Example 2: General Linear Models with a Single Quantitative or Binary 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 Example 1). The current example will use general linear models to predict a single quantitative outcome (annual income in 1000s) from a quantitative predictor (a linear slope for years of education) and from a binary predictor (marital status: 0=unmarried and 1=married). It will also introduce how to obtain linear combinations of fixed effects to create predicted outcomes using STATA LINCOM and R GLHT (and SAS ESTIMATE online).

STATA Syntax for Data Import and Manipulation:

```
// Paste in the folder address where "GSS_Example.xlsx" is saved between " "
// Using the UIowa virtual desktop, it would look like this
cd "\\Client\C:\Dropbox\24_PSQF6243\PSQF6243_Example2"

// Import GSS_Example.xlsx data from working directory and 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", sheet("GSS_Example") 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 marry   "marry: Marital Status (1=unmarried, 2=married)"
label variable educ    "educ: Years of Education"
label variable income  "income: Annual Income in 1000s"
```

R Syntax for Importing and Preparing Data for Analysis
(after loading packages *readxl*, *psych*, *supernova*, *multcomp*, 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_Example2")

# Import GSS_Example.xlsx data from working directory -- path = file name
Example2 = read_excel(path="GSS_Example.xlsx", sheet="GSS_Example")
# Convert to data frame to use for analysis
Example2 = as.data.frame(Example2)
# Labels added only as comments in R syntax file
```

STATA Descriptive Statistics:

```
display "STATA Descriptive Statistics for Quantitative or Binary Variables"
summarize income educ marry, detail

               income: Annual Income in 1000s
-------------------------------------------------------------
      Percentiles      Smallest
 1%         .245           .245
 5%          .98           .245
10%        2.695           .245       Obs                 734
25%       6.7375           .245       Sum of Wgt.         734

50%       13.475                      Mean           17.30287
                        Largest       Std. Dev.      13.79163
75%        22.05           58.8
90%       40.425           68.6       Variance        190.209
95%           49           68.6       Skewness        1.15836
99%         58.8           68.6       Kurtosis       4.086398
```

Remember, SD^2 = variance

$13.792^2 = 190.209$

```
                  educ: Years of Education
-------------------------------------------------------------
      Percentiles      Smallest
 1%            6              2
 5%            9              4
10%           11              4       Obs                 734
25%           12              4       Sum of Wgt.         734

50%           14                      Mean           13.81199
                        Largest       Std. Dev.      2.909282
75%           16             20
90%           18             20       Variance       8.463922
95%           19             20       Skewness      -.2301836
99%           20             20       Kurtosis       3.786849

       marry: Marital Status (1=unmarried, 2=married)
-------------------------------------------------------------
      Percentiles      Smallest
 1%            1              1
 5%            1              1
10%            1              1       Obs                 734
25%            1              1       Sum of Wgt.         734

50%            1                      Mean           1.459128
                        Largest       Std. Dev.      .4986665
75%            2              2
90%            2              2       Variance       .2486683
95%            2              2       Skewness       .1640367
99%            2              2       Kurtosis       1.026908
```

R Descriptive Statistics:

```
# describe prints sample descriptive statistics for quantitative variables
# list variables to be included in separate quotes within c concatenate function
# wrapped a print command around to get more than two significant digits
print("R Descriptive Statistics for Quantitative for Quantitative or Binary Variables")
print(describe(x=Example2[ , c("income","educ","marry")], fast=TRUE), digits=3)
```

```
       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
educ      2 734 13.812  2.909 14.000 2.000 20.0 18.000 -0.230    0.777 0.107
marry     3 734  1.459  0.499  1.000 1.000  2.0  1.000  0.164   -1.976 0.018
```

SD^2 = variance

$13.792^2 = 190.209$

```
# Get variances too (on diagonal of output covariance matrix)
var(x=Example2[ , c("income","educ","marry")])
```

```
          income      educ    marry
income 190.2090 15.436039 1.547618
educ    15.4360  8.463922 0.074161
marry    1.5476  0.074161 0.248668
```

This is called a “covariance matrix” (or “variance–covariance matrix”). Variances are on the diagonal, and covariances are on the off-diagonal.

Empty General Linear Model (no predictors):
$\boldsymbol{Income_i = \beta_0 + e_i}$

The empty model is our starting point—the most naïve prediction of income in which everyone is predicted to have the mean income: $\hat{y}_i = \beta_0$. Thus, the variance of the e_i residuals will be ALL the y_i variance. In the output below, MS stands for Mean Square. **Mean Square Residual is the residual variance** (= 190.21 here). The Root MSE is the square root of residual variance—the residual standard deviation describes how wrong the model prediction is across people on average. Stay tuned for what the rest of the first table means! 😊

In STATA:

STATA's **regress** is general GLM routine. The first word after **regress** is the outcome variable. Level(95) requests 95% confidence intervals (the default).

```
display "STATA GLM Empty Model Predicting Income"
regress income , level(95) // level gives (95)% CI for unstandardized solution

      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(0, 733)       =      0.00
       Model |           0         0           .   Prob > F        =         .
    Residual |  139423.232       733  190.209048   R-squared       =    0.0000
-------------+----------------------------------   Adj R-squared   =    0.0000
       Total |  139423.232       733  190.209048   Root MSE        =    13.792

------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |   17.30287   .5090583    33.99   0.000     16.30349    18.30226  beta0
------------------------------------------------------------------------------
```

STATA refers to the fixed intercept as **_cons**, which stands for constant. In models with more than one fixed effect, STATA will always list the fixed intercept LAST (much to my dismay).

In R:

```
print("R Empty GLM Predicting Income -- save as ModelEmpty")
ModelEmpty = lm(data=Example2, formula=income~1) # 1 represents intercept
supernova(ModelEmpty)    # supernova prints sums of squares and residual variance

Analysis of Variance Table (Type III SS)
 Model: income ~ 1

                                 SS  df      MS   F PRE   p
 ----- --------------- | ---------- --- ------- --- --- ---
 Model (error reduced) |        --- ---     --- --- --- ---
 Error (from model)    |        --- ---     --- --- --- ---
 ----- --------------- | ---------- --- ------- --- --- ---
 Total (empty model)   | 139423.232 733 190.209

summary(ModelEmpty)      # summary prints fixed effects solution

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   17.303      0.509      34   <2e-16  Beta0

Residual standard error: 13.8 on 733 degrees of freedom

confint(ModelEmpty, level=.95) # confint prints level% CI for fixed effects

             2.5 % 97.5 %
(Intercept) 16.303 18.302
```

Now let's see if years of education can predict income by giving it a fixed linear slope!

$$Income_i = \beta_0 + \beta_1(Educ_i) + e_i$$

Interpret β_0 = intercept:

Interpret β_1 = slope of education:

How much income variance is leftover after considering education?

How wrong is the model-predicted income on average?

In STATA:

```
display "STATA GLM Predicting Income from Original Education"
regress income educ, level(95)

      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(1, 732)       =    127.16
       Model |  20634.9817         1  20634.9817   Prob > F        =    0.0000
    Residual |   118788.25       732   162.27903   R-squared       =    0.1480
-------------+----------------------------------   Adj R-squared   =    0.1468
       Total |  139423.232       733  190.209048   Root MSE        =    12.739

------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        educ |   1.823746    .161731    11.28   0.000     1.506234    2.141258  beta1
       _cons |  -7.886679   2.282778    -3.45   0.001    -12.36825   -3.405107  beta0
------------------------------------------------------------------------------
```

STATA always lists the fixed intercept last!

In R:

```
print("R GLM Predicting Income from Original Education -- save as ModelEduc")
ModelEduc = lm(data=Example2, formula=income~1+educ)
supernova(ModelEduc)   # supernova prints sums of squares and residual variance

Analysis of Variance Table (Type III SS)
                                  SS  df        MS       F   PRE     p
 ----- ---------------- | ---------- --- --------- ------- ----- -----
 Model (error reduced) |  20634.982   1 20634.982 127.157 .1480 .0000
 Error (from model)    | 118788.250 732   162.279
 ----- ---------------- | ---------- --- --------- ------- ----- -----
 Total (empty model)   | 139423.232 733   190.209

summary(ModelEduc)     # summary prints fixed effects solution

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -7.887      2.283   -3.45  0.00058  Beta0
educ           1.824      0.162   11.28  < 2e-16  Beta1

Residual standard error: 12.7 on 732 degrees of freedom
Multiple R-squared:  0.148,       Adjusted R-squared:  0.147
F-statistic:  127 on 1 and 732 DF,  p-value: <2e-16

confint(ModelEduc, level=.95) # confint prints level% CI for fixed effects

               2.5 %  97.5 %
(Intercept) -12.3683 -3.4051
educ          1.5062  2.1413
```

Given that no one actually had education = 0 years, let's center the education predictor so 0 now indicates 12 years to create a more meaningful model intercept (i.e., the "you are here" sign as the model reference point).

Add a linear slope of a CENTERED quantitative years of education predictor:
$$Income_i = \beta_0 + \beta_1(Educ_i - 12) + e_i$$

Interpret β_0 = intercept:

Interpret β_1 = slope of (education−12):

In STATA:

```
// Center education predictor so that 0 is meaningful
gen educ12=educ-12
label variable educ12 "educ12: Education (0=12 years)"

display "STATA GLM Predicting Income from Centered Education (0=12)"
regress income educ12, level(95)  // with 95% CI for unstandardized solution

      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(1, 732)       =    127.16
       Model |  20634.9817         1  20634.9817   Prob > F        =    0.0000
    Residual |  118788.25        732  162.27903    R-squared       =    0.1480
-------------+----------------------------------   Adj R-squared   =    0.1468
       Total |  139423.232       733  190.209048   Root MSE        =    12.739

------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      educ12 |   1.823746    .161731    11.28   0.000     1.506234    2.141258 beta1 is same
       _cons |   13.99827   .5540485    25.27   0.000     12.91055    15.08598 beta0 differs
------------------------------------------------------------------------------
```

In R:

```
# Center education predictor so that 0 is meaningful: new = old-12
Example2$educ12 = Example2$educ-12
# educ12: Education (0=12 years)  # label as a comment only

print("R GLM Predicting Income from Centered Education 0=12 -- save as ModelEduc12")
ModelEduc12 = lm(data=Example2, formula=income~1+educ12)
supernova(ModelEduc12)   # supernova prints residual variance

Analysis of Variance Table (Type III SS)
                                SS  df        MS       F   PRE     p
 ----- --------------- | ---------- --- --------- ------- ----- -----
 Model (error reduced) |  20634.982   1 20634.982 127.157 .1480 .0000
 Error (from model)    | 118788.250 732   162.279
 ----- --------------- | ---------- --- --------- ------- ----- -----
 Total (empty model)   | 139423.232 733   190.209

summary(ModelEduc12) # summary prints fixed effects solution

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   13.998      0.554    25.3   <2e-16
educ12         1.824      0.162    11.3   <2e-16

confint(ModelEduc12, level=.95) # confint prints level% CI for fixed effects
              2.5 %  97.5 %
(Intercept) 12.9106 15.0860
educ12       1.5062  2.1413
```

The next set of commands in each program illustrate how to compute predicted $\hat{y}_i$ outcomes given any value(s) of the predictor(s). Model: $Income_i = \beta_0 + \beta_1(Educ_i - 12) + e_i$

Predicted income for 8 years education: $\hat{y}_i = 14.00(1) + 1.82(-4) = 6.70$
Predicted income for 12 years education: $\hat{y}_i = 14.00(1) + 1.82(0) = 14.00$
Predicted income for 16 years education: $\hat{y}_i = 14.00(1) + 1.82(4) = 21.29$
Predicted income for 20 years education: $\hat{y}_i = 14.00(1) + 1.82(8) = 28.59$

```
// In STATA LINCOMs below, _cons is the intercept, words refer to the beta fixed effect,
// and values are the multiplier for the requested predictor value
lincom _cons*1 + educ12*-4   // Pred Income for  8 years (educ12=-4)
lincom _cons*1 + educ12*0    // Pred Income for 12 years (educ12= 0)
lincom _cons*1 + educ12*4    // Pred Income for 16 years (educ12= 4)
lincom _cons*1 + educ12*8    // Pred Income for 18 years (educ12= 8)
```

```
print("R Get predicted outcomes using glht from multcomp package -- save as PredEduc12")
print("In number lists below, the values are multipliers for each fixed effect in order")
PredEduc12 = glht(model=ModelEduc12, linfct=rbind(
  "Pred Income at  8 years (educ12=-4)" = c(1,-4),
  "Pred Income at 12 years (educ12= 0)" = c(1, 0),
  "Pred Income at 16 years (educ12= 4)" = c(1, 4),
  "Pred Income at 20 years (educ12= 8)" = c(1, 8)))
print("Print glht linear combination results with unadjusted p-values")
summary(PredEduc12, test=adjusted("none"))
confint(PredEduc12, level=.95, calpha=univariate_calpha())
```

These are the results from STATA LINCOMs:

```
. lincom _cons*1 + educ12*-4  // Pred Income for  8 years (educ12=-4)
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   6.703285   1.051023     6.38   0.000     4.639907    8.766664
------------------------------------------------------------------------------

. lincom _cons*1 + educ12*0   // Pred Income for 12 years (educ12= 0)
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   13.99827   .5540485    25.27   0.000     12.91055    15.08598
------------------------------------------------------------------------------

. lincom _cons*1 + educ12*4   // Pred Income for 16 years (educ12= 4)
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   21.29325   .5884829    36.18   0.000     20.13793    22.44857
------------------------------------------------------------------------------

. lincom _cons*1 + educ12*8   // Pred Income for 18 years (educ12= 8)
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   28.58823   1.105747    25.85   0.000     26.41742    30.75905
------------------------------------------------------------------------------
```

These are the results from R GLHTs:

```
Linear Hypotheses:
                                         Estimate Std. Error t value       Pr(>|t|)
Pred Income for  8 years (educ12=-4) == 0    6.703      1.051    6.38 0.00000000032
Pred Income for 12 years (educ12= 0) == 0   13.998      0.554   25.27        < 2e-16
Pred Income for 16 years (educ12= 4) == 0   21.293      0.588   36.18        < 2e-16
Pred Income for 20 years (educ12= 8) == 0   28.588      1.106   25.85        < 2e-16
```

```
Simultaneous Confidence Intervals
                                        Estimate lwr      upr
Pred Income at  8 years (educ12=-4) == 0  6.70329  4.63991  8.76666
Pred Income at 12 years (educ12= 0) == 0 13.99827 12.91055 15.08598
Pred Income at 16 years (educ12= 4) == 0 21.29325 20.13793 22.44857
Pred Income at 20 years (educ12= 8) == 0 28.58823 26.41742 30.75905
```

Standardized Solution for Education Predicting Income: Results using standardized variables (z-scored income and education), in which fixed slopes are then in a correlation metric (−1 to 1)

In STATA:

```
display "STATA GLM Predicting Income from Centered Education (0=12)"
regress income educ12, beta  // beta gives standardized solution

-----------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|                    Beta
-------------+---------------------------------------------------------------
      educ12 |   1.823746    .161731    11.28   0.000                .3847109 beta1
       _cons |   13.99827   .5540485    25.27   0.000                       . beta0  (=0)
-----------------------------------------------------------------------------
```

In R:

```
print ("R standardized fixed effect solution using lm.beta package")
lm.beta(ModelEduc12)

Standardized Coefficients::
(Intercept)      educ12
         NA     0.38471
```

Now let's see if binary marital status can predict income by giving it a fixed linear slope!

$$Income_i = \beta_0 + \beta_1(Marry01_i) + e_i$$

Interpret β_0 = intercept:

Interpret β_1 = slope of marry01:

<u>Results will be:</u>

Predicted income unmarried (marry01=0): $\hat{y}_i = 14.45(1) + 6.22(0) = 14.45$

Predicted income unmarried (marry01=1): $\hat{y}_i = 14.45(1) + 6.22(1) = 20.67$

How much income variance is leftover after considering education?

How wrong is the model-predicted income on average?

In STATA:

```
// Recode marry predictor so that 0 is meaningful
gen marry01=. // Create new empty variable, then recode
replace marry01=0 if marry==1
replace marry01=1 if marry==2
label variable marry01 "marry01: 0=unmarried, 1=married"

display "STATA GLM Predict Income from Marry01 (0=Unmarried,1=Married)"
regress income marry01, level(95) // with 95% CI for unstandardized solution
// Save fixed effects solution in a matrix " marryresults" for use in computation below
matrix marryresults = r(table)
```

```
      Source |       SS           df       MS      Number of obs   =       734
-------------+----------------------------------   F(1, 732)       =     39.04
       Model |  7060.10161         1  7060.10161   Prob > F        =    0.0000
    Residual |   132363.13       732  180.823948   R-squared       =    0.0506
-------------+----------------------------------   Adj R-squared   =    0.0493
       Total |  139423.232       733  190.209048   Root MSE        =    13.447

------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     marry01 |   6.223623   .9960148     6.25   0.000     4.268237     8.17901 beta1
       _cons |   14.44543   .6748896    21.40   0.000     13.12048    15.77038 beta0
------------------------------------------------------------------------------
```

```
lincom _cons*1 + marry01*0 // Pred Income for Unmarried=0 = Beta0
lincom _cons*1 + marry01*1 // Pred Income for Married=1   = Beta0 + Beta1
```

```
. lincom _cons*1 + marry01*0 // Pred Income for Unmarried=0 = Beta0
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   14.44543   .6748896    21.40   0.000     13.12048    15.77038
------------------------------------------------------------------------------

. lincom _cons*1 + marry01*1 // Pred Income for Married=1 = Beta0 + Beta1
------------------------------------------------------------------------------
      income |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   20.66906   .7325091    28.22   0.000     19.23099    22.10713
------------------------------------------------------------------------------
```

In R:

```
# Recode marry predictor so that 0 is meaningful
Example2$marry01=NA  # Create new empty variable
Example2$marry01[which(Example2$marry==1)]=0  # marry01=0 if marry=1
Example2$marry01[which(Example2$marry==2)]=1  # marry01=1 if marry=2
# marry01: 0=unmarried, 1=married             # label as a comment only

print("R GLM Predicting Income from Marry01 (0=Unmarried,1=Married) -- save ModelMarry01")
ModelMarry01 = lm(data=Example2, formula=income~1+marry01)
supernova(ModelMarry01)   # supernova prints residual variance
```

```
Analysis of Variance Table (Type III SS)
                                SS  df       MS      F   PRE     p
 ----- --------------- | ---------- --- -------- ------ ----- -----
 Model (error reduced) |   7060.102   1 7060.102 39.044 .0506 .0000
 Error (from model)    | 132363.130 732  180.824
 ----- --------------- | ---------- --- -------- ------ ----- -----
 Total (empty model)   | 139423.232 733  190.209
```

```
summary(ModelMarry01)     # summary prints fixed effects solution
```

```
Coefficients:
            Estimate Std. Error t value     Pr(>|t|)
(Intercept)   14.445      0.675   21.40      < 2e-16
marry01        6.224      0.996    6.25 0.0000000007

Residual standard error: 13.4 on 732 degrees of freedom
Multiple R-squared:  0.0506,      Adjusted R-squared:  0.0493
F-statistic:   39 on 1 and 732 DF,  p-value: 0.000000000703
```

```
confint(ModelMarry01, level=.95) # confint to print level% CI for fixed effects
```

```
              2.5 % 97.5 %
(Intercept) 13.1205 15.770
marry01      4.2682  8.179
```

```
print("R Get predicted outcomes using glht from multcomp package -- save as PredMarry01")
print("In number lists below, values are multiplier for each fixed effect in order")
PredMarry01 = glht(model=ModelMarry01, linfct=rbind(
  "Pred Income for Unmarried=0" = c(1,0),
  "Pred income for Married=1"   = c(1,1)))
print("Print glht linear combination results with unadjusted p-values")
summary(PredMarry01, test=adjusted("none"))
confint(PredMarry01, level=.95, calpha=univariate_calpha())

Linear Hypotheses:
                                 Estimate Std. Error t value Pr(>|t|)
Pred Income for Unmarried=0 == 0   14.445      0.675    21.4   <2e-16
Pred income for Married=1 == 0     20.669      0.733    28.2   <2e-16

Simultaneous Confidence Intervals
Linear Hypotheses:
                                 Estimate lwr      upr
Pred Income for Unmarried=0 == 0 14.44543 13.12048 15.77038
Pred income for Married=1 == 0   20.66906 19.23099 22.10713
```

One last thing: To get a Cohen's d effect size for the mean income difference between unmarried and married persons, we can calculate d from the t test-statistic: $d = \frac{2t}{\sqrt{DF_{den}}} = \frac{2*6.25}{\sqrt{732}} = 0.462$ → mean income is about 0.462 standard deviations higher for married than unmarried persons.

In STATA:

```
display "STATA Compute Cohen's d effect size from t test-statistic manually"
display 2*6.25/sqrt(732)  // d = 2*t/SQRT(DF_den)
.46201329

display "STATA Compute Cohen's d effect size from t test-statistic using internal values"
matrix list marryresults   // Show internally saved object of fixed effects

marryresults[9,2]
          marry01      _cons
     b  6.2236234  14.445435
    se  .99601482  .67488958
     t  6.2485249  21.404145
pvalue  7.029e-10  2.621e-79
    ll   4.268237  13.120484
    ul  8.1790097  15.770385
    df        732        732
  crit  1.9632101  1.9632101
 eform          0          0

// t test-statistic we want is in row 3 column 1
display 2*marryresults[3,1]/ sqrt(e(df_r))   // d = 2*t/SQRT(DF_den)
.46190425
```

In R:

```
print("R Compute Cohen's d effect size from t test-statistic manually")
2*6.25/sqrt(732)
[1] 0.46201329
```

```
print("Compute Cohen's d effect size from t test-statistic using internal values")
as.matrix(summary(ModelMarry01)$coefficients[,3]) # print saved t values

                 [,1]
(Intercept) 21.4041
marry01      6.2485

# t test-statistic we want is in row 2 column 1
as.matrix(summary(ModelMarry01)$coefficients[,3])[2,1]*2 / sqrt(ModelMarry01$df.residual)
marry01
 0.4619
```

Here is what the saved objects for the last model look like in the R environment:

Name	Type	Value
ModelMarry01	list [12] (S3: lm)	List of length 12
coefficients	double [2]	14.45 6.22
(Intercept)	double [1]	14.445
marry01	double [1]	6.2236
residuals	double [734]	-11.26 -6.48 6.28 7.60 12.41 28.33 ...
effects	double [734]	-468.78 84.02 6.44 8.20 12.56 28.49 ...
rank	integer [1]	2
fitted.values	double [734]	14.4 14.4 20.7 14.4 20.7 20.7 ...
assign	integer [2]	0 1
qr	list [5] (S3: qr)	List of length 5
df.residual	integer [1]	732

Example Results Section (although it's more verbose than would be typical for the sake of completeness):

The extent to which annual income in thousands of US dollars (*M* = 17.30, *SD* = 13.79) could be predicted from years of education (*M* = 13.81, *SD* = 2.91) and binary marital status (1 = unmarried 54.09%, 2 = married 45.91%) was examined in separate general linear models (i.e., simple linear regressions). All analyses were conducted using [the regress function in Stata v. 18] or [the lm function in R v. 4.4.0]. Predicted outcomes were generated using [lincom in Stata] or [the glht function within the multicomp package v. 1.4-25 in R].

To create a meaningful model intercept, education was centered such that 0 = 12 years. Education was found to be a significant predictor of annual income: Relative to the reference expected income for a person with 12 years of education provided by the model intercept of 14.00k (SE = 0.55), for every additional year of education, annual income was expected to be higher by 1.82k (SE = 0.16, $p < .001$), resulting in a standardized coefficient = 0.38 (i.e., the Pearson correlation between annual income and education). For example, persons with only 8 years of education were predicted to have an annual income of only 6.70k (SE = 1.05), persons with 16 years of education were predicted to have an annual income of 21.29k (SE = 0.59), and persons with 20 years of education were predicted to have an annual income of 28.59k (SE = 1.11). *[Spoiler alert: we will test the adequacy of only a linear (constant) effect for years of education in Example 3.]*

We then examined prediction of annual income by binary marital status. To create a meaningful model intercept, marital status was dummy-coded so that 0 = unmarried persons and 1 = married persons. Marital status was also a significant predictor of annual income: Relative to the reference expected income for unmarried persons provided by the model intercept of 14.45k (SE = 0.67), married persons were expected to have significantly greater income by 6.22k (SE = 1.00, $p < .001$), resulting in a predicted income for married persons of 20.67k (SE = 0.73) and a standardized mean difference of Cohen's $d = 0.462$.

Note: because a GLM with a single binary predictor is also known as a "two-sample t-test" here is what the results would look like written from that angle… A two-sample *t*-test (i.e., assuming homogeneous variance across groups) was used to examine mean differences between unmarried and married persons in annual income. A significant mean difference was found, $t(732) = 6.25$, $p < .001$, such that annual income for married persons (*M* = 20.67k, SE = 0.73) was significantly higher than for unmarried persons (*M* = 14.45k, SE = 0.67).

Bonus: Bivariate Pearson Correlation Matrix, Significance Tests, and Confidence Intervals

In STATA:

```
display "STATA Pearson Correlations and CIs"
pwcorr income educ marry, sig

             |   income     educ    marry
-------------+---------------------------
      income |   1.0000
             |
        educ |   0.3847   1.0000
             |   0.0000
       marry |   0.2250   0.0511   1.0000
             |   0.0000   0.1665
```

In this "correlation matrix" the top value is the correlation coefficient ***r*** and the bottom value is the *p*-value for that correlation.

These same values are in separate tables in the R output below.

```
// To get CI using r-to-z, need to download and run a special module
ssc install ci2
ci2 income educ, corr
ci2 income marry, corr

Confidence interval for Pearson's product-moment correlation of income and educ, based on Fisher's
transformation. Correlation = 0.385 on 734 observations (95% CI: 0.321 to 0.445)

Confidence interval for Pearson's product-moment correlation of income and marry, based on Fisher's
transformation. Correlation = 0.225 on 734 observations (95% CI: 0.155 to 0.293)
```

In R after loading the Hmisc package:

```
print("R Pearson Correlation Matrix with P-values using rcorr from Hmisc package")
cor(x=cbind(Example2$income,Example2$educ,Example2$marry), method="pearson")

        income educ12 marry01
income    1.00   0.38    0.23
educ12    0.38   1.00    0.05
marry01   0.23   0.05    1.00

P
        income educ12 marry01
income         0.0000 0.0000
educ12  0.0000        0.1665
marry01 0.0000 0.1665

print("R Pearson Correlation Pairwise Significance tests and CIs")
cor.test(x=Example2$income, y=Example2$educ,  method="pearson", conf.level=.95)
cor.test(x=Example2$income, y=Example2$marry, method="pearson", conf.level=.95)

data:  Example2$income and Example2$educ
t = 11.2764, df = 732, p-value < 0.0000000000000000222
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval: 0.32129033 0.44469587
sample estimates:
       cor
0.38471088

data:  Example2$income and Example2$marry
t = 6.24852, df = 732, p-value = 0.00000000070292
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval: 0.15519069 0.29262863
sample estimates:
      cor
0.2250287
```