Intro

This is material to prepare for application in R in Psychometrics.

Resources: website

Material made by: *Susu Zhang*

(*Additional)

CTT Item Analysis

1.1) Response data

```
resp <- read.table('resp.txt', header = F, sep = '\t')
```

The imported data set resp contains 100 subjects' dichotomous responses to 40 GRE questions.

Output 1.1

1.2) CTT Item Analysis

The total score of each subject can be obtained by the row sums.

```
total.score <- rowSums(resp)
```

Output 1.2

The item difficulty in CTT can be obtained by calculating the proportion of correct answers of each item.

```
item.diff <- colMeans(resp)
```

Output 1.3

1.3) Item discrimination

The item discrimination in CTT can be obtained by the point biserial correlation between the item response and the total score.

```
n.items <- ncol(resp) # number of items
total.score <- rowSums(resp) # total score
item.disc <- numeric(n.items) # output vector
for(i in 1:n.items){ # sequence
```

CTT Item Analysis (cont)

```
item.disc[i] <- cor(total.score, resp[,i])} # body
```

Output 1.4

(*Additional)

Output 1.1

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 1  0  0  1  0  1  0  1  0  1   1   1   0   1   1   0   1   1   1   0   1   1
## 2  0  0  0  1  1  1  1  1  1   1   1   1   1   1   1   1   1   1   0   1   1
## 3  1  1  1  0  0  0  0  1  0   0   1   0   1   1   1   0   0   1   1   1   1
## 4  1  0  1  0  1  1  1  1  0   0   1   1   0   1   1   0   0   1   1   1   1
## 5  1  1  1  1  1  1  0  0  1   1   1   0   1   1   1   0   1   0   0   1   1
## 6  0  1  1  1  0  1  0  0  1   1   1   1   1   1   1   1   0   0   1   0   0
##   V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34 V35 V36 V37 V38 V39 V40
## 1   1   1   0   0   0   1   1   1   1   1   1   1   1   1   0   1   1   0   0
## 2   1   0   1   1   0   1   1   1   1   1   1   1   1   1   0   1   0   1   0
## 3   1   0   0   0   0   0   0   0   1   0   0   1   0   1   0   0   0   0   0
## 4   0   0   1   0   0   0   0   1   1   0   0   0   1   0   1   1   0   1   0
## 5   1   1   0   1   0   0   1   1   1   1   0   1   0   0   1   1   1   1   1
## 6   1   0   1   1   0   1   1   0   1   1   1   1   1   1   0   1   0   0   1
```

Output 1.2

```
##   [1] 26 31 16 21 28 26 18 20 18 32 14 13 24 35 25 16 11 14 15 15 35 17 33 17 22
##  [26] 26 22 16 22 28 20 26 18 18 29 21 20 32 21 27 18 26 26 26 19 16 20 27 17 31
##  [51] 19 29 16 12 20 25 13 18 20 22 25 20 21 21 28 17 10 13 14 25 23 18 21 29 19
##  [76] 16 22 21 17 24 11 15 18 14 22 22 16 24 28 30 22 18 29 27 23 31 25 25 22 31
```

Output 1.3

```
##   V1   V2   V3   V4   V5   V6   V7   V8   V9  V10  V11  V12  V13  V14  V15  V16
## 0.66 0.47 0.77 0.47 0.75 0.35 0.48 0.40 0.51 0.66 0.77 0.66 0.60 0.77 0.79 0.60
##  V17  V18  V19  V20  V21  V22  V23  V24  V25  V26  V27  V28  V29  V30  V31  V32
## 0.70 0.50 0.61 0.46 0.52 0.60 0.30 0.45 0.56 0.21 0.49 0.62 0.54 0.51 0.48 0.47
##  V33  V34  V35  V36  V37  V38  V39  V40
## 0.61 0.37 0.54 0.43 0.67 0.53 0.36 0.41
```

Output 1.4

```
##  [1] 0.1098685 0.1124206 0.2052317 0.3376083 0.1687113 0.5483058 0.2658013
##  [8] 0.1376503 0.3488277 0.2011214 0.2956240 0.2850741 0.3176545 0.2750803
## [15] 0.3593535 0.3035366 0.2848757 0.2057620 0.2706637 0.3299308 0.2844905
## [22] 0.3494200 0.3603395 0.4266307 0.3782924 0.2222346 0.1907570 0.5225895
## [29] 0.3257677 0.4041697 0.4942243 0.3168218 0.2316683 0.2970732 0.3986230
## [36] 0.2132221 0.3436395 0.2721307 0.4056162 0.2438073
```

Validity

A test has validity if it measures what it purports to measure. There are three types of validities; content validity, criterion-related validity, and construct validity.

```
test1 <- read.table("test1.txt")
test2 <- read.table("test2.txt")
```

2.1) Criterion Related Validity

```
total_X <- rowSums(test1)
```

Validity (cont)

```
total_Y <- rowSums(test2)
rho_xy <- cor(total_X, total_Y)
```

The validity coefficient is 0.6024.

2.2) Correction for attenuation

```
coeff.alpha <- function(responses){
# Get number of items (N) and individuals
n.items <- ncol(responses)
n.persons <- nrow(responses)
# Get individual total scores
x <- rowSums(responses)
# Get observed-score variance of whole test (X)
var.x <- var(x)*(n.persons-1)/n.persons
# Get observed-score variance of each item (Y_j)
var.y <- numeric(n.items)
for(i in 1:n.items){
var.y[i] <- var(responses[,i])*(n.persons-1)/n.persons
}
# Apply the alpha formula
alpha <- (n.items/(n.items-1))*(1 - sum(var.y)/var.x)
return(alpha)
}
rho_xx <- coeff.alpha(test1)
rho_xx
## [1] 0.6061799
rho_yy <- coeff.alpha(test2)
rho_yy
## [1] 0.6439105
rho_txty <- rho_xy / (sqrt(rho_xx) * sqrt(rho_yy))
rho_txty
```

Validity (cont)

```
## [1] 0.9641909
```

Obtain the true score correlation.

(*Additional)

Decision Table

Decision table or a contingency table based on a test score and a criterion score. From the contingency table, students will be able to understand and obtain the hit rate, sensitivity, specificity, and base rate in R. Test scores are often used for making screening decisions.

3.1) Decision Table

```
test1 <- read.table("test1.txt")
test2 <- read.table("test2.txt")
# total scores
X <- rowSums(test1)
Y <- rowSums(test2)
predicted <- (X >= 13)
actual <- (Y >= 13)
head(predicted)
## [1] FALSE TRUE FALSE FALSE
FALSE FALSE
head(actual)
## [1] FALSE FALSE TRUE FALSE
FALSE FALSE
sum(predicted)
## [1] 44
sum(actual)
## [1] 33
mean(predicted)
## [1] 0.44
mean(actual)
```

Decision Table (cont)

```
## [1] 0.33
match <- cbind(predicted,
actual)
decision <- table(actual,
predicted)
decision
```

Output 3.1

3.2) Hit Rate

```
mean(predicted == actual)
## [1] 0.71
decision[1,1] # Correct
rejection
## [1] 47
decision[2,2] # Hit
## [1] 24
sum(decision) # Number of
examinees
## [1] 100
(decision[1,1] + decision[2,2])
/ sum(decision) # hit rate
## [1] 0.71
```

3.3) Sensitivity and Specificity

```
decision[2,2] # hit
## [1] 24
decision[2,1] # miss
## [1] 9
decision[2,2] / (decision[2,2] +
decision[2,1]) # sensitivity
## [1] 0.7272727
decision[1,1] # correct
rejection
## [1] 47
decision[1,2] # false alarm
## [1] 20
```

Decision Table (cont)

```
decision[1,1] / (decision[1,1] +
decision[1,2]) # specificity
## [1] 0.7014925
mean(predicted[actual == TRUE]
== TRUE) # sensitivity
## [1] 0.7272727
mean(predicted[actual == FALSE]
== FALSE) # specificity
## [1] 0.7014925
```

3.4) Base Rate

```
mean(actual)
## [1] 0.33
(decision[2,1] + decision[2,2])
/ sum(decision)
## [1] 0.33
sum(decision[2,]) /
sum(decision)
## [1] 0.33
rowSums(decision)[2] /
sum(decision)
## TRUE
## 0.33
```

(*Additional)

Output 3.1

```
##        predicted
## actual  FALSE TRUE
##   FALSE    47   20
##   TRUE      9   24
```

Test Construction

Suppose that we developed a new test containing N newly written items. The test is designed to measure a unidimensional latent construct. Also, the test items are assumed to be essentially tau– equivalent. This N-item test is administered to a pretest sample.

4.1) Item Difficulty

```
responses <- read.table(".")
Y <- read.table(".")
difficulty <- colMeans(responses)
difficulty
```

Output 4.1

4.2) Item Discrimination

```
X <- rowSums(responses) # total score
discrimination <- numeric(ncol(responses)) # outcome vector
for(i in 1:ncol(responses)){
discrimination[i] <- cor(responses[,i], X) # Pearson correlation between the i-th item score and the total score
}
discrimination
```

Output 4.2

4.3) Item-score SD

```
item_sd <- sqrt(difficulty * (1-difficulty))
item_sd
```

Output 4.3

4.4) Item Reliability

```
item_rel <- item_sd * discrimination
round(item_rel, 2) # round to 2 digits
```

Output 4.4

4.5) Item Validity

Test Construction (cont)

```
r_iy <- numeric(ncol(responses))
for(i in 1:ncol(responses)){
r_iy[i] <- cor(responses[,i], Y)
# replace each value by r_iy
}
item_val <- item_sd * r_iy
round(item_val, 2)
```

Output 4.5

(*Additional)

Output 4.1

```
##    V1    V2    V3    V4    V5    V6    V7    V8    V9   V10   V11   V12   V13
## 0.630 0.384 0.705 0.577 0.685 0.425 0.439 0.359 0.328 0.560 0.676 0.602 0.516
##   V14   V15   V16   V17   V18   V19   V20   V21   V22   V23   V24   V25   V26
## 0.489 0.778 0.606 0.531 0.324 0.493 0.325 0.520 0.627 0.243 0.388 0.510 0.247
##   V27   V28   V29   V30   V31   V32   V33   V34   V35   V36   V37   V38   V39
## 0.389 0.621 0.428 0.340 0.339 0.400 0.443 0.279 0.457 0.199 0.445 0.281 0.171
##   V40
## 0.194
```

Output 4.2

```
##  [1] 0.2003023 0.2633101 0.2157009 0.2513577 0.3098860 0.2639948 0.2631635
##  [8] 0.3384545 0.3131245 0.3318597 0.2618446 0.3135515 0.3345164 0.3089618
## [15] 0.3208180 0.3625530 0.3684249 0.3470304 0.3606028 0.3569275 0.3955714
## [22] 0.3782499 0.3770550 0.3779781 0.4531812 0.3824007 0.4378612 0.4120439
## [29] 0.4788896 0.4611929 0.3954019 0.4476655 0.4963732 0.4260595 0.4204738
## [36] 0.3638530 0.4955949 0.4764930 0.4339662 0.4974743
```

Output 4.3

```
##        V1        V2        V3        V4        V5        V6        V7        V8
## 0.4828043 0.4863579 0.4560428 0.4940354 0.4645159 0.4943430 0.4962651 0.4797072
##        V9       V10       V11       V12       V13       V14       V15       V16
## 0.4694848 0.4963869 0.4680000 0.4894854 0.4997439 0.4998790 0.4155911 0.4886348
##       V17       V18       V19       V20       V21       V22       V23       V24
## 0.4990381 0.4680000 0.4999510 0.4683748 0.4995998 0.4836021 0.4288951 0.4872946
##       V25       V26       V27       V28       V29       V30       V31       V32
## 0.4999000 0.4312667 0.4875233 0.4851381 0.4947888 0.4737088 0.4733698 0.4898979
##       V33       V34       V35       V36       V37       V38       V39       V40
## 0.4967404 0.4485075 0.4981476 0.3992480 0.4969658 0.4494875 0.3765090 0.3954289
```

Output 4.4

```
##   V1   V2   V3   V4   V5   V6   V7   V8   V9  V10  V11  V12  V13  V14  V15  V16
## 0.10 0.13 0.10 0.12 0.14 0.13 0.13 0.16 0.15 0.16 0.12 0.15 0.17 0.15 0.13 0.18
##  V17  V18  V19  V20  V21  V22  V23  V24  V25  V26  V27  V28  V29  V30  V31  V32
## 0.18 0.16 0.18 0.17 0.20 0.18 0.16 0.18 0.23 0.16 0.21 0.20 0.24 0.22 0.19 0.22
##  V33  V34  V35  V36  V37  V38  V39  V40
## 0.25 0.19 0.21 0.15 0.25 0.21 0.16 0.20
```

Output 4.5

```
##    V1    V2    V3    V4    V5    V6    V7    V8    V9   V10   V11   V12   V13
##  0.00  0.02  0.07  0.03  0.05  0.07  0.01  0.09  0.04  0.00  0.05  0.04  0.11
##   V14   V15   V16   V17   V18   V19   V20   V21   V22   V23   V24   V25   V26
## -0.03  0.06  0.04  0.04  0.02  0.03  0.05  0.02  0.11  0.04  0.11  0.10  0.08
##   V27   V28   V29   V30   V31   V32   V33   V34   V35   V36   V37   V38   V39
##  0.03  0.06  0.08  0.07  0.02  0.01  0.06  0.10  0.03  0.03  0.10  0.09  0.06
##   V40
##  0.06
```

Exploratory Factor Analysis

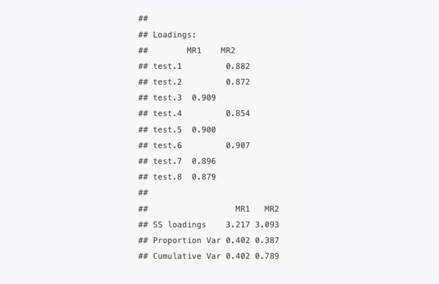
Factor analysis (FA) assumes that there are a number of underlying (latent) factors affecting the observed scores on items/-tests. In other words, the traits underlying a test might be multidimensional.

```
scores <- read.table(".")
cor.subtests <- cor(scores)
factanal(x = scores, factors = 1)
factanal(x = scores, factors = 2)
cov.mat <- cov(scores)
factanal(covmat=cov.mat, factors = 2, n.obs = nrow(scores))
output <- factanal(x = scores, factors = 2, scores = 'regression')
varimax <- factanal(scores, factors = 2, rotation="varimax", scores="regression")
promax <- factanal(scores, factors = 2, rotation="promax", scores="regression")
library(psych)
output2 <- fa(scores, # input data
nfactors = 2, # number of factors
rotate = "varimax", # rotation
scores = "regression") # factor score estimation
output2$loadings # factor loadings
```

Output 5

(*Additional)

Output 5



```
##
## Loadings:
##        MR1    MR2
## test.1         0.882
## test.2         0.872
## test.3  0.909
## test.4         0.854
## test.5  0.900
## test.6         0.907
## test.7  0.896
## test.8  0.879
##
##                  MR1   MR2
## SS loadings    3.217 3.093
## Proportion Var 0.402 0.387
## Cumulative Var 0.402 0.789
```

Item Response Theory

Item response theory models the nonlinear relationship between the examinee's trait and the probability of response.

```
GRE40 <- read.table(".")
par(mfrow=c(1,3))
hist(GRE40$a)
hist(GRE40$b)
hist(GRE40$c)
```

Output 6.1

Output 6.2

```
irt.p <- function(theta,a,b,c){
#ability and item parameters as scalars
p <- c + (1-c)/(1 + exp(-1.7a(theta-b)))
return(p)
}
irt.p(theta = 0, a = 1, b = -.2, c = .25)
## [1] 0.6881429
irt.p(theta = -1, a = 1, b = -.2, c = .25)
## [1] 0.4031802
theta_seq <- seq(from = -4, to = 4, by = .1)
p_item1 <-
numeric(length(theta_seq))
for(t in 1:length(theta_seq)){
p_item1[t] <- irt.p(theta =
theta_seq[t],
```

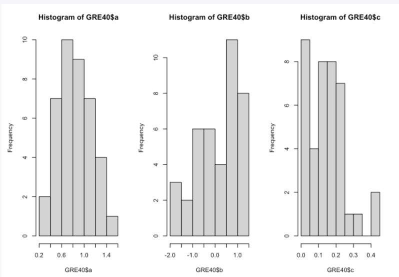
Item Response Theory (cont)

```
a = GRE40$a[1],
b = GRE40$b[1],
c = GRE40$c[1])
}
plot(theta_seq, p_item1, type =
'l', ylim=c(0,1))
```

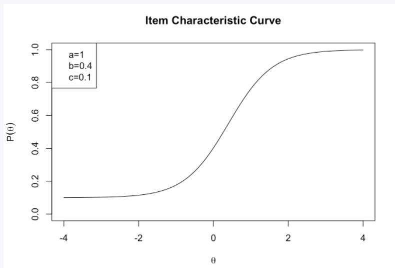
Output 6.3

(*Additional)

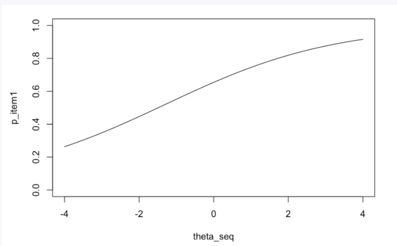
Output 6.1



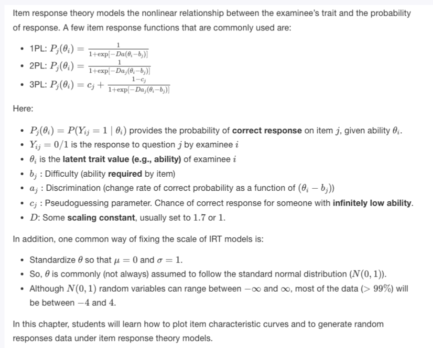
Output 6.2



Output 6.3



Item Response Theory



Item response theory models the nonlinear relationship between the examinee's trait and the probability of response. A few item response functions that are commonly used are:

- 1PL: $P_j(\theta_i) = \frac{1}{1+\exp[-Da(\theta_i - b_j)]}$
- 2PL: $P_j(\theta_i) = \frac{1}{1+\exp[-Da_j(\theta_i - b_j)]}$
- 3PL: $P_j(\theta_i) = c_j + \frac{1-c_j}{1+\exp[-Da_j(\theta_i - b_j)]}$

Here:

- $P_j(\theta_i) = P(Y_{ij} = 1 \mid \theta_i)$ provides the probability of **correct response** on item j, given ability θ_i.
- $Y_{ij} = 0/1$ is the response to question j by examinee i
- θ_i is the **latent trait value (e.g., ability)** of examinee i
- b_j : Difficulty (ability **required** by item)
- a_j : Discrimination (change rate of correct probability as a function of $(\theta_i - b_j)$)
- c_j : Pseudoguessing parameter. Chance of correct response for someone with **infinitely low ability**.
- D: Some **scaling constant**, usually set to 1.7 or 1.

In addition, one common way of fixing the scale of IRT models is:

- Standardize θ so that $\mu = 0$ and $\sigma = 1$.
- So, θ is commonly (not always) assumed to follow the standard normal distribution ($N(0,1)$).
- Although $N(0,1)$ random variables can range between $-\infty$ and ∞, most of the data ($> 99\%$) will be between -4 and 4.

In this chapter, students will learn how to plot item characteristic curves and to generate random responses data under item response theory models.