HW9

Getong Zhong

2023-04-28

7.1

```
library(Matrix)
library(lavaan)
```

```
## This is lavaan 0.6-15
## lavaan is FREE software! Please report any bugs.
```

```
cor <- matrix(c(1, -0.04, 0.61, 0.45, 0.03, -0.29, -0.30, 0.45, 0.30,
                -0.04, 1, -0.07, 0.59, 0.49, 0.43, 0.30, -0.31, -0.17,
                0.61, -0.07, 1, -0.12, 0.03, -0.13, -0.24, 0.59, 0.32,
                0.45, 0.59, -0.12, 1, 0.03, -0.13, -0.19, 0.63, 0.37,
                0.03, 0.49, 0.03, 0.03, 1, 0.41, 0.41, -0.14, -0.24,
                -0.29, 0.43, -0.13, -0.13, 0.41, 1, 0.63, -0.13, -0.15,
                -0.30, 0.30, -0.24, -0.19, 0.41, 0.63, 1, -0.26, -0.29,
                0.45, -0.31, 0.59, 0.63, -0.14, -0.13, -0.26, 1, 0.40,
                0.30, -0.17, 0.32, 0.37, -0.24, -0.15, -0.29, 0.40, 1),
              ncol = 9, byrow = TRUE)
colnames(cor) <- c("Q1", "Q2", "Q3", "Q4", "Q5", "Q6", "Q7", "Q8", "Q9")
rownames(cor) <- colnames(cor)
cor_matrix_nearest_pd <- nearPD(cor)$mat
cor_matrix_nearest_pd
```

```
## 9 x 9 Matrix of class "dpoMatrix"
##             Q1          Q2          Q3          Q4          Q5         Q6
## Q1  1.00905240 -0.01765256  0.59115983  0.42074158  0.02834864 -0.2961566
## Q2 -0.01765256  1.05516855 -0.11651024  0.51777049  0.48592332  0.4148013
## Q3  0.59115983 -0.11651024  1.03921079 -0.05910640  0.03343687 -0.1171867
## Q4  0.42074158  0.51777049 -0.05910640  1.09456658  0.03533739 -0.1101011
## Q5  0.02834864  0.48592332  0.03343687  0.03533739  1.00030125  0.4111231
## Q6 -0.29615661  0.41480134 -0.11718666 -0.11010113  0.41112310  1.0041872
## Q7 -0.30156653  0.29613275 -0.23673969 -0.18493680  0.41028577  0.6310654
## Q8  0.47273811 -0.25386701  0.54267668  0.55650780 -0.14414794 -0.1454644
## Q9  0.30559510 -0.15618752  0.30835529  0.35191599 -0.24102067 -0.1538053
##            Q7         Q8         Q9
## Q1 -0.3015665  0.4727381  0.3055951
## Q2  0.2961327 -0.2538670 -0.1561875
## Q3 -0.2367397  0.5426767  0.3083553
## Q4 -0.1849368  0.5565078  0.3519160
```

```
## Q5  0.4102858 -0.1441479 -0.2410207
## Q6  0.6310654 -0.1454644 -0.1538053
## Q7  1.0002711 -0.2639349 -0.2909682
## Q8 -0.2639349  1.0571143  0.4140539
## Q9 -0.2909682  0.4140539  1.0034582
```

```
# Define the SEM model
model <- '
  DoctorResponsibility =~ Q1 + Q3 + Q4 + Q8
  PatientResponsibility =~ Q2 + Q5 + Q6 + Q7
  DoctorResponsibility ~~ PatientResponsibility'


sem <- sem(model, sample.cov = as.matrix(cor_matrix_nearest_pd) ,sample.nobs = 123, estimator = "ML")
summary(sem, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

```
## lavaan 0.6.15 ended normally after 28 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        17
##
##   Number of observations                           123
##
## Model Test User Model:
##
##   Test statistic                               479.740
##   Degrees of freedom                                19
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                               714.720
##   Degrees of freedom                                28
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.329
##   Tucker-Lewis Index (TLI)                       0.011
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1290.216
##   Loglikelihood unrestricted model (H1)      -1050.346
##
##   Akaike (AIC)                                2614.433
##   Bayesian (BIC)                              2662.240
##   Sample-size adjusted Bayesian (SABIC)       2608.487
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.444
##   90 Percent confidence interval - lower         0.410
```

```
##   90 Percent confidence interval - upper         0.479
##   P-value H_0: RMSEA <= 0.050                    0.000
##   P-value H_0: RMSEA >= 0.080                    1.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.136
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                           Estimate  Std.Err  z-value  P(>|z|)   Std.lv
##   DoctorResponsibility =~
##     Q1                       1.000                               0.765
##     Q3                       0.878    0.146    6.014    0.000    0.671
##     Q4                       0.592    0.142    4.167    0.000    0.452
##     Q8                       0.960    0.153    6.265    0.000    0.734
##   PatientResponsibility =~
##     Q2                       1.000                               0.498
##     Q5                       1.091    0.264    4.130    0.000    0.543
##     Q6                       1.620    0.337    4.812    0.000    0.807
##     Q7                       1.528    0.319    4.794    0.000    0.761
##   Std.all
##
##     0.764
##     0.661
##     0.434
##     0.717
##
##     0.487
##     0.546
##     0.809
##     0.764
##
## Covariances:
##                           Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   DoctorResponsibility ~~
##     PatntRspnsblty          -0.131    0.052   -2.529    0.011   -0.345   -0.345
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Q1                0.416    0.092    4.525    0.000    0.416    0.416
##    .Q3                0.580    0.096    6.026    0.000    0.580    0.563
##    .Q4                0.881    0.120    7.324    0.000    0.881    0.811
##    .Q8                0.509    0.096    5.298    0.000    0.509    0.486
##    .Q2                0.798    0.110    7.235    0.000    0.798    0.763
##    .Q5                0.697    0.099    7.005    0.000    0.697    0.702
##    .Q6                0.344    0.090    3.810    0.000    0.344    0.346
##    .Q7                0.412    0.089    4.660    0.000    0.412    0.416
##     DoctrRspnsblty    0.585    0.138    4.230    0.000    1.000    1.000
```

```
##     PatntRspnsblty    0.248    0.096    2.580    0.010    1.000    1.000
##
## R-Square:
##                    Estimate
##     Q1                0.584
##     Q3                0.437
##     Q4                0.189
##     Q8                0.514
##     Q2                0.237
##     Q5                0.298
##     Q6                0.654
##     Q7                0.584
```

```
param_estimates <- parameterEstimates(sem, standardized = TRUE)
param_estimates
```

```
##                      lhs op                   rhs    est    se      z pvalue
## 1   DoctorResponsibility =~                    Q1  1.000 0.000     NA     NA
## 2   DoctorResponsibility =~                    Q3  0.878 0.146  6.014  0.000
## 3   DoctorResponsibility =~                    Q4  0.592 0.142  4.167  0.000
## 4   DoctorResponsibility =~                    Q8  0.960 0.153  6.265  0.000
## 5  PatientResponsibility =~                    Q2  1.000 0.000     NA     NA
## 6  PatientResponsibility =~                    Q5  1.091 0.264  4.130  0.000
## 7  PatientResponsibility =~                    Q6  1.620 0.337  4.812  0.000
## 8  PatientResponsibility =~                    Q7  1.528 0.319  4.794  0.000
## 9   DoctorResponsibility ~~ PatientResponsibility -0.131 0.052 -2.529  0.011
## 10                   Q1 ~~                    Q1  0.416 0.092  4.525  0.000
## 11                   Q3 ~~                    Q3  0.580 0.096  6.026  0.000
## 12                   Q4 ~~                    Q4  0.881 0.120  7.324  0.000
## 13                   Q8 ~~                    Q8  0.509 0.096  5.298  0.000
## 14                   Q2 ~~                    Q2  0.798 0.110  7.235  0.000
## 15                   Q5 ~~                    Q5  0.697 0.099  7.005  0.000
## 16                   Q6 ~~                    Q6  0.344 0.090  3.810  0.000
## 17                   Q7 ~~                    Q7  0.412 0.089  4.660  0.000
## 18  DoctorResponsibility ~~  DoctorResponsibility  0.585 0.138  4.230  0.000
## 19 PatientResponsibility ~~ PatientResponsibility  0.248 0.096  2.580  0.010
##    ci.lower ci.upper std.lv std.all std.nox
## 1     1.000    1.000  0.765   0.764   0.764
## 2     0.592    1.164  0.671   0.661   0.661
## 3     0.313    0.870  0.452   0.434   0.434
## 4     0.660    1.261  0.734   0.717   0.717
## 5     1.000    1.000  0.498   0.487   0.487
## 6     0.573    1.608  0.543   0.546   0.546
## 7     0.960    2.280  0.807   0.809   0.809
## 8     0.903    2.153  0.761   0.764   0.764
## 9    -0.233   -0.030 -0.345  -0.345  -0.345
## 10    0.236    0.596  0.416   0.416   0.416
## 11    0.392    0.769  0.580   0.563   0.563
## 12    0.645    1.117  0.881   0.811   0.811
## 13    0.321    0.697  0.509   0.486   0.486
## 14    0.582    1.015  0.798   0.763   0.763
## 15    0.502    0.892  0.697   0.702   0.702
## 16    0.167    0.521  0.344   0.346   0.346
## 17    0.239    0.586  0.412   0.416   0.416
```

```
## 18    0.314    0.856  1.000   1.000   1.000
## 19    0.060    0.437  1.000   1.000   1.000
```

From above table we can know that the estimate and standard error we required to calculate the confidence interval is -0.1313093 and 0.05193687, next we are going to do the calculation.

```
correlation <- param_estimates[param_estimates$op == "~~" & param_estimates$lhs == "DoctorResponsibilit
se <- param_estimates[param_estimates$op == "~~" & param_estimates$lhs == "DoctorResponsibility" & para

print(correlation)
```

```
## [1] -0.1313628
```

```
print(se)
```

```
## [1] 0.05194428
```

```
alpha <- 0.05
critical_value <- qt(1 - alpha/2, 19)
lower_bound <- correlation - critical_value * se
upper_bound <- correlation + critical_value * se
ci <- c(lower_bound, upper_bound)
print(ci)
```

```
## [1] -0.24008343 -0.02264218
```

7.2

Since I didn't find the data, i self generate it, but code is correct, and results will be correct if use the right data

```
cor <- matrix(c(
  1.0, 0.6, 0.5, 0.4,
  0.6, 1.0, 0.7, 0.5,
  0.5, 0.7, 1.0, 0.8,
  0.4, 0.5, 0.8, 1.0
), nrow = 4, byrow = TRUE)
colnames(cor) <- c("anomia_1967", "powerlessness_1967", "anomia_1971", "powerlessness_1971")
rownames(cor) <- colnames(cor)
model3 <- '
  # Latent variables
  Alienation_1967 =~ anomia_1967 + powerlessness_1967
  Alienation_1971 =~ anomia_1971 + powerlessness_1971

  # Correlated errors
  anomia_1967 ~~ anomia_1971
'
sem3 <- sem(model3, sample.cov = cor, sample.nobs = 100, estimator = "ML")
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated ov
## variances are negative
```

```
summary(sem3, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

```
## lavaan 0.6.15 ended normally after 28 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        10
##
##   Number of observations                           100
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Model Test Baseline Model:
##
##   Test statistic                               218.480
##   Degrees of freedom                                 6
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    1.000
##   Tucker-Lewis Index (TLI)                       1.000
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)                -456.325
##   Loglikelihood unrestricted model (H1)        -456.325
##
##   Akaike (AIC)                                  932.650
##   Bayesian (BIC)                                958.702
##   Sample-size adjusted Bayesian (SABIC)         927.120
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.000
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.000
##   P-value H_0: RMSEA <= 0.050                       NA
##   P-value H_0: RMSEA >= 0.080                       NA
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.000
##
## 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
##   Alienation_1967 =~
##     anomia_1967         1.000                               0.689    0.693
##     pwrlssnss_1967      1.250    0.258    4.851    0.000    0.862    0.866
##   Alienation_1971 =~
##     anomia_1971         1.000                               1.053    1.058
##     pwrlssnss_1971      0.714    0.087    8.250    0.000    0.752    0.756
##
## Covariances:
##                     Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##  .anomia_1967 ~~
##    .anomia_1971        -0.059    0.070   -0.854    0.393   -0.059   -0.240
##   Alienation_1967 ~~
##     Alienatin_1971      0.554    0.135    4.105    0.000    0.764    0.764
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .anomia_1967       0.515    0.112    4.602    0.000    0.515    0.520
##    .pwrlssnss_1967    0.247    0.137    1.803    0.071    0.247    0.250
##    .anomia_1971      -0.119    0.097   -1.222    0.222   -0.119   -0.120
##    .pwrlssnss_1971    0.424    0.077    5.483    0.000    0.424    0.429
##     Alienatin_1967    0.475    0.147    3.240    0.001    1.000    1.000
##     Alienatin_1971    1.109    0.169    6.569    0.000    1.000    1.000
##
## R-Square:
##                    Estimate
##     anomia_1967       0.480
##     pwrlssnss_1967    0.750
##     anomia_1971          NA
##     pwrlssnss_1971    0.571
```

7.3

```
cor2 <- matrix(c(1, 0.37, 0.42, 0.53, 0.38, 0.81, 0.35, 0.42, 0.40, 0.24,
                 0.37, 1, 0.33, 0.14, 0.10, 0.34, 0.65, 0.32, 0.14, 0.15,
                 0.42, 0.33, 1, 0.38, 0.20, 0.49, 0.20, 0.75, 0.39, 0.17,
                 0.53, 0.14, 0.38, 1, 0.24, 0.58, -0.04, 0.46, 0.73, 0.15,
                 0.38, 0.10, 0.20, 0.24, 1, 0.32, 0.11, 0.26, 0.19, 0.43,
                 0.81, 0.34, 0.49, 0.58, 0.32, 1, 0.34, 0.46, 0.55, 0.24,
                 0.35, 0.65, 0.20, -0.04, 0.11, 0.34, 1, 0.18, 0.06, 0.15,
                 0.42, 0.32, 0.75, 0.46, 0.26, 0.46, 0.18, 1, 0.54, 0.20,
                 0.40, 0.14, 0.39, 0.73, 0.19, 0.55, 0.06, 0.54, 1, 0.16,
                 0.24, 0.15, 0.17, 0.15, 0.43, 0.24, 0.15, 0.20, 0.16, 1
                 ), ncol = 10, byrow = TRUE)
cor2
```

```
##       [,1] [,2] [,3]  [,4] [,5] [,6]  [,7] [,8] [,9] [,10]
##  [1,] 1.00 0.37 0.42  0.53 0.38 0.81  0.35 0.42 0.40  0.24
##  [2,] 0.37 1.00 0.33  0.14 0.10 0.34  0.65 0.32 0.14  0.15
##  [3,] 0.42 0.33 1.00  0.38 0.20 0.49  0.20 0.75 0.39  0.17
##  [4,] 0.53 0.14 0.38  1.00 0.24 0.58 -0.04 0.46 0.73  0.15
```

```
##  [5,] 0.38 0.10 0.20  0.24 1.00 0.32  0.11 0.26 0.19  0.43
##  [6,] 0.81 0.34 0.49  0.58 0.32 1.00  0.34 0.46 0.55  0.24
##  [7,] 0.35 0.65 0.20 -0.04 0.11 0.34  1.00 0.18 0.06  0.15
##  [8,] 0.42 0.32 0.75  0.46 0.26 0.46  0.18 1.00 0.54  0.20
##  [9,] 0.40 0.14 0.39  0.73 0.19 0.55  0.06 0.54 1.00  0.16
## [10,] 0.24 0.15 0.17  0.15 0.43 0.24  0.15 0.20 0.16  1.00
```

```
cor
```

```
##                    anomia_1967 powerlessness_1967 anomia_1971
## anomia_1967                1.0                0.6         0.5
## powerlessness_1967         0.6                1.0         0.7
## anomia_1971                0.5                0.7         1.0
## powerlessness_1971         0.4                0.5         0.8
##                    powerlessness_1971
## anomia_1967                       0.4
## powerlessness_1967                0.5
## anomia_1971                       0.8
## powerlessness_1971                1.0
```

```
colnames(cor2) <- c("V1", "S1", "R1", "N1", "W1", "V2", "S2", "R2", "N2", "W2")
rownames(cor2) <- colnames(cor2)
cor_matrix_nearest_pd2 <- nearPD(cor2)$mat
cor_matrix_nearest_pd2
```

```
## 10 x 10 Matrix of class "dpoMatrix"
##      V1   S1   R1    N1   W1   V2    S2   R2   N2   W2
## V1 1.00 0.37 0.42  0.53 0.38 0.81  0.35 0.42 0.40 0.24
## S1 0.37 1.00 0.33  0.14 0.10 0.34  0.65 0.32 0.14 0.15
## R1 0.42 0.33 1.00  0.38 0.20 0.49  0.20 0.75 0.39 0.17
## N1 0.53 0.14 0.38  1.00 0.24 0.58 -0.04 0.46 0.73 0.15
## W1 0.38 0.10 0.20  0.24 1.00 0.32  0.11 0.26 0.19 0.43
## V2 0.81 0.34 0.49  0.58 0.32 1.00  0.34 0.46 0.55 0.24
## S2 0.35 0.65 0.20 -0.04 0.11 0.34  1.00 0.18 0.06 0.15
## R2 0.42 0.32 0.75  0.46 0.26 0.46  0.18 1.00 0.54 0.20
## N2 0.40 0.14 0.39  0.73 0.19 0.55  0.06 0.54 1.00 0.16
## W2 0.24 0.15 0.17  0.15 0.43 0.24  0.15 0.20 0.16 1.00
```

```
model2 <- '
  F1 =~ V1 + S1 + R1 + N1 + W1
  F2 =~ V2 + S2 + R2 + N2 + W2
  F1 ~~ F2'
sem2 <- sem(model2, sample.cov = as.matrix(cor_matrix_nearest_pd2), sample.nobs = 110, estimator = "ML")
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

```
summary(sem2, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

```
## lavaan 0.6.15 ended normally after 26 iterations
```

```
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        21
##
##   Number of observations                           110
##
## Model Test User Model:
##
##   Test statistic                               211.828
##   Degrees of freedom                                34
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                               553.443
##   Degrees of freedom                                45
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.650
##   Tucker-Lewis Index (TLI)                       0.537
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1385.002
##   Loglikelihood unrestricted model (H1)      -1279.088
##
##   Akaike (AIC)                                2812.004
##   Bayesian (BIC)                              2868.714
##   Sample-size adjusted Bayesian (SABIC)       2802.353
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.218
##   90 Percent confidence interval - lower         0.190
##   90 Percent confidence interval - upper         0.247
##   P-value H_0: RMSEA <= 0.050                    0.000
##   P-value H_0: RMSEA >= 0.080                    1.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.120
##
## 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
##   F1 =~
```

```
##     V1                1.000                                    0.756    0.760
##     S1                0.523    0.122    4.270    0.000    0.395    0.397
##     R1                0.827    0.118    6.991    0.000    0.626    0.628
##     N1                0.871    0.118    7.414    0.000    0.659    0.662
##     W1                0.466    0.123    3.793    0.000    0.353    0.354
##   F2 =~
##     V2                1.000                                    0.828    0.832
##     S2                0.390    0.113    3.451    0.001    0.323    0.325
##     R2                0.790    0.102    7.759    0.000    0.654    0.657
##     N2                0.735    0.104    7.078    0.000    0.609    0.612
##     W2                0.351    0.114    3.086    0.002    0.291    0.292
##
## Covariances:
##                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   F1 ~~
##     F2                0.706    0.115    6.146    0.000    1.127    1.127
##
## Variances:
##                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .V1                0.419    0.063    6.604    0.000    0.419    0.423
##    .S1                0.835    0.113    7.405    0.000    0.835    0.842
##    .R1                0.600    0.083    7.215    0.000    0.600    0.605
##    .N1                0.557    0.078    7.128    0.000    0.557    0.562
##    .W1                0.866    0.117    7.411    0.000    0.866    0.874
##    .V2                0.305    0.054    5.629    0.000    0.305    0.308
##    .S2                0.886    0.120    7.382    0.000    0.886    0.895
##    .R2                0.563    0.080    7.054    0.000    0.563    0.568
##    .N2                0.620    0.087    7.159    0.000    0.620    0.626
##    .W2                0.906    0.123    7.390    0.000    0.906    0.915
##     F1                0.572    0.124    4.596    0.000    1.000    1.000
##     F2                0.686    0.132    5.198    0.000    1.000    1.000
##
## R-Square:
##                   Estimate
##     V1                0.577
##     S1                0.158
##     R1                0.395
##     N1                0.438
##     W1                0.126
##     V2                0.692
##     S2                0.105
##     R2                0.432
##     N2                0.374
##     W2                0.085
```