Multi-level fitting algorithms for population synthesis

Author(s):
Müller, Kirill; Axhausen, Kay W.

Publication Date:
2012

Permanent Link:
https://doi.org/10.3929/ethz-a-010192943

Rights / License:
In Copyright - Non-Commercial Use Permitted
Multi-level fitting algorithms for population synthesis

Kirill Müller
Kay W. Axhausen

Working paper
IVT, ETH Zurich

September 2012
Multi-level fitting algorithms for population synthesis

Kirill Müller, Kay W. Axhausen
IVT
ETH Zurich
8093 Zurich
phone: +41-44-633 33 17
fax: +41-44-633 10 57
kirill.mueller@ivt.baug.ethz.ch

September 2012

Abstract

Agent-based microsimulation models for land use or transportation simulate the behavior of agents over time, although at different time scales and with different goals. For both kinds of models, the initial step is the definition of agents and their relationships. Synthesizing the population of agents often is the only solution, due to privacy and cost constraints. In this paper, we assume that the model simulates persons grouped into households, and a person/household population needs to be synthesized.

Generating a synthetic population requires (a) reweighting of an initial population, taken from census or other survey data, with respect to current constraints, and (b) choosing the households that belong to the generated population. Recently, three multi-level fitting algorithms have been proposed, all of which aim at reweighting the initial population so that all constraints at both household and person levels are satisfied.

The main contribution of this paper is twofold. First, we propose an algorithmic framework for the three algorithms in which the implementation of each algorithm consists only of subtle changes, implying inherent similarity of the three algorithms. Second, we demonstrate formal equivalence of one of the multi-level fitting algorithms to a special case of generalized raking, a procedure known and used in the field of survey statistics for almost 20 years but largely ignored by transportation planners. This allows for the first time to benefit from an enormous amount of theoretical results from a field that focuses primarily on analyzing data from different sources.

Keywords
Population synthesis, Microsimulation, Households, Disaggregation, IPF, Iterative Proportional Fitting, Hierarchical, Simultaneous Control, Multi-Level, Multi-Domain, Relative Entropy
Preferred citation style
1 Introduction

Agent-based microsimulation model systems for land use and transportation planning have come into widespread use. They simulate decisions of agents within an urban area, allowing for more detailed and accurate simulation and prediction of land pricing and travel demand than traditional aggregate models. Often, the agents represent the individual people living in the study area, grouped into households. This paper focuses on such person/household populations.

When implementing such a model system, the initial step is the definition of agents and their relationships; this process is called population synthesis. The main idea is to combine census microdata (the reference sample) with aggregate data at various levels in order to generate a set of agents for which (a) the distribution and correlation of the agents’ attributes are similar to those in the census microsample, and (b) the number of agents within each category matches the aggregate data. Many efforts to generate a synthetic population use the Synthetic Reconstruction (SR) method (Beckman et al., 1996). Here, the reference sample is reweighted to match the aggregate controls, and the weights are used as a probability distribution for generating the agents. See (Müller and Axhausen, 2011b) for a literature review over SR techniques.

In reality, other household members may affect personal decisions (Jones et al., 1983). Thus, properly replicating the household structure is necessary to be able to simulate these interactions. In this paper we assume that a reference sample of households that contains detailed data for all persons is provided. For the SR method, two options are available: (a) the weights obey only household-level constraints, person-level constraints are considered when selecting households (single-level fitting), or (b) the weights obey constraints at both person and household levels, household selection is unconstrained (multi-level fitting). The multi-level strategy greatly simplifies the construction of the final synthetic population using more complicated reweighting algorithms that have become available only recently (Ye et al., 2009; Bar-Gera et al., 2009; Lee and Fu, 2011; Müller and Axhausen, 2011a), and hence can be considered superior to the single-level strategy.

The main contribution of this paper is twofold. First, we propose an algorithmic framework for three multi-level fitting algorithms in which the implementation of each algorithm consists only of subtle changes. This suggests that the three algorithms are inherently similar. Second, we demonstrate formal equivalence of one of the multi-level fitting algorithms to a special case of generalized raking, a procedure that has been long known and used in the field of survey statistics but largely ignored by transportation planners. This allows for the first time to benefit from an enormous amount of theoretical and practical results from a field that focuses primarily on analyzing data from different sources.
The remainder of this paper is structured as follows. In the next section the three multi-level fitting algorithms are described in a common framework. The subsequent section presents the generalized raking procedure and establishes equivalence (for a certain setting) and superiority (in the generic case) to one of the multi-level fitting procedures presented before. We conclude with a summary and an outlook.

2 Multi-level fitting

Generating synthetic populations using the Synthetic Reconstruction (SR) method consists of two principal stages: fitting and generation. In the fitting stage, a disaggregate sample of agents (the reference sample) is reweighted, yielding a positive, usually non-integer weight for each household. The reweighted reference sample corresponds to the full population of the study area and is required to satisfy aggregate constraints (referred to as control totals or controls). Hence, the weights are also referred to as expansion factors; we use both terms interchangeably in this paper. After the reweighting, in the generation stage, this expansion factor is used to construct a disaggregate set of persons and households with attributes required by the microsimulation model.

Expansion factors that satisfy the household-level control totals can be estimated using the well-known IPF algorithm (Deming and Stephan, 1940). In order to also satisfy the person-level controls, one can employ a biased selection procedure that prefers households with persons in still underrepresented categories (Auld and Mohammadian, 2010; Srinivasan and Ma, 2009; Guo and Bhat, 2007). This approach is sketched in Fig. 1(a). However, the biased selection complicates the generation stage and sometimes requires time-consuming computations not suitable for frequent repetition. In addition, it seems to be difficult to specify a mathematical model for the population generated by this procedure; the authors are not aware of such efforts.

In contrast, multi-level algorithms estimate household-level expansion factors that satisfy the controls at both household and person levels. These procedures include Iterative Proportional Updating (IPU) by Ye et al. (2009), entropy maximization (Ent) by Bar-Gera et al. (2009), and Hierarchical Iterative Proportional Fitting (HIPF) by Müller and Axhausen (2011a). (Lee and Fu (2011) present a slightly different formulation of the Ent algorithm.) For all three approaches, the weights define a probability distribution of the households, and construction of the final population is possible using weighted random sampling with replacement. Figure 1(b) illustrates this strategy. This simple model allows generating thousands of instances of similar yet different populations, e.g., in the context of multiple imputation (Rubin, 1987). In the remainder of the paper, only the multi-level fitting approach is considered.
Figure 1: Illustration of single- and multi-level fitting algorithms

(a) Single-level fitting

Reference sample

Control totals (household level)

Result of fitting

Synthetic population

(b) Multi-level fitting

Reference sample

Control totals (person level)

Result of fitting

Synthetic population

ML fit

Biased sampling

Fitting stage
Generation stage
The IPU, Ent and HIPF algorithms were described independently, using different notations and from different perspectives. In this paper, we describe all three algorithms in a common framework using the same notation. This allows a better understanding of the functional differences of the three strategies.

2.1 A common framework for multi-level fitting

All three algorithms for multi-level fitting operate on the following input data: (a) a representative reference sample that contains the characteristics of sampled households and all constituent persons, and (b) control totals for a selection of attributes on both household and person levels. The objective is to estimate a positive weight for each household so that all control totals are satisfied; these weight can be used as expansion factor to obtain a synthetic population. Without loss of generality, we consider only control totals that restrict a single attribute; for all methods presented, interactions of multiple attributes can be controlled for by creating a suitable dummy attribute from all affected attributes.

In the following, we present a pseudocode description of the algorithms and procedures common to all methods, and formally describe the notation which is used in the pseudocode and in the remainder of the paper and closely follows Müller and Axhausen (2011a).

2.1.1 Pseudocode

Figure 2 shows a framework of routines common to all three algorithms. The procedure ML-FIT in Figure 2(a) is the main routine. The household weights are initialized to unity (or to the weight given by the reference sample). Then, the control totals are processed iteratively until convergence. For each category of each control total, household weights are adjusted to match (cf. procedures H-FIT and P-FIT in Figs. 2(b) and 2(c)). In turn, these procedures call H-ADJUST and P-ADJUST for adjustment of each controlled attribute (cf. Figs. 2(d) and 2(e)).

Interestingly, the adjustment of the household-level control totals is carried out identically for all three algorithms (cf. procedure H-ADJUST in Fig. 2(d)). It is identical to the IPF algorithm in the list-based version (Pritchard and Miller, 2009). Thus, all three algorithms are equivalent to IPF if applied to only one level of aggregation. The algorithms differ only in how the adjustment is carried out for control totals at person level; the different flavors of the P-ADJUST procedure (Fig. 2(e)), or the replacement P-FIT procedure for the HIPF algorithm, will be presented in the subsequent sections.
Multi-level fitting algorithms for population synthesis

September 2012

Figure 2: A generic framework for multi-level fitting algorithms

(a) Procedure \textbf{ML-FIT}(H,P,C_{x_1},\ldots,C_{\xi_1},\ldots) – the main routine of the generic framework

\begin{algorithmic}
\REQUIRE Reference sample \((H,P)\)
\REQUIRE Household-level control totals \(C_{x_1},C_{x_2},\ldots\)
\REQUIRE Person-level control totals \(C_{\xi_1},C_{\xi_2},\ldots\)
\REQUIRE Prior weights \(f_h^{(0)}\) (unity if absent)
\ENSURE Expansion factors \(f_h\) obeying all control totals
\STATE \(f_h \leftarrow f_h^{(0)} \text{ for all } h \in H\)
\WHILE {convergence not reached}
\STATE \(f_h \leftarrow \text{H-FIT}(H,f_h,C_{x_1},\ldots)\)
\STATE \(f_h \leftarrow \text{P-FIT}(P,f_h,C_{\xi_1},\ldots)\)
\ENDWHILE
\RETURN \(f_h\)
\end{algorithmic}

(b) Subroutine \textbf{H-FIT}(H,f_h,C_{x_1},\ldots) – fitting at household level

\begin{algorithmic}
\REQUIRE Households \(H\) from reference sample
\REQUIRE Current expansion factors \(f_h\)
\REQUIRE Household-level control totals \(C_{x_1},C_{x_2},\ldots\)
\ENSURE Improved expansion factors \(f_h\)
\FORALL {household-level controls \(C_x\)}
\FORALL {attributes \(x\)}
\STATE \(f_h \leftarrow \text{H-ADJUST}(H,f_h,x,C_x(x))\)
\ENDFOR
\ENDFOR
\RETURN \(f_h\)
\end{algorithmic}

(c) Subroutine \textbf{P-FIT}(P,f_h,C_{\xi_1},\ldots) – fitting at person level

\begin{algorithmic}
\REQUIRE Persons \(P\) from reference sample
\REQUIRE Current expansion factors \(f_h\)
\REQUIRE Person-level control totals \(C_{\xi_1},C_{\xi_2},\ldots\)
\ENSURE Improved expansion factors \(f_h\)
\FORALL {person-level controls \(C_\xi\)}
\FORALL {attributes \(\xi\)}
\STATE \(f_h \leftarrow \text{P-ADJUST}(P,f_h,\xi,C_\xi(\xi))\)
\ENDFOR
\ENDFOR
\RETURN \(f_h\)
\end{algorithmic}

(d) Subroutine \textbf{H-ADJUST}(H,f_h,x,C_x) – fitting one category at household level

\begin{algorithmic}
\REQUIRE Current household-level category \(x\)
\REQUIRE Value \(C_x\) of the control total for category \(x\)
\ENSURE Expansion factors \(f_h\) that match control \(C_x\)
\STATE \(r \leftarrow C_x \div F_x\)
\STATE \(f_h \leftarrow f_h \cdot r \text{ for all } h \in H_x\)
\RETURN \(f_h\)
\end{algorithmic}

(e) Subroutine \textbf{P-ADJUST}(P,f_h,\xi,C_\xi) – fitting one category at person level

\begin{algorithmic}
\REQUIRE Current person-level category \(\xi\)
\REQUIRE Value \(C_\xi\) of the control total for category \(\xi\)
\ENSURE Expansion factors \(f_h\) that match control \(C_\xi\)
\STATE (implementation differs for each algorithm)
\end{algorithmic}

2.1.2 Notation

We use two sets \(H\) and \(P\) to denote household and person attributes of the hierarchical reference sample:

\begin{align}
H &= \{(a_1,b_1,\ldots),\ldots,(a_h,b_h,\ldots),\ldots,(a_n,b_n,\ldots)\} \subset a \times b \times \ldots \tag{1} \\
P &= \{(a_1,b_1,h_1),\ldots,(a_p,b_p,h_p),\ldots,(a_m,b_m,h_m)\} \subset a \times b \times \cdots \times H. \tag{2}
\end{align}
The two multisets are linked through \( h_p \in H \); this attribute assigns each person to a household.

We define the set of households \( H_x \) for a given category \( x \) as follows:

\[
H_x := \{ h \in H : (\ldots, x_h, \ldots) = (\ldots, x, \ldots) \}.
\] (3)

In addition, we define the sum \( F_x \) of all expansion factors for category \( x \):

\[
F_x := \sum_{h \in H_x} f_h.
\] (4)

Analogously, \( P_\xi \) and \( \Phi_\xi \) denote the persons in the category \( \xi \) and the sum of their expansion factors, respectively. The expansion factor for a person is equivalent to its household’s expansion factor:

\[
P_\xi := \{ p \in P : (\ldots, \xi_p, \ldots) = (\ldots, \xi, \ldots) \}
\] (5)

\[
\phi_p := f_{h_p}
\] (6)

\[
\Phi_\xi := \sum_{p \in P_\xi} \phi_p.
\] (7)

Thus, \( P_h \subset P \) lists the constituent persons of each household. The implicit household attribute \( p_h := |P_h| \) defines the number of persons in household \( h \).

By \( C_x : x \rightarrow \mathbb{N} \) and \( C_\xi : \xi \rightarrow \mathbb{N} \) we denote the control total for the attributes \( x \) and \( \xi \), respectively. The objective of multi-level fitting is to estimate expansion factors \( f_h \) so that all control totals are satisfied:

\[
F_x = C_x(x) \text{ for all categories } x \text{ of all controlled household attributes } x,
\] (8)

\[
\Phi_\xi = C_\xi(\xi) \text{ for all categories } \xi \text{ of all controlled person attributes } \xi.
\]

Thus, each category in all controlled household or person attributes defines exactly one constraint in Eq. (8).

The following three sections describe how IPU, Ent, and HIPF fit into this framework.

2.2 Iterative Proportional Updating

The IPU algorithm by Ye et al. (2009) uses the following adjustment at person level: Proportionally rescale the weights of all households that contain at least one person that falls into the current category \( \xi \). That is, the rescaling proportion for a household does not depend on the number
Figure 3: The implementations of **P-ADJUST** used for the IPU and Ent algorithms

(a) Subroutine **P-ADJUST-IPU**\((P, f_h, \xi, C_\xi)\) – fitting one category at person level for the IPU algorithm

**Require:** Current person-level category \(\xi\)

**Require:** Value \(C_\xi\) of the control total for category \(\xi\)

**Ensure:** Expansion factors \(f_h\) that match control \(C_\xi\)

1. \(H_\xi \leftarrow \{h_p : p \in P_\xi\}\)
2. \(r \leftarrow C_\xi \div \Phi_\xi\)
3. \(f_h \leftarrow f_h \cdot r\) for all \(h \in H_\xi\)
4. return \(f_h\)

(b) Subroutine **P-ADJUST-ENT**\((P, f_h, \xi, C_\xi)\) – fitting one category at person level for the Ent algorithm

**Require:** Current person-level category \(\xi\)

**Require:** Value \(C_\xi\) of the control total for category \(\xi\)

**Ensure:** Expansion factors \(f_h\) that match control \(C_\xi\)

1. \(\lambda'_\xi \leftarrow \lambda_\xi\)
2. Find \(\lambda_\xi\) that satisfies the constraint corresponding to \(\xi\) in Eq. (8)
3. \(r \leftarrow \lambda_\xi \div \lambda'_\xi\)
4. for all \(h \in H\) do
5. \(i \leftarrow |P_h \cap P_\xi|\)
6. \(f_h \leftarrow f_h \cdot r^i\) for all \(p \in P_\xi\)
7. return \(f_h\)

of constituent persons that fall into the current category \(\xi\). The procedure **P-ADJUST-IPU** in Fig. 3(a) formalizes this.

### 2.3 Entropy Maximization

The Ent approach suggested by Bar-Gera et al. (2009) seeks to obtain the desired household weights by solving the following optimization problem: Minimize

\[
\sum_h f_h \ln \left( \frac{f_h}{f_h^{(0)}} - 1 \right) \tag{9}
\]

with respect to Eq. (8).

The objective function is very similar to relative entropy (Kullback and Leibler, 1951), which has been widely applied in urban and regional modeling as a measure of similarity between distributions (e.g., Wilson, 1970). The main idea behind optimizing this measure is that the reweighting should introduce the least possible amount of new information. In fact, Lee and Fu (2011) suggest optimizing relative entropy instead:

\[
I = \sum_h f_h \ln \left( \frac{f_h}{f_h^{(0)}} \right) \tag{10}
\]

Note that the difference between the two objective functions vanishes if there is at least one household-level constraint that covers all household types so that the sum of all expansion
factors $\sum_h f_h$ is fixed. In the following, we use the objective function defined in Eq. (9).

It is not immediately obvious how the Ent approach fits into our framework. In what follows, we sketch this transition. Applying the method of Lagrange Multipliers to Eqs. (8) and (9) yields the following necessary condition for $f_h$:

$$\ln \left( \frac{f_h}{f_h(0)} \right) - \lambda_{xh} - \ldots - \sum_{p \in P_h} \lambda_{\xi_p} - \ldots = 0.$$  \hspace{1cm} (11)

This can be rewritten as

$$f_h = f_h(0) \cdot \exp \left( \lambda_{xh} + \ldots + \sum_{p \in P_h} \lambda_{\xi_p} + \ldots \right)$$

and plugged back into Eq. (8).

Bar-Gera et al. (2009) suggest adjusting the $\lambda_x$ and $\lambda_\xi$ in a round-robin fashion so that after each adjustment the corresponding constraint from Eq. (8) is satisfied. Each adjustment step boils down to finding the unique positive real-valued root of a polynomial. For the adjustment of a $\lambda_x$ in a household-level constraint, this polynomial is just a linear equation, and the solution is equivalent to the one obtained by H-ADJUST. The procedure P-ADJUST-ENT in Fig. 3(b) summarizes the adjustment for person-level constraints.

As suggested by Lee and Fu (2011), a direct solution can be obtained by employing BGFS or similar algorithms for solving nonlinear equation systems. Our experience suggests that this approach clearly outperforms iterative adjustment of the $\lambda_x$ and $\lambda_\xi$, however no formal runtime measurements have been carried out so far.

Note that the execution of line 6 in this routine is equivalent to recomputing the $f_h$ with the new $\lambda_\xi$ from Eq. (8). A household’s weight is readjusted depending on how many persons of the given category it contains – the more matching persons, the stronger. Informally speaking, it “pays off” more to adjust a household with many matching persons.

### 2.4 Hierarchical Iterative Proportional Fitting

The HIPF algorithm by Müller and Axhausen (2011a) works in a slightly different way. From the point of view of our framework, it uses an alternative implementation of the P-FIT subroutine that converts the household-level weights into person-level weights and vice versa. While obtaining person-level weights from household-level weights is straightforward (cf. Eq. (6)),
Figure 4: Subroutine \( \text{P-FIT-HIPF}(P, f_h, C_{\xi_1}, \ldots) \) – person-level fitting for the HIPF algorithm

\begin{verbatim}
Require: Persons \( P \) from reference sample
Require: Current expansion factors \( f_h \)
Require: Person-level control totals \( C_{\xi_1}, C_{\xi_2}, \ldots \)
Ensure: Improved expansion factors \( f_h \)
1: \( \phi_p \leftarrow f_{hp} \) for all \( p \in P \)
2: \( \phi_p \leftarrow \text{H-FIT}(P, \phi_p, C_{\xi_1}, \ldots) \)
3: Compute \( f_h \) from \( \phi_p \), respecting the person-per-household ratio
4: return \( f_h \)
\end{verbatim}

the reverse direction is more complicated; Müller and Axhausen (2011a) suggest an entropy-minimizing step that controls for the persons-per-household ratio.

Figure 4 shows the \( \text{P-FIT-HIPF} \) procedure that implements the fitting at person level using the HIPF approach. Note the invocation of \( \text{H-FIT} \) in line 2! This performs a single-level fitting at the person level, using person-level weights. Recall that IPF, and thus \( \text{H-FIT} \), computes entropy-minimal estimates (Ireland and Kullback, 1968). The repeated transition between the household and the person level and the application of entropy-minimizing procedures at each level considers the entropy at both levels, in contrast to the Ent procedure which minimizes only household-level entropy.

2.5 Summary

In the common framework, IPU and Ent are surprisingly similar: Only the \( \text{P-ADJUST} \) procedure is implemented differently, and the difference lies only in the treatment of households with more than one person in a certain category. IPU treats identically all households with at least one such person, while Ent reweights stronger the households with more than one such person. HIPF switches to person-level weights when adjusting for person-level controls and thus also requires an alternative implementation of the \( \text{P-FIT} \) procedure; however, the \( \text{P-FIT-HIPF} \) subroutine mostly makes use of existing routines. The large amount of shared (pseudo-)code suggests an inherent similarity between the three fitting methods.

Conflicting or infeasible control totals are not discussed here. Most of the methods presented above will happily accept such malformed control totals but fail to converge to a stable solution. It may happen that a result that satisfies some of the constraints can be obtained; however, it is difficult to precisely describe the properties of such an incomplete solution. The only exception is the formulation of the Ent algorithm as a constrained optimization problem: Here, conflicting
control totals will render the optimization problem intractable, and some nonlinear solvers will not even provide a “partial” solution. Bar-Gera et al. (2009) suggest a relaxed formulation of the problem that seeks for a “best approximation” in the presence of conflicting or infeasible control totals.

3 Survey calibration

A lot of literature on synthetic population generation in the field of transport planning refers to the seminal paper by Beckman et al. (1996). In turn, this paper refers, among others, to the works by Deming and Stephan (1940), Ireland and Kullback (1968), and Little and Wu (1991). The latter were driven by the need to adjust (or calibrate) survey data to known marginal totals, a frequent task in the field of survey statistics. Survey calibration is used to correct nonresponse and selection bias before performing statistical analyses on response variables.

Deville and Särndal (1992) have proposed a common framework for weighting systems of which both generalized regression (GREG) estimation and IPF are special cases. The subsequent paper by Deville et al. (1993) elaborates on this idea by focusing more on the reproduction of known marginal counts (generalized raking), and presents CALMAR, a software implementation for the statistical system SAS. An excellent review of calibration methods, including the somewhat trivial case of poststratification (calibration against only one set of marginals), can be found in Zhang (2000). The survey package (Lumley, 2012, 2010) is a recent implementation for the R statistical software package (R Development Core Team, 2012). In the case of household surveys, generalized raking supports simultaneous calibration against household-level and person-level control totals.

The papers on GREG estimation (Deville and Särndal, 1992) and generalized raking (Deville et al., 1993) consider the estimation of population totals (i.e., the expected number of individuals with a given characteristic), as well as of their variance, the latter being the focus of both papers. For the generation of synthetic populations, the methodology for obtaining the weights is of primary interest; the estimated variance can help validating the generated population.

In a sense, the generation of a synthetic population can be interpreted as a complicated statistical analysis on many response variables, namely, one for each combination of categories for all attributes of interest (including geographical location). A disaggregate synthetic population can be derived directly from these joint estimates. This justifies the use of the mathematical machinery developed in the field of survey statistics for generating synthetic populations and assessing their accuracy.
The advances in the field of survey statistics do not seem to be widely applied within the trans-
p ortation planning community, and in particular not within the population synthesis community;
we have found only one paper, by Armoogum and Madre (2002), that refers to (Deville et al.,
1993) and describes the usage of the CALMAR software for the calibration of a long-distance
travel survey. On a side note, generalized raking operates directly on the unrolled survey data
instead of creating a (potentially huge) crosstabulation – this has been suggested independently
by Pritchard and Miller (2012). Other than that, several methods (including our own) that solve
a very similar problem appeared in the recent past in the transportation planning literature,
cf. Section 2. The vast differences in terminology, notation, and perhaps application might be
 the reasons for this parallel development. This paper bridges the gap by formally demonstrating
equivalence of the methods used in both fields, thus justifying the usage of theoretical results,
algorithms and software implementations from survey statistics for the engineering problem at
hand.

In the following, we show that the Ent algorithm introduced in Section 2.3 corresponds to
the raking ratio case of generalized raking (Deville et al., 1993), abbreviated by Cal. After
introducing the terminology and notation used by Deville et al. (1993) and mapping it to the
population synthesis problem, equivalence is proven formally. We argue that the generation of a
synthetic population can be interpreted as a statistical analysis, to which the tools and methods
derived for the field of survey statistics can be applied as well. Finally, we outline potential
benefits from using generalized raking for the fitting stage.

3.1 Terminology and notation

For the Cal method (Deville et al., 1993), a tight sequence of natural numbers $U = \{1, \ldots, k, \ldots N\}$
constitutes the finite population, and $y_k$ for $k \in \mathbb{N}$ denotes the value of the variable of interest
of the $k$th population element. Here, $U$ corresponds to the entire population of the study area
at household level; the sample used as input for the fitting stage is denoted by a subset $s \subseteq U$.
Without loss of generality we can fix $s := \{1, \ldots, n\}$ to be consistent with our notation.

In addition, the Cal method defines an auxiliary vector value $x_k = (x_{k1}, \ldots, x_{kj}, \ldots, x_{kJ})^T$ for
each member of the population for which the (vector-valued) population total $t_x := \sum_{k \in U} x_k$
is accurately known. The auxiliary vector and the population total correspond to categories of
household and person attributes, and to control totals, respectively, in our notation. Categories of
household attributes are converted to dummy values (zero or one), and for categories of person
attributes the number of persons within this category is used. Formally, for each household
category $a$ or person category $\alpha$ of each controlled attribute $a$ or $\alpha$ (in our notation), a new index
is allocated in the auxiliary vector:

\[ x_{ka} := \begin{cases} 
1 : a_k = a \\
0 : \text{else}
\end{cases} \]  

(13)

\[ x_{\alpha} := |P_k \cap P_{\alpha}|. \]  

(14)

In the introduction of their exposition, Deville et al. (1993) note, on a more abstract level, the application of generalized raking for simultaneously matching to, say, known household and person counts in a household sample.

For the Cal method, the subsample \( s \subseteq U \) is treated as a random sample with given inclusion probabilities \( \pi_k \), denoting the probability that the population element \( k \) appears in the sample. The inclusion probabilities depend on the sampling design of the survey. The design weights are derived from the Horvitz-Thompson estimator \( \sum_{k \in U} y_k \) of the population total of the variable of interest (Horvitz and Thompson, 1952) and denoted by \( d_k := 1/\pi_k \). The objective is to find new weights \( w_k \) close to the design weights but satisfying the calibration equation, requiring that the weighted sum of the auxiliary vectors equals the population total:

\[ \sum_{k \in s} w_k x_k = t_x = \sum_{k \in U} x_k. \]  

(15)

It is easy to verify that above equation is equivalent to Eq. (8) in our notation; for this, recall that in our notation the weights \( w_k \) are denoted by \( f_h \).

### 3.2 Equivalence of Cal and Ent

Similarity of weights is assessed through a distance function \( G \) with argument \( w_k/d_k \) and certain regularity properties. The distance of the weights for the entire sample is given by \( \Delta = \sum_{k \in s} d_k G(w_k/d_k) \). The goal is to minimize \( \Delta \) while fulfilling the calibration equation. For Cal, \( G(x) := x \log x - x + 1 \) is used, for which the distance becomes

\[ \Delta = \sum_{k \in s} d_k ((w_k/d_k) \log(w_k/d_k) - (w_k/d_k) + 1) \]

(16)

\[ = \sum_{k \in s} (w_k \log(w_k/d_k) - w_k + d_k) \]  

(17)

\[ = \sum_{k \in s} w_k (\log(w_k/d_k) - 1) + \sum_{k \in s} d_k \]  

(18)

\[ = \sum_{k \in s} w_k (\log(w_k/d_k) - 1) + \text{const.} \]  

(19)
Equation (19) holds because the $d_k$ are constant. Thus, the objective function corresponds precisely to the one used for the Ent method (Eq. (9)).

In the Cal method, the constrained optimization problem is then transformed to an unconstrained equation system using Lagrange transformation, very similarly to [Bar-Gera et al., 2009; Lee and Fu, 2011]. For solving the equation system, both Newton’s method and an algorithm similar to the coordinate search (Bar-Gera et al., 2009) are described. The coordinate search is reported to be equivalent to IPF and to converge slower than Newton’s method; this aligns well with our own experience.

3.3 Application

In generalized raking, the weights are used to estimate the population total of $y$, the variable of interest: $\sum_{k \in s} w_k y_k \approx \sum_{k \in U} y_k$. This can be generalized to a vector of variables of interest. Consider the extreme case where each population element corresponds to an element of the vector of variables of interest. Here, by the Horvitz-Thompson estimator, the weight of a population element equals the expected number of times a similar population element is present in the finite population. The calibration totals are satisfied exactly by the weights.

3.4 Discussion

As shown above, generalized raking is equivalent to Ent when using the “raking ratio” distance, uniform design weights, and specially crafted auxiliary vector values. Obviously, by dropping one or several of these constraints the method becomes only more powerful:

- It is possible to calibrate against continuous or natural-valued attributes, e.g. the number of cars or the income, if population totals are available.
- Variance estimation is possible under the assumption that the selection probabilities are pairwise independent. Weight (i.e., expected frequency) and variance can be transformed to a distribution of counts, which can be directly sampled from to obtain integer expansion factors for each observation in the reference sample.
- Other distance measures, for instance “logit” or “linear”, perhaps combined with bounds on the weights, can be used. However, Deville et al. (1993) note that while individual weights might change considerably, the total and variance estimates usually do not vary very much; according to Deville and Särndal (1992), generalized raking is asymptotically equivalent for all feasible distance functions.
Just like with multi-level fitting, the calibration totals are required to be consistent to allow for a well-defined solution. In practice, this seems to require manual readjustment of the calibration totals and hence an implicit decision on which data is more trustworthy.

4 Conclusion

In this paper, we presented three multi-level fitting techniques – Iterative Proportional Updating (Ye et al., 2009), Entropy (Bar-Gera et al., 2009) and Hierarchical Iterative Proportional Fitting (Müller and Axhausen, 2011a) – in a unified algorithmic framework using the same notation. From this perspective, all three methods are inherently similar. Furthermore, we formally established equivalence between the Entropy method and a special case of generalized raking, a well-known technique in survey statistics. This leads to the following conclusions:

- The application of multi-level fitting methods is methodologically justified, now in the broader scope of generalized raking.
- Standard software like (R Development Core Team, 2012; Lumley, 2012) can be applied for the fitting (or calibration) stage in population synthesis tasks.
- Access is provided to strong theoretical results from a field where combining data from different sources is the main interest and not only a necessary but annoying nuisance.

So far, the results of this paper are of purely theoretical nature. A prototype implementation is available for the three multi-level fitting techniques mentioned above. We intend to provide a well-tested and well-documented reimplementation as an R package; this will also contain a wrapper for the survey package so that now all four methods can be used interchangeably. This will enable us to conduct various tests using real-world data in order to assess the practical differences of the four methods. Special attention will be paid to the following issues:

- inconsistent or infeasible control totals – a significant problem in practice;
- the effect of prior weights;
- variance estimation based on (Deville et al., 1993); and
- incomplete control totals, i.e. not controlling for certain categories.

The methods discussed here operate on the following assumptions w.r.t. the reference sample:

Representativeness The reference sample is a valid if skewed description of the study area’s population.

Comprehensiveness The reference sample contains at least one observation for each type of individual in the true population.
**Accuracy**  The weights attached to each observation in the reference sample have a low expected difference from the “true” weights (variance/standard deviation).

**Availability**  A trivial prerequisite.

More often than not, one or several of these prerequisites cannot be fulfilled. It is worthwhile to apply and develop alternative methods that can be used where the data situation is less than ideal.

5 **Acknowledgements**

The authors gratefully acknowledge the financial support of the Swiss National Science Foundation for the project “A generalized approach to population synthesis” (grant 138270). The authors would also like to thank Dr. Hillel Bar-Gera for helpful remarks on alternative formulations of the entropy objective function, and Pieter Fourie for valuable comments to an earlier version of this paper and for experiments using a prototype implementation of the methods presented here.

6 **References**


