

ASSIGNMENT SEVEN 7

AHMAD SAQUIB SINA

April 18, 2019

```
library(lavaan)

## Warning: package 'lavaan' was built under R version 3.5.3

## This is lavaan 0.6-3

## lavaan is BETA software! Please report any bugs.

lab7 = read.csv("C:/Users/sinax006.AD/Desktop/epsy8266-master/course_materials/labs/lab7/lab7.csv")
```

Question 1: When you are modeling *just* BMI, how many free parameters can you have to have a just identified model? What you are modeling BMI and negative affect, how many free parameters can you have to have a just identified model? (1 pt)

There are five time points to measure the BMI. As we include covariances and means, the number of free parameters for *just* BMI is $(5*8)/2=20$ to have a just identified model. If I am modeling BMI and negative affect, then the number of free parameters for BMI and negative affect is $(6*9)/2=27$ to have a just identified model.

Question 2: Create a connected means plot of BMI and describe the change in BMI across the study. Hint: To do this you will need to first reshape the data. (3 pts)

```
library(reshape)

## Warning: package 'reshape' was built under R version 3.5.3

lab7.long <- reshape(lab7,
                     direction = "long",
                     varying = paste0("bmi.", c("0", "2", "8", "14", "26")),
                     timevar = "month",
                     idvar = "id")

lab7.long <- lab7.long[order(lab7.long$id, lab7.long$month), ]
head(lab7.long)

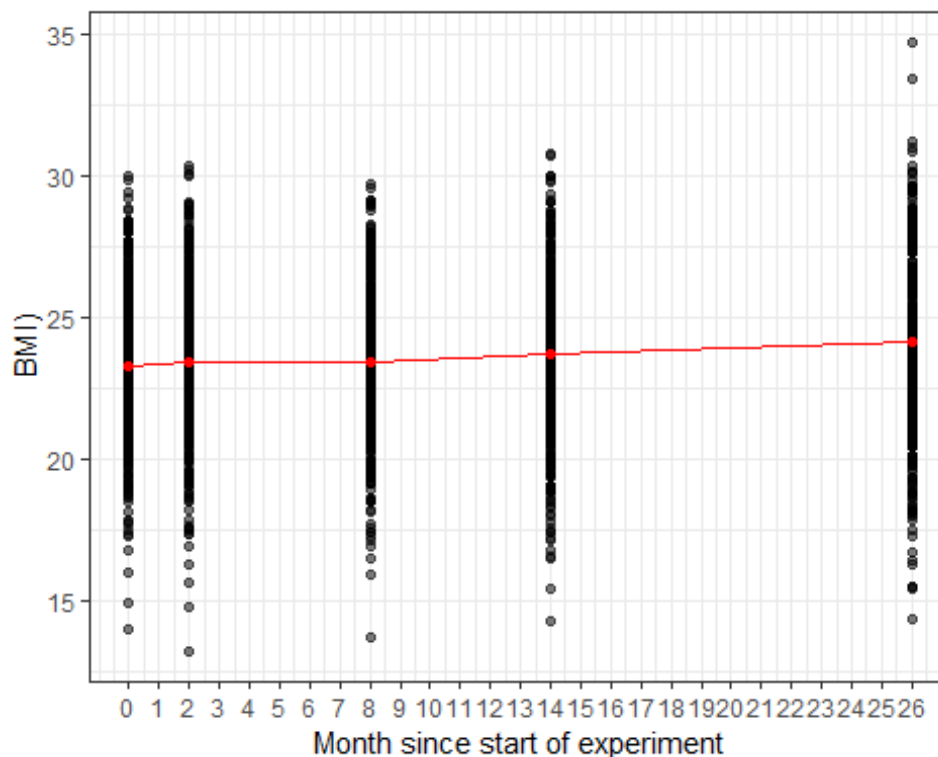
##      id neg.aff month      bmi
## 1.0    1 1.115209     0 20.18862
## 1.2    1 1.115209     2 19.70724
## 1.8    1 1.115209     8 21.57151
## 1.14   1 1.115209    14 22.55225
```

```
## 1.26  1 1.115209    26 20.59329
## 2.0   2 3.487453     0 20.69621

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.5.3

library(ggplot2)
ggplot(data = lab7.long, aes(x = month, y = bmi)) + geom_point(alpha=0.5) +
stat_summary(fun.y = "mean", geom = "point", col = "red", size = 1.5) +
stat_summary(fun.y = "mean", geom = "line", col = "red") +
xlab("Month since start of experiment") +
ylab("BMI") +
scale_x_continuous(breaks = 0:26) +
theme_bw()
```



From the connected means plot, it is apparent that the BMI changes across the study. For example, the plot shows that the BMI increases from the start of the intervention (Month 0) to the end of the intervention (Month 2). However, The BMI decreases 6 months after the end of the intervention (month 8, i.e., 8 months since the start of the intervention). The BMI again increases from Month 8 to Month 14 that is 12 months after the end of the intervention. Finally, the BMI continuously increases from Month 14 to Month 24 that is 24 months after the end of the intervention (month 26).

Question 3: What is the highest order polynomial model you can fit? (1 pt)

In this study, we have five time points. We need four points for quadratic polynomial model. Therefore, we could fit cubic polynomial model for this study.

Question 4: Do you think a linear, quadratic, or nonlinear curve model would fit best? Why? (1 pt)

-0.5 You're right that they are all possible, but the question is asking which one would fit *best* based on the observed pattern of means

Yes, I think a linear, quadratic, or nonlinear curve model would fit best. Linear and polynomial growth include an intercept and a slope. For linear, we need three time points. For quadratic, we need four time points. For nonlinear curve, we need at least three time points and an intercept and a slope. It is apparent that we have all data available to fit linear, quadratic, or nonlinear curve model. Therefore, I think a linear, quadratic, or nonlinear curve model would fit best.

Question 5: Fit a linear model for BMI where you a) constrain the error variances for the BMI manifest variables to be equal and b) ignore autocorrelation between the errors (I will refer to this model as **linear.fixed**). Hint, the 2nd unstandardized pattern coefficient for the slope factor does not have to be 1. (2 pts)

```
line.mod <- "  
# define factors  
int =~ 1*bmi.0 + 1*bmi.2 + 1*bmi.8 + 1*bmi.14 +1*bmi.26  
slope =~ 0*bmi.0 + 2*bmi.2 + 8*bmi.8 + 14*bmi.14 +26*bmi.26  
# estimate factor means  
int ~ 1  
slope ~ 1  
# estimate factor variance/covariance  
int ~~ int + slope  
slope ~~ slope  
# assume variances are equivalent across time  
bmi.0 ~~ e*bmi.0  
bmi.2 ~~ e*bmi.2  
bmi.8 ~~ e*bmi.8  
bmi.14 ~~ e*bmi.14  
bmi.26 ~~ e*bmi.26  
"  
  
linear.fixed <- lavaan(model = line.mod, data=lab7)  
fitMeasures(linear.fixed, c("chisq", "df", "p", "rmsea", "cfi", "tli", "srmr"  
"))  
  
##   chisq      df  rmsea    cfi    tli    srmr  
## 352.412 14.000  0.258  0.891  0.922  0.044
```

Question 6: Based on global fit (chi-square test of model fit, RMSEA, CFI, and SRMR) is the **linear.fixed** model a good model? (2 pts)

```
fitMeasures(linear.fixed, c("chisq", "df", "p", "rmsea", "cfi", "tli", "srmr"  
"))
```

##	chisq	df	rmsea	cfi	tli	srmr
##	352.412	14.000	0.258	0.891	0.922	0.044

The CFI value is 0.891 and the TLI is 0.922. 0.8 is way too generous for good fit on CFI/TLI. The RMSEA value is 0.258. RMSEA less than 0.06 or 0.08 are considered indicators of good fit. However, here the RMSEA value is greater than the acceptable range which does not indicate a good fit for model. The SRMR value less than 0.1 indicates a good fit and the value of our result is 0.044. The value of chisq is comparatively high with respect to degrees of freedom. Although, SRMR, CFI and TLI value indicate good fit, RMSEA and chisq value with respect to df do not express a good fit for this model. Therefore, based on global fit, the linear.fixed model is not a good model.

Question 7: Examine the correlation residuals and the mean residuals for the **linear.fixed** model. Do you see evidence of local misfit? (3 pts)

```
lavResiduals(linear.fixed, type="cor")
```

```
## $type
## [1] "cor.bollen"
##
## $cov
##      bmi.0  bmi.2  bmi.8  bmi.14 bmi.26
## bmi.0    0.000
## bmi.2    0.056  0.000
## bmi.8    0.010  0.026  0.000
## bmi.14   -0.007  0.006 -0.010  0.000
## bmi.26    0.029  0.016 -0.059 -0.024  0.000
##
## $mean
##      bmi.0  bmi.2  bmi.8  bmi.14 bmi.26
##      0.015  0.025 -0.048 -0.012  0.016
##
## $cov.z
##      bmi.0  bmi.2  bmi.8  bmi.14 bmi.26
## bmi.0    0.000
## bmi.2   14.314  0.000
## bmi.8    1.918  5.991  0.000
## bmi.14   -0.927  0.870 -1.674  0.000
## bmi.26    6.142  3.443 -6.165 -3.577  0.000
##
## $mean.z
##      bmi.0  bmi.2  bmi.8  bmi.14 bmi.26
##      1.529  3.433 -3.330 -0.856  2.151
##
## $summary
##      crmr crmr.se crmr.z crmr.pvalue ucrmr ucrmr.se
## cov    0.030  0.002 11.118      0.000 0.030  0.003
## mean    0.027  0.005  3.261      0.001 0.026  0.008
## total   0.029  0.003  8.222      0.000 0.028  0.003
```

From the value it is visible that there is no absolute value that is greater than 0.10 for the correlation residuals. Therefore, this value of the correlation residuals do not show the evidence of local misfit. However, for the standardized residuals, there are six absolute values that are greater than 1.96. Therefore, the values of standardized residuals show the evidence of local fit though correlation residuals do not show it.

-1 What about the mean residuals?

Question eight

```
library(lavaan)
line.mod <- "
# define factors
int =~ 1*bmi.0 + 1*bmi.2 + 1*bmi.8 + 1*bmi.14 +1*bmi.26
slope =~ 0*bmi.0 + 2*bmi.2 + 8*bmi.8 + 14*bmi.14 +26*bmi.26
# estimate factor means
int ~ 1
slope ~ 1
# estimate factor variance/covariance
int ~~ int + slope
slope ~~ slope
# assume variances are not equivalent across time
bmi.0 ~~ bmi.0
bmi.2 ~~ bmi.2
bmi.8 ~~ bmi.8
bmi.14 ~~ bmi.14
bmi.26 ~~ bmi.26
"

linear.free <- lavaan(model = line.mod, data=lab7)
fitMeasures(linear.free, c("chisq", "df", "p", "rmsea", "cfi", "tli", "srmr"))

##   chisq      df  rmsea    cfi    tli    srmr
## 132.661 10.000  0.184  0.960  0.960  0.053

anova(linear.fixed, linear.free)

## Chi Square Difference Test
##
##              Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## linear.free   10 6141.7 6180.7 132.66
## linear.fixed  14 6353.5 6376.9 352.41    219.75      4 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Yes, this linear.free model is an improvement in model fit over the linear.fixed model. The chisq value for linear.free model is lower than the linear.fixed model. The p value for the chi-square test is statistically significant. Linear.free model includes more complex model than the linear.fixed model because in this model we allow the error variances of the BMI manifest variables to be free. Ideally, the more complex of the two models compared with chi-squared test should fit the data reasonably well. The “larger” model with more freely estimated parameters fits the data better than the “smaller” model in which the parameters in question are fixed. Therefore, this model is an improvement over the linear.fixed model.

Question 9: Based on global fit (chi-square test of model fit, RMSEA, CFI, and SRMR) is the linear.free model a good model? (2 pts)

```
fitMeasures(linear.free, c("chisq", "df", "p", "rmsea", "cfi", "tli", "srmr"))
```

	chisq	df	rmsea	cfi	tli	srmr
##	132.661	10.000	0.184	0.960	0.960	0.053

The CFI value is 0.960 and the TLI is 0.960. 0.8 is way too generous for good fit on CFI/TLI. The RMSEA value is 0.184. RMSEA less than 0.06 or 0.08 are considered indicators of good fit. However, here the RMSEA value is greater than the acceptable range which does not indicate a good fit for model. The SRMR value less than 0.1 indicates a good fit and the value of our result is 0.053. The value of chisq is 132.661 and df is 10. Compared to linear.fixed model, this model shows overall good fit as the chisq value for linear.free model is lower than the linear.fixed model. The "larger" model with more freely estimated parameters fits the data better than the "smaller" model in which the parameters in question are fixed. Therefore, this linear.free model is a good model compared to linear.fixed model though the RMSEA value is greater than 0.06 or 0.08. The question was asking whether it was a good fit on its own, not just in comparison to linear.fixed

Question 10: Examine the correlation residuals and the mean residuals for the linear.free model. Do you see evidence of local misfit? (3 pts)

```
lavResiduals(linear.free, type="cor")
```

```
## $type
## [1] "cor.bollen"
##
## $cov
##      bmi.0  bmi.2  bmi.8  bmi.14 bmi.26
## bmi.0    0.000
## bmi.2    0.002  0.000
## bmi.8   -0.008  0.003  0.000
## bmi.14  -0.022 -0.014  0.011  0.000
## bmi.26   0.031  0.016 -0.018  0.029  0.000
##
## $mean
##      bmi.0  bmi.2  bmi.8  bmi.14  bmi.26
## -0.003   0.010 -0.056 -0.012   0.030
##
## $cov.z
##      bmi.0  bmi.2  bmi.8  bmi.14  bmi.26
## bmi.0    0.000
## bmi.2    4.595  0.000
## bmi.8   -2.307  1.186  0.000
## bmi.14  -5.216 -4.139  1.801  0.000
## bmi.26   4.817  2.925 -2.214  6.927  0.000
##
## $mean.z
##      bmi.0  bmi.2  bmi.8  bmi.14  bmi.26
## -0.417   2.313 -3.508 -0.913   2.282
```

```
##
## $summary
##          crmr crmr.se crmr.z crmr.pvalue ucrmr ucrmr.se
## cov    0.018  0.002  6.944          0 0.017  0.002
## mean    0.029  0.005  3.459          0 0.028  0.009
## total   0.022  0.003  5.299          0 0.021  0.004
```

From the value it is visible that there is no absolute value that is greater than 0.10 for the correlation residuals. Therefore, this value of the correlation residuals do not show the evidence of local misfit. However, for the standardized residuals, there are eight absolute values that are greater than 1.96. Therefore, the values of standardized residuals show the evidence of local misfit though correlation residuals do not show it.

Question 11: Fit a nonlinear curve model for BMI where a) the error variances for the BMI manifest variables are free and b) you ignore autocorrelation with your errors (`nlc.model`). Does this model fit better than `linear.free` model? (3 pts)

```
```{r}
library(lavaan)
line.mod <- "
define factors

int =~ 1*bmi.0 + 1*bmi.2 + 1*bmi.8 + 1*bmi.14 +1*bmi.26

slope =~ 0*bmi.0 + 2*bmi.2 + lam8*bmi.8 + lam14*bmi.14+lam26*bmi.26

estimate factor means
int ~ 1
slope ~ 1

estimate factor variance/covariance
int ~~ int + slope
slope ~~ slope

assume variances are not equivalent across time
bmi.0 ~~ bmi.0
bmi.2 ~~ bmi.2
bmi.8 ~~ bmi.8
bmi.14 ~~ bmi.14
bmi.26 ~~ bmi.26
```

"

```
nlc.model <- lavaan(model = line.mod, data=lab7)
```

```
fitMeasures(nlc.model, c("chisq", "df", "p", "rmsea", "cfi", "tli"))
```

...

```
Error in lav_fit_measures(object = object, fit.measures = fit.measures, :
lavaan ERROR: fit measures not available if model did not converge
```

No, this model does not fit better than linear.free model. We do not get any result from this model as model did not converge so that fit measures are not available.

**Question 12. Interpret the “intercepts” of the intercept and slope factors for the linear.free model (2 pts).**

```
summary(linear.free)
```

```
lavaan 0.6-3 ended normally after 110 iterations

Optimization method NLMINB
Number of free parameters 10

Number of observations 364

Estimator ML
Model Fit Test Statistic 132.661
Degrees of freedom 10
P-value (Chi-square) 0.000

Parameter Estimates:

Information Expected
Information saturated (h1) model Structured
Standard Errors Standard

Latent Variables:
Estimate Std.Err z-value P(>|z|)
int =~
bmi.0 1.000
bmi.2 1.000
bmi.8 1.000
bmi.14 1.000
bmi.26 1.000
slope =~
bmi.0 0.000
bmi.2 2.000
bmi.8 8.000
bmi.14 14.000
bmi.26 26.000
##
```



```
Covariances:
Estimate Std.Err z-value P(>|z|)
int ~~
slope -0.021 0.011 -1.887 0.059
##
Intercepts:
Estimate Std.Err z-value P(>|z|)
int 23.308 0.149 156.612 0.000
slope 0.029 0.004 7.301 0.000
.bmi.0 0.000
.bmi.2 0.000
.bmi.8 0.000
.bmi.14 0.000
.bmi.26 0.000
##
Variances:
Estimate Std.Err z-value P(>|z|)
int 7.957 0.598 13.313 0.000
slope 0.004 0.000 8.355 0.000
.bmi.0 0.264 0.038 6.899 0.000
.bmi.2 0.147 0.029 4.969 0.000
.bmi.8 0.800 0.067 11.885 0.000
.bmi.14 0.919 0.088 10.440 0.000
.bmi.26 1.524 0.218 6.987 0.000
```

### Interpret the “intercepts” of the intercept

The expected average BMI at the start of the intervention is 23.308. This is the average BMI at the start of the intervention.

### Interpret the “intercepts” of the slope

-0.5 this is a linear slope -- it's expected to increase 0.029 each month, not over the course of the first two months

The BMI, on average, was expected to increase 0.029 unit from the start of the intervention to the end of the 2 month after the initial intervention.

Question 13. Add negative affect (neg.aff) to your model as predictor of initial BMI (the intercept factor) and change in BMI (the slope factor) and then fit this model ( linear.na ) (2 pts)

```
library(lavaan)
line.mod <- "
define factors
int =~ 1*bmi.0 + 1*bmi.2 + 1*bmi.8 + 1*bmi.14 +1*bmi.26
slope =~ 0*bmi.0 + 2*bmi.2 + 8*bmi.8 + 14*bmi.14 +26*bmi.26
estimate factor means
int ~ 1+neg.aff
slope ~ 1+neg.aff
estimate factor variance/covariance
int ~~ int + slope
slope ~~ slope
```

```
assume variances are not equivalent across time
bmi.0 ~~ bmi.0
bmi.2 ~~ bmi.2
bmi.8 ~~ bmi.8
bmi.14 ~~ bmi.14
bmi.26 ~~ bmi.26
"

linear.na <- lavaan(model = line.mod, data=lab7)
fitMeasures(linear.na, c("chisq", "df", "p", "rmsea", "cfi", "tli", "srmr"))

chisq df rmsea cfi tli srmr
138.483 13.000 0.163 0.960 0.954 0.046
```

Question 14. Can you compare the `linear.na` model to the `linear.free` model? Why or why not? (1 pt)

```
anova(linear.free, linear.na)

Warning in lavTestLRT(object = new("lavaan", version = "0.6.3", call =
lavaan(model = line.mod, : lavaan WARNING: some models are based on a
different set of observed variables

Chi Square Difference Test
##
Df AIC BIC Chisq Chisq diff Df diff Pr(>Chisq)
linear.free 10 6141.7 6180.7 132.66
linear.na 13 6125.7 6172.5 138.48 5.8215 3 0.1206
```

The lavaan warning indicates that we can not compare the `linear.na` model to the `linear.free` model because `linear.na` model includes different observed variable such as negative affect compared to the 'linear.free' model.

Question 15: Based on global fit (chi-square test of model fit, RMSEA, CFI, and SRMR) is the `linear.na` a good model? (2 pts)

```
fitMeasures(linear.na, c("chisq", "df", "p", "rmsea", "cfi", "tli", "srmr"))

chisq df rmsea cfi tli srmr
138.483 13.000 0.163 0.960 0.954 0.046
```

The CFI value is 0.960 and the TLI is 0.954. 0.8 is way too generous for good fit on CFI/TLI. The RMSEA value is 0.163. RMSEA less than 0.06 or 0.08 are considered indicators of good fit. However, here the RMSEA value is greater than the acceptable range which does not indicate a good fit for model. The SRMR value less than 0.1 indicates a good fit and the value of our result is 0.046. The value of chisq with respect to df shows reasonable which indicates a good fit for model. The "larger" model with more freely estimated parameters and adding predictor fits the data better than the "smaller" model in which the parameters in question are fixed. Therefore, this model is a comparatively good model though RMSEA value does not show overall good fit.

-0.5 The chi-square is highly significant which suggest misfit, not good fit

Question 16: Examine the correlation residuals and the mean residuals for the **linear.na** model. Do you see evidence of local misfit? (3 pts)

```
lavResiduals(linear.na, type="cor")
```

```
$type
[1] "cor.bollen"
##
$cov
bmi.0 bmi.2 bmi.8 bmi.14 bmi.26 neg.ff
bmi.0 0.000
bmi.2 0.002 0.000
bmi.8 -0.008 0.003 0.000
bmi.14 -0.023 -0.013 0.010 0.000
bmi.26 0.031 0.017 -0.017 0.030 0.000
neg.aff -0.016 0.008 0.010 0.009 -0.016 0.000
##
$mean
bmi.0 bmi.2 bmi.8 bmi.14 bmi.26 neg.aff
-0.003 0.010 -0.056 -0.012 0.030 0.000
##
$cov.z
bmi.0 bmi.2 bmi.8 bmi.14 bmi.26 neg.ff
bmi.0 0.000
bmi.2 4.223 0.000
bmi.8 -2.496 1.229 0.000
bmi.14 -5.257 -3.974 1.789 0.000
bmi.26 4.802 3.042 -2.063 6.852 0.000
neg.aff -2.161 1.753 0.602 0.708 -1.273 0.000
##
$mean.z
bmi.0 bmi.2 bmi.8 bmi.14 bmi.26 neg.aff
-0.400 2.305 -3.498 -0.894 2.280 0.000
##
$summary
crmr crmr.se crmr.z crmr.pvalue ucrmr ucrmr.se
cov 0.016 0.003 3.243 0.001 0.015 0.003
mean 0.029 0.006 3.445 0.000 0.028 0.009
total 0.020 0.003 4.600 0.000 0.019 0.004
```

From the value it is visible that there is no absolute value that is greater than 0.10 for the correlation residuals. Therefore, this value of the correlation residuals do not show the evidence of local misfit. However, for the standardized residuals, there are nine absolute values that are greater than 1.96. Therefore, the values of standardized residuals show the evidence of local misfit though correlation residuals do not show it.

Question 17. 1) Please interpret the relationship between negative affect and the intercept factor and negative affect slope factor regardless of statistical significance for the **linear.na**. 2a) Report the 95% confidence interval for the effect of negative affect on the intercept factor only. Recall that the confidence interval indicate a range of values that are compatible with your data given your assumptions. 2b) Would you say any of the potential effects in the 95% CI that are compatible with your data are practically large? I am not referring to the width of this interval, but given that BMI typically ranges between 18 and 25, would you say that any of the compatible effects would be large (i.e., are there values in the interval that you would consider representing a large practical impact of negative affect on BMI at the beginning of the intervention)? (3 pts)

```
summary(linear.na, standardized=TRUE)
```

```
lavaan 0.6-3 ended normally after 87 iterations
##
Optimization method NLMINB
Number of free parameters 12
##
Number of observations 364
##
Estimator ML
Model Fit Test Statistic 138.483
Degrees of freedom 13
P-value (Chi-square) 0.000
##
Parameter Estimates:
##
Information Expected
Information saturated (h1) model Structured
Standard Errors Standard
##
Latent Variables:
Estimate Std.Err z-value P(>|z|) Std.lv Std.all
int =~
bmi.0 1.000 2.819 0.984
bmi.2 1.000 2.819 0.995
bmi.8 1.000 2.819 0.959
bmi.14 1.000 2.819 0.940
bmi.26 1.000 2.819 0.854
slope =~
bmi.0 0.000 0.000 0.000
bmi.2 2.000 0.121 0.043
bmi.8 8.000 0.485 0.165
bmi.14 14.000 0.849 0.283
bmi.26 26.000 1.576 0.477
##
```

```

Regressions:
Estimate Std.Err z-value P(>|z|) Std.lv Std.all
int ~
neg.aff 0.367 0.220 1.672 0.094 0.130 0.088
slope ~
neg.aff 0.022 0.006 3.914 0.000 0.366 0.247
##
Covariances:
Estimate Std.Err z-value P(>|z|) Std.lv Std.all
.int ~~
.slope -0.025 0.011 -2.243 0.025 -0.149 -0.149
##
Intercepts:
Estimate Std.Err z-value P(>|z|) Std.lv Std.all
.int 22.598 0.449 50.325 0.000 8.018 8.018
.slope -0.014 0.012 -1.239 0.215 -0.237 -0.237
.bmi.0 0.000 0.000 0.000 0.000 0.000
.bmi.2 0.000 0.000 0.000 0.000 0.000
.bmi.8 0.000 0.000 0.000 0.000 0.000
.bmi.14 0.000 0.000 0.000 0.000 0.000
.bmi.26 0.000 0.000 0.000 0.000 0.000
##
Variances:
Estimate Std.Err z-value P(>|z|) Std.lv Std.all
.int 7.883 0.592 13.311 0.000 0.992 0.992
.slope 0.003 0.000 8.112 0.000 0.939 0.939
.bmi.0 0.256 0.038 6.797 0.000 0.256 0.031
.bmi.2 0.152 0.029 5.151 0.000 0.152 0.019
.bmi.8 0.796 0.067 11.878 0.000 0.796 0.092
.bmi.14 0.920 0.088 10.484 0.000 0.920 0.102
.bmi.26 1.555 0.217 7.169 0.000 1.555 0.143

```

#### interpret the relationship between negative affect and the intercept factor

Each one unit increase or difference in negative affect is associated with a 0.367 predicted increase or difference in BMI at the start of the intervention (MONTH 0). This means that persons with higher negative affect has higher initial BMI, on average, at the start of the intervention (MONTH 0) and that is 0.367 unit. In the case of standardized result, each one sd increase or difference in negative affect is associated with a 0.088 sd predicted increase or difference in BMI at the start of the intervention (MONTH 0)

#### interpret the relationship between negative affect and the slope factor

Also, each one unit increase or difference in negative affect is associated with a 0.022 unit linear increase of BMI, on average over time and that is 0.022. It means that the rates of linear increase in average BMI is generally higher for persons with higher negative affect. Also, each one sd increase or difference in negative affect is associated with a 0.247 sd linear increase of BMI, on average over time

```
summary(linear.na, ci=TRUE)
```

```
lavaan 0.6-3 ended normally after 87 iterations
```

```
##
```

```
Optimization method NLMINB
```

```
Number of free parameters 12
```

```
##
```

```
Number of observations 364
```

```
##
```

```
Estimator ML
```

```
Model Fit Test Statistic 138.483
```

```
Degrees of freedom 13
```

```
P-value (Chi-square) 0.000
```

```
##
```

```
Parameter Estimates:
```

```
##
```

```
Information Expected
```

```
Information saturated (h1) model Structured
```

```
Standard Errors Standard
```

```
##
```

```
Latent Variables:
```

```
Estimate Std.Err z-value P(>|z|) ci.lower ci.upper
```

```
int =~
```

```
bmi.0 1.000 1.000 1.000
```

```
bmi.2 1.000 1.000 1.000
```

```
bmi.8 1.000 1.000 1.000
```

```
bmi.14 1.000 1.000 1.000
```

```
bmi.26 1.000 1.000 1.000
```

```
slope =~
```

```
bmi.0 0.000 0.000 0.000
```

```
bmi.2 2.000 2.000 2.000
```

```
bmi.8 8.000 8.000 8.000
```

```
bmi.14 14.000 14.000 14.000
```

```
bmi.26 26.000 26.000 26.000
```

```
##
```

```
Regressions:
```

```
Estimate Std.Err z-value P(>|z|) ci.lower ci.upper
```

```
int ~
```

```
neg.aff 0.367 0.220 1.672 0.094 -0.063 0.797
```

```
slope ~
```

```
neg.aff 0.022 0.006 3.914 0.000 0.011 0.033
```

```
##
```

```
Covariances:
```

```
Estimate Std.Err z-value P(>|z|) ci.lower ci.upper
```

```
.int ~~
```

```
.slope -0.025 0.011 -2.243 0.025 -0.046 -0.003
```

```
##
```

```
Intercepts:
```

```
Estimate Std.Err z-value P(>|z|) ci.lower ci.upper
```

```
.int 22.598 0.449 50.325 0.000 21.718 23.478
```

```
.slope -0.014 0.012 -1.239 0.215 -0.037 0.008
.bmi.0 0.000 0.000 0.000 0.000
.bmi.2 0.000 0.000 0.000 0.000
.bmi.8 0.000 0.000 0.000 0.000
.bmi.14 0.000 0.000 0.000 0.000
.bmi.26 0.000 0.000 0.000 0.000
##
Variances:
Estimate Std.Err z-value P(>|z|) ci.lower ci.upper
.int 7.883 0.592 13.311 0.000 6.722 9.043
.slope 0.003 0.000 8.112 0.000 0.003 0.004
.bmi.0 0.256 0.038 6.797 0.000 0.182 0.329
.bmi.2 0.152 0.029 5.151 0.000 0.094 0.209
.bmi.8 0.796 0.067 11.878 0.000 0.665 0.927
.bmi.14 0.920 0.088 10.484 0.000 0.748 1.092
.bmi.26 1.555 0.217 7.169 0.000 1.130 1.980
```

The 95% CI for the negative effect on intercept factor is [-0.063,0.797]

Yes, in my view, there are values in the interval that I would consider representing a large practical impact of negative affect on BMI at the beginning of the intervention. For example, the upper ci value of negative affect is 0.797. It means that each one unit increase or difference in negative affect is associated with a 0.797 unit predicted increase or difference in BMI at the start of the intervention (MONTH 0). It means that for each unit increase of the negative affect has a large impact on BMI on average at the start of the intervention if we consider the BMI range normal from 18 to 25. This increase is large because other numerous factors are also accountable for higher BMI.

Question 18. Looking at your modification indices, do you think adding autocorrelations between the error terms of adjacent measurements of BMI (e.g., month 0 and month 2, month 2 and month 8, and so on) would fix global model fit? (2 pts)

`modificationindices(linear.na,sort.=TRUE)`

```
lhs op rhs mi epc sepc.lv sepc.all sepc.nox
38 bmi.8 ~~ bmi.26 45.761 -0.626 -0.626 -0.563 -0.563
37 bmi.8 ~~ bmi.14 25.161 0.289 0.289 0.337 0.337
30 bmi.0 ~~ bmi.2 24.838 0.423 0.423 2.146 2.146
9 slope == bmi.14 23.904 6.505 0.394 0.131 0.131
33 bmi.0 ~~ bmi.26 22.505 0.277 0.277 0.439 0.439
3 int == bmi.8 16.006 -0.009 -0.024 -0.008 -0.008
32 bmi.0 ~~ bmi.14 12.535 -0.139 -0.139 -0.286 -0.286
26 bmi.8 ~1 12.287 -0.176 -0.176 -0.060 -0.060
31 bmi.0 ~~ bmi.8 10.187 -0.131 -0.131 -0.290 -0.290
8 slope == bmi.8 9.115 -2.755 -0.167 -0.057 -0.057
39 bmi.14 ~~ bmi.26 8.941 0.510 0.510 0.427 0.427
2 int == bmi.2 7.631 0.004 0.010 0.004 0.004
5 int == bmi.26 7.550 0.013 0.036 0.011 0.011
```

## 10	slope =~	bmi.26	6.726	-6.933	-0.420	-0.127	-0.127
## 28	bmi.26 ~1		6.341	0.271	0.271	0.082	0.082
## 25	bmi.2 ~1		4.738	0.066	0.066	0.023	0.023
## 34	bmi.2 ~~	bmi.8	1.452	0.047	0.047	0.135	0.135
## 36	bmi.2 ~~	bmi.26	0.979	-0.050	-0.050	-0.102	-0.102
## 4	int =~	bmi.14	0.777	-0.002	-0.006	-0.002	-0.002
## 27	bmi.14 ~1		0.668	-0.049	-0.049	-0.016	-0.016
## 1	int =~	bmi.0	0.594	-0.001	-0.003	-0.001	-0.001
## 35	bmi.2 ~~	bmi.14	0.286	-0.020	-0.020	-0.052	-0.052
## 24	bmi.0 ~1		0.165	-0.014	-0.014	-0.005	-0.005
## 6	slope =~	bmi.0	0.133	0.277	0.017	0.006	0.006
## 7	slope =~	bmi.2	0.006	0.050	0.003	0.001	0.001

Based on the modification indices, I think that adding correlations between the error terms of adjacent measurements of BMI would fix global fit. When the modification indices are greater than 3.84, will significantly improve the global model fit. Here we have found that the value of Modification indices for adjacent measurements of BMI such as bmi.0 and bmi.2, bmi.8 and bmi.14, bmi.14 and bmi.26 are greater than 3.84. Therefore, I think adding autocorrelations between the error terms of adjacent measurements of BMI would fix global model fit.