```r
rm(list = ls())
setwd("~/Dropbox/Classes/Longitudinal Data Analysis")
wgt <- read.table("weightslong.csv", sep = ",", header = TRUE)
dems <- read.table("agegender.csv", sep = ",", header = TRUE)
plan <- read.table("plandiet.csv", sep = ",", header = TRUE)
dems$age.c <- dems$age - 49.094
data <- merge(wgt, dems, by = "ID")
data <- dplyr::left_join(data, plan, by = c("ID", "wave"))
data$gender[data$gender == 1] <- 0
data$gender[data$gender == 2] <- 1
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 3.3.2
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.3.2
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(plyr)
```

```
## ------------------------------------------------------------------------------
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## ------------------------------------------------------------------------------
##
## Attaching package: 'plyr'
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
## The following object is masked from 'package:plyr':
##
##     here
## The following object is masked from 'package:base':
##
##     date
```

```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

```
library(lme4)
```

```
## Loading required package:  Matrix
##
## Attaching package:  'Matrix'
## The following object is masked from 'package:tidyr':
##
##    expand
```

```
library(reghelper)
```

```
## Warning:  package 'reghelper' was built under R version 3.3.2
##
## Attaching package:  'reghelper'
## The following object is masked from 'package:base':
##
##    beta
```

```
library(MuMIn)
```

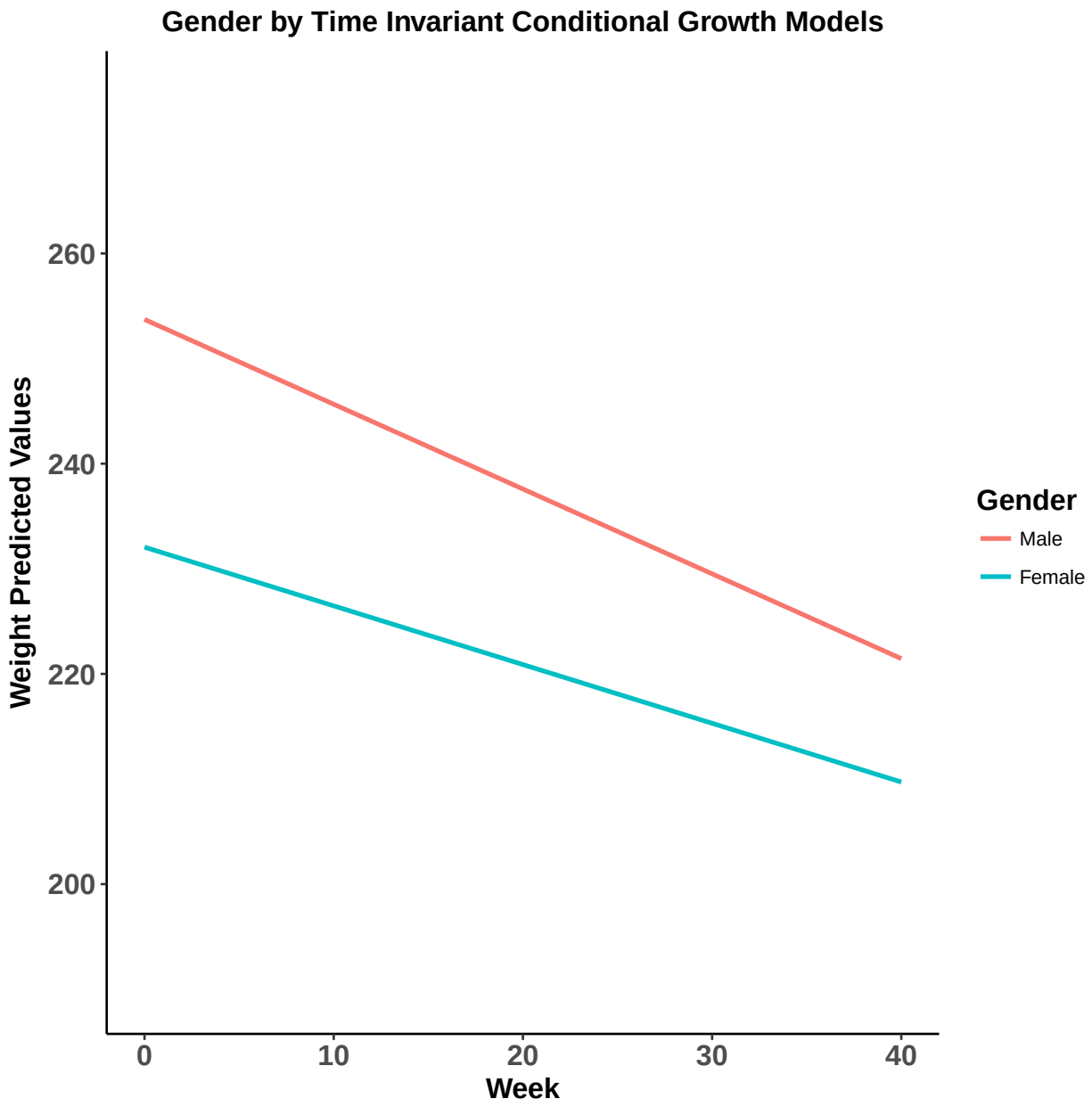
```
## Warning:  package 'MuMIn' was built under R version 3.3.2
```

1 Graphing with Categorical Level 2 Predictors

```
# model with gender as a Level 2 predictor
mod1b <- lmer(weight ~ wave + gender + gender * wave + (wave | ID), data = data)
summary(mod1b)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: weight ~ wave + gender + gender * wave + (wave | ID)
##    Data: data
##
## REML criterion at convergence: 23120.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.6273 -0.5696 -0.0630  0.4844  5.2610
##
## Random effects:
##  Groups   Name        Variance  Std.Dev. Corr
##  ID       (Intercept) 2512.2950 50.1228
##           wave           0.1655  0.4068  -0.20
##  Residual               14.4695  3.8039
## Number of obs: 3900, groups:  ID, 139
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) 253.7208    12.9461  19.598
## wave         -0.8067     0.1064  -7.583
## gender      -21.6634    13.7068  -1.580
```

```
## wave:gender   0.2481     0.1130   2.195
##
## Correlation of Fixed Effects:
##             (Intr) wave   gender
## wave        -0.201
## gender      -0.944  0.190
## wave:gender  0.189 -0.941 -0.200

# plotting categorical predictors using the predict function
fixed.frame <- data.frame(expand.grid(wave = seq(0, 40, 1), gender = c(0, 1))) %>%
    mutate(pred = predict(mod1b, newdata = ., re.form = NA))

fixed.frame %>% mutate(Gender = factor(gender, levels = c(0, 1), labels = c("Male",
    "Female"))) %>% ggplot(aes(x = wave, y = pred, colour = Gender)) + geom_line(size = 1) +
    lims(y = c(190, 275)) + labs(x = "Week", y = "Weight Predicted Values",
    title = "Gender by Time Invariant Conditional Growth Models") + theme_classic() +
    theme(axis.text = element_text(face = "bold", size = rel(1.2)), axis.title = element_text(face = "b
        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 = 0.5))
```



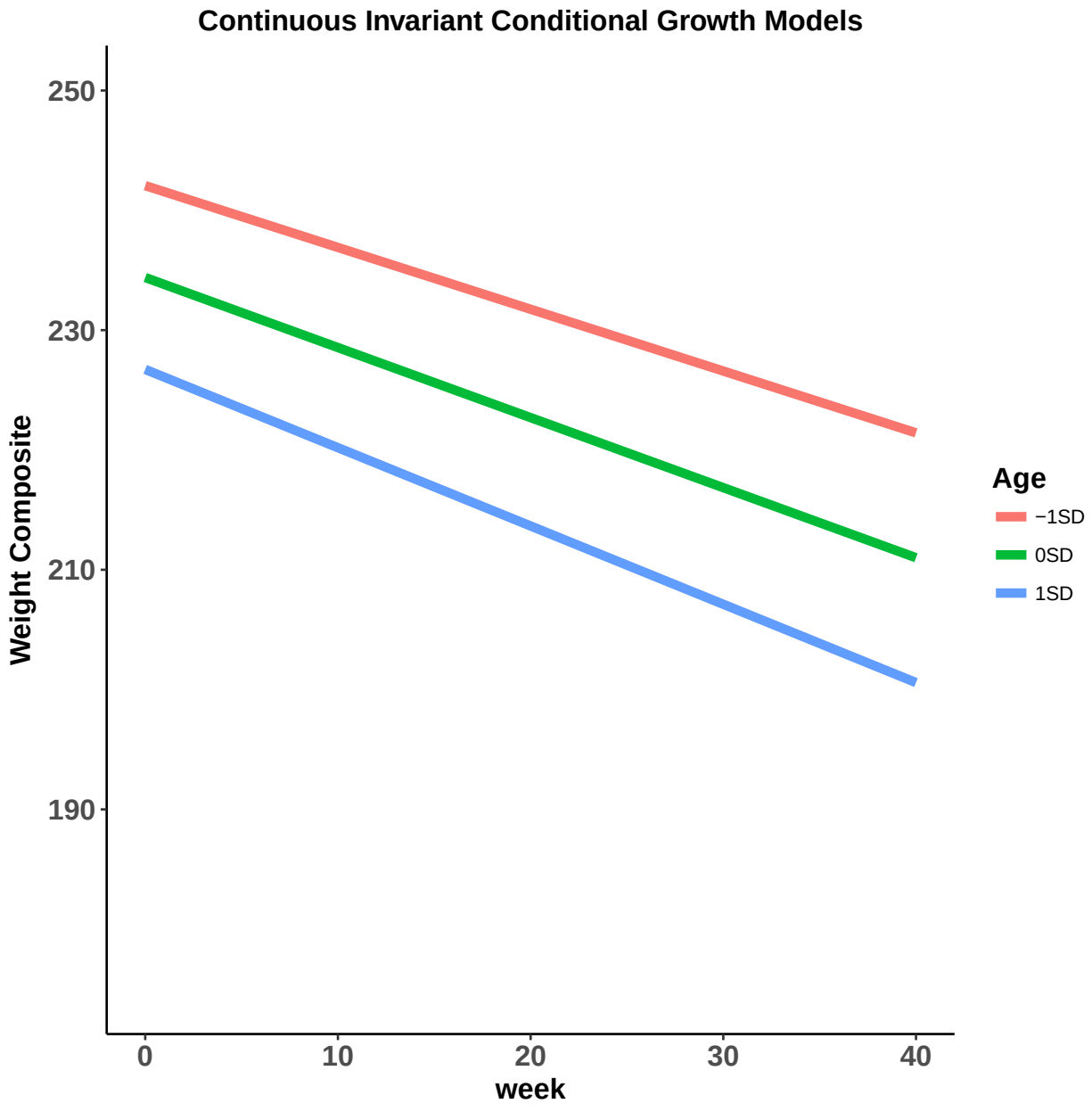
2 Graphing with Continuous Level 2 Predictors

```
#model with age as a continuous predictor
mod2c <-lmer(weight ~ wave + age.c + age.c*wave + (wave|ID),data=data)
summary(mod2c)

## Linear mixed model fit by REML ['lmerMod']
## Formula: weight ~ wave + age.c + age.c * wave + (wave | ID)
##    Data: data
##
```

```
## REML criterion at convergence: 23131.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.6268 -0.5666 -0.0638  0.4828  5.2589
##
## Random effects:
##  Groups   Name        Variance  Std.Dev. Corr
##  ID       (Intercept) 2498.7452 49.9875
##           wave           0.1677  0.4095  -0.25
##  Residual               14.4677  3.8036
## Number of obs: 3900, groups:  ID, 139
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept) 234.391903   4.241540   55.26
## wave         -0.584635   0.036183  -16.16
## age.c        -0.761723   0.421352   -1.81
## wave:age.c   -0.006835   0.003614   -1.89
##
## Correlation of Fixed Effects:
##             (Intr) wave   age.c
## wave        -0.249
## age.c        0.000  0.001
## wave:age.c   0.001 -0.035 -0.248
```

```
#graph of continuous predictors using the predict function
fixed.frame <- data %>%
  summarise(mean = mean(age.c, na.rm = T),
            sd = sd(age.c, na.rm = T))

fixed.frame <-
  data.frame(
    expand.grid(
      # here, you add values for your time variable and predictors
      wave = seq(0,40,1),
      age.c = c(fixed.frame$mean-fixed.frame$sd,
                    fixed.frame$mean,
                    fixed.frame$mean+fixed.frame$sd))) %>%
  mutate(pred = predict(mod2c, newdata = ., re.form = NA))


fixed.frame %>%
  mutate(Age = factor(age.c, levels = unique(age.c), labels = c("-1SD", "0SD", "1SD"))) %>%
  ggplot(aes(x = wave, y = pred, color = Age)) +
    geom_line(size = 2) +
    lims(y = c(175,250)) +
    labs(x = "week", y = "Weight 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))
```



3 Making a Table

```
## 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) {
```

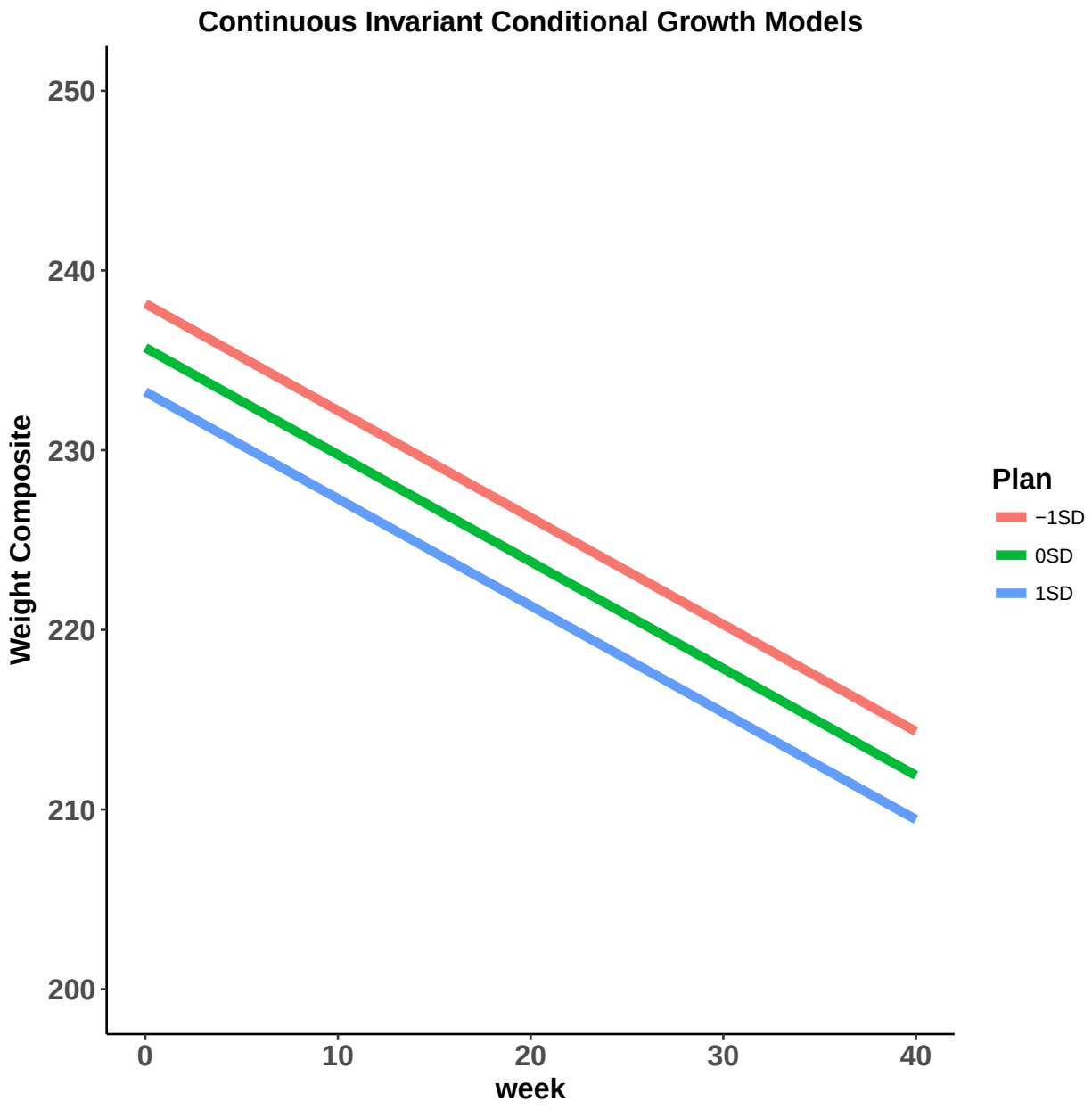
```
    fixed <- broom::tidy(model) %>% filter(group == "fixed") %>% select(term,
        estimate)
    ## add random effects ##
    rand <- broom::tidy(model) %>% filter(group != "fixed") %>% select(term,
        estimate)
    ## get confidence intervals ##
    CI <- data.frame(confint.merMod(model, 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 %>% 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(mod2c)

## Warning:  package 'bindrcpp' was built under R version 3.3.2
## Computing bootstrap confidence intervals ...
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
## Warning in norm.inter(t, alpha):  extreme order statistics used as endpoints
```

```
## Joining, by = "term"
## Joining, by = "term"
## Joining, by = c("term", "estimate", "lower", "upper", "type")
## Joining, by = c("term", "estimate", "type")
## Warning:  Column ‘estimate‘ has different attributes on LHS and RHS of join
```

4 Time-Varying Covariate: Continuous (don't have any categorical)

```
#adding planning diet as main effect (question 13)
modTV1 <- lmer(weight ~ wave + plans + (wave|ID), data = data)
summary(modTV1)

## Linear mixed model fit by REML ['lmerMod']
## Formula: weight ~ wave + plans + (wave | ID)
##    Data: data
##
## REML criterion at convergence: 3484.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7045 -0.3998  0.0125  0.4285  2.4614
##
## Random effects:
##  Groups   Name        Variance  Std.Dev. Corr
##  ID       (Intercept) 2580.9195 50.803
##           wave           0.1739  0.417   -0.40
##  Residual               23.2457  4.821
## Number of obs: 429, groups:  ID, 129
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) 245.33396    4.68368   52.38
## wave         -0.59530    0.04625  -12.87
## plans        -2.48103    0.39465   -6.29
##
## Correlation of Fixed Effects:
##       (Intr) wave
## wave  -0.254
## plans -0.284 -0.278

fixed.frame <- data %>%
  summarise(mean = mean(plans, na.rm = T),
            sd = sd(plans, na.rm = T))

fixed.frame <-
  data.frame(
    expand.grid(
      # here, you add values for your time variable and predictors
```

```
      wave = seq(0,40,1),
      plans = c(fixed.frame$mean-fixed.frame$sd,
                fixed.frame$mean,
                fixed.frame$mean+fixed.frame$sd))) %>%
  mutate(pred = predict(modTV1, newdata = ., re.form = NA))


fixed.frame %>%
  mutate(Plan = factor(plans, levels = unique(plans), labels = c("-1SD", "0SD", "1SD"))) %>%
  ggplot(aes(x = wave, y = pred, color = Plan)) +
    geom_line(size = 2) +
    lims(y = c(200,250)) +
    labs(x = "week", y = "Weight 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))
```



5 Time-Varying Covariate with interaction: Continuous (don't have any categorical)

```
#adding interaction
modTV2 <- lmer(weight ~ wave + plans + wave:plans + (wave|ID), data = data)
summary(modTV2)

## Linear mixed model fit by REML ['lmerMod']
## Formula: weight ~ wave + plans + wave:plans + (wave | ID)
```

```
##    Data: data
##
## REML criterion at convergence: 3488.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.70606 -0.39277  0.02421  0.43815  2.47574
##
## Random effects:
##  Groups   Name        Variance  Std.Dev. Corr
##  ID       (Intercept) 2582.2788 50.8161
##           wave           0.1794  0.4235  -0.40
##  Residual               22.9539  4.7910
## Number of obs: 429, groups:  ID, 129
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept) 246.13958    4.75358   51.78
## wave         -0.73244    0.14951   -4.90
## plans        -2.71202    0.45700   -5.93
## wave:plans    0.03452    0.03569    0.97
##
## Correlation of Fixed Effects:
##            (Intr) wave   plans
## wave       -0.241
## plans      -0.327  0.411
## wave:plans  0.172 -0.950 -0.510
```

```
#interaction not significant meaning the correlation between planning and weight does not increase/decr
```

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

fixed.frame <-
  data.frame(
    expand.grid(
      # here, you add values for your time variable and predictors
      wave = seq(0,40,1),
      plans = c(fixed.frame$mean-fixed.frame$sd,
                      fixed.frame$mean,
                      fixed.frame$mean+fixed.frame$sd))) %>%
  mutate(pred = predict(modTV2, newdata = ., re.form = NA))


fixed.frame %>%
  mutate(Plan = factor(plans, levels = unique(plans), labels = c("-1SD", "0SD", "1SD"))) %>%
  ggplot(aes(x = wave, y = pred, color = Plan)) +
    geom_line(size = 2) +
    lims(y = c(200,250)) +
    labs(x = "week", y = "Weight 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))
```