Rcdm Package Demo

Michel Philipp
University of Zurich

September 30, 2016

Contents

1 Installation

To install the package from the source code, Rcpp and RcppArmadillo are required. If you have not installed Rcpp and RcppArmadillo packages, run:

```
install.packages(c("Rcpp", "RcppArmadillo"))
```

Then, install the package either by the command:

```
install.packages("path/filename.tar.gz", repos = NULL, type = "source")
```

or if you are using RStudio go to "Packages - Install - Install from" and choose "Package Archive File" then Browse the File and klick Install. Alternatively your can install Rcdm directly from GitHub:

```
install.packages("devtools") # if you have not installed devools
devtools::install_github("mphili/cdm")
```

Rcdm currently depends on Rcpp, Matrix, limSolve and gtools packages, that need to be installed on your system.

```
install.packages(c("Rcpp", "Matrix", "limSolve", "gtools"))
```

2 Fitting a GDINA model

```
library("Rcdm")
```

Load your own data or use our data example from the pks package:

```
# install.packages("pks")
data("probability", package = "pks")

# reduce to 12 items from the first test
items <- sprintf("b1%.2i", 1:12)
resp <- probability[, items]
resp <- resp[complete.cases(resp),]

qmat <- t(read.table(header = FALSE, text = "
  0    1    0    0    1    1    0    0    0    1    1    0
  0    0    0    1    0    0    0    0    1    1    1    1
  1    0    0    0    1    1    1    1    1    0    1    1
  0    0    1    0    0    0    1    1    0    0    0    1
"))
```

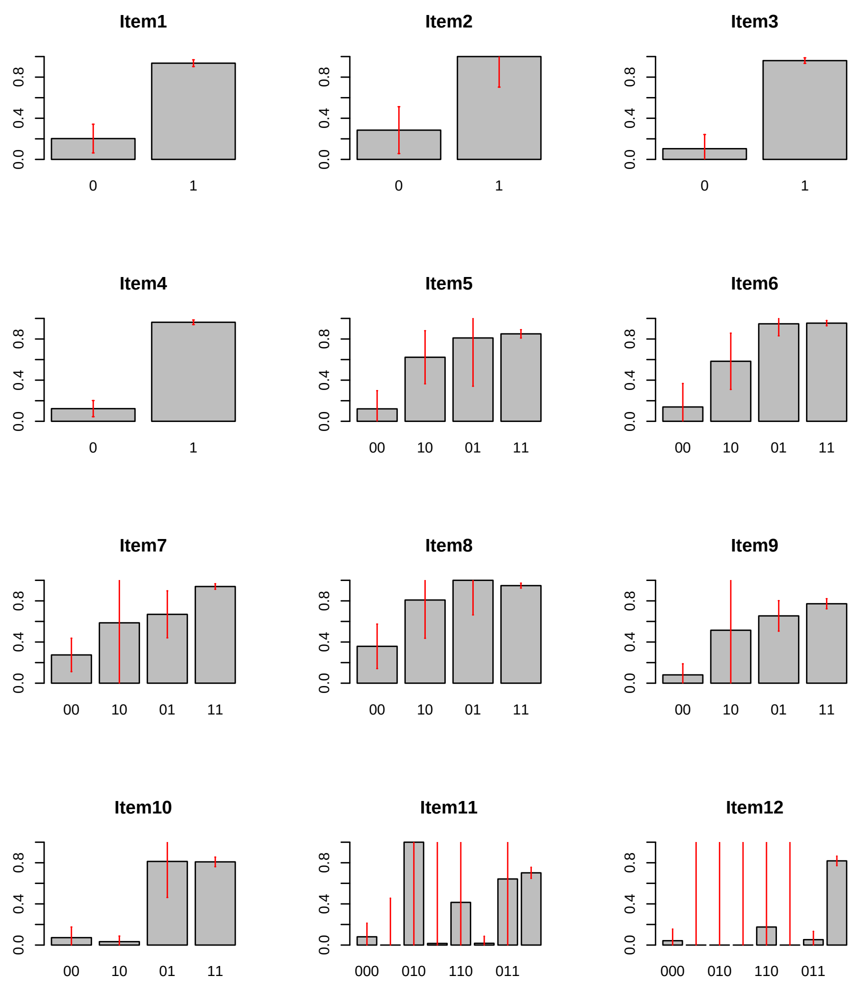
```r
colnames(qmat) <- c("cp", "id", "pb", "un")
rownames(qmat) <- colnames(resp)
```

Fit a GDINA model using the Rcdm package:

```r
mGDINA <- gdina(resp, qmat)
```

Plot the estimated conditional response probabilities:

```r
plot(mGDINA)
```



3 Analyzing the Result

3.1 Parameter Estimates

Extract the estimates of the item parameters:

```
mGDINA$dj

## $b101
##       d_0       d_1
## 0.2026237 0.7328790
##
## $b102
##       d_0       d_1
## 0.2845926 0.7154054
##
## $b103
##       d_0       d_1
## 0.1043282 0.8561985
##
## $b104
##       d_0       d_1
## 0.1231026 0.8395613
##
## $b105
##        d_0        d_1        d_2       d_12
##  0.1209243  0.5015454  0.6894077 -0.4617671
##
## $b106
##        d_0        d_1        d_2       d_12
##  0.1392815  0.4441071  0.8083905 -0.4376127
##
## $b107
##         d_0         d_1         d_2        d_12
##  0.27455117  0.31206864  0.39445421 -0.04133606
##
## $b108
##        d_0        d_1        d_2       d_12
##  0.3582082  0.4501938  0.6417898 -0.5016836
##
## $b109
##         d_0         d_1         d_2        d_12
##  0.08063802  0.43401073  0.57355589 -0.31613747
##
## $b110
##         d_0         d_1         d_2        d_12
```

```
##  0.07219741 -0.03921185  0.74069189  0.03489225
##
## $b111
##         d_0         d_1         d_2         d_3        d_12        d_13
##  0.08022087 -0.08021887  0.91977713 -0.06451234 -0.50500065  0.08168869
##        d_23       d_123
## -0.29302834  0.56309866
##
## $b112
##         d_0         d_1         d_2         d_3        d_12        d_13
##  0.04242436 -0.04242236 -0.04242236 -0.04242236  0.21772990  0.04242236
##        d_23       d_123
##  0.09498799  0.54806811
```

Extract the estimates of the latent class distribution

```
mGDINA$pa
```

```
##         0000         1000         0100         0010         0001
## 8.118021e-02 1.222217e-02 5.123528e-03 1.949008e-02 1.064277e-02
##         1100         1010         1001         0110         0101
## 9.999996e-07 9.999996e-07 4.669068e-02 9.999996e-07 1.821362e-03
##         0011         1110         1101         1011         0111
## 3.593686e-03 2.728925e-03 1.150701e-02 1.149323e-01 3.068450e-02
##         1111
## 6.593798e-01
```

Extract the estimates of the response probabilities:

```
mGDINA$pj
```

```
## $b101
##         0         1
## 0.2026237 0.9355027
##
## $b102
##         0         1
## 0.2845926 0.9999990
##
## $b103
##         0         1
## 0.1043282 0.9605268
##
## $b104
##         0         1
## 0.1231026 0.9626639
```

```
##
## $b105
##        00        10        01        11
## 0.1209243 0.6224697 0.8103320 0.8501103
##
## $b106
##        00        10        01        11
## 0.1392815 0.5833885 0.9476719 0.9541663
##
## $b107
##        00        10        01        11
## 0.2745512 0.5866198 0.6690054 0.9397380
##
## $b108
##        00        10        01        11
## 0.3582082 0.8084021 0.9999990 0.9485082
##
## $b109
##         00         10         01         11
## 0.08063802 0.51464876 0.65419391 0.77206717
##
## $b110
##         00         10         01         11
## 0.07219741 0.03298556 0.81288930 0.80856969
##
## $b111
##        000        100        010        001        110        101
## 0.08022087 0.00000100 0.99999900 0.01570852 0.41477848 0.01717834
##        011        111
## 0.64245732 0.70202515
##
## $b112
##        000        100        010        001        110        101
## 0.04242436 0.00000100 0.00000100 0.00000100 0.17530953 0.00000100
##        011        111
## 0.05256762 0.81836563
```

Get a nice table that contains the estimates of the item parameters:

```
coef(mGDINA)

##        item itemno   rule         est
## d_0    b101      1 G-DINA  0.20262371
## d_1    b101      1 G-DINA  0.73287898
## d_01   b102      2 G-DINA  0.28459258
## d_11   b102      2 G-DINA  0.71540542
```

```
## d_02   b103      3 G-DINA  0.10432823
## d_14   b103      3 G-DINA  0.85619854
## d_03   b104      4 G-DINA  0.12310264
## d_15   b104      4 G-DINA  0.83956128
## d_04   b105      5 G-DINA  0.12092427
## d_16   b105      5 G-DINA  0.50154544
## d_2    b105      5 G-DINA  0.68940769
## d_12   b105      5 G-DINA -0.46176715
## d_05   b106      6 G-DINA  0.13928146
## d_17   b106      6 G-DINA  0.44410707
## d_21   b106      6 G-DINA  0.80839046
## d_121  b106      6 G-DINA -0.43761267
## d_06   b107      7 G-DINA  0.27455117
## d_18   b107      7 G-DINA  0.31206864
## d_22   b107      7 G-DINA  0.39445421
## d_122  b107      7 G-DINA -0.04133606
## d_07   b108      8 G-DINA  0.35820825
## d_19   b108      8 G-DINA  0.45019381
## d_24   b108      8 G-DINA  0.64178975
## d_124  b108      8 G-DINA -0.50168364
## d_08   b109      9 G-DINA  0.08063802
## d_110  b109      9 G-DINA  0.43401073
## d_25   b109      9 G-DINA  0.57355589
## d_125  b109      9 G-DINA -0.31613747
## d_09   b110     10 G-DINA  0.07219741
## d_111  b110     10 G-DINA -0.03921185
## d_26   b110     10 G-DINA  0.74069189
## d_126  b110     10 G-DINA  0.03489225
## d_010  b111     11 G-DINA  0.08022087
## d_112  b111     11 G-DINA -0.08021887
## d_27   b111     11 G-DINA  0.91977713
## d_3    b111     11 G-DINA -0.06451234
## d_127  b111     11 G-DINA -0.50500065
## d_13   b111     11 G-DINA  0.08168869
## d_23   b111     11 G-DINA -0.29302834
## d_123  b111     11 G-DINA  0.56309866
## d_011  b112     12 G-DINA  0.04242436
## d_113  b112     12 G-DINA -0.04242236
## d_28   b112     12 G-DINA -0.04242236
## d_31   b112     12 G-DINA -0.04242236
## d_128  b112     12 G-DINA  0.21772990
## d_131  b112     12 G-DINA  0.04242236
## d_231  b112     12 G-DINA  0.09498799
## d_1231 b112     12 G-DINA  0.54806811
```

3.2 Standard Errors

Compute the (correct) variance covariance matrix of all model parameters:

```
v0 <- vcov(mGDINA)
```

Compute the (incorrect) variance covariance matrix by ignoring the skill parameters:

```
v1 <- vcov(mGDINA, type = "partial")
```

Compute the (incorrect) variance covariance matrix by item-wise computation of the information matrix:

```
v2 <- vcov(mGDINA, type = "itemwise")
```

3.2.1 Standard errors for the item parameters

```
cbind("full (correct)"       = sqrt(diag(v0))[seq(1, 2*nrow(qmat))],
      "partial (incorrect)"  = sqrt(diag(v1)),
      "itemwise (incorrect)" = sqrt(diag(v2)))

##       full (correct) partial (incorrect) itemwise (incorrect)
##  [1,]     0.07121171          0.06518312           0.05217388
##  [2,]     0.07279614          0.06718363           0.05464946
##  [3,]     0.11612098          0.08626799           0.06796940
##  [4,]     0.17182358          0.09082291           0.07200665
##  [5,]     0.07021230          0.05737784           0.04947524
##  [6,]     0.07127242          0.05879708           0.05108984
##  [7,]     0.04012112          0.03712668           0.03196255
##  [8,]     0.04161989          0.03877674           0.03398180
##  [9,]     0.09022412          0.08127162           0.05204203
## [10,]     0.15288442          0.14595823           0.11984529
## [11,]     0.24267117          0.17271489           0.10754958
## [12,]     0.30099310          0.21197154           0.15516515
## [13,]     0.11624850          0.07362337           0.05860907
## [14,]     0.18365387          0.15544856           0.12465630
## [15,]     0.13378764          0.09471415           0.08028801
## [16,]     0.19781948          0.16807885           0.13786396
## [17,]     0.08273508          0.07972691           0.06892772
## [18,]     0.38165644          0.23151316           0.19463657
## [19,]     0.14913728          0.14203399           0.12228028
## [20,]     0.40243539          0.26098010           0.22077888
## [21,]     0.11027818          0.10202863           0.07466451
## [22,]     0.23194382          0.21088713           0.17283776
```

```
## [23,]     0.21392272          0.13561825           0.10649104
## [24,]     0.31876822          0.23353762           0.18993072
## [25,]     0.07121171          0.05117352           0.03965984
## [26,]     0.07279614          0.33093551           0.21983244
## [27,]     0.11612098          0.09225305           0.08007924
## [28,]     0.17182358          0.34467225           0.23263022
## [29,]     0.07021230          0.04495428           0.03709611
## [30,]     0.07127242          0.05504762           0.04790852
## [31,]     0.04012112          0.17987668           0.11732563
## [32,]     0.04161989          0.18640738           0.12498224
## [33,]     0.09022412          0.05686949           0.04556542
## [34,]     0.15288442          0.16883727           0.14127159
## [35,]     0.24267117          1.17855729           0.58522110
## [36,]     0.30099310          0.17203164           0.13230816
## [37,]     0.11624850          1.34215412           0.73703531
## [38,]     0.18365387          0.26305703           0.20132970
## [39,]     0.13378764          1.18352783           0.59909999
## [40,]     0.19781948          1.34629442           0.75301474
## [41,]     0.08273508          0.04349507           0.03255495
## [42,]     0.38165644          1.28860960           0.66098386
## [43,]     0.14913728          0.47840854           0.37345415
## [44,]     0.40243539          0.27081987           0.21384399
## [45,]     0.11027818          1.81386396           1.28479053
## [46,]     0.23194382          1.51520043           0.93376039
## [47,]     0.21392272          0.55708288           0.43506161
## [48,]     0.31876822          1.98297735           1.44835048
```

3.2.2 Standard errors for the latent class parameters

```
sqrt(diag(v1))[-seq(1, 2*nrow(qmat))]

##  [1] 0.05117352 0.33093551 0.09225305 0.34467225 0.04495428 0.05504762
##  [7] 0.17987668 0.18640738 0.05686949 0.16883727 1.17855729 0.17203164
## [13] 1.34215412 0.26305703 1.18352783 1.34629442 0.04349507 1.28860960
## [19] 0.47840854 0.27081987 1.81386396 1.51520043 0.55708288 1.98297735
```

3.3 Confidence intervals

Confidence intervals for the intem parameters using the correct computation of the standard errors:

```
confint(mGDINA, alpha = 0.05)
```

```
##              itemno         lower       upper
## b101.d_0          1   0.063051317  0.34219611
## b101.d_1          1   0.590201164  0.87555680
## b102.d_0          2   0.056999643  0.51218552
## b102.d_1          2   0.378637393  1.05217344
## b103.d_0          3  -0.033285348  0.24194181
## b103.d_1          3   0.716507162  0.99588991
## b104.d_0          4   0.044466697  0.20173858
## b104.d_1          4   0.757987789  0.92113478
## b105.d_0          5  -0.055911749  0.29776029
## b105.d_1          5   0.201897496  0.80119339
## b105.d_2          5   0.213780936  1.16503445
## b105.d_12         5  -1.051702786  0.12816849
## b106.d_0          6  -0.088561417  0.36712434
## b106.d_1          6   0.084152099  0.80406205
## b106.d_2          6   0.546171496  1.07060942
## b106.d_12         6  -0.825331729 -0.04989362
## b107.d_0          7   0.112393391  0.43670894
## b107.d_1          7  -0.435964239  1.06010151
## b107.d_2          7   0.102150511  0.68675791
## b107.d_12         7  -0.830094927  0.74742281
## b108.d_0          8   0.142066992  0.57434950
## b108.d_1          8  -0.004407718  0.90479533
## b108.d_2          8   0.222508928  1.06107058
## b108.d_12         8  -1.126457865  0.12309059
## b109.d_0          9  -0.026767989  0.18804404
## b109.d_1          9  -0.361300942  1.22932240
## b109.d_2          9   0.378491204  0.76862057
## b109.d_12         9  -1.136093436  0.50381850
## b110.d_0         10  -0.030869172  0.17526400
## b110.d_1         10  -0.162227099  0.08380339
## b110.d_2         10   0.375569476  1.10581429
## b110.d_12        10  -0.342500140  0.41228464
## b111.d_0         11  -0.052037188  0.21247892
## b111.d_1         11  -0.555819291  0.39538156
## b111.d_2         11  -9.697870384 11.53742465
## b111.d_3         11  -3.266888533  3.13786385
## b111.d_12        11 -11.345357072 10.33535576
## b111.d_13        11  -3.225420402  3.38879778
## b111.d_23        11 -11.487323073 10.90126640
## b111.d_123       11 -10.845750188 11.97194751
## b112.d_0         12  -0.070306683  0.15515541
## b112.d_1         12  -4.311337857  4.22649313
## b112.d_2         12  -4.198409819  4.11356509
## b112.d_3         12  -1.256776158  1.17193143
```

```
## b112.d_12      12  -7.043770998  7.47923079
## b112.d_13      12  -6.107679782  6.19252451
## b112.d_23      12  -3.859605176  4.04958115
## b112.d_123     12  -7.982720942  9.07885715
```

3.4 Item-level fit

Item-level comparison of saturated and reduced models (de la Torre, 2013):

```
gdina_wald(mGDINA)
```

```
##             Kj   W value df    Pr(>W)
## b101  1.000000        NA NA        NA
## b102  1.000000        NA NA        NA
## b103  1.000000        NA NA        NA
## b104  1.000000        NA NA        NA
## b105  2.000000 14.529871  2 0.0006996 ***
## b106  2.000000 38.464606  2 4.441e-09 ***
## b107  2.000000  7.259821  2 0.0265186 *
## b108  2.000000  9.369927  2 0.0092331 **
## b109  2.000000 33.230430  2 6.083e-08 ***
## b110  2.000000 18.538968  2 9.426e-05 ***
## b111  3.000000  7.210987  6 0.3017754
## b112  3.000000  0.046081  6 0.9999980
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3.5 Fit indices

```
logLik(mGDINA)

## 'log Lik.' -2425.988 (df=63)

BIC(mGDINA)

## [1] 5243.999

AIC(mGDINA)

## [1] 4977.977
```

4 Fitting Reduced Models

Fit the reduced DINA model

```r
mDINA <- gdina(resp, qmat, rule = "DINA")
```

Fit the reduced DINO model

```r
mDINO <- gdina(resp, qmat, rule = "DINO")
```

Fit the reduced ACDM

```r
mACDM <- gdina(resp, qmat, rule = "ACDM")
```

4.1 Model comparisons

Compare (non-nested) models

```r
AIC(mDINA, mDINO, mACDM, mGDINA)

##        df      AIC
## mDINA  39 5035.778
## mDINO  39 5204.394
## mACDM  49 5000.156
## mGDINA 63 4977.977

BIC(mDINA, mDINO, mACDM, mGDINA)

##        df      BIC
## mDINA  39 5200.459
## mDINO  39 5369.075
## mACDM  49 5207.062
## mGDINA 63 5243.999
```

Compare (nested) models

```r
anova(mDINA, mACDM, mGDINA)

## Analysis of Variance Table
##
##    Npar  logLik    AIC    BIC Df Deviance  Pr(>Chi)
## m1   39 -2478.9 5035.8 5200.5
## m2   49 -2451.1 5000.2 5207.1 10   55.622 2.416e-08 ***
## m3   63 -2426.0 4978.0 5244.0 14   50.179 5.700e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

anova(mDINO, mACDM, mGDINA)
```

```
## Analysis of Variance Table
##
##    Npar  logLik    AIC    BIC Df Deviance Pr(>Chi)
## m1   39 -2563.2 5204.4 5369.1
## m2   49 -2451.1 5000.2 5207.1 10  224.239  < 2e-16 ***
## m3   63 -2426.0 4978.0 5244.0 14   50.179  5.7e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```