# A tutorial on automatic post-stratification in regression-based norming of psychometric tests

Sebastian Gary1, Alexandra Lenhard1, Wolfgang Lenhard2\*, David Herzberg3

1 Test Development Center, Psychometrica, Dettelbach, Bavaria, Germany

2 Institute of Psychology, University of Würzburg, Bavaria, Germany

3 WPS, Torrance, CA, USA

\* Corresponding author:

E-Mail: wolfgang.lenhard@uni-wuerzburg.de

**Declaration of interests**

The R package cNORM is freely available under the AGPL-3 license. The authors do not receive royalties for the software and neither do they have other financial interests regarding the described procedure. We would like to thank WPS for funding the further development of cNORM with the goal of evaluating and integrating weighting procedures and post-stratification techniques into the software package.

# Abstract

Normed scores are an essential source of information in individual diagnostics. Given the scope of the decisions this information may entail, establishing high-quality, representative norms is of tremendous importance in test construction. Representativeness is hard to establish, though, especially under limited resources and when several stratification variables come into play. Post-stratification techniques like iterative proportional fitting (= raking) aim at increasing the representativeness of the norm sample and thus also the overall quality of the normed scores. This tutorial describes how to apply raking to norm samples in order to determine weights, apply these weights for percentile estimation and retrieve continuous, regression-based norm models with the cNORM package in R. We demonstrate this procedure using a large, non-representative dataset on vocabulary development in childhood and adolescence (N = 4542), using the stratifying variables of sex and parental ethnicity.

*Keywords*: raking, post-stratification, regression-based norming, iterative proportional fitting

Normed scores make it possible to compare an individual’s test result to an adequate reference group (Gary & Lenhard, 2021). Especially in psychodiagnostics, normed scores are often used as a criterion for decisions that have far-reaching consequences for the evaluated individual, like school placement or the provision of support measures for learning disorders (A. Lenhard et al., 2019). Therefore, the computation and inclusion of high-quality norms is a vitally important aspect of high-standard psychological tests. Yet, this goal is a frequent challenge in test construction.

Since the exact distribution of raw scores in the reference population is usually unknown, unfortunately, normed scores cannot be computed directly. Instead, they must be derived from a norm sample, that is, a much smaller subsample representative of the reference population by applying appropriate statistical methods. Over time, different norming approaches have been developed and compared with respect to their influence on the norm quality (for an overview, see Gary & Lenhard, 2021). In many psychometric tests, such as intelligence scales, the normed scores refer only to individuals of the same age. Therefore, conventional approaches usually split the norm sample into several distinct age groups. By contrast, continuous solutions, as for example the regression-based norming approach implemented in the R package cNORM (A. Lenhard et al., 2018), try to model the distribution of the raw scores as a function of explanatory variables such as age. Simulation studies have demonstrated the superiority of regression-based norming over the conventional methods (Gary & Lenhard, 2021; W. Lenhard & Lenhard, 2021; A. Lenhard et al., 2019; Oosterhuis, 2017; Voncken et al., 2021). In this tutorial, we will demonstrate how to combine post-stratification and continuous norming to reduce the bias caused by non-representative norm samples.

## The problem of non-representativeness

The quality of normed scores heavily depends on the chosen norming method. For example, continuous norming requires a much smaller sample size than conventional norming to achieve the same precision (W. Lenhard & Lenhard, 2021). Moreover, it can even smooth out local violations of representativeness (Gary et al., submitted). However, it cannot compensate for a general lack of representativeness in the norm sample. A norm sample is considered representative with respect to the relevant stratification variables (SVs) if the proportions of the various subgroups in the sample match the proportions of the respective strata in the reference population (Kruskal & Mosteller, 1979; Moosbrugger & Kelava, 2012). In other words, representativeness is established, if the marginals and joint distributions of a set of relevant SVs equal the corresponding proportions in the reference population. For example, a norm sample containing an equal size of females and males would be representative regarding the variable sex (assuming an almost equal proportion of female and male individuals in the reference population), whereas a sample of 300 females and 700 males would not be representative, since the respective proportions (30% and 70%) do not match the proportions in the reference population (50% both).

Non-representative norm samples can reduce the quality of normed scores and, therefore, the validity of psychological test results, particularly, if the norm sample is non-representative with respect to SVs that highly correlate with the latent ability to be measured (A. Lenhard et al., 2018). For example, Hernández et al. (2017) showed that parental educational level is an important predictor for children’s cognitive ability and, therefore, recommend including parental educational level as SV in norm tables of intelligence tests for children. By contrast, if the representativeness of the norm sample with regard to such variables is neglected, the distribution parameters may be distorted in various ways. As a consequence, the normed scores may generally be too high or too low or they may be biased for a specific range of percentiles (Gary et al., submitted). In the following, we will describe how to correct, or at least mitigate, the biases of normed scores introduced by non-representative samples.

## Countermeasures for non-representative norm samples

### Probability sampling and sample stratification

Probability or random sampling is possibly the most well-known strategy to establish representativeness of norm samples: The data is drawn in such a way that every individual in the population has the same chance to get included in the norm sample (Lumley, 2011). For example, norming a reading comprehension test for US American children in elementary school, a psychometrician must make sure that theoretically every single student at every elementary school has the same chance to be included in the norm sample. Unfortunately, in most cases, this is an unrealistic requirement, since it is not only very time-consuming and cost-intensive but even neglects ethical guidelines (consent of students and parents, gatekeeper approval …).

Therefore, it is a common approach to use stratification. To this end, the reference population is first divided into homogeneous subgroups, so-called strata. These strata may be based on one or several SVs. In the second step, a specified number of cases is drawn randomly from each of these strata in such a way that the resulting norm sample is representative regarding the proportions of the used strata. For example, to obtain a sample of 1,000 individuals representative with regard to the variable sex (51% male/49% female), 510 males and 490 females must be drawn from the reference population randomly (Moosbrugger & Kelava, 2012). When more than one SV is used, perfect representativeness can only be achieved if all combinations of the different levels of SVs are considered, that is, the joint-distributions have to be used instead of marginal probabilities to match the proportions of the different strata in the reference population.

Unfortunately, sample stratification requires knowing in advance which stratum a person belongs to. But the necessary variables for an individual’s classification, such as the socio-economic status or demographic characteristics, are often collected together with the actual survey or test data (Lumley, 2011; Mercer, Lau, & Kennedy, 2018). Additionally, there may be too many SVs and/or cross-classifications. For example, using three variables with four levels each would result in 4 x 4 x 4 = 64 different strata, with some of the combinations possibly being extremely rare. Such a high number of strata increases the required sample size dramatically, probably far beyond the resources available for a specific norming project. Moreover, individuals with rare combinations of the SVs may not even be available. Finally, when data are collected in clusters, as is the case with many school-based studies, there may be a systematic drop-out. Therefore, it is often impossible to meet all the requirements necessary to establish representativeness with limited resources. To nevertheless minimize the bias caused by unstratified samples, so-called post-stratification methods have been proposed (Lumley, 2011).

### Raking – a post-stratification countermeasure for non-representativeness

While the term stratification usually refers to the selection of individuals for data collection, the term post-stratification refers to techniques aimed at improving the representativeness after the data collection is finished. One way to deal with non-representative data is to randomly reduce cases from overrepresented strata until the required sample composition is achieved. However, this option has the disadvantage that many cases have to be sorted out, resulting in a significant decrease of the original sample size and thus in a loss of statistical power. Therefore, this approach can only be applied to relatively large samples (e.g., A. Lenhard et al., 2015).

But post-stratification can also take a different path: instead of sorting out cases it can use weights. A weight that is assigned to a raw score in a certain subsample stands for the number of individuals this raw score represents. The weights must be determined in such a way that the proportions of the different strata in the reference population are achieved, that is, they must correspond to the ratio between the proportion of a certain stratum in the reference population and the proportion of the respective subgroup in the norm sample (Lumley, 2011; Mercer et al., 2018). Suppose, for example, we have a norm sample that is to be post-stratified with respect to the variable *sex*, for a reference population that contains 49% females and 51% males. If the norm sample contains 45% females and 55% males, every female case will be weighted with , while every male case will be weighted with .

While this procedure can be easily applied with only one SV, more than one SV would require knowledge of the joint distributions, that is, the probabilities of all cross-classifications of the SVs. Moreover, each combination would have to be represented in the data. Unfortunately, these requirements often are not fulfilled. A method to solve this problem is an iterative proportional fitting approach for the weights, which is called raking (Lumley, 2011). Raking comes into play when more than one SV is used. The procedure calculates the required weights for each case in the norm sample considering the different SVs consecutively instead of simultaneously. The process is continued until the weights converge. In the following, we will illustrate this process with an example based on Mercer, Lau & Kennedy (2018): Suppose the marginal probabilities of the SVs *sex* and *parental education* in a reference population are 48% male and 52% female as well as 40% with low parental education, 31% with medium parental education, and 29% with high parental education. The joint probabilities in the actual norm sample are indicated in Table 1.

**Table 1**  
Joint distributions of a fictitious norm sample

|  |  |  |  |
| --- | --- | --- | --- |
| **Parental Education** | **Sex** | |  |
|  | **Female** | **Male** |  |
| Low | 15.33% | 20.00% | 35.33% |
| Medium | 19.17% | 18.82% | 35.00% |
| High | 17.17% | 12.50% | 29.67% |
|  | 51.67% | 48.33% | 100% |

First, weights are computed with respect to the variable *sex*, i.e., the resulting weights are calculated such that the norm sample represents the population with respect to the proportions of males and females (see Table 2, step 1). Second, the weights are adjusted to match the target proportions regarding *parental education* without considering the SV *sex*. (see Table 2, Step 2). But adjusting the weights according to *parental education* again changes the sex ratio slightly. Therefore, the process is repeated until the raking weights have converged (see Table 2, step 5 and step 6).

**Table 2**

Iterative adaption of raking weights

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Parental Education** | **Sex** | |
|  |  | **Female** | **Male** |
| 0 | Low | 1.0000 | 1.0000 |
|  | Medium | 1.0000 | 1.0000 |
|  | High | 1.0000 | 1.0000 |
| 1 | Low | 0.9290 | 1.0759 |
|  | Medium | 0.9290 | 1.0759 |
|  | High | 0.9290 | 1.0759 |
| 2 | Low | 1.0391 | 1.2033 |
|  | Medium | 0.8266 | 0.9573 |
|  | High | 0.9165 | 1.0613 |
| 3 | Low | 1.0498 | 1.1921 |
|  | Medium | 0.8351 | 0.9483 |
|  | High | 0.9260 | 1.0514 |
| 4 | Low | 1.0514 | 1.1939 |
|  | Medium | 0.8346 | 0.9476 |
|  | High | 0.9247 | 1.0500 |
| 5 | Low | 1.0516 | 1.1937 |
|  | Medium | 0.8347 | 0.9475 |
|  | High | 0.9249 | 1.0499 |
| 6 | Low | 1.0516 | 1.1938 |
|  | Medium | 0.8347 | 0.9475 |
|  | High | 0.9249 | 1.0499 |

*Note*. Step 0 represents the non-weighted approach corresponding to weights all equal to one.

As simulations show, usually the raking weights converge within the first five to 20 steps, while generally the number of necessary iterations increases with the number and levels of variables (Battaglia et al., 2009). Since the raking approach only draws on the marginal probabilities, that is, the proportions of the single levels of each SV separately, it is not necessary to know the exact joint distributions of the SVs in the reference population.

### Combining raking and regression-based norming

In this tutorial, we will demonstrate the use of weights in the cNORM package on R. In this package, the norming process has so far consisted of (a) determining percentiles of the raw score distribution, (b) transforming them into preliminary manifest normed scores by means of inverse normal transformation (INT), and (c) multiple regression to continuously model raw scores as a function of the normed scores and age or other explanatory variables. Further details regarding the norming approach used in the cNORM package can be found in A. Lenhard et al. (2019). The cNORM package has recently been extended in order to be able to weight the cases. The weights are used in step (a) and (c) of the norming process. Furthermore, a new function has been added that can be used to generate weights for a specific norm sample by means of raking.

In the following, we will explain the integration of weighting in the cNORM package in the following three steps:

1. Computation and standardization of weights via raking.
2. Weighted ranking of the test raw scores using the standardized raking weights
3. Weighted multiple regression to generate a norm model.

Afterwards, we will demonstrate the whole process using a real data example.

#### Step 1: Computation and standardization of weights via raking.

To generate weights with the raking approach, the function ‘computeWeights()’ in the cNORM package can be used. For this purpose, the marginal probabilities of all SVs (e.g., sex, parental education, region or ethnicity) in the reference population must be known. The information should be based on according census data. The ‘computeWeights()’ function does not only determine the raking weights according to the description above, but additionally divides every weight by the smallest weight, which shifts the weights to values of 1 and above. This transformation preserves the ratio between the proportions of the subgroups but makes them easier to interpret. For example, a weight of 2 can be interpreted as indicating that this specific subgroup should be twice as large to reach the corresponding proportion in the reference population.

#### Step 2: Weighted ranking of the test raw scores using the standardized raking weights

In order to apply the generated weights, they have to be passed via the ‘weights’ function parameter in the ‘cnorm()’ function. This function performs step 2 and step 3 in one single process. If the data are to be analyzed with conventional norming only (i.e., rank based INT per age group) or if step 2 and step 3 are to be performed separately, the ‘rankByGroups()’ function can be used instead. The standardized raking weights are used in both functions to determine the ranks of the different raw scores achieved in the norm sample. To this end, it is counted how often each raw score occurred taking into account the weights, that is, instead of the actual number of cases in the norm sample, the respective weights are added up. In case of ties, average ranks are assigned to the corresponding raw scores. Finally, the weighted ranks are converted into percentiles and preliminary person locations are estimated with INT for every case in the norm sample.

Both the ‘cnorm()’ function and the ‘rankByGroups()’ function return the original observed data, weighted percentiles, manifest normed scores as well as the weights for every case in the norm sample.

#### Step 3: Weighted multiple regression to generate a norm model.

Finally, the standardized raking weights are applied in a multiple regression to estimate a continuous norm model. While steps 1 and 2 can be regarded as data preparation, the multiple regression generates the actual norm model, that is, it fits a model predicting raw scores as a function of person location and age or other comparable explanatory variables such as grade. There is in fact some disagreement about whether stratification weights should be integrated as weights in regression analyses at all (DuMouchel & Duncan, 1983; Skinner & Mason, 2012). We nevertheless consider the integration of raking weights into the multiple regression to be useful, because multiple regression otherwise risks overfitting the data in areas of high data density and underfitting the data in areas of low data density. Besides, we also performed a simulation study comparing multiple regression with or without weights, and both options delivered very similar results.

The ‘cnorm()’ function automatically applies weights in the regression analysis in case they are available. Finally, the resulting continuous norm models can be used to generate norm tables or to directly predict norms for individual cases with desired precision.

# Step-by-step example: Weighted norming of a vocabulary test

## Overview of used norm sample and reference population marginals

In the following, we will illustrate the introduced weighted norming process in a step-by-step example based on a non-representative norm sample for the Peabody Picture Vocabulary Test (PPVT-IV, German adaption; A. Lenhard et al., 2015). The example data set is already contained in the cNORM package and can directly be retrieved using the statistical software R. All code is available in the appendix of the paper.

In our example, we first load the package and assign the PPVT-IV data to the object ‘data’ (cf. Appendix). The data set contains *N* = 4,542 cases with raw scores on receptive vocabulary knowledge (ranging from 0 to 228), spanning an age range from 2;6 years to 17;0 years. Figure 1 shows the used variables, beginning with the explanatory variable *age* in the first row. The next rows contain the SVs *sex* (1 = male, 2 = female), *migration* (i.e., migration to Germany from another country; 0 = native, 1 = migrated) and *region* (west/south/north/east), followed by the variable *raw*, which contains the raw score of the test scale. Finally, it includes a grouping variable *group* with 15 equidistant age groups. Please not that this variable must contain the average age within each age group. In our example, we used age groups spanning one year each, but this must not necessarily be the case. The ideal number of groups depends on the total sample size and also on the functional relation between the dependent variable *raw* and the explanatory variable *age*. During childhood and adolescence, age groups of 6 to 12 month will usually deliver good results for cognitive variables or school achievement. For adults, of course, larger age intervals will be appropriate. But the sample size usually should not be less than 100 per age group.

**Figure 1**

Structure of the PPVT-IV data set



To calculate weights, we need to generate a data frame containing the names of all SVs (column 1), all levels of the variables (column 2) and the population shares of these levels (i.e., the marginal probabilities) in the reference population. In our example, we use the SVs *sex* and *migration* (cf. Appendix, Step 1a), with a sex ratio of 51% vs. 49% (male vs. female) and migration ratio of 70% vs. 30% (native vs. migrated). In the data frame, the proportions must be specified for each level of each variable in terms of decimal fractions. Please note that the proportions for each variable must add up to a value close to one with a tolerable interval of [0.95; 1.05]. Figure 2 displays the composition in the generated data frame ‘marginals’ (table on the right) in comparison to the actual composition of the data sample (table on the left). In the given example, the marginal probabilities of the SV *sex* barely deviate from the reference population. Please note that nevertheless all relevant SVs should always be included in the generation of the weights, since raking can change the proportions of the individual SVs.

**Figure 2**

Composition of unstratified sample in comparison to census-based marginals

## Computation of raking weights

The weights can subsequently be computed using the ‘computeWeights()’ function by passing the norm sample data and the population marginals for the SVs as function parameters. The resulting vector "weights" contains a weight for each individual case in the norm sample obtained by raking and subsequent standardization of the weights (Appendix, Step 1b).

## Ranking with standardized raking weights

In our example, both the ranking and the following multiple regression are conducted with the ‘cnorm()’-function. This function returns an object containing the original data, the weights, the group-specific ranks, the preliminary normed scores as well as powers of the normed scores, of the grouping variable and all interactions between them up to the power parameter *k* (cf. Gary et al., 2021). The object also contains the final statistical model describing the functional relation between raw scores, normed scores and the explanatory variable, which in our case is *age*.

The ‘cnorm()’-function ranks the data groupwise and converts the ranks to percentiles. Subsequently, inverse normal transformation (INT) is applied to convert percentiles to normed scores. The normed scores are returned as *T* scores (*M* = 50, *SD* = 10) by default, but other types of normed scores such as *z* scores or *IQ* scores can also be used. In case, weights are provided, cNORM determines weighted percentiles as already described above (Appendix, Step 2). Since the distribution of the raw scores in the representative population is approximated with this procedure, weighting usually increases the reliability and therefore also the predictive validity of the normed scores (Gary et al., submitted).

## Regression-based norming with inclusion of the raking weights

Finally, the standardized raking weights are automatically used as regression weights to determine the final norm model via multiple regression. This regression is based on the principle of the so-called Taylor polynomials (cf. Dienes, 1957). In short, this principle states that any smooth function can be approximated by a polynomial. Therefore, the raw scores are regressed on powers of the normed scores (up to power parameter *k*), powers of the age or grouping variable (up to power parameter *t*) and all products between these powers. The cNORM package draws on the regression subset selection implemented in the *leaps* package (Lumley, 2020) to be able to select the most parsimonious model that still has a good model fit. The leaps package helps to solve this task by returning a sequence of regression models with an ascending number of terms. Each regression model represents the best selection of predictors for a given number terms. Choosing a model with a higher number of terms increases *R*2 but also carries the risk of overfitting. To avoid the latter, cNORM offers percentile plots for visual inspection, fit indices like *R2*, *Malow’s Cp* and *BIC* and procedures for repeated cross validation. By default, the first model exceeding *R2* > .99 is selected. In most cases, regression models with four to five terms already meet this criterion, leading to parsimonious and well-fitting statistical models.

The ‘cnorm()’ function performs the regression as well as the model selection and also provides a graphical illustration of the model in form of a percentile plot (Figure 3). In our specific example, the selected function includes the following four terms: *raw* ~ *a* + *la* + *l*2*a*2 + *l*3*a*3; with *l* representing the person location in the form of *T* scores and *a* representing age. The adjusted *R*2 amounts to .9911. Furthermore, visual inspection of the graphical model illustration shows no inconsistencies as, for example, intersecting percentile curves, and no signs of overfitting, such as wavy percentile curves. As can be seen in Figure 3, the modelled scores (lines) match the manifest normed scores (dots) very well. To be able to visually inspect and compare different models, we advise to plot a whole series of models with ascending number of terms with the ‘plotPercentileSeries()’ function (Appendix, Step 2 and 3). In our specific example, the returned model stands up to scrutiny and the integrated numerical check yields, *“No violations of model consistency found.”*. If, however, the graphical inspection indicates that another model might even fit better or if the model fit is not satisfying, the ‘cnorm()’ function should be rerun with a fixed number of terms (parameter ‘*terms*’) and/or different power parameters *k* or *t*. In our experience, good models are usually obtained by choosing k = 5 and t = 3. But the power parameters should be reduced in case of overfit of the data. For example, if the correlation between the explanatory variable and the dependent variable is low, t = 2 or even t = 1 may yield better results.

**Figure 3**

Continuous norm model based on weighted cases



*Note*. The plot depicts the manifest percentiles (dots) and the fitted percentile curves (lines) for a selected set of percentiles ranging from PR 2.5 to PR 97.5. The curves are smooth and do not intersect, which is a requirement for a valid norm model.

In summary, the resulting weighted regression model seems to cover a high amount of variance in the norm sample with a high fit and no indications of inconsistencies. It is now ready to be used for either dynamically retrieving normed scores for individual results (e.g., in computer-based testing), computing normed scores for a complete dataset or generating norm tables for manually scoring test results (cf. Appendix).

# Final remarks

Ensuring the representativeness of test norms is of paramount importance in test construction to avoid biased decisions. At the same time, it can be complex or even impossible to obtain fully representative norm samples. In this case, post-stratification strategies such as raking can help to improve the quality of the norms. In this tutorial, we demonstrated the computation and application of raking weights with the cNORM package on the R platform. This package is specifically designed for continuous norming. As demonstrated in simulation studies (Gary et al., in submission), the combination of raking and continuous norming as implemented in the cNORM package in most cases reduces the error introduced in normed scores through non-representative norm samples. Nevertheless, we would like to point out that this is not always the case. For example, weighting cannot fully compensate for very high deviations from representativeness. Moreover, the use of weights may even have negative effects if particularly subgroups at the upper and lower ends of the performance range are severely underrepresented. Therefore, weighting does not replace the effort to acquire a representative sample of sufficient size. In fact, we advise to use the method only if deviations from representativeness are at most moderate, if outlying groups are sufficiently represented, and if the sample size is at least 100 per age group.

# References

Battaglia, M. P., Hoaglin, D. C., & Frankel, M. R. (2009). Practical considerations in raking survey data. *Survey Practice, 2*(5), 2953.

Dienes, P. (1957). *The Taylor series: an introduction to the theory of functions of a complex variable.* New York: Dover Publications.

DuMouchel, W. H., & Duncan, G. J. (1983). Using sample survey weights in multiple regression analyses of stratified samples. *Journal of the American Statistical Association, 78*(383), 535–543.

Gary, S., Lenhard, A., Lenhard, W. & Herzberg, D. (submitted). How to mitigate violations of representativeness in norm samples: Weighting in continuous norming.

Gary, S., & Lenhard, W. (2021). In norming we trust—Verfahren zur statistischen Modellierung kontinuierlicher Testnormen auf dem Prüfstand. *Diagnostica, 67*(2), 75–86.

Gary, S., Lenhard, W., & Lenhard, A. (2021). Modelling Normed scores with the cNORM Package in R. *Psych, 3*(3), 501–521.

Hernández, A., Aguilar, C., Paradell, È., Muñoz, M. R., Vannier, L.-C., & Vallar, F. (2017). The effect of demographic variables on the assessment of cognitive ability. *Psicothema, 29*(4), 469–474.

Kruskal, W., & Mosteller, F. (1979). Representative sampling, III: The current statistical literature. *International Statistical Review/Revue Internationale de Statistique, 47*(3), 245–265.

Lenhard, A., Lenhard, W., & Gary, S. (2018). Continuous Norming (cNORM). *The Comprehensive R Archive Network.* Retrieved from https://CRAN.R-project.org/package=cNORM

Lenhard, A., Lenhard, W., & Gary, S. (2019). Continuous norming of psychometric tests: A simulation study of parametric and semi-parametric approaches. *PLOS ONE, 14*(9), e0222279.

Lenhard, A., Lenhard, W., Segerer, R., & Suggate, S. (2015). *Peabody Picture Vocabulary Test (PPVT-4)*. Pearson Clinical Assesment.

Lenhard, A., Lenhard, W., Suggate, S., & Segerer, R. (2018). A continuous solution to the norming problem. *Assessment, 25*(1), 112–125.

Lenhard, W. & Lenhard, A. (2021). Improvement of Norm Score Quality via Regression-Based Continuous Norming. *Educational and psychological measurement, 81*(2), 229-261. https://doi.org/10.1177/0013164420928457

Lumley, T. (2011). *Complex Surveys – A Guide to Analyses Using R*. Wiley.

Lumley, T. (2020). leaps: Regression Subset Selection. *The Comprehensive R Network.* https://cran.r-project.org/web/packages/leaps/index.html

Mercer, A., Lau, A., & Kennedy, C. (2018). *For weighting online opt-in samples, what matters most.* Pew Research Center.

Moosbrugger, H., & Kelava, A. (2012). *Testtheorie und Fragebogenkonstruktion.* Springer

Oosterhuis, H. E. M. (2017). *Regression-Based Norming for Psychological Tests and Questionnaires* (Dissertation). https://www.google.de/url?sa=t&rct=j&q=&esrc=s& source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiUo7n6tbH3AhUKr6QKHSg9 CFEQFnoECAMQAQ&url=https%3A%2F%2Fpure.uvt.nl%2Fws%2Ffiles%2F16257245%2FOosterhuis\_Regression\_12\_04\_2017.pdf&usg=AOvVaw0kRXxjiYG5kUPC1QVCGSJS

Skinner, C., & Mason, B. (2012). Weighting in the regression analysis of survey data with a cross-national application. *Canadian Journal of Statistics, 40*(4), 697–711.

Voncken, L., Albers, C. J., & Timmerman, M. E. (2021). Bias-Variance Trade-Off in Continuous Test Norming. *Assessment*, *28*(8), 1932–1948. https://doi.org/10.1177/1073191120939155

# Appendix

The following code demonstrates the step-by-step procedure of the tutorial, using RStudio and R.

#####################################################################

### PREPARATION ###

#####################################################################

# install and load library

install.packages(c("cNORM", "survey"), dependencies = TRUE)

library(cNORM)

# assign data object on the basis of the demonstration and display data

data <- ppvt

View(data)

#####################################################################

### STEP 1A: DEFINE MARGINALS ###

#####################################################################

marginals <- data.frame(

variables = c("sex", "sex", "migration", "migration"),

levels = c(1, 2, 0, 1),

share = c(.51, .49, .7, .3))

View(marginals)

# For a comparison: Show marginals and joint distributions in norm sample

table(data$sex)/nrow(data)

table(data$migration)/nrow(data)

#####################################################################

### STEP 1B: COMPUTE WEIGHTS ###

#####################################################################

weights <- computeWeights(data, marginals)

#####################################################################

### STEP 2 & 3: WEIGHTED RANKING AND NORMED SCORE MODELLING ###

#####################################################################

model <- cnorm(raw = data$raw, group = data$group, weights = weights)

# check consistency of the model and search for intersection percentiles

checkConsistency(model)

# plot information function and series of models up to 8 terms

plot(model, "subset")

plot(model, "series", end = 8)

#####################################################################

### RETRIEVING NORMED SCORES ###

#####################################################################

# predict normed score for individual result, raw score 120 at age 4

predictNorm(120, 4, model)

# compute normed score tables for age 4.0, 4.2 and 4.4

# confidence interval and reliability are optional

normTable(c(4.0, 4.2, 4.4), model, CI = .90, reliability = .95)