <dbl> <dbl>   <fctr>     <dbl>    <dbl>      <dbl>
## 1     1601    16  2006     2 CommServ 0.4285714 3.666667  0.8159399
## 2     1601    18  2008     4 CommServ 2.0000000 3.000000  0.8159399
## 3     9102    16  2012     2     None 0.1818182 3.333333  0.6712397
## 4     9501    14  2000     0     Jail 0.5000000 3.000000  0.5477584
## 5     9501    18  2004     4     Jail 0.4285714 3.000000  0.5477584
## 6     9501    22  2008     8     Jail 0.4285714 3.000000  0.5477584
```

Simple Growth Curve Model

- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * time_{ij} + \varepsilon ij$
- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + U_{1j}$



In R

```r
mod0 <- lmer(SensSeek ~ age0 + (1|PROC_CID), data = sample_dat)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: SensSeek ~ age0 + (1 | PROC_CID)
##    Data: sample_dat
##
## REML criterion at convergence: 3404.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.6782 -0.5396  0.0276  0.4739  3.2174
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  PROC_CID (Intercept) 0.1349   0.3673
##  Residual             0.2003   0.4475
## Number of obs: 2084, groups:  PROC_CID, 924
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  2.765851   0.020067  137.83
## age0        -0.005879   0.003407   -1.73
##
## Correlation of Fixed Effects:
##      (Intr)
## age0 -0.611
```

Conditional Models: Adding Predictors

Let's see if we can better predict participants' change in sensation seeking over time by adding covariates.

Predictor	Continuous	Categorical
Time Invariant	Weight for Age	Group
Time Varying	CESD Scores	Depression

Time Invariant Predictors

Time Invariant Predictors: Continuous

The basic equation, specifying a random intercept and slope:

- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * time_{1j} + \varepsilon ij$
- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + \gamma_{01} * X_{2j} + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + \gamma_{11} * X_{2j} + U_{1j}$

But we need to break this down to see that adding additional predictors results in interaction terms:

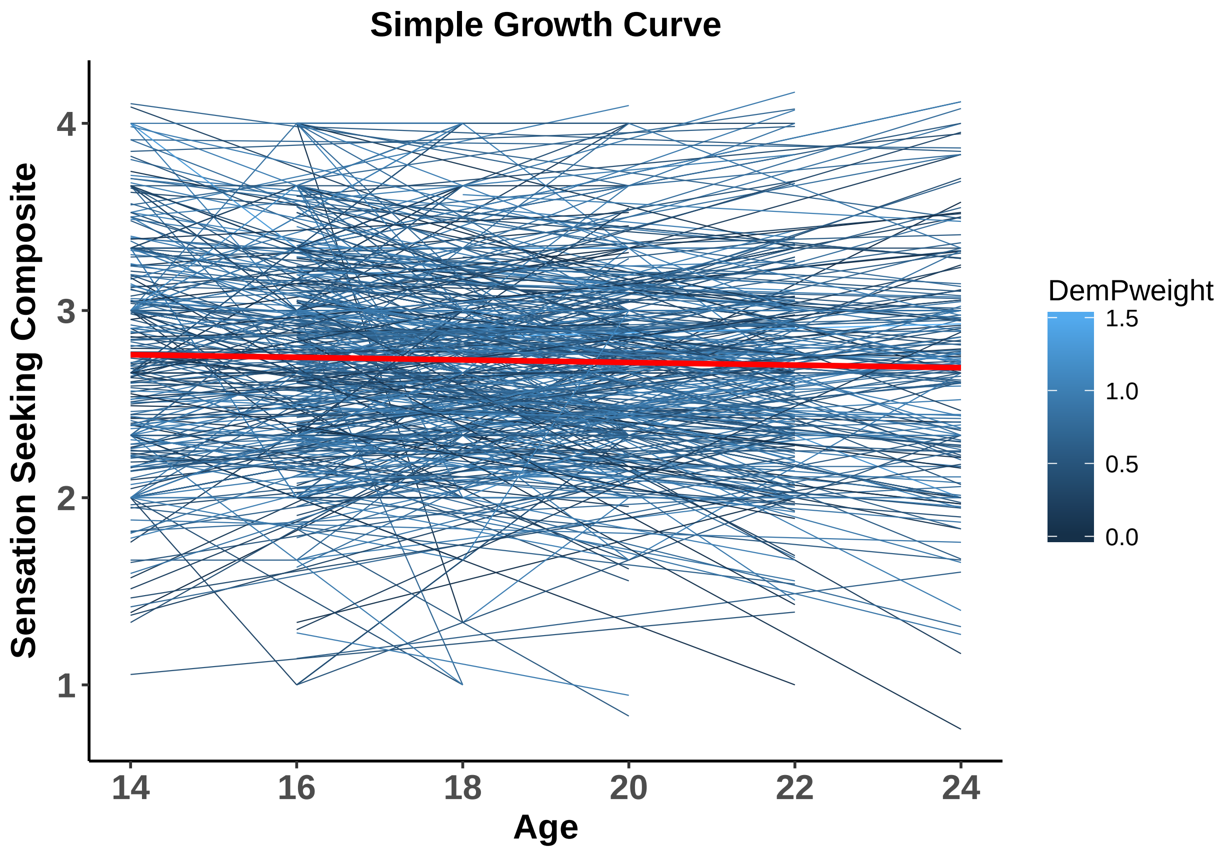
$Y_{ij} = \gamma_{00} + \gamma_{01} * X_{2j} + U_{0j} + (\gamma_{10} + \gamma_{11} * X_{2j} + U_{1j}) * X_{1j} + \varepsilon ij$
$Y_{ij} = \gamma_{00} + \gamma_{01} * X_{2j} + \gamma_{10} * X_{1j} + \gamma_{11} * X_{2j} * X_{1j} + U_{0j} + U_{1j} * X_{1j} + \varepsilon ij$

We can also fit this with intercepts depending on weight, but without the change (slope) dependent on weight: $Y_{ij} = \gamma_{00} + \gamma_{01} * X_{2j} + U_{0j} + (\gamma_{10} + U_{1j}) * X_{1j} + \varepsilon ij$ $Y_{ij} = \gamma_{00} + \gamma_{01} * X_{2j} + \gamma_{10} * X_{1j} + U_{0j} + U_{1j} * X_{1j} + \varepsilon ij$

Continuous Example - Weight for Age Percentile

```
describe(sample_dat$DemPweight)
```

```
##    vars    n mean   sd median trimmed  mad   min  max range  skew kurtosis
## X1    1 2084 0.66 0.31   0.69    0.67 0.36 -0.06 1.62  1.68 -0.29    -0.56
##      se
## X1 0.01
```




```r
# time invariant covariate with random intecept (with weight as covariate)
# and slope (without weight as a covariate)
mod1a <- lmer(SensSeek ~ age0 + DemPweight + (age0|PROC_CID),
              data = sample_dat)

summary(mod1a)

# time invariant predictor with random slope and intercept
mod1b <- lmer(SensSeek ~ age0 + DemPweight + age0*DemPweight +
                (age0|PROC_CID), data = sample_dat)

summary(mod1b)
```

Time Invariant Predictors: Categorical

Categorical Example - 2 level group

Let's start with the basic syntax:

- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * time_{1j} + \varepsilon ij$
- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + \gamma_{01} * X_{2j} + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + \gamma_{11} * X_{2j} + U_{1j}$

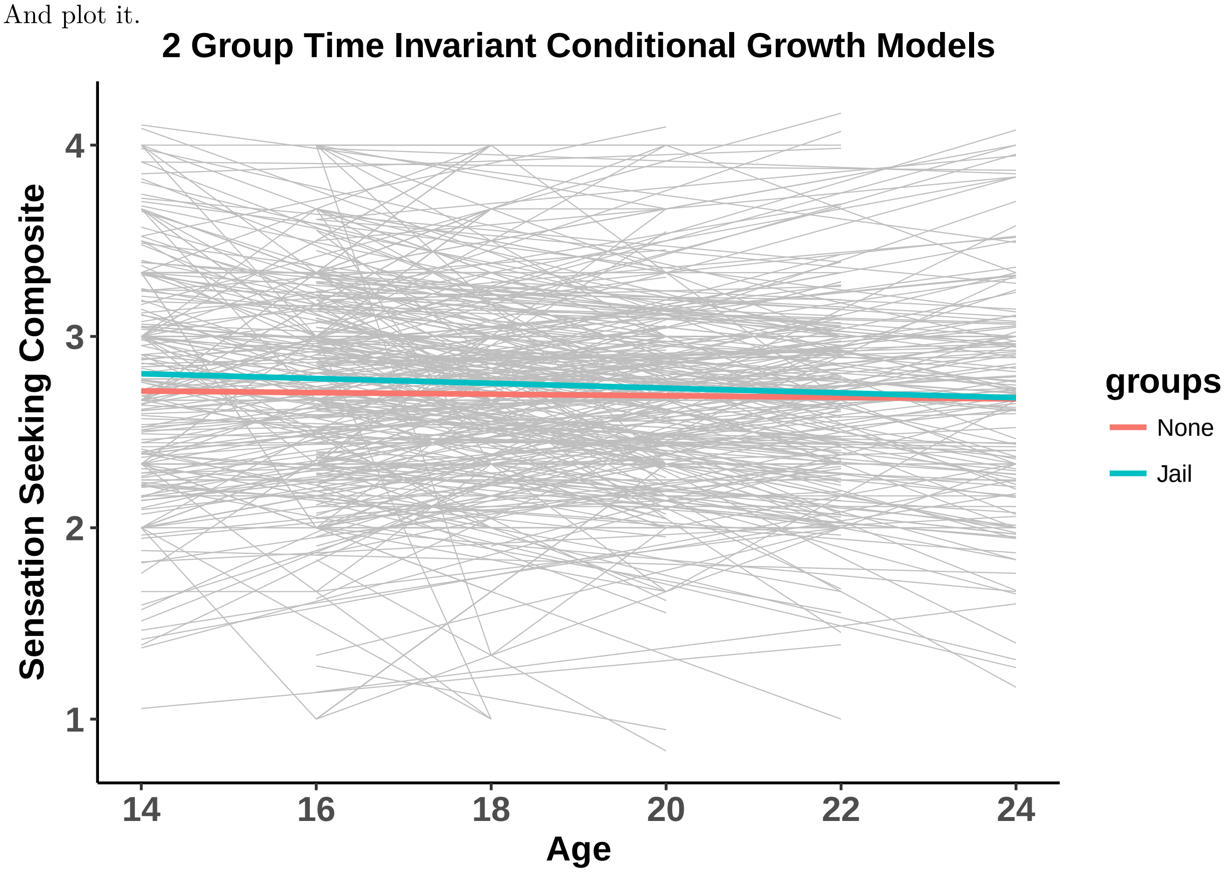
Now let's swap that out for a 2 group sample from the present data:

- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * age0_{ij} + \varepsilon ij$

- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + \gamma_{01} * groupsNone + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + \gamma_{11} * groupsNone + U_{1j}$

Variable	D1
Jail	0
None	1

And plot it.



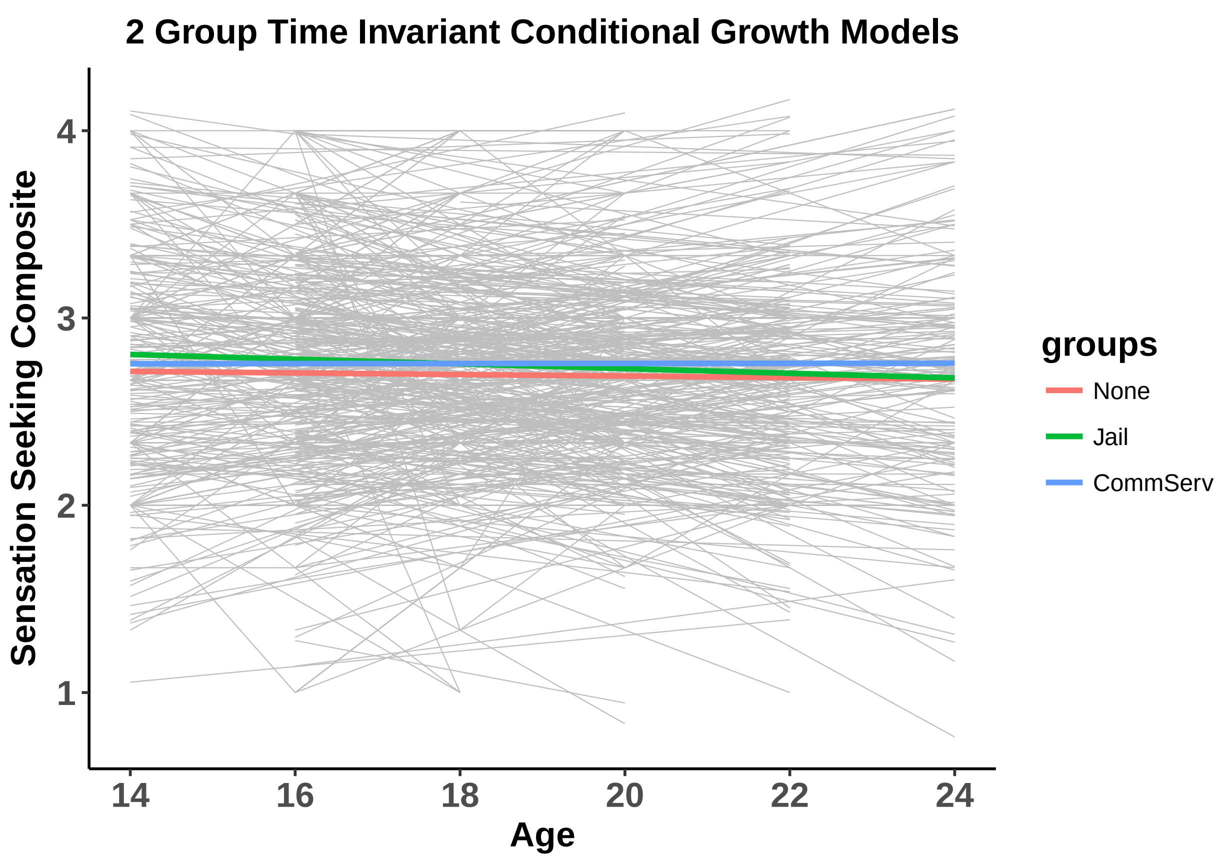
And model it:

```
mod2g <- lmer(SensSeek ~ age0 + groups + age0*groups + (age0|PROC_CID),
              data = sample_dat %>% filter(groups != "CommServ"))
summary(mod2g)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: SensSeek ~ age0 + groups + age0 * groups + (age0 | PROC_CID)
##    Data: sample_dat %>% filter(groups != "CommServ")
##
## REML criterion at convergence: 2607.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2324 -0.4860  0.0463  0.4643  3.0578
##
## Random effects:
```

```
##   Groups   Name        Variance  Std.Dev. Corr
##   PROC_CID (Intercept) 0.1613721 0.40171
##            age0        0.0008963 0.02994  -0.31
##   Residual             0.1897644 0.43562
## Number of obs: 1573, groups:  PROC_CID, 689
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)      2.717417   0.036423   74.61
## age0            -0.003998   0.006382   -0.63
## groupsJail       0.093497   0.048257    1.94
## age0:groupsJail -0.007432   0.008272   -0.90
##
## Correlation of Fixed Effects:
##             (Intr) age0   grpsJl
## age0        -0.624
## groupsJail  -0.755  0.471
## age0:grpsJl  0.481 -0.772 -0.623
```

Categorical Example - 3 level group



- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * age0_{ij} + \varepsilon ij$
- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + \gamma_{01} * D1 + \gamma_{02} * D2 + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + \gamma_{11} * D1 + \gamma_{12} * D2 + U_{1j}$

Variable	D1	D2
Jail	0	0
None	1	0
CommServ	0	1

```
mod3g <- lmer(SensSeek ~ age0 + groups + age0*groups +
                (age0|PROC_CID), data = sample_dat)
summary(mod3g)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: SensSeek ~ age0 + groups + age0 * groups + (age0 | PROC_CID)
##    Data: sample_dat
##
## REML criterion at convergence: 3418.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2994 -0.5006  0.0368  0.4533  3.0815
##
## Random effects:
##  Groups   Name        Variance  Std.Dev. Corr
##  PROC_CID (Intercept) 0.1446194 0.38029
##           age0        0.0008903 0.02984  -0.23
##  Residual             0.1888364 0.43455
## Number of obs: 2084, groups:  PROC_CID, 924
##
## Fixed effects:
##                      Estimate Std. Error t value
## (Intercept)          2.717703   0.035554   76.44
## age0                -0.004101   0.006376   -0.64
## groupsJail           0.092594   0.047097    1.97
## groupsCommServ       0.035192   0.053386    0.66
## age0:groupsJail     -0.007181   0.008265   -0.87
## age0:groupsCommServ  0.005939   0.009871    0.60
##
## Correlation of Fixed Effects:
##             (Intr) age0   grpsJl grpsCS ag0:gJ
## age0        -0.619
## groupsJail  -0.755  0.467
## gropsCmmSrv -0.666  0.412  0.503
## age0:grpsJl  0.478 -0.772 -0.618 -0.318
## ag0:grpsCmS  0.400 -0.646 -0.302 -0.612  0.498
```

Side Notes: Practical Applications

Side Note: `lme4` helper functions

```
vcov(mod2g)
VarCorr(mod2g)
fixef(mod2g)
head(ranef(mod2g)[[1]])
head(coef(mod2g)[[1]])
confint.merMod(mod2g, method = "boot")
```

```r
reghelper::ICC(mod2g)
MuMIn::r.squaredGLMM(mod2g)
```

```r
vcov(mod2g)
```

```
## 4 x 4 Matrix of class "dpoMatrix"
##                   (Intercept)          age0    groupsJail age0:groupsJail
## (Intercept)      0.0013266680 -1.449788e-04 -0.0013266680    1.449788e-04
## age0            -0.0001449788  4.072833e-05  0.0001449788   -4.072833e-05
## groupsJail      -0.0013266680  1.449788e-04  0.0023287011   -2.486054e-04
## age0:groupsJail  0.0001449788 -4.072833e-05 -0.0002486054    6.842322e-05
```

```r
VarCorr(mod2g)
```

```
##  Groups   Name        Std.Dev. Corr
##  PROC_CID (Intercept) 0.401711
##           age0        0.029938 -0.313
##  Residual             0.435620
```

```r
fixef(mod2g)
```

```
##     (Intercept)            age0      groupsJail age0:groupsJail
##     2.717416961    -0.003997618     0.093496648    -0.007431764
```

```r
head(ranef(mod2g)[[1]])
```

```
##       (Intercept)          age0
## 9102     0.2825889 -0.0036110312
## 9501     0.1582912  0.0013310875
## 9502     0.1542974 -0.0003923893
## 9503     0.1411350  0.0002529397
## 10001    0.1814453 -0.0008900323
## 12802    0.2999653 -0.0027624589
```

```r
head(coef(mod2g)[[1]])
```

```
##       (Intercept)         age0 groupsJail age0:groupsJail
## 9102     3.000006 -0.007608649 0.09349665    -0.007431764
## 9501     2.875708 -0.002666530 0.09349665    -0.007431764
## 9502     2.871714 -0.004390007 0.09349665    -0.007431764
## 9503     2.858552 -0.003744678 0.09349665    -0.007431764
## 10001    2.898862 -0.004887650 0.09349665    -0.007431764
## 12802    3.017382 -0.006760077 0.09349665    -0.007431764
```

```r
confint.merMod(mod2g, method = "boot", nsim = 10)
```

```
##                        2.5 %      97.5 %
## .sig01           0.350527987 0.434967349
## .sig02          -1.000000000 0.999999999
## .sig03           0.001946570 0.045350613
## .sigma           0.415917502 0.470609407
## (Intercept)      2.673128594 2.761488523
## age0            -0.008689551 0.004274218
## groupsJail       0.021738062 0.165404602
## age0:groupsJail -0.020663537 0.003186735
```

All units of the random effects are in standard deviation units (which means you need to square them to get the variance!!)

- .sig01 = sd of random intercept = $\sqrt{\tau_{00}}$
- .sig02 = correlation between slope and intercept = $\sqrt{\tau_{10}}$
- .sig03 = sd of random slope = $\sqrt{\tau_{11}}$

- .sigma = residual variance = $\hat{\sigma}$

```
reghelper::ICC(mod2g)
```

```
## [1] 0.4609468
```

Conditional R^2: How much variance fixed + random effects explain
Marginal R^2: how much variance the fixed effects explain

explained here

```
MuMIn::r.squaredGLMM(mod2g)
```

```
##         R2m         R2c
## 0.005234242 0.452019164
```

Side Note: Creating MLM Tables

There are lots of helpful packages for this, including `stargazer` and `sjPlot`, which are demonstrated below.

```
stargazer::stargazer(mod2g)
sjPlot::sjt.lmer(mod2g)
```

The problem is that `stargazer()` doesn't include all the terms we want, and `sjt.lmer()` only renders html. Embedded in the `.Rmd` version of these slides is some code that should help you to extract the terms you need and create a table using `dplyr` and `tidyr` that you can render in LaTeXusing `stargazer`.

But let's understand where those variables came from. To do so, we'll use the `broom` package in R to grab the terms we need.

Description	Math Notation
Fixed Effect Intercept	γ_{00}
Fixed Effect Group Intercept	γ_{01}
Fixed Effect Age Slope	γ_{10}
Fixed Effect Group Slope	γ_{11}
Individual Random Intercepts	U_{0j}
Variance of Random Intercepts	τ_{00}
Random Age Slopes	U_{10}
Variance of Random Age Slopes	τ_{11}
Correlation b/w Random Slopes and Intercepts	τ_{10}
Residual Variance	$\hat{\sigma}^2$
Intraclass Correlation	ICC
Conditional R^2	R_c^2
Marginal R^2	R_m^2

```
broom::tidy(mod2g)
broom::glance(mod2g)
```

```
##                              term     estimate   std.error  statistic
## 1                     (Intercept)  2.717416961 0.036423454 74.6062393
## 2                            age0 -0.003997618 0.006381875 -0.6264017
## 3                      groupsJail  0.093496648 0.048256617  1.9374887
## 4                 age0:groupsJail -0.007431764 0.008271833 -0.8984423
## 5        sd_(Intercept).PROC_CID  0.401711448          NA         NA
## 6                sd_age0.PROC_CID  0.029938129          NA         NA
## 7 cor_(Intercept).age0.PROC_CID -0.312526843          NA         NA
```

```
## 8       sd_Observation.Residual  0.435619527         NA         NA
##      group
## 1    fixed
## 2    fixed
## 3    fixed
## 4    fixed
## 5 PROC_CID
## 6 PROC_CID
## 7 PROC_CID
## 8 Residual
```

```
##       sigma    logLik      AIC      BIC deviance df.residual
## 1 0.4356195 -1303.786 2623.571 2666.457 2579.802        1565
```

Below is code that *should* work for all models. Just run the function and save it as an `R` object. You can use this with `papaja` and the `apa_table()` function pretty easily. The trick is that if you are not using the papaja template, the proper LaTeX packages may not be loaded. You can get around this by attaching a .tex file calling the packages under "in_header: header.tex" in your YAML header. The YAML header of this .Rmd file contains the necessary syntax and the header.tex file with the proper packages.

```r
## here's some code to make a table. You shouldn't need to modify anything here
# unless you add additional random effects terms
## fixed effects first ##
table_fun <- function(model){
  fixed <- broom::tidy(model) %>% filter(group == "fixed") %>%
    select(term, estimate)
  ## add random effects ##
  rand <- broom::tidy(model) %>% filter(group != "fixed") %>%
    select(term, estimate)
  ## get confidence intervals ##
  CI <- data.frame(confint.merMod(model, method = "boot", nsim = 10)) %>%
    mutate(term = rownames(.)) %>% setNames(c("lower", "upper", "term"))

  ## Get ICC & R2 values ##
  ICC <- reghelper::ICC(model)
  R2 <- MuMIn::r.squaredGLMM(model)

  ## format the fixed effects
  fixed <- fixed %>% left_join(CI %>% filter(!grepl(".sig", term))) %>%
    mutate(type = "Fixed Parts")

  rand <- rand %>%
    mutate(estimate = ifelse(grepl("cor", term) == T, estimate, estimate^2),
          term = mapvalues(term, unique(term),
           c("$\\tau_{00}$", "$\\tau_{11}$", "$\\tau_{10}$", "$\\hat{\\sigma^2}$"))) %>%
    left_join(
      CI %>% filter(grepl(".sig", term)) %>%
        mutate(term = mapvalues(term, unique(term),
            c("$\\tau_{00}$", "$\\tau_{10}$", "$\\tau_{11}$", "$\\hat{\\sigma^2}$")),
            lower = lower^2, upper = upper^2)) %>%
    mutate(type = "Random Parts")

  mod_terms <- tribble(
    ~term, ~estimate, ~type,
    "ICC", ICC, "Model Terms",
    "$R^2_m$", R2[1], "Model Terms",
    "$R^2_c$", R2[2], "Model Terms"
  )

  tab <- fixed %>%
```

```r
    full_join(rand) %>%
    mutate(CI = sprintf("(%.2f, %.2f)", lower, upper)) %>%
    select(-lower, -upper) %>%
    full_join(mod_terms) %>%
    mutate(estimate = sprintf("%.2f", estimate)) %>%
    dplyr::rename(b = estimate) %>%
    select(type, everything())
  return(tab)
}
# you can use this with papaja and the apa_table function pretty easily
# the trick is that if you are not using the papaja template, the proper
# LaTeX packages may not be loaded. You can get around this by attaching
# a .tex file calling the packages under "in_header: header.tex" in your YAML
# header the YAML header of this .Rmd file contains the necessary syntax and
# the header.tex file with the proper packages

tab <- table_fun(mod2g)
```

Basic: kable()

```r
options(knitr.kable.NA = '')
knitr::kable(tab, caption = "Ugly MLM Table Example")
```

Table 6: Ugly MLM Table Example

type	term	b	CI
Fixed Parts	(Intercept)	2.72	(2.67, 2.74)
Fixed Parts	age0	-0.00	(-0.01, 0.01)
Fixed Parts	groupsJail	0.09	(0.03, 0.17)
Fixed Parts	age0:groupsJail	-0.01	(-0.03, 0.01)
Random Parts	τ_{00}	0.16	(0.14, 0.17)
Random Parts	τ_{11}	0.00	(0.00, 0.00)
Random Parts	τ_{10}	-0.31	(0.23, 1.00)
Random Parts	$\hat{\sigma}^2$	0.19	(0.18, 0.22)
Model Terms	ICC	0.46	
Model Terms	R^2_m	0.01	
Model Terms	R^2_c	0.45	

More Advanced: kable() + kableExtra

```r
library(kableExtra)
options(knitr.kable.NA = '')
knitr::kable(tab %>% #select(-type) %>%
    mutate(term = gsub("[()]", "", term)),
             caption = "Not Quite Right kableExtra MLM Table Example",
    format = "latex",
    #longtable = T,
    booktabs = T, escape = F) %>%
  # group_rows("Fixed", 1,4) %>%
  # group_rows("Random", 5,9) %>%
  # group_rows("Model", 9,11) %>%
  collapse_rows(1) %>%
  #kable_styling(latex_options = c("striped","repeat_header"),full_width = F)
  add_header_above(c(" ", " ", "Model 1" = 2))
```

Table 7: Not Quite Right kableExtra MLM Table Example

		Model 1	
type	term	b	CI
Fixed Parts	Intercept	2.72	(2.67, 2.74)
	age0	-0.00	(-0.01, 0.01)
	groupsJail	0.09	(0.03, 0.17)
	age0:groupsJail	-0.01	(-0.03, 0.01)
Random Parts	τ_{00}	0.16	(0.14, 0.17)
	τ_{11}	0.00	(0.00, 0.00)
	τ_{10}	-0.31	(0.23, 1.00)
	$\hat{\sigma}^2$	0.19	(0.18, 0.22)
	ICC	0.46	
Model Terms	R^2_m	0.01	
Model Terms	R^2_c	0.45	

Alternative: `papaja` + `apa_table()`

```
papaja::apa_table(tab %>% select(-type),caption = "papaja MLM Table Example",
    na_string = "", stub_indents = list(Fixed = c(1:4), Random = c(5:8), Summary = c(9:11)),
    col_spanners = list(`Sensation Seeking` = c(2,3)))
```

The Case of Multiple Models

```
# basically you can run this multiple times and then join the data frames together.
# The only trick is that you'll have to change the column names to make sure they are distinct.
# My favorite way of doing this is with the purrr package in R.
# So basically, you make a dataframe of your models, which looks like this:

mod.df <- tibble(
  outcome = c("Model 2g", "Model 3g", "Model Cont"), # just plug in whatever name for your outcome
  mod = c(mod2g, mod3g, mod1b) # basically just listing models that correspond to your outcomes
)

# then you use purrr

mod.df.long <- mod.df %>%
  mutate(tab = map(mod, table_fun)) %>%
  unnest(tab, .drop = T)

table.df <- mod.df.long %>%
  gather(key = param, value = value, b, CI) %>%
  unite(param, outcome, param, sep = ".") %>%
  spread(key = param, value = value) %>%
  mutate(type = factor(type, levels = c("Fixed Parts", "Random Parts", "Model Terms"))) %>%
  arrange(type)
```

Table 8: papaja MLM Table Example

	Sensation Seeking	
term	b	CI
Fixed		
(Intercept)	2.72	(2.67, 2.74)
age0	-0.00	(-0.01, 0.01)
groupsJail	0.09	(0.03, 0.17)
age0:groupsJail	-0.01	(-0.03, 0.01)
Random		
τ_{00}	0.16	(0.14, 0.17)
τ_{11}	0.00	(0.00, 0.00)
τ_{10}	-0.31	(0.23, 1.00)
$\hat{\sigma^2}$	0.19	(0.18, 0.22)
Summary		
ICC	0.46	
R^2_m	0.01	
R^2_c	0.45	

```r
# then you can use apa_table or kable + kableExtra to make them pretty
options(papaja.na_string = " ")
papaja::apa_table(table.df %>% select(-type), caption = "Multi-Model Table",
          col_spanners = list(`2 group` = c(2,3), `3 group` = c(4,5), `Continuous` = c(6,7)),
          stub_indents = list(Fixed = seq(1,8), Random = seq(9,12), `Model Terms` = seq(13,15)),
          na_string = " ",
          col.names = c(" ", rep(c("b", "CI"), times = 3)),
          align = c("l", rep("c", 6)))
```

Side Note: Plotting

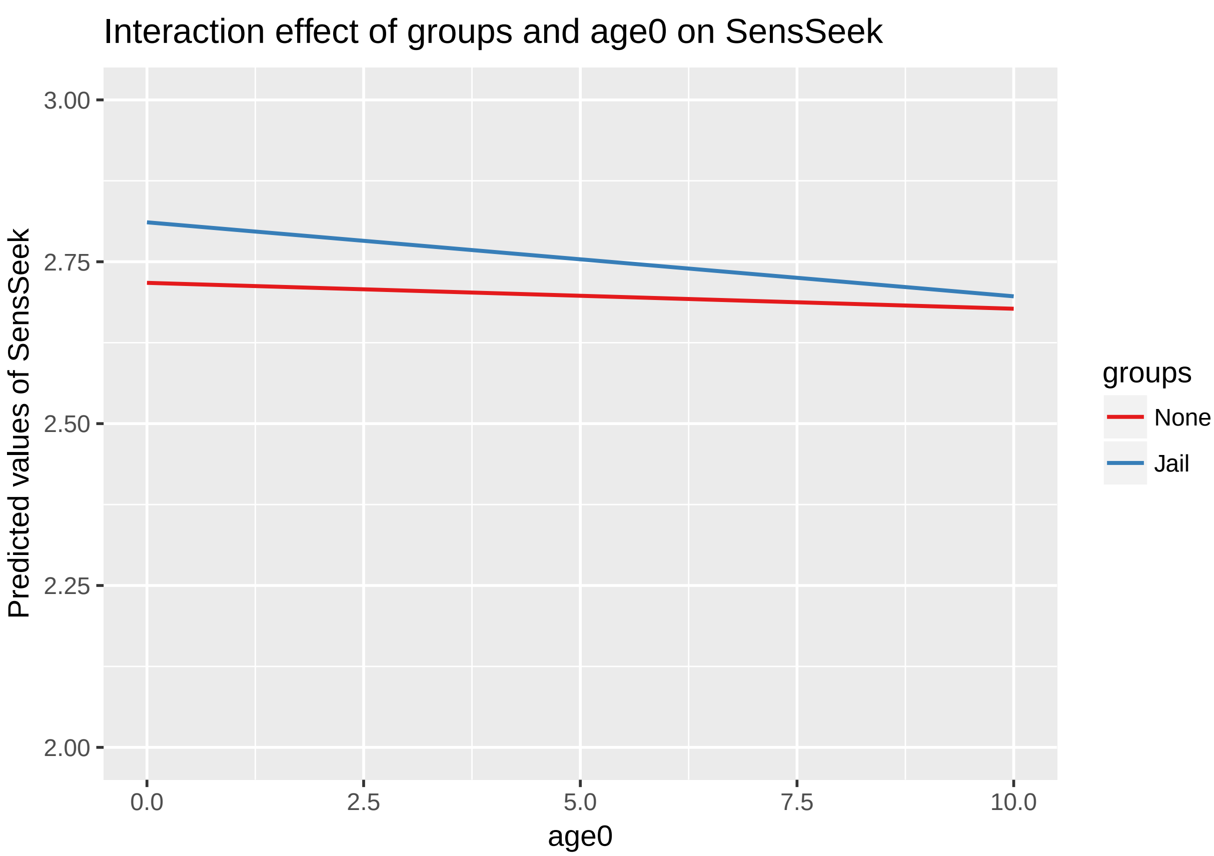
Lazy Method: `sjPlot` + `sjt.int()`

Categorical

```r
sjp.int(mod2g, type = "eff", p.kr = F, swap.pred = T)
```

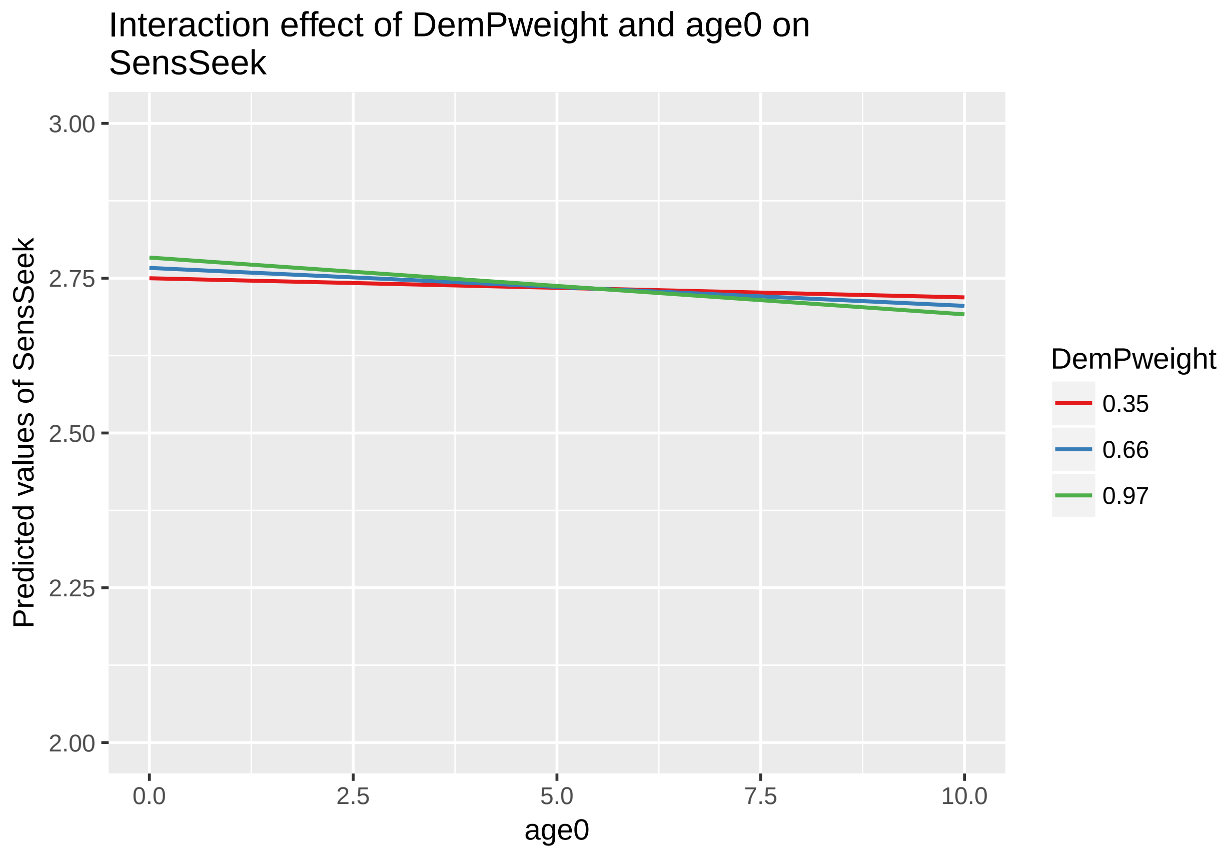
Table 9: Multi-Model Table

	2 group		3 group		Continuous	
	b	CI	b	CI	b	CI
Fixed						
(Intercept)	2.72	(2.68, 2.74)	2.72	(2.66, 2.76)	2.73	(2.67, 2.82)
age0	-0.00	(-0.01, 0.01)	-0.00	(-0.01, 0.00)	0.00	(-0.02, 0.01)
age0:DemPweight					-0.01	(-0.02, 0.01)
age0:groupsCommServ			0.01	(-0.00, 0.03)		
age0:groupsJail	-0.01	(-0.02, 0.00)	-0.01	(-0.02, -0.00)		
DemPweight					0.05	(-0.06, 0.13)
groupsCommServ			0.04	(-0.07, 0.09)		
groupsJail	0.09	(0.02, 0.10)	0.09	(0.03, 0.16)		
Random						
$\hat{\sigma^2}$	0.19	(0.18, 0.20)	0.19	(0.17, 0.20)	0.19	(0.17, 0.20)
τ_{00}	0.16	(0.14, 0.19)	0.14	(0.10, 0.18)	0.14	(0.12, 0.16)
τ_{10}	-0.31	(1.00, 0.00)	-0.23	(0.12, 0.15)	-0.22	(0.19, 0.06)
τ_{11}	0.00	(0.00, 0.00)	0.00	(0.00, 0.00)	0.00	(0.00, 0.00)
Model Terms						
R_c^2	0.45		0.44		0.44	
R_m^2	0.01		0.00		0.00	
ICC	0.46		0.44		0.43	



Continuous

```r
sjp.int(mod1b, type = "eff", p.kr = F, swap.pred = T, mdrt.values = "meansd")
```

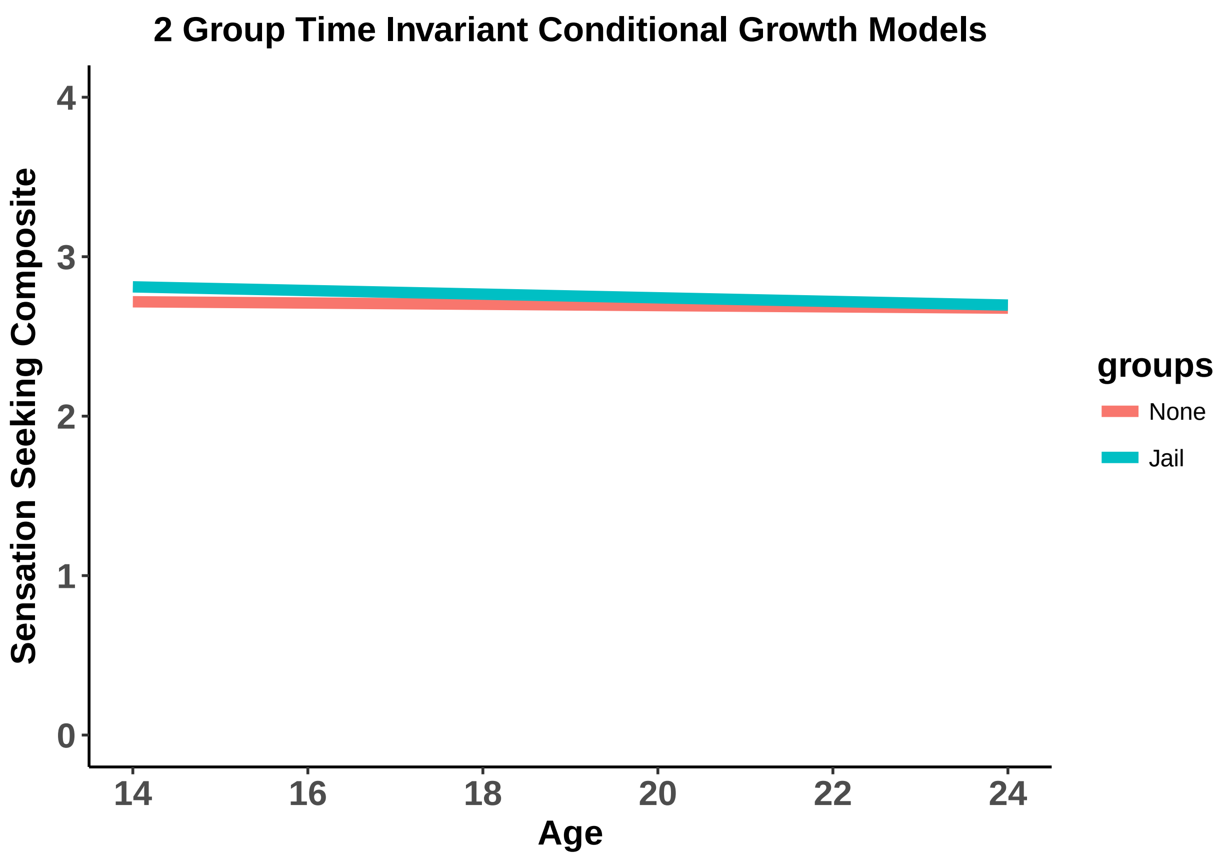


Medium Advanced: Using the `predict()` function

Categorical

```r
fixed.frame <-
  data.frame(expand.grid(age0 = seq(0,10,2),
             groups = c("None","Jail"))) %>%
  mutate(pred = predict(mod2g, newdata = ., re.form = NA))

fixed.frame %>%
  mutate(age = age0 + 14) %>%
  ggplot(aes(x = age, y = pred, color = groups)) +
    geom_line(size = 2) +
    lims(y = c(0,4)) +
    labs(x = "Age", y = "Sensation Seeking Composite",
         title = "2 Group Time Invariant Conditional Growth Models") +
    theme_classic() +
    theme(axis.text = element_text(face = "bold", size = rel(1.2)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```

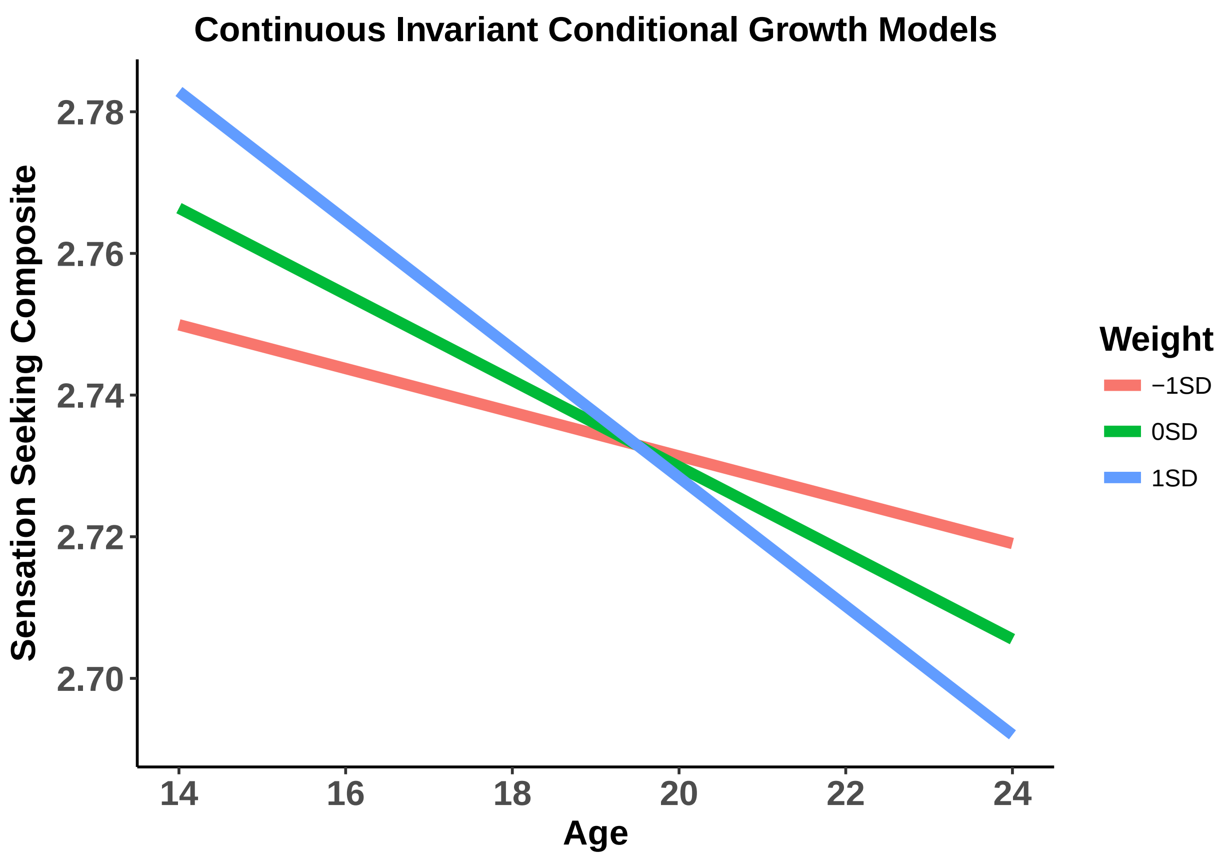


Continuous

```
fixed.frame <- sample_dat %>%
  summarise(mean = mean(DemPweight, na.rm = T),
            sd = sd(DemPweight, na.rm = T))

fixed.frame <-
  data.frame(
    expand.grid(
      # here, you add values for your time variable and predictors
      age0 = seq(0,10,2),
      DemPweight = c(fixed.frame$mean-fixed.frame$sd,
                     fixed.frame$mean,
                     fixed.frame$mean+fixed.frame$sd))) %>%
  mutate(pred = predict(mod1b, newdata = ., re.form = NA))


fixed.frame %>%
  mutate(Weight = factor(DemPweight, levels = unique(DemPweight), labels = c("-1SD", "0SD", "1SD")),
         age = age0 + 14) %>%
  ggplot(aes(x = age, y = pred, color = Weight)) +
    geom_line(size = 2) +
    labs(x = "Age", y = "Sensation Seeking Composite",
         title = "Continuous Invariant Conditional Growth Models") +
    theme_classic() +
    theme(axis.text = element_text(face = "bold", size = rel(1.2)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



More advanced: `expand.grid()` + `fixef()` + `ggplot2()`

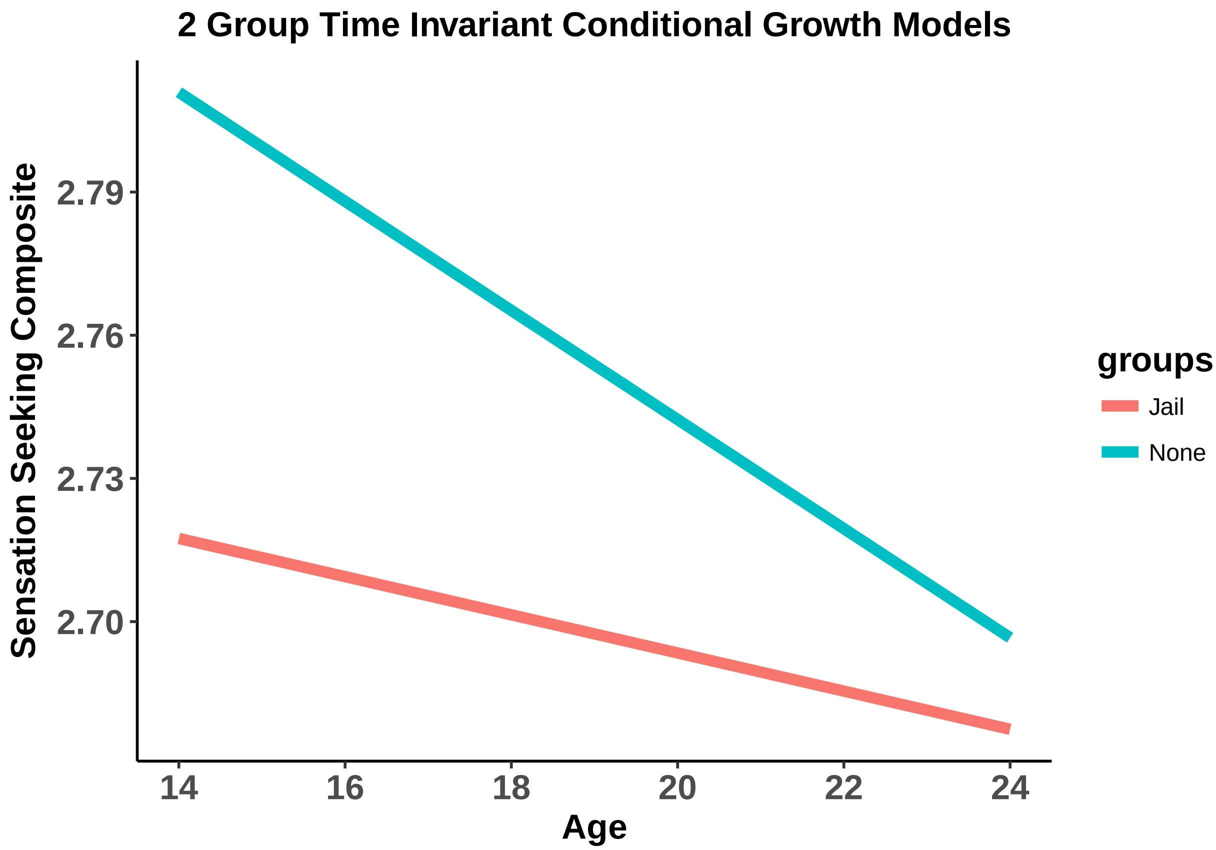
Categorical

```
# example for categorical
fixed.frame <-
  data.frame(
    expand.grid(
      # here, you add values for your time variable and predictors
      Intercept = 1,
      age0 = seq(0,10,2),
      groupsNone = c(0,1))) %>%
  # now take care of interactions and add an intercept
  mutate(`age0:groupsNone` = age0*groupsNone)

# multiplying to get values for model frame
fixed.frame$pred <- as.vector(as.matrix(fixed.frame) %*% fixef(mod2g))

fixed.frame %>%
  mutate(groups = factor(groupsNone, levels = c(0,1), labels = c("Jail", "None")),
         age = age0 + 14) %>%
  ggplot(aes(x = age, y = pred, color = groups)) +
    geom_line(size = 2) +
    labs(x = "Age", y = "Sensation Seeking Composite",
         title = "2 Group Time Invariant Conditional Growth Models") +
    theme_classic() +
    theme(axis.text = element_text(face = "bold", size = rel(1.2)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
```

```
        plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



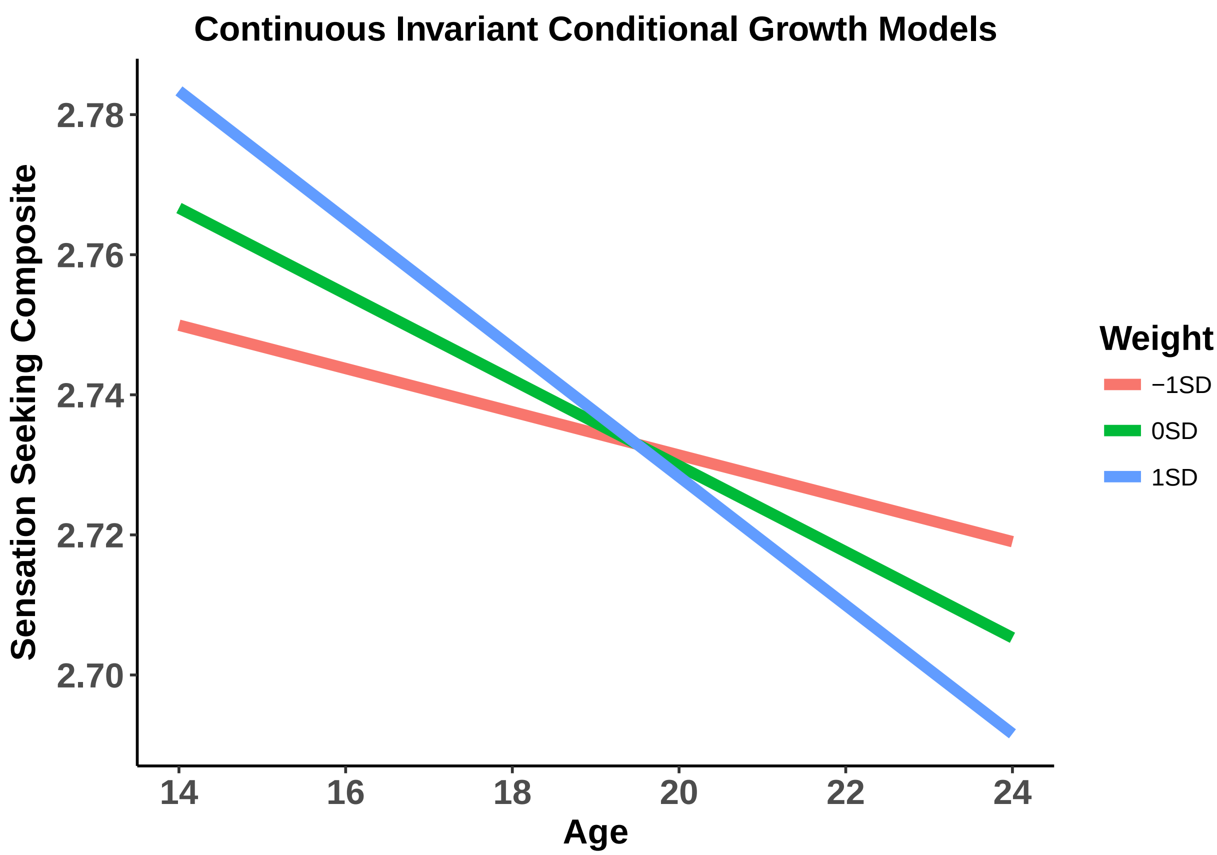
Continuous

```
detach(package:plyr)
library(dplyr)
# example for continuous
fixed.frame <- sample_dat %>%
  group_by(PROC_CID) %>%
  summarise(DemPweight = mean(DemPweight, na.rm = T)) %>%
  ungroup() %>%
  summarise(mean = mean(DemPweight, na.rm = T),
            sd = sd(DemPweight, na.rm = T))

fixed.frame <-
  data.frame(
    expand.grid(
      # here, you add values for your time variable and predictors
      age0 = seq(0,10,2),
      DemPweight = c(fixed.frame$mean-fixed.frame$sd,
                     fixed.frame$mean,
                     fixed.frame$mean+fixed.frame$sd))) %>%
  # now take care of interactions and add an intercept
  mutate(`age0:DemPweight` = age0*DemPweight,
         Intercept = 1) %>%
  # reordering everything
  select(Intercept, everything())
```

```r
# multiplying to get values for model frame
fixed.frame$value <- as.vector(as.matrix(fixed.frame) %*% fixef(mod1b))

fixed.frame %>%
  mutate(Weight = factor(DemPweight, levels = unique(DemPweight), labels = c("-1SD", "0SD", "1SD")),
         age = age0 + 14) %>%
  ggplot(aes(x = age, y = value, color = Weight)) +
    geom_line(size = 2) +
    labs(x = "Age", y = "Sensation Seeking Composite",
         title = "Continuous Invariant Conditional Growth Models") +
    theme_classic() +
    theme(axis.text = element_text(face = "bold", size = rel(1.2)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```

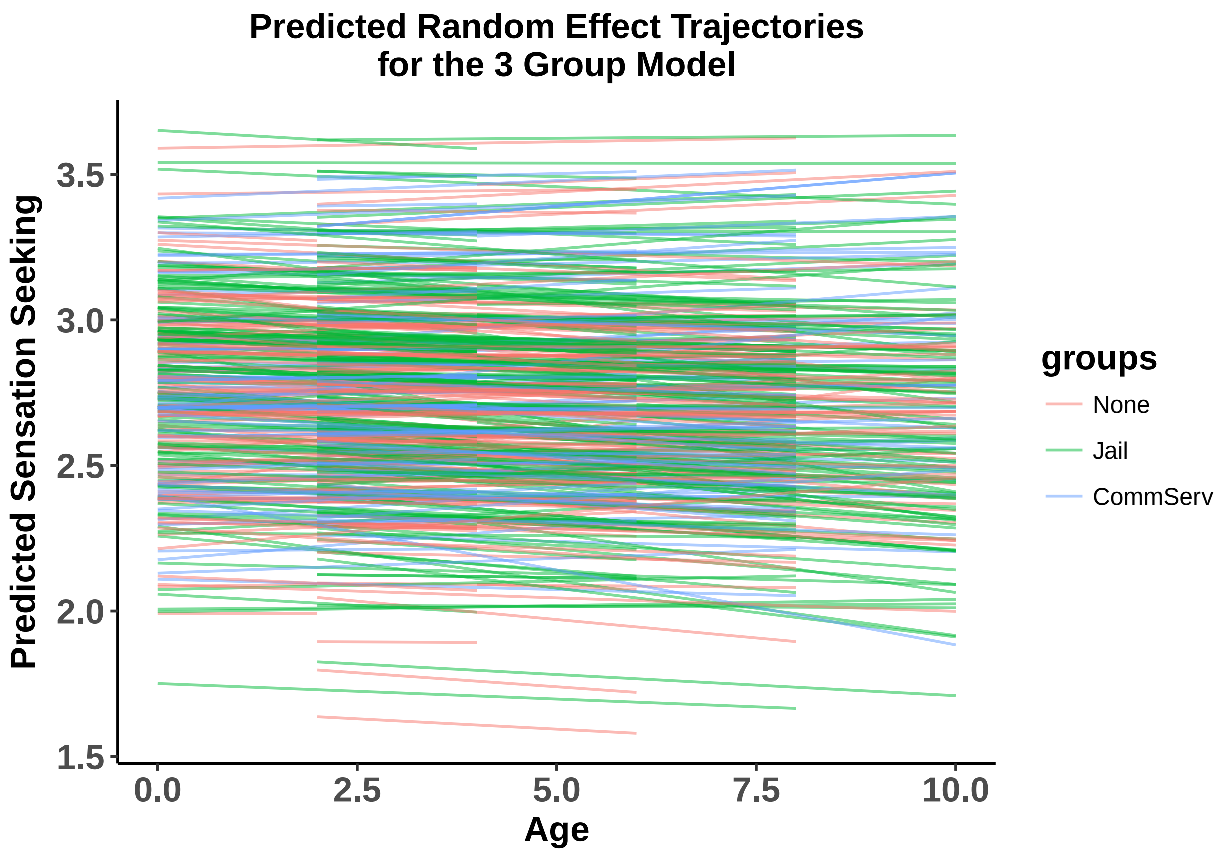


Random Effects

Categorical

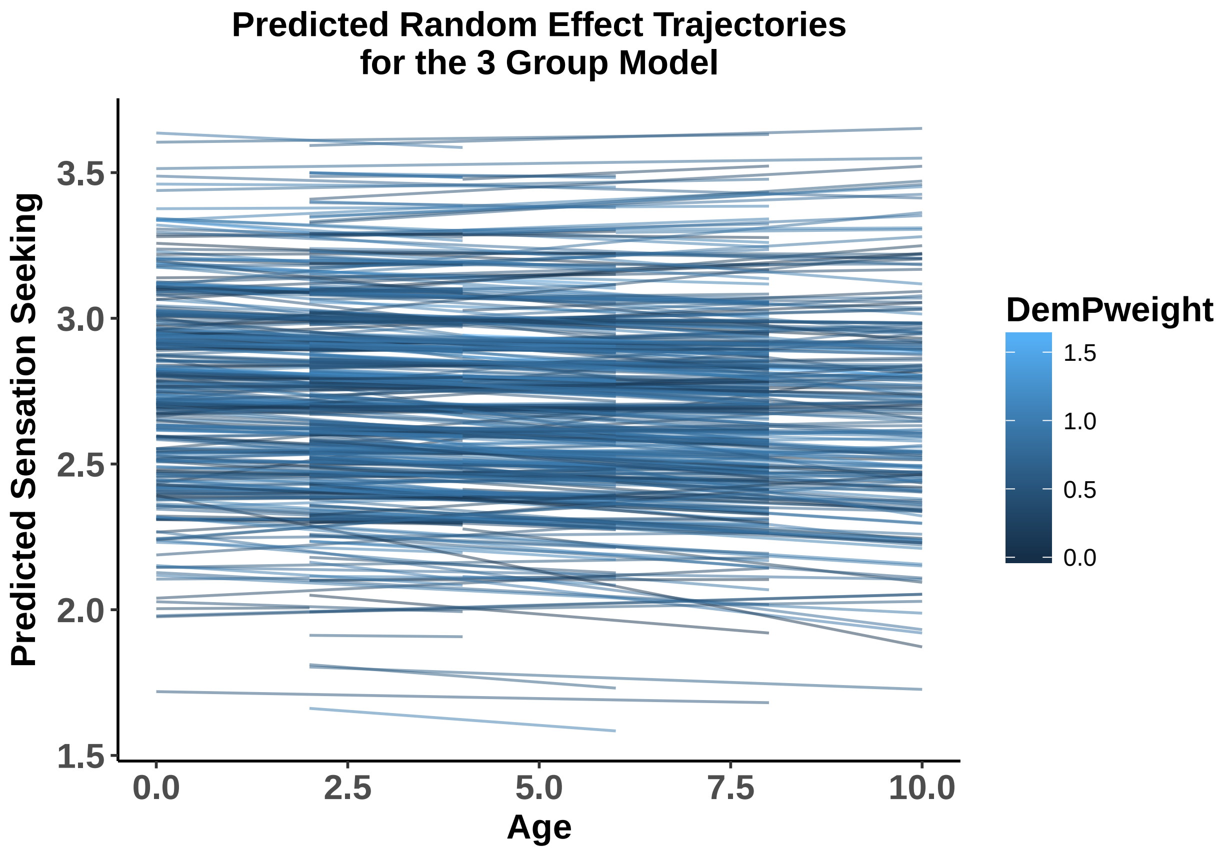
```r
tbl_df(mod3g@frame) %>% select(PROC_CID, age0, groups) %>%
  mutate(pred = predict(mod3g, newdata = .)) %>%
  ggplot(aes(x = age0, y = pred, group = PROC_CID, color = groups)) +
    geom_line(size = .5, alpha = .5) +
    labs(x = "Age", y = "Predicted Sensation Seeking",
         title = "Predicted Random Effect Trajectories\nfor the 3 Group Model") +
    theme_classic() +
```

```r
  theme(axis.text = element_text(face = "bold", size = rel(1.2)),
        axis.title = element_text(face = "bold", size = rel(1.2)),
        legend.title = element_text(face = "bold", size = rel(1.2)),
        plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



Categorical

```r
mod1b@frame %>% select(-SensSeek) %>%
  mutate(pred = predict(mod1b)) %>%
  ggplot(aes(x = age0, y = pred, group = PROC_CID, color = DemPweight)) +
    geom_line(size = .5, alpha = .5) +
    labs(x = "Age", y = "Predicted Sensation Seeking",
         title = "Predicted Random Effect Trajectories\nfor the 3 Group Model") +
    theme_classic() +
    theme(axis.text = element_text(face = "bold", size = rel(1.2)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



Side Note: Comparisons with `lsmeans`

The `lsmeans` package has a lot of useful functions. They are listed below. Then I'll demonstrate them in turn.

```r
# create a reference grid
ref.grid2g <- ref.grid(mod2g)
# create the lsmeans object
lsgroups   <- lsmeans(ref.grid2g, "groups")
# compact letter display
cld(lsgroups, alpha = .10)
# plot
plot(lsgroups)
# contrasts of the ref.grid object
contrast(ref.grid2g, method = "eff")
# comparisons
groups.sum <- summary(lsgroups, infer = c(TRUE,TRUE),
                      level = .90, adjust = "bon", by = "groups")
```

```r
# create a reference grid
(ref.grid2g <- ref.grid(mod2g))
```

```
## 'ref.grid' object with variables:
##     age0 = 3.9123
##     groups = None, Jail
```

```r
# create the lsmeans object
(lsgroups   <- lsmeans(ref.grid2g, "groups"))
```

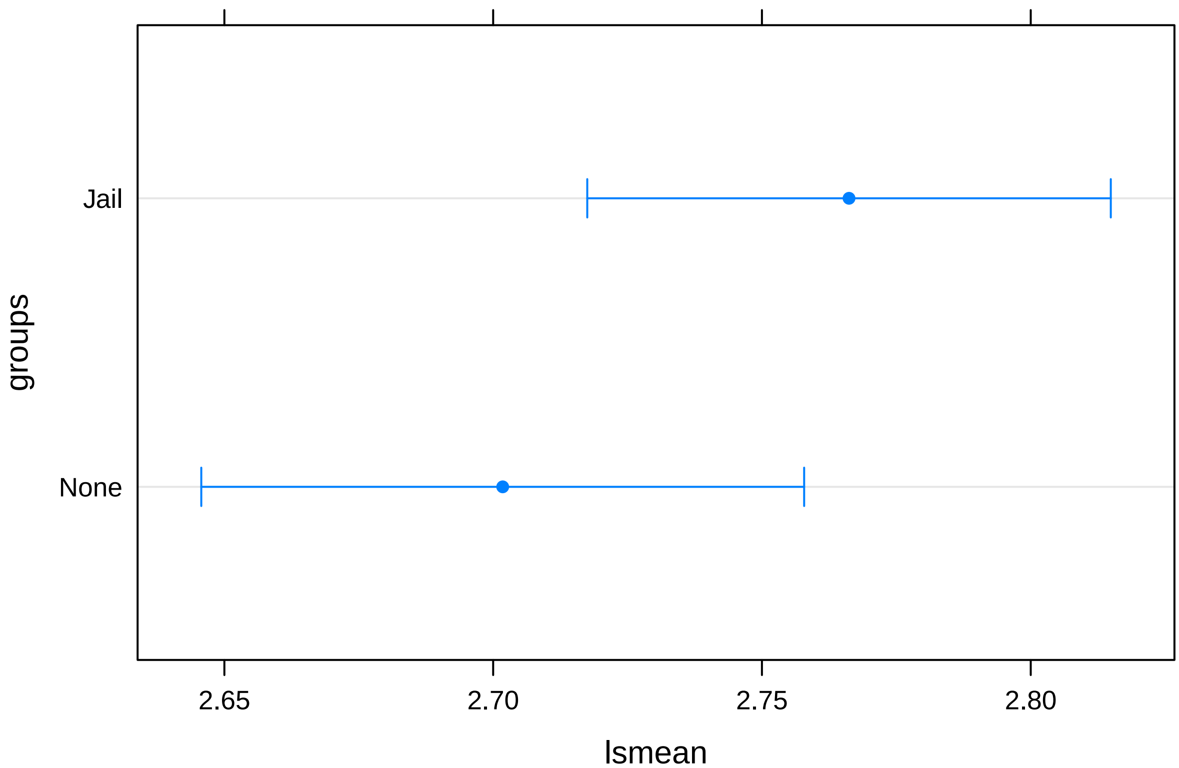
```
##  groups   lsmean         SE     df lower.CL upper.CL
##  None   2.701777 0.02855972 701.42 2.645704 2.757850
```

```
##   Jail   2.766199 0.02480113 676.80 2.717505 2.814892
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
```

```
# compact letter display
cld(lsgroups, alpha = .10)
```

```
##   groups   lsmean         SE     df lower.CL upper.CL .group
##   None   2.701777 0.02855972 701.42 2.645704 2.757850  1
##   Jail   2.766199 0.02480113 676.80 2.717505 2.814892   2
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## significance level used: alpha = 0.1
```

```
# plot
plot(lsgroups)
```




```
# contrasts of the ref.grid object
contrast(ref.grid2g, method = "eff")
```

```
##   contrast                          estimate         SE     df  t.ratio
##   3.91226954863318,None effect -0.03221079 0.01891265 690.74  -1.703
##   3.91226954863318,Jail effect  0.03221079 0.01891265 690.74   1.703
##   p.value
##    0.0890
##    0.0890
##
## P value adjustment: fdr method for 2 tests
```

```
# comparisons
(groups.sum <- summary(lsgroups, infer = c(TRUE,TRUE),
          level = .90, adjust = "bon", by = "groups"))
```

```
## groups = None:
##    lsmean         SE     df lower.CL upper.CL t.ratio p.value
##  2.701777 0.02855972 701.42 2.654739 2.748816  94.601  <.0001
##
## groups = Jail:
##    lsmean         SE     df lower.CL upper.CL t.ratio p.value
##  2.766199 0.02480113 676.80 2.725351 2.807047 111.535  <.0001
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.9
```

Time Varying Predictors

Time Varying Predictors: Continuous

Next, we'll add in a time-varying predictor. Maybe it's not that our participants sensation seeking is moderated by early life experiences of jail or court-ordered community service. Instead, their sensation seeking is moderated by depression.
How does this look?

- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * time + \beta_{2j} * CESD + \varepsilon ij$
- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + U_{1j}$
  - $\beta_{2j} = \gamma_{20}$

To Interaction or Not - That Is the Question

- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * age0 + \beta_{2j} * CESD + \varepsilon ij$
- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + U_{1j}$
  - $\beta_{2j} = \gamma_{20}$

$$Y_{ij} = \gamma_{00} + \gamma_{01} + U_{0j} + (\gamma_{10} + U_{1j}) * age0 + \gamma_{20} * CESD$$

Example: Does depression influence changes in sensation seeking over time?

```r
modTV1 <- lmer(SensSeek ~ age0 + CESD + (age0|PROC_CID), data = sample_dat)
```

```r
summary(modTV1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: SensSeek ~ age0 + CESD + (age0 | PROC_CID)
##    Data: sample_dat
##
## REML criterion at convergence: 3391.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.4390 -0.5035  0.0363  0.4423  3.1508
##
```

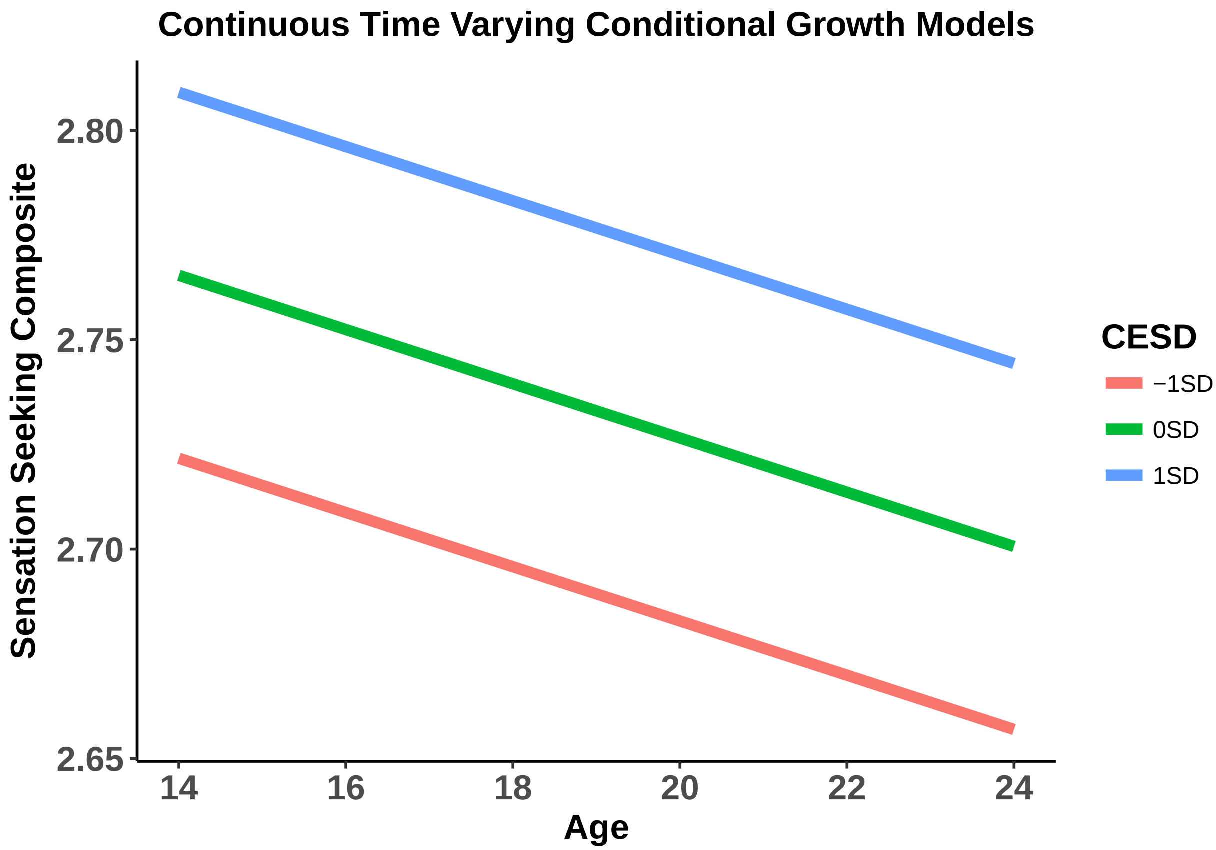
```
## Random effects:
##  Groups   Name        Variance  Std.Dev. Corr
##  PROC_CID (Intercept) 0.1412389 0.37582
##           age0        0.0008117 0.02849  -0.21
##  Residual             0.1892164 0.43499
## Number of obs: 2084, groups:  PROC_CID, 924
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  2.710845   0.025121  107.91
## age0        -0.006475   0.003553   -1.82
## CESD         0.078617   0.021519    3.65
##
## Correlation of Fixed Effects:
##      (Intr) age0
## age0 -0.467
## CESD -0.604 -0.036
```

```r
# example for continuous
# note MEANS ARE AT AGE0 = 0
fixed.frame <- sample_dat %>%
  filter(age0 == 0) %>%
  summarise(mean = mean(CESD, na.rm = T),
            sd = sd(CESD, na.rm = T))

fixed.frame <-
  data.frame(
    expand.grid(
      # here, you add values for your time variable and predictors
      age0 = seq(0,10,2),
      CESD = c(fixed.frame$mean-fixed.frame$sd,
               fixed.frame$mean,
               fixed.frame$mean+fixed.frame$sd))) %>%
  # now take care of interactions and add an intercept
  mutate(Intercept = 1) %>%
  # reordering everything
  select(Intercept, everything())

# multiplying to get values for model frame
fixed.frame$value <- as.matrix(fixed.frame) %*% as.vector(fixef(modTV1))

fixed.frame %>%
  mutate(CESD = factor(CESD, levels = unique(CESD), labels = c("-1SD", "0SD", "1SD")),
         age = age0 + 14) %>%
  ggplot(aes(x = age, y = value, color = CESD)) +
    geom_line(size = 2) +
    labs(x = "Age", y = "Sensation Seeking Composite",
         title = "Continuous Time Varying Conditional Growth Models") +
    theme_classic() +
    theme(axis.text = element_text(face = "bold", size = rel(1.2)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



Time Varying Predictors: Categorical

Next, we'll add in a time-varying predictor. Maybe it's not that our participants sensation seeking is moderated by early life experiences of jail or court-ordered community service. Instead, their sensation seeking is moderated by depression.
How does this look?

- **Level 1:** $Y_{ij} = \beta_{0j} + \beta_{1j} * time + \beta_{2j} * depressed + \varepsilon ij$
- **Level 2:**
  - $\beta_{0j} = \gamma_{00} + \gamma_{01} + U_{0j}$
  - $\beta_{1j} = \gamma_{10} + U_{1j}$
  - $\beta_{2j} = \gamma_{20}$

```
# creating a dummy variable for time varying categorical depression
sample_dat <- sample_dat %>%
  mutate(depressed =
           factor(ifelse(CESD <= 1.5, 0, 1), levels = c(0,1),
                  labels = c("Depressed", "Not Depressed")))
modTV2 <- lmer(SensSeek ~ age0 + depressed + (age0|PROC_CID),
               data = sample_dat)
summary(modTV2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: SensSeek ~ age0 + depressed + (age0 | PROC_CID)
##    Data: sample_dat
##
```

```
## REML criterion at convergence: 3401
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.3686 -0.5094  0.0363  0.4522  3.1406
##
## Random effects:
##  Groups   Name        Variance  Std.Dev. Corr
##  PROC_CID (Intercept) 0.1427349 0.37780
##           age0        0.0008415 0.02901  -0.21
##  Residual             0.1895332 0.43535
## Number of obs: 2084, groups:  PROC_CID, 924
##
## Fixed effects:
##                         Estimate Std. Error t value
## (Intercept)             2.760189   0.020388  135.38
## age0                   -0.006154   0.003564   -1.73
## depressedNot Depressed  0.068617   0.039992    1.72
##
## Correlation of Fixed Effects:
##             (Intr) age0
## age0        -0.599
## dprssdNtDpr -0.174 -0.024
```