Assignment 2 solutions

10/1/2024

EPSY 6637 Assignment #2 Fall 2024

This assignment is due on Monday September 30th.

1) Suppose the 6 items on test had the following item parameters for a 2PL:

```
      a    b     c
```

[1,] 1.2 -1.5 0 [2,] 2.5 -1.0 0 [3,] 1.2 0.0 0 [4,] 2.0 0.5 0 [5,] 1.8 1.0 0 [6,] 1.5 1.8 0

a) Find the Maximum Likelihood Estimate (MLE) for a person with response pattern (1,0,1,0,1,0), where 1 = correct/endorse.

Make sure to have the loglike function loaded up. Then make a matrix with the given parameters.

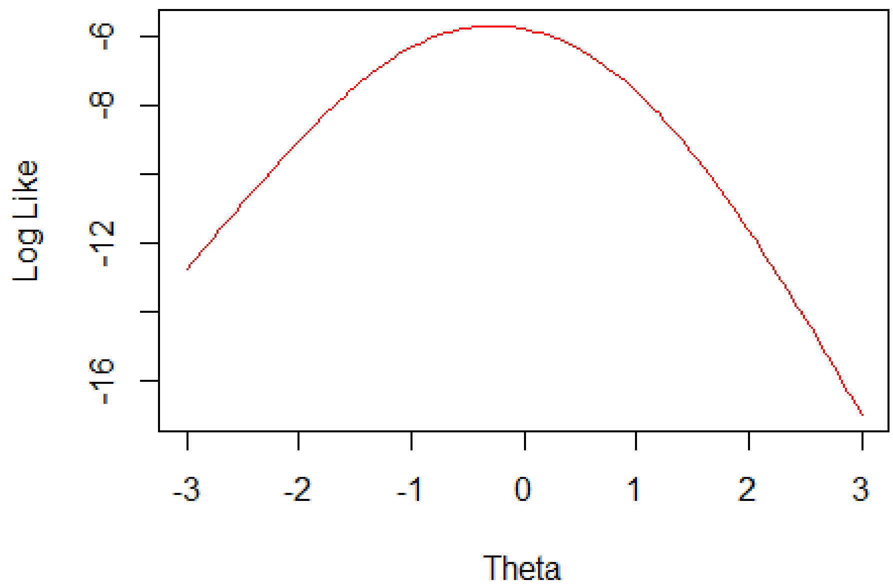
```
q1params<-matrix(c(1.2,-1.5,0,2.5,-
1.0,0,1.2,0.0,0,2.0,0.5,0,1.8,1.0,0,1.5,1.8,0),nrow=6, byrow=T)

response<-c(1,0,1,0,1,0)

getloglike(6,q1params,response) ->q1a

seq(-3,3,.05)[which.max(q1a)]

## [1] -0.3
```

The MLE is thus at $\hat{\theta} = -0.3$ *as this is the value of* θ *that maximizes the likelihood of the data.*

b) Plot the likelihood on a grid of theta values from -3 to 3, at .05 intervals (that is, use the scale we've been commonly using). Comment on how the shape, all the way across the range of theta, makes sense given the response vector.

In the plot below we see that the function is peaked at -0.3, consistent with what we found in part a. The person got a mix of answers correct and incorrect, it makes sense that the function falls off to the left (low θ*) and to the right (high* θ*).*

```
plot(seq(-3,3,.05),q1a,type="l",col="red",xlab="Theta",ylab="Log Like")
```

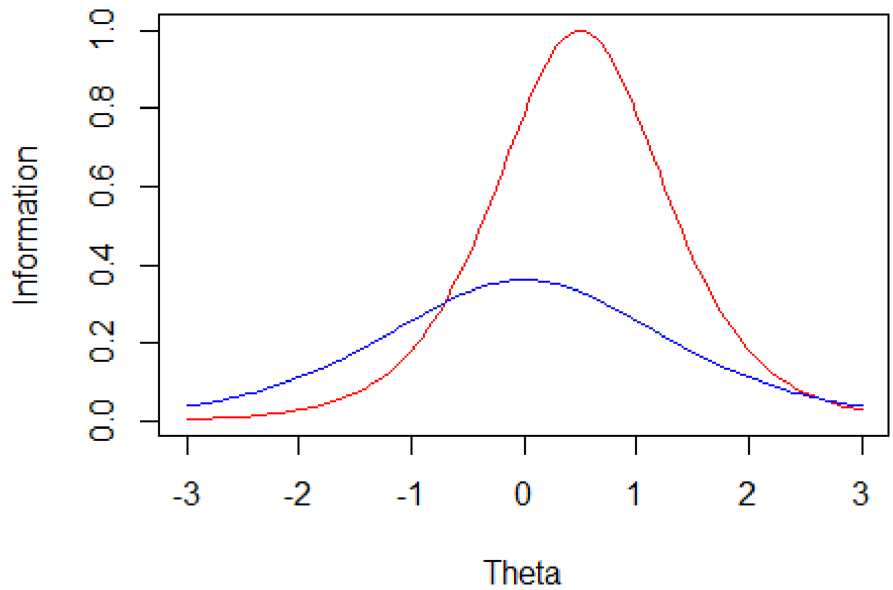


c) Plot the information function for item 3. Is the peak where you would expect it given the parameters for item 3? Will the peak information for this item be greater than or less than the peak for item 4? Explain.

For good measure, the code below plots the information functions for both items 3 (blue) and 4 (red) at the same time. With respect to item three, it has $b_3 = 0$ *so we expect maximum information at 0.*

If we then consider item 4, $a_4 = 2$ *which is much bigger than the slope for item 3, which means we expect the information function to be more sharply peaked and higher. It will also be shifted left to* $b = 0.5$ *.*

```
item<-q1params[3,]
temp<- item[1]*(seq(-3,3,.05) - item[2])
info3<-item[1]^2*(1-item[3])/((item[3] + exp(temp))*(1 + exp(-temp))*(1 +
exp(-temp)))

item<-q1params[4,]
temp<- item[1]*(seq(-3,3,.05) - item[2])
info4<-item[1]^2*(1-item[3])/((item[3] + exp(temp))*(1 + exp(-temp))*(1 +
exp(-temp)))

plot(seq(-3,3,.05),info4, type="l",col="red",xlab="Theta",ylab="Information")

points(seq(-3,3,.05),info3, type="l",col="blue")
```



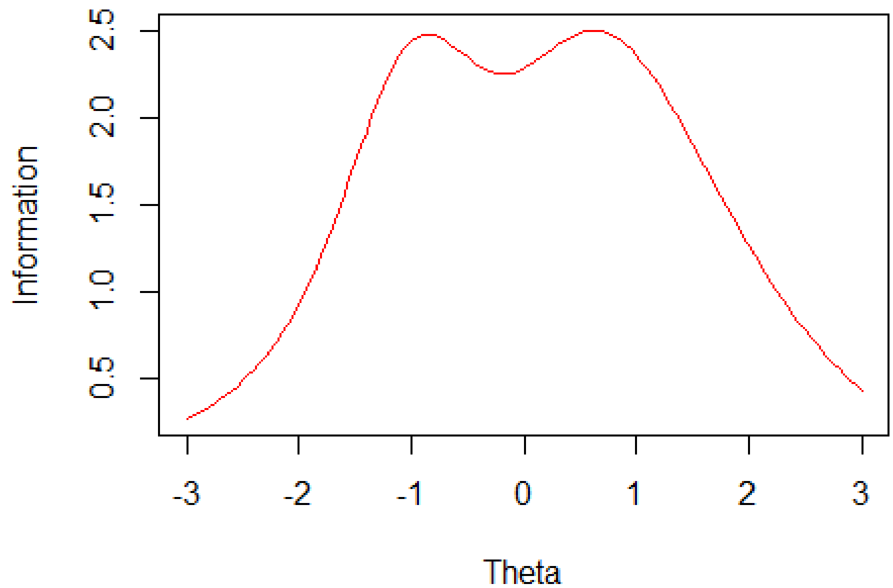
d) Plot the item information function for the entire test. Comment on the shape. What would be the approximate minimum standard error of estimation for a specific value of theta?

Make sure you have the infogrid function loaded.

The plot below has a curious double hump. That's just because it is a small test with a couple of more influential (higher discrimination) items. In general, test information functions will aggregate to a smooth function often centered in the middle. However, if we had a 50 item test, with half the items super easy and half super hard, then the function would have two separate and distinct peaks.

Here the peak of the total information is at about 2.5. Therefore the minimal standard error of estimation will be about $1/\sqrt{2.5} = 0.63$ *which will occur for* θ *under the peaks.*

```
infogrid(6,q1params)->testinfo

plot(seq(-
3,3,.05),apply(testinfo,1,sum),type="l",col="red",xlab="Theta",ylab="Informat
ion")
```



e) Imagine you could generate 6 new items that had item parameters exactly like the first 6. What would the new minimum standard error of estimation be?

If we had access to another set of 6 items the same as these, and then wanted to know what the new standard error of estimation would be, we just need to double the information:
$1/\sqrt{5} = 0.45$

2) In the class workspace we have already worked with the dataset called astro. It contains correct and incorrect responses from a 27 question astronomy exam.

a) Run 1, 2 and 3 PL models for the data. (Recall that mirt calls these itemtype= "Rasch", "2PL" and "3PL" respectively.) Compare the likelihood values, the AIC and the BIC. Which model do you think you would select based on these criteria alone?

```r
library(mirt)

mirt(astro,1,"Rasch") ->m1
mirt(astro,1,"2PL") ->m2
mirt(astro,1,"3PL") ->m3
```

	Loglike	AIC	BIC		Loglike	AIC	BIC
Astro,1PL	-1914	3885	3966	astro[,-c(6,16)],1PL	-1837	3726	3801
Astro,2PL	-1848	3805	3960	astro[,-c(6,16)],2PL	-1796	3693	3837
Astro,3PL	-1814	3790	4022	astro[,-c(6,16)],3PL	-1765	3680	3895

Picture

The figure summarizes the log-likelihood, AIC and BIC for both this question and the one below. For now, focus on the left hand side of the chart. The log-likelihood keeps improving fr each model because adding parameters will never make the fit worse. The challenge is to figure out whether the improvement was worth the cost of making the model more complex.

The AIC penalizes the likelihood function by 2 times the number of parameters. The BIC penalizes teh log-likelihood by log(N) times the number of parameters. The means that the penality for adding parameters is higher for the BIC and therefore it tends to me more conservative than the AIC. The numbers in the table are $AIC = -2LL + 2k$ *and* $BIC = -2LL + log(N)k$ *where* k *is the number of parameters for each model.*

In this case, the AIC favors the 3PL and the BIC favors the 2PL.

b) Examine the item parameters for the 2PL model. Recall that in Assignment 1 there was a problem with one of the items. How has that manifested here? Are any of the item discriminations suspiciously large? Can you guess why?

The most glaring problem is with Item 6 which has a negative slope. In assignment 1 we saw that the item had a negative item-total correlation. The negative slope here indicates the same issue - people with higher θ *are more likely to answer the item correctly*

The other problematic item is number 16 which has a suspiciously high a value. This is an easy item with a 92 *correct rate. For some reason the curve is very steep, and it makes sense to drop this one too (unless we study and see that it is a super important item - like the suicide ideation item on a depression inventory).*

```
round(coef(m2,simplify=T,IRTpars=T)$items,2)

##          a     b g u
## V1    0.67  2.19 0 1
## V2    1.05  0.41 0 1
## V3    1.13 -0.83 0 1
## V4    1.40 -1.13 0 1
## V5    1.39 -0.55 0 1
## V6   -1.28 -2.29 0 1
## V7    0.96  0.02 0 1
## V8    1.84 -0.92 0 1
## V9    0.43 -0.01 0 1
## V10   3.34 -0.80 0 1
## V11   1.16 -1.49 0 1
## V12   0.58  0.57 0 1
## V13   1.63 -0.46 0 1
## V14   1.12 -0.62 0 1
## V15   0.31  0.41 0 1
## V16   5.78 -1.35 0 1
## V17   1.05 -0.53 0 1
## V18   2.18 -0.68 0 1
## V19   1.42  1.06 0 1
## V20   1.75 -0.59 0 1
## V21   1.41 -1.24 0 1
```

```
## V22  1.59 -0.32 0 1
## V23  1.73 -0.21 0 1
## V24  0.66  1.71 0 1
## V25  1.86 -0.29 0 1
## V26  0.95  1.33 0 1
## V27  0.39  2.34 0 1
```

c) Rerun part a, but instead of passing astro to mirt, pass astro[,-c(6,16)] which is the whole dataset except the data for items 6 and 16. Now which model is preferred in terms of the BIC?

Now we look at the right hand side of the table above. Note that we cannot compare the two sides because with two fewer items, the likelihood functions are different. But we can use the same relative comparisons to see that the AIC still favors the 3PL and the BIC now supports the Rasch model. Those two problematic items - with their very strange slopes (one negative and one super high) were causing the impetus for estimating different slopes for the items.

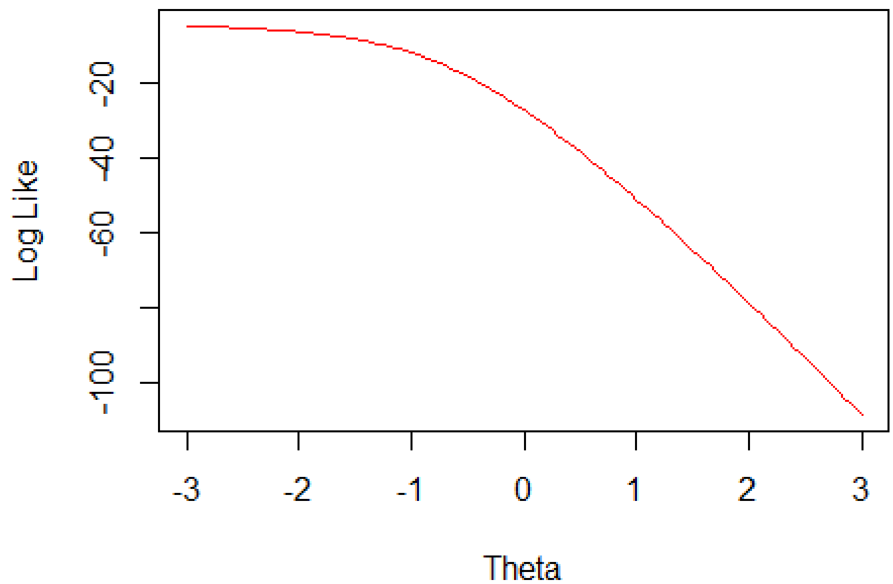
d) Assuming that in c you saved your model, get head(fscores(model,method="ML"). What are the theta estimates for the first and third persons in the class list?

```
head(fscores(m2b,method="ML"))

##             F1
## [1,] -3.638776
## [2,] -2.572494
## [3,] -1.218934
## [4,] -1.976212
## [5,] -1.522774
## [6,] -1.744056
```

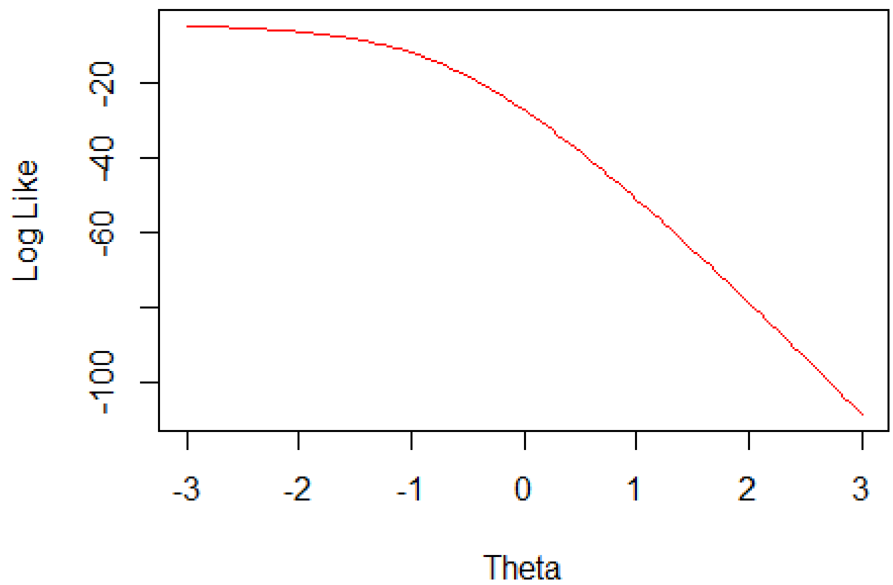
Person 1 is estimated at $\widehat{\theta_1} = -3.6$ *and person 3* $\widehat{\theta_3} = -1.2$*. These are the maximum likelihood estimates calculated after the parameters are estimated for the model.*

e) Use the getloglike function to confirm these estimates, assuming the model's item parameters. First pass the response vector for person 1, and then for person 3. Plot the likelihoods and confirm as best you can the agreement with the values from (d).

```
getloglike(25,coef(m2b,IRTpars=T,simplify=T)$items[,1:3],as.numeric(astro[1,-
c(6,16)]))->x1

plot(seq(-3,3,.05),x1,type="l",col="red",xlab="Theta",ylab="Log Like")
```




```r
which.max(x1)

## [1] 1

getloglike(25,coef(m2b,IRTpars=T,simplify=T)$items[,1:3],as.numeric(astro[3,-
c(6,16)]))->x3

plot(seq(-3,3,.05),x1,type="l",col="red",xlab="Theta",ylab="Log Like")
```




```
which.max(x3)

## [1] 37

seq(-3,3,0.05)[37]

## [1] -1.2
```

Using our function, and passing it the mirt model parameters and the astro response vectors, we get the same estimates for $\hat{\theta}$. Well, for person 1 it runs off our scale since we capped it at -3. but for person 3 we are bang on with the fscores estimate.

Hint: to get a matrix of the model item parameters: coef(model,IRTpars=T,simplify=T)$items[,1:3]

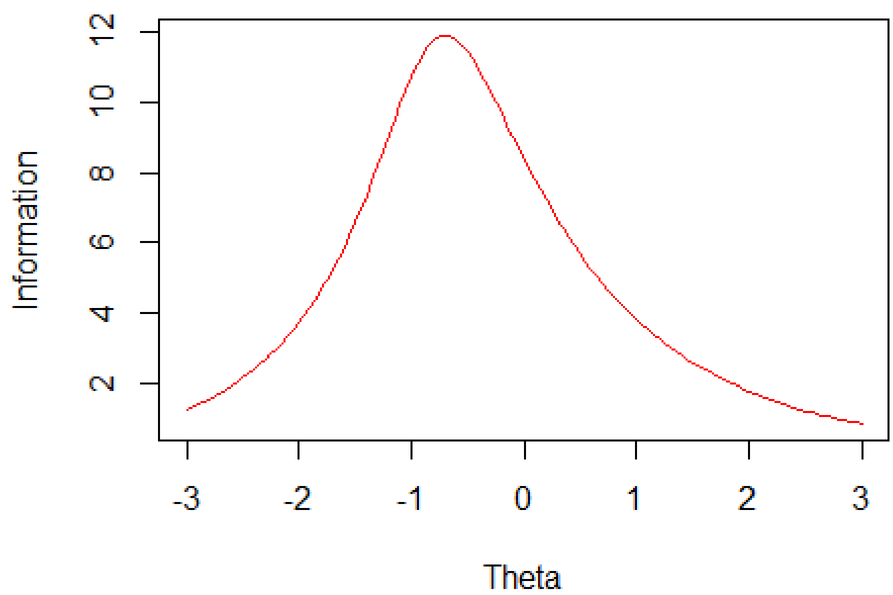
To make the response vectors: as.numeric(astro[1,-c(6,16)])

f) Use the infogrid function to calculate the information in the 2PL test (where we dropped 6 and 16). Make a plot of the test information function. Compare with plot(model,type=”info”).

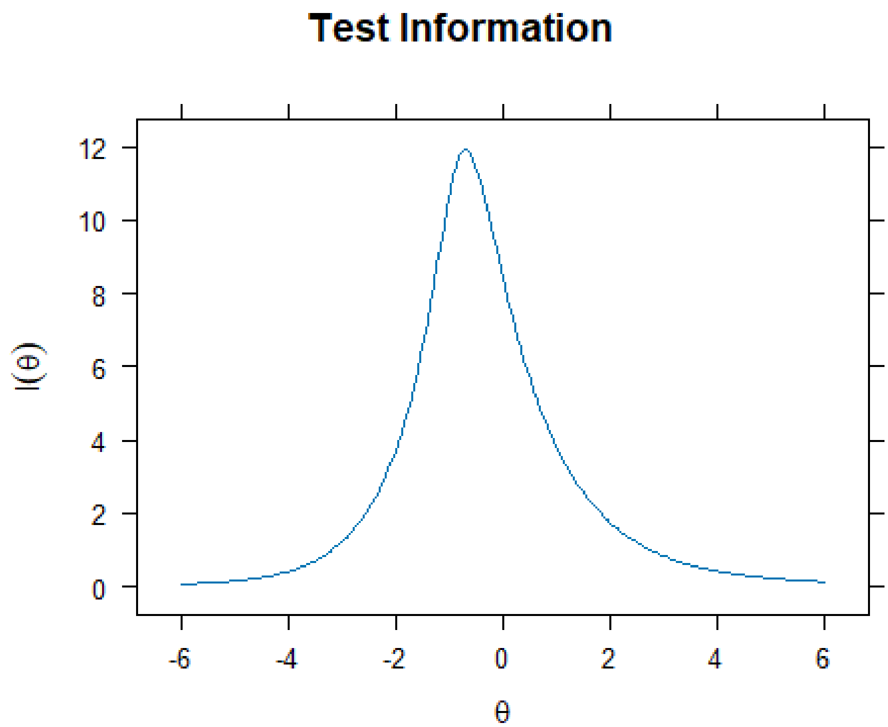
we have a nice information curve for the test. It peaks at about 12 for below average ability - therefore the SEE is about $1/\sqrt{12} = 0.28$ at those levels. The test provides predictions with much wider uncertainty at higher abilities.

Making plots like this is a default option in mirt, see next plot.

```
infogrid(25,coef(m2b,IRTpars=T,simplify=T)$items[,1:3])->q2testinfo


 plot(seq(-
3,3,.05),apply(q2testinfo,1,sum),type="l",col="red",xlab="Theta",ylab="Inform
ation")
```




```
 plot(m2b,type="info")
```



Extra note:

There is a mirt command **testinfo** which you can do after running a model.

testinfo(model, c(-1,0,1)) will get you the test information for theta = -1, 0, and 1 respectively.

You can put any values you want for theta – this just an example.