**Tor Thogersen**

**DA 410 -- Multivariate Analysis -- Winter 2018**

**Project 7**

**Part 1:** Use R to solve Chapter 13 Page 476, #13.8(a). Make sure you include the commands and outputs, as well as the interpretations of the outputs.

**# Loading data and adding column names to table**

> BONE <- read.table("~/School/DA 410/multivariate\_analysis - 3rd Ed/multivariate\_analysis - 3rd Ed/T3\_7\_BONE.DAT", quote="\"", comment.char="")

> View(BONE)

> colnames(BONE) <- c("Ind", "8yr", "8.5yr", "9yr", "9.5yr")

> attach(BONE)

**# calculate the correlation matrix**

> cor.bone <- cor(BONE[,2:5])

> cor.bone

8yr 8.5yr 9yr 9.5yr

8yr 1.0000000 0.9686511 0.8729938 0.8068860

8.5yr 0.9686511 1.0000000 0.9212312 0.8529521

9yr 0.8729938 0.9212312 1.0000000 0.9659379

9.5yr 0.8068860 0.8529521 0.9659379 1.0000000

**# use fa() to conduct an oblique principal-axis exploratory factor analysis**

**# save the solution to an R variable**

> bone.pcl <- fa(r = cor.bone, nfactors = 2, rotate = "oblimin", fm = "pa")

Loading required namespace: GPArotation

> bone.pcl

Factor Analysis using method = pa

Call: fa(r = cor.bone, nfactors = 2, rotate = "oblimin", fm = "pa")

Standardized loadings (pattern matrix) based upon correlation matrix

PA1 PA2 h2 u2 com

8yr 0.96 -0.24 0.95 0.046 1.1

8.5yr 0.99 -0.18 0.99 0.012 1.1

9yr 0.96 0.20 0.99 0.009 1.1

9.5yr 0.91 0.30 0.95 0.046 1.2

PA1 PA2

SS loadings 3.66 0.22

Proportion Var 0.92 0.06

Cumulative Var 0.92 0.97

Proportion Explained 0.94 0.06

Cumulative Proportion 0.94 1.00

With factor correlations of

PA1 PA2

PA1 1.00 0.06

PA2 0.06 1.00

Mean item complexity = 1.1

Test of the hypothesis that 2 factors are sufficient.

The degrees of freedom for the null model are 6 and the objective function was 7.57

The degrees of freedom for the model are -1 and the objective function was 0.03

The root mean square of the residuals (RMSR) is 0

The df corrected root mean square of the residuals is NA

Fit based upon off diagonal values = 1

Measures of factor score adequacy

PA1 PA2

Correlation of (regression) scores with factors 1.00 0.96

Multiple R square of scores with factors 1.00 0.92

Minimum correlation of possible factor scores 0.99 0.84

## **# Rotation of Factor Loadings**

> bone.varimax <- fa(r = cor.bone, nfactors = 2, rotate = "varimax", fm = "pa")

> bone.varimax

Factor Analysis using method = pa

Call: fa(r = cor.bone, nfactors = 2, rotate = "varimax", fm = "pa")

Standardized loadings (pattern matrix) based upon correlation matrix

PA1 PA2 h2 u2 com

8yr 0.86 0.47 0.95 0.046 1.5

8.5yr 0.84 0.53 0.99 0.012 1.7

9yr 0.58 0.81 0.99 0.009 1.8

9.5yr 0.48 0.85 0.95 0.046 1.6

PA1 PA2

SS loadings 2.00 1.88

Proportion Var 0.50 0.47

Cumulative Var 0.50 0.97

Proportion Explained 0.52 0.48

Cumulative Proportion 0.52 1.00

Mean item complexity = 1.7

Test of the hypothesis that 2 factors are sufficient.

The degrees of freedom for the null model are 6 and the objective function was 7.57

The degrees of freedom for the model are -1 and the objective function was 0.03

The root mean square of the residuals (RMSR) is 0

The df corrected root mean square of the residuals is NA

Fit based upon off diagonal values = 1

Measures of factor score adequacy

PA1 PA2

Correlation of (regression) scores with factors 0.98 0.98

Multiple R square of scores with factors 0.96 0.96

Minimum correlation of possible factor scores 0.91 0.91

**Manual process for finding Varimax**

> R <- cor(BONE[,2:5])

> round(R, 2)

8yr 8.5yr 9yr 9.5yr

8yr 1.00 0.97 0.87 0.81

8.5yr 0.97 1.00 0.92 0.85

9yr 0.87 0.92 1.00 0.97

9.5yr 0.81 0.85 0.97 1.00

**> # Calculate and replace the diagonal of R with the estimated communalities.**

> R.smc <- (1 - 1 / diag(solve(R)))

> diag(R) <- R.smc

> round(R, 2)

8yr 8.5yr 9yr 9.5yr

8yr 0.94 0.97 0.87 0.81

8.5yr 0.97 0.96 0.92 0.85

9yr 0.87 0.92 0.97 0.97

9.5yr 0.81 0.85 0.97 0.94

**> # find the eigenvalues and eigenvectors of the R−Ψ^ matrix.**

> r.eigen <- eigen(R)

> r.eigen$values

[1] 3.64977030 0.20395842 -0.01384332 -0.02353020

> tot.prop <- 0

> for (i in r.eigen$values) {

+ tot.prop <- tot.prop + i / sum(r.eigen$values)

+ print(tot.prop)

+ }

[1] 0.9563497

[1] 1.009793

[1] 1.006166

[1] 1

> #

**> # Obtain the factor loadings as before by multiplying the square root of the first two eigenvalues by their respective eigenvectors.**

> r.lambda <- as.matrix(r.eigen$vectors[,1:2]) %\*% diag(sqrt(r.eigen$values[1:2]))

> r.lambda

[,1] [,2]

[1,] -0.9397811 0.2560011

[2,] -0.9705014 0.1852559

[3,] -0.9759056 -0.1659327

[4,] -0.9339792 -0.2767101

**> # The communalities, specific variances and complexity of the factor loadings can then be calculated.**

> r.h2 <- rowSums(r.lambda^2)

> r.u2 <- 1 - r.h2

> com <- rowSums(r.lambda^2)^2 / rowSums(r.lambda^4)

**> # Collect the results into a data.frame.**

> cor.pa <- data.frame(cbind(round(r.lambda, 2), round(r.h2, 2), round(r.u2, 3), round(com, 1)))

> colnames(cor.pa) <- c('PA1', 'PA2', 'h2', 'u2', 'com')

> cor.pa

PA1 PA2 h2 u2 com

1 -0.94 0.26 0.95 0.051 1.1

2 -0.97 0.19 0.98 0.024 1.1

3 -0.98 -0.17 0.98 0.020 1.1

4 -0.93 -0.28 0.95 0.051 1.2