# Example 3: General Linear Models with Multiple Fixed Effects of a Single Conceptual Predictor
***(complete syntax, data, and output available for STATA, R, and SAS electronically)***

The data for this example were selected from the 2012 General Social Survey dataset (and were also used for examples 1 and 2). The current example will use general linear models to predict a single quantitative outcome (annual income) in which multiple fixed effects are needed to describe a predictor's relationship to the outcome: for categorical predictors with more than two categories (3-category working class), for quantitative predictors with nonlinear effects (quadratic years of age or piecewise years of education), or for testing the assumption of a single linear slope for ordinal predictors (5-category happiness).

## STATA Syntax for Importing and Preparing Data for Analysis:

```
// Paste in the folder address where "GSS_Example.xlsx" is saved between " "
cd "\\Client\C:\Dropbox\24_PSQF6243\PSQF6243_Example3"

// IMPORT GSS_Example.xlsx data from working directory using exact file name
// To change all variable names to lowercase, remove "case(preserve")
clear // Clear before means close any open data
import excel "GSS_Example.xlsx", case(preserve) firstrow clear
// Clear after means re-import if it already exists (if need to start over)

// Label variables and apply value formats for variables used below
// label variable name    "name: Descriptive Variable Label"
label variable workclass "workclass: 1=Lower, 2=Middle, 3=Upper"
label variable age       "age: Years of Age"
label variable educ      "educ: Years of Education"
label variable happy     "happy: 5-Category Happy Rating"
label variable income    "income: Annual Income in 1000s"
```

## R Syntax for Importing and Preparing Data for Analysis
## (after loading packages *readxl*, *psych*, *supernova*, *multcomp*, *ppcor*, and *TeachingDemos*):

```
# Set working directory (to import and export files to)
# Paste in the folder address where "GSS_Example.xlsx" is saved in quotes
setwd("C:/Dropbox/24_PSQF6243/PSQF6243_Example3")

# Import GSS_Example.xlsx data from working directory -- path = file name
Example3 = read_excel(path="GSS_Example.xlsx", sheet="GSS_Example")
# Convert to data frame to use for analysis
Example3 = as.data.frame(Example3)
# Label variables used below (add descriptive titles) using comments instead
```

## Syntax and Output for Data Description:

```
display "STATA Descriptive Statistics for Quantitative and Ordinal Variables"
summarize income age educ happy
```

```
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
      income |        734    17.30287    13.79163       .245       68.6
         age |        734    42.06267    13.37838         18         75
        educ |        734    13.81199    2.909282          2         20
       happy |        734    3.555858    .8950446          1          5
```

```
print("R Descriptive Statistics for Quantitative or Ordinal Variables")
describe(x=Example3[ , c("income","age","educ","happy")])
```

```
       vars   n   mean     sd median    min  max  range   skew kurtosis    se
income    1 734 17.303 13.792 13.475  0.245 68.6 68.355  1.156    1.075 0.509
age       2 734 42.063 13.378 41.000 18.000 75.0 57.000  0.293   -0.769 0.494
educ      3 734 13.812  2.909 14.000  2.000 20.0 18.000 -0.230    0.777 0.107
happy     4 734  3.556  0.895  4.000  1.000  5.0  4.000 -0.641    0.713 0.033
```

```
display "SAS Descriptive Statistics for Categorical Variables"
tabulate workclass
  workclass |      Freq.     Percent        Cum.
------------+-----------------------------------
          1 |        436       59.40       59.40
          2 |        278       37.87       97.28
          3 |         20        2.72      100.00
------------+-----------------------------------
      Total |        734      100.00
```

We will need 2 slopes to represent the differences across these 3 categories.

```
tabulate happy
      happy |      Freq.     Percent        Cum.
------------+-----------------------------------
          1 |         26        3.54        3.54
          2 |         39        5.31        8.86
          3 |        256       34.88       43.73
          4 |        327       44.55       88.28
          5 |         86       11.72      100.00
------------+-----------------------------------
      Total |        734      100.00
```

We will need 4 slopes to represent the differences across these 5 categories.

```
print("R Descriptive Statistics for Categorical Variables")
prop.table(table(x=Example3$workclass,useNA="ifany"))

       1        2        3 
0.594005 0.378747 0.027248 

prop.table(table(x=Example3$happy,useNA="ifany"))

       1        2        3        4        5 
0.035422 0.053134 0.348774 0.445504 0.117166 
```

---

## Syntax to Create Indicator-Dummy-Coded Predictors—2 needed for 3 categories of *workclass*:

**Categorical variables with 3+ categories cannot be included directly as predictors in the model**, or else a single linear slope will be estimated to differentiate the total $C$ categories—this doesn't make any sense, especially for nominal predictor variables. Instead, **we need to create $C - 1$ new predictors to distinguish the predicted outcome for each of the $C$ categories**. The coding scheme we are using is "indicator-dummy-coding" where each category has a 1 for only a single predictor (that "activates" the predictor for that category).

```
// STATA code to create 2 new indicator-dummy-coded binary predictors
gen LvM=. // Make two new empty variables
gen LvU=.
replace LvM=0 if workclass==1 // Replace each for lower
replace LvU=0 if workclass==1
replace LvM=1 if workclass==2 // Replace each for middle
replace LvU=0 if workclass==2
replace LvM=0 if workclass==3 // Replace each for upper
replace LvU=1 if workclass==3
label variable LvM "LvM: Lower=0 v Middle=1 Class"
label variable LvU "LvU: Lower=0 v Upper=1 Class"
```

| Group ($N$ = 734) | LvM | LvU |
|---|---|---|
| 1. Lower ($n$ = 436) | 0 | 0 |
| 2. Middle ($n$ = 278) | **1** | 0 |
| 3. Upper ($n$ = 20) | 0 | **1** |

```
# R code to create indicator-dummy-coded binary predictors
Example3$LvM=NA; Example3$LvU=NA  # Make 2 new empty variables
Example3$LvM[which(Example3$workclass==1)]=0  # Replace each for lower
Example3$LvU[which(Example3$workclass==1)]=0
Example3$LvM[which(Example3$workclass==2)]=1  # Replace each for middle
Example3$LvU[which(Example3$workclass==2)]=0
Example3$LvM[which(Example3$workclass==3)]=0  # Replace each for upper
Example3$LvU[which(Example3$workclass==3)]=1
# LvM: Lower=0 vs Middle=1 Class
# LvU: Lower=0 vs Upper=1 Class
```

## Syntax and Output for 3-Category Working Class Predicting Income:

## Model including workclass via two indicator-dummy-coded predictors:

$$Income_i = \beta_0 + \beta_1(LvM_i) + \beta_2(LvU_i) + e_i$$

**Interpret $\beta_0$ = Intercept:**

**Interpret $\beta_1$ = Lower vs Middle slope:**

**Interpret $\beta_2$ = Lower vs Upper slope:**

```
display "STATA GLM Predicting Income from 2 New Binary Variables for workclass"
regress income c.LvM c.LvU, level(95)
```

```
      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(2, 731)       =     42.14
       Model |  14414.0265         2  7207.01325   Prob > F        =    0.0000
    Residual |  125009.205       731  171.011225   R-squared       =    0.1034
-------------+----------------------------------   Adj R-squared   =    0.1009
       Total |  139423.232       733  190.209048   Root MSE        =    13.077
```

**Mean Square Error/Residual**, the residual variance, is 171.01 after including 2 fixed slopes for *workclass* as a predictor (which accounted for 10.34% of the variance in income, as given by the model $R^2$ = .1034.). The *F*-test tells us this $R^2$ is significantly > 0, written as: ***F*(2, 731) = 42.14, MSE = 171.01, *p* < .001**.

```
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         LvM |   8.854267   1.003681     8.82   0.000     6.883826    10.82471  beta1
         LvU |   10.98471    2.99045     3.67   0.000     5.113816     16.8556  beta2
       _cons |   13.65004   .6262808    21.80   0.000     12.42052    14.87956  beta0
------------------------------------------------------------------------------
```

```
print("R GLM Predicting Income from 2 New Binary Variables for workclass")
ModelClass = lm(data=Example3, formula=income~1+LvM+LvU)
supernova(ModelClass)   # supernova prints sums of squares and residual variance
```

```
Analysis of Variance Table (Type III SS)
                                 SS  df         MS      F   PRE     p
 ----- --------------- | ---------- --- ---------- ------ ----- -----
 Model (error reduced) |  14414.026   2   7207.013 42.144 .1034 .0000
   LvM                 |  13308.783   1  13308.783 77.824 .0962 .0000
   LvU                 |   2307.432   1   2307.432 13.493 .0181 .0003
 Error (from model)    | 125009.205 731    171.011
 ----- --------------- | ---------- --- ---------- ------ ----- -----
 Total (empty model)   | 139423.232 733    190.209
```

In the table above, **PRE for the model is $R^2$**; PRE for each slope is its squared partial correlation (stay tuned). However, you can safely ignore the slope-specific rows and **refer only to the Model and Error rows**.

```
summary(ModelClass)    # summary prints fixed effects solution
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   13.650      0.626   21.80  < 2e-16  beta0
LvM            8.854      1.004    8.82  < 2e-16  beta1
LvU           10.985      2.990    3.67  0.00026  beta2

Residual standard error: 13.1 on 731 degrees of freedom
Multiple R-squared:  0.103,      Adjusted R-squared:  0.101
F-statistic: 42.1 on 2 and 731 DF,  p-value: <2e-16
```

LM prints this summary of the sums of squares table only (which is why I requested the table using supernova).

```
confint(ModelClass, level=.95) # confint for level% CI for fixed effects

              2.5 % 97.5 %
(Intercept) 12.4205 14.880
LvM          6.8838 10.825
LvU          5.1138 16.856
```

## Syntax and R Output to Compute Predicted Means per Category and Mean Differences:

Predicted Income: $\hat{y}_i = \beta_0(1) + \beta_1(LvM_i) + \beta_2(LvU_i)$

Lower Mean: $\hat{y}_L = \beta_0(1) + \beta_1(0) + \beta_2(0) = \beta_0$ ← fixed effect #1
Middle Mean: $\hat{y}_M = \beta_0(1) + \beta_1(1) + \beta_2(0) = \beta_0 + \beta_1$ ← linear combination
Upper Mean: $\hat{y}_U = \beta_0(1) + \beta_1(0) + \beta_2(1) = \beta_0 + \beta_2$ ← linear combination

Difference for Lower vs Middle: $(\beta_0 + \beta_1) - (\beta_0) = \beta_1$ ← fixed effect #2
Difference for Lower vs. Upper: $(\beta_0 + \beta_2) - (\beta_0) = \beta_2$ ← fixed effect #3
Difference for Middle vs Upper: $(\beta_0 + \beta_2) - (\beta_0 + \beta_1) = \beta_2 - \beta_1$ ← linear combination

```
// STATA code to ask for predicted income per category and category differences
   lincom _cons*1 + c.LvM*0  + c.LvU*0  // Pred Income: Lower (already in model)
   lincom _cons*1 + c.LvM*1  + c.LvU*0  // Pred Income: Middle
   lincom _cons*1 + c.LvM*0  + c.LvU*1  // Pred Income: Upper
   lincom           c.LvM*1  + c.LvU*0  // Lower vs Middle Diff (already in model)
   lincom           c.LvM*0  + c.LvU*1  // Lower vs Upper Diff  (already in model)
   lincom           c.LvM*-1 + c.LvU*1  // Middle vs Upper Diff

print("R code to ask for predicted income per category and category differences")
print("In number lists below, values are multiplier for each fixed effect IN ORDER")
PredClass = glht(model=ModelClass, linfct=rbind(
  "Pred Income: Lower"    = c(1, 0, 0),  # already in model
  "Pred Income: Middle"   = c(1, 1, 0),
  "Pred Income: Upper"    = c(1, 0, 1),
  "Lower vs Middle Diff" = c(0, 1, 0),  # already in model
  "Lower vs Upper Diff"   = c(0, 0, 1),  # already in model
  "Middle vs Upper Diff" = c(0,-1, 1)))
print("Save glht linear combination results with unadjusted p-values and 95% CIs")
SavePredClass = summary(PredClass, test=adjusted("none")); SavePredClass
confint(PredClass, level=.95, calpha=univariate_calpha())

Linear Hypotheses:
                          Estimate Std. Error t value Pr(>|t|)
Pred Income: Lower == 0     13.650      0.626   21.80  < 2e-16  beta0
Pred Income: Middle == 0    22.504      0.784   28.69  < 2e-16  beta0 + beta1
Pred Income: Upper == 0     24.635      2.924    8.42  2.2e-16  beta0 + beta2
Lower vs Middle Diff == 0    8.854      1.004    8.82  < 2e-16  beta1
Lower vs Upper Diff == 0    10.985      2.990    3.67  0.00026  beta2
Middle vs Upper Diff == 0    2.130      3.027    0.70  0.48184  beta2 - beta1
(Adjusted p values reported -- none method)

Quantile = 1.963
95% confidence level

Linear Hypotheses:
                          Estimate lwr    upr
Pred Income: Lower == 0   13.650   12.421 14.880
Pred Income: Middle == 0  22.504   20.965 24.044
Pred Income: Upper == 0   24.635   18.894 30.375
Lower vs Middle Diff == 0  8.854    6.884 10.825
Lower vs Upper Diff == 0  10.985    5.114 16.856
Middle vs Upper Diff == 0  2.130   -3.813  8.074
```

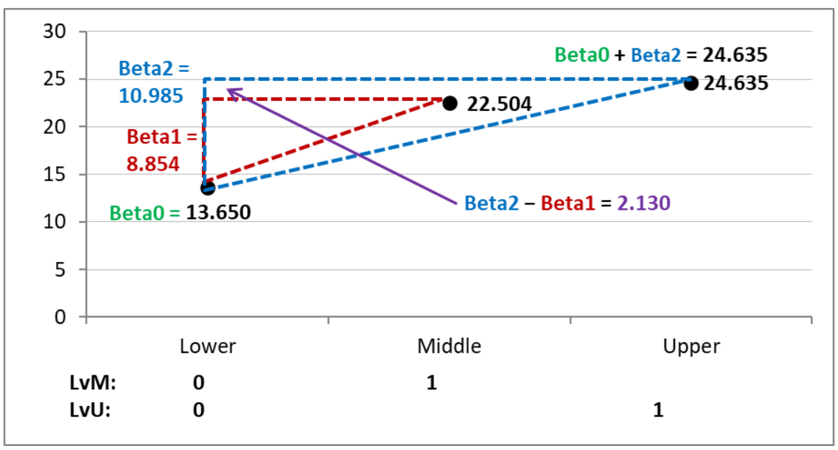
**Syntax and R Output to Compute Cohen's $d$ and Partial $r$ Effect Sizes for Mean Differences:**

$$d = \frac{2t}{\sqrt{DF_{den}}}, \text{ partial } r = \frac{t}{\sqrt{t^2 + DF_{den}}}$$

```
// STATA code to compute effect sizes from stored results per lincom
lincom c.LvM*1 + c.LvU*0  // Low vs Mid Diff
   display "PartialD= " (2*(r(estimate)/r(se)))/sqrt(r(df))
   display "PartialR= " (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))
lincom c.LvM*0 + c.LvU*1  // Low vs Upp Diff
   display "PartialD= " (2*(r(estimate)/r(se)))/sqrt(r(df))
   display "PartialR= " (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))
lincom c.LvM*-1 + c.LvU*1 // Mid vs Upp Diff
   display "PartialD= " (2*(r(estimate)/r(se)))/sqrt(r(df))
   display "PartialR= " (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))

# R code to compute effect sizes from stored model and GLHT results
PredClassPartialD=(2*SavePredClass$test$tstat)/sqrt(ModelClass$df.residual)
PredClassPartialR=SavePredClass$test$tstat/
             sqrt(SavePredClass$test$tstat^2+ModelClass$df.residual)
# Concatenate effect sizes to results table for mean differences
data.frame(Estimate=SavePredClass$test$coefficients,
           SE=SavePredClass$test$sigma,
           pvalue=SavePredClass$test$pvalues,
           PartialD=PredlassPartialD, PartialR=PredClassPartialR)
```

| | Estimate | SE | pvalue | PartialD | PartialR |
|---|---|---|---|---|---|
| Pred Income: Lower | 13.6500 | 0.62628 | 0.0000e+00 | 1.612264 | 0.627602 |
| Pred Income: Middle | 22.5043 | 0.78431 | 0.0000e+00 | 2.122497 | 0.727797 |
| Pred Income: Upper | 24.6348 | 2.92413 | 2.2204e-16 | 0.623192 | 0.297489 |
| Lower vs Middle Diff | 8.8543 | 1.00368 | 0.0000e+00 | **0.652572** | 0.310191 |
| Lower vs Upper Diff | 10.9847 | 2.99045 | 2.5695e-04 | **0.271721** | 0.134624 |
| Middle vs Upper Diff | 2.1304 | 3.02749 | 4.8184e-01 | **0.052054** | 0.026018 |

Btw, effect sizes for predicted outcomes are not meaningful (but the first 3 rows were already included in the dataset of saved estimates).




In your results sections, make sure to state what software and function you used (and which version), along with any extra functions (i.e., in separate R packages).

**Example Results Section for Income Mean Differences by Working Class:**

We used a general linear model (i.e., analysis of variance) to examine the extent to which annual income in thousands of dollars ($M$ = 17.30, $SD$ = 13.79) could be predicted from three categories of self-reported working class membership (lower = 59.40%, middle = 37.87%, and upper = 2.72%). We created two contrasts to distinguish the three classes, in which lower-class respondents served as the reference group to be compared separately to middle-class and upper-class respondents. Cohen's $d$ standardized means differences were then computed from the $t$ test-statistics to index effect size per slope. We found that class membership significantly predicted annual income, $F(2, 731)$ = 42.14, MSE = 171.01, $p < .001$, $R^2$ = .10. Relative to lower-class respondents, annual income was significantly higher for both middle-class respondents (difference = 8.85, SE = 1.00, $d$ = 0.65) and upper-class respondents (difference = 10.98, SE = 2.99, $d$ = 0.27). However, upper-class respondents did not differ significantly from middle-class respondents (difference = 2.13, SE = 3.03, $d$ = 0.05).

### Syntax to Center Age at 18 years (minimum of sample):

```
// STATA code to create 1 new age variable centered at 18 (minimum in sample)
gen age18=age-18
label variable age18 "age18: Age (0=18 years)"

# R code to make new age variable centered at 18 (minimum in sample)
Example3$age18=Example3$age-18 # age18: Age (0=18 years)
```

### Syntax and STATA Output for Linear Age (Centered at 18 Years) Predicting Income:

$$Income_i = \beta_0 + \beta_1(Age_i - 18) + e_i$$

**Interpret $\beta_0$ = Intercept:**

**Interpret $\beta_1$ = Linear age slope:**

The syntax shown next will also request the predicted income for example ages 30, 50, and 70.

```
display "STATA GLM Predicting Income from Linear Centered Age (0=18)"
regress income c.age18, level(95)

      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(1, 732)       =     30.52
       Model |  5580.74243         1  5580.74243   Prob > F        =    0.0000
    Residual |  133842.489       732  182.844931   R-squared       =    0.0400
-------------+----------------------------------   Adj R-squared   =    0.0387
       Total |  139423.232       733  190.209048   Root MSE        =    13.522
```

**Mean Square Error/Residual**, the residual variance, is 182.84 after including a fixed linear slope of age (which accounted for 4.00% of the variance in income, as given by the model $R^2$ = .0400).
The $F$-test tells us this $R^2$ is significantly > 0, written as: **$F(1, 732) = 30.52$, MSE = 182.84, $p < .001$**.

```
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       age18 |   .2062483   .0373324     5.52   0.000      .132957    .2795397  beta1
       _cons |   12.33999   1.027658    12.01   0.000     10.32248     14.3575  beta0
------------------------------------------------------------------------------
```

If you square the $t$ test-statistic for the age18 slope, $t^2 = F = 30.52$. So given only one fixed slope in a model, the $F$-test of the model is equivalent to the $t$-test of that slope (which is why we ignored $F$ in Example 2).

```
// Ask for predicted income for example ages
   lincom _cons*1 + c.age18*12  // Pred Income: Age 30 (age18=12)
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   14.81497   .6722375    22.04   0.000     13.49523    16.13471
------------------------------------------------------------------------------

   lincom _cons*1 + c.age18*32  // Pred Income: Age 50 (age18=32)
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   18.93994   .5804419    32.63   0.000     17.80041    20.07947
------------------------------------------------------------------------------

   lincom _cons*1 + c.age18*52  // Pred Income: Age 70 (age18=52)
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |    23.0649   1.156239    19.95   0.000     20.79496    25.33484
------------------------------------------------------------------------------
```

```
print("R GLM Predicting Income from Linear Centered Age")
ModelLinAge = lm(data=Example3, formula=income~1+age18)
supernova(ModelLinAge)  # supernova prints sums of squares and residual variance
summary(ModelLinAge)    # summary prints fixed effects solution
confint(ModelLinAge, level=.95) # confint to print level% CI for fixed effects

print("R Ask for predicted income for example ages")
print("In number lists below, values are multiplier for each fixed effect IN ORDER")
PredLinAge = glht(model=ModelLinAge, linfct=rbind(
  "Pred Income: Age 30 (age18=12)" = c(1,12),
  "Pred Income: Age 50 (age18=32)" = c(1,32),
  "Pred Income: Age 70 (age18=52)" = c(1,52)))
print("Print glht linear combination results with unadjusted p-values and 95% CIs")
summary(PredLinAge, test=adjusted("none"))
confint(PredLinAge, level=.95, calpha=univariate_calpha())
```

---

## Syntax and R Output Adding Quadratic Age (Centered at 18 Years) Predicting Income:

$$Income_i = \beta_0 + \beta_1(Age_i - 18) + \beta_2(Age_i - 18)^2 + e_i$$

**Interpret $\beta_0$ = Intercept:**

**Interpret $\beta_1$ = Linear age slope:**

**Interpret $\beta_2$ = Quadratic age slope:**

**Interpret $R^2$ two different ways:**

**The $R^2$ went from .040 to .114, an increase of .074. Do we know if the $R^2$ increased significantly relative to the linear age model?**

```
display "STATA GLM Predicting Income from Linear+Quadratic Centered Age (0=18)"
regress income c.age18 c.age18#c.age18, level(95) // Hashtag multiplies predictors

print("R GLM Predicting Income from Linear+Quadratic Centered Age")
ModelQuadAge = lm(data=Example3, formula=income~1+age18+I(age18^2)) # I(x^2) squares x
supernova(ModelQuadAge)  # supernova prints sums of squares and residual variance
```

```
Analysis of Variance Table (Type III SS)
                                    SS  df        MS      F   PRE     p
 ---------- --------------- | ---------- --- --------- ------ ----- -----
      Model (error reduced) |  15885.462   2  7942.731 46.999 .1139 .0000
      age18                 |  13856.343   1 13856.343 81.991 .1009 .0000
 I(age18^2)                 |  10304.719   1 10304.719 60.975 .0770 .0000
      Error (from model)    | 123537.770 731   168.998
 ---------- --------------- | ---------- --- --------- ------ ----- -----
      Total (empty model)   | 139423.232 733   190.209
```

**Mean Square Error/Residual**, the residual variance, is now 169.00 from the two effects of age (which accounted for 11.39% of the variance in income, as given by the model $R^2$ = .1139).
The *F*-test says this $R^2$ is significantly > 0, written as: ***F*(2, 731) = 47.00, MSE = 169.00, *p* < .001**.

```
summary(ModelQuadAge)    # summary prints fixed effects solution
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.6766     1.5835    1.69    0.091  beta0
age18         1.2231     0.1351    9.05   <2e-16  beta1
I(age18^2)   -0.0195     0.0025   -7.81    2e-14  beta2
```

```
Residual standard error: 13 on 731 degrees of freedom
Multiple R-squared:  0.114,      Adjusted R-squared:  0.112
F-statistic:   47 on 2 and 731 DF,  p-value: <2e-16
```

```
confint(ModelQuadAge, level=.95) # confint to print level% CI for fixed effects

                2.5 %    97.5 %
(Intercept) -0.432210  5.785405
age18        0.957901  1.488260
I(age18^2)  -0.024449 -0.014625
```

The syntax shown next will also request not only the predicted outcome for example ages 30, 50, and 70, but also the predicted instantaneous linear slopes at those ages too:

$$\widehat{Income_i} = \beta_0 + \beta_1(Age_i - 18) + \beta_2(Age_i - 18)^2$$

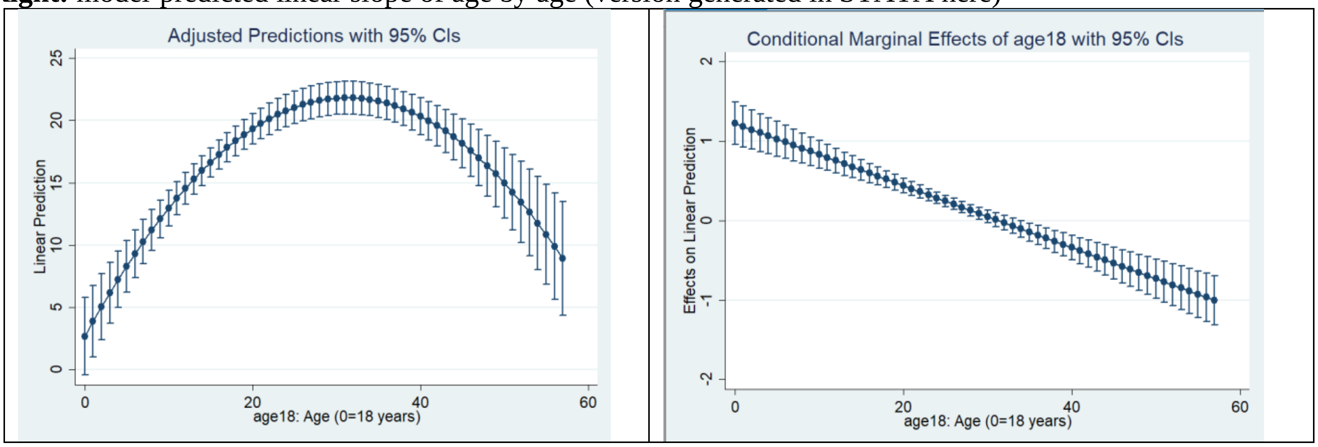
$$Linear\ Age\ Slope = \beta_1 + 2\beta_2(Age_i - 18)$$

```
// STATA Ask for predicted income for example ages
   lincom _cons*1 + c.age18*12 + c.age18#c.age18*144  // Pred Income: Age 30 (age18=12)
   lincom _cons*1 + c.age18*32 + c.age18#c.age18*1024 // Pred Income: Age 50 (age18=32)
   lincom _cons*1 + c.age18*52 + c.age18#c.age18*2704 // Pred Income: Age 70 (age18=52)

// STATA Linear age slope changes by 2*quadratic coefficient, so multiply age*2
   lincom  c.age18*1 + c.age18#c.age18*24  // Pred Linear Age Slope: Age 30 (age18=12)
   lincom  c.age18*1 + c.age18#c.age18*64  // Pred Linear Age Slope: Age 50 (age18=32)
   lincom  c.age18*1 + c.age18#c.age18*104 // Pred Linear Age Slope: Age 70 (age18=52)

print("R Ask for predicted income and predicted linear age slopes for example ages")
print("In number lists below, values are multiplier for each fixed effect IN ORDER")
PredQuadAge = glht(model=ModelQuadAge, linfct=rbind(
  "Pred Income: Age 30 (age18=12)" = c(1,12, 144),
  "Pred Income: Age 50 (age18=32)" = c(1,32,1024),
  "Pred Income: Age 70 (age18=52)" = c(1,52,2704),
  "Pred Linear Age Slope: Age 30 (age18=12)" = c(0,1, 24),
  "Pred Linear Age Slope: Age 50 (age18=32)" = c(0,1, 64),
  "Pred Linear Age Slope: Age 70 (age18=52)" = c(0,1,104)))
print("Print glht linear combination results with unadjusted p-values and 95% CIs")
summary(PredQuadAge, test=adjusted("none"))
confint(PredQuadAge, level=.95, calpha=univariate_calpha())
```

```
Linear Hypotheses:
                                             Estimate Std. Error t value Pr(>|t|)
Pred Income: Age 30 (age18=12) == 0           14.5402     0.6472   22.46  < 2e-16
Pred Income: Age 50 (age18=32) == 0           21.8091     0.6681   32.64  < 2e-16
Pred Income: Age 70 (age18=52) == 0           13.4482     1.6590    8.11  2.2e-15
Pred Linear Age Slope: Age 30 (age18=12) == 0  0.7542     0.0788    9.57  < 2e-16
Pred Linear Age Slope: Age 50 (age18=32) == 0 -0.0273     0.0467   -0.58     0.56
Pred Linear Age Slope: Age 70 (age18=52) == 0 -0.8088     0.1349   -6.00  3.1e-09
(Adjusted p values reported -- none method)

Quantile = 1.963
95% confidence level

Linear Hypotheses:
                                             Estimate lwr     upr
Pred Income: Age 30 (age18=12) == 0           14.5402  13.2695 15.8109
Pred Income: Age 50 (age18=32) == 0           21.8091  20.4974 23.1208
Pred Income: Age 70 (age18=52) == 0           13.4482  10.1912 16.7052
Pred Linear Age Slope: Age 30 (age18=12) == 0  0.7542   0.5995  0.9089
Pred Linear Age Slope: Age 50 (age18=32) == 0 -0.0273  -0.1190  0.0644
Pred Linear Age Slope: Age 70 (age18=52) == 0 -0.8088  -1.0735 -0.5440
```

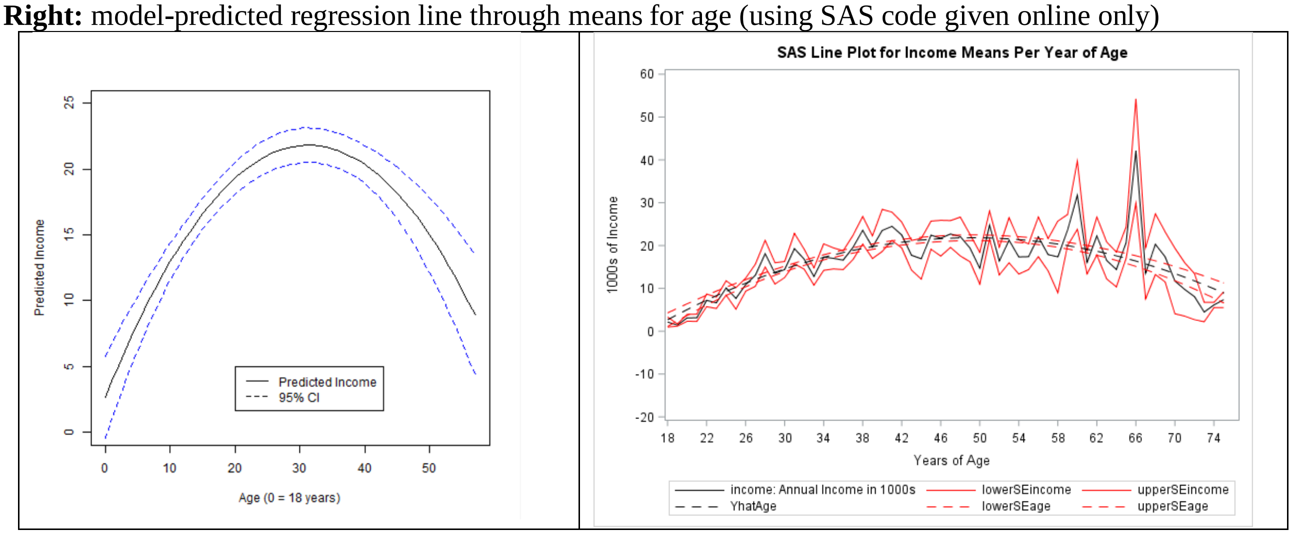
```
// STATA Get predicted values using margins more efficiently and plotting them
// quietly means don't print that output, predictor=(from(by)to)
   quietly margins, at(c.age18=(0(1)57)) // Real ages 18 to 75 (min and max)
   marginsplot, xdimension(age18) name(predicted_age, replace)
   graph export "STATA Predicted Income by Age Plot.png", replace

// STATA Get instantaneous linear age slopes to show effect of quadratic age slope
// dydx in margins provides linear age slopes at each value of age
   quietly margins, at(c.age18=(0(1)57)) dydx(c.age18)
   marginsplot, xdimension(age18) name(predicted_linear, replace)
   graph export "STATA Predicted Linear Slopes by Age Plot.png", replace
```

**Left:** model-predicted regression line with 95% CI (version generated in STATA here)
**Right:** model-predicted linear slope of age by age (version generated in STATA here)




```
# R Generating predicted values using predict more efficiently and plotting them
PredAge = data.frame(age18=seq(from=0, to=57, by=1)) # Real ages 18 to 75 (min and max)
PredAge = predict(object=ModelQuadAge, newdata=PredAge, se.fit=TRUE, interval="confidence")
PredAge = as.data.frame(PredAge) # Need to put x variable back in next
PredAge = cbind(PredAge, data.frame(age18=seq(from=0, to=57, by=1)))

png(file "R Predicted Income by Quadratic Age Plot.png")  # open file
plot(y=PredAge$fit.fit, x=PredAge$age18, ylim=c(0,25), xlim=c(0,57),
     lty=1, type="l", ylab="Predicted Income", xlab="Age (0 = 18 years)")
lines(y=PredAge$fit.upr, x=PredAge$age18, lty=2, col="blue1") # Upper CI
lines(y=PredAge$fit.lwr, x=PredAge$age18, lty=2, col="blue1") # Lower CI
legend(x=20, y=5, legend=c("Predicted Income", "95% CI"), lty=1:2); dev.off()  # close file
```

**Left:** model-predicted regression line through scatterplot (version generated in R here)
**Right:** model-predicted regression line through means for age (using SAS code given online only)



We forgo requesting standardized slopes for this model given the ambiguity of how to interpret them for models with interactions… $R^2$ is a sufficiently useful effect size to describe the overall effect (trend) of age here.

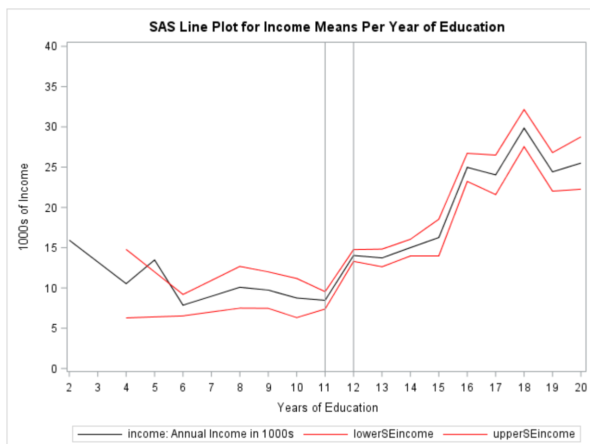
**<u>Example Results Section for the Linear and Quadratic Age Slopes:</u>**
We used a general linear model (i.e., linear regression) to examine the extent to which annual income in thousands of dollars (*M* = 17.30, *SD* = 13.79) could be predicted from years of age (*M* = 42.06, *SD* = 13.38, range = 18 to 75). We first examined the means of income by age to identify plausible types of nonlinear associations. Given the apparent curvilinear trend (in which age appeared positively associated with income until middle age, after which it appeared negatively associated instead), we fit a model including fixed linear and quadratic slopes for age (in which age was centered such that 0 = 18 years, the minimum age in the sample). The quadratic age model captured a significant amount of variance in annual income, $F(2, 731) = 47.00$, MSE = 169.00, $p < .001$, $R^2 = .114$. The quadratic age model was also a significant improvement over a linear age model, as indicated by the significant fixed slope for the quadratic effect of age. The model fixed effects can be interpreted as follows. The fixed intercept indicated that at age 18, annual income was predicted to be 2.676 thousand dollars (SE = 1.584) and was expected to be significantly greater by 1.223 thousand dollars per year of age (i.e., the instantaneous linear slope for age at age 18; SE = 0.135, $p < .001$). The linear age slope at age 18 was predicted to become significantly more negative per year of age by twice the quadratic coefficient of −0.020 (SE = 0.002, $p < .001$). As given by the quantity (−1*linear slope) / (2*quadratic slope) + 18, the age of maximum predicted personal income was 48.575 (i.e., the age at which the linear age slope is predicted to be 0). For example, the linear effect of age as evaluated at age 30 was significantly positive (Est = 0.754, SE = 0.079), the linear effect of age as evaluated at age 50 was nonsignificantly negative (Est = −0.027, SE = 0.047), and the linear effect of age as evaluated at age 70 was significantly negative (Est = −0.809, SE = 0.135).

---

## Syntax to Create 3 Predictors for Piecewise Linear Slopes for Education:

The idea is to represent the 3 different sections of education using 3 different predictors, that way the slope for each section is captured separately.

| Years Educ (x) | lessHS: Slope if x <12 | gradHS: HS Grad? (0=no, 1=yes) | overHS: Slope if x >12 |
|---|---|---|---|
| 9 | -2 | 0 | 0 |
| 10 | -1 | 0 | 0 |
| 11 (int) | 0 | 0 | 0 |
| 12 | 0 | 1 | 0 |
| 13 | 0 | 1 | 1 |
| 14 | 0 | 1 | 2 |
| 15 | 0 | 1 | 3 |
| 16 | 0 | 1 | 4 |
| 17 | 0 | 1 | 5 |
| 18 | 0 | 1 | 6 |




```
// STATA code to create 3 new predictor variables for sections of education
gen lessHS=. // Make 3 new empty variables
gen gradHS=.
gen overHS=.
// Replace each for educ less than 12
replace lessHS=educ-11 if educ <  12
replace gradHS=0       if educ <  12
replace overHS=0       if educ <  12
// Replace each for educ greater or equal to 12
replace lessHS=0       if educ >= 12
replace gradHS=1       if educ >= 12
replace overHS=educ-12 if educ >= 12
// Label variables
label variable lessHS "lessHS: Slope for Years Ed Less Than High School"
label variable gradHS "gradHS: Acute Bump for Graduating High School"
label variable overHS "overHS: Slope for Years Ed After High School"
```

```
# R code to make to make 3 new variables for sections of education
# Make 3 new empty variables
Example3$lessHS=NA; Example3$gradHS=NA; Example3$overHS=NA
# Replace each for educ less than 12
Example3$lessHS[which(Example3$educ<12)]=Example3$educ[which(Example3$educ<12)]-11
Example3$gradHS[which(Example3$educ<12)]=0
Example3$overHS[which(Example3$educ<12)]=0
# Replace each for educ greater or equal to 12
Example3$lessHS[which(Example3$educ>=12)]=0
Example3$gradHS[which(Example3$educ>=12)]=1
Example3$overHS[which(Example3$educ>=12)]=Example3$educ[which(Example3$educ>=12)]-12
```

## Syntax and STATA Output for Piecewise Linear Slopes of Education Predicting Income:

$$Income_i = \beta_0 + \beta_1(LessHS_i) + \beta_2(GradHS_i) + \beta_3(OverHS_i) + e_i$$

**Interpret $\beta_0$ = Intercept:**

**Interpret $\beta_1$ = LessHS slope:**

**Interpret $\beta_2$ = GradHS slope:**

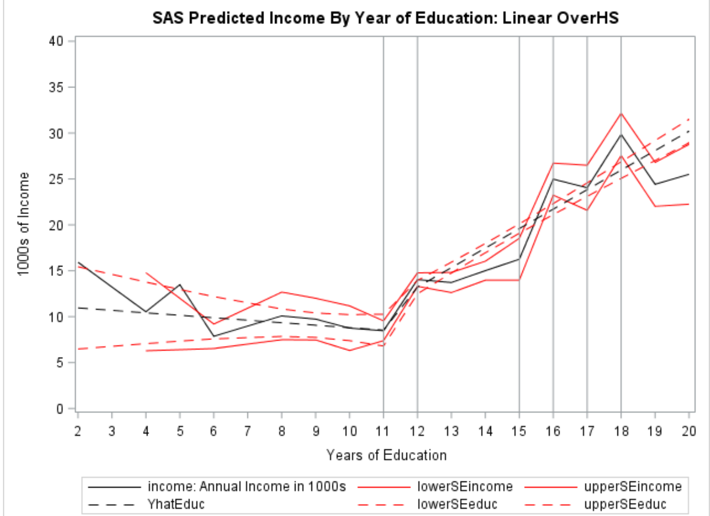
**Interpret $\beta_3$ = OverHS slope:**

```
display "STATA GLM Predicting Income from 3 Piecewise Linear Slopes for Education"
regress income c.lessHS c.gradHS c.overHS, level(95)

      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(3, 730)       =     47.84
       Model |  22906.5605         3  7635.52017   Prob > F        =    0.0000
    Residual |  116516.671       730  159.611879   R-squared       =    0.1643
-------------+----------------------------------   Adj R-squared   =    0.1609
       Total |  139423.232       733  190.209048   Root MSE        =    12.634
```

**Mean Square Error/Residual**, the residual variance, is 159.61 given the piecewise education slopes (which accounted for 16.43% of the variance in income, as given by the model $R^2 = .1643$).
The *F*-test says this $R^2$ is significantly > 0, written as: ***F*(3, 730) = 47.84, MSE = 159.61, *p* < .001**.

```
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      lessHS |  -.2687845   .5988015    -0.45   0.654    -1.444363     .906794  beta1
      gradHS |   4.684746   1.875684     2.50   0.013     1.002368    8.367124  beta2
      overHS |   2.124529   .2137244     9.94   0.000     1.704941    2.544117  beta3
       _cons |   8.534867   1.729351     4.94   0.000     5.139773    11.92996  beta0
------------------------------------------------------------------------------

// STATA Example of how to test differences between slopes
   lincom c.lessHS*-1 + c.gradHS*1 // Diff in ed slope:  2-11 vs 11-12

------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   4.953531   2.282227     2.17   0.030     .4730194    9.434042
------------------------------------------------------------------------------

   lincom c.gradHS*-1 + c.overHS*1 // Diff in ed slope: 11-12 vs 12-20
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |  -2.560217   1.946734    -1.32   0.189    -6.382082    1.261648
------------------------------------------------------------------------------
```

**Comparisons of Slopes Above:** The GradHS slope = 4.68 is significantly more positive than the LessHS slope = −0.27 by **4.95** per year (indicating that they should not be constrained to be the same). The OverHS slope = 2.12 is nonsignificantly less positive than the GradHS slope = 4.68 by **−2.56** per year, indicating that they *could* be constrained to be the same. However, the OverHS slope—implying a linear effect of each additional year of education past 12 years—does not appear to fit the means well, as shown in the overlaid plot below. So efforts to refine the model should focus on solving this problem first!




**Left:** model-predicted regression line through means for education (see SAS code online)

As shown by the misfit of the data to the model (dashed line), it looks like the effect of education after 12 years should have additional piecewise slopes (i.e., 12–15, 15–17, 17–18, 18–20)… if you are feeling brave, give it a try and let me know what happens!

```
print("R GLM Predicting Income from 3 Piecewise Linear Slopes for Education ")
ModelEd3 = lm(data=Example3, formula=income~1+lessHS+gradHS+overHS)
supernova(ModelEd3) # supernova prints sums of squares and residual variance
SaveModelEd3 = summary(ModelEd3) # saving summary that prints fixed effects solution
SaveModelEd3; confint(ModelEd3, level=.95) # confint for level% CI for fixed effects

print("R Example of how to test differences between slopes")
print("In number lists below, values are multiplier for each fixed effect IN ORDER")
PredEd3 = glht(model=ModelEd3, linfct=rbind(
  "Diff in ed slope:  2-11 vs 11-12" = c(0,-1, 1, 0),
  "Diff in ed slope: 11-12 vs 12-20" = c(0, 0,-1, 1)))
print("Print glht linear combination results with unadjusted p-values and 95% CIs")
SavePredEd3 = summary(PredEd3, test=adjusted("none"))
Print(SavePredEd3); confint(PredEd3, level=.95, calpha=univariate_calpha())
```

## Syntax and R Output to Compute Partial Effect Sizes for Piecewise Slopes and Differences:

```
// STATA code to compute partial correlations from fixed slopes
display "STATA Partial Correlations of Income with Education Slopes"
pcorr income lessHS gradHS overHS
```

STATA pcorr above only works for directly estimated fixed slopes, whereas the code below creating effect sizes out of stored results can be used for linear combinations as well (as shown).

```
// STATA code to compute effect sizes from stored results per lincom
lincom c.lessHS*-1 + c.gradHS*1 // Diff in ed slope:  2-11 vs 11-12
   display "PartialR= " (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))
lincom c.gradHS*-1 + c.overHS*1 // Diff in ed slope: 11-12 vs 12-20
   display "PartialR= " (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))

# R code to compute effect sizes from stored model fixed effects
ModelEd3PartialR=SaveModelEd3$coefficients[,"t value"]/
            sqrt(SaveModelEd3$coefficients[,"t value"]^2+ModelEd3$df.residual)
# Concatenate effect sizes to results table for fixed effects
data.frame(SaveModelEd3$coefficients, PartialR=ModelEd3PartialR)
```

```
             Estimate Std..Error  t.value   Pr...t..  PartialR
(Intercept)  8.53487    1.72935  4.93530 9.9215e-07  0.179691
lessHS      -0.26878    0.59880 -0.44887 6.5366e-01 -0.016611
gradHS       4.68475    1.87568  2.49762 1.2722e-02  0.092049
overHS       2.12453    0.21372  9.94051 6.3642e-22  0.345287
```

```
# R code to compute effect sizes from stored glht results
PredEd3PartialR=SavePredEd3$test$tstat/sqrt(SavePredEd3$test$tstat^2+ModelEd3$df.residual)
# Concatenate effect sizes to results table for mean differences
data.frame(Estimate=SavePredEd3$test$coefficients, SE=SavePredEd3$test$sigma,
           pvalue=SavePredEd3$test$pvalues, PartialR=PredEd3PartialR)
```

```
                                 Estimate     SE   pvalue  PartialR
Diff in ed slope:  2-11 vs 11-12   4.9535 2.2822 0.030292  0.080075
Diff in ed slope: 11-12 vs 12-20  -2.5602 1.9467 0.188878 -0.048618
```

R pcor.test (from R package ppcor) below only works for directly estimated fixed slopes, whereas the code above creating effect sizes out of stored results can be used for linear combinations as well.

```
# R alternative method to compute partial correlations for fixed slopes
print("R Partial Correlation of income with lessHS")
pcor.test(Example3$income,Example3$lessHS, Example3[,c("gradHS","overHS")])
```

```
   estimate p.value statistic   n gp  Method
1 -0.016611 0.65366  -0.44887 734  2 pearson
```

```
print("R Partial Correlation of income with gradHS")
pcor.test(Example3$income,Example3$gradHS, Example3[,c("lessHS","overHS")])
```

```
  estimate  p.value statistic   n gp  Method
1 0.092049 0.012722    2.4976 734  2 pearson
```

```
print("R Partial Correlation of income with overHS")
pcor.test(Example3$income,Example3$overHS, Example3[,c("lessHS","gradHS")])
```

```
  estimate    p.value statistic   n gp  Method
1  0.34529 6.3642e-22    9.9405 734  2 pearson
```

**Example Results Section for 3 Piecewise Linear Slopes for the Effect of Education:**

We used a general linear model (i.e., linear regression) to examine the extent to which annual income in thousands of dollars (*M* = 17.30, *SD* = 13.79) could be predicted from years of education (*M* = 13.81, *SD* = 2.91). We first examined the means of income by each level of education to identify plausible types of nonlinear associations. The effect of education predicting annual income appeared to differ across regions of education, suggesting a piecewise trend with the distinct region slopes to be captured by linear splines. Specifically, we fit one linear slope for the effect of education from 2 to 11 years, a second linear slope of education from 11 to 12 years, and a third linear slope of education from 12 to 20 years. Partial correlations were then computed from the *t* test-statistics to index effect size per slope. The model including these three education slopes captured a significant amount of variance in annual income, $F(3, 730) = 47.84$, MSE = 159.61, $p < .001$, $R^2 = .164$. The model fixed slopes can be interpreted as follows. Annual income was expected to be nonsignificantly lower by 0.27 thousand dollars per year of education from 2 to 11 years (*SE* = 0.60, *p* = .654, *r* = −.017), resulting in predicted annual income of 8.53 thousand dollars (SE = 1.73) at 11 years of education (i.e., as given by the fixed intercept). Annual income was then expected to be significantly higher by 4.68 thousand dollars (SE =1.88, *p* = .013, *r* = .092) for those achieving a high school degree (i.e., a significant difference between 11 and 12 years of education). Annual income was expected to be significantly higher by 2.12 thousand dollars (SE = 0.21, $p <$ .001, *r* = .345) per year of additional education past 12 years. However, examining a plot of the observed versus predicted means for annual income at each year of education suggested a linear slope was not sufficient in capturing the observed differences in income from 12 to 20 years of education. We recommend considering in future research the use of additional piecewise slopes corresponding to distinct levels of higher education (e.g., bachelors, masters, or doctoral college degrees).

## Syntax to Center 5-Category Ordinal Happiness at 1 (minimum):

```
// STATA code to create 1 new happy variable centered at lowest value
   gen happy1=happy-1
   label variable happy1 "happy1: Happy Category (0=1)"

# R code to make a single happy variable centered at lowest value
Example3$happy1=Example3$happy-1 # happy1: Happy Category (0=1)
```

## Syntax and STATA Output for 5-Category Ordinal Happiness Predicting Income:

**First Testing a Linear Effect of Happy (0=1):** $Income_i = \beta_0 + \beta_1(Happy_i - 1) + e_i$

**Interpret $\beta_0$ = Intercept:**

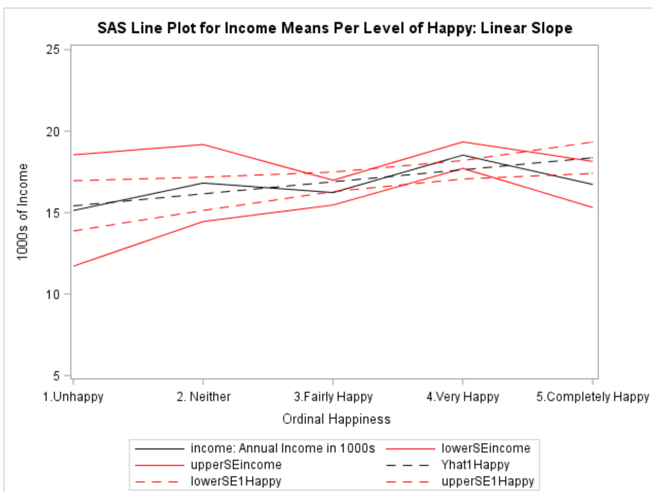
**Interpret $\beta_1$ = Happy1 slope:**

```
display "STATA GLM Predicting Income from Linear Happy Centered at 1"
regress income c.happyc1, level(95)

      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(1, 732)       =      1.69
       Model |  320.398119         1  320.398119   Prob > F        =    0.1945
    Residual |  139102.834       732  190.031194   R-squared       =    0.0023
-------------+----------------------------------   Adj R-squared   =    0.0009
       Total |  139423.232       733  190.209048   Root MSE        =    13.785
```

**Mean Square Error/Residual**, the residual variance, is 190.03 after a linear slope of happy (which accounted for 0.23% of the variance in income, as given by the model $R^2$ = .1945). The *F*-test tells us this $R^2$ is **not** significantly > 0, written as: ***F*(1, 732) = 1.69, MSE = 190.03, *p* = .195**. The same result is given by the *t*-test of the linear slope below ($t^2 = F$ for model when testing only one fixed slope).

```
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      happy1 |   .7386664   .5688736     1.30   0.195    -.3781521    1.855485  beta1
       _cons |   15.41495   1.540422    10.01   0.000     12.39078    18.43912  beta0
------------------------------------------------------------------------------
```

```
print("R GLM Predicting Income
from Linear Happy Centered at 1")
ModelHappy1 =
lm(data=Example3,
   formula=income~1+happy1)
supernova(ModelHappy1)
summary(ModelHappy1)
confint(ModelHappy1, level=.95)
```



## Syntax to Create Sequential-Dummy-Coded Predictors—4 needed for 5 happy categories:

In addition to not really making sense (i.e., these values are ordinal, so they aren't really numbers), a single linear slope predicting the same difference between each pair of happiness categories doesn't seem to fit the pattern of means. So let's fit a model with piecewise linear slopes created through sequential-dummy-coding, in which the slopes capture each shift between adjacent categories.

```
// STATA code to make 4 new sequential-dummy-coded variables for happy
// Make 4 new empty variables
   gen h1v2=.
   gen h2v3=.
   gen h3v4=.
   gen h4v5=.
// Replace each with 0 values
   replace h1v2=0 if happy < 2
   replace h2v3=0 if happy < 3
   replace h3v4=0 if happy < 4
   replace h4v5=0 if happy < 5
// Replace each with 1 values
   replace h1v2=1 if happy >= 2
   replace h2v3=1 if happy >= 3
   replace h3v4=1 if happy >= 4
   replace h4v5=1 if happy == 5
// Label variables
   label variable h1v2 "Slope from Happy 1 to 2"
   label variable h2v3 "Slope from Happy 2 to 3"
   label variable h3v4 "Slope from Happy 3 to 4"
   label variable h4v5 "Slope from Happy 4 to 5"
```

| Happy (x) | h1v2: Dif from 1 to 2 | h2v3: Dif from 2 to 3 | h3v4: Dif from 3 to 4 | h4v5: Dif from 4 to 5 |
|---|---|---|---|---|
| 1 (int) | 0 | 0 | 0 | 0 |
| 2 | **1** | 0 | 0 | 0 |
| 3 | 1 | **1** | 0 | 0 |
| 4 | 1 | 1 | **1** | 0 |
| 5 | 1 | 1 | 1 | **1** |

```
# R code to create 4 new sequential-dummy-coded predictors for happy
# Make 4 new empty variables
Example3$h1v2=NA; Example3$h2v3=NA; Example3$h3v4=NA; Example3$h4v5=NA;
# Replace each with 0 values
Example3$h1v2[which(Example3$happy<2)]=0
Example3$h2v3[which(Example3$happy<3)]=0
Example3$h3v4[which(Example3$happy<4)]=0
Example3$h4v5[which(Example3$happy<5)]=0
# Replace each with 1 values
Example3$h1v2[which(Example3$happy>=2)]=1
Example3$h2v3[which(Example3$happy>=3)]=1
Example3$h3v4[which(Example3$happy>=4)]=1
Example3$h4v5[which(Example3$happy>=5)]=1
```

## Syntax and R Output for 4 Sequential Slopes for 5-Category Happiness Predicting Income:

$$Income_i = \beta_0 + \beta_1(h1v2_i) + \beta_2(h2v3_i) + \beta_3(h3v4_i) + \beta_3(h4v5_i) + e_i$$

```
display "STATA GLM Predicting Income from 4 Sequential Slopes for Happy"
regress income c.h1v2 c.h2v3 c.h3v4 c.h4v5, level(95)

print("R GLM Predicting Income from 4 Sequential Slopes for Happy")
ModelHappy5 = lm(data=Example3, formula=income~1+h1v2+h2v3+h3v4+h4v5)
supernova(ModelHappy5) # supernova prints sums of squares and residual variance
```

```
                                  SS  df       MS     F   PRE     p
 ----- --------------- | ---------- --- -------- ----- ----- -----
 Model (error reduced) |    946.335   4  236.584 1.245 .0068 .2902
  h1v2                 |     44.300   1   44.300 0.233 .0003 .6293
  h2v3                 |     11.641   1   11.641 0.061 .0001 .8045
  h3v4                 |    759.119   1  759.119 3.996 .0055 .0460
  h4v5                 |    219.865   1  219.865 1.157 .0016 .2823
 Error (from model)    | 138476.897 729  189.955
 ----- --------------- | ---------- --- -------- ----- ----- -----
 Total (empty model)   | 139423.232 733  190.209
```

**Mean Square Error/Residual**, the residual variance, is 189.95 after adding the 4 sequential happy slopes (which accounted for 0.68% of the variance in income, as given by the model $R^2$ = .0068).
The *F*-test tells us this $R^2$ is **not** significantly > 0, written as: ***F*(4, 729) = 1.25, MSE = 189.95, *p* = .290**.

```
SaveModelHappy5 = summary(ModelHappy5) # saving summary that prints fixed effects solution
SaveModelHappy5; confint(ModelHappy5, level=.95) # confint for level% CI for fixed effects
```

```
Coefficients:
            Estimate Std. Error t value    Pr(>|t|)
(Intercept)   15.129      2.703    5.60 0.000000031  beta0
h1v2           1.685      3.489    0.48       0.629  beta1
h2v3          -0.586      2.369   -0.25       0.805  beta2
h3v4           2.299      1.150    2.00       0.046  beta3
h4v5          -1.797      1.670   -1.08       0.282  beta4
```

The fixed intercept gives the mean for happy=1, and each slope gives the difference to the next category.

```
Residual standard error: 13.8 on 729 degrees of freedom
Multiple R-squared:  0.00679,     Adjusted R-squared:  0.00134
F-statistic: 1.25 on 4 and 729 DF,  p-value: 0.29

                2.5 %  97.5 %
(Intercept)  9.822253 20.4352
h1v2        -5.165498  8.5358
h2v3        -5.237563  4.0646
h3v4         0.041241  4.5574
h4v5        -5.075962  1.4821
```

```
print("R Example of how to test differences between slopes")
print("In number lists below, values are multiplier for each fixed effect IN ORDER")
PredHappy5 = glht(model=ModelHappy5, linfct=rbind(
        "Diff in Slope 1-2 vs Slope 2-3" = c(0,-1, 1, 0, 0),
        "Diff in Slope 2-3 vs Slope 3-4" = c(0, 0,-1, 1, 0),
        "Diff in Slope 3-4 vs Slope 4-5" = c(0, 0, 0,-1, 1)))
print("Print glht linear combination results with unadjusted p-values and 95% CIs")
SavePredHappy5 = summary(PredHappy5, test=adjusted("none"))
print(SavePredHappy5); confint(PredHappy5, level=.95, calpha=univariate_calpha())
```

```
Linear Hypotheses:
                                    Estimate Std. Error t value Pr(>|t|)
Diff in Slope 1-2 vs Slope 2-3 == 0    -2.27       5.25   -0.43    0.665
Diff in Slope 2-3 vs Slope 3-4 == 0     2.89       2.90    0.99    0.320
Diff in Slope 3-4 vs Slope 4-5 == 0    -4.10       2.30   -1.78    0.075
(Adjusted p values reported -- none method)
```

**Comparisons of Slopes Above:** No pairwise differences between slopes are significant, which means we would not lose anything predictive informative by constraining the slopes to be equal in these data.

```
Quantile = 1.963
95% confidence level

Linear Hypotheses:
                                    Estimate lwr     upr
Diff in Slope 1-2 vs Slope 2-3 == 0  -2.272  -12.573   8.029
Diff in Slope 2-3 vs Slope 3-4 == 0   2.886   -2.811   8.582
Diff in Slope 3-4 vs Slope 4-5 == 0  -4.096   -8.605   0.413
```

## Syntax and R Output to Compute Partial Effect Sizes from Requested Piecewise Slopes:

```
// STATA code to compute partial correlations for fixed slopes
display "STATA Partial Correlations of Income with Happy Slopes"
pcorr income h1v2 h2v3 h3v4 h4v5
```

```
// STATA code to compute effect sizes from stored results per lincom
lincom c.h1v2*-1 + c.h2v3*1 // Diff in Slope 1-2 vs Slope 2-3
   display "PartialR=" (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))
lincom c.h2v3*-1 + c.h3v4*1 // Diff in Slope 2-3 vs Slope 3-4
   display "PartialR=" (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))
lincom c.h3v4*-1 + c.h4v5*1 // Diff in Slope 3-4 vs Slope 4-5
   display "PartialR=" (r(estimate)/r(se))/sqrt((r(estimate)/r(se))^2+r(df))

# R code to compute effect sizes from stored model fixed effects
ModelHappy5PartialR=SaveModelHappy5$coefficients[,"t value"]/
               sqrt(SaveModelHappy5$coefficients[,"t value"]^2+ModelHappy5$df.residual)
# Concatenate effect sizes to results table for fixed effects
data.frame(SaveModelHappy5$coefficients, PartialR=ModelHappy5PartialR)

            Estimate Std..Error  t.value       Pr...t..   PartialR
(Intercept) 15.12875     2.7030  5.59712 0.000000030865  0.2029852
h1v2         1.68516     3.4895  0.48292 0.629294831709  0.0178832
h2v3        -0.58649     2.3691 -0.24756 0.804546633997 -0.0091684
h3v4         2.29930     1.1502  1.99908 0.045970569657  0.0738379
h4v5        -1.79692     1.6702 -1.07585 0.282349065906 -0.0398148

# R code to compute effect sizes from stored glht results
PredHappy5PartialR=SavePredHappy5$test$tstat/
              sqrt(SavePredHappy5$test$tstat^2+ModelHappy5$df.residual)
# Concatenate effect sizes to results table for mean differences
data.frame(Estimate=SavePredHappy5$test$coefficients, pvalue=SavePredHappy5$test$pvalues,
           PartialR=PredHappy5PartialR)

                               Estimate   pvalue  PartialR
Diff in Slope 1-2 vs Slope 2-3  -2.2716 0.665182 -0.016033
Diff in Slope 2-3 vs Slope 3-4   2.8858 0.320293  0.036810
Diff in Slope 3-4 vs Slope 4-5  -4.0962 0.074905 -0.065916
```

**Example Results Section for the Linear and Piecewise Sequential Slopes for Happy:**

We used a general linear model (i.e., linear regression) to examine the extent to which annual income in thousands of dollars ($M$ = 17.30, $SD$ = 13.79) could be predicted from five-category ordinal happiness (unhappy = 3.54%, neither happy nor unhappy = 5.31%, fairly happy = 34.88%, very happy = 44.55%, completely happy = 11.72%). In first examining a linear effect of happiness (centered at unhappy = 0), the model fixed effects indicated that annual income was predicted to be 15.42 thousand dollars (SE = 1.54) for unhappy respondents (i.e., as given by the fixed intercept), and that annual income was predicted to be nonsignificantly greater by 0.74 thousand dollars (SE = 0.57, $p$ = .195, $R^2$ = .002) per additional ordinal level of happiness.

However, given that a linear slope for happiness assumes interval differences with respect to predicted income, we tested this assumption by specifying a piecewise slopes model by which to estimate all sequential differences in predicted annual income by ordinal level of happiness. Partial correlations were then computed from the $t$ test-statistics to index effect size per slope and slope difference. The revised model—predicting four sequential differences across the five levels of happiness—did not capture a significant amount of variance in annual income, $F$(4, 729) = 1.25, MSE = 189.95, $p$ = .290, $R^2$ = .007. The model fixed effects indicated that annual income was 15.13 thousand dollars (SE = 2.70) for unhappy respondents (i.e., as given by the fixed intercept). Annual income was nonsignificantly higher by 1.69 thousand dollars (SE = 3.49, $p$ = .629, $r$ = .018) for neither than unhappy respondents, nonsignificantly lower by 0.59 thousand dollars (SE = 2.37, $p$ = .804, $r$ = −.009) for fairly happy than neither respondents, significantly higher by 2.30 thousand dollars (SE = 1.15, $p$ = .046, $r$ = .073) for very happy than fairly happy respondents, and nonsignificantly lower by 1.80 thousand dollars (SE = 1.67, $p$ = .282, $r$ = −.040) for completely happy than very happy respondents. None of the differences between these adjacent differences were significant (as given by linear combinations of the model fixed effects, requested separately). Thus, there is little evidence that annual income can be predicted by self-rated happiness, whether treated as interval (through a linear slope) or treated as ordinal (through piecewise slopes).