

Name Jerry

Student Number _____

STA 431 Quiz 10

1. (7 points) On the left is a centered original model, and on the right is a familiar surrogate model. Let

$$\begin{aligned} d_1 &= \lambda_1 F + e_1 & d_1 &= F' + e'_1 \\ d_2 &= \lambda_2 F + e_2 & d_2 &= \lambda'_2 F' + e'_2 \\ d_3 &= \lambda_3 F + e_3 & d_3 &= \lambda'_3 F' + e'_3 \end{aligned}$$

where $F \sim N(0, \phi)$, e_1 , e_2 and e_3 are normal and independent of F and each other with expected value zero, $Var(e_1) = \omega_1$, $Var(e_2) = \omega_2$, and $Var(e_3) = \omega_3$. The λ_j (factor loadings) are non-zero constants.

- (a) In terms of the original model, what is F' ?

$$F' = \lambda_1 F$$

- (b) Fill in the table. For each symbol from the surrogate model, write what it equals in terms of the original model. Show any necessary work below the table.

Surrogate	ϕ'	λ'_2	λ'_3	ω'_1	ω'_2	ω'_3
Original	$\lambda_1^2 \phi$	λ_2 / λ_1	λ_3 / λ_1	ω_1	ω_2	ω_3

$$\phi' = Var(F') = Var(\lambda_1 F) = \lambda_1^2 \phi$$

$$d_2 = \lambda_2 F + e_2 = \left(\frac{\lambda_2}{\lambda_1} \right) (\lambda_1 F) + e_2$$

2. (3 points) For the R part of the assignment (last question), you fit a 2-factor model to the poverty data. Give an estimate and a 95% confidence interval for the correlation between factors. The answer is three numbers from your printout. Write the numbers in the space below and circle them on your printout.

$$\hat{\phi}_{12} = 0.961, \text{ CI} = (0.901, 1.020)$$

Please attach your printout to the quiz paper. The printout should show your *complete* R input and output. Make sure your name and student number appear on the printout.

R version 4.2.3 (2023-03-15) -- "Shortstop Beagle"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin17.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.79 (8198) x86_64-apple-darwin17.0]

[Workspace restored from /Users/brunner/.RData]
[History restored from /Users/brunner/.Rapp.history]

```
> # CFA of poverty data
> # install.packages("lavaan", dependencies = TRUE) # Only need to do this once
> library(lavaan)
This is lavaan 0.6-15
lavaan is FREE software! Please report any bugs.
>
> rm(list=ls())
> # pov = read.table("poverty.data.txt") # Local copy
> pov = read.table("https://www.utstat.toronto.edu/brunner/data/illegal/poverty.data.txt",
na.strings = ".")
>
> head(pov)
  V1  V2  V3  V4  V5  V6 V7          V8
1 24.7  5.7 30.8 69.6 75.5 600 1      Albania
2 12.5 11.9 14.4 68.3 74.7 2250 1      Bulgaria
3 13.4 11.7 11.3 71.8 77.7 2980 1      Czechoslovakia
4 12.0 12.4  7.6 69.8 75.9  NA  1      Former_E._Germany
5 11.6 13.4 14.8 65.4 73.8 2780 1      Hungary
6 14.3 10.2 16.0 67.2 75.7 1690 1      Poland
> colnames(pov) = c("birthrate", "deathrate", "infmort", "lifexM", "lifexF", "gnp",
"continent", "country")
>
> # Data transformations
> pov = within(pov, {
+ LifeEx = (lifexM+lifexF)/2
+ gnp1000 = gnp/1000
+ }) # End within pov
>
> summary(pov)
  birthrate      deathrate      infmort      lifexM      lifexF
Min.   : 9.70   Min.   : 2.20   Min.   : 4.5   Min.   :38.10   Min.   :41.20
1st Qu.:14.50   1st Qu.: 7.80   1st Qu.:13.1   1st Qu.:55.80   1st Qu.:57.50
```

Median :29.00	Median : 9.50	Median : 43.0	Median :63.70	Median :67.80
Mean :29.23	Mean :10.84	Mean : 54.9	Mean :61.49	Mean :66.15
3rd Qu.:42.20	3rd Qu.:12.50	3rd Qu.: 83.0	3rd Qu.:68.60	3rd Qu.:75.40
Max. :52.20	Max. :25.00	Max. :181.6	Max. :75.90	Max. :81.80

gnp	continent	country	gnp1000	LifeEx
Min. : 80	Min. :1.000	Length:97	Min. : 0.080	Min. :39.65
1st Qu.: 475	1st Qu.:3.000	Class :character	1st Qu.: 0.475	1st Qu.:56.25
Median : 1690	Median :4.000	Mode :character	Median : 1.690	Median :65.80
Mean : 5741	Mean :3.948		Mean : 5.741	Mean :63.82
3rd Qu.: 7325	3rd Qu.:6.000		3rd Qu.: 7.325	3rd Qu.:71.55
Max. :34064	Max. :6.000		Max. :34.064	Max. :78.85
NA's :6			NA's :6	

```

>
> # Use LifeEx, infmort, birthrate, gnp1000
>
> mod = "Health =~ LifeEx + infmort
+       Wealth =~ gnp1000 + birthrate "
>
> fit1 = cfa(mod,std.lv=TRUE, data=pov) # Standardizing latent variables
> summary(fit1)
lavaan 0.6.15 ended normally after 47 iterations

```

Estimator	ML	
Optimization method	NLMINB	
Number of model parameters	9	
	Used	Total
Number of observations	91	97

Model Test User Model:

Test statistic	1.098
Degrees of freedom	1
P-value (Chi-square)	0.295

Parameter Estimates:

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

Latent Variables:

	Estimate	Std.Err	z-value	P(> z)
Health =~				
LifeEx	10.256	0.775	13.225	0.000
infmort	-44.025	3.577	-12.309	0.000
Wealth =~				
gnp1000	5.454	0.758	7.196	0.000
birthrate	-12.648	1.140	-11.091	0.000

Covariances:

	Estimate	Std.Err	z-value	P(> z)
Health ~~				
Wealth	0.961	0.030	31.660	0.000

Variances:

	Estimate	Std.Err	z-value	P(> z)
.LifeEx	1.432	1.815	0.789	0.430
.infmort	182.139	42.803	4.255	0.000
.gnp1000	35.041	5.484	6.389	0.000
.birthrate	25.637	10.190	2.516	0.012
Health	1.000			
Wealth	1.000			

> # Fairly close to proc calis values from 2013, very close to 2017

> # Ah I was using vardef=n in 2017.

> parameterEstimates(fit1)

	lhs	op	rhs	est	se	z	pvalue	ci.lower	ci.upper
1	Health	==	LifeEx	10.256	0.775	13.225	0.000	8.736	11.775
2	Health	==	infmort	-44.025	3.577	-12.309	0.000	-51.035	-37.015
3	Wealth	==	gnp1000	5.454	0.758	7.196	0.000	3.968	6.940
4	Wealth	==	birthrate	-12.648	1.140	-11.091	0.000	-14.883	-10.413
5	LifeEx	==	LifeEx	1.432	1.815	0.789	0.430	-2.125	4.988
6	infmort	==	infmort	182.139	42.803	4.255	0.000	98.247	266.032
7	gnp1000	==	gnp1000	35.041	5.484	6.389	0.000	24.292	45.791
8	birthrate	==	birthrate	25.637	10.190	2.516	0.012	5.665	45.610
9	Health	==	Health	1.000	0.000	NA	NA	1.000	1.000
10	Wealth	==	Wealth	1.000	0.000	NA	NA	1.000	1.000
11	Health	==	Wealth	0.961	0.030	31.660	0.000	0.901	1.020

>

> fit2 = cfa(mod, data=pov) # Setting first factor loading to one (default)

> summary(fit2, standardized=TRUE)

lavaan 0.6.15 ended normally after 109 iterations

Estimator	ML	
Optimization method	NLMINB	
Number of model parameters	9	
	Used	Total
Number of observations	91	97

Model Test User Model:

Test statistic	1.098
Degrees of freedom	1
P-value (Chi-square)	0.295

Parameter Estimates:

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

Latent Variables:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
Health ==						
LifeEx	1.000				10.256	0.993
infmort	-4.293	0.164	-26.155	0.000	-44.025	-0.956
Wealth ==						
gnp1000	1.000				5.454	0.678
birthrate	-2.319	0.294	-7.875	0.000	-12.648	-0.928

Covariances:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
Health ~~ Wealth	53.733	10.371	5.181	0.000	0.961	0.961

Variances:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.LifeEx	1.432	1.815	0.789	0.430	1.432	0.013
.infmort	182.139	42.803	4.255	0.000	182.139	0.086
.gnp1000	35.041	5.484	6.389	0.000	35.041	0.541
.birthrate	25.637	10.190	2.516	0.012	25.637	0.138
Health	105.176	15.906	6.613	0.000	1.000	1.000
Wealth	29.746	8.268	3.598	0.000	1.000	1.000

```
> # Compare Sigma(theta-hat) matrices. Illustrated this in lecture.
> fitted(fit1)
```

```
$cov
      LifeEx  infmrt  gnp1000  brthrt
LifeEx  106.608
infmort -451.501 2120.346
gnp1000  53.733 -230.666  64.788
birthrate -124.607 534.911 -68.981 185.604
```

```
> fitted(fit2)
```

```
$cov
      LifeEx  infmrt  gnp1000  brthrt
LifeEx  106.608
infmort -451.501 2120.346
gnp1000  53.733 -230.666  64.788
birthrate -124.607 534.911 -68.981 185.604
```

```
>
> # lambda2/lambda1 (sqrt{phi11} cancels)
> -44.025/10.256
[1] -4.292609
>
> # 15e (v) Proportion of variance in infant mortality rate explained by Health
> (-0.956)^2
[1] 0.913936
>
```

```
> ##### Single-factor model
> singlemod = "Wealth =~ gnp1000 + birthrate + LifeEx + infmort"
> fit3 = cfa(singlemod, data=pov)
> summary(fit3, standardized=TRUE)
```

lavaan 0.6.15 ended normally after 66 iterations

Estimator	ML	
Optimization method	NLMINB	
Number of model parameters	8	
	Used	Total
Number of observations	91	97

Model Test User Model:

Test statistic	2.823
Degrees of freedom	2
P-value (Chi-square)	0.244

Parameter Estimates:

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

Latent Variables:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
Wealth =~						
gnp1000	1.000				5.262	0.654
birthrate	-2.313	0.308	-7.509	0.000	-12.172	-0.893
LifeEx	1.947	0.241	8.092	0.000	10.243	0.992
infmort	-8.374	1.058	-7.917	0.000	-44.062	-0.957

Variances:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.gnp1000	37.103	5.561	6.672	0.000	37.103	0.573
.birthrate	37.453	6.083	6.157	0.000	37.453	0.202
.LifeEx	1.679	1.709	0.982	0.326	1.679	0.016
.infmort	178.867	41.144	4.347	0.000	178.867	0.084
Wealth	27.685	7.916	3.497	0.000	1.000	1.000

```
>
> # With factanal, get same factor loadings (reflected), but a slightly different fit test.
Probably using Lawley's original test instead of G^2.
> dat = pov[c(1,3,9,10)]; colnames(dat)
[1] "birthrate" "infmort" "gnp1000" "LifeEx"
> factanal(na.omit(dat), factors=1)
```

Call:

```
factanal(x = na.omit(dat), factors = 1)
```

Uniquenesses:

birthrate	infmort	gnp1000	LifeEx
0.202	0.084	0.573	0.016

Loadings:

	Factor1
birthrate	0.893
infmort	0.957
gnp1000	-0.654
LifeEx	-0.992

	Factor1
SS loadings	3.125
Proportion Var	0.781

Test of the hypothesis that 1 factor is sufficient.
The chi square statistic is 2.7 on 2 degrees of freedom.
The p-value is 0.259

```
>
>
```