Reproducible R code for SEM
Situated Academic Writing Self-Efficacy Scale Validation
Uses nursing student data
Includes total scale score for nursing identity PIQ, Writing apprehension, three context items labled support, value, stakes

```
library(lavaan)
library(sem)
library(semPlot)
library(knitr)
library(dplyr)
library(tidyr)
```

SAWSES item labels (because the item labels in the code are different than the ones in the data file

Latent variable 1 Relate.Reflect
SAWSES7 meaning
SAWSES8 improve
SAWSES10 reflect
SAWSES12 ideas
SAWSES13 overall
SAWSES18 wander
SAWSES19 adapt
SAWSES20 feedback

Latent variable 2 Creative.Identity
SAWSES6 creativity
SAWSES11 spark
SAWSES15 voice
SAWSES16 original
SAWSES17 discipline

Latent variable 3 Essentials
SAWSES1 overcome
SAWSES3 words
SAWSES5 synthesize

(the missing SAWSES variables were the ones excluded at the EFA stage 2,4,9,14)

#Model 1 base model from the original hypothesis with stakes, value, support, apprehension and identity predicting the SAWSES higher order latent variable

```
model.SAWSES1 = '
```

```
SAWSES <~ Stakes + Value + Support + Apprehension + Identity
Essentials =~ overcome + words + synthesize
Relate.Reflect =~ meaning + improve + reflect + ideas + overall + wander + adapt + feedback
Creative.Identity =~ creativity + spark + voice + original + discipline
SAWSES =~ Essentials + Relate.Reflect + Creative.Identity'

SAWSES1fit = lavaan::sem(model.SAWSES1, estimator = "MLR", data = nursingdata,
std.lv=TRUE)
summary(SAWSES1fit, rsquare=TRUE, standardized=TRUE, fit.measures=TRUE)

modindices(SAWSES1fit, sort. = TRUE)

#picture of model
semPaths(SAWSES1fit, whatLabels = "std", layout = "tree2")

parameterEstimates(SAWSES1fit, standardized=TRUE)
parameterEstimates(SAWSES1fit, standardized=TRUE) %>%
  filter(op == "=~") %>%
  select('Latent Factor'=lhs, Indicator=rhs, B=est, SE=se, Z=z, 'p-value'=pvalue, Beta=std.all) %>%
  kable(digits = 3, format="pandoc", caption="Factor Loadings")

# Model 2: Modification indices were considered and applied to this above base model. All
reduce covariance between particular items, but despite the possible high correlations between
these items, the items are asking often about very different writing aspects

model.SAWSES2 = '
SAWSES <~ Stakes + Value + Support + Apprehension + Identity
Essentials =~ overcome + words + synthesize
Relate.Reflect =~ meaning + improve + reflect + ideas + overall + wander + adapt + feedback
Creative.Identity =~ creativity + spark + voice + original + discipline
SAWSES =~ Essentials + Relate.Reflect + Creative.Identity
synthesize~~creativity
words~~discipline
improve~~feedback
creativity~~spark'

SAWSES2fit = lavaan::sem(model.SAWSES2, estimator = "MLR", data = nursingdata,
std.lv=TRUE)
summary(SAWSES2fit, rsquare=TRUE, standardized=TRUE, fit.measures=TRUE)

parameterEstimates(SAWSES2fit, standardized=TRUE)

parameterEstimates(SAWSES2fit, standardized=TRUE) %>%
  filter(op == "=~") %>%
```

```
  select('Latent Factor'=lhs, Indicator=rhs, B=est, SE=se, Z=z, 'p-value'=pvalue, Beta=std.all) %>%
  kable(digits = 3, format="pandoc", caption="Factor Loadings")

modindices(SAWSES2fit, sort. = TRUE)

#Picture of data
semPaths(SAWSES2fit, whatLabels = "std", layout = "tree2")
```

#Model 3 Removes Non Significant predictors Value – this model did a weird thing and made all the Betas on the latent variable Relate.Reflect non significant. Statistician feels likely SAWSES higher order variable is weak.

```
model.SAWSES3 = '
SAWSES <~ Stakes + Support + Apprehension + Identity
Essentials =~ overcome + words + synthesize
Relate.Reflect =~ meaning + improve + reflect + ideas + overall + wander + adapt + feedback
Creative.Identity =~ creativity + spark + voice + original + discipline
SAWSES =~ Essentials + Relate.Reflect + Creative.Identity
synthesize~~creativity
words~~discipline
improve~~feedback
creativity~~spark'

SAWSES3fit = lavaan::sem(model.SAWSES3, estimator = "MLR", data = nursingdata,
std.lv=TRUE)
summary(SAWSES3fit, rsquare=TRUE, standardized=TRUE, fit.measures=TRUE)

parameterEstimates(SAWSES3fit, standardized=TRUE)

modindices(SAWSES3fit, sort. = TRUE)
semPaths(SAWSES3fit, whatLabels = "std", layout = "tree2")

parameterEstimates(SAWSES3fit, standardized=TRUE) %>%
  filter(op == "=~") %>%
  select('Latent Factor'=lhs, Indicator=rhs, B=est, SE=se, Z=z, 'p-value'=pvalue, Beta=std.all) %>%
  kable(digits = 3, format="pandoc", caption="Factor Loadings")
```

#Model 4 removes the weak predictors and leaves apprehension and support as the only predictors

```
model.SAWSES4 = '
```

```
SAWSES <~ Apprehension + Support
Essentials =~ overcome + words + synthesize
Relate.Reflect =~ meaning + improve + reflect + ideas + overall + wander + adapt + feedback
Creative.Identity =~ creativity + spark + voice + original + discipline
SAWSES =~ Essentials + Relate.Reflect + Creative.Identity
synthesize~~creativity
words~~discipline
improve~~feedback
creativity~~spark'

SAWSES4fit = lavaan::sem(model.SAWSES5, estimator = "MLR", data = nursingdata,
std.lv=TRUE)
summary(SAWSES4fit, rsquare=TRUE, standardized=TRUE, fit.measures=TRUE)

parameterEstimates(SAWSES4fit, standardized=TRUE)

modindices(SAWSES4fit, sort. = TRUE)
semPaths(SAWSES4fit, whatLabels = "std", layout = "tree2")

parameterEstimates(SAWSES4fit, standardized=TRUE) %>%
  filter(op == "=~") %>%
  select('Latent Factor'=lhs, Indicator=rhs, B=est, SE=se, Z=z, 'p-value'=pvalue, Beta=std.all) %>%
  kable(digits = 3, format="pandoc", caption="Factor Loadings")

#And here we were just playing
#removes higher order latent variable SAWSES with model

model.SAWSES6 = '
Essentials <~ Stakes + Support + Apprehension + Identity
Relate.Reflect <~ Stakes + Support + Apprehension + Identity
Creative.Identity <~ Stakes + Support + Apprehension + Identity
Essentials =~ overcome + words + synthesize
Relate.Reflect =~ meaning + improve + reflect + ideas + overall + wander + adapt + feedback
Creative.Identity =~ creativity + spark + voice + original + discipline'

SAWSES6fit = lavaan::sem(model.SAWSES6, estimator = "MLR", data = nursingdata,
std.lv=TRUE)
summary(SAWSES6fit, rsquare=TRUE, standardized=TRUE, fit.measures=TRUE)

parameterEstimates(SAWSES6fit, standardized=TRUE)

modindices(SAWSES6fit, sort. = TRUE)
semPaths(SAWSES6fit, whatLabels = "std", layout = "tree2")
```

```
parameterEstimates(SAWSES6fit, standardized=TRUE) %>%
  filter(op == "=~") %>%
  select('Latent Factor'=lhs, Indicator=rhs, B=est, SE=se, Z=z, 'p-value'=pvalue, Beta=std.all) %>%
  kable(digits = 3, format="pandoc", caption="Factor Loadings")

#adjusts predictors from Model 6 (more playing)

model.SAWSES7 = '
Essentials <~ Apprehension
Relate.Reflect <~ Stakes + Support + Apprehension + Identity
Creative.Identity <~ Apprehension
Essentials =~ overcome + words + synthesize
Relate.Reflect =~ meaning + improve + reflect + ideas + overall + wander + adapt + feedback
Creative.Identity =~ creativity + spark + voice + original + discipline'

SAWSES7fit = lavaan::sem(model.SAWSES7, estimator = "MLR", data = nursingdata,
std.lv=TRUE)
summary(SAWSES7fit, rsquare=TRUE, standardized=TRUE, fit.measures=TRUE)

parameterEstimates(SAWSES7fit, standardized=TRUE)

modindices(SAWSES7fit, sort. = TRUE)
semPaths(SAWSES7fit, whatLabels = "std", layout = "tree2")

parameterEstimates(SAWSES7fit, standardized=TRUE) %>%
  filter(op == "=~") %>%
  select('Latent Factor'=lhs, Indicator=rhs, B=est, SE=se, Z=z, 'p-value'=pvalue, Beta=std.all) %>%
  kable(digits = 3, format="pandoc", caption="Factor Loadings")


#relate.reflect examination (more playing to see if the Relate.Reflect factor by itself had an
unusual fit)

model.SAWSES8 = '

Relate.Reflect <~ Stakes + Support + Apprehension + Identity
Relate.Reflect =~ meaning + improve + reflect + ideas + overall + wander + adapt + feedback'

SAWSES8fit = lavaan::sem(model.SAWSES8, estimator = "MLR", data = nursingdata,
std.lv=TRUE)
summary(SAWSES8fit, rsquare=TRUE, standardized=TRUE, fit.measures=TRUE)

parameterEstimates(SAWSES8fit, standardized=TRUE)
```

```
modindices(SAWSES8fit, sort. = TRUE)
semPaths(SAWSES8fit, whatLabels = "std", layout = "tree2")

parameterEstimates(SAWSES8fit, standardized=TRUE) %>%
  filter(op == "=~") %>%
  select('Latent Factor'=lhs, Indicator=rhs, B=est, SE=se, Z=z, 'p-value'=pvalue, Beta=std.all) %>%
  kable(digits = 3, format="pandoc", caption="Factor Loadings")
```