Teacher's Job Satisfaction and Workplace Stress: Multilevel Analysis of the TALIS 2018 Italian Sample

Script with output

Giulia Bertoldo

5/11/2022

```r
# Load packages ----------------------------------------------------------
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```r
library(ggplot2)
library(DataExplorer)
library(lme4)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```r
library(lattice)
library(sjPlot)
```

```
## Registered S3 method overwritten by 'parameters':
##   method                           from
##   format.parameters_distribution datawizard
```

```r
library(r2mlm)
```

```
## Loading required package: nlme
```

```
##
## Attaching package: 'nlme'

## The following object is masked from 'package:lme4':
##
##     lmList

## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
library(broom)
theme_set(theme_bw())

# Load data ----------------------------------------------------------------
df4 <- read.csv('../talis_data/df4.csv')

# Glimpse data
glimpse(df4)
```

```
## Rows: 3,398
## Columns: 9
## $ IDSCHOOL <int> 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3~
## $ IDTEACH  <int> 300101, 300102, 300103, 300104, 300105, 300106, 300107, 30010~
## $ TT3G01   <int> 1, 2, 1, 2, 1, 1, 2, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1~
## $ TCHAGEGR <int> 6, 6, 5, 5, 5, 4, 5, 5, 5, 5, 5, 4, 4, 4, 4, 5, 4, 3, 3, 6, 4~
## $ T3JOBSA  <dbl> 14.936916, 11.785688, 13.767235, 13.611691, 15.126234, 12.716~
## $ T3WELS   <dbl> 12.672764, 9.108597, 7.937173, 7.596593, 7.744105, 9.303835, ~
## $ T3SELF   <dbl> 15.50926, 12.22587, 11.26625, 13.63967, 15.50926, 13.50305, 1~
## $ T3PLACRE <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ SCHLOC   <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2~
```

```
# Clean data ---------------------------------------------------------------
# * Convert to factors ----------------------------------------------------
# Convert to factor: School ID
df4$IDSCHOOL <- as.factor(df4$IDSCHOOL)
nlevels(df4$IDSCHOOL)
```

```
## [1] 186
```

```
# Convert to factor: Teacher ID
df4$IDTEACH <- as.factor(df4$IDTEACH)
nlevels(df4$IDTEACH)
```

```
## [1] 3398
```

```
# Convert to factor: Gender
df4$TT3G01 <- as.factor(df4$TT3G01)
nlevels(df4$TT3G01)
```

```
## [1] 2
```

```
# Convert to factor: Age group
df4$TCHAGEGR <- as.factor(df4$TCHAGEGR)
nlevels(df4$TCHAGEGR)
```

```
## [1] 6
```

```
# Convert to factor: Lack of resources
df4$T3PLACRE <- as.factor(df4$T3PLACRE)
```

```r
nlevels(df4$T3PLACRE)
```

```
## [1] 3
```

```r
# Convert to factor: School location
df4$SCHLOC <- as.factor(df4$SCHLOC)
nlevels(df4$SCHLOC)
```

```
## [1] 3
```

```r
# Glimpse data
glimpse(df4)
```

```
## Rows: 3,398
## Columns: 9
## $ IDSCHOOL <fct> 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3~
## $ IDTEACH  <fct> 300101, 300102, 300103, 300104, 300105, 300106, 300107, 30010~
## $ TT3G01   <fct> 1, 2, 1, 2, 1, 1, 2, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1~
## $ TCHAGEGR <fct> 6, 6, 5, 5, 5, 4, 5, 5, 5, 5, 5, 4, 4, 4, 4, 5, 4, 3, 3, 6, 4~
## $ T3JOBSA  <dbl> 14.936916, 11.785688, 13.767235, 13.611691, 15.126234, 12.716~
## $ T3WELS   <dbl> 12.672764, 9.108597, 7.937173, 7.596593, 7.744105, 9.303835, ~
## $ T3SELF   <dbl> 15.50926, 12.22587, 11.26625, 13.63967, 15.50926, 13.50305, 1~
## $ T3PLACRE <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ SCHLOC   <fct> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2~
```

```r
# * Center lvl-1 predictors WELS & create lvl-2 predictor -----------------
df4 <- df4 %>%
  group_by(IDSCHOOL) %>%
  # CM = Cluster Mean
  # CMC = Cluster Mean Centered variable
  mutate(T3WELS_CM = mean(T3WELS),
         T3WELS_CMC = T3WELS - T3WELS_CM,
         T3SELF_CM = mean(T3SELF),
         T3SELF_CMC = T3SELF - T3SELF_CM) %>%
  ungroup() %>%
  # Grand mean centering (GMC) of the aggregated variable
  mutate(T3WELS_CM_GMC = T3WELS_CM - mean(T3WELS_CM))

# Check centering results
df4 %>%
  select(T3WELS,
         T3WELS_CM,
         T3WELS_CMC,
         T3WELS_CM_GMC,
         T3SELF_CM,
         T3SELF_CMC) %>%
  summary
```

```
##      T3WELS          T3WELS_CM        T3WELS_CMC      T3WELS_CM_GMC
##  Min.   : 7.449   Min.   : 8.244   Min.   :-3.4026   Min.   :-1.08215
##  1st Qu.: 7.937   1st Qu.: 8.940   1st Qu.:-1.2381   1st Qu.:-0.38596
##  Median : 8.768   Median : 9.235   Median :-0.4596   Median :-0.09041
##  Mean   : 9.326   Mean   : 9.326   Mean   : 0.0000   Mean   : 0.00000
##  3rd Qu.:10.475   3rd Qu.: 9.713   3rd Qu.: 1.0521   3rd Qu.: 0.38726
##  Max.   :15.504   Max.   :11.340   Max.   : 6.5836   Max.   : 2.01399
##    T3SELF_CM        T3SELF_CMC
```

```
##  Min.   :11.57   Min.   :-9.69565
##  1st Qu.:12.43   1st Qu.:-1.17782
##  Median :12.69   Median :-0.05129
##  Mean   :12.70   Mean   : 0.00000
##  3rd Qu.:13.00   3rd Qu.: 1.15192
##  Max.   :14.06   Max.   : 3.77964
```

```r
# Dataset description -----------------------------------------------------
# How many teachers?
length(unique(df4$IDTEACH))
```

```
## [1] 3398
```

```r
# How many schools?
length(unique(df4$IDSCHOOL))
```

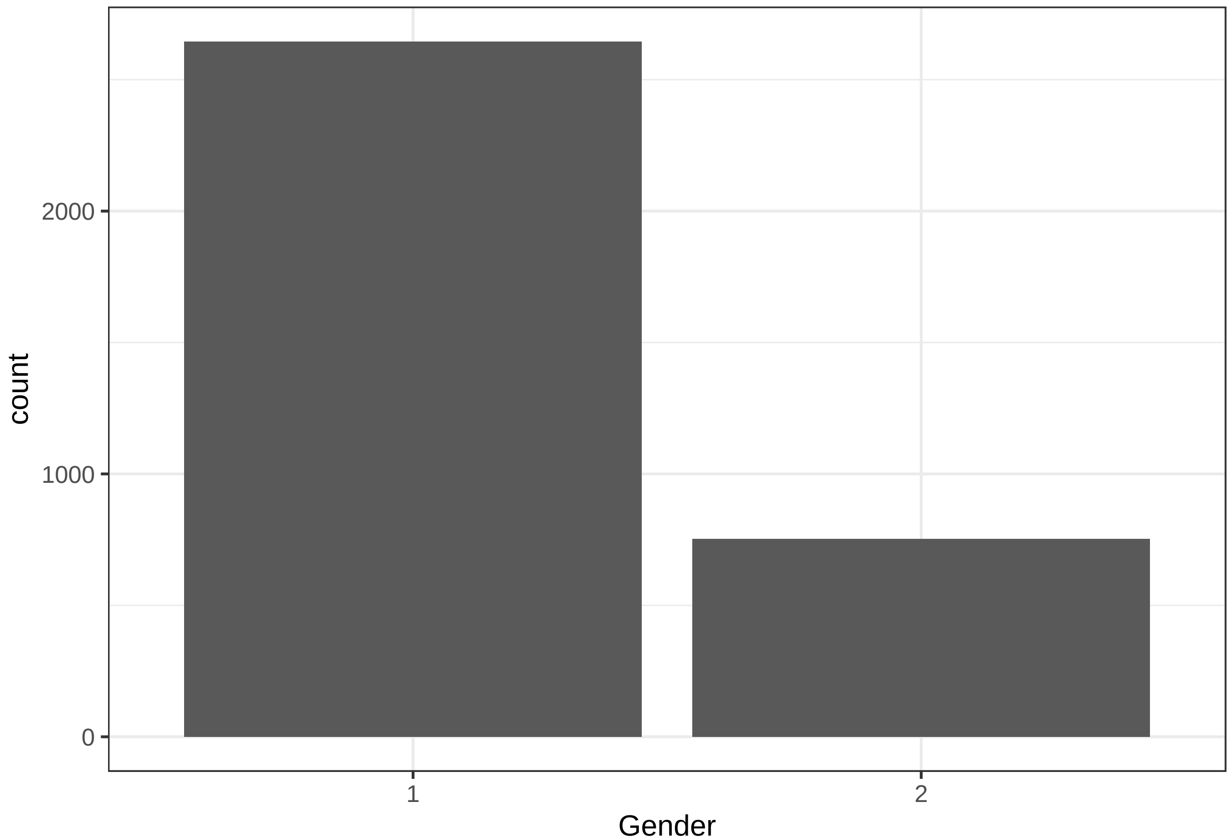
```
## [1] 186
```

```r
# Average number of teacher's per school
df4 %>%
  group_by(IDSCHOOL) %>%
  summarise(count = n()) %>%
  mutate(max = max(count),
         min = min(count),
         average = mean(count))
```

```
## # A tibble: 186 x 5
##    IDSCHOOL count   max   min average
##    <fct>    <int> <int> <int>   <dbl>
##  1 3001        19    29     7    18.3
##  2 3002        20    29     7    18.3
##  3 3003        19    29     7    18.3
##  4 3004        14    29     7    18.3
##  5 3005        18    29     7    18.3
##  6 3006        19    29     7    18.3
##  7 3007        20    29     7    18.3
##  8 3008        19    29     7    18.3
##  9 3009        20    29     7    18.3
## 10 3010        14    29     7    18.3
## # ... with 176 more rows
```

```r
# Exploratory data analysis  ----------------------------------------------
# * Univariate analysis ---------------------------------------------------

# ** Teachers' gender -----------------------------------------------------
ggplot(df4, aes(x=TT3G01)) +
  geom_bar() +
  labs(x = 'Gender')
```



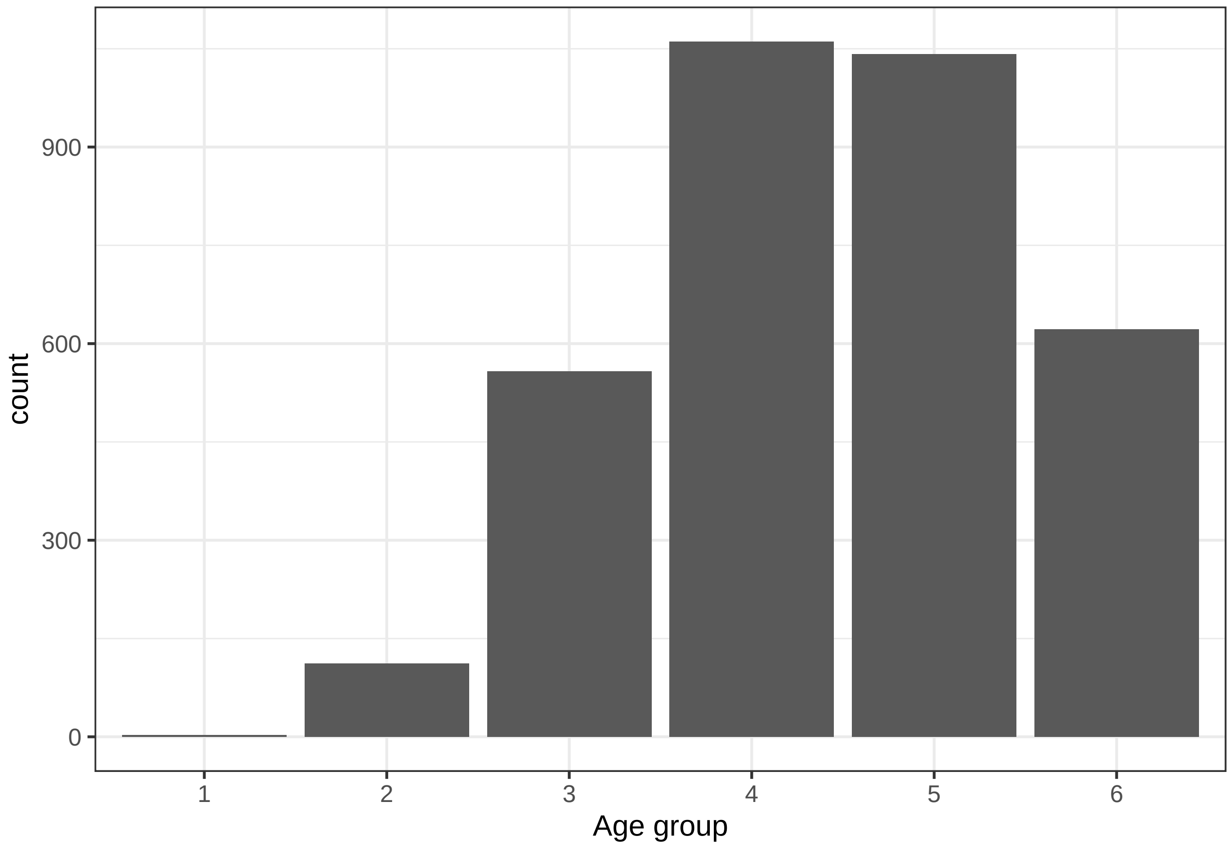

```
gender <- summary(df4$TT3G01)
(per_female <- gender[1]/(gender[1]+gender[2]))
```

```
##         1
## 0.7783991
```

```
(per_male <- gender[2]/(gender[1]+gender[2]))
```

```
##         2
## 0.2216009
```

```
# ** Teachers' age group --------------------------------------------------
ggplot(df4, aes(x=TCHAGEGR)) +
  geom_bar() +
  labs(x = 'Age group')
```




```
# ** Teachers' self efficacy ----------------------------------------------
ggplot(df4, aes(x=T3SELF_CMC)) +
  geom_histogram( colour="black", fill="white") +
  labs(x = "Teacher's Self Efficacy, composite")
```

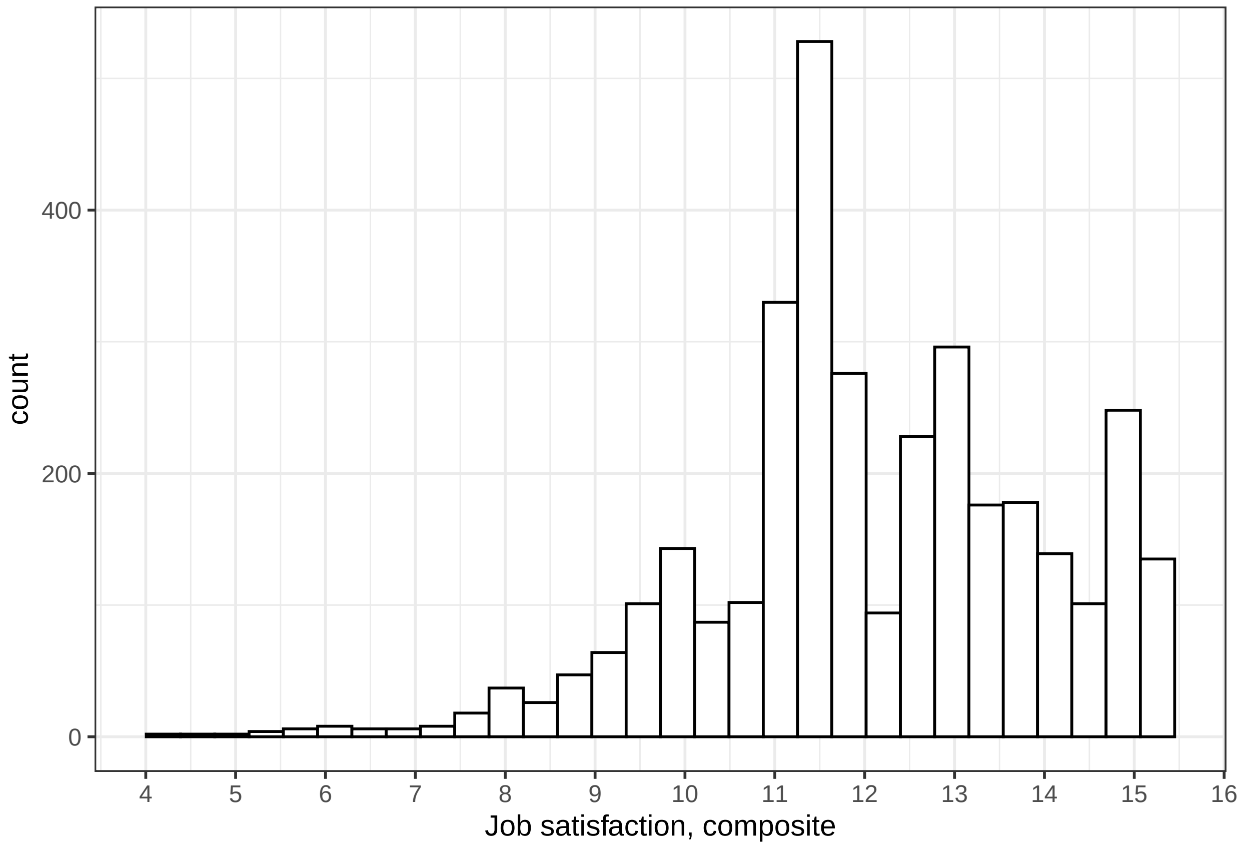
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


```
summary(df4$T3SELF_CMC)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -9.69565 -1.17782 -0.05129  0.00000  1.15192  3.77964
```

```
# ** Job Satisfaction, composite ------------------------------------------
ggplot(df4, aes(x=T3JOBSA)) +
  geom_histogram( colour="black", fill="white") +
  labs(x = 'Job satisfaction, composite') +
  scale_x_continuous(breaks = seq(1:17))
```

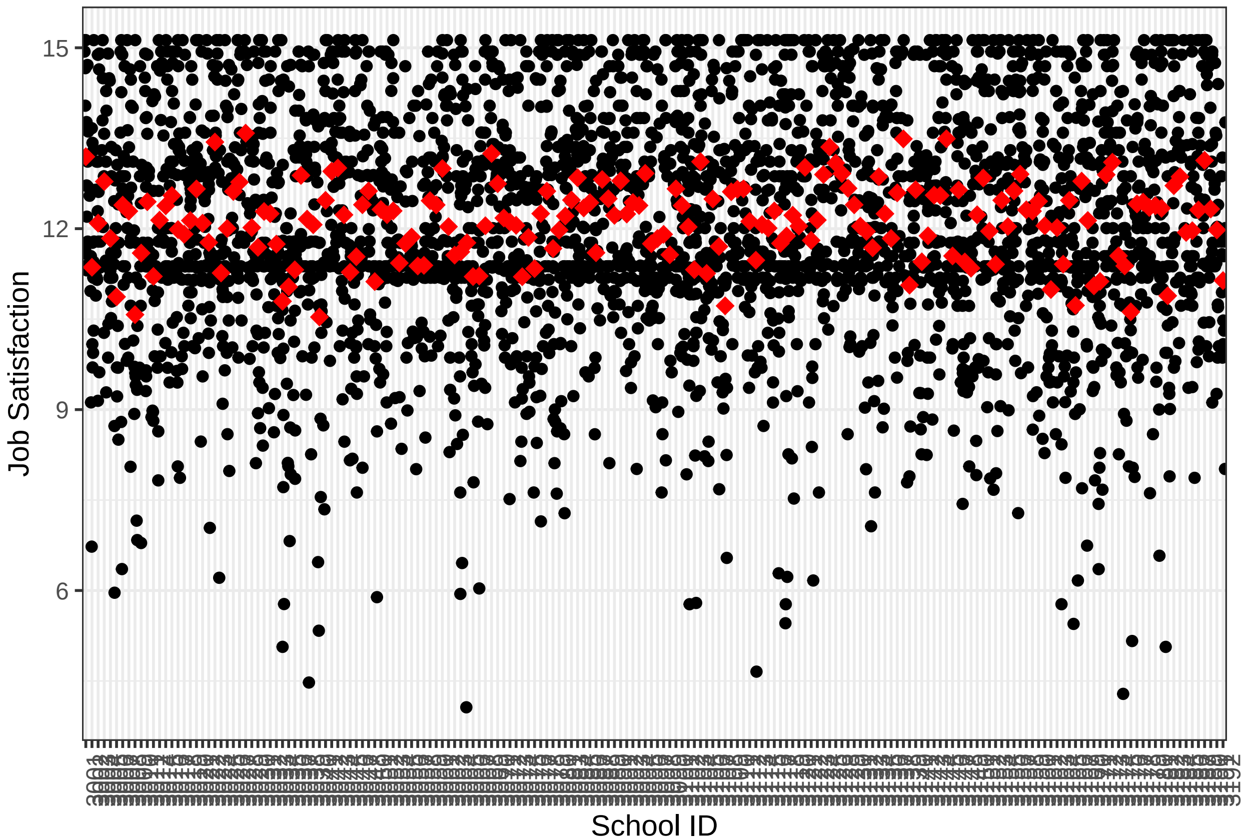
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



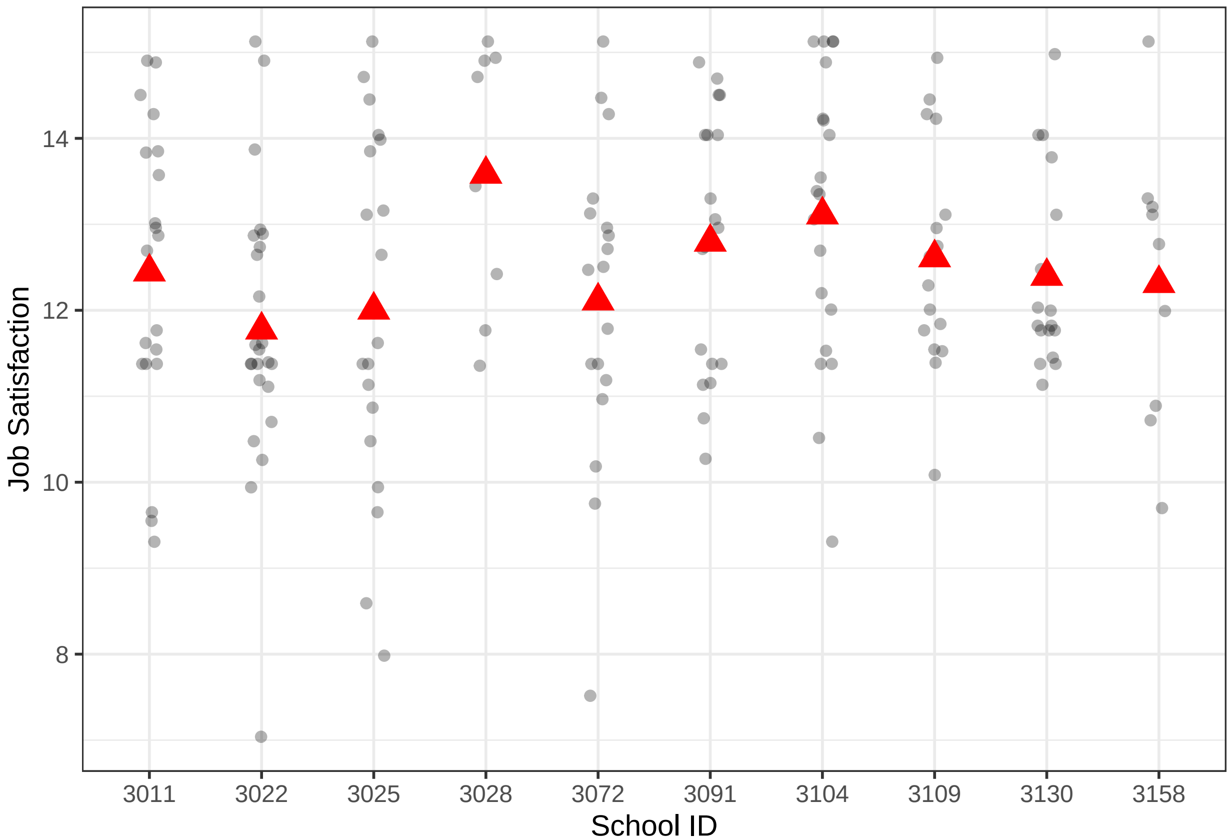

```r
summary(df4$T3JOBSA)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.064  11.154  11.819  12.087  13.369  15.126
```

```r
# Basic stripchart
ggplot(df4, aes(x=IDSCHOOL, y=T3JOBSA)) +
  geom_jitter() +
  stat_summary(fun=mean, geom="point", shape=18,
               size=3, color="red") +
  labs(y = "Job Satisfaction",
       x = "School ID") +
  theme(axis.text.x = element_text(angle = 90))
```

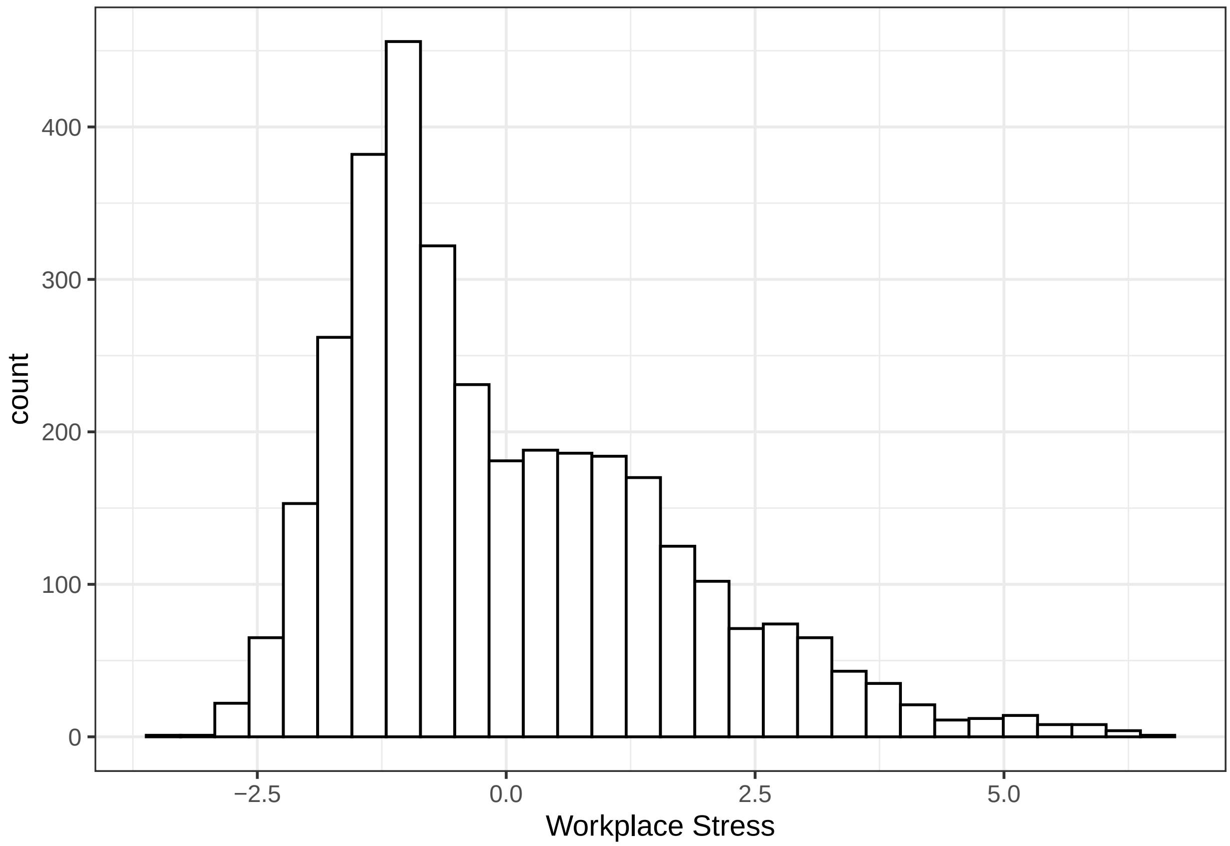



```r
# Show variation across a subset of ten schools
set.seed(1994)
random_id <- sample(unique(df4$IDSCHOOL), size = 10)
(p_subset <- df4 %>%
    filter(IDSCHOOL %in% random_id) %>%  # select only 10 schools
    ggplot(aes(x = IDSCHOOL, y = T3JOBSA )) +
    geom_jitter(height = 0, width = 0.1, alpha = 0.3) +
    labs(y = "Job Satisfaction",
         x = "School ID") +
    # Add school means
    stat_summary(
      fun = "mean",
      geom = "point",
      col = "red",
      shape = 17,
      # use triangles
      size = 4
    )  # make them larger
)
```




```r
# ** Workplace stress ----------------------------------------------------
ggplot(df4, aes(x=T3WELS_CMC)) +
  geom_histogram( colour="black", fill="white") +
  labs(x = 'Workplace Stress')
```

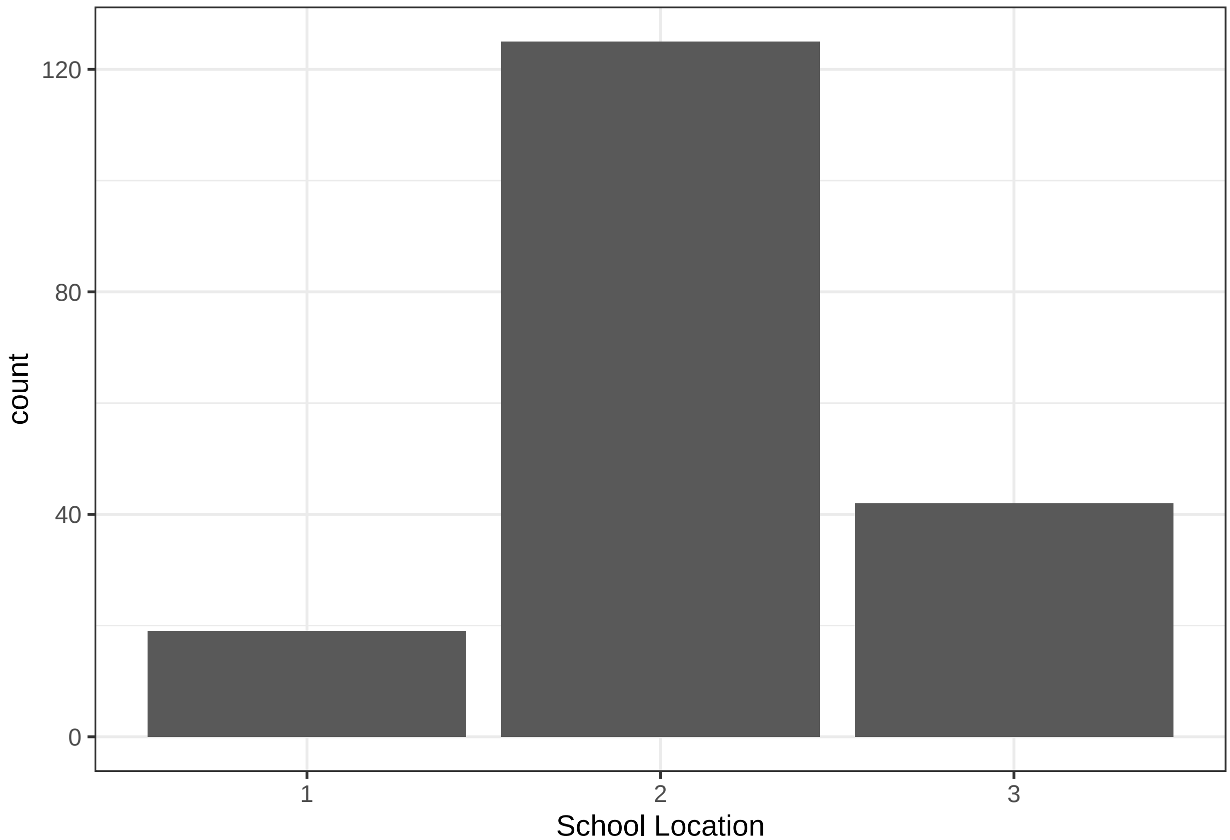
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
summary(df4$T3WELS_CMC)
```

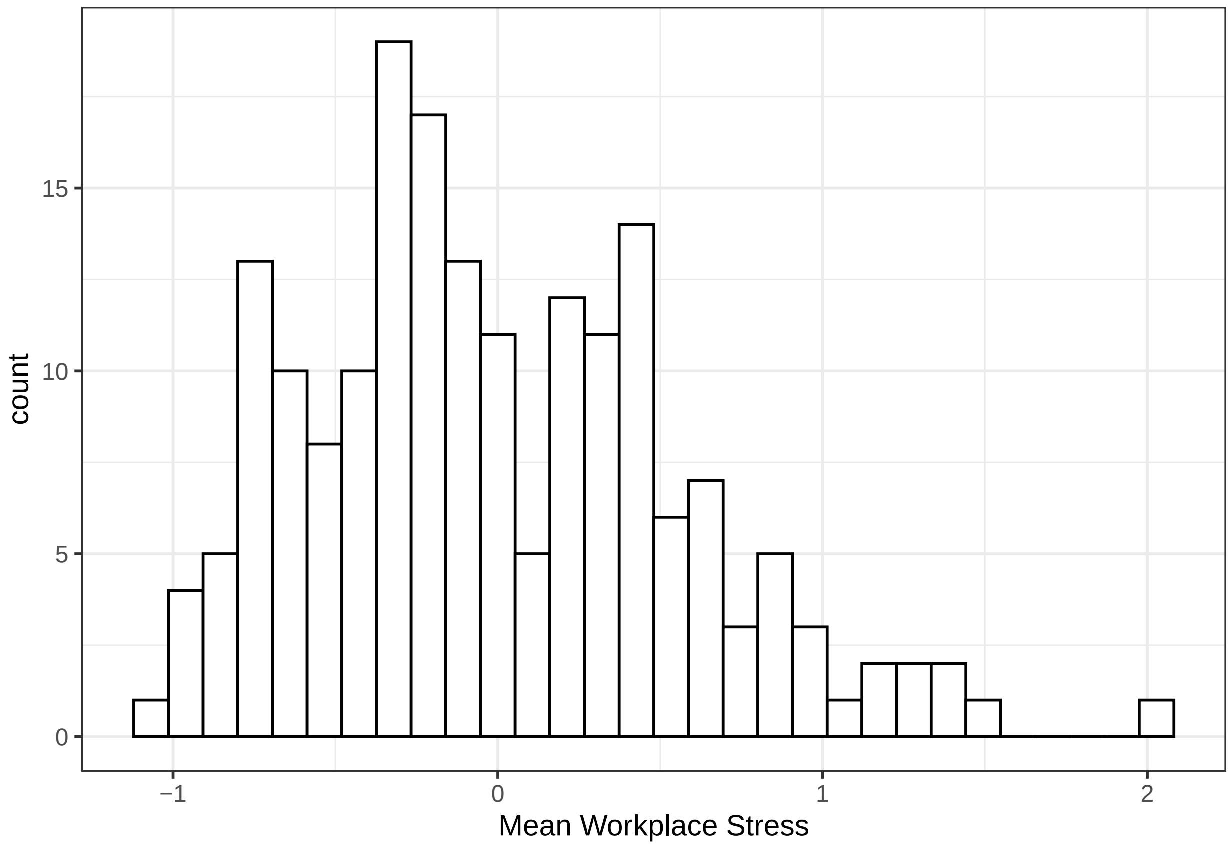
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.4026 -1.2381 -0.4596  0.0000  1.0521  6.5836
```

```r
# ** School location ------------------------------------------------------
df4 %>%
  group_by(IDSCHOOL) %>%
  filter(row_number()==1) %>%
  ggplot(aes(x=SCHLOC)) +
  geom_bar() +
  labs(x = "School Location")
```




```
# ** Mean workplace stress -----------------------------------------------
df4 %>%
  group_by(IDSCHOOL) %>%
  filter(row_number()==1) %>%
  ggplot(aes(x=T3WELS_CM_GMC)) +
  geom_histogram( colour="black", fill="white") +
  labs(x = 'Mean Workplace Stress')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
summary(df4$T3WELS_CM_GMC)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -1.08215 -0.38596 -0.09041  0.00000  0.38726  2.01399
```

```r
# * Bivariate analysis ----------------------------------------------------

# ** Workplace Stress vs Job Satisfaction ---------------------------------
# Overall regression line
ggplot(data  = df4,
       aes(x = T3WELS_CMC,
           y = T3JOBSA,
           col = IDSCHOOL))+
  geom_point(size = 1.2,
             alpha = .8)+
  geom_smooth(method = lm,
              se     = FALSE,
              col = "black",
              size   = .5,
              alpha  = .8)+
  theme(legend.position = "none")+
  labs(title = "Job Satisfaction vs. Workplace Stress",
       subtitle = "Colored by school",
       x = "Workplace Stress",
       y = "Job Satisfaction")
```

```
## `geom_smooth()` using formula 'y ~ x'
```


```r
# One regression line per school
ggplot(data  = df4,
       aes(x = T3WELS_CMC,
           y = T3JOBSA,
           col = IDSCHOOL,
           group = IDSCHOOL))+
  geom_point(size = 1.2,
             alpha = .8)+
  geom_smooth(method = lm,
              se     = FALSE,
              size   = .5,
              alpha  = .8)+
  theme(legend.position = "none")+
  labs(title = "Job Satisfaction vs. Workplace Stress",
       subtitle = "Colored by school",
       x = "Workplace Stress",
       y = "Job Satisfaction")
```

```
## `geom_smooth()` using formula 'y ~ x'
```


```r
# Unconditional random intercept model ------------------------------------
model1 <- lmer(T3JOBSA ~ 1 + (1 | IDSCHOOL), data = df4)
# Summarize results
summary(model1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: T3JOBSA ~ 1 + (1 | IDSCHOOL)
##    Data: df4
##
## REML criterion at convergence: 13765.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3803 -0.5639 -0.0489  0.6991  2.1757
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  IDSCHOOL (Intercept) 0.217    0.4658
##  Residual             3.218    1.7938
## Number of obs: 3398, groups:  IDSCHOOL, 186
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) 12.09876    0.04619   261.9
```

```r
# Profile likelihood confidence intervals
confint(model1)
```

```
## Computing profile confidence intervals ...

##                  2.5 %     97.5 %
## .sig01       0.3793898  0.5547561
## .sigma       1.7508008  1.8385610
## (Intercept) 12.0081355 12.1897162
```

```
# Testing for "school effects"
# Null single level model
fit <- lm(T3JOBSA ~ 1, data = df4)
# Likelihood ratio
(ll_simple<-logLik(fit)[1])
```

```
## [1] -6916.383
```

```
(ll_complex <-logLik(model1)[1])
```

```
## [1] -6882.907
```

```
(LR <- -2*ll_simple-(-2*ll_complex)) # 1df because only 1 parameter difference
```

```
## [1] 66.95039
```

```
(pval_lr <- (pchisq(LR, df=1, lower.tail = FALSE)/2))
```

```
## [1] 1.392129e-16
```

```
# * ICC ------------------------------------------------------------------
variance_components <- as.data.frame(VarCorr(model1))
(between_var <- variance_components$vcov[1])
```

```
## [1] 0.2169971
```

```
(within_var <- variance_components$vcov[2])
```

```
## [1] 3.217665
```

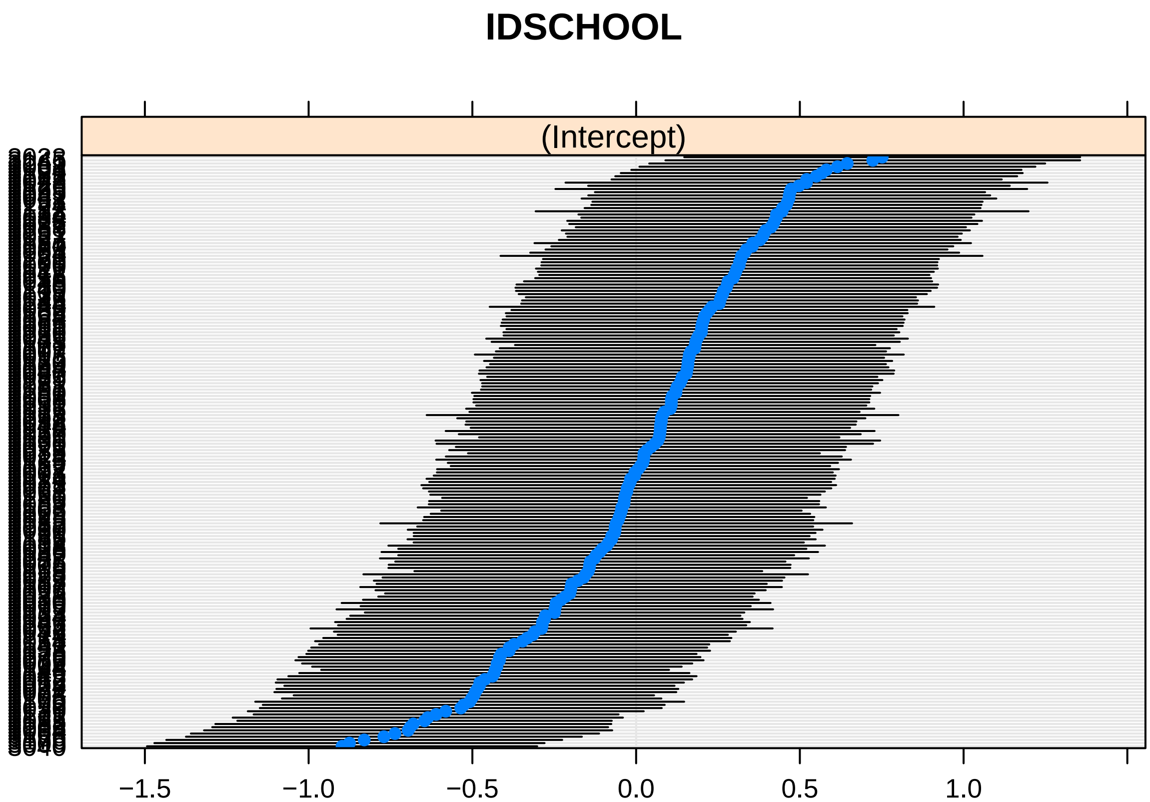
```
(icc <- between_var / (between_var + within_var))
```

```
## [1] 0.06317859
```

```
# * Empirical Bayes estimates --------------------------------------------

dotplot(ranef(model1, condVar = TRUE))
```

```
## $IDSCHOOL
```




```
# * Design effect --------------------------------------------------------
cluster_size <- df4 %>%
  group_by(IDSCHOOL) %>%
  summarise(count = n())
(average_cluster_size <- mean(cluster_size$count))
```

```
## [1] 18.26882
```

```
(design_effect <- 1 + (average_cluster_size - 1) * icc)
```

```
## [1] 2.091019
```

```
# Effective sample size
(n_eff <- length(df4$IDTEACH)/design_effect)
```

```
## [1] 1625.045
```

```
# Add lvl-2 predictor: mean workplace stress ------------------------------
model2_pre <- lmer(T3JOBSA ~ T3WELS_CM_GMC + (1 | IDSCHOOL), data = df4)
# Summarize results
summary(model2_pre)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: T3JOBSA ~ T3WELS_CM_GMC + (1 | IDSCHOOL)
##    Data: df4
##
## REML criterion at convergence: 13702.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4256 -0.5753 -0.0469  0.7082  2.4371
```

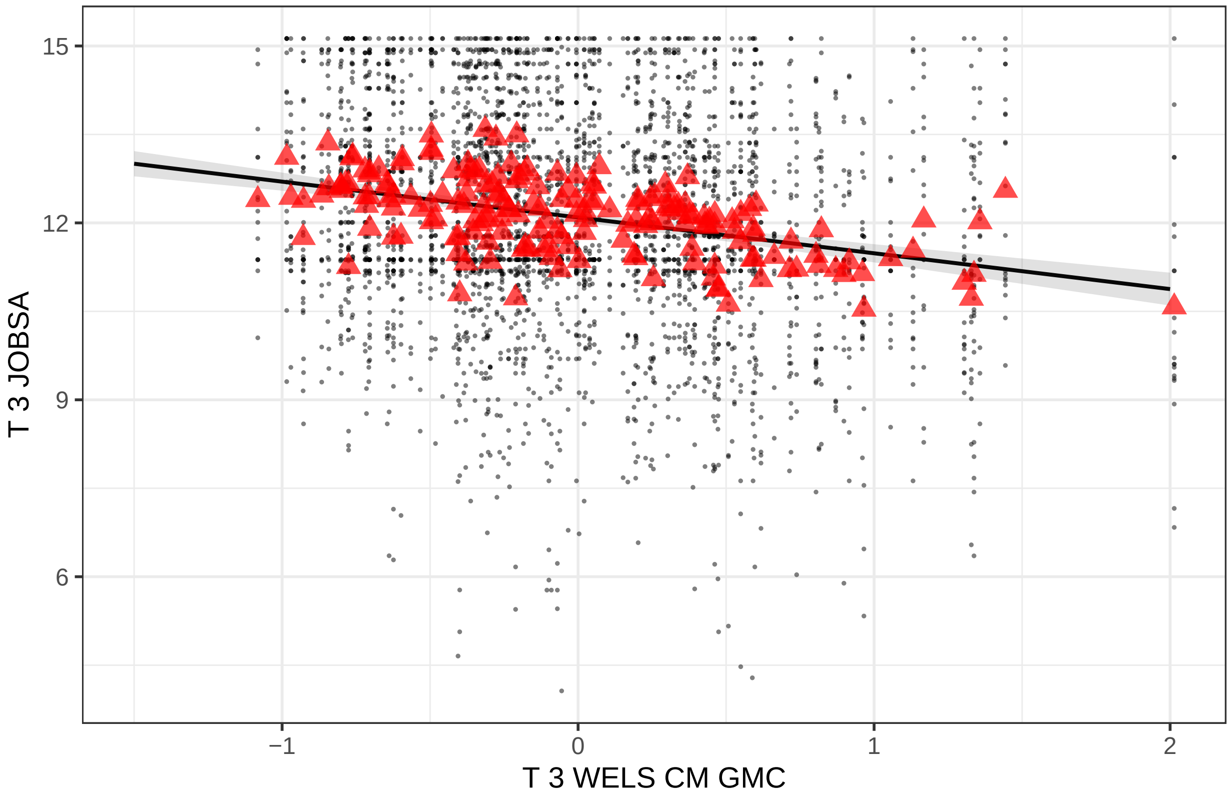
```
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  IDSCHOOL (Intercept) 0.1008   0.3176
##  Residual             3.2164   1.7934
## Number of obs: 3398, groups:  IDSCHOOL, 186
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   12.09210    0.03875 312.062
## T3WELS_CM_GMC -0.60739    0.06829  -8.895
##
## Correlation of Fixed Effects:
##             (Intr)
## T3WELS_CM_G 0.007
```

```
# Likelihood-based confidence intervals for fixed effects
confint(model2_pre)
```

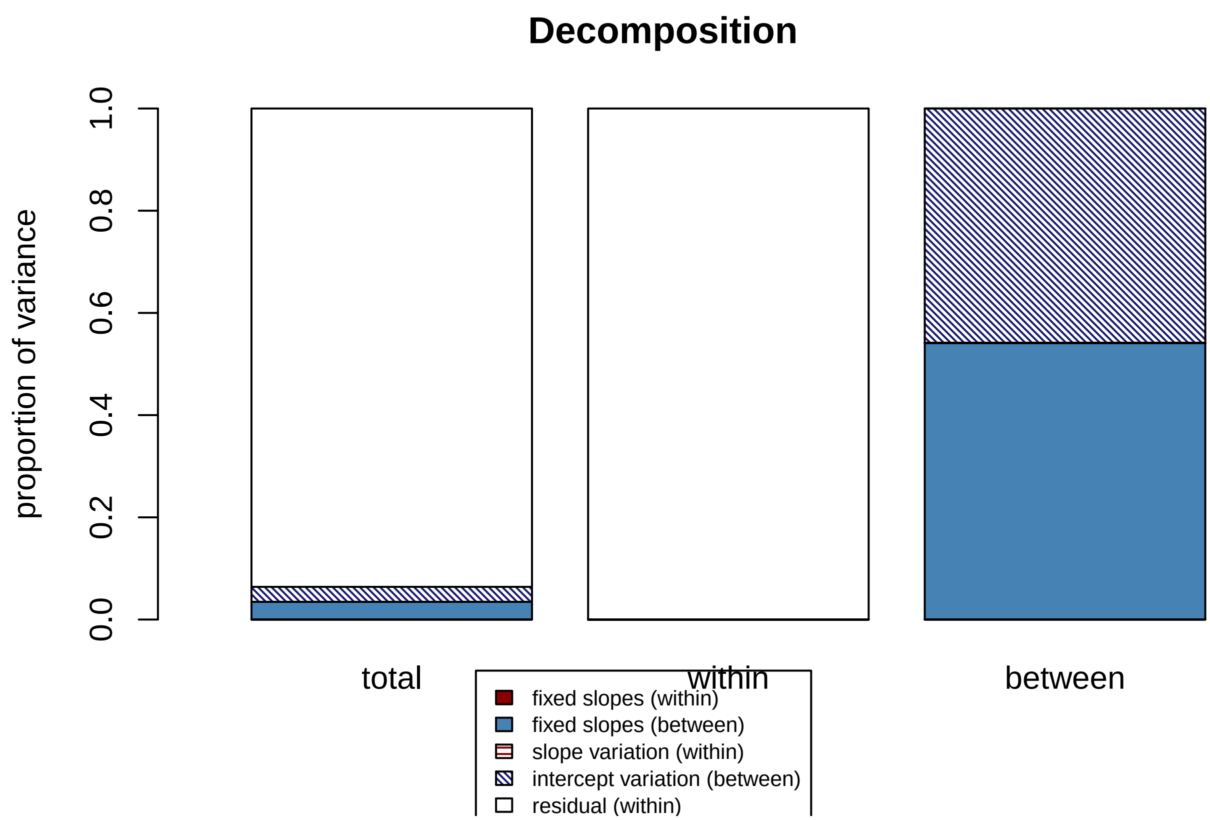
```
## Computing profile confidence intervals ...
```

```
##                    2.5 %     97.5 %
## .sig01         0.2194844  0.4033291
## .sigma         1.7504740  1.8381819
## (Intercept)   12.0162400 12.1681472
## T3WELS_CM_GMC -0.7412955 -0.4736586
```

```
# Plot
sjPlot::plot_model(model2_pre,
                   type = "pred",
                   terms = "T3WELS_CM_GMC",
                   show.data = TRUE,
                   title = "",
                   dot.size = 0.5) +
  stat_summary(data = df4, aes(x = T3WELS_CM_GMC, y = T3JOBSA),
               fun = mean, geom = "point",
               col = "red",
               shape = 17,
               # use triangles
               size = 3,
               alpha = 0.7)
```




```r
# Proportion of variance explained
## Use Rights & Sterba (2019)
r2mlm::r2mlm(model2_pre)
```




```
## $Decompositions
##                    total               within between
## fixed, within      0                   0      NA
## fixed, between     0.0346214414535156 NA     0.541221947359481
## slope variation 0                   0      NA
## mean variation   0.0293475857125612 NA     0.458778052640519
## sigma2            0.936030972833923  1      NA
##
## $R2s
##     total              within between
## f1  0                  0      NA
## f2  0.0346214414535156 NA     0.541221947359481
## v   0                  0      NA
## m   0.0293475857125612 NA     0.458778052640519
## f   0.0346214414535156 NA     NA
## fv  0.0346214414535156 0      NA
## fvm 0.0639690271660768 NA     NA
```

```
# Add lvl-1 predicor: workplace stress (between-within model) -------------
model2 <- lmer(T3JOBSA ~ T3WELS_CMC+ T3WELS_CM_GMC + (1 | IDSCHOOL), data = df4)
# Summarize results
summary(model2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: T3JOBSA ~ T3WELS_CMC + T3WELS_CM_GMC + (1 | IDSCHOOL)
##    Data: df4
##
## REML criterion at convergence: 13031.2
##
```

```
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3598 -0.6744 -0.0129  0.7310  3.4977
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  IDSCHOOL (Intercept) 0.1342   0.3663
##  Residual             2.6055   1.6142
## Number of obs: 3398, groups:  IDSCHOOL, 186
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   12.09396    0.03876 312.019
## T3WELS_CMC    -0.45864    0.01670 -27.469
## T3WELS_CM_GMC -0.60916    0.06833  -8.915
##
## Correlation of Fixed Effects:
##             (Intr) T3WELS_CMC
## T3WELS_CMC  0.000
## T3WELS_CM_G 0.009  0.000
```

```
# Likelihood-based confidence intervals for fixed effects
confint(model2)
```

```
## Computing profile confidence intervals ...
```

```
##                    2.5 %     97.5 %
## .sig01         0.2856582  0.4428430
## .sigma         1.5752431  1.6541868
## (Intercept)   12.0180716 12.1700097
## T3WELS_CMC    -0.4913706 -0.4259122
## T3WELS_CM_GMC -0.7431460 -0.4753427
```

```
# * Contextual effect -----------------------------------------------------
-0.60916 - (-0.45864)
```

```
## [1] -0.15052
```

```
## Look at the reparametrized model
model2_context <- lmer(T3JOBSA ~ T3WELS + T3WELS_CM_GMC + (1 | IDSCHOOL), data = df4)
summary(model2_context)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: T3JOBSA ~ T3WELS + T3WELS_CM_GMC + (1 | IDSCHOOL)
##    Data: df4
##
## REML criterion at convergence: 13031.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3598 -0.6744 -0.0129  0.7310  3.4977
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  IDSCHOOL (Intercept) 0.1342   0.3663
##  Residual             2.6055   1.6142
## Number of obs: 3398, groups:  IDSCHOOL, 186
```

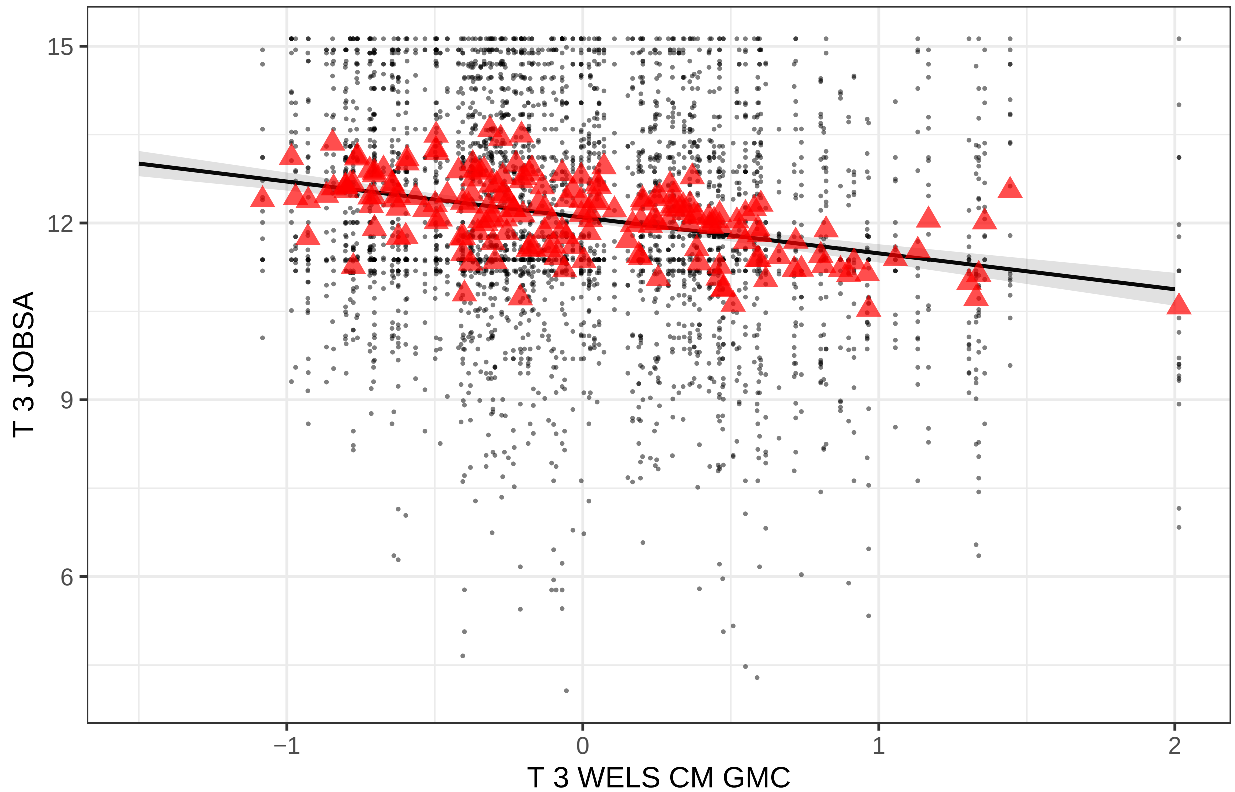
```
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)    16.37113    0.16046  102.03
## T3WELS         -0.45864    0.01670  -27.47
## T3WELS_CM_GMC -0.15052    0.07034   -2.14
##
## Correlation of Fixed Effects:
##              (Intr) T3WELS
## T3WELS       -0.970
## T3WELS_CM_G  0.233 -0.237
```

```
# Likelihood-based confidence intervals for fixed effects
confint(model2_context )
```

```
## Computing profile confidence intervals ...
```

```
##                     2.5 %      97.5 %
## .sig01          0.2856582  0.44284296
## .sigma          1.5752431  1.65418680
## (Intercept)    16.0565977 16.68547926
## T3WELS         -0.4913706 -0.42591222
## T3WELS_CM_GMC -0.2883959 -0.01278995
```

```
# * Visualizing the difference --------------------------------------------
sjPlot::plot_model(model2,
                   type = "pred",
                   terms = "T3WELS_CM_GMC",
                   show.data = TRUE,
                   title = "",
                   dot.size = 0.5) +
  stat_summary(data = df4, aes(x = T3WELS_CM_GMC, y = T3JOBSA),
               fun = mean, geom = "point",
               col = "red",
               shape = 17,
               # use triangles
               size = 3,
               alpha = 0.7)
```

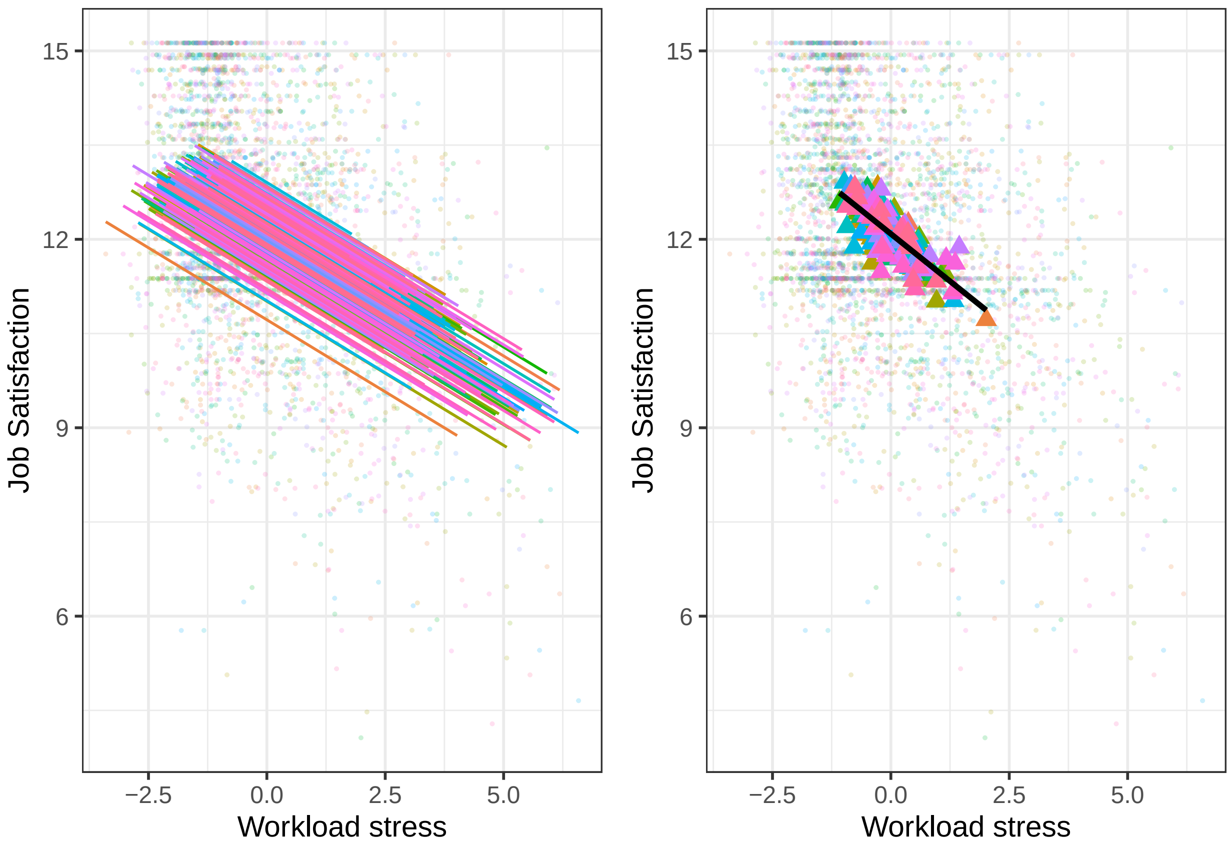



```r
# Create a common base graph
pbase <- augment(model2, data = df4) %>%
  ggplot(aes(x = T3WELS_CMC, y = T3JOBSA, color = factor(IDSCHOOL))) +
  # Add points
  geom_point(size = 0.2, alpha = 0.2) +
  labs(y = "Job Satisfaction") +
  # Suppress legend
  guides(color = "none")
# Lv-1 effect
p1 <- pbase +
  # Add within-cluster lines
  geom_smooth(aes(y = .fitted),
              method = "lm", se = FALSE, size = 0.5) +
  labs(x="Workload stress")
# Lv-2 effect
p2 <- pbase +
  # Add group means
  stat_summary(aes(x = T3WELS_CM_GMC, y = .fitted),
               fun = mean,
               geom = "point",
               shape = 17,
               # use triangles
               size = 2.5) +
  # Add between coefficient
  geom_smooth(aes(x = T3WELS_CM_GMC, y = .fitted),
              method = "lm", se = FALSE,
              color = "black") +
  labs(x="Workload stress")
```

```r
# Put the two graphs together (need the gridExtra package)
gridExtra::grid.arrange(p1, p2, ncol = 2)
```

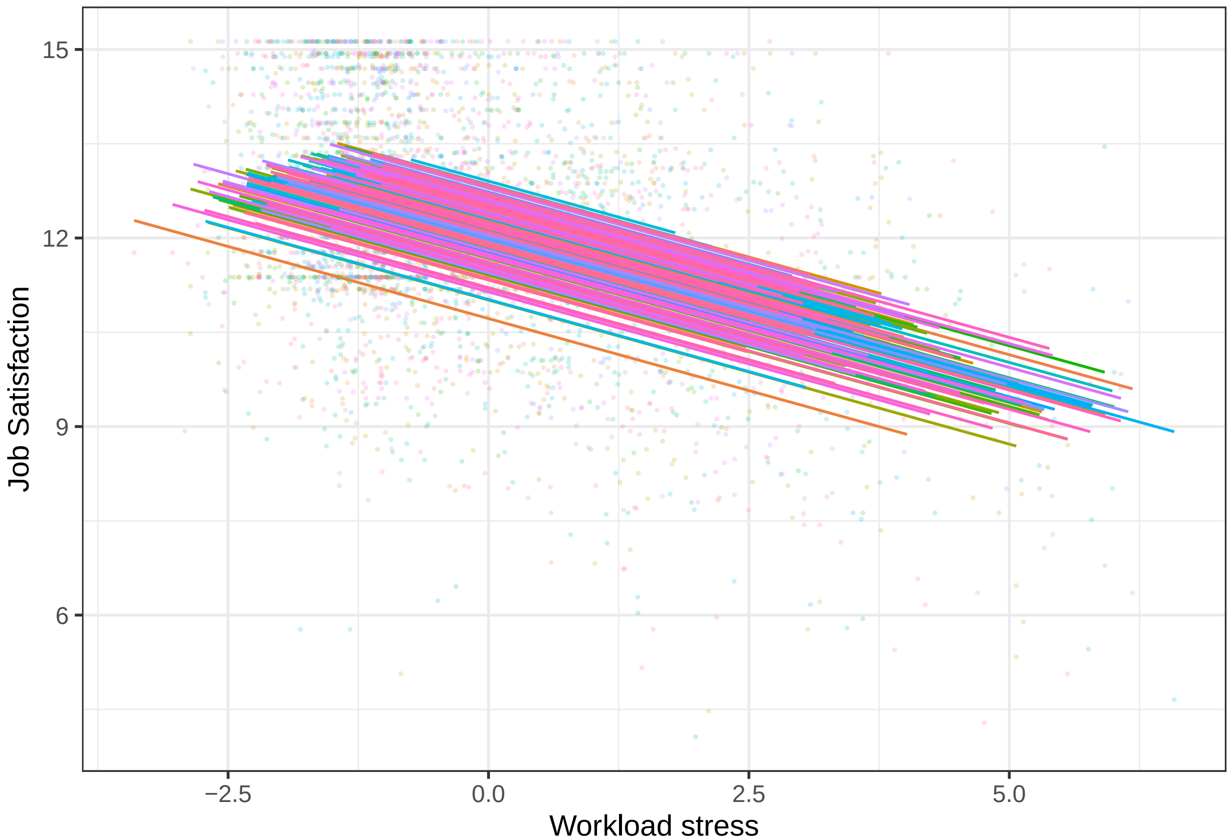
```
## `geom_smooth()` using formula 'y ~ x'

## `geom_smooth()` using formula 'y ~ x'
```



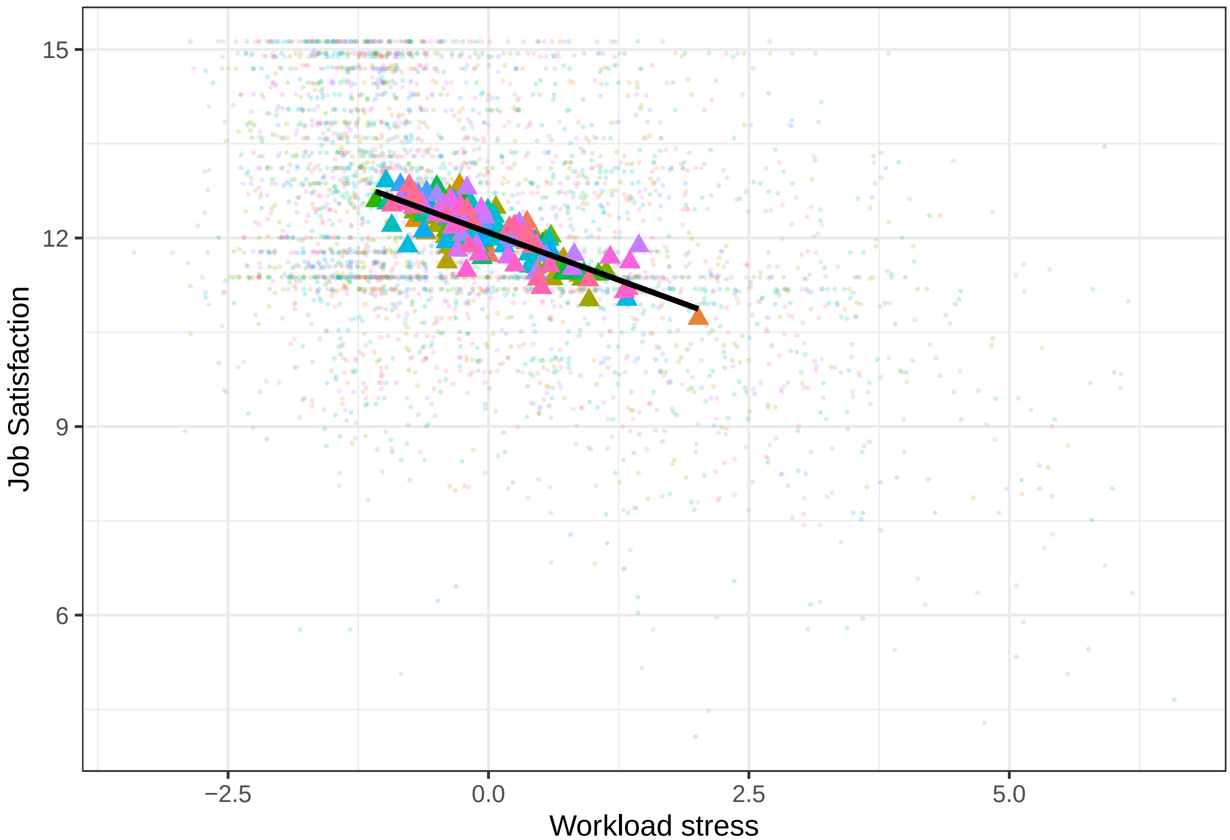

```r
# Two separate graphs
p1
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
p2
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```r
# * How much variance explained? -------------------------------------------
# School-level variance
round((1-(0.1342/0.217)),2)
```

```
## [1] 0.38
```

```r
# Teacher-level variance
round((1-(2.6055/3.218)),2)
```

```
## [1] 0.19
```

```r
# * LRT for school-level variance ------------------------------------------
# Null single level model
fit <- lm(T3JOBSA ~ T3WELS + T3WELS_CM_GMC, data = df4)
# Likelihood ratio
(ll_simple<-logLik(fit)[1])
```

```
## [1] -6531.295
```

```r
(ll_complex <-logLik(model2)[1])
```

```
## [1] -6515.607
```

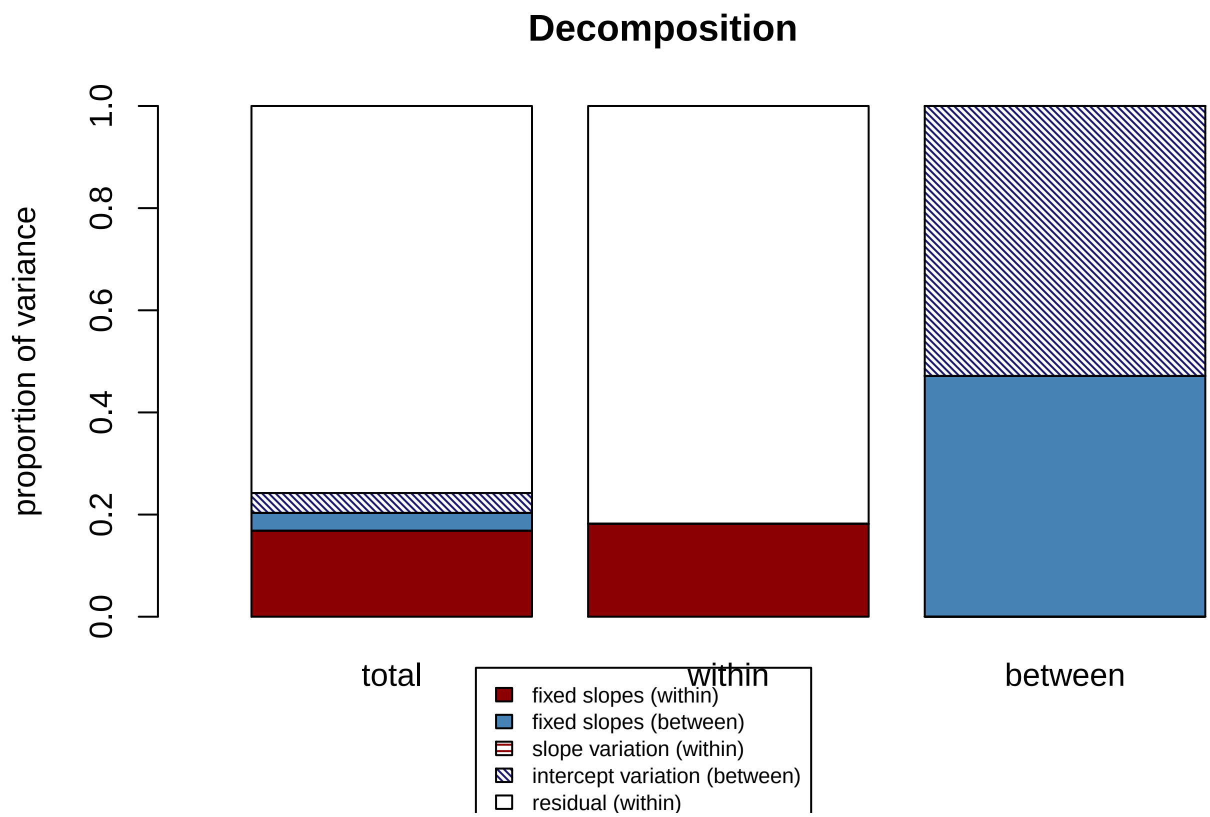
```r
(LR <- -2*ll_simple-(-2*ll_complex)) # 1df because only 1 parameter difference
```

```
## [1] 31.37435
```

```r
(pval_lr <- (pchisq(LR, df=1, lower.tail = FALSE)/2))
```

```
## [1] 1.063867e-08
```

```r
# * Proportion of variance explained ------------------------------------
## Use Rights & Sterba (2019)
r2mlm::r2mlm(model2)
```

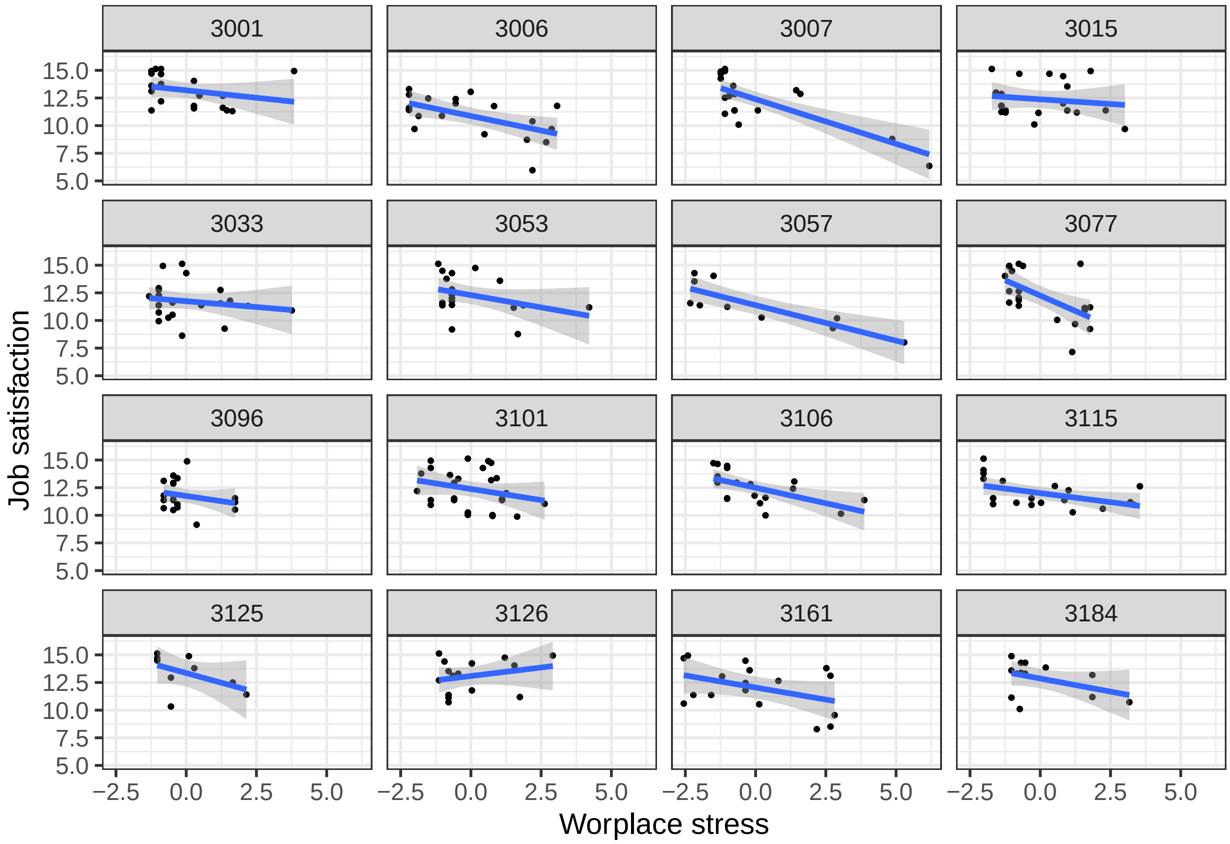



```
## $Decompositions
##                  total              within             between
## fixed, within    0.168335407006222  0.181754683645787  NA
## fixed, between   0.0348042999926375 NA                 0.471399815696785
## slope variation  0                  0                  NA
## mean variation   0.0390275065412548 NA                 0.528600184303215
## sigma2           0.757832786459886  0.818245316354213  NA
##
## $R2s
##     total              within             between
## f1  0.168335407006222  0.181754683645787  NA
## f2  0.0348042999926375 NA                 0.471399815696785
## v   0                  0                  NA
## m   0.0390275065412548 NA                 0.528600184303215
## f   0.20313970699886   NA                 NA
## fv  0.20313970699886   0.181754683645787  NA
## fvm 0.242167213540114  NA                 NA
```

```r
# Random coefficients model ---------------------------------------------
# Graphically explore slopes of a sample
set.seed(2022)
df4 %>%
  # randomly sample 16 schools
  filter(IDSCHOOL %in% sample(unique(IDSCHOOL), 16)) %>%
  ggplot(aes(x = T3WELS_CMC, y = T3JOBSA)) +
```

```r
  geom_point(size = 0.5) +
  geom_smooth(method = "lm") +
  facet_wrap(~IDSCHOOL) +
  labs(x="Worplace stress",
       y="Job satisfaction")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```r
# Model 3
model3 <- lmer(T3JOBSA ~ T3WELS_CM_GMC + T3WELS_CMC + (1 + T3WELS_CMC | IDSCHOOL),
               data = df4,
               control = lmerControl(optimizer = "bobyqa"))
# Summarize results
summary(model3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: T3JOBSA ~ T3WELS_CM_GMC + T3WELS_CMC + (1 + T3WELS_CMC | IDSCHOOL)
##    Data: df4
## Control: lmerControl(optimizer = "bobyqa")
##
## REML criterion at convergence: 13025.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3863 -0.6838 -0.0072  0.7276  3.3600
##
## Random effects:
```

```
##  Groups   Name        Variance Std.Dev. Corr
##  IDSCHOOL (Intercept) 0.134613 0.36690
##           T3WELS_CMC  0.003715 0.06095  0.92
##  Residual             2.594878 1.61086
## Number of obs: 3398, groups:  IDSCHOOL, 186
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   12.09370    0.03875  312.13
## T3WELS_CM_GMC -0.62521    0.06730   -9.29
## T3WELS_CMC    -0.45534    0.01736  -26.24
##
## Correlation of Fixed Effects:
##             (Intr) T3WELS_CM_
## T3WELS_CM_G 0.009
## T3WELS_CMC  0.169  0.052
```

```r
# Likelihood-based confidence intervals for fixed effects
confint(model3)
```

```
## Computing profile confidence intervals ...

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): unexpected decrease in
## profile: using minstep

## Warning in FUN(X[[i]], ...): non-monotonic profile for .sig01

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep
```

```
## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep
```

```
## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep
```

```
## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep
```

```
## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep
```

```
## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): Last two rows have
## identical or NA .zeta values: using minstep

## Warning in FUN(X[[i]], ...): non-monotonic profile for .sig02

## Warning in confint.thpr(pp, level = level, zeta = zeta): bad spline fit
## for .sig01: falling back to linear interpolation

## Warning in confint.thpr(pp, level = level, zeta = zeta): bad spline fit
## for .sig02: falling back to linear interpolation

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

##                   2.5 %     97.5 %
## .sig01         0.28652790  0.4434053
## .sig02         0.11522518  1.0000000
## .sig03         0.01032606  0.1224459
## .sigma         1.57143308  1.6313482
## (Intercept)   12.01783013 12.1697194
## T3WELS_CM_GMC -0.75799081 -0.4925897
## T3WELS_CMC    -0.48935498 -0.4209101
```

```
# * LRT Random effects --------------------------------------------------
anova(model2, model3)
```

```
## refitting model(s) with ML (instead of REML)

## Data: df4
## Models:
```

```
## model2: T3JOBSA ~ T3WELS_CMC + T3WELS_CM_GMC + (1 | IDSCHOOL)
## model3: T3JOBSA ~ T3WELS_CM_GMC + T3WELS_CMC + (1 + T3WELS_CMC | IDSCHOOL)
##        npar   AIC   BIC  logLik deviance  Chisq Df Pr(>Chisq)
## model2    5 13027 13057 -6508.3    13017
## model3    7 13025 13068 -6505.5    13011 5.6673  2     0.0588 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# * Variance-covariance matrix -------------------------------------------
VarCorr(model3)$IDSCHOOL
```

```
##             (Intercept) T3WELS_CMC
## (Intercept)  0.13461345 0.02063522
## T3WELS_CMC   0.02063522 0.00371502
## attr(,"stddev")
## (Intercept)  T3WELS_CMC
##  0.36689706  0.06095096
## attr(,"correlation")
##             (Intercept) T3WELS_CMC
## (Intercept)   1.0000000  0.9227503
## T3WELS_CMC    0.9227503  1.0000000
```

```r
# Plot of schools slopes vs schools intercepts
(myrandomeff <- ranef(model3, condVar = TRUE))
```

```
## $IDSCHOOL
##       (Intercept)    T3WELS_CMC
## 3001  0.409844586  0.0637460620
## 3002 -0.375977193 -0.0578190353
## 3003  0.136298050  0.0211553061
## 3004  0.392384888  0.0604185635
## 3005 -0.173580200 -0.0237142271
## 3006 -0.460765331 -0.0706396987
## 3007 -0.153789316 -0.0291079574
## 3008  0.168956725  0.0246972689
## 3009  0.056984796  0.0190549736
## 3010 -0.233282051 -0.0364513879
## 3011 -0.012897625 -0.0002068569
## 3012 -0.103213004 -0.0133029591
## 3014  0.157025703  0.0235563127
## 3015  0.183264902  0.0302369217
## 3016  0.003919042  0.0020325817
## 3017 -0.160032906 -0.0247783506
## 3018 -0.267671394 -0.0416947615
## 3019  0.025556065  0.0088925867
## 3020  0.052807502  0.0087790737
## 3021  0.103144405  0.0174387903
## 3022 -0.364142282 -0.0544992848
## 3023  0.622603441  0.0976796401
## 3024 -0.267189197 -0.0405463408
## 3025 -0.014573658 -0.0048527837
## 3026  0.019089572  0.0033997157
## 3027  0.282441300  0.0433157914
## 3028  0.362402506  0.0549109260
## 3029 -0.113759270 -0.0122089253
## 3030 -0.007145036 -0.0026856534
```

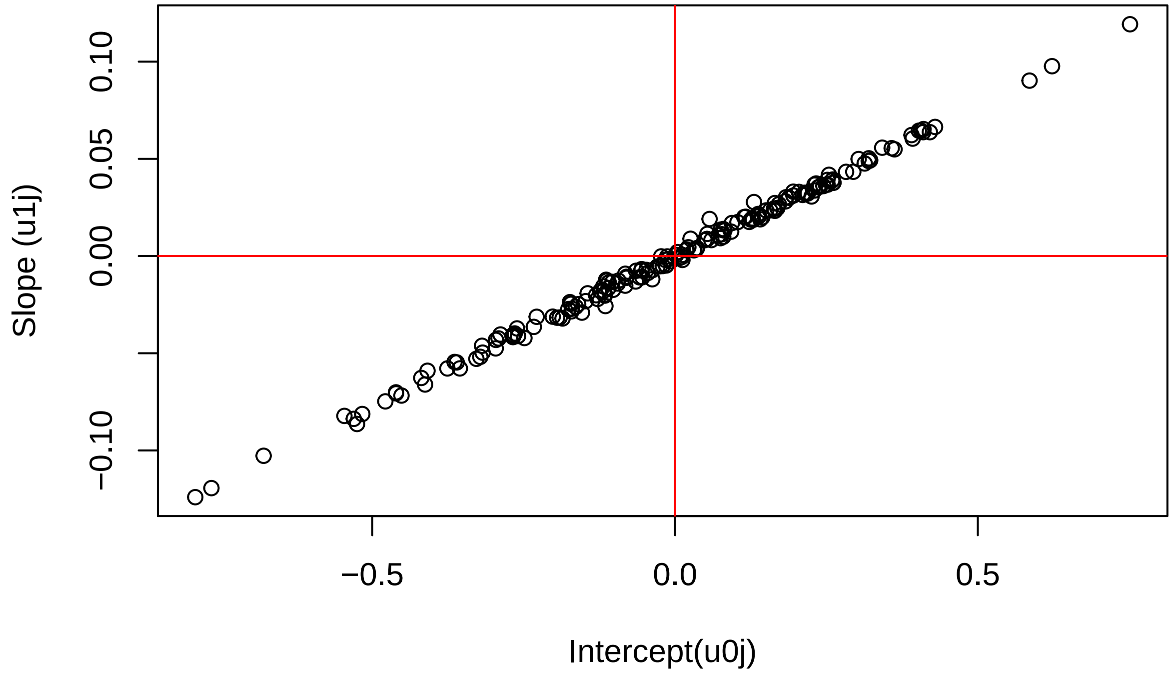
```
## 3031  0.001777426  0.0005836431
## 3032  0.075543842  0.0135979888
## 3033 -0.291428116 -0.0423950647
## 3034 -0.765634676 -0.1193790324
## 3035 -0.412830450 -0.0660824088
## 3036 -0.516504596 -0.0812671051
## 3037  0.219278479  0.0322245955
## 3038  0.233015783  0.0373264286
## 3039 -0.194760522 -0.0317106617
## 3040 -0.530610510 -0.0837605876
## 3041  0.076959827  0.0112181447
## 3042  0.420803011  0.0637322692
## 3043  0.357758144  0.0554890100
## 3044 -0.080424175 -0.0112132556
## 3045 -0.164016140 -0.0261466344
## 3046  0.116660845  0.0201987979
## 3047  0.140664949  0.0189088679
## 3048  0.018298504  0.0030265208
## 3050 -0.201994996 -0.0311738588
## 3051  0.261603711  0.0377120261
## 3052  0.195486746  0.0311315791
## 3053 -0.117840859 -0.0178513629
## 3054 -0.080131972 -0.0106722336
## 3055 -0.259264265 -0.0411446325
## 3056 -0.130339552 -0.0201552929
## 3057 -0.171008061 -0.0284409262
## 3058  0.130333822  0.0277696903
## 3059  0.053358303  0.0114102557
## 3060 -0.176238840 -0.0272858973
## 3061  0.402627833  0.0644880997
## 3062  0.125494459  0.0187080141
## 3063 -0.317779438 -0.0496895818
## 3064 -0.321610272 -0.0518584241
## 3065 -0.118552455 -0.0155963105
## 3066 -0.172036690 -0.0246060363
## 3067 -0.144379526 -0.0191742824
## 3068 -0.110088519 -0.0164424686
## 3069  0.429372560  0.0664186756
## 3070  0.230325957  0.0367240262
## 3071  0.138798897  0.0205726367
## 3072  0.092635764  0.0125420233
## 3073 -0.029051620 -0.0053774275
## 3074 -0.419070554 -0.0627182737
## 3075  0.007538616 -0.0009837932
## 3076 -0.055129008 -0.0067285356
## 3077 -0.185807719 -0.0321150252
## 3078  0.251848317  0.0367376814
## 3079 -0.260968627 -0.0372279595
## 3080 -0.037262188 -0.0072384087
## 3081  0.114271046  0.0200543354
## 3083  0.259439361  0.0388681756
## 3084  0.322268537  0.0492789186
## 3085  0.122552520  0.0175686814
## 3086 -0.147611009 -0.0232136003
```

```
## 3087 -0.228532097 -0.0312475127
## 3088  0.150527339  0.0234739635
## 3089  0.075028637  0.0092141136
## 3090  0.071777238  0.0107181487
## 3091  0.390483419  0.0622426236
## 3092  0.251553957  0.0391642420
## 3093 -0.044780024 -0.0086647418
## 3094  0.031582220  0.0031340348
## 3095  0.406249925  0.0645517149
## 3096 -0.460892671 -0.0701245466
## 3097 -0.094600370 -0.0141257901
## 3098 -0.037613411 -0.0119370617
## 3099 -0.064283184 -0.0075776113
## 3100  0.319955720  0.0502903338
## 3101  0.182409664  0.0281391412
## 3102 -0.102256703 -0.0173606713
## 3103 -0.355418153 -0.0578126419
## 3104  0.231969529  0.0370611594
## 3105 -0.679450809 -0.1027590545
## 3106  0.060002887  0.0082255264
## 3107 -0.093575897 -0.0126589218
## 3108 -0.296193136 -0.0475054698
## 3109  0.254158947  0.0418001228
## 3110  0.188272653  0.0299711271
## 3111  0.011988517  0.0005839491
## 3112  0.195311203  0.0331147666
## 3113 -0.525169798 -0.0864040339
## 3114 -0.022868831 -0.0001729091
## 3115  0.079733295  0.0139054807
## 3116 -0.058547273 -0.0109475477
## 3117 -0.360532875 -0.0547247536
## 3118 -0.170340737 -0.0270877070
## 3119 -0.053954706 -0.0108821262
## 3120  0.137218297  0.0216250037
## 3121  0.171446578  0.0266438916
## 3122 -0.115020132 -0.0257130125
## 3123 -0.020362759 -0.0051312106
## 3124  0.261353262  0.0377790440
## 3125  0.216052740  0.0324651591
## 3126  0.342082888  0.0557509203
## 3128  0.163497566  0.0242102384
## 3129  0.012180298 -0.0020144615
## 3130  0.164808313  0.0272463029
## 3131  0.048617683  0.0081416111
## 3132  0.021956881  0.0045042416
## 3133 -0.082236571 -0.0152278437
## 3134  0.244998355  0.0360125149
## 3135  0.079552461  0.0099239929
## 3136 -0.112389195 -0.0126534605
## 3137 -0.010980939 -0.0016224731
## 3138  0.249199121  0.0367470589
## 3139 -0.318716073 -0.0461675897
## 3140  0.319844426  0.0489055937
## 3141 -0.268443365 -0.0412061357
```

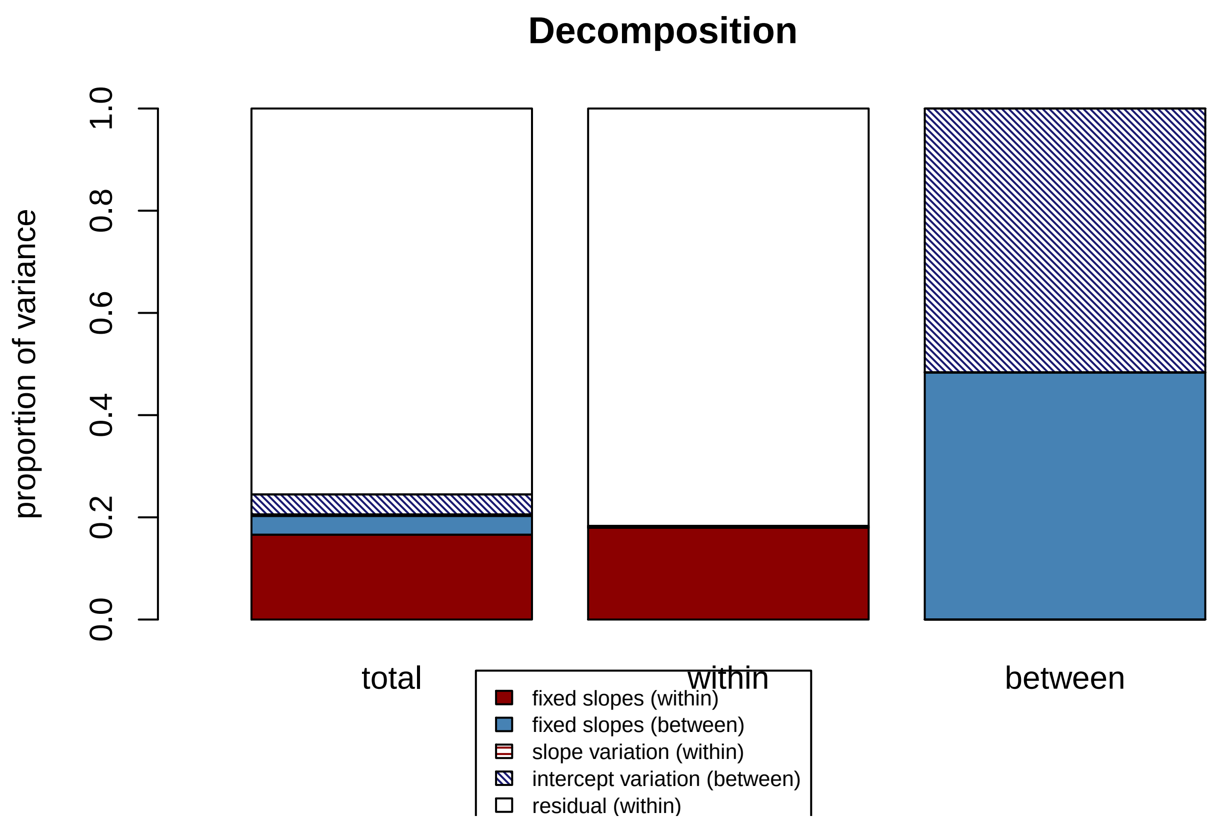
```
## 3142  0.128747934  0.0186155523
## 3143  0.213849358  0.0325345989
## 3144  0.751367296  0.1192508419
## 3145  0.585402335  0.0902456393
## 3146 -0.264765593 -0.0398254249
## 3147  0.163729661  0.0244994257
## 3148 -0.115761960 -0.0197449426
## 3149 -0.408819159 -0.0590172982
## 3150 -0.065032582 -0.0130782292
## 3151  0.294372162  0.0433636040
## 3152  0.035012171  0.0035123957
## 3153 -0.328087832 -0.0528710303
## 3154  0.093799992  0.0169714917
## 3155 -0.123356894 -0.0179661569
## 3156  0.031026695  0.0029599737
## 3157  0.225403555  0.0306550628
## 3158 -0.016222218 -0.0019601065
## 3159  0.164899708  0.0232650817
## 3160  0.229984318  0.0337120686
## 3161  0.313148864  0.0476261112
## 3162 -0.082136074 -0.0091020137
## 3163  0.410221529  0.0653246580
## 3164 -0.451757560 -0.0717607983
## 3165 -0.127832513 -0.0220602478
## 3166 -0.792336991 -0.1240430974
## 3167  0.236511487  0.0353848591
## 3169 -0.055721987 -0.0073993490
## 3170 -0.288280209 -0.0403632747
## 3171 -0.116073623 -0.0203507867
## 3172  0.204653604  0.0329843955
## 3173  0.239542500  0.0358633830
## 3174 -0.295758824 -0.0431426136
## 3175 -0.248746720 -0.0421668986
## 3176 -0.546040435 -0.0822408988
## 3177  0.216224897  0.0318770682
## 3178 -0.047150001 -0.0071543114
## 3179 -0.024946947 -0.0052092999
## 3180 -0.190585422 -0.0316139676
## 3181  0.144026917  0.0199557075
## 3182 -0.478427985 -0.0747709766
## 3183  0.303099944  0.0499078603
## 3184  0.127599283  0.0193757085
## 3185  0.052871937  0.0087343582
## 3186  0.081490723  0.0131224987
## 3187  0.210611654  0.0314409550
## 3189  0.260575037  0.0393661413
## 3190  0.036188041  0.0040890717
## 3191  0.011285310 -0.0004598365
## 3192 -0.109506619 -0.0136019822
##
## with conditional variances for "IDSCHOOL"
```

```
plot(myrandomeff[[1]],
     xlab = "Intercept(u0j)",
```

```r
      ylab = "Slope (u1j)")
abline(h = 0 , col = "red")
abline(v=0, col = 'red')
```



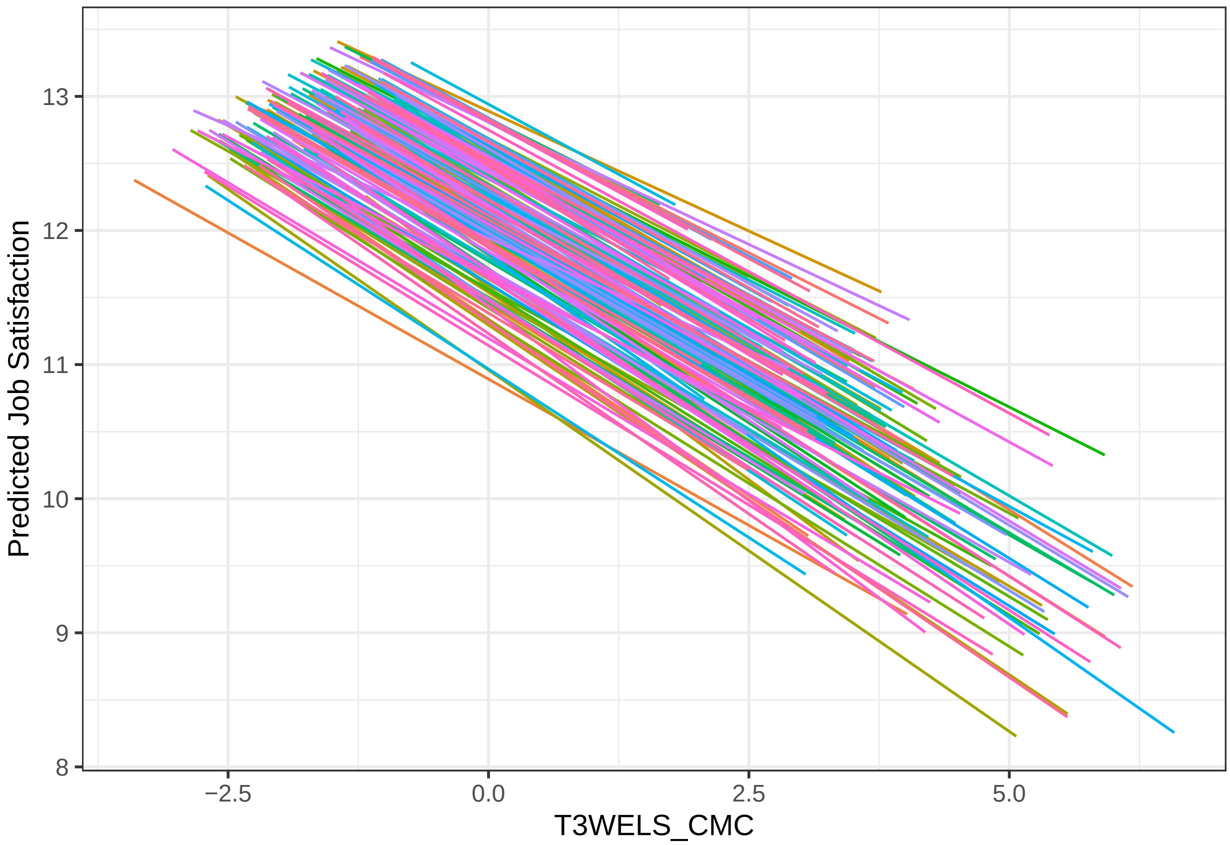

```r
# * Proportion of variance explained -------------------------------------
## Use Rights & Sterba (2019)
r2mlm::r2mlm(model3)
```




```
## $Decompositions
##                     total                 within                between
## fixed, within      0.166012446271133    0.179639394811518    NA
## fixed, between     0.0366823322642797   NA                   0.483570621016243
## slope variation    0.00297460464010367  0.00321877178099671  NA
## mean variation     0.0391749069269481   NA                   0.516429378983757
## sigma2             0.755155709897535    0.817141833407485    NA
##
## $R2s
##     total                 within                between
## f1  0.166012446271133    0.179639394811518    NA
## f2  0.0366823322642797   NA                   0.483570621016243
## v   0.00297460464010367  0.00321877178099671  NA
## m   0.0391749069269481   NA                   0.516429378983757
## f   0.202694778535413    NA                   NA
## fv  0.205669383175517    0.182858166592515    NA
## fvm 0.244844290102465    NA                   NA
```

```r
# Plotting random slopes
augment(model3, data = df4) %>% # augmented data (adding EB estimates)
  ggplot(aes(x = T3WELS_CMC, y = .fitted, color = factor(IDSCHOOL))) +
  geom_smooth(method = "lm", se = FALSE, size = 0.5) +
  labs(y = "Predicted Job Satisfaction") +
  guides(color = "none")
```

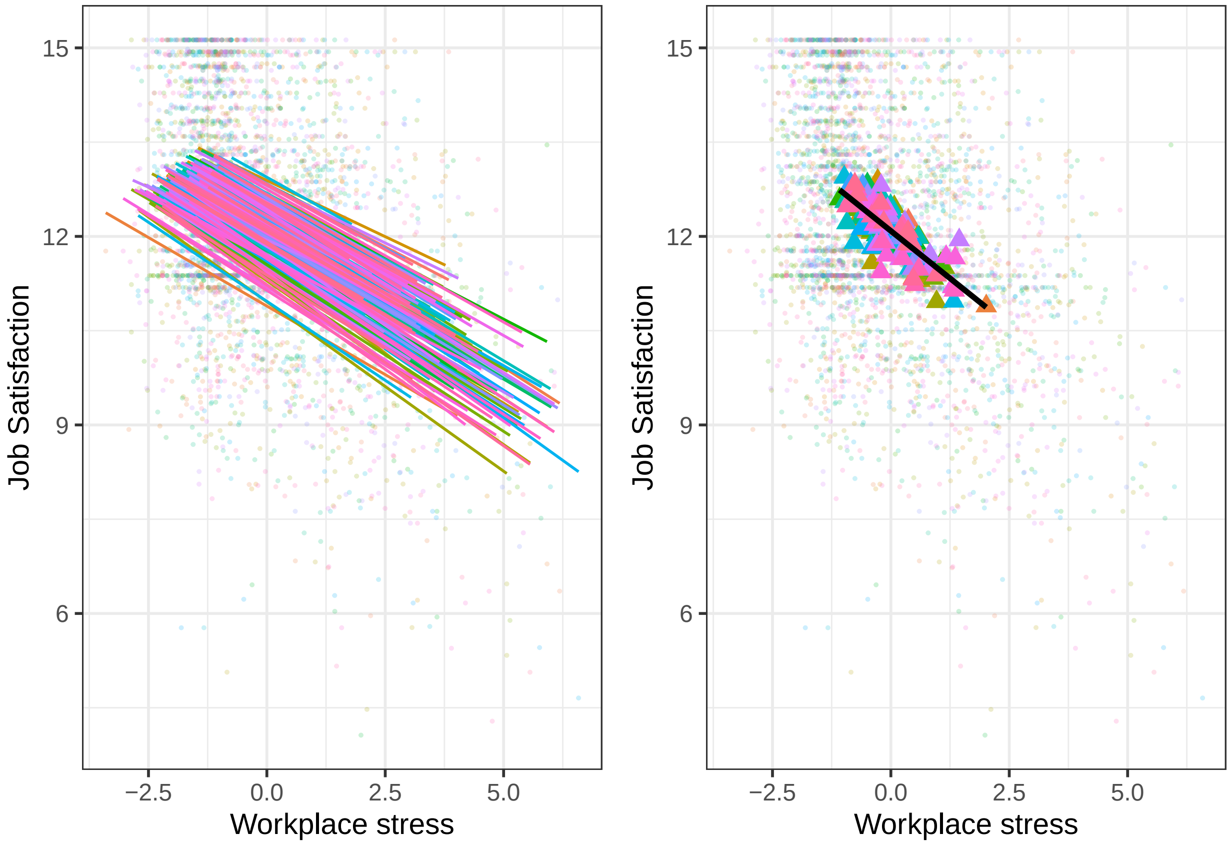
```
## `geom_smooth()` using formula 'y ~ x'
```




```r
# Create a common base graph
pbase <- augment(model3, data = df4) %>%
  ggplot(aes(x = T3WELS_CMC, y = T3JOBSA, color = factor(IDSCHOOL))) +
  # Add points
  geom_point(size = 0.2, alpha = 0.2) +
  labs(y = "Job Satisfaction") +
  # Suppress legend
  guides(color = "none")
# Lv-1 effect
p1 <- pbase +
  # Add within-cluster lines
  geom_smooth(aes(y = .fitted),
              method = "lm", se = FALSE, size = 0.5)+
  labs(x= "Workplace stress")
# Lv-2 effect
p2 <- pbase +
  # Add group means
  stat_summary(aes(x = T3WELS_CM_GMC, y = .fitted),
               fun = mean,
               geom = "point",
               shape = 17,
               # use triangles
               size = 2.5) +
  # Add between coefficient
  geom_smooth(aes(x = T3WELS_CM_GMC, y = .fitted),
              method = "lm", se = FALSE,
              color = "black") +
```

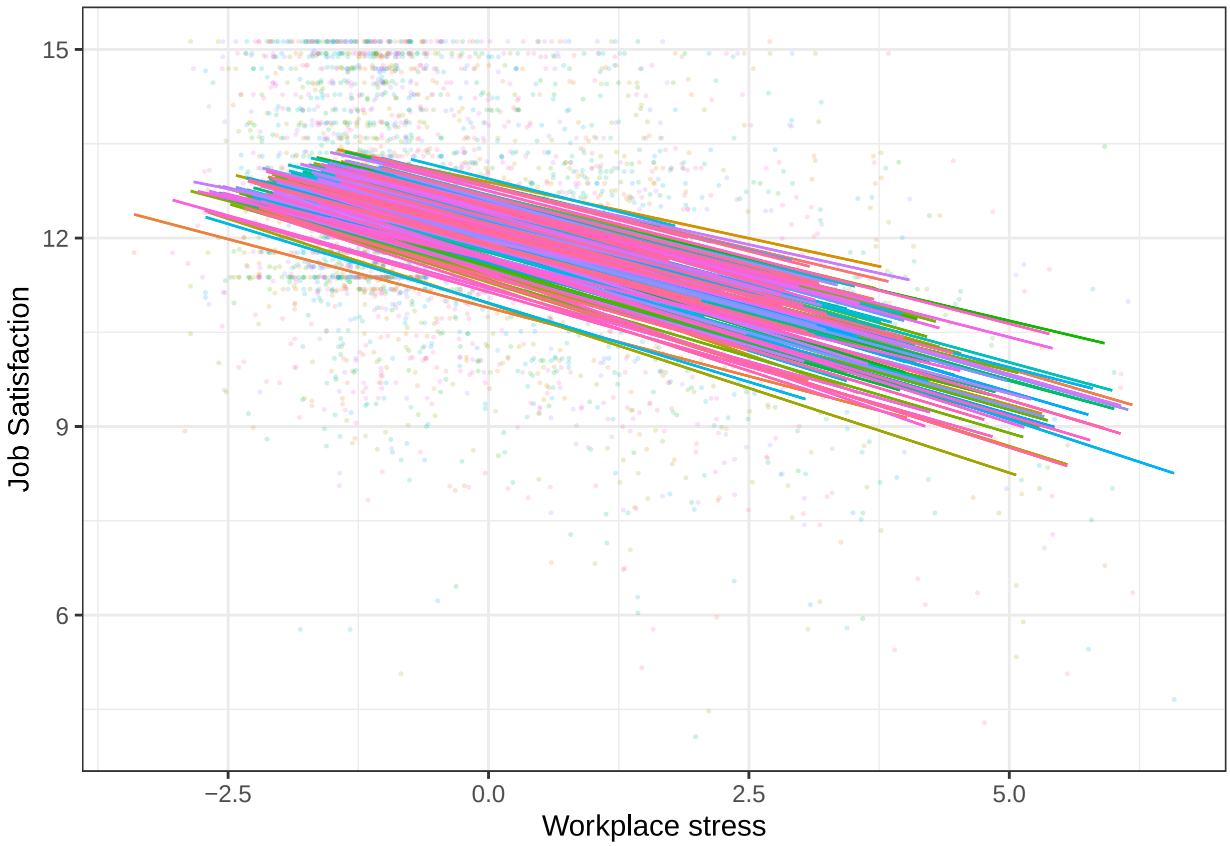
```r
  labs(x= "Workplace stress")
# Put the two graphs together (need the gridExtra package)
gridExtra::grid.arrange(p1, p2, ncol = 2)
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```




```r
# Print separate
p1
```

```
## `geom_smooth()` using formula 'y ~ x'
```


15
12
9
6
−2.5
0.0
2.5
5.0
Job Satisfaction
Workplace stress