THIS IS NOT THE VERSION OF RECORD. PLEASE VISIT
MEASUREMENT INTERDISCIPLINARY RESEARCH
AND PERSPECTIVES AT

https://doi.org/10.1080/15366367.2020.1732155

FOR THE OFFICIAL MANUSCRIPT

Using the eRm package for Rasch modeling

R. Noah Padgett & Grant B. Morgan

Baylor University

Author Note

R. Noah Padgett, Department of Educational Psychology, Baylor University; Grant B. Morgan, Department of Educational Psychology, Baylor University.

This research was supported by Baylor University and the Department of Educational Psychology as funding toward the completion of R. Noah Padgett's doctoral degree.

Correspondence concerning this article should be addressed to R. Noah Padgett, Department of Educational Psychology, One Bear Place # 97304, Baylor University, Waco, TX 76798. Contact: noah_padgett1@baylor.edu

Abstract

The "extended Rasch modeling" (eRm) package in R provides users with a comprehensive set of tools for Rasch modeling for scale evaluation and general modeling. We provide a brief introduction to Rasch modeling followed by a review of literature that utilizes the eRm package. Then, the key features of the eRm package for scale evaluation are reviewed. The ease of use and the advantages of the R environment make the eRm package an appealing free option for rigorous Rasch modeling. We demonstrate the functionality with a small example using data from the Rosenberg Self-Esteem scale. Using these data we show how to 1) fit dichotomous and polytomous Rasch models, 2) obtain key figures (e.g., item characteristic curves and person-item maps) to aid model assessment , 3) assess dimensionality, and 4) obtain item fit statistics.

Keywords: software review, Rasch, eRm, R

Using the eRm package for Rasch modeling

In educational settings, data are often of the ordered categorical type given the common use of various response formats (e.g., correct-incorrect, Likert-type, etc.). The use of the ordered categorical format comes from the need for obtaining more precise scoring of items (Tuerlinckx & Wang, 2004) and, therefore, a more accurate measurement of the construct of interest. Along with advancements in item development, advanced statistical techniques are necessary for evaluating and scoring tests. Many techniques have been developed to analyze such polytomous data, including extensions of the Rasch model (Rasch, 1960); item response theory (IRT; de Ayala, 2009; de Boeck & Wilson, 2004), exploratory factor analysis (EFA), confirmatory factor analysis (CFA; Brown, 2015), and structural equation modeling (SEM; Kaplan, 2009; Kline, 2015; Schumacker & Lomax, 2016).

The variety of analytical choices may be daunting at first for an analyst that is unfamiliar with the numerous approaches available. However, the frameworks listed above have unique features that could help answer a wide range of research questions. Deciding which approach is appropriate requires understanding the type of question asked. For example, EFA is commonly used to help find evidence for scale development to help assess the dimensionality of an underlying construct of interest (Benson & Nasser, 1998). However, when more fine grained information about a the performance of a particular item is of interest, an IRT or Rasch framework is frequently used (de Ayala, 2009). IRT and Rasch, though mathematically similar, offer distinctly different viewpoints on item and scale evaluation. From an IRT viewpoint, researchers typically aim to identify an IRT model that best explains the observed data (de Ayala, 2009). While from a Rasch perspective, the aim is to find evidence that the data at hand support a chosen Rasch model (Smith & Smith, 2004). The focus of this review is on scale evaluation within a Rasch framework.

A consideration for any analysis nowadays is to figure out which software to use.

Some popular Rasch software tools are *Winsteps* (Linacre, 2019b) and *Facets* (Linacre, 2019a). However useful these software tools are, many such tools are not free. In this review, we focus on the free *extended Rasch modeling* (eRm) package (Mair & Hatzinger, 2007; Mair et al., 2013). The eRm package is built in R (R Core Team, 2019), meaning that there exists a large ecosystem of open-source tools and support available to researchers and practitioners in case an issues arises. In this review and brief tutorial, we describe the features of the *eRm* package, and illustrate how to utilize the package for item and scale analysis through a Rasch framework.

Rasch and the eRm Package

The use of Rasch methods comes with some particularly unique features that can be utilized when analyzing ordinal data. When the Rasch model assumptions are tenable, the Rasch model can provide researchers with 1) a common, interval-level metric by which persons and items are measured, 2) fit statistics to evaluate which (if any) persons or items do not fit the model, and 3) evaluation of coverage of the latent construct. The use of Rasch models is an important step for understanding the characteristics of an assessment. Importantly, the Rasch models places persons and items on a common, sample-independent scale that allows for a rigorous investigation of the data.

The eRm package was built to provide modeling flexibility for the user to explore a wide array of models. The vast array of models are built from the underlying flexibility of the *linear partial credit model* (LPCM). The LPCM is basis for the general architecture by which Rasch models can be specified where convenient wrapper functions are available for important special cases such as the partial credit model (PCM),rating scale model (RSM), and Rasch Model. The hierarchy of extended Rasch models is broken up into two major strands: one for the traditional RM, RSM, and PCM, and the other for the linearized counterparts linear logistic test model (LLTM), linear RSM (LRSM), and LPCM. These counterparts are simply specializations of one another. The specification of the wide range of models comes from the implementation of a *design matrix*. A large portion of the functionality of the design matrix is out of scope for this review, but the interested reader can refer to Mair and Hatzinger (2007) and Hatzinger and Rusch (2009) for more details. Some of the functionality is reviewed next through examples in the literature that utilize the eRm package.

Literature Utilizing eRm

The eRm has been reported in literature to help answer a variety of questions. Some researchers have used the package to assist with Monte Carlo simulation studies (Edelsbrunner & Dablander, 2019; Krammer, 2019; Strobl et al., 2015). For example, in Edelsbrunner and Dablander (2019), as part of their review of scientific reasoning research the authors conducted a Monte Carlo simulation study to investigate how INFIT mean-square was sensitive to violations of unidimensionality. They used the eRm package to generate data from a four-dimensional Rasch model then fit a unidimensional model to these data. These authors capitalized on the data simulation procedures in the eRm package to generate data specifics types of assumption violations to conduct a small scale focused simulation study. Others have reportedly used the eRm to help find validity evidence during the development of a scale (Conrad et al., 2014; Elsman et al., 2019). For example, in Elsman et al. (2019), the authors investigated fit of their data to the partial credit model on an assessment of quality of life. The use of the eRm package for Rasch modeling has spread to a variety of disciplines and has been used in a to help address a wide range of issues. Next, we discuss the features of the eRm package in more detail starting with the estimation underlying many of the Rasch models.

Estimation

With respect to estimation, the eRm package uses conditional maximum likelihood (CML) for the dichotomous and polytomous models. CML (Andersen, 1972) may be used to estimate models for which there are sufficient statistics available, as is the case for the family of Rasch models. That is, the item mean and person mean are sufficient statistics for the item and person measures, respectively. Given this property of Rasch models, joint maximum likelihood (JML), or unconditional maximum likelihood (Wright & Stone, 1979), is also available and is employed by other Rasch software, such as WINSTEPS (Linacre, 2019b). However, JML cannot produce estimates for individuals with sum scores of zero or the maximum possible score (e.g., all items incorrect or all items correct), and the estimates produced are inconsistent and asymptotically normal (Andersen, 1973; Haberman, 1977, 2004). Additionally, the limitations of JML are due to the simultaneous estimation of item and person measures. On the other hand, CML does not have the same limitations as JML so the CML estimator can be viewed as a strength if considering different software for modeling. For instance, CML can produce consistent maximum likelihood estimates because it separates the item and person parameter estimates by conditioning the estimation of the likelihood function on the person sufficient statistics (Andersen, 1972; de Ayala, 2009). The person measures are estimated in eRm in a step following the estimation of item measures using JML. This two-step process sidesteps the limitations of JML because (1) the parameters are not estimated simultaneously and (2) the item parameters estimated through CML can be treated as "known" (i.e., fixed) in order to estimate the person measures. The two-step process uses the advantages of both estimation procedures. The estimates for persons and items can be viewed in the person-item map using the plotPImap command, which is a very attractive graphic compared with other software. CML is used for the dichotomous (RM), partial credit (PCM), and rating scale (RSM) models as well as the linear versions of these models (LLTM, LPCM, LRSM; commands shown in parentheses).

Model Diagnostics

The eRm package offers an array of tests and statistics to assist with diagnosing multiple areas of potential model misfit. First, the information-weighted mean square (INFIT) and unweighted mean square (OUTFIT) estimates are provided for each item and each person, which is customary in evaluating Rasch model-data fit. These estimates along with their χ^2 values, degrees of freedom, p value, and t test statistics are available with the itemfit and personfit commands, respectively, for items and persons. The INFIT t test statistics can also be plotted for items or persons using the plotPImap command, in which users may plot with or without confidence intervals. This plot may be particularly helpful in examining many items or person efficiently.

Secondly, the eRm package offers the Anderson likelihood ratio test (Andersen, 1973) using the LRtest command. The procedure tests for person homogeneity, which can be an indication of differential item function or violations of monotonicity, by estimating the item measures using two or three subgroups from the sample. Two or three subgroups are the only options, but this will likely meet most users' needs. Invariance between the two or three sets of parameter estimates supports the model-data fit. The LRtest command produces the LR value (i.e., $\sim \chi^2$), degrees of freedom, and p value. By default, two groups are formed based on raw scores above or below the median. Subgroups can also be created by the user based on a variable coded as a factor (e.g., sex) or based on the distribution of a quantitative variable. Because the test divides the sample in subgroups, a sufficiently large sample size is an important consideration to ensure that all subgroups are large enough.

Third, the generalized Martin-Löf test is available for testing dimensionality, which is a different approach from the conventional method of conducting principal components analysis on the standardized residual matrix. Martin-Löf (1970) proposed a test of unidimensionality for dichotomous Rasch models that is a likelihood ratio test for comparing the null hypothesis of unidimensionality against the alternative hypothesis of two dimensions. Christensen et al. (2002) generalized the test for (1) polytomous models and (2) an alternative hypothesis with more than two dimensions. The MLoef procedure splits the items into two subgroups using the median (default) or mean of the item raw scores, or the user can specify a grouping of two or three subgroups of items. The command produces the log-likelihood value for each item grouping as well as for the overall model. The MLoef procedure also produces the likelihood ratio value, degrees of freedom, and p value, which is the output that is examined when making a decision about the tenability of the unidimensionality assumption. The MLoef command is a large sample approximation (i.e., $\sim \chi^2$), but the eRm package also provides an exact test using the NPtest command. This command offers 12 different methods for testing unidimensionality with the default method checking for local dependence based on inter-item correlations. (Verguts2000) showed that the Martin-Löf test is a bit conservative unless used with a large sample but that it has appropriate power when the split of items is correct. (Christensen2007) showed that the Martin-Löf test works well but notes the requirement of a very large sample for the null distribution to follow the asymptotic χ^2 distribution. In light of the available research on the Martin-Löf test and its generalization, the exact test (`NPtest`) may be an attractive option for some users.

Tutorial on Using eRm for Rasch Analyses

Generally, the use of R for statistical analyses has grown exponentially in recent years given the increase in personal computing power and continued development of open source tools. In our demonstration of the `eRm` package use for Rasch modeling, we highlight various functionalities through a guiding example with data on the Rosenberg Self-Esteem Scale (RSES; Rosenberg, 1989). The RSES is analyzed from two perspectives. First, using the Rasch model for binary outcomes to discuss the use of the classic Rasch model. Secondly, using the partial-credit model (PCM; Masters, 1982) as a polytomous extension of the Rasch model.

Data and Instrument

Data used in the following example were collected as part of previous research (DiStefano & Morgan, 2011; Morgan et al., 2016) and used here to show the utility of the `eRm` package. Briefly, these data contain 757 complete responses to the Rosenberg Self-Esteem scale providing measures of global self-esteem (Rosenberg, 1989). The Rosenberg Self-Esteem scale used here consists of 10 items, five of which were positively worded and five negatively worded. All items on the scale were rated on a four-point Likert scale with verbal anchors of Strongly Disagree (0), Disagree (1), Agree (2), and Strongly Agree (3). The frequency of responses to each category are reported in Table 1. The survey respondents were an average age was 22.4 years (SD $\pm$ 7.2 years) ranging from 16 to 75 years. The sample was 42% male, 48% female, and the remaining 10% did not disclose sex. The respondents were 67.9% Caucasian, 12.8% Black, 2.4% Hispanic, 3.4% Asian, 1.7% Multiracial, 0.3% other races, and 11.5% did not report race.

Responses to these items are analyzed in two ways in the example Rasch analyses that follow. First, we dichotomized these responses based on direction of agreement to create binary data. These artificially binary data were then analyzed with the Rasch model. Secondly, we used the original responses and analyzed these data with the partial credit model. For each analysis we describe how to 1) fit the model, 2) plot the item characteristic curves (ICC), 3) obtain person-item maps, 4) assess dimensionality and tenability of other assumptions, and 5) obtain item/person fit statistics.

Table 1

Category response frequency across items

	Item									
Response	1	2	3	4	5	6	7	8	9	10
SD	18	66	13	30	14	69	32	14	37	18
D	27	254	32	89	87	250	75	63	168	68
A	280	284	313	256	390	268	189	371	258	363
SA	432	153	399	382	266	170	461	309	294	308

Note. SD = Strongly Disagree; D = Disagree; A = Agree; and SA = Strongly Agree.

Applying the Rasch Model in eRm

Applying the Rasch Model in eRm is straightforward once data are read into the R environment. Here, we demonstrate how the eRm package can be used to analyze data from the RSE scale. The top six rows of our data are

```
> head(data)
  Item_1 Item_2 Item_3 Item_4 Item_5 Item_6 Item_7 Item_8 Item_9 Item_10
1      1      1      1      1      1      1      1      1      1       1
2      1      1      1      1      1      0      1      1      1       1
3      1      0      1      1      1      1      1      1      1       1
4      1      1      1      1      1      1      1      1      1       1
5      1      0      1      1      1      1      1      1      0       1
6      1      1      1      1      1      1      1      1      1       1
```

where the head function was used to just look at these top six rows (or participants) in our data (named rse_dich in R for dichotomously scored RSE data). Most of the responses are 1's, which makes sense given that a large proportion of responses shown in Table 1 are on the agree side of the response scale. Based on the agreeable response versus a disagreeable response, these data may be reflective of a person's level of self-esteem.

The RSE scale is thought to measure a single underlying construct, or said another way, a unidimensional construct. The Rasch model was easily fit to these data and obtain the model results using the RM function (see below).

```
> fit.rm <- RM(data)
> summary(fit.rm)

Results of RM estimation:

Call:  RM(X = data)
```

```
Conditional log-likelihood: -1309.919
Number of iterations: 16
Number of parameters: 9

Item (Category) Difficulty Parameters (eta): with 0.95 CI:
        Estimate Std. Error lower CI upper CI
Item_2     2.101      0.102    1.901    2.302
Item_3    -1.586      0.168   -1.916   -1.256
Item_4    -0.059      0.117   -0.289    0.171
Item_5    -0.341      0.123   -0.582   -0.099
Item_6     2.093      0.102    1.892    2.293
Item_7    -0.243      0.121   -0.480   -0.006
Item_8    -0.781      0.136   -1.047   -0.515
Item_9     1.007      0.104    0.803    1.211
Item_10   -0.605      0.130   -0.860   -0.350

Item Easiness Parameters (beta) with 0.95 CI:
             Estimate Std. Error lower CI upper CI
beta Item_1     1.586      0.168    1.256    1.916
beta Item_2    -2.101      0.102   -2.302   -1.901
beta Item_3     1.586      0.168    1.256    1.916
beta Item_4     0.059      0.117   -0.171    0.289
beta Item_5     0.341      0.123    0.099    0.582
beta Item_6    -2.093      0.102   -2.293   -1.892
beta Item_7     0.243      0.121    0.006    0.480
beta Item_8     0.781      0.136    0.515    1.047
beta Item_9    -1.007      0.104   -1.211   -0.803
beta Item_10    0.605      0.130    0.350    0.860
```

A feature of the eRm package is the item easiness parameters are also reported along with the item difficulties. The default settings fix the scale by fixing the sum of item easiness parameters to zero. The scale can easily be set by fixing the first easiness (beta) to 0 instead by using sum0=FALSE as an argument in the function. These options provide users for flexibility in modeling and even greater flexibility can be achieved by utilizing the design matrix formulation.

However, for the given model, more detailed information about the fitted model can be investigated next. The item characteristic curves for all ten items can be easily obtained using the plotICC command. The command nicely plots the ICC for each item on a separate window so that each item can be inspected individually. All the ICCs can be plotted on the same window as well, using plotjointICC, which is shown in Figure 1.

The ICC plot highlights how the items are distributed and ordered the item legend based on the difficulty. The difficulty in jointly plotting all the ICCs is that some of the curves can be difficult to see (e.g., item 7). Another useful feature for examining the relationship between persons and items is the person-item map. These figures help provide a useful representation of how the difficulty of items relates to

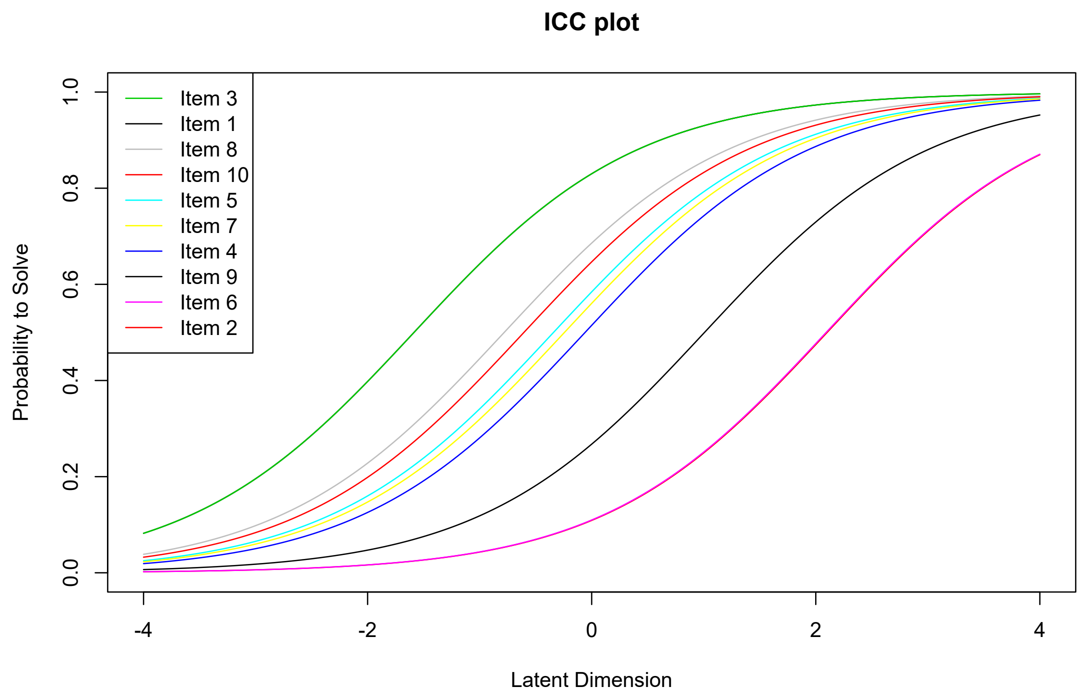

Figure 1
Joint ICC plot for Rasch model

the person-parameters for the fitted Rasch model. This map is created using plotPImap(fit.rm, sorted = TRUE), where the argument "sorted = TRUE" is not necessary, but will help significantly with interpretation and identifying how items are distributed across the person parameter distributions. The person-item map is shown in Figure 2.

The fit of these data to the Rasch model can be evaluated in numerous ways in the eRm package. The package supports tests such as Anderson's likelihood ratio test, Wald-type tests, Martin-Löf test, various nonparametric tests, item and person fit indices, and graphical procedures. Testing for unidimensionality based on the fitted Rasch model using the Martin-Löf test resulted in

```
> MLoef(fit.rm)

Martin-Loef-Test (split criterion: median)
LR-value: 109.281
Chi-square df: 24
p-value: 0
```

where, based on the default settings which uses the median as the split criterion, there is evidence that unidimensionality may not be tenable. Other tests for unidimensionality are available through NPtest

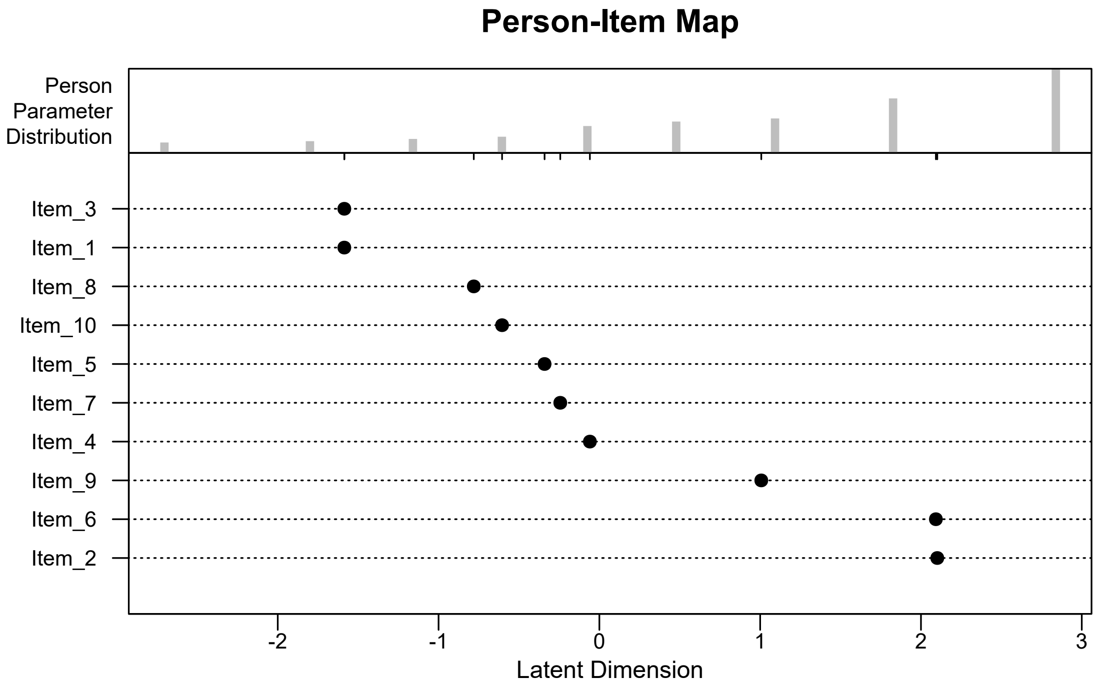

Figure 2
Person-Item map for Rasch model

function. Then, if we find evidence of unidimensionality then an investigation of the item fit to the model could be the next step. The item fit information is obtained through a two-step process. First, we compute the person parameters along will all the residuals. Then, the investigation of the item fit statistics is a straightforward extension. The item fit statistics (i.e., INFIT/OUTFIT t or MSQ) are obtained using

```
> pp <- person.parameters(fit.rm)
> itemfit(pp)

Itemfit Statistics:
          Chisq  df p-value Outfit MSQ Infit MSQ Outfit t Infit t
Item_1  318.101 491   1.000      0.647     0.848    -1.38   -1.30
Item_2  951.205 491   0.000      1.933     1.154     7.15    3.38
Item_3  342.213 491   1.000      0.696     0.789    -1.14   -1.87
Item_4  373.736 491   1.000      0.760     0.843    -2.12   -2.42
Item_5  304.879 491   1.000      0.620     0.838    -3.07   -2.29
Item_6  546.926 491   0.041      1.112     1.061     1.08    1.38
Item_7  390.352 491   1.000      0.793     0.870    -1.61   -1.87
Item_8  467.049 491   0.775      0.949     0.939    -0.23   -0.69
Item_9  430.829 491   0.976      0.876     0.885    -1.62   -2.25
Item_10 439.997 491   0.952      0.894     0.867    -0.61   -1.68
```

where, these results give additional evidence that these data, nd potentially the scale, do not fit the Rasch model. One source of misfit is likely due to item 2. The INFIT/OUTFIT statistics appear to be quite extreme, especially the OUTFIT t. What this indicates is that the residuals for item 2 are more variable than the Rasch model predicts the residuals should. The high values of OUTFIT for item 2 indicates that individuals with person measures high above or below the item measure are responding more erratically than expected under the Rasch model. A likely next step would be to go back to content of item 2 to try to identify whether there is a reason based on content for why individuals may be responding different than the model predicts. Other features of the eRm package utilize the person-parameters and we encourage exploring the options such as person fit statistics and residual plots. We have already discussed a few of the graphical capabilities, but various other graphical displays (e.g., the plotGOF command) can be created to help assess the fit of the data to the Rasch model. Next, we used the polytomous data to fit the partial credit model.

Applying the Partial Credit Model in eRm

The previous example with the Rasch model is somewhat artificial given that we dichotomized the four response categories. In practice we would likely not impose such a transformation on our data and instead analyzed the survey data as polytomous responses from the outset of the scale evaluation. Next, we show how the polytomous RSE data can be modeled and explored in the eRm package. Again, many of the functions and capabilities are straightforwardly extended from the Rasch to the polytomous models (partial credit model and rating scale model). Fitting the partial credit model proceeds by the same methods used in the Rasch model. The only difference is that we now use PCM instead. The function names in the package were well chosen to facilitate finding the needed tools more easily. The results from the fitted PCM are the following

```
> fit.pcm <- PCM(data)
> summary(fit.pcm)

Results of PCM estimation:

Call:  PCM(X = data)

Conditional log-likelihood: -4571.338
Number of iterations: 48
Number of parameters: 29

Item (Category) Difficulty Parameters (eta): with 0.95 CI:
          Estimate Std. Error lower CI upper CI
Item_1.c2   -1.691      0.260   -2.202   -1.181
Item_1.c3   -0.117      0.265   -0.637    0.403
Item_2.c1   -0.515      0.145   -0.799   -0.232
Item_2.c2    1.139      0.167    0.811    1.466
```

```
Item_2.c3      4.902       0.222     4.466     5.337
...
```

where the full output is shown in the appendix. The results are displayed nearly identically which eases use when fitting different models. Based on the four category response scale for this version of the Rosenberg Self-Esteem scale, we get three parameters per item corresponding to the three thresholds. By default, the first threshold of the first item is fixed to 0. A feature of this package is that the easiness parameters are reported for this polytomous model.

The ICCs for the partial credit model are each reported separately using the same plotICC function. Because the characteristic curves are plotted for each category, the joint plotting object is not available because such a figure would be very difficult to interpret. We selected the ICC for items one and four. Item 1 provides of an example an item with disordered categories which may indicate that item 1 needs to be revised. However, item 4 seems to be performing much better in comparison due to each response category being the most likely response for *some* range of the latent dimension. The ICCs provide a general idea of the fit of specific items to the PCM, but more fine grained information is obtained with the item fit statistics.

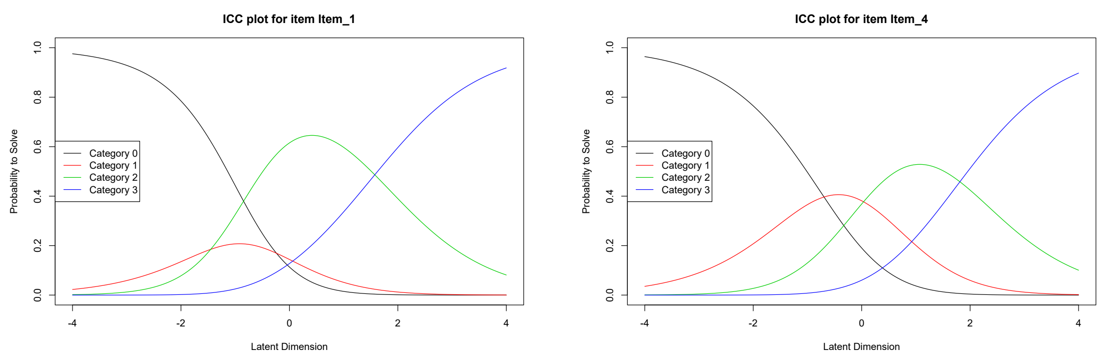

Figure 3
Item characteristic curves for the partial credit model

The item fit statistics indicated similar results compared to the dichotomous fit statistics. We did not observe the same magnitude of OUTFIT statistics for item 2. However, we still would recommend revisiting the content of item 2. The estimates of the item fit statistics for the PCM are

```
> pp <- person.parameters(fit.pcm)
> itemfit(pp)

Itemfit Statistics:
          Chisq  df p-value  Outfit MSQ  Infit MSQ  Outfit t  Infit t
Item_1  685.483 705   0.694       0.971      0.955     -0.35    -0.65
```

```
Item_2   811.456 705   0.003        1.149       1.116       2.86      2.23
Item_3   623.756 705   0.987        0.884       0.887      -1.70     -1.79
Item_4   642.962 705   0.954        0.911       0.843      -1.26     -2.87
Item_5   566.736 705   1.000        0.803       0.818      -3.77     -3.43
Item_6   766.279 705   0.054        1.085       1.076       1.66      1.49
Item_7   514.063 705   1.000        0.728       0.736      -3.07     -4.73
Item_8   698.060 705   0.567        0.989       0.956      -0.17     -0.75
Item_9   578.299 705   1.000        0.819       0.845      -3.23     -3.12
Item_10  588.841 705   0.999        0.834       0.850      -2.95     -2.69
```

where new items were identified as potentially misfitting. That is, item 5 and item 7 resulted in low OUTIFT and INFIT estimates. The additional information has made the evaluation of which items to retain and which to revise (or omit) has become more difficult. The item level information is useful, but next a more global investigation of the item person relationship is described.

Next, the person-item map for partial credit model is shown in Figure 4. Here we see that the threshold locations are represented along with the item location. This provides users with an efficient way to determine if any items have disordered thresholds, which is an indicator that a higher category is not frequently used. A nice feature is that all items with disordered categories are marked with a (*) on the right vertical axis. The map also provides an nice visual representation of how the person distribution relates to the items and thresholds. The identification of items with disordered thresholds leads us to a point in evaluating the scale where we have to make a choice on how to utilize this information. One option would be to do nothing and leave the item in the scale and acknowledge that these data may not fit the PCM. Another option would be to remove the item entirely, which would side step the issue and allow for the measurement of self-esteem to occur with less items. But, we don't recommend removing an item based solely on the disordered threshold because the interpretation of the construct may change by removing items. A potentially useful option is to collapse categories. Collapsing the *Strongly Disagree* and *Disagree* categories for item 2 may be more justifiable. We recommend the reader refer to (Smith & Smith, 2004)'s chapter on optimizing category effectiveness.

The fit (or misfit) of the items may be masked by potential issues of dimensionality. Testing dimensionality is a straightforward extension from the methods used previous for the Rasch model. The MLoef function also works for the testing the dimensionality of the PCM. The Martin-Löef test was not significant when the polytomous data were fit to the PCM,

```
> MLoef(fit.pcm)

Martin-Loef-Test (split criterion: median)
LR-value: 216.568
Chi-square df: 224
```

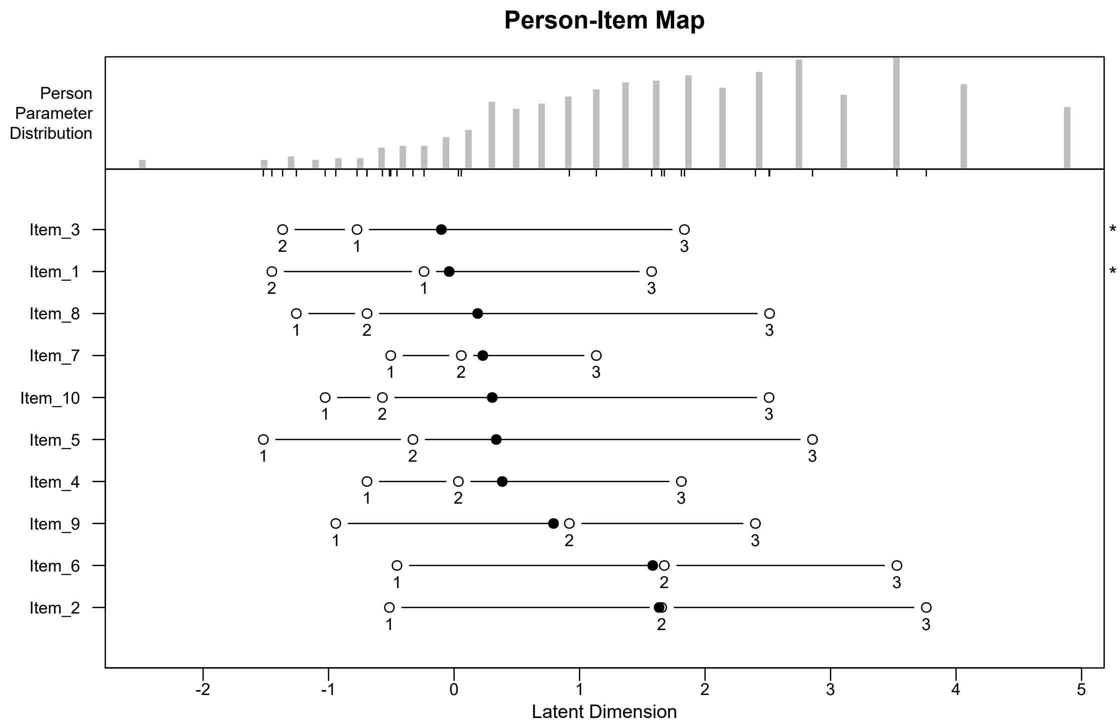

Figure 4
Person-Item map for the partial credit model

```
p-value: 0.627
```

where we conclude that there is evidence for unidimensionality.

The evaluation has lead to some conflicting and somewhat troubling information with regard to the fit of these data to the PCM. The disordered categories and the potential misfit of multiple items prompts next steps in the evaluation of the Rosenberg self-esteem scale in this population. We would take the information on which items may not be performing as expected under the PCM (e.g., item 2, 5 and 7) and review the content in these items. We may decide that we need to keep all items based on theoretical grounds that the items are necessary to retain the interpretation of self-esteem. The decision for how to the address the sources of misfit to the PCM should be done thoughtfully and with clear justification.

Closing Remarks

The eRm package provides a flexibly modeling environment for a variety of Rasch models. The flexibility provides users with a broad toolkit for modeling data from educational settings whether the data is correct-incorrect, Likert-type, or longitudinal (see Mair et al., 2013). Additionally, the use of conditional maximum likelihood (CML) fits into the Rasch framework where for whichever model is of interest, a researcher can have the advantage of consistent maximum likelihood estimates. This feature is in line with

Rasch models where total scores are a sufficient statistic for person measures. These features along with the array of tests and diagnostics tools make the eRm package a useful free tool for modeling data within the Rasch framework.

The package has been utilized through published literature in a variety of fields partly due to the flexibility, ease of use, and built into the R environment. An added benefit of being within the R environment is the advantage for data management and the utility of incorporating other R packages in a project. For example, additional plotting capabilities with ggplot2 (Wickham, 2016) or utilizing other analysis packages such as ltm (Rizopoulos, 2006) are possible, further adding to the attractiveness of the eRm package.

Future Developments

The use of the eRm package has prompted future developments to further increase the functionality. According to the developers (Mair et al., 2013), an implementation of mixed Rasch models (MIRA; von Davier & Rost, 1995) is in development. A feature that is currently not implemented, but may be of interest to researchers fitting survey data to Rasch models, is the use of sampling weights in the computation of the item parameters. Differentially weighting cases based on the sampling frame is common practice with large national datasets. The incorporation of sampling weights for the item parameter estimation may be out of scope for the eRm package, though. Researchers interested in such features may find the sirt package (Robitsch, 2019) of interest, also available in the R environment. However, for many practitioners and researchers, the flexibility in modeling in the eRm package is likely more than sufficient while providing many nice features and graphics to aid the modeling process.

References

Andersen, E. B. (1972). The Numerical Solution of a Set of Conditional Estimation Equations. *Journal of the Royal Statistical Society: Series B (Methodological)*, *34*(1), 42–54. https://doi.org/10.1111/j.2517-6161.1972.tb00887.x

Andersen, E. B. (1973). A goodness-of-fit test for the Rasch model. *Psychometrika*, *38*(1), 123–140.

Benson, J., & Nasser, F. (1998). On the Use of Factor Analysis as a Research Tool. *Journal of Vocational Education Research*, *23*(1), 13–33.

Brown, T. A. (2015). *Confirmatory factor analysis for applied research*. The Guilford Press.

Christensen, K. B., Bjorner, J. B., Kreiner, S., & Petersen, J. H. (2002). Testing unidimensionality in polytomous Rasch models. *Psychometrika*, *67*(4), 563–574. https://doi.org/10.1007/BF02295131

Conrad, I., Matschinger, H., Riedel-Heller, S., von Gottberg, C., & Kilian, R. (2014). The psychometric properties of the German version of the WHOQOL-OLD in the German population aged 60 and older. *Health and Quality of Life Outcomes*, *12*(1), 105. https://doi.org/10.1186/s12955-014-0105-4

de Ayala, R. J. (2009). *The theory and practice of item response theory*. New York, NY, Guilford Publications.

de Boeck, P., & Wilson, M. (2004). *Exploratory item response models: A generalized linear and nonlinear approach*. New York, NY, Springer Science and Business Media.

DiStefano, C., & Morgan, G. (2011). Examining Classification Criteria: A Comparison of Three Cut Score Methods. *Psychological Assessment*, *23*(2), 354–363. https://doi.org/10.1037/a0021745

Edelsbrunner, P. A., & Dablander, F. (2019). The Psychometric Modeling of Scientific Reasoning: a Review and Recommendations for Future Avenues. *Educational Psychology Review*, *31*(1), 1–34. https://doi.org/10.1007/s10648-018-9455-5

Elsman, E. B., Tadić, V., Peeters, C. F., Van Rens, G. H., Rahi, J. S., & Van Nispen, R. M. (2019). Cross-cultural validation of the Functional Vision Questionnaire for Children and Young People with visual impairment in the Dutch population: Challenges and opportunities. *BMC Medical Research Methodology*, *19*(221).

Haberman, S. J. (1977). Log-linear models and frequency tables with small expected cell counts. *The Annuals of Statistics*, *5*(6), 1148–1169.

Haberman, S. J. (2004). Joint and conditional maximum likelihood etimation for the Rasch model for binary responses. *ETS Research Report Series*, (1), i–63. https://doi.org/10.1002/j.2333-8504.2004.tb01947.x

Hatzinger, R., & Rusch, T. (2009). IRT Models with Relaxed Assumptions in eRm : A Manual-like Instruction. *Psychological Science Quarterly*, *51*(1), 87–120.

Kaplan, D. (2009). *Structural equation modeling: Foundations and extensions* (2nd). Upper Saddle River, NJ, Prentice-Hall.

Kline, R. B. (2015). *Principles and practice of structural equation modeling* (4th). New York, NY, The Guilford Press.

Krammer, G. (2019). The Andersen Likelihood Ratio Test with a Random Split Criterion Lacks Power. *Journal of Modern Applied Statistical Methods*, *17*(2). https://doi.org/10.22237/jmasm/1555594442

Linacre, J. M. (2019a). *Facets computer program for many-facets Rasch measurement, version 3.82.0*. Beaverton, Oregon, WINSTEPS.com.

Linacre, J. M. (2019b). *WINSTEPS Rasch measurement computer program*. Beaverton, Oregon, WINSTEPS.com.

Mair, P., & Hatzinger, R. (2007). Extended Rasch modeling: The eRm package for the application of IRT models in R. *Journal of Statistical Software*, *20*(9), 1–20.

Mair, P., Hatzinger, R., & Maier, M. J. (2013). *eRm: Extended Rasch Modeling* (tech. rep.).

Martin-Löf, P. (1970). On the notion of randomness. *Studies in Logic and the Foundations of Mathematics*, *60*(100), 73–78.

Masters, G. N. (1982). A rasch model for partial credit scoring. *Psychometrika*, *47*(2), 149–174. https://doi.org/10.1007/BF02296272

Morgan, G. B., DiStefano, C., & Motl, R. W. (2016). Examining Class Differences in Method Effects Related to Negative Wording: An Example using Rasch Mixture Modeling. *Journal of applied measurement*, *17*(4), 441–457.

R Core Team. (2019). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria.

Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests*. Copenhagen, Danish Institute for Educational Research.

Rizopoulos, D. (2006). ltm: An R package for latent variable modeling and item response theory analyses. *Journal of statistical software*, *17*(5), 1–25.

Robitsch, A. (2019). *Sirt: Supplementary item response theory models* [R package version 3.7-40]. R package version 3.7-40. https://CRAN.R-project.org/package=sirt

Rosenberg, M. (1989). *Society and the adolescent self image*. (Revised). Middletown, CT, Wesleyan University Press.

Schumacker, R. E., & Lomax, R. G. (2016). *A beginner's guide to structural equation modeling* (4th). New York, NY, Routledge.

Smith, E. V., & Smith, R. M. (2004). *Introduction to Rasch measurement: Theory, models and applications*. Maple Grove, MN, JAM Press.

Strobl, C., Kopf, J., & Zeileis, A. (2015). Rasch Trees: A New Method for Detecting Differential Item Functioning in the Rasch Model. *Psychometrika*, *80*(2), 289–316.

Tuerlinckx, F., & Wang, W. C. (2004). Models for polytomous data. In P. de Boeck & M. Wilson (Eds.), *Exploratory item response models* (pp. 75–109). New York, NY, Springer New York.

von Davier, M., & Rost, J. (1995). Polytomous Mixed Rasch Models. In *Rasch models* (pp. 371–379). New York, NY, Springer New York. https://doi.org/10.1007/978-1-4612-4230-7_20

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis*. Springer.

Wright, B. D., & Stone, M. H. (1979). *Best test design*. Chicago, MESA Press.

Appendix

Fitted partial credit model full results

```
> fit.pcm <- PCM(data)
> summary(fit.pcm)

Results of PCM estimation:

Call:  PCM(X = data)

Conditional log-likelihood: -4571.338
Number of iterations: 48
Number of parameters: 29

Item (Category) Difficulty Parameters (eta): with 0.95 CI:
            Estimate Std. Error lower CI upper CI
Item_1.c2     -1.691      0.260   -2.202   -1.181
Item_1.c3     -0.117      0.265   -0.637    0.403
Item_2.c1     -0.515      0.145   -0.799   -0.232
Item_2.c2      1.139      0.167    0.811    1.466
Item_2.c3      4.902      0.222    4.466    5.337
Item_3.c1     -0.773      0.340   -1.439   -0.108
Item_3.c2     -2.140      0.298   -2.724   -1.556
Item_3.c3     -0.304      0.298   -0.889    0.281
Item_4.c1     -0.694      0.216   -1.118   -0.271
Item_4.c2     -0.660      0.209   -1.070   -0.249
Item_4.c3      1.152      0.226    0.708    1.595
Item_5.c1     -1.520      0.295   -2.099   -0.941
Item_5.c2     -1.848      0.280   -2.397   -1.298
Item_5.c3      1.008      0.290    0.441    1.576
Item_6.c1     -0.455      0.143   -0.735   -0.175
Item_6.c2      1.220      0.166    0.895    1.545
Item_6.c3      4.748      0.218    4.320    5.177
Item_7.c1     -0.505      0.216   -0.929   -0.081
Item_7.c2     -0.447      0.209   -0.857   -0.038
Item_7.c3      0.686      0.221    0.253    1.120
Item_8.c1     -1.257      0.304   -1.853   -0.661
Item_8.c2     -1.950      0.283   -2.504   -1.396
Item_8.c3      0.564      0.289   -0.003    1.130
Item_9.c1     -0.943      0.186   -1.307   -0.578
Item_9.c2     -0.024      0.194   -0.405    0.356
Item_9.c3      2.377      0.221    1.943    2.811
Item_10.c1    -1.027      0.272   -1.560   -0.495
Item_10.c2    -1.598      0.253   -2.094   -1.102
Item_10.c3     0.912      0.265    0.393    1.431

Item Easiness Parameters (beta) with 0.95 CI:
                  Estimate Std. Error lower CI upper CI
beta Item_1.c1       0.239      0.312   -0.372    0.851
beta Item_1.c2       1.691      0.260    1.181    2.202
beta Item_1.c3       0.117      0.265   -0.403    0.637
beta Item_2.c1       0.515      0.145    0.232    0.799
```

beta	Item_2.c2	−1.139	0.167	−1.466	−0.811
beta	Item_2.c3	−4.902	0.222	−5.337	−4.466
beta	Item_3.c1	0.773	0.340	0.108	1.439
beta	Item_3.c2	2.140	0.298	1.556	2.724
beta	Item_3.c3	0.304	0.298	−0.281	0.889
beta	Item_4.c1	0.694	0.216	0.271	1.118
beta	Item_4.c2	0.660	0.209	0.249	1.070
beta	Item_4.c3	−1.152	0.226	−1.595	−0.708
beta	Item_5.c1	1.520	0.295	0.941	2.099
beta	Item_5.c2	1.848	0.280	1.298	2.397
beta	Item_5.c3	−1.008	0.290	−1.576	−0.441
beta	Item_6.c1	0.455	0.143	0.175	0.735
beta	Item_6.c2	−1.220	0.166	−1.545	−0.895
beta	Item_6.c3	−4.748	0.218	−5.177	−4.320
beta	Item_7.c1	0.505	0.216	0.081	0.929
beta	Item_7.c2	0.447	0.209	0.038	0.857
beta	Item_7.c3	−0.686	0.221	−1.120	−0.253
beta	Item_8.c1	1.257	0.304	0.661	1.853
beta	Item_8.c2	1.950	0.283	1.396	2.504
beta	Item_8.c3	−0.564	0.289	−1.130	0.003
beta	Item_9.c1	0.943	0.186	0.578	1.307
beta	Item_9.c2	0.024	0.194	−0.356	0.405
beta	Item_9.c3	−2.377	0.221	−2.811	−1.943
beta	Item_10.c1	1.027	0.272	0.495	1.560
beta	Item_10.c2	1.598	0.253	1.102	2.094
beta	Item_10.c3	−0.912	0.265	−1.431	−0.393