Assignment 4.5

Cory J. Cascalheira

03/13/2021

# Assignment Instructions

**Important Note**: Stepwise methods of model selection are generally frowned upon by the statistics community. It is viewed similar to what has been called “p-hacking.” However, they can sometimes be helpful when doing exploratory analysis. It is more important to have theoretical reasons for model specification, and model building techniques, such as either starting with a full model as recommended by theory and removing things based on output, or starting with the most basic model and iteratively adding variables as per theory, are preferred over things like stepwise methods.

First, compare the following hierarchical models (meaning 1 is nested within 2 because it contains a subset of it). The confidence variable (ST37Q05) asked students how confident they were in solving an equation like 3x+5=17, where 1=Very Confident and 4=Not at all Confident. Make sure to change it to a continuous, numeric variable if it is not already so.

Model 1: PV1MATH = ESCS + FEMALE + LANG + ENJOY

Model 2: PV1MATH = ESCS + FEMALE + LANG + ENJOY + CONFIDENCE

In R, you have to do a couple of things first. In order to compare the models, you have to be using the same number of people in each one (something SPSS does in the background). So, you first have to create a subset of the data for just the set of variables you will be using, and then remove all the rows with missing data in them, like this, using the dplyr package (I called my dataset “pisa” so swap that out with whatever you called yours):

pisa2<-dplyr::select(pisa,PV1MATH,ESCS,FEMALE,LANG,ST29Q04,ST37Q05)

Then call the names(pisa2) function to check that it subset correctly. Then, remove all NAs: pisa2<-na.omit(pisa2)

Then you can store the two models (for instance just store them into model1 and model2). Then you can use the anova() function to compare the models and the tab\_model() function from sjPlots will give you a nice comparison table combining results from both models and allowing you to see all the same information as in SPSS (albeit in slightly different places).

anova(model1, model2)

tab\_model(model1, model2)

**ANSWER THE QUESTION**: Is model 2 better? How do you know?

Second, use the stepwise function to compare iterative sets of nested, hierarchical models and determine which one you think might be best. (the model to start with should be the one that includes all the variables, including confidence; model 2 from above).

In R, first install the package “olsrr”, and then create the model that includes all the variables (perhaps you store it as model3). Then run the function: ols\_step\_both\_p(model3,pent=.05,prem=.10). pent=.05 means add if p<.05 and prem=.10 means remove if p>.10 (these are the same as the default specs used by SPSS). This will give you just simple output showing you the change in R2, and which variable was added to the previous model at each step. You can ignore all columns to the right of Adjusted R2. To get the full information for every different model, add details=TRUE to the function: ols\_step\_both\_p(model3,pent=.05,prem=.10,details=TRUE).

Note that this almost reproduces the SPSS output. You do not, however, get the same comparison across models (e.g., change in F with associated p-value). To get that, you would have to create each model on your own and use anova() to compare across them. I am not going to make you do that, however.

**ANSWER THE QUESTION**: Which model do you think might be best? What makes you say so?

# Establish the Work Environment

Load all the dependencies and import the data.

# Dependencies  
library(car)  
library(psych)  
library(ppcor)  
library(tidyverse)  
library(rio)  
library(sjPlot)  
library(olsrr)  
library(MASS)  
  
# Import  
pisa <- import("../data/2012 PISA multiple countries selected variables.sav") %>%  
 as\_tibble  
  
# Construct the APA theme for plots  
# Construct the APA theme  
apa\_theme <- theme\_bw() +  
 theme(panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(),  
 panel.border = element\_blank(),  
 axis.line = element\_line(),  
 plot.title = element\_text(hjust = 0.5),  
 text = element\_text(size = 12, family = "sans"),  
 axis.text.x = element\_text(color = "black"),  
 axis.text.y = element\_text(color = "black"),  
 axis.title = element\_text(face = "bold"))

## Recode Data

First, I recoded the data to facilitate coding and interpretation.

# Rename pisa  
pisa\_1 <- pisa %>%  
 rename(country = CNT, language = ST25Q01, enjoy\_math = ST29Q04,  
 gender = ST04Q01, math\_career = ST48Q05, applied\_math = ST76Q01,  
 confidence = ST37Q05, math\_score = PV1MATH, ses = ESCS)  
  
# Recode values  
pisa\_2 <- pisa\_1 %>%  
 mutate(  
 # 0 = same; 1 = different  
 language = recode(language, `1` = 0, `2` = 1),  
 # 0 = male; 1 = female  
 gender = recode(gender, `1` = 1, `2` = 0),  
 # Recode enjoy math so increasing numbers mean increased enjoyment  
 enjoy\_math = recode(enjoy\_math, `1` = 4, `2` = 3, `3` = 2, `4` = 1),  
 # Recode confidence so increasing numbers mean increased confidence  
 confidence = recode(confidence, `1` = 4, `2` = 3, `3` = 2, `4` = 1)  
 )  
pisa\_2

## # A tibble: 65,535 x 9  
## country math\_score ses language enjoy\_math gender math\_career applied\_math  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 CAN 492. 0.93 NA NA 1 NA 3  
## 2 CAN 394. -0.78 0 NA 1 NA 3  
## 3 CAN 390. -1.3 0 1 0 2 2  
## 4 CAN 504. 0.56 0 2 1 2 3  
## 5 CAN 466. -0.03 0 3 0 1 NA  
## 6 CAN 398. 0.74 0 1 1 2 2  
## 7 CAN 404. NA NA NA 1 NA NA  
## 8 CAN 406. -2.58 0 4 1 2 NA  
## 9 CAN 609. 0.88 0 4 0 1 NA  
## 10 CAN 452. 0.44 0 1 1 2 NA  
## # ... with 65,525 more rows, and 1 more variable: confidence <dbl>

## Assumption Checking

Other major assumptions (e.g., normality, linearity) were assumed met since we have worked with this dataset several times before.

### Collinearity

Collinearity is a subset of multicollinearity applied to two predictors (Field et al., 2012). Multicollinearity occurs when two or more predictors are highly correlated within a regression model. As values of collinearity increase, problems are introduced into the model, such as:

* unstable beta weights, which might fit the sample but not represent the population;
* less contribution to multiple correlation (i.e., *R*) and variance explained because collinear variables share too much common variance; and
* difficulty assessing which predictors are most important.

First, we need to specify the model.

# Create the model  
model <- lm(math\_score ~ ses + gender + language + enjoy\_math, data = pisa\_2)

The variance inflation factor (VIF) is a measure of “whether a predictor has a strong linear relationship with other predictor(s)” (Field et al., 2012, p. 276). Values of 10 are problematic. If the average VIF is greater than 1, then multicollinearity is likely a problem.

Tolerance, according to Field and colleagues, is the reciprocal of VIF. Problems occurs when tolerance is less than 0.1 and multicollinearity is more likely a problem when tolerance is less than 0.2.

# Check the variance inflation factor (VIF)  
vif(model)

## ses gender language enjoy\_math   
## 1.019888 1.005030 1.006297 1.018768

# Average VIF  
mean(vif(model))

## [1] 1.012496

# Tolerance  
1 / vif(model)

## ses gender language enjoy\_math   
## 0.9805001 0.9949950 0.9937428 0.9815776

None of the above values reached the thresholds suggested by Field et al. (2012), therefore multicollinearity is likely not a problem. The average VIF is barely greater than 1.

## Part Correlations

A part correlation (i.e., a semi-partial correlation) is the *unique* correlation after overlapping association between two variables is removed. Conceptually, the procedure carves out a “part” of the multiple correlation coefficient; when the part correlation is squared, it carves out the variance explained (i.e., the multiple R^2) by the unique predictor.

Partial is like a relative frequency. It is out of 100% of multiple R^2.

Whereas the part correlation carves out a part of the multiple R^2. For example, if the R^2 is 0.241 and the part correlation for SES is .483, then the squared value of the part correlation is 0.231, which is in the same units as multiple R^2.

# Select the variables in the model  
pisa\_3 <- pisa\_2 %>%  
 select(-country, -math\_career, -applied\_math, -confidence)  
  
# Calculate semi-partial correlation (i.e., part)  
# Remove missing values  
part <- spcor(na.omit(as.data.frame(pisa\_3)), method = "pearson")  
part

## $estimate  
## math\_score ses language enjoy\_math gender  
## math\_score 1.00000000 0.482515108 0.031543578 0.06892420 -0.038743018  
## ses 0.47969178 1.000000000 0.040210691 -0.12373319 -0.007309905  
## language 0.03606365 0.046243313 1.000000000 0.03294749 -0.006582796  
## enjoy\_math 0.07812423 -0.141074622 0.032664616 1.00000000 -0.059851654  
## gender -0.04430790 -0.008409073 -0.006584751 -0.06038789 1.000000000  
##   
## $p.value  
## math\_score ses language enjoy\_math gender  
## math\_score 0.000000e+00 0.000000e+00 1.186177e-10 4.543661e-45 2.528916e-15  
## ses 0.000000e+00 0.000000e+00 2.172373e-16 7.503922e-142 1.356189e-01  
## language 1.776280e-13 3.539479e-21 0.000000e+00 1.721094e-11 1.789921e-01  
## enjoy\_math 1.966051e-57 3.277066e-184 2.555886e-11 0.000000e+00 2.164618e-34  
## gender 1.433418e-19 8.603270e-02 1.788631e-01 5.571087e-35 0.000000e+00  
##   
## $statistic  
## math\_score ses language enjoy\_math gender  
## math\_score 0.000000 112.461929 6.442733 14.104210 -7.915214  
## ses 111.606456 0.000000 8.215537 -25.455372 -1.492335  
## language 7.367079 9.450544 0.000000 6.729784 -1.343887  
## enjoy\_math 15.997723 -29.090901 6.671944 0.000000 -12.240480  
## gender -9.054217 -1.716748 -1.344286 -12.350546 0.000000  
##   
## $n  
## [1] 41681  
##   
## $gp  
## [1] 3  
##   
## $method  
## [1] "pearson"

## Model Comparison

First, we must construct the models to compare. The models must have the same number of participants.

# Select variables of interest  
pisa\_compare <- pisa\_2 %>%  
 select(math\_score, ses, gender, language, enjoy\_math, confidence)  
  
# Remove missing values  
pisa\_compare\_1 <- na.omit(pisa\_compare)  
  
# Model 1: PV1MATH = ESCS + FEMALE + LANG + ENJOY  
model\_1 <- lm(math\_score ~ ses + gender + language + enjoy\_math,   
 data = pisa\_compare\_1)  
  
# Model 2: PV1MATH = ESCS + FEMALE + LANG + ENJOY + CONFIDENCE  
model\_2 <- lm(math\_score ~ ses + gender + language + enjoy\_math + confidence,   
 data = pisa\_compare\_1)

Now we can compare the models.

# Are the models significantly different from one another?  
anova(model\_1, model\_2)

## Analysis of Variance Table  
##   
## Model 1: math\_score ~ ses + gender + language + enjoy\_math  
## Model 2: math\_score ~ ses + gender + language + enjoy\_math + confidence  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 41339 274126325   
## 2 41338 246867733 1 27258592 4564.5 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Find the adjusted multiple R-squared, but do not print the results due to a formatting issue.

# Show table in R studio but not in the Word document  
tab\_model(model\_1, model\_2)

## Stepwise Multiple Regression

Now we compared the models using iterations.

# Construct the model  
model\_3 <- lm(math\_score ~ ., data = pisa\_compare\_1)  
  
# Perform the stepwise multiple regression  
ols\_step\_both\_p(model\_3, pent = .05, prem = .10)

## Error in if (pvals[minp] <= pent) {: argument is of length zero

Since there was an error in the command from *olsrr*, I [used a function](http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/154-stepwise-regression-essentials-in-r/) from the *MASS* package. I chose “backward” stepwise regression since it started with the full model (Field et al., 2012) and for the ease of interpretation.

I consulted [this resource to interpret the Akaike information criterion (AIC)](https://stats.stackexchange.com/questions/347652/default-stepaic-in-r/347655).

# Stepwise regression model from MASS  
step\_model <- stepAIC(model\_3, direction = "backward")

## Start: AIC=359484.9  
## math\_score ~ ses + gender + language + enjoy\_math + confidence  
##   
## Df Sum of Sq RSS AIC  
## <none> 246867733 359485  
## - enjoy\_math 1 105148 246972882 359500  
## - language 1 153478 247021212 359509  
## - gender 1 948459 247816192 359641  
## - confidence 1 27258592 274126325 363813  
## - ses 1 55732262 302599996 367899

summary(step\_model)

##   
## Call:  
## lm(formula = math\_score ~ ses + gender + language + enjoy\_math +   
## confidence, data = pisa\_compare\_1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -336.76 -52.07 0.33 51.77 307.33   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 368.6067 1.7615 209.261 < 2e-16 \*\*\*  
## ses 30.7645 0.3185 96.604 < 2e-16 \*\*\*  
## gender -9.6196 0.7633 -12.602 < 2e-16 \*\*\*  
## language 6.5780 1.2976 5.070 4.01e-07 \*\*\*  
## enjoy\_math -1.8569 0.4425 -4.196 2.72e-05 \*\*\*  
## confidence 33.3723 0.4940 67.561 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 77.28 on 41338 degrees of freedom  
## Multiple R-squared: 0.3168, Adjusted R-squared: 0.3167   
## F-statistic: 3834 on 5 and 41338 DF, p-value: < 2.2e-16

# Stepwise regression model from stats  
step(model\_3)

## Start: AIC=359484.9  
## math\_score ~ ses + gender + language + enjoy\_math + confidence  
##   
## Df Sum of Sq RSS AIC  
## <none> 246867733 359485  
## - enjoy\_math 1 105148 246972882 359500  
## - language 1 153478 247021212 359509  
## - gender 1 948459 247816192 359641  
## - confidence 1 27258592 274126325 363813  
## - ses 1 55732262 302599996 367899

##   
## Call:  
## lm(formula = math\_score ~ ses + gender + language + enjoy\_math +   
## confidence, data = pisa\_compare\_1)  
##   
## Coefficients:  
## (Intercept) ses gender language enjoy\_math confidence   
## 368.607 30.765 -9.620 6.578 -1.857 33.372

# Interpretation

# Summarize the model  
summary(model)

##   
## Call:  
## lm(formula = math\_score ~ ses + gender + language + enjoy\_math,   
## data = pisa\_2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -378.80 -55.64 -0.97 54.88 335.62   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 456.6267 1.2423 367.579 < 2e-16 \*\*\*  
## ses 36.4991 0.3229 113.047 < 2e-16 \*\*\*  
## gender -7.2765 0.8016 -9.077 < 2e-16 \*\*\*  
## language 10.0581 1.3610 7.390 1.49e-13 \*\*\*  
## enjoy\_math 7.1521 0.4429 16.148 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 81.56 on 41676 degrees of freedom  
## (23854 observations deleted due to missingness)  
## Multiple R-squared: 0.2407, Adjusted R-squared: 0.2407   
## F-statistic: 3304 on 4 and 41676 DF, p-value: < 2.2e-16

# Confidence intervals  
confint(model)

## 2.5 % 97.5 %  
## (Intercept) 454.191858 459.061551  
## ses 35.866242 37.131888  
## gender -8.847711 -5.705252  
## language 7.390562 12.725737  
## enjoy\_math 6.284013 8.020233

# Partial correlations - ses  
round(part$estimate[2,1]^2, 4) \* 100

## [1] 23.01

# Partial correlations - language  
round(part$estimate[3,1]^2, 4) \* 100

## [1] 0.13

# Partial correlations - enjoy\_math  
round(part$estimate[4,1]^2, 4) \* 100

## [1] 0.61

# Partial correlations - gender  
round(part$estimate[5,1]^2, 4) \* 100

## [1] 0.2

Results from the multiple regression were significant, *F*(4, 41,676) = 3,304, *p* < .001, and the model explained 24.07% of the variance in math achievement. All of the variables were significant predictors. Socioeconomic status (*t* = 113.05, *p* < .001) significantly contributed to the variance in math achievement, indicating that for every one unit increase in standardized socioeconomic status, a student’s score on the math achievement assessment would increase by 36.5 points (95% CI [35.87, 37.13]). Gender (*t* = 9.08, *p* < .001) and language (*t* = 10.06, *p* < .001) were also significant predictors, but the practical significance was limited: being female (versus male) was associated with a 7.28 point increase in math achievement (95% CI [5.71, 8.85]) and having a native language that was different from the test (versus the same) was associated with a 10.06 point increase in math achievement (95% CI [7.39, 12.73]). Finally, enjoyment of math exhibited a significant relationship with math achievement (*t* = 16.15, *p* < .001), indicating that participants tended to score 7.15 points higher (95% CI [6.28, 8.05]) for every one unit increase in math enjoyment. Again, math enjoyment is not considered practically significant given the scale of math achievement.

Part correlations were conducted to examine the unique contributions to the overall variance in math scores. Part correlations were squared to determine the unique variance. Results indicated that standardized socioeconomic status explained 23.01% of the variance in math achievement, speaking a language different from the test explained 0.13%, student enjoyment of math explained 0.61%, and being female explained 0.2%. Therefore, only socioeconomic status explained a meaningful amount of the variance in math achievement.

## Model Comparison: Is model 2 better? How do you know?

Model 2 is significantly better than model 1, *F*(1, 41,338) = 4,564.5, *p* < .001. The multiple increased from .241 to .317, indicating the the new model explained an additional 7.6% of the variance in math scores (i.e., .317 - .241). This seems meaningful since the confidence variable predicts a 33.37 point increase in math scores for every one unit increase in confidence.

## Stepwise Hierachical Regression: Which model do you think might be best? What makes you say so?

A lower AIC is ideal (Field et al., 2012). The stepwise regression model with the lowest AIC (359,485) is the maximal model (i.e., the model [with all predictor variables](https://stats.stackexchange.com/questions/347652/default-stepaic-in-r/347655) included). However, the maximal model is only marginally better than a model without the enjoy math variable (AIC = 359,500). Removing either the confidence in math (AIC = 363,813) or the standardized SES (AIC = 367,899) results in the poorest prediction as evidenced by the larger AIC values. Strictly speaking, we should retain the maximal model. However, in practice, the maximal model may not add much meaningful variance over regression models that exclude variables like enjoy math or language given that the changes in the AIC are less than .006% (i.e., 1 - 359,485 / 359,509).

# References

Field, A., J. Miles, & Z. Field. (2012). *Discovering statistics using R*. SAGE Publications Ltd.