# Analysis of reporting behavior using the **hopit** package

Maciej J. Dańko[*1]

[1]*Max Planck Institute for Demographic Research*

## Abstract

Self-reported health, happiness, attitudes, and other statuses or perceptions are often the subject of biases that may come from different sources. For example, the evaluation of an individual's own health may depend on previous medical diagnoses, functional status, and symptoms and signs of illness; as on well as life-style behaviors, including contextual social, gender, age-specific, linguistic and other cultural factors (Jylha 2009; Oksuzyan et al. 2019). The **hopit** package offers versatile functions for analyzing different self-reported ordinal variables, and for helping to estimate their biases. Specifically, the package provides the function to fit a generalized ordered probit model that regresses original self-reported status measures on two sets of independent variables (King et al. 2004; Jürges 2007; Oksuzyan et al. 2019). The first set of variables (e.g., health variables) included in the regression are individual statuses and characteristics that are directly related to the self-reported variable. In the case of self-reported health, these could be chronic conditions, mobility level, difficulties with daily activities, performance on grip strength tests, anthropometric measures, and lifestyle behaviors. The second set of independent variables (threshold variables) is used to model cut-points between adjacent self-reported response categories as functions of individual characteristics, such as gender, age group, education, and country (Oksuzyan et al. 2019). The model helps to adjust for specific socio-demographic and cultural differences in how the continuous latent health is projected onto the ordinal self-rated measure. The fitted model can be used to calculate an individual predicted latent status variable, a latent index, and standardized latent coefficients; and makes it possible to reclassify a categorical status measure that has been adjusted for inter-individual differences in reporting behavior.

---

[*]Corresponding author: danko@demogr.mpg.de, maciej.danko@gmail.com

# 1 Introduction

**hopit** is an open source software library written in the R (R-Core-Team 2018) and C++ (Bates and Eddelbuettel 2013; Eddelbuettel and François 2011) programming languages. The **hopit** package provides versatile methods for fitting and analyzing ordered response data to adjust for heterogeneity in reporting behavior.

The ordered response data classify a measure of interest into ordered categories provided as a response option for a question asked during a survey. If, for example, the dependent variable is a happiness rating, then a respondent typically answers a question such as: "Taking all things together, would you say you are … ?"; and then selects from response options such as: "very happy", "pretty happy", "not too happy", and "very unhappy" (Liao, Fu, and Yi 2005). Similarly, if interviewees are asked to evaluate their general health (e.g., "Would you say your health is … ?"), they may choose among several categories, such as "very good", "good", "fair", "bad", and "very bad" (King et al. 2004; Jürges 2007; Rebelo and Pereira 2014; Oksuzyan et al. 2019). In political science, respondents may be asked for their opinions about recent legislation (e.g., "Rate your feelings about the proposed legislation."), and instructed to choose among categories like "strongly oppose", "mildly oppose", "indifferent", "mildly support", and "strongly support" (Greene and Hensher 2010). It is easy to imagine other multi-level ordinal variables that might be used in a survey, and to which the methodology described below could be applied.

In practice, it is assumed that when responding to a survey question about their general happiness, health, feelings, attitudes, or other status, participants are assessing the true value of this unobserved continuous variable, and project it onto the discrete scale provided. The thresholds that different individuals use to categorize their true status by selecting a specific response option may be affected by the reference group chosen, their earlier life experiences, and cross-cultural differences in the use of such scales. Thus, the responses of individuals may vary depending on their gender, age, cultural background, education, and personality traits; among other factors.

From the perspective of reporting behavior modeling, among the main tasks researchers face are computing this continuous estimate of the underlying, latent measure for each individual based on several specific characteristics of the responses considered, and accounting for variations in reporting across socio-demographic and cultural groups. More specifically, to build a latent, underlying measure, a generalized hierarchical ordered threshold model is fitted that regresses the reported status/attitude/feeling on two sets of independent variables (Boes and Winkelmann 2006; Greene et al. 2014). When the dependent reported ordered variable is self-rated health status, then the first set of variables - i.e., health variables - assess specific aspects of individuals' health, such as measures of chronic conditions, mobility, difficulties with a range of daily activities, grip strength, anthropometric characteristics, and lifestyle behaviors. Using the second set of independent variables (threshold variables), the model also adjusts for differences across socio-demographic and cultural groups. It allows the threshold between adjacent response categories to differ among individuals of different ages, genders, educational levels, and cultural backgrounds. (King et al. 2004; Jürges 2007; Oksuzyan et al. 2019; but see Rebelo and Pereira 2014).

Once the model is fitted, its estimates (latent measure and threshold coefficients) can be used to calculate the differences in reporting behavior among groups of people with different contextual characteristics through the calculation of differences between the expected and the reported ordinal response measures (Jürges 2007; Oksuzyan et al. 2019).

## 2 Methods

**Table 1.** Glossary.

| Term | Symbol | Definition | Exemplary case specific synonyms |
|---|---|---|---|
| Categorical response | $y$ | Dependent variable obtained during the survey | Self-rated health, self-rated happiness |
| Latent measure | $h$ | Modeled continuous latent measure of the investigated response variable | Latent health, latent happiness |
| Latent index | $H$ | Standardized latent measure | Health index, happiness index |
| Latent variables | — | Variables used to model the latent measure | Health variables, happiness variables |
| Latent terms | $X$ | Terms of the design matrix used to model the latent measure | |
| Latent coefficients | $\beta$ | Coefficients corresponding to each latent term | |
| Standardized coefficient | $D$ | Standardized value of a coefficient | Disability weights |
| Thresholds | $\alpha$ | Thresholds used to group the latent measure | Cut-points |
| Threshold variables | — | Variables used to model the thresholds | Socio-demographic, cultural, contextual variables |
| Threshold terms | $Y$ | Terms of the design matrix used to model the latent measure | |
| Threshold coefficients | $\gamma, \lambda$ | Coefficients corresponding to each threshold term | |

### 2.1 Generalized (hierarchical) ordered threshold model

Ordered threshold models are used to fit ordered categorical dependent variables. The generalized ordered threshold models (Ierza 1985; Boes and Winkelmann 2006; Greene et al. 2014) are an extension of ordered threshold models (McKelvey and Zavoina 1975). Whereas in the latter models the thresholds are constant, in the generalized models the thresholds are allowed to be dependent on covariates. Greene and Hensher (2010) and Greene et al. (2014) pointed out that for a model to make sense, the thresholds must also be ordered. This observation motivated Greene and coauthors to call these models *HOPIT*, which stands for hierarchical ordered probit models.

3

In the self-rated health example, the response variable is self-rated health, and the latent measure $h_i$ can depend on different health conditions and diseases (health variables). These variables are modeled with the parallel regression assumption. According to this assumption, the coefficients that describe the relationship between the lowest response category and all of the higher response categories are the same as the coefficients that describe the relationship between another (e.g., adjacent) lowest response category and the remaining higher response categories. In the considered case, $h_i$ is modeled as a linear function of the design matrix of health variables $X$ and its corresponding coefficients $\beta$:

$$h_i = \sum_{k=1}^{K} \beta_k X_{i,k} = X'\beta \tag{1}$$

where index $i \in 1...N$ is a number of cases (e.g., respondents), $X$ is in the form of a design matrix, and $K$ is the number of columns in $X$. As described above, the categorization (response mechanism) of the latent measure $h_i$ is modeled in terms of thresholds $\alpha_{i,j}$, while assuming that the lower order thresholds are never greater than the higher order thresholds (hierarchical assumption):

$$\begin{cases} y_i = 1 & \Leftrightarrow & \alpha_{i,0} \leq h_i < \alpha_{i,1} \\ y_i = 2 & \Leftrightarrow & \alpha_{i,1} \leq h_i < \alpha_{i,2} \\ \ldots \\ y_i = j & \Leftrightarrow & \alpha_{i,j-1} \leq h_i < \alpha_{i,j} \\ \ldots \\ y_i = J & \Leftrightarrow & \alpha_{i,J-1} \leq h_i < \alpha_{i,J} \end{cases} \tag{2}$$

The thresholds (cut-points, $\alpha$) are modeled using threshold variables coded as design matrix $Y$, their coefficients $\gamma$, and intercepts $\lambda$. These variables model the contextual characteristics of the respondent (e.g., country, gender, and age). As the threshold variables are modeled without applying the parallel regression assumption, each threshold is modeled by a variable separately (Boes and Winkelmann 2006; Greene et al. 2014).

Different parametrizations of thresholds exist (Greene et al. 2014; Rebelo and Pereira 2014; Jürges 2007). In the package, the parametrization of King et al. (2004) and Jürges (2007) is used, which assumes that:

$$\alpha_{i,\ j} = \begin{cases} -\infty & for\ j = 0 \\ \lambda_1 + \sum_{m=1}^{M} \gamma_{1,m} Y_{i,m} & for\ j = 1 \\ \alpha_{i,\ j-1} + exp(\lambda_j + \sum_{m=1}^{M} \gamma_{j,m} Y_{i,m}) & for\ J - 1 \geq j \geq 2 \\ \infty & for\ j = J \end{cases} \tag{3}$$

The condition $y_i = j \Leftrightarrow \alpha_{j-1,i} \leq h_i < \alpha_{j,i}$ can be easily expressed in terms of the probability, which leads to:

$$P(y_i = j) = P(\alpha_{j-1,i} \leq h_i < \alpha_{j,i}), \tag{4}$$

hence

$$P(y_i = j) = \Phi(\alpha_{i,\ j} - h_i) - \Phi(\alpha_{i,\ j-1} - h_i), \tag{5}$$

where $\Phi$ is a distribution function (cdf, cumulative density function). For example, for probit regression it is standard normal cdf $\Phi(x) = \frac{1}{2} + \frac{1}{2} * erf\left(\frac{x}{\sqrt{2}}\right)$, whereas for logit regression it takes the form $\Phi(x) = \frac{1}{1+e^{-x}}$. In reporting behavior analyses the probit model is typically chosen. This model simply assumes that $h_i$ is affected by a random noise $\epsilon_i$ having standard normal distribution $\epsilon_i \sim \mathcal{N}(0, 1)$.

Using all of the definitions presented above, the log likelihood function can be constructed as:

$$\ln L = \sum_{i=1}^{N} \sum_{j=1}^{J} z_{i,\,j} \ln \left[ \Phi(\alpha_{i,\,j} - h_i) - \Phi(\alpha_{i,\,j-1} - h_i) \right], \tag{6}$$

where $z_{i,j}$ is an indicator function defined as:

$$z_{i,\,j} = \begin{cases} 0 & for\ y_i = j \\ 1 & for\ y_i \neq j \end{cases} \tag{7}$$

## 2.2 Analysis of reporting heterogeneity

The model estimates are used to determine reporting behavior; i.e., how the continuous latent measure is projected onto the categorical response. In practice, this is done by comparing actual categorical ordered responses with theoretical responses that are adjusted for heterogeneity in reporting behaviors, and are, therefore, more comparable across individuals.

One of the first steps of the analysis is the standardization of the latent measure to obtain the latent index $H_i$.

$$H_i = 1 - \frac{h_i - \min\limits_{i} h_i}{\max\limits_{i} h_i - \min\limits_{i} h_i} \tag{8}$$

In the self-rated health example, $H_i$ is a proxy for the true underlying health of an individual, and varies from 0, representing the (model-based) worst health state in the sample, to 1, representing the (model-based) best health state in the sample.

The predicted latent measure $h_i$ obtained from the model is also used to standardize the latent variable coefficients. In the self-rated health example, the standardized coefficients are called disability weights $D_k$ (Jürges 2007; Oksuzyan et al. 2019) and are calculated for each health variable to provide information about the impact of a specific health measure on the latent index $H_i$. The disability weight for a health variable is equal to the ratio of the corresponding health coefficient and the difference between the lowest and the highest values of predicted latent health. In other words, the disability weight reduces $H_i$ by some given amount or percentage (i.e., the $H_i$ of each individual is reduced by the same amount if the person had a heart attack or other heart problems)(Jürges 2007; Oksuzyan et al. 2019).

$$D_k = \frac{\beta_k}{\max\limits_{i} h_i - \min\limits_{i} h_i} \tag{9}$$

While the latent index $H_i$ is intended to reflect the underlying health, happiness, or other status across individuals, the standardized coefficients $D_k$ (e.g., disability weights in the case of self-rated

health) are computed for an average individual in the study population. The relationship between $H_i$ and $D_k$ follows the equation:

$$H_i = C - \sum_{k=1}^{K} D_k X_{i,k}, \quad \text{where } C = \frac{\max_i h_i}{\max_i h_i - \min_i h_i} \tag{10}$$

Reporting behavior analysis is based on the reclassification of individuals into new response categories. There are two methods of reclassification: (1) the Jürges (2007) percentile method (see also Rebelo and Pereira 2014; Oksuzyan et al. 2019) and (2) the direct reclassification method based on model-estimated thresholds.

In the first method, the classification is based on the calculated latent index $H_i$, which is adjusted for inter-individual differences in reporting behavior. This method is based on the original distribution of the categorical response variable (see also 4.2). First, for each category $j$, an empirical distribution function is constructed:

$$\hat{F}(j) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}_{y_i} \leq j \tag{11}$$

where $\mathbf{1}$ is an indicator function taking 1 if the condition is true, or is 0 otherwise. The calculated cumulative frequencies of the latent index $H_i$ are used as percentiles (cut-points) so that each individual $i$ can be reclassified into a new response category.

In the second method the reclassification is based on eq. (2), so that each individual has her/his own model-derived cut-points.

## 3 Fitting the model to the exemplary health survey data using the hopit package

### 3.1 Installing and loading the package

To install and load the package simply run the following code:

```
install.packages("hopit")

library(hopit)
```

### 3.2 Exemplary data

In the examples presented below I use `healthsurvey`, which is a completely artificial data set that is simulated using the distributions of some major health and socio-demographic characteristics. The distributions and the data structure are roughly based on the WAVE1 SHARE database (DOIs: 10.6103/SHARE.w1.600); see Börsch-Supan et al. (2013) for technical details about the SHARE database. The structure of `healthsurvey` for the first six individuals (IDs) is presented below.

```r
print(t(healthsurvey[1:6,-1]), quote=FALSE, na.print='NA', right=TRUE)
```

```
                                 1        2       3        4         5        6
health                   Very good     Good    Good     Good Excellent     Good
diabetes                        no       no     yes       no        no       no
obese                           no       no      no       no        no       no
IADL_problems                   no       no      no       no        no       no
hypertension                    no      yes      no       no        no      yes
high_cholesterol                no      yes      no       no        no      yes
respiratory_problems            no       no      no       no        no      yes
heart_attack_or_stroke          no      yes      no       no        no       no
poor_mobility                   no       no      no       no        no      yes
very_poor_grip                  no       no      no       no        no       no
depression                      no       no      no      yes        no       no
other_diseases                 yes      yes      no       no        no      yes
sex                            man      man     man      man     woman      man
ageclass                       80+    70-79   50-59    60-69       80+      80+
education                    prim-    prim-   prim-     sec+     prim-    prim-
country                          Y        Y       X        Y         Z        Y
csw                        2407.48  1198.12  885.26   772.04   1304.24   917.16
psu                             YB       YB      XC       YA        ZB       YD
```

The first variable on the list (`health`) is the categorical self-reported health status. This variable is followed by 11 determinants of health, which include information on the presence of chronic diseases and other health conditions. The `sex`, `ageclass`, `education`, and `country` variables describe the contextual characteristics of individuals. The last group of variables (`csw`, `psu`, and `ssu`) describes the survey design.

## 3.3  Fitting a basic model

The generalized ordered probit models can be fitted using the `hopit` function. The function takes two kinds of formulas: (1) `latent.formula`, which models the impact of the latent variables on categorical health; and (2) `thresh.formula`, which models the thresholds.
One of the crucial steps needed for the proper interpretation of the model results is determining the order of the dependent variable, which must be a `factor`.

```r
levels(healthsurvey$health)
```

```
[1] "Excellent" "Very good" "Good"      "Fair"      "Poor"
```

To fit the model, the `hopit` function is called. Because the order is decreasing (from the best to the worst health state), the `hopit` parameter `decreasing.levels` is set to TRUE.

```
model1 <- hopit(latent.formula = health ~ hypertension +
                              high_cholesterol +
                              heart_attack_or_stroke +
                              poor_mobility +
                              very_poor_grip +
                              depression +
                              respiratory_problems +
                              IADL_problems +
                              obese +
                              diabetes +
                              other_diseases,
                thresh.formula = ~ sex + ageclass,
                decreasing.levels = TRUE,
                data = healthsurvey)
```

The summary of the fitted model can be created using a standard generic function.

```
summary(model1)
```

```
Formula (latent variables):
health ~ hypertension + high_cholesterol + heart_attack_or_stroke +
    poor_mobility + very_poor_grip + depression + respiratory_problems +
    IADL_problems + obese + diabetes + other_diseases
Formula (threshold variables): ~sex + ageclass
Link: probit
Number of cases: 10000
Response levels: Excellent, Very good, Good, Fair, Poor

Robust SE were used (sandwich estimator of the variance-covariance matrix).
```

|                            | Estimate | Std. Error | z value | Pr(>\|z\|) |     |
| -------------------------- | -------- | ---------- | ------- | --------- | --- |
| hypertensionyes            | 0.19232  | 0.02478    | 7.76    | 8.4e-15   | *** |
| high_cholesterolyes        | 0.09780  | 0.02918    | 3.35    | 0.00080   | *** |
| heart_attack_or_strokeyes  | 0.34401  | 0.03183    | 10.81   | < 2e-16   | *** |
| poor_mobilityyes           | 0.72832  | 0.03564    | 20.44   | < 2e-16   | *** |
| very_poor_gripyes          | 0.49720  | 0.12299    | 4.04    | 5.3e-05   | *** |
| depressionyes              | 0.25323  | 0.02390    | 10.59   | < 2e-16   | *** |
| respiratory_problemsyes    | 0.36777  | 0.03337    | 11.02   | < 2e-16   | *** |
| IADL_problemsyes           | 0.61579  | 0.03637    | 16.93   | < 2e-16   | *** |
| obeseyes                   | 0.18991  | 0.03295    | 5.76    | 8.3e-09   | *** |
| diabetesyes                | 0.33726  | 0.04010    | 8.41    | < 2e-16   | *** |
| other_diseasesyes          | 0.33533  | 0.02370    | 14.15   | < 2e-16   | *** |
| (L).1\|2                   | -0.09248 | 0.03194    | -2.90   | 0.00379   | **  |
| (L).2\|3                   | -0.26826 | 0.03236    | -8.29   | < 2e-16   | *** |
| (L).3\|4                   | 0.07514  | 0.02905    | 2.59    | 0.00968   | **  |
| (L).4\|5                   | -0.20346 | 0.05222    | -3.90   | 9.8e-05   | *** |
| (G).sexwoman.1\|2          | 0.02373  | 0.03015    | 0.79    | 0.43111   |     |

```
(G).sexwoman.2|3            0.01366    0.03460    0.39  0.69304
(G).sexwoman.3|4            0.03661    0.02869    1.28  0.20193
(G).sexwoman.4|5            0.11847    0.05039    2.35  0.01872 *
(G).ageclass60-69.1|2      -0.01835    0.03383   -0.54  0.58763
(G).ageclass60-69.2|3       0.05336    0.04068    1.31  0.18963
(G).ageclass60-69.3|4       0.06003    0.03616    1.66  0.09693 .
(G).ageclass60-69.4|5       0.16842    0.06492    2.59  0.00949 **
(G).ageclass70-79.1|2      -0.32157    0.04391   -7.32  2.4e-13 ***
(G).ageclass70-79.2|3       0.17131    0.04774    3.59  0.00033 ***
(G).ageclass70-79.3|4       0.19360    0.03777    5.13  3.0e-07 ***
(G).ageclass70-79.4|5       0.23235    0.06653    3.49  0.00048 ***
(G).ageclass80+.1|2        -0.33134    0.07274   -4.56  5.2e-06 ***
(G).ageclass80+.2|3         0.14976    0.07590    1.97  0.04848 *
(G).ageclass80+.3|4         0.17851    0.05025    3.55  0.00038 ***
(G).ageclass80+.4|5         0.22379    0.07674    2.92  0.00354 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Sigma: 1
Log-likelihood: -12945.98
Deviance: 25891.96
AIC: 25953.96
```

`model1` contains 11 dichotomous health variables and two threshold variables. The fitted coefficient includes $\beta$ (`latent.params`, first 11 coefficients in the summary), $\lambda$ (`thresh.lambda`, threshold intercepts, "(L)" prefix in the summary), and $\gamma$ (`thresh.gamma`, parameters related to threshold covariates, "(G)" prefix in the summary). The model coefficients can be accessed directly by the generic `coef(model1, aslist = TRUE)` function, which has an option to group parameters into a list (`aslist = TRUE`); or simply by calling `model1$coef.ls`.

### 3.4   Model selection and latent-threshold interactions

`model1` can be further extended by adding the country of origin to the threshold formula to control for cultural and linguistic differences.

```
model2 <- hopit(latent.formula = health ~ hypertension +
                high_cholesterol +
                heart_attack_or_stroke +
                poor_mobility +
                very_poor_grip +
                depression +
                respiratory_problems +
                IADL_problems +
                obese +
                diabetes +
                other_diseases,
         thresh.formula = ~ sex + ageclass + country,
```

```
                decreasing.levels = TRUE,
                data = healthsurvey)
```

The fit of the two models can be compared using the **AIC** function:

```
AIC(model2, model1)
```

```
   model2    model1
 25154.19 25953.96
```

or using the Likelihood Ratio Test (LRT), because the models are nested:

```
print(anova(model2, model1), short = TRUE)
```

```
    Likelihood ratio test:
      Chi^2 df Pr(>Chi^2)
     815.78  8     <2e-16 ***
     ---
    Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In the code above, the option `short = TRUE` is used to generate a shortened summary of the test. The results of LRT show that `model2` is more parsimonious than `model1`.

While both `latent.formula` and `thresh.formula` allow the user to specify the interactions within the formulas, the interactions between latent and threshold variables can also be modeled. Depending on how an interaction is interpreted, it can be added to either the latent or the threshold formula:

```
model3 <- hopit(latent.formula = health ~ hypertension +
                   high_cholesterol +
                   heart_attack_or_stroke +
                   poor_mobility +
                   very_poor_grip +
                   depression +
                   respiratory_problems +
                   IADL_problems +
                   obese +
                   diabetes +
                   other_diseases +
                   sex : respiratory_problems,
               thresh.formula = ~ sex +
                   ageclass +
                   country +
                   sex : depression,
               decreasing.levels = TRUE,
               data = healthsurvey)
```

There are two additional interactions `sex : respiratory_problems` and `sex : depression` in `model3`. The first interaction has an impact on the latent health variable; i.e., `respiratory_problems` has a gender-specific effect on this variable. In other words, respiratory problems (if present) affect the health of men and women differently. The second interaction modulates the effect of the `gender` variable on the thresholds if depression is present; i.e., women with depression have different thresholds than women without depression.

## 3.5 Including a survey design

The `hopit` function also has an option to include a survey design using the **survey** package (Lumley 2004, 2019). The example below fits a model using a simple two-level cluster sampling design.

```r
design <- svydesign(ids = ~ country + psu, weights = healthsurvey$csw,
                    data = healthsurvey)

model2s <- hopit(latent.formula = health ~ hypertension +
                    high_cholesterol +
                    heart_attack_or_stroke +
                    poor_mobility +
                    very_poor_grip +
                    depression +
                    respiratory_problems +
                    IADL_problems +
                    obese +
                    diabetes +
                    other_diseases,
                 thresh.formula = ~ sex + ageclass + country,
                 decreasing.levels = TRUE,
                 design = design,
                 data = healthsurvey)
```

Generally, ignoring the survey design can lead to biased results. However, in the example presented here, the design is of little importance. This becomes clear when comparing the coefficients of the latent variable for the two models:

```r
round(cbind('No survey design'  = model2$coef.ls$latent.par,
            'Has survey design' = model2s$coef.ls$latent.par),4)
```

|                          | No survey design | Has survey design |
|--------------------------|------------------|-------------------|
| hypertensionyes          | 0.1848           | 0.1878            |
| high_cholesterolyes      | 0.0897           | 0.0937            |
| heart_attack_or_strokeyes| 0.3466           | 0.3468            |
| poor_mobilityyes         | 0.7035           | 0.7060            |
| very_poor_gripyes        | 0.5142           | 0.5477            |
| depressionyes            | 0.2500           | 0.2492            |
| respiratory_problemsyes  | 0.3786           | 0.3798            |

```
IADL_problemsyes                           0.5926              0.6100
obeseyes                                   0.1904              0.1887
diabetesyes                                0.3284              0.3233
other_diseasesyes                          0.3294              0.3288
```

## 3.6   Checking the convergence of the fitted model

The accuracy of the model fit can be assessed using the `profile` function, which calculates and
plots the profile of the log likelihood function around fitted coefficient values.

```
profile(model2)
```
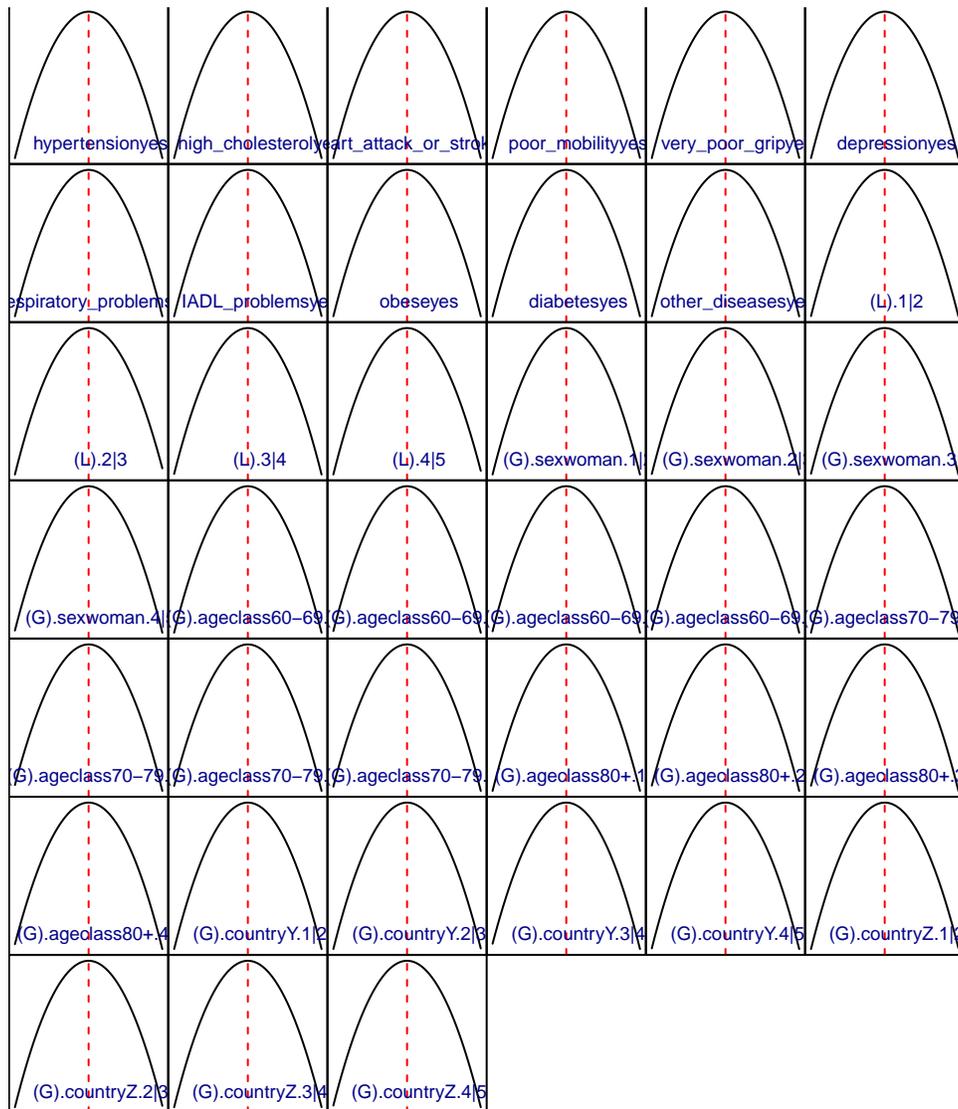


Figure 1: Log likelihood profile of the fitted model

It then becomes clear that the model converged to the maximum log likelihood (Fig. 1).

# 4 Analysis of reporting heterogeneity for the health data example

## 4.1 Disability weights and latent index

The fitted model can be further used to analyze reporting heterogeneity. Let us ignore latent-threshold interactions for the sake of simplicity, and consider `model2` for further analyses. The latent health variables can be directly accessed using `model2$coef.ls$latent.params`. We can, however, standardize these coefficients using Jürges' approach (Jürges 2007) in order to obtain so called disability weights. The disability weights are the same for each individual in the modeled population. The standardization is done using the `standardizeCoef` function.

```
txtfun <- function(x) gsub('_',' ',substr(x, 1, nchar(x)-3))
sc <- standardizeCoef(model1, namesf = txtfun)
```

The `namesf` argument is a function or a character vector that is used to rename the coefficients. The exact values of disability weights are stored in `sc`:

```
print(sc)
```

```
                            Std. coef
hypertension                0.0536
high cholesterol            0.0272
heart attack or stroke      0.0958
poor mobility               0.2028
very poor grip              0.1385
depression                  0.0705
respiratory problems        0.1024
IADL problems               0.1715
obese                       0.0529
diabetes                    0.0939
other diseases              0.0934
```

and can be summarized and plotted by the standard generic functions.

```
print(summary(sc), show.coef.names=FALSE)
```

```
                            Standardized coeficient Pr(>|z|)
hypertension                               0.0536   0.0000 ***
high cholesterol                           0.0272   0.0008 ***
heart attack or stroke                     0.0958   0.0000 ***
poor mobility                              0.2028   0.0000 ***
very poor grip                             0.1385   0.0001 ***
depression                                 0.0705   0.0000 ***
respiratory problems                       0.1024   0.0000 ***
IADL problems                              0.1715   0.0000 ***
obese                                      0.0529   0.0000 ***
```

```
diabetes                                        0.0939    0.0000 ***
other diseases                                  0.0934    0.0000 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(sc)
```
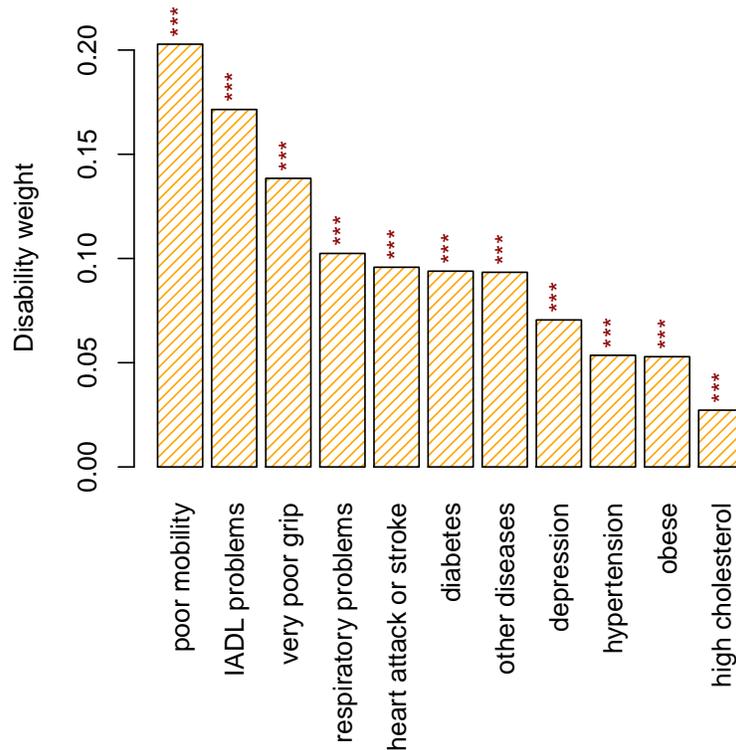


Figure 2: Disability weights

The standardized health status for each individual is called the health index (or the latent index). It is simply calculated using the `latentIndex` function.

```
hi <- latentIndex(model2)
```

The generic functions `print`, `summary`, and `plot` also apply here.

```
summary(hi)
```

```
  (0,0.05] (0.05,0.1] (0.1,0.15] (0.15,0.2] (0.2,0.25] (0.25,0.3]
         0          8          8         22         50         71
(0.3,0.35] (0.35,0.4] (0.4,0.45] (0.45,0.5] (0.5,0.55] (0.55,0.6]
        84        162        215        303        310        221
(0.6,0.65] (0.65,0.7] (0.7,0.75] (0.75,0.8] (0.8,0.85] (0.85,0.9]
       413        407        779        715       1410       1140
(0.9,0.95]   (0.95,1]
      2567       1113
```

```
plot(hi, response = "data", ylab = 'Health index', col='deepskyblue3')
```
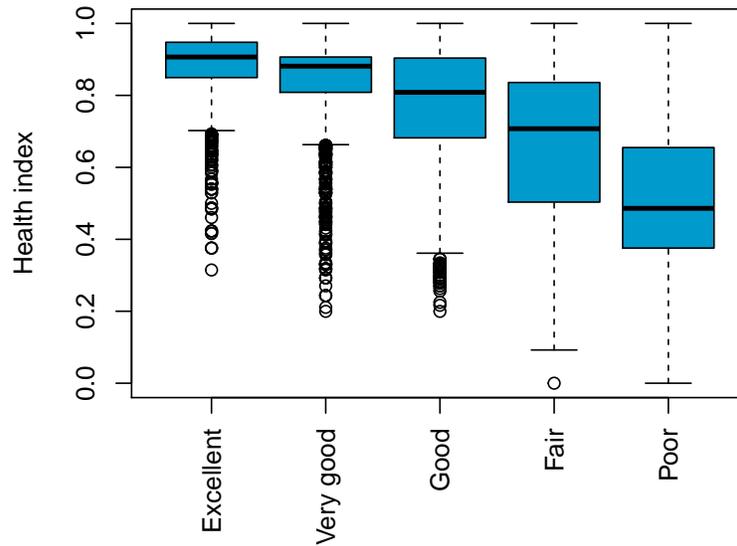


Figure 3: Health index vs. self-reported health

The boxplot above (Fig. 3) shows the reported health status versus the health index. It is also possible to plot the expected categorical health status on the Y axis, calculated according to eq. (2), by setting `response = "fitted"`.

## 4.2 Cut-points and adjusted health levels

The main aim of the reporting heterogeneity analyses is to determine the cut-points used to calculate the adjusted health status for each individual. The calculation of cut-points is realized using the `getCutPoints` function.

```r
z <- getCutPoints(model = model2)
plot(z)
```
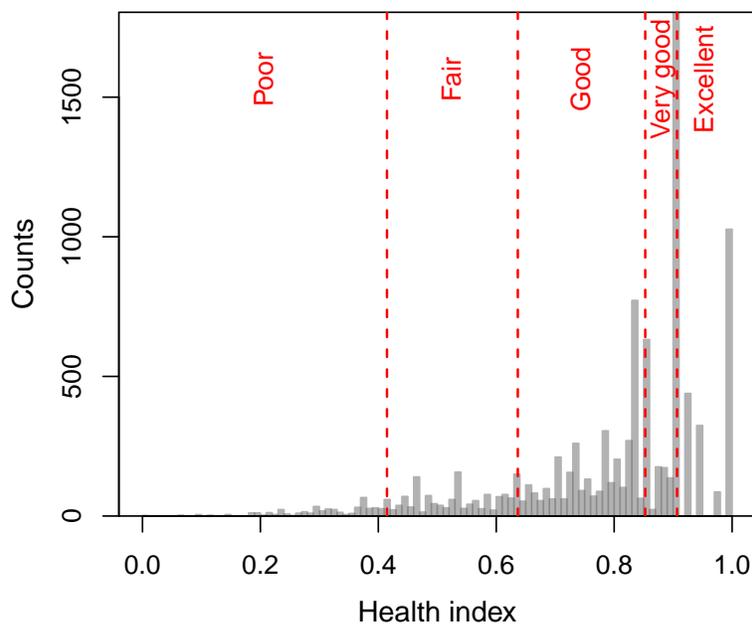


Figure 4: Health index cut-points for adjusted health levels

Returned cut-points:

```r
round(z$cutpoints, 4)
```

```
 4.41% 17.68% 52.34% 77.63%
0.4147 0.6366 0.8527 0.9067
```

The function also returns adjusted health levels for each individual that are calculated according the Jürges (2007) method:

```r
rev(table(z$adjusted.levels))
```

```
Excellent Very good      Good      Fair      Poor
     1897      2806      3499      1339       459
```

It is, however, also possible to obtain adjusted health levels for each individual using estimated model thresholds:

```
table(model2$Ey_i)
```

```
Excellent Very good      Good      Fair      Poor
     637      4472      4011       818        62
```

The analysis of health level frequencies is performed using the `getLevels` function:

```
hl <- getLevels(model = model2, formula = ~ sex + ageclass, sep = ' ')
plot(hl)
```
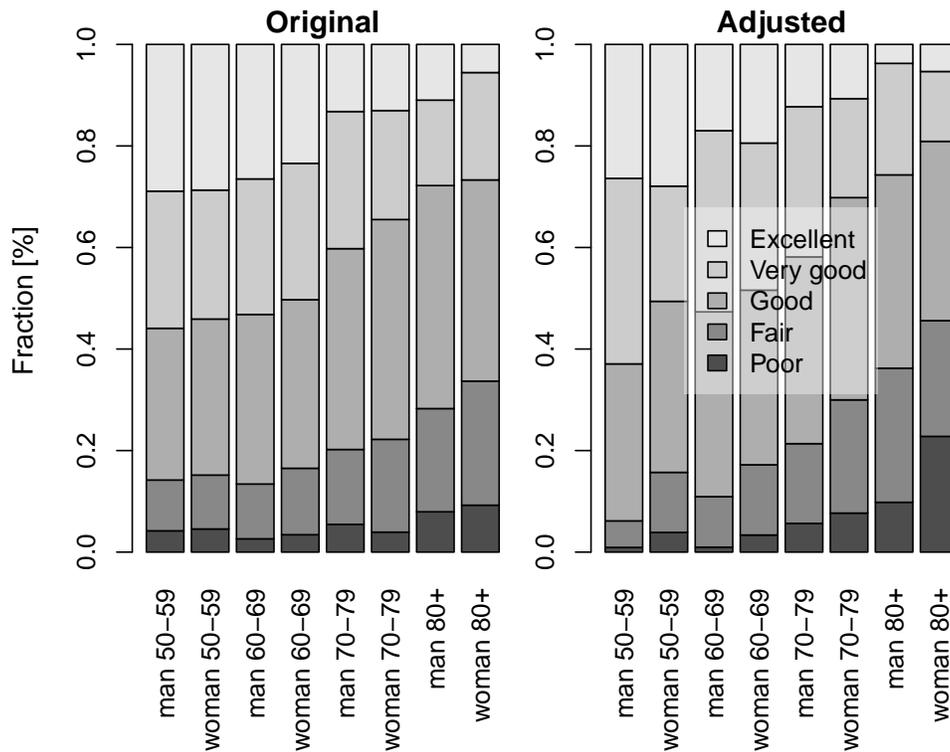


Figure 5: Adjusted and original health levels

The formula describes the grouping variables. It is set to the threshold formula by default, but it can be changed. Here, the country variable is omitted, which means that the age-gender-specific frequencies are calculated for the combined countries. The results are plotted using the generic `plot` function (Fig. 5). The differences between the original and the adjusted frequencies (%) can be calculated directly using the `getLevels` output:

```
round(100*(hl$original - hl$adjusted),2)
```

```
             Poor   Fair   Good Very good Excellent
  man 50-59    3.28   4.77  -1.03     -9.54      2.53
  woman 50-59  0.68  -1.21  -2.96      2.71      0.78
  man 60-69    1.66   0.83  -3.00     -9.00      9.51
  woman 60-69  0.06  -0.75  -1.18     -2.12      3.99
  man 70-79   -0.19  -0.96   2.78     -2.59      0.96
  woman 70-79 -3.75  -4.03   3.47      1.97      2.34
  man 80+     -1.87  -6.07   5.84     -5.14      7.24
  woman 80+  -13.55   1.64   4.31      7.39      0.21
```

## 4.3  Bootstrapping the confidence intervals

The package offers the `boot_hopit` and `percentile_CI` functions for calculating the confidence intervals for any measure derived from the model using parametric bootstrap methods. In each of the bootstrap repetitions, a new set of model coefficients is drawn from the multivariate normal distribution, taking the originally estimated model coefficients as a mean, and using the model estimated variance-covariance matrix. The drawn coefficients are then used to calculate the measure of interest via a user-defined function. In the example below, I calculate the confidence intervals of the difference between the original and the adjusted frequencies of bad health. The bad health is determined by the presence of "Poor" or "Fair" self-rated health categories.

First, a function to be bootstrapped is defined, which is then used to calculate the mentioned difference:

```
diff_BadHealth <- function(model) {
  hl <- getLevels(model = model, formula = ~ sex + ageclass, sep = ' ')
  hl$original[,1] + hl$original[,2] - hl$adjusted[,1]- hl$adjusted[,2]
}
```

The function is used in the `boot_hopit` function, which also needs to specify a fitted model and the data used to fit `model2`

```
B <- boot_hopit(model = model2, func = diff_BadHealth, nboot = 100)
```

The confidence intervals are calculated using the `percentile_CI`

```
est.CI <- percentile_CI(B)
```

The same function is used to calculate the difference based on the estimates of the model.

```
est.org <- diff_BadHealth(model = model2)
```

The code below plots the differences and the bootstrapped confidence intervals

```r
pmar <- par('mar'); par(mar = c(9.5, 6, pmar[3:4]))
m <- max(abs(est.CI))
pos <- barplot(est.org, names.arg = names(est.org), las = 3,
 ylab = 'Original - adjusted differences \nin the prevalences of bad health (%)',
 ylim = c(-m, m), density = 20, angle = c(45, -45), col = c('blue', 'orange'))
for (k in seq_along(pos))
  lines(c(pos[k,1],pos[k,1]), est.CI[,k], lwd = 2, col = 2)
abline(h = 0); box(); par(mar = pmar)
```
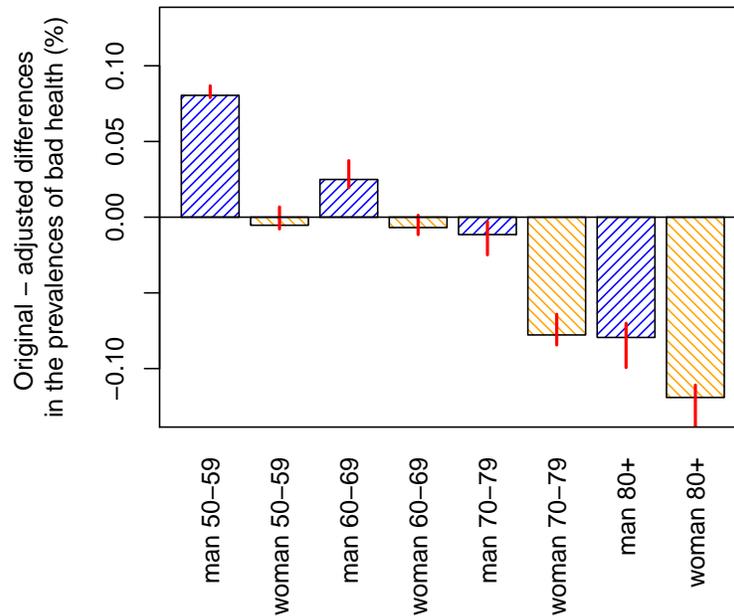


Figure 6: Difference between the original and the adjusted prevalences of bad health. The confidence intervals were calculated using the percentile bootstrap method.

The results (Fig. 6) show that men tend to over-report bad health at ages (50,60] and (50,70], whereas women tend to over-report bad health at ages [70,80]; and that both sexes tend to under-report bad health at ages (80, 120]. See also Oksuzyan et al. (2019) for similar analyses that were performed using real SHARE data.

# 5    Acknowledgements

# References

Bates, Douglas, and Dirk Eddelbuettel. 2013. "Fast and Elegant Numerical Linear Algebra Using the RcppEigen Package." *Journal of Statistical Software* 52 (5): 1–24. https://doi.org/10.18637/jss.v052.i05.

Boes, Stefan, and Rainer Winkelmann. 2006. "Ordered Response Models." *Allgemeines Statistisches Archiv* 90 (1): 167–81. https://doi.org/10.1007/s10182-006-0228-y.

Börsch-Supan, Axel, Martina Brandt, Christian Hunkler, Thorsten Kneip, Julie Korbmacher, Frederic Malter, Barbara Schaan, Stephanie Stuck, and Sabrina Zuber. 2013. "Data Resource Profile: The Survey of Health, Ageing and Retirement in Europe (Share)." *International Journal of Epidemiology* 42 (4): 992–1001. https://doi.org/10.1093/ije/dyt088.

Eddelbuettel, Dirk, and Romain François. 2011. "Rcpp: Seamless R and C++ Integration." *Journal of Statistical Software* 40 (8): 1–18. https://doi.org/10.18637/jss.v040.i08.

Greene, William, Mark N. Harris, Bruce Hollingsworth, and Timothy A. Weterings. 2014. "Heterogeneity in Ordered Choice Models: A Review with Applications to Self-Assessed Health." *Journal of Economic Surveys* 28 (1): 109–33. https://doi.org/10.1111/joes.12002.

Greene, William, and David Hensher. 2010. *Modeling Ordered Choices.* Cambridge University Press.

Ierza, Joseph V. 1985. "Ordinal Probit: A Generalization." *Communications in Statistics - Theory and Methods* 14 (1). Taylor & Francis: 1–11. https://doi.org/10.1080/03610928508828893.

Jürges, Hendrik. 2007. "True Health Vs Response Styles: Exploring Cross-Country Differences in Self-Reported Health." *Health Economics* 16 (2): 163–78. https://doi.org/10.1002/hec.1134.

Jylha, Marja. 2009. "What Is Self-Rated Health and Why Does It Predict Mortality? Towards a Unified Conceptual Model." *Social Science & Medicine* 69 (3): 307–16. https://doi.org/10.1016/j.socscimed.2009.05.013.

King, Gary, Christopher J. L. Murray, Joshua A. Salomon, and Ajay Tandon. 2004. "Enhancing the Validity and Cross-Cultural Comparability of Measurement in Survey Research." *American Political Science Review* 98 (1). Cambridge University Press: 191–207. https://doi.org/10.1017/S000305540400108X.

Liao, Pei-Shan, Yang-Chih Fu, and Chin-Chun Yi. 2005. "Perceived Quality of Life in Taiwan and Hong Kong: An Intra-Culture Comparison." *Journal of Happiness Studies* 6 (1): 43–67. https://doi.org/10.1007/s10902-004-1753-6.

Lumley, Thomas. 2004. "Analysis of Complex Survey Samples." *Journal of Statistical Software* 9 (1): 1–19. https://doi.org/10.18637/jss.v009.i08.

———. 2019. "Survey: Analysis of Complex Survey Samples." *R Package Version 3.35-1.* https://cran.r-project.org/web/packages/survey/index.html.

McKelvey, R. D., and W. Zavoina. 1975. "A Statistical Model for the Analysis of Ordinal Level Dependent Variables." *Journal of Mathematical Sociology* 4 (1): 103–20. https://doi.org/10.1080/0022250X.1975.9989847.

Oksuzyan, Anna, Maciej J. Dańko, Jennifer Caputo, Domantas Jasilionis, and Vladimir M. Shkolnikov. 2019. "Is the Story About Sensitive Women and Stoical Men True? Gender Differences in Health After Adjustment for Reporting Behavior." *Social Science & Medicine* 228: 41–50. https://doi.org/10.1016/j.socscimed.2019.03.002.

R-Core-Team. 2018. "R: A Language and Environment for Statistical Computing [Internet]. Vienna, Austria. R Version 3.4.4 (2018-03-15)." https://www.r-project.org.

Rebelo, Luis Pina, and Nuno Sousa Pereira. 2014. "Assessing Health Endowment, Access and Choice Determinants: Impact on Retired Europeans' (in)activity and Quality of Life." *Social Indicators Research* 119 (3). Germany: Springer: 1411–46. https://doi.org/10.1007/s11205-013-0542-1.