Calibration in survey sampling as an optimization problem

Gareth Davies, Jonathan Gillard and Anatoly Zhigljavsky

Dedicated to Professor Panos Pardalos on occasion of his 60-th birthday

Abstract Calibration is a technique of adjusting sample weights routinely used in sample surveys. In this paper, we consider calibration as an optimization problem and show that the choice of optimization function has an effect on the calibrated weights. We propose a class of functions that have several desirable properties, which includes satisfying necessary range restrictions for the weights. In this paper, we explore the affect these new functions have on the calibrated weights.

1 Introduction

Calibration of survey samples is one of the key issues in official statistics and analysis of panel data (in particular, in market research). The problem of calibration can be defined informally as follows. Suppose there are some initial weights $d_1,\ldots,d_n$ assigned to n objects of a survey. Suppose further that there are m auxiliary variables and that for these auxiliary variables the sample values are known, either exactly or approximately. The calibration problem seeks to improve the initial weights by finding new weights $w_1,\ldots,w_n$ that incorporate the auxiliary information. In a typical practical problem, the sample size n is rather large (samples of order 10^4 and larger are very common). The number of auxiliary variables m can also be large although it is usually much smaller than n.

Three main reasons are advocated for using calibration in practice (see, for example, [2]). The first of these is to produce estimates consistent with other sources of data. Indeed, when a statistical office publishes the same statistics via two data sources, the validity of the statistics will be questioned if there are contradictions between the sources. The second reason is to reduce the sampling variance of estimates as the inclusion of the additional calibration information can lead to a reduction in

Gareth Davies, Jonathan Gillard and Anatoly Zhigljavsky
Cardiff School of Mathematics, Cardiff University, e-mail: {DaviesGP2, GillardJW, ZhigljavskyAA}@cardiff.ac.uk

the variance of the estimators (see for example [11]). The third argument for calibration is a reduction of the coverage and/or non-response bias (see for example [10]).

In this paper, we properly formulate the problem of calibration of weights as an optimization problem, study properties of the corresponding optimization problems and give recommendations on how to choose the objective function. We claim that the literature on calibration has ignored this important issue which lead to the recipes which were inefficient or even incorrect.

Notation

We use the following key notation throughout the paper:

$D = (d_1, ..., d_n)'$:	vector of initial weights,
$W = (w_1, ..., w_n)'$:	vector of calibrated weights,
$G = (g_1, ..., g_n)'$:	vector of the g-weights $g_i = w_i/d_i$,
$L = (l_1, ..., l_n)'$:	vector of lower bounds for the g-weights,
$U = (u_1, ..., u_n)'$:	vector of upper bounds for the g-weights,
$X = (x_{ij})_{i,j=1}^{n,m}$:	given $n \times m$ matrix,
$A = (a_{ij})_{i,j=1}^{n,m}$:	$n \times m$ matrix with entries $a_{ij} = d_i x_{ij}$,
$T = (t_1, ..., t_m)'$:	an arbitrary $m \times 1$ vector,
$\mathbf{1} = (1, 1, \ldots, 1)'$	$n \times 1$ vector of ones,
$\mathbb{G}$	feasible domain in the calibration problem.

2 Calibration as an Optimization Problem

A vector of initial weights $D = (d_1, ..., d_n)'$ is given. The d_i are always assumed to be positive: $d_i > 0$ for all i. Our aim is to calibrate (improve) these initial weights in view of some additional information. The vector of calibrated (improved) weights will be denoted by $W = (w_1, ..., w_n)'$.

We are given a matrix $X = (x_{ij})_{i,j=1}^{n,m}$ of realizations of m auxiliary variables. The (i, j)-th entry x_{ij} of X denotes the value which the ith member of the sample takes on the jth auxiliary variable Formally, X is an arbitrary $n \times m$ matrix. Given the vector $T = (t_1, ..., t_m)'$, exact (hard) constraints can be written as $X'W = T$, whereas approximate (soft) constraints are $X'W \simeq T$. These constraints, whether exact or approximate, define the additional information we use in the calibration of the weights.

It is sometimes natural to impose a constraint on the sum of the weights. In this paper, we shall consider the sum of weights constraint $\sum_{i=1}^{n} w_i = \sum_{i=1}^{n} d_i$ or, in vector notation, $\mathbf{1}'W = \mathbf{1}'D$, where $\mathbf{1} = (1, 1, \ldots, 1)'$. This constraint is motivated in [17]. The condition $\mathbf{1}'W = \mathbf{1}'D$ can be added to the set of the main constraints $X'W = T$ (see, for example, [16]). Hence we do not formally distinguish the cases when the condition $\mathbf{1}'W = \mathbf{1}'D$ is required or not.

In most practical cases of survey sampling and panel data analysis, the ratios of the weights w_i and d_i are of prime importance rather than the weights w_i themselves and the so-called g-weights $g_i = w_i/d_i$ are considered. Denote the vector of g-weights by $G = (g_1,...,g_n)'$ and consider this vector as the vector of calibrated weights we are seeking.

Since $d_i > 0$ for all i, the hard constraints $X'W = T$ can be written in the form $A'G = T$, where the matrix $A = (a_{ij})_{i,j=1}^{n,m}$ has elements $a_{ij} = d_i x_{ij}$. Correspondingly, soft constraints $X'W \simeq T$ have the form $A'G \simeq T$.

In addition to either hard or soft constraints, the following constraints on G have to be imposed. First of all, the calibrated weights must be nonnegative; that is, $g_i \geq 0$ for all i. Moreover, much of the calibration literature, see for example [4] and [18], recommends imposing stricter constraints on the g-weights of the form $L \leq G \leq U$, where $L = (l_1,...,l_n)'$ and $U = (u_1,...,u_n)'$ are some given $n \times 1$ vectors such that $0 \leq l_i < 1 < u_i \leq \infty$ for all i. That is, the g-weights should satisfy $l_i \leq g_i \leq u_i$ for some sets of lower and upper bounds l_i and u_i. If $l_i = 0$ and $u_i = \infty$ for all i, then the constraint $l_i \leq g_i \leq u_i$ coincides with the simple non-negativity constraint $g_i \geq 0$. In the majority of practical problems $l_i = l$ and $u_i = u$ for all i with $0 \leq l < 1 < u \leq \infty$, where the strict inequalities $l > 0$ and $u < \infty$ are very common.

In the process of calibration, the weights W have to stay as close as possible to the initial weights D. Equivalently, the g-weights G have to stay as close as possible to the vector $\mathbf{1}$. To measure the closeness of G and $\mathbf{1}$, we use some function $\Phi(G) = \Phi(g_1,\ldots,g_n)$. This function is required to satisfy the following properties (see [5] for a related discussion): (a) $\Phi(G) \geq 0\ \forall G$, (b) $\Phi(\mathbf{1}) = 0$, (c) $\Phi(G)$ is twice continuously differentiable, and (d) $\Phi(G)$ is strictly convex. The function Φ often has the form

$$\Phi(G) = \Phi(g_1,\ldots,g_n) = \sum_{i=1}^{n} q_i \phi_i(g_i)\,, \tag{1}$$

where $q_1,\ldots,q_n$ are given non-negative numbers; in the majority of applications $q_i = d_i$ for all i. We shall concentrate on this form of Φ; in Sect. 3, we discuss the choice of the functions ϕ_i.

Hard constraints $A'G = T$ enter the definition of the feasible domain of G. Soft constraints $A'G \simeq T$ can either enter the definition of the feasible domain of G in the form $\|A'G - T\| \leq \varepsilon$ for some vector norm $\|\cdot\|$ and some given $\varepsilon > 0$, or can be put as a penalty $\Psi(A'G,T)$ into the objective function. The properties required for Ψ (as a function of G) are similar to those required for Φ. The most common choice for Ψ is

$$\Psi(A'G,T) = \beta(A'G-T)'C(A'G-T) \tag{2}$$

where C is some user-specified $m \times m$ positive definite (usually, diagonal) matrix and $\beta > 0$ is some constant (see for example [2], equation (2.3)).

Summarizing, we have the following versions of the calibration problem formulated in terms of the g-weights G.

Hard constraint case:

$$\Phi(G) \to \min_{G \in \mathbb{G}}, \text{ where } \mathbb{G} = \{G : L \le G \le U \text{ and } A'G = T\}. \quad (3)$$

Soft constraint case I:

$$\Phi(G) \to \min_{G \in \mathbb{G}}, \text{ where } \mathbb{G} = \{G : L \le G \le U \text{ and } \Psi(A'G, T) \le 1\}. \quad (4)$$

Soft constraint case II:

$$\Phi(G) + \Psi(A'G, T) \to \min_{G \in \mathbb{G}}, \text{ where } \mathbb{G} = \{G : L \le G \le U\}. \quad (5)$$

In problems (3)–(5), the matrix A and the vectors T, L and U are given, and in the majority of applications the functions Φ and Ψ have the forms (1) and (2) correspondingly.

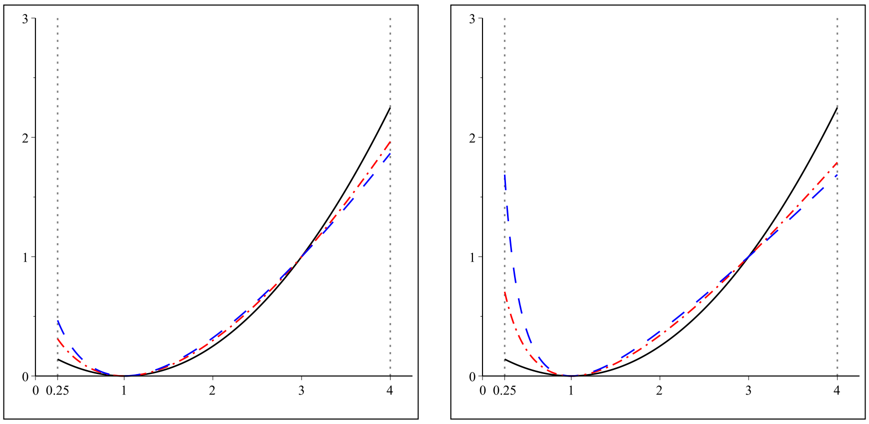
The optimization problems (3) and (4) may have no solutions; that is, the feasible domain $\mathbb{G}$ in these problems may be empty. The case when $\mathbb{G}$ is empty means that the constraints on G are too strong. The feasible domain $\mathbb{G}$ in the problem (5) is always non-empty and the optimal solution always exists. In view of the strict convexity of Φ and Ψ as well as the compactness of $\mathbb{G}$, if the optimal solution exists then it is necessarily unique. The optimization problem (4) is considered too difficult by practitioners and hence it is never considered (despite it looking rather natural). We therefore consider problems (3) and (5) only.

3 Choice of Functions ϕ_i in (1)

Here we discuss the choice of the functions ϕ_i in (1). See Sect. 4 for examples of calibrated weights obtained using different forms of functions ϕ_i. By slightly modifying the assumptions of [4], we require the function $\phi_i : (l_i, u_i) \to \mathbb{R}_+$ to satisfy the following properties: (i) $\phi_i(g) \ge 0$ for all $g \in (l_i, u_i)$, (ii) $\phi_i(1) = 0$, (iii) ϕ_i is twice continuously differentiable and strictly convex. The function ϕ_i does not have to be defined outside the open interval (l_i, u_i). If all ϕ_i satisfy the conditions (i)-(iii) then the function Φ defined in (1) satisfies the conditions (a)-(d) formulated above.

Since these functions are chosen in the same manner for all i, the subscript i will be dropped and the function ϕ_i will be denoted simply by ϕ. Correspondingly, the lower and upper bounds l_i and u_i for the g-weights g_i will be denoted by l and u respectively.

We will illustrate the shape of several functions ϕ in Figs. 1–3. In all these figures, we choose $l = 1/4, u = 4$ and plot all the functions in the interval $(l, u) = (\frac{1}{4}, 4)$, despite some of the functions are defined in a larger region. As our intention in this section is illustrating shapes of the possible calibration functions ϕ we thus plot scaled versions of these functions using appropriate multiples (so that different functions become visually comparable).



(a) $\phi^{(1)}$ (line), $\phi^{(2)}$ (dot-dash) and $\phi^{(3)}$ (dash) (b) $\phi^{(1)}$ (line), $\phi^{(4)}$ (dot-dash) and $\phi^{(5)}$ (dash)

Fig. 1 Classical calibration functions of Type I scaled so that $c_k\phi^{(k)}(3) = 1$, $k = 1,...,5$.

We distinguish the following two types of functions ϕ:

Type I $\phi(g)$ is defined for all g either in $\mathbb{R}$ or $\mathbb{R}_+ = (0,\infty)$ and does not depend on l and u.

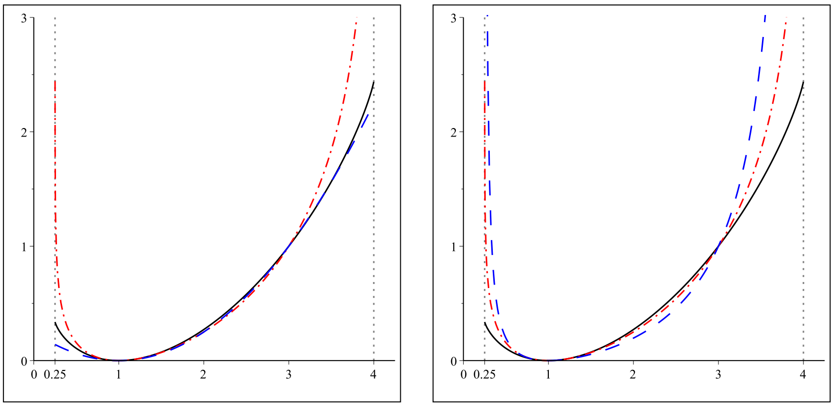
Type II $\phi(g)$ is defined for $g \in (l,u)$ but not outside the interval $[l,u]$. The functional form of g depends on l and u and hence we will use the notation $\phi(g;l,u)$ for the functions ϕ of this type.

The authors of the classical papers [4] and [5] suggest six choices for the function ϕ. Five of these are Type I and are: $\phi^{(1)}(g) = (g-1)^2$, $\phi^{(2)}(g) = g\ln g - g + 1$, $\phi^{(3)}(g) = (\sqrt{g}-1)^2$, $\phi^{(4)}(g) = -\ln g + g - 1$ and $\phi^{(5)}(g) = (g-1)^2/g$. Fig. 1 shows the shapes of these five functions.

The function $\phi^{(1)}$ is simply quadratic; in the literature on calibration it is usually referred to as the 'chi-square' function (see for example [14], equation (2.10)). It is by far the most popular in practice. The function $\phi^{(2)}$ is often referred to as the multiplicative or raking function in literature, (see for example [1]).

Many authors consider solving the optimization problem (3) without the constraint $L \le G \le U$. However, in this case using the function $\phi^{(1)}$ in the optimization may lead to extreme and negative weights. Whilst the function $\phi^{(2)}$, by the nature of its domain, only permits non-negative values for the optimized weights, the weights may still take very large values. This also applies to functions $\phi^{(3)}$, $\phi^{(4)}$ and $\phi^{(5)}$. The functions $\phi^{(3)}$, $\phi^{(4)}$ and $\phi^{(5)}$ have received much less attention in the literature on calibration.

The above criticism of the functions $\phi^{(1)}$–$\phi^{(5)}$ can be extended to all functions of Type I. Note that if we use the functions ϕ of Type I then the optimization problem (3) is an optimization problem with many variables and many constraints (recall than n is typically very large).



(a) $\phi^{(6)}$ (line) and $\phi^{(7)}$ (dot-dash) and $\phi^{(1)}$ (dash)

(b) $\phi^{(6)}$ (line), $\phi^{(7)}$ (dot-dash) and $\phi^{(8)}$ with $\alpha = 1$ (dash)

Fig. 2 Functions $\phi^{(1)}$ $\phi^{(6)}$, $\phi^{(7)}$ and $\phi^{(8)}$ scaled so that $c_1\phi^{(1)}(3) = 1$ and $c_k\phi^{(k)}(3; \frac{1}{4}, 4) = 1$, $k = 6, 7$ and $c_{8,1}\phi^{(8)}(3; \frac{1}{4}, 4, 1) = 1$.

Let us consider three functions ϕ of Type II:

$$\phi^{(6)}(g;l,u) = (g-l)\ln\left(\frac{g-l}{1-l}\right) + (u-g)\ln\left(\frac{u-g}{u-1}\right),$$

$$\phi^{(7)}(g;l,u) = (1-l)\ln\left(\frac{1-l}{g-l}\right) + (u-1)\ln\left(\frac{u-1}{u-g}\right), \quad (6)$$

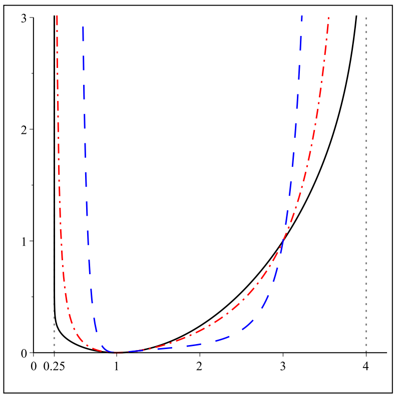
$$\phi^{(8)}(g;l,u,\alpha) = \frac{(g-1)^2}{[(u-g)(g-l)]^{\alpha}}, \quad \alpha > 0. \quad (7)$$

In Fig. 2, we plot the functions $c_1\phi^{(1)}(g)$, $c_6\phi^{(6)}(g; \frac{1}{4}, 4)$, $c_7\phi^{(7)}(g; \frac{1}{4}, 4)$ and $c_{8,1}\phi^{(8)}(g; \frac{1}{4}, 4, 1)$ with the constants c_1, c_6, c_7 and $c_{8,1}$ chosen so that $c_1\phi^{(1)}(3) = 1$, $c_k\phi^{(k)}(3; \frac{1}{4}, 4) = 1$ for $k = 6, 7$ and $c_{8,1}\phi^{(8)}(3; \frac{1}{4}, 4, 1) = 1$.

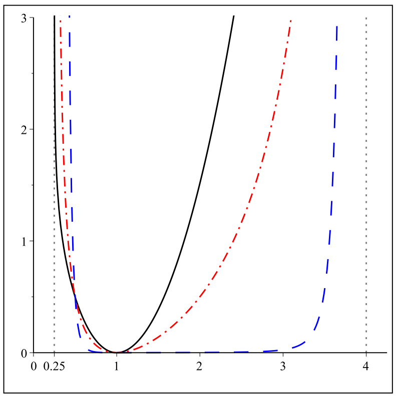
In Fig. 3, we plot function $\phi^{(8)}$ for various values of the parameter α. In Fig. 3(a), we choose the constants $c_{8,\alpha}$ so that $c_{8,\alpha}\phi^{(8)}(3; \frac{1}{4}, 4, \alpha) = 1$. In Fig. 3(b), we choose the constants $c_{8,\alpha}$ so that $c_{8,\alpha}\phi^{(8)}(\frac{1}{2}; \frac{1}{4}, 4, \alpha) = \frac{1}{2}$.

The function $\phi^{(6)}$ is defined on the closed interval $g \in [l, u]$ so that by continuity we have $\phi^{(6)}(l; l, u) = (u-l)\ln\frac{u-l}{u-1}$ and $\phi^{(6)}(u; l, u) = (u-l)\ln\frac{u-l}{1-l}$. The function $\phi^{(6)}(g; l, u)$ is not defined outside the interval $[l, u]$. Using this function in (1) creates difficulties for the algorithms that optimize the function (1) because of the discontinuity (and the loss of convexity) of $\phi^{(6)}(g; l, u)$ at $g = l$ and $g = u$. A way around this is the use of constrained optimization algorithms but then the criticism above directed to the functions of Type I can be extended to the function $\phi^{(6)}$.

The functions $\phi^{(7)}(g; l, u)$ and $\phi^{(8)}(g; l, u, \alpha)$ are derived by us. These two functions are defined only in the open interval $g \in (l, u)$ and tend to infinity as g tends



(a) $\phi^{(8)}$ scaled so that $c_{8,\alpha}\phi^{(8)}(3;l,u,\alpha) = 1$: $\alpha = 0.2$ (line), $\alpha = 1$ (dot-dash) and $\alpha = 5$ (dash)



(b) $\phi^{(8)}$ scaled so that $c_{8,\alpha}\phi^{(8)}(\frac{1}{2};l,u,\alpha) = \frac{1}{2}$: $\alpha = 0.2$ (line), $\alpha = 1$ (dot-dash) and $\alpha = 5$ (dash)

Fig. 3 Function $\phi^{(8)}(g;l,u,\alpha)$ for various values of α with $l = 1/4$ and $u = 4$.

to either l or u so that they can be classified as interior penalty functions. We have derived the expression for the function $\phi^{(7)}$ by applying a suitable transformation (including taking a logarithm) to the density of the Beta-distribution on $[0,1]$. The convexity of the function $\phi^{(7)}$ follows from the expression for its second derivative:

$$\frac{\partial^2\phi^{(7)}(g;l,u)}{\partial g^2} = \frac{(u-l)\left(g^2 - lu - 2g + l + u\right)}{(g-l)^2\,(u-g)^2} = \frac{(u-l)\left[(g-1)^2 + (u-1)(1-l)\right]}{(g-l)^2\,(u-g)^2}.$$

Since $0 < l < 1 < u < \infty$, this second derivative is positive for all $g \in (l,u)$ so that the function $\phi^{(7)}(g;l,u)$ is convex. The analytic forms of the functions $\phi^{(6)}$ and $\phi^{(7)}$ are very similar but we believe the properties of the function $\phi^{(7)}$ are much more attractive for the problem at hand than the properties of the function $\phi^{(6)}$.

For any $\alpha > 0$, the function $\phi^{(8)}$ has properties similar to the function $\phi^{(7)}$: it is defined in the open interval $g \in (l,u)$, it is convex in this interval, and it tends to infinity as $g \to l$ or $g \to u$. The function $\phi^{(8)}$ depends on an extra shape parameter α, see Fig. 3, so that the penalty for g deviating from 1 can be adjusted by the user.

A very important special case of the function $\phi^{(8)}$ occurs when $\alpha = 1$:

$$\phi^{(8)}(g;l,u,1) = \frac{(g-1)^2}{(u-g)(g-l)}. \tag{8}$$

The most attractive property of the function (8) is its invariance with respect to the change $g \leftrightarrow 1/g$ in the case $l = 1/u$ (which is a very common case in practice). Recall that $g = w/d$ is the ratio of the calibrated weight w to the initial weight d and therefore the multiplicative scale for measuring deviations of g from 1 is the most appropriate. This means that it is very natural to penalize g as much as $1/g$ for deviating from 1. Assuming $\alpha = 1$ and $l = 1/u$ we have:

$$\phi(g;u) = \phi^{(8)}(g;1/u,u,1) = \frac{(g-1)^2}{(u-g)(g-1/u)}.$$

For this function, we have $\phi(g;u) = \phi(1/g;u)$ so that this function possesses the additional property of equally penalizing g and $1/g$.

4 Hard Calibration

In Sect. 2, we introduced the calibration problem with both hard and soft constraints. In this section we consider the optimization problem (3), namely calibration with hard constraints. We shall refer to this class of calibration problems as hard calibration. For several examples, we shall compare the calibrated weights obtained using each of the functions considered in Sect. 3.

We solve the optimization problem (3) using the 'solnp' function within R's Rsolnp package (see [6]). Using this software, we directly solve the optimization problem (3) using the Augmented Lagrange Multiplier (ALM) method (see [9] for more details) for any choice of Type I or Type II function. For a comprehensive optimization software guide, see [12].

We consider two approaches to the hard calibration problem. The first of these is the classical approach considered in [4]. For this approach, the constraint $L \leq G \leq U$ is not included within the optimization. This means negative and extreme weights are in the domain of the feasible solution. This motivates the second approach that considers the optimization problem (3) including the constraint $L \leq G \leq U$. The classical approach can be considered a particular case of the the second approach, with L and U chosen to be vectors whose entries are $l = -\infty$ and $u = \infty$ respectively.

We remark that there are software packages that solve the calibration problem using an iterative Newton method as detailed in [5]. Examples of these include: the 'calib' function within R's sampling package (see [19]), the G-CALIB-S module within SPSS (see [20]) and the package CALMAR in SAS (see [4]). These packages allow the user to solve the hard calibration problem using the classical approach (no constraint $L \leq G \leq U$) for the functions $\phi^{(1)}$ and $\phi^{(2)}$. The packages also allow the user to solve the hard calibration problem including the constraint $L \leq G \leq U$ for functions $\phi^{(1)}$ and $\phi^{(6)}$ (see [5] for more details).

Many statistical offices throughout Europe use these packages to perform calibration. When comparing the weights obtained using direct optimization with the weights given by these packages, the answers in our examples were the same to within computer error (despite the running time was in some cases very different). Therefore, for the remainder of this paper, we only solve the optimization problem (3) using the ALM method.

To illustrate the case of negative and extreme weights, we consider the following example adapted from [8] using data from [3].

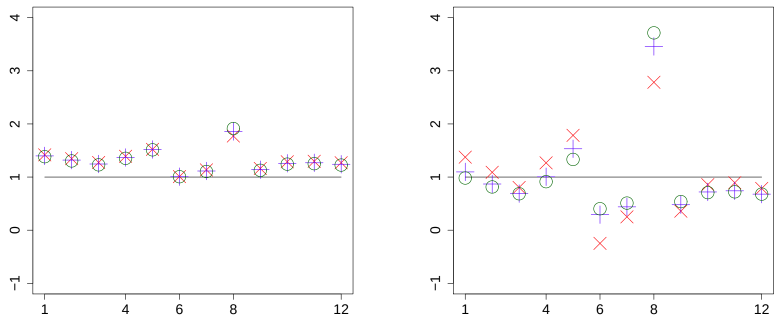
4.1 Example 1: A Classical Example

Throughout this example, we are working in units of thousands of people. Suppose we have a sample of $n = 12$ cities, sampled from 49 possible cities. We wish to weight our sample of cities appropriately to estimate the population total of the 49 cities.

For the 12 sampled cities, we know their size in 1920. Suppose we also know the population total of the 49 cities in 1920, namely $T = 5054$. We begin with the vector $G = \mathbf{1}$ and take the initial weights $D = (49/12, 49/12, \ldots, 49/12)'$. These initial weights are derived using the classical Horvitz-Thompson estimator [7].

Recall from Sect. 2, that the hard calibration constraint can be written in the form $X'W = T$ or equivalently $A'G = T$, with $a_{ij} = d_i x_{ij}$. We only have one auxiliary variable in this example, thus X and A reduce to 12×1 vectors. Suppose we are given the sample values for the auxiliary variable in the 12×1 vector X, where $X = (93, 77, 61, 87, 116, 2, 30, 172, 36, 64, 66, 60)'$. Note that in this case $X'D = A'\mathbf{1} = 3528 \neq 5054$. Therefore, for the initial weights $G = \mathbf{1}$, the constraint $A'G = T$ is not satisfied. This motivates the need to calibrate.

Figure 4 shows the g-weights obtained when optimizing (3) for functions $\phi^{(1)}$, $\phi^{(2)}$ and $\phi^{(3)}$ using classical hard calibration (recall L and U are taken as vectors whose entries are $-\infty$ and ∞ respectively). We consider the case $q_i = d_i$ in (1). Figure 4(a) shows the calibrated weights when we do not impose the constraint $\mathbf{1}'G = 12$. The calibrated weights obtained when we impose the constraint $\mathbf{1}'G = 12$ are shown in Fig. 4(b).



(a) g-weights for functions $\phi^{(1)}$ (cross), $\phi^{(2)}$ (plus) and $\phi^{(3)}$ (circle) with the sum of weights unconstrained

(b) g-weights for functions $\phi^{(1)}$ (cross), $\phi^{(2)}$ (plus) and $\phi^{(3)}$ (circle) with the sum of weights constrained

Fig. 4 Comparison of g-weights with $\mathbf{1}$ (line) for the functions $\phi^{(1)}$, $\phi^{(2)}$ and $\phi^{(3)}$.

For these functions, observe that when we do not impose the constraint $\mathbf{1}'G = 12$, all the weights increase from, or remain at, their initial value of 1. It can be verified that the calibrated weights for each of these functions satisfy the constraint $A'G = T$. We remark that $\mathbf{1}'G = 15.883$ for the calibrated weights using function $\phi^{(1)}$, $\mathbf{1}'G = 15.738$ for the calibrated weights using function $\phi^{(2)}$ and $\mathbf{1}'G = 15.653$ for the calibrated weights using function $\phi^{(3)}$; in all cases $\mathbf{1}'G > 12$ due to the calibrated weights being larger than the initial weights of 1.

Imposing the extra constraint $\mathbf{1}'G = 12$ results in weights that are distributed both above and below the initial g-weights of $\mathbf{1}$. One of the g-weights for function $\phi^{(1)}$ (indexed 6 in Figure 4(b)) is negative, whilst the weight indexed 8 has taken a large value in comparison to the other g-weights. For functions $\phi^{(2)}$ and $\phi^{(3)}$, we do not have a negative weight at index 6, however the value of the weight at index 8 is still large in comparison with the other weights. Thus, whilst functions $\phi^{(2)}$ and $\phi^{(3)}$ prevent negative weights, they do not prevent large positive weights.

We remark that the behaviour of the weights for functions $\phi^{(4)}$ and $\phi^{(5)}$ is very similar to that for functions $\phi^{(2)}$ and $\phi^{(3)}$. Plots of the weights comparing functions $\phi^{(1)}$, $\phi^{(4)}$ and $\phi^{(5)}$ are very similar to the plots in Figs. 4(a) and 4(b). Hence we do not plot the weights for functions $\phi^{(4)}$ and $\phi^{(5)}$ here.

To overcome the issue of negative and extreme weights, we include constraint $L \leq G \leq U$, where L and U have entries l and u respectively with $0 \leq l < 1 < u \leq \infty$. Any feasible solution to this problem is guaranteed to be within the bounds pre-specified by the user. However, recall from Sect. 2 that the feasible solution of this problem may be empty depending on the choice of L and U.

Returning to the example, suppose the calibrated weights G must satisfy the bounds $L \leq G \leq U$ where $L = (l, l, ..., l)'$ and $U = (u, u, ..., u)'$ are both 12×1 vectors. Consider the particular case of $l = \frac{12}{49}$ and $u = \frac{120}{49}$. This means the g-weights g_i will be bounded between the lower bound of $\frac{12}{49}$ and the upper bound of $\frac{120}{49}$, whilst the weights w_i will be bounded between the lower bound of $ld_i = 1$ and the upper bound of $ud_i = 10$ for all i.

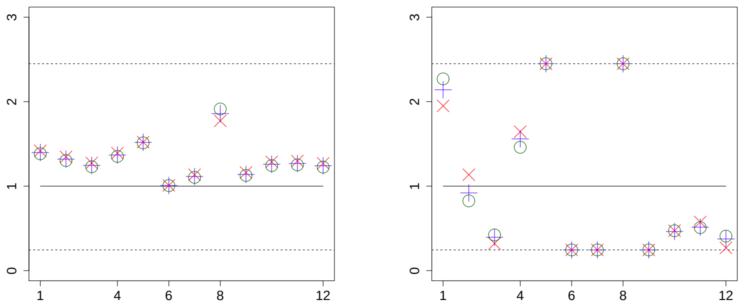
Figure 5 shows the g-weights obtained by optimizing (3) for functions $\phi^{(1)}$, $\phi^{(2)}$ and $\phi^{(3)}$. Figure 5(a) shows the calibrated weights when we do not impose the constraint $\mathbf{1}'G = 12$. Figure 5(b) shows the calibrated weights when we include this constraint.

For the weights in Fig. 5(a), we observe that imposing the constraint $\mathbf{1}'G = 12$ results in all the weights increasing from, or remaining at, their initial value of 1. The weights in Fig. 5(a) are identical to those in Fig. 4(a).

However, in Fig. 5(b), we see that imposing the extra constraint $\mathbf{1}'G = 12$ results in weights that are at, or very close to, the upper and lower bounds u and l respectively. The weights in Fig. 5(b) are different to those in Fig. 4(b).

In this case, the behaviour of the weights for functions $\phi^{(4)}$ and $\phi^{(5)}$ is very similar to that for functions $\phi^{(2)}$ and $\phi^{(3)}$, both with and without the constraint $\mathbf{1}'G = 12$ included in the optimization. Hence we do not plot the weights for these functions here.

Recall the relationship $w_i = d_i g_i$. Since the vector of initial weights D is given, and we have calculated the g-weights, we can compute the weights w_i. Computing the weights w_i for function $\phi^{(1)}$ from the corresponding g-weights in Fig. 5(b) gives the same weights as those derived in [8].

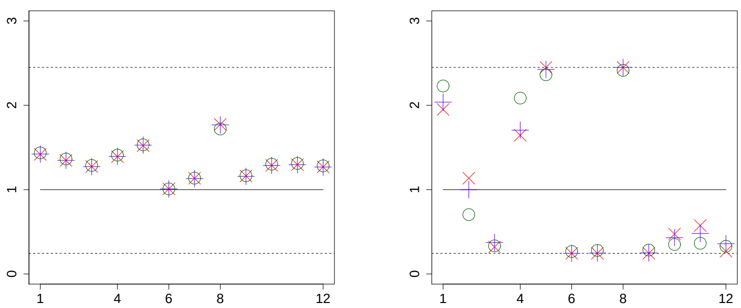


(a) g-weights for functions $\phi^{(1)}$ (cross), $\phi^{(2)}$ (plus) and $\phi^{(3)}$ (circle) with the sum of weights unconstrained

(b) g-weights for functions $\phi^{(1)}$ (cross), $\phi^{(2)}$ (plus) and $\phi^{(3)}$ (circle) with the sum of weights constrained

Fig. 5 Comparison of g-weights with $\mathbf{1}$ (line) for the functions $\phi^{(1)}$, $\phi^{(2)}$ and $\phi^{(3)}$, dotted lines indicate the upper and lower bounds.

Figure 6 shows the g-weights obtained by optimizing (3) for the functions $\phi^{(1)}$, $\phi^{(6)}$ and $\phi^{(7)}$. Figure 6(a) shows the calibrated weights when we do not impose the constraint $\mathbf{1}'G = 12$. Figure 6(b) shows the calibrated weights when the constraint is included within the optimization.

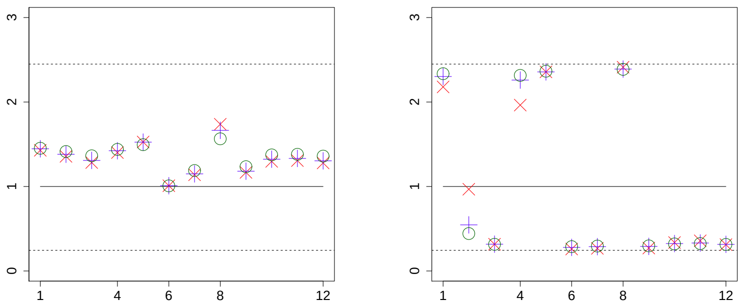


(a) g-weights for functions $\phi^{(1)}$ (cross), $\phi^{(6)}$ (plus) and $\phi^{(7)}$ (circle) with the sum of weights unconstrained

(b) g-weights for functions $\phi^{(1)}$ (cross), $\phi^{(6)}$ (plus) and $\phi^{(7)}$ (circle) with the sum of weights constrained

Fig. 6 Comparison of g-weights with $\mathbf{1}$ (line) for functions $\phi^{(1)}$, $\phi^{(6)}$ and $\phi^{(7)}$, dotted lines indicate the upper and lower bounds.

Figure 7 shows the g-weights obtained by optimizing (3) for function $\phi^{(8)}$ with α chosen to be 0.2, 1 and 5. Figure 7(a) shows the calibrated weights when we do not impose the constraint $\mathbf{1}'G = 12$. Figure 7(b) shows the calibrated weights when we include this constraint within the optimization.



(a) g-weights for functions $\phi^{(8)}$ with $\alpha = 0.2$ (cross), $\alpha = 1$ (plus) and $\alpha = 5$ (circle) with the sum of weights unconstrained

(b) g-weights for function $\phi^{(8)}$ with $\alpha = 0.2$ (cross), $\alpha = 1$ (plus) and $\alpha = 5$ (circle) with the sum of weights constrained

Fig. 7 Comparison of g-weights with **1** (line) for function $\phi^{(8)}$, dotted lines indicate the upper and lower bounds.

Observe that when the constraint $\mathbf{1}'G = 12$ is not imposed, the weights all increase or remain at the initial values of 1. When the constraint is imposed, we see that the weights are distributed both above and below the initial values of 1, with several weights clustered at the bounds.

In summary, we have seen that not imposing the constraint $\mathbf{1}'G = 12$ results in calibrated weights exhibiting less variability than the calibrated weights obtained including the constraint. For this example, the calibrated weights all increased from the initial values of 1 but did not exhibit any extremal behaviour, lying well within the considered bounds. However, including the constraint $\mathbf{1}'G = 12$ gave calibrated weights that were more variable and likely to move towards the boundaries.

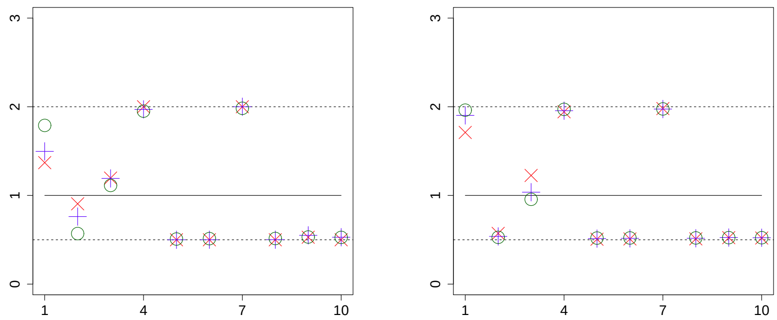
For the remaining examples in this paper, we shall explore the effects the choice of L and U have on the calibrated weights G. In all the examples we will include the constraint $\mathbf{1}'G = n$, and take $q_i = d_i$ in (1).

4.2 Example 2

Suppose we are given the vector $X = (93, 77, 87, 116, 2, 30, 172, 36, 64, 60)'$ and the 10×1 vector of initial weights $D = (4, 4, \ldots, 4)'$. The parameter value $T = 3900$ is assumed known. Recall that we impose the upper and lower bounds $U = (u, u, \ldots, u)'$ and $L = (l, l, \ldots, l)'$, where U and L are both 10×1 vectors whose entries are u and

l respectively. Consider the case $l = 1/u$. We wish to find the smallest value of u such that the optimization problem (3) has a feasible solution. In this example, experimentation gave the smallest value of u as approximately 2.0.

In Fig. 8 we plot the calibrated weights when we take $l = 1/2$ and $u = 2$. In this case, solving the optimization problem (3) for functions $\phi^{(1)}$, $\phi^{(6)}$ and $\phi^{(7)}$ gives the weights in Fig. 8(a). Figure 8(b) shows the weights for function $\phi^{(8)}$ with $\alpha = 0.2$, $\alpha = 1$ and $\alpha = 5$.



(a) Weights obtained for $\phi^{(1)}$ (cross), $\phi^{(6)}$ (plus) and $\phi^{(7)}$ (circle)

(b) Weights obtained for $\phi^{(8)}$ with $\alpha = 0.2$ (cross), and $\alpha = 1$ (plus) and $\alpha = 5$ (circle)

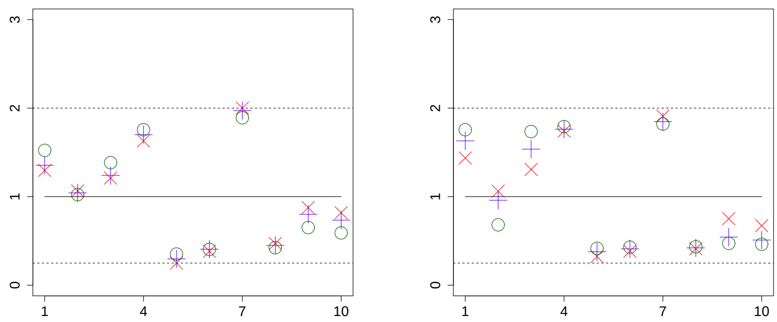
Fig. 8 Comparison of weights for functions $\phi^{(1)}$, $\phi^{(6)}$, $\phi^{(7)}$, and $\phi^{(8)}$ for various α with $l = 1/2$ and $u = 2$ (dotted lines indicate bounds).

For this example, a feasible solution to the problem (3) exists for the (approximate) bounds $0 \leq l \leq 1/2$ and $u \geq 2$. Let us consider the effect of changing the values of l and u.

Figure 9 shows the calibrated weights when $l = 1/4$ and $u = 2$. In Fig. 9(a) we plot the weights for functions $\phi^{(1)}$, $\phi^{(6)}$ and $\phi^{(7)}$ whilst in Fig. 9(b) we plot the weights for function $\phi^{(8)}$ with $\alpha = 0.2$, $\alpha = 1$ and $\alpha = 5$. We see that reducing the lower bound results in less weights taking values at the lower bound. Generally, the calibrated weights for function $\phi^{(8)}$ appear to move towards the boundaries more than the weights obtained for functions $\phi^{(1)}$, $\phi^{(6)}$ and $\phi^{(7)}$.

We now consider the effect of increasing u. In Fig. 10, we keep $l = 1/4$ and consider the calibrated weights when $u = 4$. In Fig. 10(a) we plot the calibrated weights for the functions $\phi^{(1)}$, $\phi^{(6)}$ and $\phi^{(7)}$ whilst in Fig. 10(b) we plot the calibrated weights for function $\phi^{(8)}$ with $\alpha = 0.2$, $\alpha = 1$ and $\alpha = 5$. We see that increasing the upper bound has resulted in some of the weights increasing slightly in comparison to the weights in Fig. 9. However, there are no weights on the upper bound.

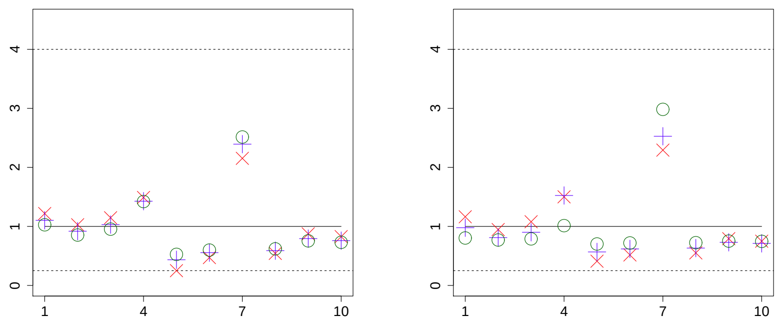
To conclude, this example has shown that taking $l = 1/u$ and minimizing the value of u such that the calibration problem (3) has a feasilbe solution often results



(a) Weights obtained for $\phi^{(1)}$ (cross), $\phi^{(6)}$ (plus) and $\phi^{(7)}$ (circle)

(b) Weights obtained for $\phi^{(8)}$ with $\alpha = 0.2$ (cross), and $\alpha = 1$ (plus) and $\alpha = 5$ (circle)

Fig. 9 Comparison of weights for functions $\phi^{(1)}$, $\phi^{(6)}$, $\phi^{(7)}$, and $\phi^{(8)}$ for various α with $l = 1/4$ and $u = 2$ (dotted lines indicate bounds)



(a) Weights obtained for $\phi^{(1)}$ (cross), $\phi^{(6)}$ (plus) and $\phi^{(7)}$ (circle)

(b) Weights obtained for $\phi^{(8)}$ with $\alpha = 0.2$ (cross), and $\alpha = 1$ (plus) and $\alpha = 5$ (circle)

Fig. 10 Comparison of weights for functions $\phi^{(1)}$, $\phi^{(6)}$, $\phi^{(7)}$, and $\phi^{(8)}$ for various α with $l = 1/4$ and $u = 4$ (dotted lines indicate bounds)

in many of the weights taking values at the boundaries. Increasing the value of u gives extra freedom to the weights and, as a result, there are typically less weights at the boundaries.

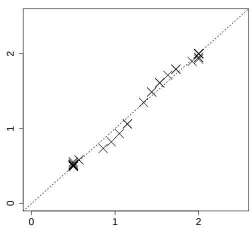
In the remaining two examples, we only consider the smallest value of u for which the optimization problem (3) has a feasible solution when $l = 1/u$. We further explore the phenomenon of weights clustering at the boundary and investigate whether different functions are more or less likely to give weights that approach the boundaries.

4.3 Example 3

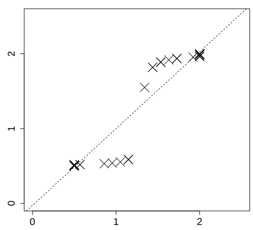
Suppose we are given the 100×1 vector of initial weights $D = (5,...,5)'$ and suppose that $T = 49500$. The vector of auxiliary values X is formed by extending the auxiliary vector from Example 4.2. We form a 100×1 vector that has the values from the auxiliary vector in Example 4.2 as its first ten entries. The next ten entries are formed by taking the auxiliary vector from Example 4.2 and adding 2 to each value. In a similar way, we subtract 3 from each value to give the next 10 values. In a similar way, we then repeat the vector, add 4 to all the entries, add 3 to all the entries, subtract 1, subtract 2, repeat the vector and finally add 4 to give the remaining 70 values.

We impose the upper and lower bounds $U = (u,u,...,u)'$ and $L = (l,l,...,l)'$, where L and U are both 100×1 vectors whose entries are u and $l = 1/u$ respectively. For this example, experimentation gives the smallest value of u as approximately $u = 2$ and so $l = 1/2$.

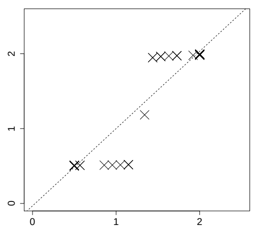
In Fig. 11, we compare the calibrated weights for functions $\phi^{(6)}$, $\phi^{(7)}$ and $\phi^{(8)}$ with those for function $\phi^{(1)}$. In Fig. 11(a), we observe that most of the points in the scatterplot are on the diagonal. This indicates the similarity of the weights for functions $\phi^{(1)}$ and $\phi^{(6)}$. However, in Fig. 11(b), we observe that there are fewer weights on the diagonal. This indicates that, for function $\phi^{(7)}$, more of the weights approach the boundary. In Fig. 11(c), we see this even more clearly with a distinct band of weights at the upper and lower bounds of 2 and $\frac{1}{2}$ for function $\phi^{(8)}$, compared with the weights for $\phi^{(1)}$ that are more evenly distributed between the upper and lower bounds.



(a) Weights for function $\phi^{(1)}$ against $\phi^{(6)}$



(b) Weights for function $\phi^{(1)}$ against $\phi^{(7)}$



(c) Weights for function $\phi^{(1)}$ against $\phi^{(8)}$

Fig. 11 Comparison of weights for function $\phi^{(1)}$ against functions $\phi^{(6)}$, $\phi^{(7)}$ and $\phi^{(8)}$, with $l = 1/2$ and $u = 2$

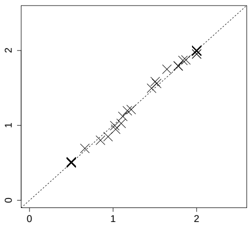
For the next example, we keep the sample size at $n = 100$ and increase the number of auxiliary variables to $m = 3$.

4.4 Example 4

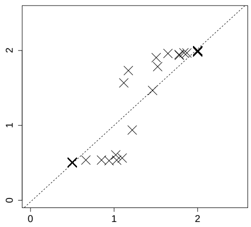
Suppose we are given a 100×1 vector of initial weights $D = (5, 5, ..., 5)'$, and let $T = (49500, 49540, 41000)'$. Suppose that the 100×3 matrix of auxiliary values X is defined as follows: for the first column of X we take the auxiliary vector from Example 3 in Sect. 4.3. For the second column of X, we form a 100×1 vector whose first ten values are formed by taking the auxiliary vector in Example 4.2 and subtracting 1. The next ten entries are formed by adding one to each of the values of the auxiliary vector in Example 4.2. In a similar way, we subtract 2 from each value to give the next 10 values, then repeat the vector, add 5 to all the entries, repeat the vector twice, subtract 1, add 1 and finally add 3 to give the remaining 70 values. For the third column, we take 100 values generated at random from a Normal distribution with mean 80 and standard deviation 48 (these are similar to the mean and standard deviations for the other columns).

We impose the upper and lower bounds $U = (u, u, ..., u)'$ and $L = (l, l, ..., l)'$, where L and U are both 100×1 vectors whose entries are u and $l = 1/u$ respectively. For this example, experimentation gives the smallest value of u as approximately $u = 2$, and so $l = 1/2$.

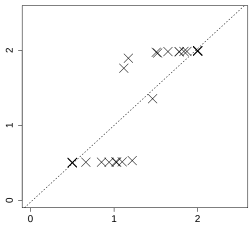
In Fig. 12, we compare the calibrated weights using function $\phi^{(1)}$ with the calibrated weights for functions $\phi^{(6)}$, $\phi^{(7)}$ and $\phi^{(8)}$ $(\alpha = 1)$.



(a) Weights for function $\phi^{(1)}$ against $\phi^{(6)}$



(b) Weights for function $\phi^{(1)}$ against $\phi^{(7)}$



(c) Weights for function $\phi^{(1)}$ against $\phi^{(8)}$ with $\alpha = 1$

Fig. 12 Comparison of weights for the function $\phi^{(1)}$ against $\phi^{(6)}$, $\phi^{(7)}$ and $\phi^{(8)}$ $(\alpha = 1)$

Figure 12 has many similarities with Fig. 11 in Example 4.3. We observe that the weights for functions $\phi^{(1)}$ and $\phi^{(6)}$ are very similar. However, the calibrated weights for functions $\phi^{(7)}$ and $\phi^{(8)}$ show clear differences to the calibrated weights for function $\phi^{(1)}$. Again, we observe the distinct band of weights at the upper and lower bounds of 2 and $\frac{1}{2}$ for functions $\phi^{(7)}$ and $\phi^{(8)}$, compared with the weights for functions $\phi^{(1)}$ and $\phi^{(6)}$ that are more evenly distributed between the upper and lower bounds.

We now compare the CPU times taken to obtain the weights in Fig. 12. These CPU times were computed on a computer with an Intel(R) Core(TM) i7-4500U CPU

Processor with 8GB of RAM. The CPU times are given in Table 1. We observe that the CPU times for functions $\phi^{(7)}$ and $\phi^{(8)}$ ($\alpha = 0.2$) are less than those for the classical functions $\phi^{(1)}$ and $\phi^{(6)}$. CPU time is related to the complexity of the optimization problem, see [13] on a comprehensive discussion of how to measure numerical complexity of an optimization problem.

Table 1 CPU times for various functions ϕ in solving the optimization problem (3) in Example 4

Function	CPU Time (seconds)
$\phi^{(1)}$	0.609
$\phi^{(6)}$	0.734
$\phi^{(7)}$	0.544
$\phi^{(8)}$ ($\alpha = 0.2$)	0.569
$\phi^{(8)}$ ($\alpha = 1$)	0.559

In these examples, we have seen that the problem (3) does not necessarily have a feasible solution for all choices of the vectors L and U. We address this issue in the next section by introducing soft calibration.

5 Soft Calibration

In this section, we consider the optimization problem (5). Recall that this requires a choice of the functions Φ and Ψ. In this section, we choose Φ to be of the form (1) with ϕ taken to be $\phi^{(1)}$, and consider the penalty function Ψ of the form (2). We do not consider other choices of Φ or Ψ in this section.

Re-writing the problem (5) with our choice of Φ and Ψ gives the following optimization problem:

$$\sum_{i=1}^{n} q_i(g_i - 1)^2 + \beta(A'G - T)'C(A'G - T) \to \min_{G \in \mathbb{G}}, \tag{9}$$

where $\mathbb{G} = \{G : L \leq G \leq U\}$, $q_1, ..., q_n$ are given non-negative numbers, C is a user-specified $m \times m$ positive definite (usually diagonal) matrix and $\beta > 0$ is some constant.

In Sect. 4, we considered two approaches to solving the hard calibration problem (3). We now consider two similar approaches for solving the problem (9). The first approach is the classical soft calibration approach (see, for example, [2]). In this approach, the constraint $L \leq G \leq U$ is not included within the optimization. Practitioners vary the value of the parameter β so that the weights are within some pre-specified bounds. The second approach is to include the constraint $L \leq G \leq U$ within the optimization algorithm, i.e. to solve the optimization problem (5). We remark that classical soft calibration is a special case of the second approach where L and U are vectors whose entries are $-\infty$ and ∞ respectively.

For the example in Sect. 4.1, we considered the calibrated weights obtained when solving the optimization problem (3) without imposing the constraint $L \leq G \leq U$. In this case, we saw that it is possible to obtain negative and extreme weights.

The classical soft calibration problem was proposed as a way to deal with these negative and extreme weights. Classical soft calibration allows an analytic solution to be found to the optimization problem (9). Let $\mathbb{D}$ be an $n \times n$ diagonal matrix, whose entries are the weights $d_1, d_2, \ldots d_n$. Furthermore, take $q_i = d_i$ and let $\gamma = \frac{1}{\beta}$. Then, for the classical soft calibration approach, the analytic form of the weights that satisfy the optimization problem (9) is given by

$$G = \mathbf{1} + A\left(A'\mathbb{D}^{-1}A + \gamma C^{-1}\right)^{-1}\left(T - A'\mathbf{1}\right). \tag{10}$$

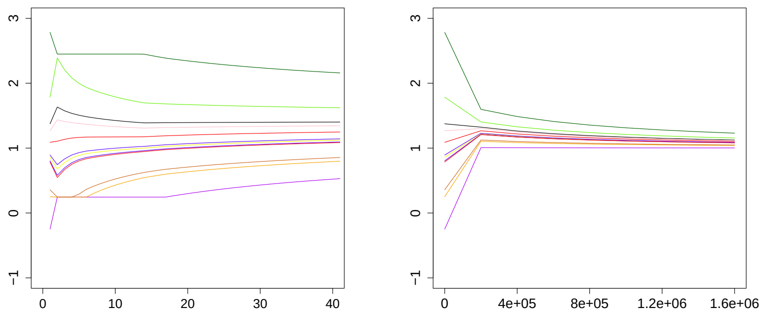
This is an equivalent formulation of equation (2.4) from [2], expressed in terms of g-weights. The term $\left(A'\mathbb{D}^{-1}A + \gamma C^{-1}\right)^{-1}$ is similar to the inverse matrix term in ridge regression (see, for example, [15]).

Let us consider the effect of changing the parameter β in (9). Recall that $\gamma = 1/\beta$ or equivalently $\beta = \gamma^{-1}$. We consider the effect of changing the parameter γ. As γ tends to zero, γ^{-1} tends to infinity and so the optimization problem (9) reduces to minimising $(A'G - T)'C(A'G - T)$ for $G \in \mathbb{G}$. As this term is quadratic, the minimum occurs when $A'G - T = 0$ or equivalently $A'G = T$. This is the hard calibration constraint. Therefore, the case $\gamma \to 0$ corresponds to the solving the hard calibration problem (3). We remark that this is consistent with (10), since taking $\gamma = 0$ in this formula gives the expression for the g-weights in classical hard calibration.

As γ tends to infinity, γ^{-1} tends to zero and so the term $(A'G - T)'C(A'G - T)$ becomes negligible. This results in the optimization (9) reducing to the problem of minimising $\Phi(G) = \sum_{i=1}^{n} q_i \phi^{(1)}(g_i)$ for $G \in \mathbb{G}$, which is minimised at $G = \mathbf{1}$ (by definition of the function Φ). Again, this is consistent with (10), since when $\gamma \to \infty$ the term $A\left(A'\mathbb{D}^{-1}A + \gamma C^{-1}\right)^{-1}(T - A'\mathbf{1})$ tends to zero giving $G = \mathbf{1}$.

To illustrate this, let us re-visit the example of Sect. 4.1. Recall that $T = 5054$, $D = (49/12, 49/12, \ldots, 49/12)'$ and $X = (93, 77, 61, 87, 116, 2, 30, 172, 36, 64, 66, 60)'$. In Fig. 13, we plot the weights given by (10) as the value of γ varies. We take $C = I_m$, where I_m denotes the $m \times m$ identity matrix. Figure 13(a) plots the weights for values of γ from 0 to 40. This plot confirms our earlier assertions that as $\gamma \to 0$, G tends to the classical hard calibration weights. Figure 13(b) plots the weights for values of γ between 0 and 1.6×10^6. This plot confirms that as $\gamma \to \infty$, the g-weights tend to their initial values of 1.

When obtaining the explicit solution, (10), to the classical soft calibration problem, we did not specify any constraints on the weights G. Suppose that we wish to impose the constraint $L \leq G \leq U$. Observe from Fig. 13(a) that as the value of γ increases, the range of the weights decreases. In classical soft calibration, having obtained the analytic solution (10) for the calibrated weights, the approach to satisfying the constraint $L \leq G \leq U$ is to choose the smallest value of γ for which the weights in (10) are within the specified bounds. Clearly, the value of γ that satisfies the constraints $L \leq G \leq U$ is sample dependent.



(a) Soft weights for γ between 0 and 40 (b) Soft weights for γ from 0 to 1.6×10^6

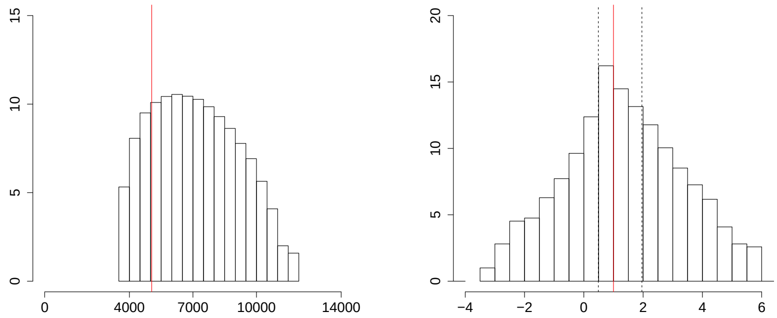
Fig. 13 Plots of classical soft calibration weights (10) as a function of γ

Consider again Example 1 from Sect. 4.1. We previously saw that, in the case of classical hard calibration, we obtain negative and extreme weights for this sample. Suppose we wish to impose the lower and upper bounds of $l = 12/49$ and $u = 120/49$. We saw that we were able to satisfy these bounds by solving the problem (3). In order to satisfy these bounds for classical soft calibration, experimentation gives the smallest value of γ as approximately $\gamma = 9.0$ in order to find a solution that lies between these bounds. This is a relatively large value of γ.

Note that in this case we have $\mathbf{1}'G = 13.527 \neq 12$ and $A'G = 5053.899 \neq 5054$, therefore our constraints $\mathbf{1}'G = 12$ and $A'G = T$ are no longer satisfied. Having relaxed these constraints in the soft calibration penalty (2), the larger the value of γ, the smaller the value of β and the less importance we assign to the penalty (2) in (9). This allows greater variation between $A'G$ and T and between $\mathbf{1}'G$ and 12. However, for large values of γ there is less variation in the weights. In contrast, for small values of γ, the penalty (2) is given more importance allowing less variation between $A'G$ and T and between $\mathbf{1}'G$ and 12. However, in this case there will be greater variability in the weights.

We illustrate this in Figs. 14 and 15. To produce these figures, we took 10,000 simple random samples of size 12 from the data in [3]. Figure 14 shows the distribution of weights and values of $A'G$ when we take $\gamma = 0.1$. Figure 15 shows the distribution of weights and values of $A'G$ when we take $\gamma = 9$, as required for this example to ensure the weights are between L and U. We observe that although $\gamma = 9$ gave g-weights satisfying the bounds $L \leq G \leq U$ for one sample, this value of γ does not guarantee that the g-weights will satisfy these bounds for every sample.

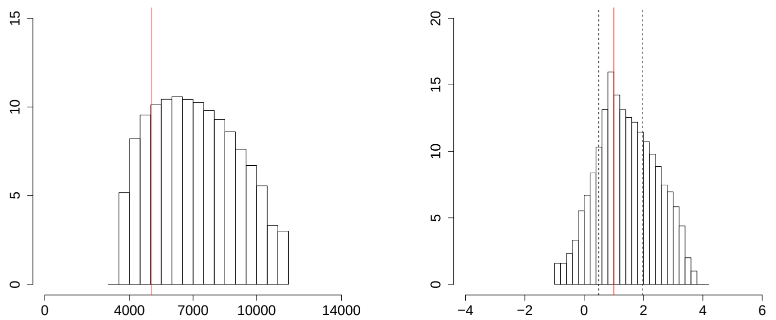
Let us now consider the second approach of directly optimizing (5). As stated in Sect. 2, the optimization problem (5) has a solution for any value of $\beta > 0$. Therefore, given any L and U, we can find a solution to the optimization problem (5) independent of the choice of β. That is what makes this approach different to classical soft calibration.



(a) $A'G$ for 10000 random samples of size 12, vertical line at 5054 ($A'G = 5054$ is hard constraint)

(b) g-weights for 10000 random samples of size 12, vertical line at 1 (initial weights), dashed lines indicate bounds

Fig. 14 Plots of $A'G$ and g-weights that satisfy the optimization problem (10) for $\gamma = 0.1$



(a) $X'W$ for 10000 random samples of size 12, vertical line at 5054 ($X'W = 5054$ is hard constraint)

(b) g-weights for 10000 random samples of size 12, vertical line at 1 (initial weights), dashed lines indicate bounds

Fig. 15 Plots of $A'G$ and g-weights that satisfy the optimization problem (10) for $\gamma = 0.1$

Let us return again to Example 1 from Sect. 4.1. Consider the problem (5) with $L \leq G \leq U$ where $L = (l,\ldots,l)'$ and $U = (u,\ldots,u)'$ are 12×1 vectors with entries $l = \frac{12}{49}$ and $u = \frac{120}{49}$ respectively. We know that small values of γ give a solution that is close to the hard calibration solution. Taking $\gamma = 0.01$, we obtain soft calibration weights that are very similar to those derived for hard calibration in Sect. 4.1. Therefore, in this instance, solving the problem (5) has little advantage over solving the corresponding hard calibration problem (3).

However, suppose we want to impose the bounds $l = 24/49$ and $u = 96/49$, corresponding to bounding the weights w_i between the lower and upper bounds of 2 and 8 respectively. In this case, there is no feasible solution to the hard calibration problem (3). Solving this problem using classical soft calibration requires a value of $\gamma = 16$ to ensure that the weights are between these bounds.

We now consider the direct optimization approach. Recall that for small values of γ, the solution to the problem (5) is approximately equal to the solution to the problem (3). Assuming we have the lower bounds $l = 24/49$ and $u = 96/49$, taking $\gamma = 10^{-9}$ we obtain weights G such that $A'G = 5053.910$ and $\mathbf{1}'G = 13.435$. Under hard calibration, we would require $A'G = 5054$ and $\mathbf{1}'G = 12$. We have almost satisfied the constraint $A'G = 5054$, however we have not satisfied the constraint $\mathbf{1}'G = 12$. This suggests that the condition $\mathbf{1}'G = 12$ was too restrictive.

Conclusions

The problem of calibrating weights in surveys is a very important practical problem. In the literature on calibration, there are many recipes but no clear understanding of what calibration is. In this paper, we have formally formulated the calibration problem as an optimization problem and defined the desired conditions for the components of the objective function and feasible region. We have demonstrated that the commonly used calibration criteria do not fully satisfy the desired criteria. The corresponding optimization problems are not flexible enough, harder than they have to be, or have some common recipes leading to wrong and contradictory recommendations. An example of the latter is the use of ridge estimators for trying to achieve positivity of the calibrated weights, see Sect. 5.

We have studied the influence of the function ϕ, the main component of objective function, on the complexity of the optimization problem and the final solution. We claim that the new functions $\phi^{(7)}$ and $\phi^{(8)}$ suggested in this paper are much more transparent and more flexible than the functions adopted in the standard calibration literature and classical calibration software packages. The functions suggested by us lead to easier optimization problems as they automatically take into account the constraint $L \leq G \leq U$. This could be of high importance in practice as the dimension of the problem (which is the size of the sample) may be very large.

In the case of large samples, one of our recommendations is to replace the hard calibration problem defined by (1) and (3) with a soft calibration problem defined by (1), (2) and (5), where β in (2) is large and the functions ϕ_i in (1) are either $\phi_i^{(7)}$ or $\phi_i^{(8)}$, see (6) and (7) respectively. In doing so we replace a potentially difficult constrained optimization problem (3) with a much simpler problem (5), which is an unconstrained convex optimization problem (recall that all constraints in (5) are taken into account due to a clever choice of the functions ϕ_i). If β is large then the solution of this problem is guaranteed to be very close to the solution of the original problem (3).

References

1. Bankier, M., Houle, A.M., Luc, M.: Calibration estimation in the 1991 and 1996 Canadian censuses. In: Proceedings of the Survey Research Methods Section, pp. 66–75 (1997)
2. Bocci, J., Beaumont, C.: Another look at ridge calibration. Metron **66**(1), 5–20 (2008)
3. Cochran, W.G.: Sampling techniques. John Wiley & Sons, New York (1977)
4. Deville, J.C., Särndal, C.E.: Calibration estimators in survey sampling. Journal of the American Statistical Association **87**(418), 376–382 (1992)
5. Deville, J.C., Särndal, C.E., Sautory, O.: Generalized raking procedures in survey sampling. Journal of the American Statistical Association **88**(423), 1013–1020 (1993)
6. Ghalanos, A., Theussl, S.: Rsolnp: General non-linear optimization. R package version (2010)
7. Horvitz, D.G., Thompson, D.J.: A generalization of sampling without replacement from a finite universe. Journal of the American Statistical Association **47**(260), 663–685 (1952)
8. Huang, E., Fuller, W.: Nonnegative regression estimation for sample survey data. In: Proceedings of the Social Statistics Section, vol. 21, pp. 300–305. American Statistical Association (1978)
9. Ito, K., Kunisch, K.: Augmented lagrangian methods for nonsmooth, convex optimization in Hilbert spaces. Nonlinear Analysis: Theory, Methods & Applications **41**(5), 591–616 (2000)
10. Kott, P.S.: Using calibration weighting to adjust for nonresponse and coverage errors. Survey Methodology **32**(2), 133–142 (2006)
11. Lundström, S., Särndal, C.E.: Calibration as a standard method for treatment of nonresponse. Journal of Official Statistics **15**, 305–327 (1999)
12. More, J.J., Wright, S.J., Pardalos, P.M.: Optimization software guide, vol. 14. Society for Industrial and Applied Mathematics (1993)
13. Pardalos, P.M.: Complexity in numerical optimization. World Scientific Singapore (1993)
14. Rao, J., Singh, A.: A ridge shrinkage method for range restricted weight calibration in survey sampling. In: Proceedings of the section on survey research methods, pp. 57–65 (1997)
15. Ryan, T.P.: Modern regression methods. John Wiley & Sons, New York (2008)
16. Särndal, C.: The calibration approach in survey theory and practice. Survey Methodology **33**(2), 99–119 (2007)
17. Singh, S., Arnab, R.: A bridge between the GREG and the linear regression estimators. In: Joint Statistical Meeting, ASA Section on Survey Research Methods, Seattle, pp. 3689–3693 (2006)
18. Théberge, A.: Calibration and restricted weights. Survey Methodology **26**(1), 99–108 (2000)
19. Tillé, Y., Matei, A.: Rsolnp: General non-linear optimization. R package version (2013)
20. Vanderhoeft, C.: Generalised calibration at Statistics Belgium: SPSS Module G-CALIB-S and current practices. Inst. National de Statistique (2001)