STAT 3690 Lecture 25

zhiyanggeezhou.github.io

Zhiyang Zhou (zhiyang.zhou@umanitoba.ca)

Mar 30, 2022

Estimating $\mathbf{L}$ and $\mathbf{\Psi}$ (con'd)

- ML method
  - Further assumptions
    * $\mathbf{F} \sim MVN_q(\mathbf{0}, \mathbf{I})$
    * $\mathbf{E} \sim MVN_p(\mathbf{0}, \mathbf{\Psi})$
    * $\mathbf{L}^\top \mathbf{\Psi}^{-1} \mathbf{L}$ is diagonal
  - `factanal` or `psych::fa`

```
install.packages(c('psych'))
library(psych)
library(tidyverse)
options(digits = 4)
head(psych::bfi)
data = bfi %>%
  select(-gender, -education, -age) %>%
  filter(complete.cases(.)) # Remove demographic variable and keep complete data
S = cov(data)

# the number of eigenvalues greater than one for the correlation matrix
(q = sum(eigen(cor(data))$values > 1))

# apply functions factanal OR psych::fa
decomp <- factanal(covmat = S, factors = q, rotation = 'none')
decomp <- psych::fa(r = S, nfactors = q, rotate = "none", fm = "ml")

L_ml <- decomp$loadings
Psi_ml <- diag(decomp$uniquenesses)
S_ml = tcrossprod(L_ml) + Psi_ml
lattice::levelplot(S - S_ml)
lattice::levelplot((S - S_ml)/S)
lattice::levelplot((S_pc - S_ml)/S)
```

- Comments on estimation of $\mathbf{L}$ and $\mathbf{\Psi}$
  - Other methods
  - Different statistical softwares may apply different methods
    * Have to look into help manuals to figure out what is going on for different softwares/packages
  - Compare the outputs of multiple estimation methods
    * For a good fit, similar answers would be reached regardless of the method

---

Factor rotation

- $\mathbf{L}$ is not uniquely defined: if $\mathbf{Y} - \boldsymbol{\mu} = \mathbf{L}\mathbf{F} + \mathbf{E}$, then $\mathbf{Y} - \boldsymbol{\mu} = \widetilde{\mathbf{L}}\widetilde{\mathbf{F}} + \mathbf{E}$, where
  - $\widetilde{\mathbf{L}} = \mathbf{L}\mathbf{P}$ and $\widetilde{\mathbf{F}} = \mathbf{P}^\top\mathbf{F}$ with $\mathbf{P}$ a $q \times q$ orthogonal matrix, i.e., $\mathbf{P}\mathbf{P}^\top = \mathbf{I}$
- A blessing to improve interpretation: pick up a $\mathbf{P}$ such that $\widetilde{\mathbf{F}}$ is more interpretable; to ease interpretation, we want:
  - Each entry of $\mathbf{Y}$ to have large loadings for merely one common factor and negligible loadings for the others
- varimax: find rotation $\mathbf{P}$ to maximize the sum of variance of squared (scaled) loadings over all the common factors

$$\sum_{j=1}^{q} \left\{ \frac{1}{p} \sum_{i=1}^{p} \tilde{\ell}_{ij}^{*4} - \left(\frac{1}{p} \sum_{i=1}^{p} \tilde{\ell}_{ij}^{*2} \right)^2 \right\}$$

  - $\tilde{\ell}_{ij}^{*} = \tilde{\ell}_{ij} / \sqrt{\sum_{j=1}^{q} \tilde{\ell}_{ij}^{2}}$ with $\tilde{\ell}_{ij}$ the (i,j)-th entry of $\widetilde{\mathbf{L}} = \mathbf{L}\mathbf{P}$

```r
decomp <- factanal(covmat = S, factors = q, rotation = "none")
L_ml <- decomp$loadings
L_ml_varimax1 = varimax(L_ml)$loadings
L_ml_varimax2 = factanal(covmat = S, factors = q, rotation = "varimax")$loadings
head(L_ml)
head(L_ml_varimax1)
head(L_ml_varimax2)

# Plot loading matrix
lattice::levelplot(unclass(t(L_ml)), xlab = "", ylab = "")
lattice::levelplot(unclass(t(L_ml_varimax1)), xlab = "", ylab = "")
lattice::levelplot(unclass(t(L_ml_varimax2)), xlab = "", ylab = "")

# The rotation matrix
varimax(L_ml)$rotmat
factanal(covmat = S, factors = q, rotation = "varimax")$rotmat
```

- Comments on factor rotation
  - Especially useful with loadings obtained through ML
  - Sometimes used for PCA loadings