IRT Assignment 3

Ibrahim Osumanu

2024-10-08

library("psych")  
library("mirt")

## Loading required package: stats4

## Loading required package: lattice

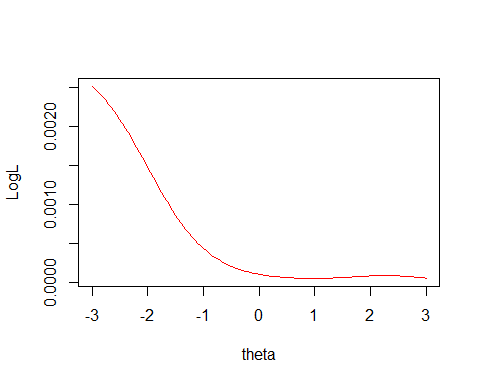
load("C:/Users/gtj24002/OneDrive - University of Connecticut/Desktop/IRT/Week1workspace.RData")

#Question 1a

params<-matrix(c(1.8,1.8,1.8,1.8,1.8,1.8,1.8,1.8,-2,2,2,2,2,2,2,2,.25,.25,.25,.25,.25,.25,.25,.25),nrow=8)  
params

## [,1] [,2] [,3]  
## [1,] 1.8 -2 0.25  
## [2,] 1.8 2 0.25  
## [3,] 1.8 2 0.25  
## [4,] 1.8 2 0.25  
## [5,] 1.8 2 0.25  
## [6,] 1.8 2 0.25  
## [7,] 1.8 2 0.25  
## [8,] 1.8 2 0.25

response<-c(0,1,1,1,1)  
  
  
  
# activate getloglike  
  
getloglike <- function(nitems,params,response) {  
   
 pq<-array(0,c(121,nitems))  
 theta<-seq(-3,3,.05)  
   
 for (j in 1:nitems) {  
   
   
 temp<- params[j,1]\*(theta - params[j,2])  
   
 p <- params[j,3] + (1-params[j,3])/(1 + exp(-temp))  
   
 pq[,j]<- response[j]\*p + (1-response[j])\*(1-p)  
   
 }  
   
 loglike<-apply(log(pq),1,sum)  
   
 loglike  
}  
  
  
  
loglike<-getloglike(5,params,response)  
theta<-seq(-3,3,.05)  
plot(theta,exp(loglike), col="red", type="l", ylab="LogL",xlab="theta")



which.max(loglike)

## [1] 1

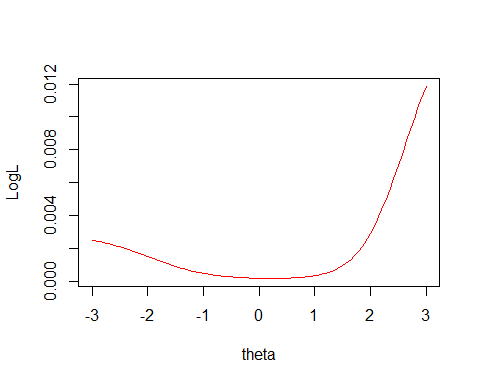
The plot of the likelihood shows that the peak of the curve is at a negative (lower) theta value (-3). The person is scored low because the first item (with a response of 0) has a high discrimination parameter (1.8) and a low difficulty parameter (-2). This means that the item is very effective at differentiating between low and high ability levels, and a response of 0 on this item strongly indicates a low ability level (theta). The other items, despite being answered correctly, do not compensate enough for the initial low response due to their similar discrimination and difficulty parameters.

#Question 1b

params<-matrix(c(1.8,1.8,1.8,1.8,1.8,1.8,1.8,1.8,-2,2,2,2,2,2,2,2,.25,.25,.25,.25,.25,.25,.25,.25),nrow=8)  
params

## [,1] [,2] [,3]  
## [1,] 1.8 -2 0.25  
## [2,] 1.8 2 0.25  
## [3,] 1.8 2 0.25  
## [4,] 1.8 2 0.25  
## [5,] 1.8 2 0.25  
## [6,] 1.8 2 0.25  
## [7,] 1.8 2 0.25  
## [8,] 1.8 2 0.25

response<-c(0,1,1,1,1)  
  
  
  
# activate getloglike  
  
getloglikeu <- function(nitems,params,response) {  
   
 pq<-array(0,c(121,nitems))  
 theta<-seq(-3,3,.05)  
   
 for (j in 1:nitems) {  
   
   
 temp<- params[j,1]\*(theta - params[j,2])  
   
 p <- params[j,3] + (0.98-params[j,3])/(1 + exp(-temp))  
   
 pq[,j]<- response[j]\*p + (1-response[j])\*(1-p)  
   
 }  
   
 loglike<-apply(log(pq),1,sum)  
   
 loglike  
}  
  
  
  
loglikeu<-getloglikeu(5,params,response)  
plot(theta,exp(loglikeu), col="red", type="l", ylab="LogL",xlab="theta")



which.max(loglikeu)

## [1] 121

When comparing the scores with an asymptote of 1 (the original IRT model) versus an asymptote of 0.98 (the modified IRT model), the score with the asymptote of 0.98 will typically be higher for a person with high ability. This is because the modified model assumes that even individuals with very high ability have a slightly less than perfect probability (0.98 instead of 1) of answering items correctly. The curve peaks at a higher theta value (3).This adjustment can lead to a higher estimated ability (theta) for individuals who answer most items correctly, as the model accounts for the possibility of occasional errors even among high-ability individuals.

#Question 2

dim(physics)

## [1] 739 16

#part a

In commenting on the relative overall fit, the 3PL model has the lowest AIC and BIC values, suggesting it fits the data best. However, it is also the most complex model.The 2PL model has lower AIC and BIC values than the Rasch model, indicating a better fit while being less complex than the 3PL model.

#Part b

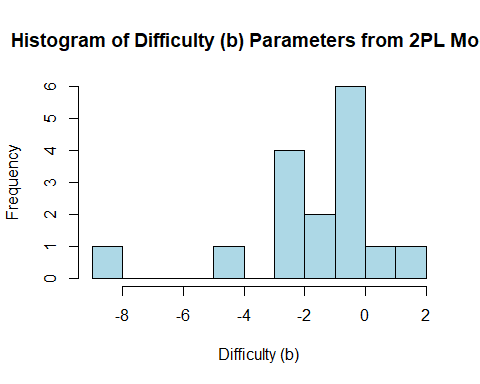
# get the item parameters (coefficients)  
  
params<-coef(m2,IRTpars=T,simplify=T)  
params

## $items  
## a b g u  
## V1 0.617 -4.086 0 1  
## V2 0.541 -2.664 0 1  
## V3 3.734 -0.648 0 1  
## V4 5.357 -0.381 0 1  
## V5 2.249 -0.365 0 1  
## V6 1.335 -0.462 0 1  
## V7 1.086 -0.525 0 1  
## V8 0.703 -8.740 0 1  
## V9 0.621 0.189 0 1  
## V10 0.682 -0.425 0 1  
## V11 -0.188 -2.938 0 1  
## V12 0.429 1.117 0 1  
## V13 0.876 -2.789 0 1  
## V14 1.087 -1.252 0 1  
## V15 0.566 -1.051 0 1  
## V16 0.428 -2.844 0 1  
##   
## $means  
## F1   
## 0   
##   
## $cov  
## F1  
## F1 1

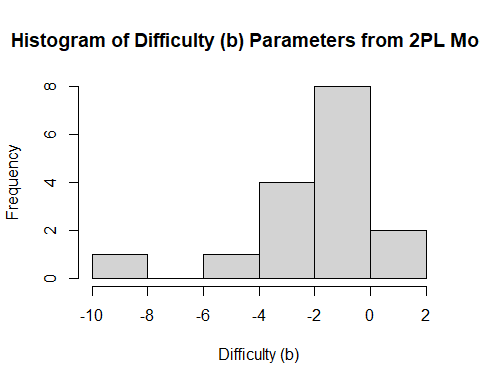
# Extract the b (difficulty) parameters from the items component  
difficulty\_params <- params$items[, "b"]  
  
# Print the difficulty parameters to verify  
difficulty\_params

## V1 V2 V3 V4 V5 V6 V7   
## -4.0857131 -2.6643032 -0.6482034 -0.3807255 -0.3649362 -0.4623406 -0.5250306   
## V8 V9 V10 V11 V12 V13 V14   
## -8.7399005 0.1889200 -0.4254160 -2.9383329 1.1168153 -2.7892080 -1.2520897   
## V15 V16   
## -1.0512734 -2.8443312

# Create a histogram of the difficulty (b) parameters  
hist(difficulty\_params,   
 main = "Histogram of Difficulty (b) Parameters from 2PL Model",  
 xlab = "Difficulty (b)",   
 col = "lightblue",   
 breaks = 10)

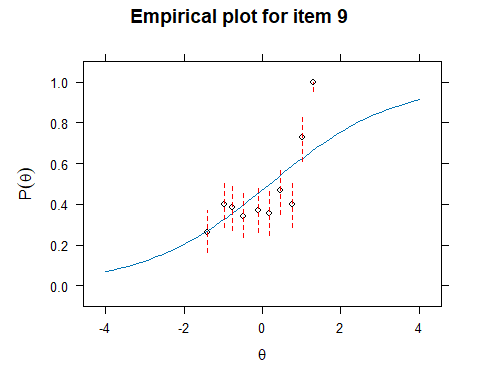


hist(difficulty\_params, main="Histogram of Difficulty (b) Parameters from 2PL Model",xlab="Difficulty (b)")

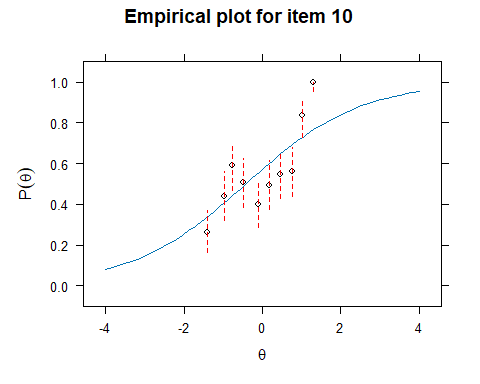
 The test seems to be relatively easy as most of the difficulty parameters are in the negatives. There are few moderately difficult parameters but they are in the minority.The peak is around -2 and 0 showing that the test is moderatiely easy.

#part c

itemfit(m2,empirical.plot = 9)



itemfit(m2,empirical.plot = 10)

 For item 9, the observed scores fit reasonably well with the predicted scores. Although there are certain deviations at lower and higher ability levels.The predicted scores increase with increasing ability levels.

Item 9s curve is gentle meaning it is not highly discriminative. Item 10 has a relatively steep slope which makes it relatively highly discriminative. its does well in differentiating between high ability and low ability test takers. the observed scores in item 10 reasonably fit well with the predicted scores especially at the higher ability levels although there are some deviations as well. Empirically, item 10 fits well better than item 9 with higher discrimination, although both items are moderately difficult.

The possible reasons why this pattern might happen is because of;

1. Measurement error or noise in the observed data.
2. Model misfit.
3. Unaccounted factors influencing the responses.

#Part d

fscores(m2, full.scores.SE = T, method = "ML")[1:10,]

## Warning: The following factor score estimates failed to converge successfully:  
## 6,25,45,74,121,131,133,145,189,211,298,307,311,315,336,370,413,472,499,506,585,605,623,636,690

## F1 SE\_F1  
## [1,] -0.75398559 0.3338127  
## [2,] 0.69992664 0.8020600  
## [3,] -0.60651191 0.2979091  
## [4,] -1.25138923 0.5358398  
## [5,] -0.01285903 0.3885208  
## [6,] 19.99997963 NA  
## [7,] 0.02047327 0.4057991  
## [8,] 3.31536501 2.1924118  
## [9,] -1.25125069 0.5357736  
## [10,] -0.89299153 0.3797763

fscores(m2, full.scores.SE = T, method = "MAP")[1:10,]

## F1 SE\_F1  
## [1,] -0.68244670 0.3000197  
## [2,] 0.45589822 0.5540103  
## [3,] -0.55835509 0.2783636  
## [4,] -1.01818242 0.3945018  
## [5,] -0.01116978 0.3628392  
## [6,] 1.29002093 0.7399428  
## [7,] 0.01758852 0.3747999  
## [8,] 1.06658641 0.7033822  
## [9,] -1.01810732 0.3944771  
## [10,] -0.78985002 0.3258525

fscores(m2, full.scores.SE = T, method = "EAP")[1:10,]

## F1 SE\_F1  
## [1,] -0.7401989 0.3300368  
## [2,] 0.6248766 0.5489472  
## [3,] -0.5883291 0.3098208  
## [4,] -1.1329396 0.4174926  
## [5,] 0.1146377 0.4038395  
## [6,] 1.4104538 0.7165209  
## [7,] 0.1500021 0.4139651  
## [8,] 1.2014661 0.6795477  
## [9,] -1.1328555 0.4174717  
## [10,] -0.8689268 0.3550336

The estimates are different because MAP and EAP methods use prior distributions, incorporating them can lead to more stable and potentially more accurate estimates, especially when data is sparse or noisy.

Again, Maximum Likelihood relies purely on the observed data, making it more sensitive to sample size and data quality. MAP and EAP, by incorporating prior information, can resolve some of these issues.Because the EAP uses the mean of the posterior distribution, it creates relatively a more robust estimate. Example in row 6, ML has an undefined SE. EAP incorporates prior information by pulling the extreme values towards the mean.

These differences in the estimates show the trade-offs between a purely driven data (ML) and estimates with prior distribution (MAP and EAP)

#Part e

The G2 value represents the likelihood ratio-chi-square statistic. The value is used to assess the goodness of fit of the model. A lower G2 value indicates a better fit of the model.

The degrees of freedom are generally calculated as the difference between the number of data points and the number of parameters estimated. For a large dataset, this can result in a very high number of degrees of freedom. Given that G2 value is 3267.6 and the degrees of freedom are 65503, it suggests that the model is being applied to a very large dataset with many items and responses.

#Part f

residuals(m2, "Q3")

## Q3 summary statistics:  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -0.321 -0.082 -0.005 -0.020 0.045 0.403   
##   
## V1 V2 V3 V4 V5 V6 V7 V8 V9 V10  
## V1 1.000 0.173 0.015 -0.087 -0.105 -0.044 0.029 0.167 -0.120 0.007  
## V2 0.173 1.000 -0.086 0.015 -0.120 -0.119 0.001 0.031 -0.042 0.060  
## V3 0.015 -0.086 1.000 -0.257 -0.248 -0.246 -0.192 0.098 -0.164 -0.125  
## V4 -0.087 0.015 -0.257 1.000 -0.219 -0.321 -0.270 -0.056 -0.062 -0.111  
## V5 -0.105 -0.120 -0.248 -0.219 1.000 -0.113 -0.109 -0.100 -0.091 -0.081  
## V6 -0.044 -0.119 -0.246 -0.321 -0.113 1.000 0.403 -0.011 0.079 0.063  
## V7 0.029 0.001 -0.192 -0.270 -0.109 0.403 1.000 -0.067 0.027 -0.027  
## V8 0.167 0.031 0.098 -0.056 -0.100 -0.011 -0.067 1.000 -0.070 0.050  
## V9 -0.120 -0.042 -0.164 -0.062 -0.091 0.079 0.027 -0.070 1.000 0.089  
## V10 0.007 0.060 -0.125 -0.111 -0.081 0.063 -0.027 0.050 0.089 1.000  
## V11 0.040 0.062 0.026 0.058 -0.013 -0.080 -0.014 0.044 -0.020 0.067  
## V12 -0.014 -0.018 -0.098 -0.125 -0.019 0.097 0.044 -0.021 0.059 -0.005  
## V13 0.109 0.069 -0.147 -0.105 -0.055 0.005 -0.005 0.151 -0.031 -0.005  
## V14 0.099 0.063 -0.154 -0.258 -0.056 0.030 -0.003 0.017 0.019 0.047  
## V15 -0.073 -0.060 -0.133 -0.132 0.027 0.017 -0.029 -0.052 0.118 0.037  
## V16 0.001 -0.002 -0.097 -0.100 -0.032 -0.005 -0.034 -0.037 0.051 0.068  
## V11 V12 V13 V14 V15 V16  
## V1 0.040 -0.014 0.109 0.099 -0.073 0.001  
## V2 0.062 -0.018 0.069 0.063 -0.060 -0.002  
## V3 0.026 -0.098 -0.147 -0.154 -0.133 -0.097  
## V4 0.058 -0.125 -0.105 -0.258 -0.132 -0.100  
## V5 -0.013 -0.019 -0.055 -0.056 0.027 -0.032  
## V6 -0.080 0.097 0.005 0.030 0.017 -0.005  
## V7 -0.014 0.044 -0.005 -0.003 -0.029 -0.034  
## V8 0.044 -0.021 0.151 0.017 -0.052 -0.037  
## V9 -0.020 0.059 -0.031 0.019 0.118 0.051  
## V10 0.067 -0.005 -0.005 0.047 0.037 0.068  
## V11 1.000 0.006 -0.031 0.044 -0.001 0.035  
## V12 0.006 1.000 0.038 0.051 0.017 0.050  
## V13 -0.031 0.038 1.000 0.058 0.035 0.057  
## V14 0.044 0.051 0.058 1.000 0.062 0.080  
## V15 -0.001 0.017 0.035 0.062 1.000 0.057  
## V16 0.035 0.050 0.057 0.080 0.057 1.000

The median Q3 which is -0.005 which is very close to 0 meaning the item pairs do not show any significant local dependence. The maximum Q3 is around 0.403 especially for items 6 and 7 which are the relatively higher showing certain level of local dependence. However, majority of the items are close to 0 which means there is no local dependence.