I used the following code to recode my qualitative responses into quantitative numbers.

```
data(package = "ltm")
library(tidyverse)
df<- Environment
x<-df %>% mutate_all(~ str_replace(., "^$", NA_character_)) %>% mutate_all(.funs = ~
as.integer(recode(.x = .,"not very concerned"=0,"slightly concerned"=1, "very concerned"=2)))
library("writexl" write_xlsx(df,"D:\\df.xlsx")
```

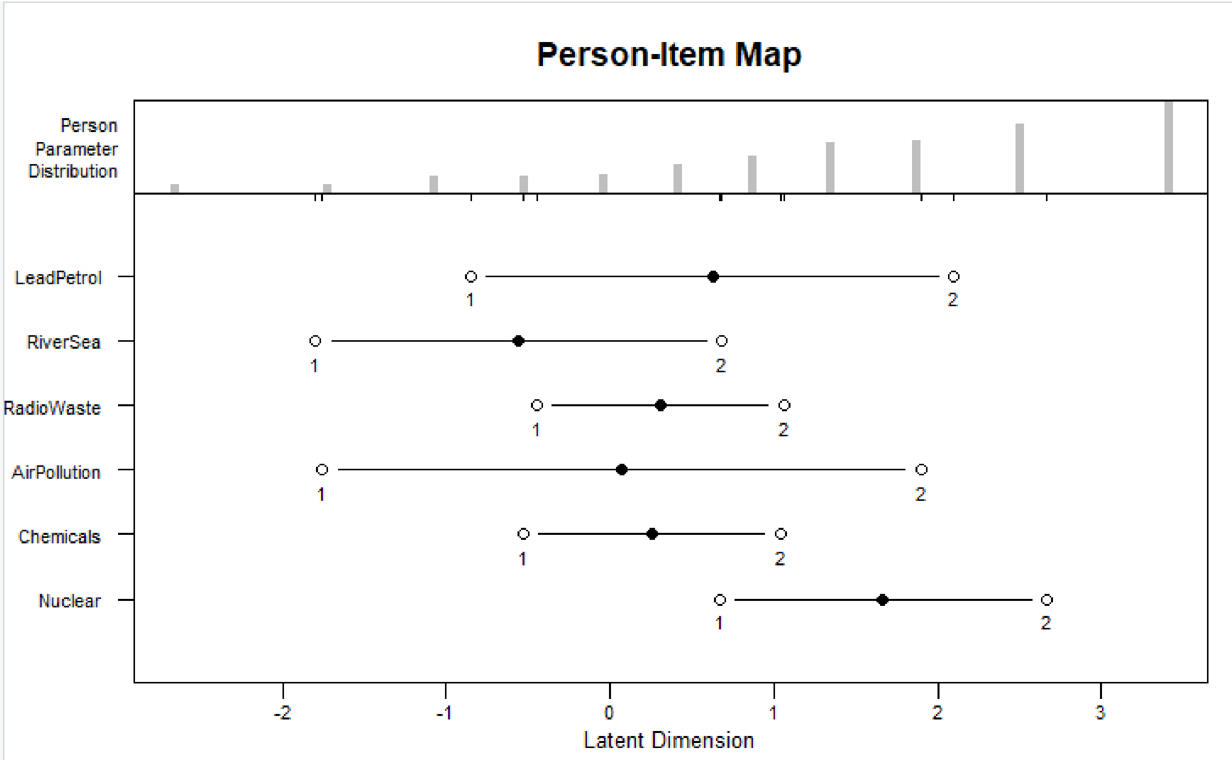
We would have ideally checked the assumptions first. The model does break the unidimensionality assumption. However i will still run the PCM model for the sake of presentation. Also the methods to check assumptions have been elaborated in the MIRT analysis of this data.

```
PC_model <- PCM(x)
> summary(PC_model)
```

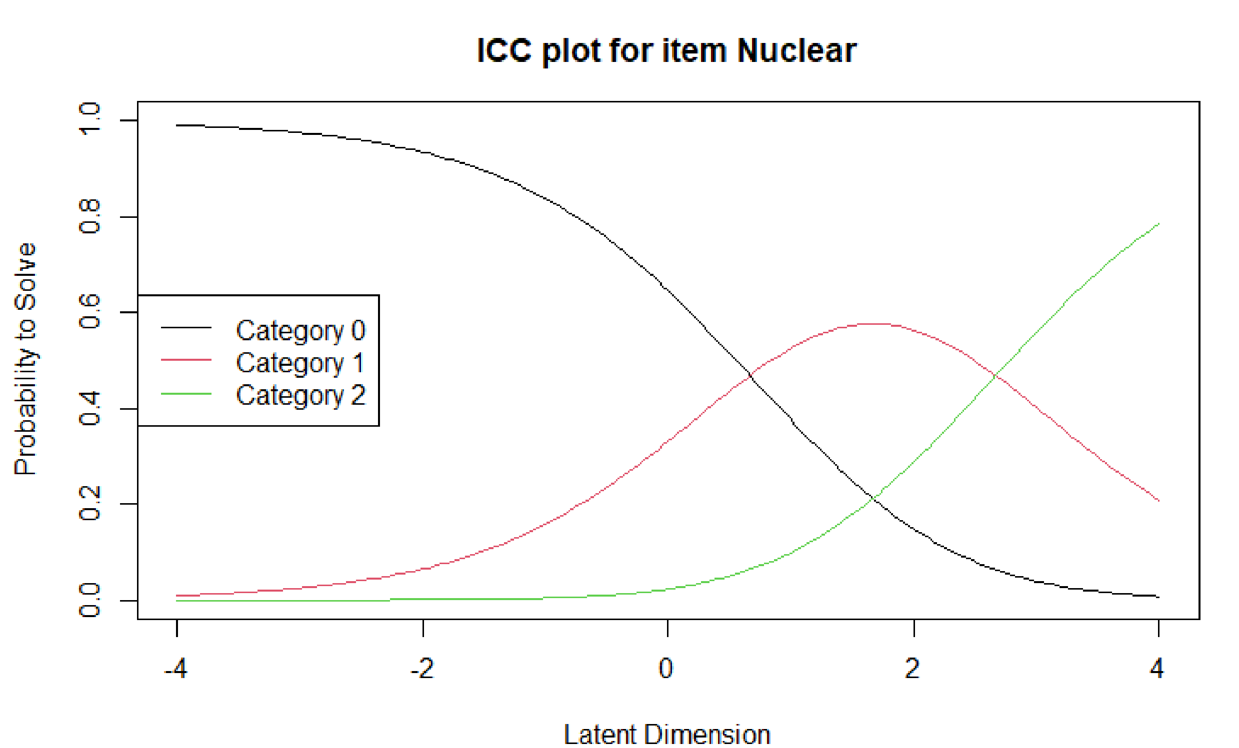
```
~/
Item (Category) Difficulty Parameters (eta): with 0.95 CI:
                 Estimate Std. Error lower CI upper CI
LeadPetrol.c2       1.249      0.320    0.621    1.877
RiverSea.c1        -1.800      0.456   -2.693   -0.907
RiverSea.c2        -1.117      0.423   -1.946   -0.287
RadioWaste.c1      -0.449      0.295   -1.028    0.130
RadioWaste.c2       0.612      0.314   -0.004    1.227
AirPollution.c1    -1.755      0.380   -2.501   -1.010
AirPollution.c2     0.140      0.378   -0.601    0.880
Chemicals.c1       -0.534      0.302   -1.125    0.057
Chemicals.c2        0.507      0.317   -0.115    1.129
Nuclear.c1          0.666      0.212    0.250    1.081
Nuclear.c2          3.334      0.308    2.731    3.937

Item Easiness Parameters (beta) with 0.95 CI:
                      Estimate Std. Error lower CI upper CI
beta LeadPetrol.c1       0.852      0.285    0.294    1.409
beta LeadPetrol.c2      -1.249      0.320   -1.877   -0.621
beta RiverSea.c1         1.800      0.456    0.907    2.693
beta RiverSea.c2         1.117      0.423    0.287    1.946
beta RadioWaste.c1       0.449      0.295   -0.130    1.028
beta RadioWaste.c2      -0.612      0.314   -1.227    0.004
beta AirPollution.c1     1.755      0.380    1.010    2.501
beta AirPollution.c2    -0.140      0.378   -0.880    0.601
beta Chemicals.c1        0.534      0.302   -0.057    1.125
beta Chemicals.c2       -0.507      0.317   -1.129    0.115
beta Nuclear.c1         -0.666      0.212   -1.081   -0.250
beta Nuclear.c2         -3.334      0.308   -3.937   -2.731

>
```



The map helps us understand the person item distribution along the logit scale. It seems that we might need to add some harder items to cover the higher range.



The ICC lists the probability of being successful given a certain theta

> item.estimates <- thresholds(PC_model)
> item.estimates
The following are the thresholds.

```
Design Matrix Block 1:
             Location Threshold 1 Threshold 2
LeadPetrol    0.62438    -0.85173     2.10050
RiverSea     -0.55829    -1.80028     0.68369
RadioWaste    0.30578    -0.44867     1.06023
AirPollution  0.06987    -1.75543     1.89517
Chemicals     0.25356    -0.53423     1.04134
Nuclear       1.66706     0.66562     2.66849

> |
```

The following are the threshold SE's

```
> item.se <- item.estimates$se.thresh
> item.se
  thresh beta LeadPetrol.c1   thresh beta LeadPetrol.c2     thresh beta RiverSea.c1
                  0.2845707                   0.2076824                   0.4555609
    thresh beta RiverSea.c2   thresh beta RadioWaste.c1   thresh beta RadioWaste.c2
                  0.2131129                   0.2953624                   0.2153098
thresh beta AirPollution.c1 thresh beta AirPollution.c2    thresh beta Chemicals.c1
                  0.3804246                   0.2030653                   0.3015543
   thresh beta Chemicals.c2      thresh beta Nuclear.c1      thresh beta Nuclear.c2
                  0.2148208                   0.2118805                   0.2245474
> |
```

The following are the person ability estimates:-

```
> person.locations.estimate <- person.parameter(PC_model)
> summary(person.locations.estimate)

Estimation of Ability Parameters

Collapsed log-likelihood: -49.9949
Number of iterations: 11
Number of parameters: 11

ML estimated ability parameters (without spline interpolated values):
             Estimate Std. Err.       2.5 %     97.5 %
theta P97  3.40784084 1.0991726  1.25350212  5.56217955
theta P98  3.40784084 1.0991726  1.25350212  5.56217955
theta P99  3.40784084 1.0991726  1.25350212  5.56217955
theta P100 3.40784084 1.0991726  1.25350212  5.56217955
theta P101 3.40784084 1.0991726  1.25350212  5.56217955
theta P102 3.40784084 1.0991726  1.25350212  5.56217955
theta P103 3.40784084 1.0991726  1.25350212  5.56217955
theta P104 3.40784084 1.0991726  1.25350212  5.56217955
theta P105 3.40784084 1.0991726  1.25350212  5.56217955
theta P106 3.40784084 1.0991726  1.25350212  5.56217955
theta P107 3.40784084 1.0991726  1.25350212  5.56217955
theta P108 3.40784084 1.0991726  1.25350212  5.56217955
theta P109 3.40784084 1.0991726  1.25350212  5.56217955
theta P110 3.40784084 1.0991726  1.25350212  5.56217955
theta P111 3.40784084 1.0991726  1.25350212  5.56217955
theta P112 3.40784084 1.0991726  1.25350212  5.56217955
theta P113 3.40784084 1.0991726  1.25350212  5.56217955
```

These are the item and person fit statistics. Their interpretation has been explained in the IRT dichotomous analysis. **We could have also calculated category based fit statistics.** I use Jmetrik to calculate those statistics

```
Itemfit Statistics:
               Chisq  df p-value Outfit MSQ Infit MSQ Outfit t Infit t Discrim
LeadPetrol   238.123 192   0.013      1.234     1.238    2.311   2.403   0.228
RiverSea     147.517 192   0.993      0.764     0.869   -1.450  -1.132   0.546
RadioWaste   132.420 192   1.000      0.686     0.708   -2.282  -2.908   0.676
AirPollution 138.950 192   0.999      0.720     0.749   -3.155  -2.970   0.637
Chemicals    131.490 192   1.000      0.681     0.732   -2.315  -2.628   0.653
Nuclear      193.197 192   0.462      1.001     1.020    0.043   0.246   0.384

> |
```

```
> item.fit.table <- cbind(item.fit[["i.outfitMSQ"]],item.fit[["i.infitMSQ"]],item.fit[["i.infitMSQ"]],
item.fit[["i.infitZ"]])
> pfit <- personfit(person.locations.estimate)
> pfit

Personfit Statistics:
     Chisq df p-value Outfit MSQ Infit MSQ Outfit t Infit t
P97  4.407  5   0.492      0.734     0.957     0.09    0.23
P98  4.407  5   0.492      0.734     0.957     0.09    0.23
P99  4.407  5   0.492      0.734     0.957     0.09    0.23
P100 4.407  5   0.492      0.734     0.957     0.09    0.23
P101 4.407  5   0.492      0.734     0.957     0.09    0.23
P102 4.407  5   0.492      0.734     0.957     0.09    0.23
P103 4.407  5   0.492      0.734     0.957     0.09    0.23
P104 4.407  5   0.492      0.734     0.957     0.09    0.23
P105 4.407  5   0.492      0.734     0.957     0.09    0.23
P106 4.407  5   0.492      0.734     0.957     0.09    0.23
P107 4.407  5   0.492      0.734     0.957     0.09    0.23
P108 4.407  5   0.492      0.734     0.957     0.09    0.23
P109 4.407  5   0.492      0.734     0.957     0.09    0.23
P110 4.407  5   0.492      0.734     0.957     0.09    0.23
P111 2.272  5   0.810      0.379     0.621    -0.40   -0.24
P112 2.272  5   0.810      0.379     0.621    -0.40   -0.24
P113 2.272  5   0.810      0.379     0.621    -0.40   -0.24
P114 2.272  5   0.810      0.379     0.621    -0.40   -0.24
P115 2.272  5   0.810      0.379     0.621    -0.40   -0.24
P116 2.272  5   0.810      0.379     0.621    -0.40   -0.24
P117 2.272  5   0.810      0.379     0.621    -0.40   -0.24
P118 2.272  5   0.810      0.379     0.621    -0.40   -0.24
```

Person/Item separation reliability is the estimate of how well we can differentiate individual items, persons, or other elements on the latent variable.

```
> # ===================================
> # compute item separation reliability
> # ===================================
>
> # Get Item scores
> ItemScores <- colSums(x)
>
> # Get Item SD
> ItemSD <- apply(x,2,sd)
>
> # Calculate the se of the Item
> ItemSE <- ItemSD/sqrt(length(ItemSD))
>
> # compute the Observed Variance (also known as Total Person Variability or Squared Standard Deviatio
n)
> SSD.ItemScores <- var(ItemScores)
>
> # compute the Mean Square Measurement error (also known as Model Error variance)
> Item.MSE <- sum((ItemSE)^2) / length(ItemSE)
>
> # compute the Item Separation Reliability
> item.separation.reliability <- (SSD.ItemScores-Item.MSE) / SSD.ItemScores
> item.separation.reliability
[1] 0.9999675
>
```

```
> # Get Person scores
> PersonScores <- rowSums(x)
>
> # Get Person SD
> PersonSD <- apply(x,1,sd)
>
> # Calculate the se of the Person
> PersonSE <- PersonSD/sqrt(length(PersonSD))
>
> # compute the Observed Variance (also known as Total Person Variability or Squared Standard Deviatio
n)
> SSD.PersonScores <- var(PersonScores)
>
> # compute the Mean Square Measurement error (also known as Model Error variance)
> Person.MSE <- sum((PersonSE)^2) / length(PersonSE)
>
> # compute the Person Separation Reliability
> person.separation.reliability <- (SSD.PersonScores-Person.MSE) / SSD.PersonScores
> person.separation.reliability
[1] 0.9998872
>
```

We could have done a post hoc analysis by performing a PCA on the residuals. I have explained that in the IRT dich analysis.