Numerical Optimization in Survey Statistics

Dissertation

zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften (Dr. rer. nat.)

Dem Fachbereich IV der Universität Trier
vorgelegt von

Matthias Wagner

Trier, Juli 2013

Gutachter: Prof. Dr. Ekkehard W. Sachs
Prof. Dr. Ralf T. Münnich

Contents

German Summary (Zusammenfassung)

In der modernen Survey-Statistik treten immer häufiger Optimierungsprobleme auf, die es zu lösen gilt. Diese sind oft von hoher Dimension und Simulationsstudien erfordern das mehrmalige Lösen dieser Optimierungsprobleme. Um dies in angemessener Zeit durchführen zu können, sind spezielle Algorithmen und Lösungsansätze erforderlich, welche in dieser Arbeit entwickelt und untersucht werden.

Bei den Optimierungsproblemen handelt es sich zum einen um Allokationsprobleme zur Bestimmung optimaler Teilstichprobenumfänge. Hierbei werden neben auf einem Nullstellenproblem basierende, stetige Lösungsmethoden auch ganzzahlige, auf der Greedy-Idee basierende Lösungsmethoden untersucht und die sich ergebenden Optimallösungen miteinander verglichen.

Zum anderen beschäftigt sich diese Arbeit mit verschiedenen Kalibrierungsproblemen. Hierzu wird ein alternativer Lösungsansatz zu den bisher praktizierten Methoden vorgestellt. Dieser macht das Lösen eines nichtglatten Nullstellenproblemes erforderlich, was mittels des nichtglatten Newton Verfahrens erfolgt.

Im Zusammenhang mit nichtglatten Optimierungsalgorithmen spielt die Schrittweitensteuerung eine große Rolle. Hierzu wird ein allgemeiner Ansatz zur nichtmonotonen Schrittweitensteuerung bei Bouligand-differenzierbaren Funktionen betrachtet.

Neben der klassischen Kalibrierung wird ferner ein Kalibrierungsproblem zur kohärenten Small Area Schätzung unter relaxierten Nebenbedingungen und zusätzlicher Beschränkung der Variation der Designgewichte betrachtet. Dieses Problem lässt sich in ein hochdimensionales quadratisches Optimierungsproblem umwandeln, welches die Verwendung von Lösern für dünn besetzte Optimierungsprobleme erfordert.

Die in dieser Arbeit betrachteten numerischen Probleme können beispielsweise bei Zensen auftreten. In diesem Zusammenhang werden die vorgestellten Ansätze abschließend in Simulationsstudien auf eine mögliche Anwendung auf den Zensus 2011 untersucht, die im Rahmen des Zensus-Stichprobenforschungsprojektes untersucht wurden.

Acknowledgements

First of all, I would like to thank Prof. Dr. Ekkehard Sachs for his support during my time as a student, while writing my diploma thesis, and while completing my PhD. I am not only grateful for having the opportunity to work as a research assistant, but even more for his advice, suggestions, inspiring discussions, and encouragement. Prof. Dr. Ralf Münnich also provided a fruitful ground for the completion of this work. I am very thankful for his suggestions, hints and ideas from practical applications, that led to a thesis with an interdisciplinary flavor.

Further, I would like to thank my colleagues at the Department of Mathematics, especially Marina Schneider and Ulf Friedrich, as well as the members of the Economic and Social Statistics Department at the University of Trier. All of them created an enjoyable as well as inspiring working atmosphere and we became good friends over the years.

Finally, my thanks go to my family and friends, who supported me and encouraged me during my academic career. Especially my friends created a diversion from work and helped clearing my mind whenever needed.

The research was financially supported by the Research Center for Regional and Environmental Statistics of the University of Trier which is part of the Rhineland-Palatinate research initiative.

Matthias Wagner
Trier, July 2013

Chapter 1

Introduction

A statistician is a person who draws a mathematically precise line from an unwarranted assumption to a foregone conclusion.

— Unknown

1.1 Motivation

Although statisticians were already employed in ancient Egypt, they do not always get the reputation they should get. Being seen as 'a person who draws a mathematically precise line from an unwarranted assumption to a foregone conclusion' is not very complimentary and makes it look like statisticians *are only able to explain things in artificial worlds.* In fact, this is not true! Conclusions are not foregone and the use of sophisticated models brings more 'reality' into the assumed world.

During the work of the German Census Sampling and Estimation Research Project it was inevitable to solve optimization problems, which are often of high dimension. This is where mathematics and especially numerical optimization takes the stage. Standard optimization algorithms may fail or may take a lot of time to get a solution, which is a big problem when doing simulation studies with a lot of repetitions. Therefore, numerical optimization approaches have to be developed further or existing approaches have to be adapted to the given setting.

This work shows how survey statistics and mathematics can collaborate, especially in the context of the German Census Sampling and Estimation Research Project. Starting with an allocation problem with one equality constraint and a box constraint (Chapter 3), we get to a calibration problem with several equality constraints and a box constraint (Chapter 5). This increase in the number of equality constraints complicates the methods needed and whilst not needed in Chapter 3, it may be helpful having a theory for (nonmonotone) step size rules in the context of solution methods for nonsmooth functions (Chapter 6). The calibration problem can also be developed further allowing the relaxation of some calibration constraints and a constraint concerning the spread ratio of weights is added (Chapter 7). Apart from the development and study of the theoretical background, implementations and simulation studies are also proceeded and analyzed.

In summary, it can be stated that this work aims at building a bridge between the survey statistical and the mathematical point of view. Not only concerning the problems and solution methods, but also concerning notations and habits.

1.2 Outline

First, we give a brief overview of each chapter. Chapter 2 and Chapter 4 aim to present the backgrounds of survey statistics and nonsmooth analysis, whereas Chapter 3, 5, 6 and 7 deal with the arising optimization problem. In Chapter 8 we conclude this thesis and give a short outlook on further research topics as well as applications of the presented methods.

Chapter 2: Fundamentals of Survey Statistics

In this chapter we briefly summarize some statistical aspects that will be needed later on. For a more detailed elaboration we refer to Münnich, Gabler, Ganninger, Burgard and Kolb (2012) or Münnich et al. (2013).

Definition 1.2.1. *A survey is a systematic method for gathering information from (a sample of) entities for the purposes of constructing quantitative descriptors of the attributes of the larger population of which the entities are members (Groves et al., 2004).*

As a survey can be very costly, the aim is to keep the number of people interviewed as small as possible. However, the gained information should be as precise as possible. Such surveys are called sample surveys and apart of being cheaper than a classical survey, the desired information is also gained faster than in the traditional way. As the information gained is only available for a certain fraction of the population, the totals

$$t_y = \sum_{k \in U} y_k,$$

have to be estimated by appropriate estimators. In this context, these estimators can be roughly divided up into design based estimators shown in Section 2.2, synthetic estimators given in Section 2.3 and composite estimators discussed in Section 2.4. An example for a design based estimator is the Horvitz-Thompson estimator

$$\hat{t}_y^{HT} = \sum_{k \in s} d_k y_k,$$

summing up the characteristic attributes of the elements of the sample multiplied by design weights.

In contrast to direct estimators, that only use domain-specific sample data, synthetic estimators make use of data from outside this domain. The synthetic estimator, model A, of the total of the variable of interest y in area d is defined as

$$\hat{t}_{y,d}^{SynthA} = t_{x,d}^T \hat{\beta},$$

where $t_{x,d}$ denotes the area totals of the auxiliary variable x and $\hat{\beta}$ is estimated from the unit level model

$$y_k = x_k^T \beta + v_d + e_k \quad \forall k \in U_d,\ d = 1, \ldots, D,$$

where v_d represents area-specific effects and e_k unit-specific effects with classical assumptions.

A composite estimator combines these two types of estimators through a convex combination in order to omit the individual disadvantages of using only one of the types of estimators mentioned before. Well known estimators are empirical best linear unbiased predictors (EBLUP) or the YOURAO estimator given in You and Rao (2002).

In order to evaluate those estimators and how they perform under certain circumstances, simulation studies are proceeded. This evaluation can be done by various measures (RRMSE, RBias) which are given in Section 2.5.

Chapter 3: Optimal Allocation Problems in Statistics

This chapter deals with optimal allocation problems of the form

$$\begin{aligned} \min_{n \in \mathbb{R}_+^H} & \sum_{h=1}^{H} \frac{d_h^2}{n_h} \\ s.t. & \sum_{h=1}^{H} n_h \leq n^s \\ & m_h \leq n_h \leq M_h \quad \forall h = 1, \ldots, H. \end{aligned}$$

Such optimization problems occur when minimizing the variance of an estimator as a function of the partial sample sizes n_h in the different strata with given sample size n^s and minimal/maximal sampling fractions. These sampling fractions are very important because they allow to gain reliable model estimates from rural vs. urban comparisons, where the classical optimal allocation would yield an extremely high sampling fractions in large towns and very low sampling fraction in rural areas. In Section 3.2 we present methods to solve the continuous allocation problem. On the one hand, we propose an approach via the Lagrange multiplier, where the sampling fractions are expressed as a function depending on the Lagrange multiplier. This expression is inserted into the equality constraint leading to a one-dimensional equation whose root has to computed. On the other hand, a fixed point iteration depending on certain subsets J_M^λ, J_m^λ and J^λ of the index set is derived, so we obtain the fixed point iteration

$$\lambda^{k+1} := \left(\frac{\sum_{h \in J^{\lambda^k}} d_h}{n^s - \sum_{h \in J_M^{\lambda^k}} M_h - \sum_{h \in J_m^{\lambda^k}} m_h} \right)^2.$$

The approaches mentioned so far solve the allocation problem in continuous variables, i.e., the integrality conditions on the variables are relaxed. A simple rounding of the continuous solution does in general not deliver the optimal and may even lead to an infeasible solution.

Therefore, in Section 3.4 we consider the integer allocation problem and study different greedy based solution methods. The drawback of the simple greedy strategy is that only increments of one unit per iteration are possible and therefore lots of (numerically cheap) iterations are needed to find the optimum. We present a refinement that generally uses only a fraction of the number of iterations of the simple strategy. Starting with an increment of $s > 1$, the increment is assigned to the most favorable activity until no such increments are possible. Then, s is decreased and the process of scaled greedy increments is repeated with the successively smaller increments until the increment equals one. Further, a binary search algorithm is presented. The entire solution can be reconstructed from the value of the marginal δ_{last} in the last iteration of the algorithm. That is because by computing the marginal at an arbitrary value of an arbitrary variable and comparing it to δ_{last}, we can decide if n_h is above or below its value in the optimal solution. Hence, the optimization problem is equivalent to finding δ_{last}, which can easily be done by a binary search.

In Section 3.5 we apply all algorithms to the simplified census problem

$$\begin{aligned} \min_{n} \; & \sum_{g=1}^{2391} \sum_{h=1}^{8} \frac{N_{g,h}^2 S_{g,h}^2}{n_{g,h}} \\ s.t. \; & \sum_{g=1}^{2391} \sum_{h=1}^{8} n_{g,h} = n^s \\ & m_{g,h} \leq n_{g,h} \leq M_{g,h} \quad \forall g = 1, \ldots, 2391, \; h = 1, \ldots, 8, \end{aligned}$$

and compare their performance. We further compare the rounded solution with the integer solution, which in general do not coincide. In the simulation study, the rounded solution leads to an allocation with 25 elements less than the allocation determined by the greedy-type algorithms (Section 3.6).

Chapter 4: Fundamentals of Nonsmooth Analysis

In this chapter several types of derivatives for smooth and nonsmooth functions are treated. Apart from the well known directional derivative, Gâteaux derivative, Hadamard derivative and Fréchet derivative given in Section 4.2, we present the Bouligand derivative in Section 4.3.

Definition 1.2.2. *A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be Bouligand differentiable at $x \in \mathbb{R}^n$ if there exists a positively homogeneous function $A : \mathbb{R}^n \to \mathbb{R}^m$, called the B-derivative of f at x, such that*

$$\lim_{h \to 0} \frac{f(x+h) - f(x) - A(h)}{\|h\|} = 0.$$

Comparing a B-differentiable function and a Fréchet differentiable function, the one fundamental distinction is the absence of linearity in the B-derivative compared to the Fréchet derivative. When the given function f is further locally Lipschitz, certain directional deriva-

tives coincide. Therefore, $f : \mathbb{R}^n \to \mathbb{R}^m$ is called Bouligand differentiable in $x \in D$ if f is locally Lipschitz and all directional derivatives exist in $x \in D$. The local Lipschitz property is important because it is known that there are functions that are directionally differentiable at a point without being continuous there. Another important consequence of the Bouligand differentiability is that the limit of the directional derivative is uniform on compact sets of directions.

If we want to develop algorithms for solving nonsmooth equations we will encounter many difficulties when sticking solely to directional derivatives. Therefore, in Section 4.4 we introduce the generalized Jacobian of the locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}^m$ in $x \in \mathbb{R}^n$ defined as

$$\partial f(x) := \text{conv } \{V \in \mathbb{R}^{m \times n} : \ \exists (x^k)_{k \in \mathbb{N}} \subset D_f : \ x^k \to x \text{ and } J_f(x^k) \to V\}.$$

The generalized Jacobian helps to extend many results from smooth analysis to locally Lipschitz functions. However, a straightforward extension of Newton's method to general nonsmooth equations by using the generalized Jacobian will not work easily, so another important notion of nonsmooth analysis, namely semismoothness, is introduced in Section 4.5.

Definition 1.2.3. *Let $D \subset \mathbb{R}^n$ and $f : D \to \mathbb{R}^m$ be a B-differentiable function. Then f is called semismooth in $x \in D$, if*

$$\lim_{h^k \to 0, V_k \in \partial f(x+h^k)} \frac{V_k h^k - f'(x; h^k)}{\| h^k \|} = 0.$$

Semismooth functions are locally Lipschitz functions for which the generalized Jacobians define a certain approximation scheme. This makes it possible to get almost the same results for Newton's method in the nonsmooth case as in the smooth case.

Chapter 5: Calibration via Semismooth Newton Method

Extending the optimization problem of Chapter 3 from one equality constraint to m equality constraints leads to the following optimization problem:

$$\begin{aligned} \min_{g \in \mathbb{R}^n} \ & d^T F(g) \\ s.t. \ & \bar{X}^T g - t_x = 0 \\ & g \in U. \end{aligned}$$

These problems occur in the calibration approach to estimation for finite populations (cf. Särndal, 2007). Weights, that incorporate specified auxiliary information and are restrained by calibration equations are computed and afterwards used to compute linearly weighted estimates of totals and other finite population parameters. This approach is easy to explain to users and is widely accepted. Furthermore, using auxiliary information allows to improve the accuracy of survey estimates and can deal effectively with surveys where auxiliary information exists at different levels. In addition to that, if the gained weights are applied to a

variable used for calibration, they deliver the given estimates or true values. This is also very important because consistency with known aggregates is a desire to promote credibility.

In Section 5.3, the vector of calibration factors g is expressed as a function depending on the Lagrange multiplier λ. Then, this expression $g(\lambda)$ is inserted into the function

$$h : \mathbb{R}^n \to \mathbb{R}^p,\ g \mapsto h(g) = \bar{X}^T g - t,$$

which leads to a p-dimensional nonsmooth equation $\psi(\lambda) = 0$ where

$$\psi : \mathbb{R}^p \to \mathbb{R}^p,\ \lambda \mapsto \bar{X}^T g(\lambda) - t_x$$

with

$$g_k(\lambda) = \mathrm{Pr}_{[m_k, M_k]}\left(f'^{-1}(-\frac{\xi_k^T \lambda}{d_k})\right) \quad (k = 1, \ldots, n).$$

In the standard case of having a quadratic objective function, we show that ψ is strongly semismooth so the 'semismooth Newton method' (cf. Qi, 1993) given in Section 5.4 can be applied. In each iteration of the semismooth Newton method a linear system of equations

$$H_k s^k = -\psi(\lambda^k),$$

with H_k being an element of the B-subdifferential $\partial_B \psi(\lambda^k)$, is solved and the resulting next iterate is computed as

$$\lambda^{k+1} = \lambda^k + s^k.$$

Under certain conditions, the iterates converge quadratically to the optimal solution. This convergence is also shown numerically in Section 5.5, where we apply the semismooth Newton method to a given calibration problem.

Chapter 6: Nonmonotone Step Size Rules for B-Differentiable Functions

The semismooth Newton method mentioned before is a special algorithm for solving nonsmooth equations. In a more general notation, the iterates are computed by

$$x_{k+1} = x_k + \alpha_k d_k,$$

with appropriated step size α_k, and d_k is computed by

$$f(x_k) + A(x_k)(d_k) = 0.$$

Depending on the choice of $A(x_k)(\cdot)$ we get different methods. In Pang (1990),e.g., $A(x_k)(d_k)$ is replaced by the B(ouligand) derivative of f in x_k applied to d_k.

In order to determine the step size α_k, a line search depending on the merit function

$$\theta : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\},\ x \mapsto \theta(x) = \|f(x)\|^2$$

can be used. Most of the methods using line search require the decrease of the function values to be monotone. However, for certain problems a nonmonotone line search may deliver better results. In Section 6.2 we generalize the approaches for B-differentiable functions by using nonmonotone step size rules deriving from Armijo (1966).

Algorithm 1.1 Nonmonotone Armijo's rule

Input: $\beta \in (0,1),\ \sigma \in (0,\bar{\sigma}),\ x_k, d_k \in \mathbb{R}^n,\ \nu_k \in \mathbb{R}_+ \cup \{0\},\ \alpha_{max} > 0,\ \theta(x_k) \neq 0$
Ensure: d_k is chosen such that (6.4) holds.
  if $\theta(x_k + \alpha_{max} d_k) - \theta(x_k) \leq -\sigma\alpha_{max}\theta(x_k) + \nu_k$ **then**
    $\alpha_k = \alpha_{max}$
  else
    determine smallest $l_k \in \mathbb{N}$ such that
    $\theta(x_k + \alpha_{max}\beta^{l_k} d_k) - \theta(x_k) \leq -\sigma\alpha_{max}\beta^{l_k}\theta(x_k) + \nu_k$
    $\alpha_k = \alpha_{max}\beta^{l_k}$
  end if

The nonmonotone Armijo's rule is well defined and when requiring that $\sum_{k=1}^{\infty} \nu_k < \infty, \nu_k \geq 0$ for all k, we can show that $\theta(x_k) \xrightarrow[k\to\infty]{} 0$. This setting allows only to prove that the iterates of the function values $\theta(x_k)$ converge to 0. Nevertheless, we can deduce that there exists at least one subsequence $(x_{k_j})_{j\in\mathbb{N}}$ converging to an accumulation point $x_* \in S$ satisfying $\theta(x_*) = 0$.

Our approach can also be applied to hybrid methods similar to the ones given in Ito and Kunisch (2009) or Qi (1993). Those methods first try a full step with Newton search direction which is tested with kind of a 'watchdog step'. If this step is 'good', a Newton step is performed. Otherwise, a search direction satisfying certain requirements is determined and a monotone line search is performed. If we further assume that there exists x_* such that $x_k \xrightarrow[k\to\infty]{} x_*$, we can prove superlinear convergence of the iterates.

In Section 6.3 we apply the nonmonotone step size rule of Zhang and Hager (2004) to nonsmooth optimization.

Chapter 7: Generalized Calibration for Coherent Small Area Estimation

Standard calibration problems and its solution methods are lacking some important aspects: A regulation of the spread of the calibrated weights w is only done by the range restriction. However, this does not take the ratio of the largest to the smallest calibrated weight into account. Following Little et al. (2009), this ratio should not exceed 10 and is unacceptable beyond 100. Further, there often exist many estimates on different levels. These estimates are gained by different estimators which leads to coherence problems, so we have to allow the benchmarks to be fulfilled with a slight perturbation. Apart from this, the known methods, for instance given in Beaumont and Bocci (2008), use penalization and do not allow to analyze which calibration benchmarks are restrictive and might be relaxed in order to get overall better estimates.

In this chapter we include this requirements into the optimization process. Therefore, the following optimization problem is defined in Section 7.2.

$$\begin{aligned}
\min_{(g,\epsilon,\alpha,\beta)\in\mathbb{R}^{n+u+2}} \quad & \sum_{k\in s} d_k \frac{(g_k-1)^2}{2} + \sum_k \delta_k \frac{(\epsilon_{dis,k}-1)^2}{2} + \sum_k \delta_k \frac{(\epsilon_{SMP,k}-1)^2}{2} \\
s.t.\ & X_{jl}^T g - t_{x_{jl}} = 0 \quad \forall j = 1,\dots,K,\ l = 1,\dots,G_j\ (SMPs) \\
& \bar{X}_j^T g - \mathrm{diag}(t_{\bar{x}_j})\epsilon_{dis} = 0 \quad \forall j = 1,\dots,K\ (districts) \\
& \bar{X}_{jl}^T g - \mathrm{diag}(t_{\bar{x}_{jl}})\epsilon_{SMP} = 0 \quad \forall j = 1,\dots,K,\ l = 1,\dots,G_j\ (SMPs) \\
& -d_k g_k + \alpha \leq 0 \quad \forall k = 1,\dots,n \\
& d_k g_k - \beta \leq 0 \quad \forall k = 1,\dots,n \\
& -GB\alpha + \beta \leq 0. \\
& (g, \epsilon_{dis}, \epsilon_{SMP}) \in [m, M] \times [m_{\epsilon_{dis}}, M_{\epsilon_{dis}}] \times [m_{\epsilon_{SMP}}, M_{\epsilon_{SMP}}].
\end{aligned}$$

This can easily be rewritten as a high dimensional quadratic program

$$\begin{aligned}
\min_{z\in\mathbb{R}^{n+u+2}} \quad & z^T Q z - q^T z \\
s.t.\ & Az \leqq t \\
& z \in [L, U].
\end{aligned}$$

The sparsity of the calibration matrix A is analyzed in Section 7.3 and an application to the German Census Sampling and Estimation Research Project is given in Section 7.4. In this simulations study, 990 samples and the corresponding values for the REG and ISCED variables are analyzed for different calibration settings. We regard resolvability and the relaxation of the equality constraints depending on different benchmarks and districts/SMPs is analyzed. Further, the initial weights and the calibrated weights as well as the gained objective values are analyzed. The loss of accuracy of the perturbed estimates for the ISCED variables gained by the calibrated Horvitz-Thompson estimator for domain d compared to the YOURAO estimator is analyzed by regarding the RRMSE. This is important because it gives us some information on the loss of accuracy in the Eurostat tables and a one number census. As we further want to know how good calibrated Horvitz-Thompson estimates of further variables, e.g., EF117, are compared to the true values, the RRMSE and the RBias are regarded.

Chapter 8: Conclusion

The last chapter briefly summarizes the thesis and gives a short outlook on further research topics as well as possible applications, especially in the context of research concerning the German Census 2011.

Chapter 2

Fundamentals of Survey Statistics

The need for statistical information seems endless in modern society.

— Särndal, Swensson & Wretman
Model Assisted Survey Sampling

As mentioned in Särndal et al. (2003), the need for statistical information seems endless in modern society. Data is collected for different needs of the demand carrier and in different ways. Collecting may be done automatically, for instance by storing every search request on `www.google.de` or by storing the list of products bought in the supermarket while collecting points for a PAYBACK account. Companies like 'Google Incorporated' or 'Payback GmbH' have the big advantage that they get this information for free and for their purpose, that is personalized advertising, they do not need to estimate certain values for a whole population. However, those data loggers cannot male a point concerning the quality of this big data. In contrast to this, we regard the whole process of collecting and evaluating data in a sophisticated and integrated manner. Therefore, we deal with surveys and especially sample surveys.

Definition 2.0.4 (Survey)**.** *A* **survey** *is a systematic method for gathering information from (a sample of) entities for the purposes of constructing quantitative descriptors of the attributes of the larger population of which the entities are members (Groves et al., 2004).*

2.1 Why Survey Statistics?

Following Rossi et al. (1983), the classic demand carriers for surveys can be classified into the government sector, the academic sector, the private and mass media sector as well as the residual sector consisting of ad hoc and in-house surveys.

Having a closer look at this sectors, we can state that the government usually legitimates a statistical office to collect data on important national characteristics and activities such as demography, that is for example age and sex distribution, fertility and mortality, agriculture, industry, trade, labor force as well as health and living conditions. For a detailed discussion on this topics we refer to Raj (1968). Another very important task of these statistical offices is to operate a national census, which in the case of being a member state of the European Union, is mandatory every ten years starting 2011. The aim of a census is to provide the most

accurate snapshot possible of basic information on the population, housing space, education, and employment (cf. Münnich, Gabler, Ganninger, Burgard and Kolb, 2012). A detailed discussion on the German Census Sampling and Estimation Research Project can be found in Chapter 3. Further information concerning the German Census 2011 can be found on `www.zensus2011.de`.

In the academic sector, surveys are used by several departments, e.g., sociology and public opinion research, economics, political science or psychology. In this context, the gained information is often needed for checking whether an anticipated model is true or not and to support theoretical findings. Nevertheless, the transition to the private and mass media sector is rather smooth. Marketing surveys are used and proceeded by the economics department as well as private companies. Furthermore, the exchange of knowledge between these parties is growing. The mass media sector also covers television audience surveys, readership surveys or polls.

The big disadvantage of a survey can be stated in one word: expensive! Therefore, the aim is to keep the number of people interviewed as small as possible whereas the gained information should be as precise as possible. Such surveys are called sample surveys (cf. Cochran, 1977; Raj, 1968; Särndal et al., 2003) and have the following principal advantages compared to complete enumeration (cf. Cochran, 1977; Raj, 1968).

(i) Reduced cost: compared to a full survey one has to train less interviewers which have to do less interviews. This gains a reduction of one of the main costs of a survey.

(ii) Greater speed: as less interviews have to be done, the information is gained more quickly which is especially important if the information is urgently needed.

(iii) Greater scope: gaining certain types of information may sometimes only be possible by using specialized equipment limited in availability, so the choice lies between obtaining the information by sampling or not at all.

(iv) Greater accuracy: personnel of higher quality can be employed and given intensive training. This and less volume of work make the results more accurate.

These sample surveys can be classified broadly into two types - descriptive and analytical (cf. Cochran, 1977). Whilst in a descriptive survey the objective is simply to obtain certain information about large groups, an analytical survey makes comparisons between different subgroups of the population, in order to discover whether differences exist among them and to form or to verify hypotheses about the reasons for these differences.

After having decided to conduct a survey, one has to properly discuss all steps involved in the planing and execution of a survey which, following Cochran (1977), are:

(i) Objectives of the survey.

(ii) Population to be sampled.

(iii) Data to be collected.

(iv) Degree of precision desired.

(v) Methods of measurement.

(vi) The frame.

(vii) Selection of the sample.

(viii) The pretest.

(ix) Organization of the field works.

(x) Summary and analysis of the data.

(xi) Information gained for further surveys.

Among this list of steps there exist further lists, like the one mentioned in Raj (1968), which are more or less the same. We will not discuss all steps in detail but stick to those of greater importance for this work. The selection of the sample, that is step (vii), plays an important role in the German Census Sampling and Estimation Research Project because not only statistical aspects but also legal boundary conditions have to be taken into account. This topic will be discussed in detail in Chapter 3. Further, in step (x), the summary and analysis of the data, the computations that lead to the estimates are performed, where different methods for estimating may be applicable. An overview of different estimators is given below and the estimation under auxiliary information via 'semismooth Newton method' (cf. Qi and Sun, 1993) is shown in Chapter 5. Another topic of the analysis of the data, which can be also assigned to step (xi), is the determination of vertical coherent estimates in case of using different estimators. For a detailed discussion on this topic we refer to Chapter 7. An application of the steps mentioned above can be found in Münnich, Gabler, Ganninger, Burgard and Kolb (2012) where they proceed those steps for the German Census 2011.

Hereafter we will briefly summarize some statistical aspects that will be needed in the following chapters. For a more detailed elaboration we refer to Münnich et al. (2013).

2.2 Design Based Estimators

Assume we have a finite population U consisting of N elements which are denoted by labels $k = 1, \ldots, N$. Furthermore we assume that measurements of the real or integer valued variable y are available for every $k = 1, \ldots, N$ and will be denoted by y_k $(k = 1, \ldots, N)$. The parameter or variable of interest is the population total

$$t_y = \sum_{k \in U} y_k,$$

where the index y indicates the variable forming the total. A sampling design is given and a sample $s \subset U$ with cardinality n and sample selection probability $p(s)$ is drawn. This sampling design leads to design weights $d_k(s)$ depending on the sample s and element k $(k \in s)$. A common choice is the reciprocal of the inclusion probabilities, i.e., $d_k(s) = \pi_k^{-1}$ where $\pi_k = \sum_{s:k \in s} p(s) \geq 0,\ k = 1, \ldots, N$. For simplification we write d_k instead of $d_k(s)$, $(k \in s)$. In the absence of auxiliary information we can estimate the population total t_y by using the **Horvitz-Thompson estimator (HT)** mentioned in Horvitz and Thompson (1952) or Cochran (1977).

Definition 2.2.1 (Horvitz-Thompson estimator). *The* **Horvitz-Thompson estimator** *of the total t_y is defined as*

$$\hat{t}_y^{HT} = \sum_{k \in s} d_k y_k,$$

where the design weights are given by $d_k = \pi_k^{-1},\ k \in s$.

For common design weights d_k we get the so called **expansion estimator**

$$\hat{t}_y = \sum_{k \in s} d_k y_k.$$

In case of different domains or areas $U_d,\ d = 1, \ldots, D$, where

$$\bigcup_{k=1}^{D} U_d = U,\ U_d \cap U_e = \emptyset \quad \forall d \neq e,$$

the Horvitz-Thompson estimator of the total $t_{y,d}$ for domain d takes the form

$$\hat{t}_{y,d}^{HT} = \sum_{k \in s_d} d_k y_k,$$

where $s_d = \{k : k \in s \text{ and } k \in U_d\}$ for all $d = 1, \ldots, D$. As the Horvitz-Thompson estimator only uses domain-specific sample data, it is called a **direct estimator**. Note that depending on the sample, these estimators do not always deliver accurate estimates.

Suppose now that for the sake of simplicity $s = \{1, \ldots, n\}$ and p ($p \ll n$) auxiliary information is available in the form of known population totals $t_{x_j},\ j = 1, \ldots, p$ and known characteristics for element k and information j denoted by $x_{kj},\ k \in s,\ j = 1, \ldots, p$. (In the case that the auxiliary information is available for every $k \in U$ instead of $k \in s$ we have complete auxiliary information.) These auxiliary information forms the design matrix X and for staying in line with common statistical notation, all p known characteristics for element k form the vector $x_k = (x_{k1}, \ldots, x_{kp})^T \in \mathbb{R}^p$ such that

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix} = \begin{pmatrix} — & x_1^T & — \\ & \vdots & \\ — & x_n^T & — \end{pmatrix} \in \mathbb{R}^{n \times p},$$

and the vector of population totals is denoted by

$$t_x = \begin{pmatrix} t_{x_1} \\ \vdots \\ t_{x_p} \end{pmatrix} \in \mathbb{R}^p.$$

An estimator that makes efficient use of this auxiliary information is the **generalized regression estimator (GREG)** which can be found in Rao (2003) or Särndal et al. (2003)

and will be defined and motivated below. For a detailed discussion of regression estimators we refer to Cochran (1977).

Definition 2.2.2 (Generalized regression estimator)**.** *The* **generalized regression estimator (GREG)** *of the total t_y is defined as*

$$\hat{t}_y^{GR} = \hat{t}_y + (t_x - \hat{t}_x)^T \hat{\beta}, \tag{2.1}$$

where $\hat{t}_x = \sum_{k \in s} d_k x_k$ and $\hat{\beta} \in \mathbb{R}^p$ is the solution of the system of linear equations

$$\left(\sum_{k \in s} d_k x_k x_k^T \right) \hat{\beta} = \sum_{k \in s} d_k x_k y_k.$$

The motivation for this is as follows. Assume that the variable of interest y has been generated by a linear regression model such that

$$E(y_k) = x_k^T \beta, \quad k = 1, \ldots, N,$$
$$V(y_k) = \sigma_k^2, \quad k = 1, \ldots, N.$$

This can be rewritten in matrix respectively vector notation with $\mathbf{y} = (y_1, \ldots, y_N)^T$, $\mathbf{X^T} = (x_1, \ldots, x_N) \in \mathbb{R}^{p \times N}$ and $\Sigma = \text{diag}(\sigma_1^2, \ldots, \sigma_N^2)$, so we get

$$\mathbf{y} = \mathbf{X}\beta + \epsilon, \quad \text{with } E(\epsilon) = 0,\ V(\epsilon) = \Sigma.$$

Assume that y_k and x_k are known for all $k = 1, \ldots, N$, so we want to get $\bar{\beta}$ such that

$$\bar{\beta} = \text{argmin}_\beta \frac{1}{2} \|\mathbf{X}\beta - \mathbf{y}\|^2_{\Sigma^{-1}}.$$

Therefore, $\bar{\beta}$ has to satisfy

$$(\mathbf{X^T}\Sigma^{-1}\mathbf{X})\bar{\beta} = \mathbf{X^T}\Sigma^{-1}\mathbf{y},$$

which is equivalent to

$$\left(\sum_{k \in U} x_k x_k^T \sigma_k^{-2} \right) \bar{\beta} = \sum_{k \in U} x_k y_k \sigma_k^{-2}.$$

As in the case of a sample survey we do not know y_k and x_k for all $k \in U$ but only for $k \in s$, we have to estimate $\bar{\beta}$ by the sample s. If we further assume that $\sigma_k^2 = 1$ for all k we get

$$\left(\sum_{k \in s} d_k x_k x_k^T \right) \hat{\beta} = \sum_{k \in s} d_k x_k y_k.$$

Therefore, $\hat{\beta}$ estimates $\bar{\beta}$ and $\bar{\beta}$ in turn estimates the model parameter β.

It is also useful to write $\hat{t}_y^{GR}$ in the expansion form

$$\hat{t}_y^{GR} = \sum_{k \in s} w_k y_k,$$

with calibrated weights $w_k = d_k g_k$ for $k \in s$ and

$$g_k = 1 + (t_x - \hat{t}_x)^T \left(\sum_{k \in s} d_k x_k x_k^T \right)^{-1} x_k \quad \forall k \in s.$$

This derives from the calibration approach and will be discussed in Chapter 5 in detail. Further, we will extend this approach by adding bounds to the calibrated weights w_k. In the context of small area estimation it is also helpful to rewrite the GREG estimator (2.1). If we replace $\hat{t}_y$ by $\sum_{k \in s} d_k y_k$ and $\hat{t}_x$ by $\sum_{k \in s} d_k x_k$, we get

$$\hat{t}_y^{GR} = t_x^T \hat{\beta} + \sum_{k \in s} d_k (y_k - x_k^T \hat{\beta}).$$

Now we can see that the estimator consists of a synthetic part and the weighted residuals $e_k := y_k - x_k^T \hat{\beta}$. In the case of different domains d the domain estimator is written as

$$\hat{t}_{y,d}^{GR} = t_{x,d}^T \hat{\beta}_g + \sum_{k \in s_d} d_k (y_k - x_k^T \hat{\beta}_g), \tag{2.2}$$

where $s_d = s \cap U_d$, $d = 1, \ldots, D$ and $\hat{\beta}_g$ indicates that the estimated β may depend on different groups g. This is discussed in detail in Särndal et al. (2003) or Lehtonen and Veijanen (2009). If the estimation of β depends on groups which contain several areas we speak of **borrowing strength** because the estimator uses information from these areas for estimating a certain value in a certain area. This is a first step towards indirect estimators, that make use of information from outside the given domain. In this context we will introduce the definition for small and large areas.

Definition 2.2.3 (Small area)**.** *A domain (area) is regarded as* **large** *(or major) if the domain-specific sample is large enough to yield direct estimates of adequate precision. A domain is regarded as* **small** *if the domain-specific sample is not large enough to support direct estimates of adequate precision (Rao, 2003).*

Another interesting aspect of the GREG estimator is mentioned in Särndal et al. (2003). The GREG estimates made for different subpopulations add up to the estimate made for the population as a whole. This useful property of the regression estimator is called **vertical coherence** and will play an important role in Chapter 7.

2.3 Synthetic Estimators

In contrast to direct estimators, that only use domain-specific sample data, **synthetic estimators** make use of data from outside this domain/area. To be precise, we speak of a

synthetic estimator if a reliable direct estimator for a large area, covering several small areas, is used to derive an indirect estimator for at least one of these small areas. This concept assumes that the underlying small areas have the same sample characteristics as the large area. If this assumption is violated, the synthetic estimator can be heavily biased. For a detailed discussion on this topic and the estimators presented below we refer to Rao (2003) or Münnich, Gabler, Ganninger, Burgard and Kolb (2012).

Definition 2.3.1 (Synthetic estimator, no auxiliary information)**.** *In the case that* **no auxiliary information** *is available, the easiest* **synthetic estimator** *for the total of the variable y in area d denoted by $t_{y,d}$ forms as follows.*

$$\hat{t}_{y,d}^{SynthN} = \frac{\sum_{k\in s_d} d_k}{\sum_{k\in s} d_k} \sum_{k\in s} d_k y_k \quad \forall d = 1,\dots,D.$$

We can see that it consists of an expansion estimator for the whole population multiplied by the estimated fraction of the area in the whole population. Usually, this estimator leads to a biased estimation which is not desirable.

Now assume that unit specific auxiliary data $x_k = (x_{k1},\dots,x_{kp})^T$ is available for each element $k \in U$ and the area specific totals $t_{x,d}$ are known. Then we are able to formulate a regression synthetic estimator of $t_{y,d}$.

Definition 2.3.2 (Synthetic regression estimator)**.** *In the case that auxiliary information is available, the* **regression synthetic estimator** *for the total of the variable y in area d denoted by $t_{y,d}$ forms as follows.*

$$\hat{t}_{y,d}^{SynthR} = t_{x,d}^T \hat{\beta} \quad \forall d = 1,\dots,D,$$

where $\hat{\beta}$ is estimated by

$$\hat{\beta} = \left(\sum_{k\in s} d_k x_k x_k^T\right)^{-1} \sum_{k\in s} d_k x_k y_k.$$

As $\hat{\beta}$ is computed by domain wide information, this estimator assumes that the regression term $\hat{\beta}$ for every domain d is the same as the regression term for the whole population.

Now, every variable of interest y_k is assumed to be related to x_k through a one-fold nested error linear regression model, i.e.,

$$y_k = x_k^T\beta + v_d + e_k \quad \forall k \in U_d,\ d = 1,\dots,D, \tag{2.3}$$

where v_d represents area-specific effects and e_k unit-specific effects for whom holds:

$$v_d \sim \text{iid } N(0;\sigma_v^2) \quad \forall d = 1,\dots,D,$$
$$e_k \sim \text{iid } N(0;\sigma_e^2) \quad \forall k \in U.$$

This means v_d and e_k are random variables, which are independent and identically normally

distributed with mean 0 and variances σ_v^2 and σ_e^2. As the auxiliary data is available for every unit $k \in U$, this model is referred to as **unit level model** (cf. Rao, 2003, where he also speaks of a 'type B' model, or Battese et al., 1988).

Definition 2.3.3 (Synthetic estimator, model A). *The* **synthetic estimator, model A** *of the total of the variable of interest y in area $d = 1, \ldots, D$ denoted by $t_{y,d}$ is defined as*

$$\hat{t}_{y,d}^{SynthA} = t_{x,d}^T \hat{\beta},$$

where $t_{x,d}$ denotes the area totals of the auxiliary variable x and $\hat{\beta}$ is estimated from the **unit level model** (2.3).

The estimation of $\hat{\beta}$ can be done by maximum likelihood or restricted maximum likelihood methods which we will not discuss in detail and refer to Harris and Stocker (1998). Apart from this model, there exist many other models deriving from model (2.3), such as the multivariate nested error regression model, random error variance linear model, two-fold nested error regression model, two-level model or general linear mixed model. An overview of these and related methods as well as links to the original papers can be found in Rao (2003).

Assume now that the auxiliary information is no longer available on unit level but therefore on area level. Furthermore, the variable of interest on area level, like the mean $\bar{y}_d$, is assumed to be related to the known mean of the auxiliary data on area d denoted by $\bar{x}_d$ through a linear model, i.e.,

$$\bar{y}_d = \bar{x}_d^T \beta + \zeta_d \quad \forall d = 1, \ldots, D, \tag{2.4}$$

where ζ_d represents area-specific effects for which holds:

$$\zeta_d \sim \text{iid } N(0; \sigma_v^2 + \frac{\sigma_e^2}{|s_d|}) \quad \forall d = 1, \ldots, D.$$

In contrast to the unit level model (2.3), the auxiliary data only needs to be available on area level. Therefore, this model is referred to as **area level model** (cf. Rao, 2003, where he also speaks of a 'type A' model, or Fay and Herriot, 1979).

Definition 2.3.4 (Synthetic estimator, model B). *The* **synthetic estimator, model B** *of the total of the variable of interest y in area $d = 1, \ldots, D$ denoted by $t_{y,d}$ is defined as*

$$\hat{t}_{y,d}^{SynthB} = t_{x,d}^T \hat{\beta},$$

where $t_{x,d}$ denotes the area totals of the auxiliary variables x and $\hat{\beta}$ is estimated from the **area level model** (2.4).

As it is also the case in model A, there exist other models deriving from model (2.4) such as the multivariate Fay-Herriot model, correlated sampling error model, spatial model as well as time series and cross-sectional model which are again mentioned in Rao (2003).

2.4 Composite Estimators

Comparing a synthetic estimator with a GREG estimator one can state that the variance of the synthetic estimator is smaller than the variance of the GREG estimator but the estimates are rather biased if the model does not fit the structures of all areas. This observation, which is discussed in detail in Lehtonen and Veijanen (2009), motivates to combine two types of estimators in order to omit the individual disadvantages of using only one of the mentioned types of estimators.

Definition 2.4.1 (Composite estimator). *Let $\hat{t}_{y,d}^{Dir}$ be a direct and $\hat{t}_{y,d}^{Synth}$ be a synthetic estimator for the total of the variable of interest y in area/domain d denoted by $t_{y,d}$. Then the* **composite estimator** *$\hat{t}_{y,d}^{Comp}$ of the total of the variable of interest y in area $d = 1, \ldots, D$ denoted by $t_{y,d}$ is defined as*

$$\hat{t}_{y,d}^{Comp} = \gamma_d \hat{t}_{y,d}^{Dir} + (1 - \gamma_d)\hat{t}_{y,d}^{Synth},$$

where $\gamma_d \in [0, 1]$ for all $d = 1, \ldots, D$.

It is now easy to define different composite estimators, which differ in the used direct and synthetic estimator as well as in the way how the parameter γ_d is chosen. We concentrate on two standard estimators of small area estimation, the so called **empirical best linear unbiased predictors (EBLUP)**. The term 'empirical' indicates that the needed variances are estimated from the sample.

When assuming that y_k is related to x_k through the unit level model (2.3) and SynthA is used as synthetic estimator and the GREG as direct estimator, we get the **EBLUPA**.

Definition 2.4.2 (EBLUPA). *Let $\hat{\mu}_{y,d}^{MLGR}$ be the multilevel GREG estimator and $\hat{\mu}_{y,d}^{SynthA}$ be the synthetic estimator, model A for the mean of the variable of interest y in area/domain d denoted by $\mu_{y,d}$. Then the* **EBLUPA** *$\hat{\mu}_{y,d}^{EBLUPA}$ of the area means $\mu_{y,d}$, $d = 1, \ldots, D$ is defined as*

$$\begin{aligned}\hat{\mu}_{y,d}^{EBLUPA} &= \hat{\gamma}_d^A \hat{\mu}_{y,d}^{MLGR} + (1 - \hat{\gamma}_d^A)\hat{\mu}_{y,d}^{SynthA} \\ &= \hat{\gamma}_d^A(\hat{\mu}_{y,d}^{MLGR} - \hat{\mu}_{x,d}^T\hat{\beta}) + \hat{\mu}_{x,d}^T\hat{\beta},\end{aligned}$$

where

$$\hat{\gamma}_d^A = \frac{\hat{\sigma}_{v,A}^2}{\hat{\sigma}_{v,A}^2 + \frac{\hat{\sigma}_{e,A}^2}{|s_d|}} \quad \forall d = 1, \ldots, D.$$

The subscript A indicates that the variance components are estimated from the unit level model (2.3) which is used for the synthetic estimator, model A.

The **EBLUPB** is defined analogously. When assuming that y_k is related to x_k through the area level model (2.4) and SynthB is used as synthetic estimator and the GREG as direct estimator, we get the **EBLUPB**.

Definition 2.4.3 (EBLUPB). *Let $\hat{\mu}_{y,d}^{GR}$ be the GREG estimator and $\hat{\mu}_{y,d}^{SynthB}$ be the synthetic estimator, model B for the mean of the variable of interest y in area/domain d denoted by $\mu_{y,d}$. Then the* **EBLUPB** *$\hat{\mu}_{y,d}^{EBLUPB}$ of the area means $\mu_{y,d},\ d = 1, \ldots, D$ is defined as*

$$\begin{aligned}\hat{\mu}_{y,d}^{EBLUPB} &= \hat{\gamma}_d^B \hat{\mu}_{y,d}^{GR} + (1 - \hat{\gamma}_d^B)\hat{\mu}_{y,d}^{SynthB} \\ &= \hat{\gamma}_d^B (\hat{\mu}_{y,d}^{GR} - \hat{\mu}_{x,d}^T \hat{\beta}) + \hat{\mu}_{x,d}^T \hat{\beta},\end{aligned}$$

where

$$\hat{\gamma}_d^B = \frac{\hat{\sigma}_{v,B}^2}{\hat{\sigma}_{v,B}^2 + \frac{\hat{\sigma}_{e,B}^2}{|s_d|}} \quad \forall d = 1, \ldots, D.$$

The subscript B indicates that the variance components are estimated from the unit level model (2.4). However, one has to keep in mind that the estimation of the variance of the units, i.e., $\hat{\sigma}_{e,B}^2$, can only be done by making further assumptions.

One lack of the EBLUPA is that the underlying unit level model does not take information of the sampling design into account. To omit this issue, You and Rao (2002) developed an estimator that takes this information into account by using the given sampling weights. Furthermore, the estimation of the parameters of the model is done with additional constraints such that the benchmark of the population total is satisfied.

Definition 2.4.4 (YOURAO). *Let $d_k = \pi_k^{-1}$ for all $k \in U$ and let the adjusted weights $\tilde{w}_k$ be defined as follows:*

$$\tilde{w}_k = \frac{d_k}{\sum_{l \in s_d} d_l} \quad \forall k \in U_d,\ d = 1, \ldots, D.$$

Furthermore, let the direct estimator $\tilde{\mu}_{y,d}$ for the mean of the variable of interest y in area d be defined as

$$\tilde{\mu}_{y,d} = \sum_{k \in s_d} \tilde{w}_k y_k \quad \forall d = 1, \ldots, D.$$

Then the **YOURAO estimator** *$\hat{\mu}_{y,d}^{YOURAO}$ of the area means $\mu_{y,d},\ d = 1, \ldots, D$ is defined as*

$$\hat{\mu}_{y,d}^{YOURAO} = \hat{\gamma}_{d,\tilde{w}} \tilde{\mu}_{y,d} + (\mu_{x,d} - \hat{\gamma}_{d,\tilde{w}} \tilde{\mu}_{x,d})^T \hat{\beta}_{\tilde{w}},$$

where

$$\begin{aligned}\hat{\gamma}_{d,\tilde{w}} &= \frac{\hat{\sigma}_v^2}{\hat{\sigma}_v^2 + \hat{\sigma}_e^2 \tilde{\delta}_d^2} \quad \forall d = 1, \ldots, D, \\ \tilde{\delta}_d^2 &= \sum_{k \in s_d} \tilde{w}_k^2 \quad \forall d = 1, \ldots, D,\end{aligned}$$

$$\hat{\beta}_{\tilde{w}} = \left(\sum_{d=1}^{D} \tilde{\gamma}_{d,\tilde{w}} \tilde{\mu}_{x,d} \tilde{\mu}_{x,d}^T \right)^{-1} \left(\sum_{d=1}^{D} \tilde{\gamma}_{d,\tilde{w}} \tilde{\mu}_{x,d} \tilde{\mu}_{y,d} \right).$$

For a detailed discussion of the YOURAO estimator we refer to the original work by You and Rao (2002). All estimators above were tested for the applicability in the German Census 2011. The results of this test can be found in Münnich, Gabler, Ganninger, Burgard and Kolb (2012).

2.5 Evaluation of Simulation Results

In the previous sections we mentioned different estimators which have certain properties. It is desirable to know, how 'good' these estimators are and whether they estimate the true value with an appropriate precision. Therefore, we need the mean and the variance of the estimator $\hat{t}_d$ in terms of Monte Carlo methods (cf. Burgard and Münnich, 2012). $\hat{t}_{d,k}$ denotes the estimate in domain d derived from sample $k = 1, \ldots, m$ so the mean and the variance are estimated as follows:

$$\hat{t}_{d,mean} = \frac{1}{m} \sum_{k=1}^{m} \hat{t}_{d,k},$$

$$\hat{t}_{d,var} = \frac{1}{m-1} \sum_{k=1}^{m} (\hat{t}_{d,k} - \hat{t}_{d,mean})^2.$$

If we want to measure the difference between an estimator and the true value t_d scaled to the true value, we make use of the relative root mean squared error. Note that it takes only nonnegative values.

Definition 2.5.1 (RRMSE). *The* **relative root mean squared error** *of the estimator* $\hat{t}_d$ *in domain* d *is defined as*

$$RRMSE_d = \frac{\sqrt{\hat{t}_{d,var} + (\hat{t}_{d,mean} - t_d)^2}}{t_d},$$

where t_d *denotes the true value.*

The relative bias is another error measure that shows how the estimated values are biased over the sample.

Definition 2.5.2 (RBias). *The* **Rbias** *of the estimator $\hat{t}_d$ in domain d is defined as*

$$RBias_d = \frac{\hat{t}_{d,mean} - t_d}{t_d},$$

where t_d denotes the true value.

If $RBias_d = 0$ the estimator is unbiased over the sample. Otherwise, a positive/negative relative bias indicates, that the estimator overestimates/underestimates the true value in average. Applications of the measures mentioned before are given in Chapter 7.

Chapter 3

Optimal Allocation Problems in Statistics

> *Just in terms of allocation of time resources, religion is not very efficient. There's a lot more I could be doing on a Sunday morning.*
>
> — Bill Gates
> *TIME Magazine, Vol. 149*

Given a fixed amount of a certain item, the task of a resource allocation problem is to determine its allocation to a certain number of activities in such a way that the given objective function under consideration is optimized. In the case of Bill Gates, the item 'time on a Sunday morning' has to be allocated to certain activities like going to church, staying in bed, playing football with his son or playing golf. From his statement we cannot conclude which activity optimizes his objective function, but we can deduce that going to church does not.

These resource allocation problems are special cases of nonlinear programing and have many applications in science, e.g., load distribution, production planing, computer scheduling, military, (survey) statistics or portfolio selection (cf. Markowitz, 1952). All these allocation problems can be written as

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} f(x_1,\dots,x_n) \\ &s.t. \sum_{k=1}^{n} x_k = T \\ &\quad m_k \le x_k \le M_k \quad \forall k = 1,\dots,n, \end{aligned}$$

and mainly differ in the given objective function, which can be separable, separable and convex, minimax, maximin, fractional or fair, and whether the solution is required to be continuous or integer valued. Depending on these factors, different algorithms are appropriate.

Regarding separable convex differentiable functions and the corresponding **continuous** optimization problem, there exist two main approaches: pegging algorithms and Lagrange multiplier algorithms (cf. Patriksson, 2008). Pegging algorithms are iterative algorithms that in each iteration solve a relaxation of the original problem, that is without box constraint, and fix the outlying variables of the solution on the values of the box constraint. Afterwards, the problem is reduced by removing the fixed variables and the value of the equality constraint is reduced. This procedure is repeated until no variable exceeds the box constraint anymore. Furthermore, there exists also a projected pegging method and as the Lagrange multiplier is

implicitly optimized in the process, this method is sometimes referred to as primal algorithm. Lagrange multiplier algorithms have an older history than pegging algorithms and utilize the simple form of the Karush-Kuhn-Tucker conditions or the Lagrangian dual problem which has only one variable. The search for the optimal value of the Lagrange multiplier is done by an easy line search, e.g., bisection method, where the choice of the method applied depends on the structure of the equation to be solved. This class of algorithms is sometimes referred to as a dual one. For a detailed discussion of the approaches mentioned in this paragraph we refer to Ibaraki and Katoh (1988), where also methods and applications for other target functions are treated.

Applications of the pegging method to stratified sampling can be found in Sanathanan (1971), where a multistage sampling problem is regarded. Further, Bretthauer et al. (1999) make use of a pegging and a Lagrange multiplier method for solving continuous subproblems of their branch and bound algorithm that solves integer stratified sampling problems. The Lagrange multiplier method is also applied to stratified sampling in Srikantan (1963), where convergence and optimality of the continuous solution is proved. Sanathanan (1971) also uses a Lagrange multiplier approach for stratified sampling. Apart from these references, there are many other articles dealing with allocation problems. These can be found in the extensive overview given in Patriksson (2008).

When dealing with **integer** allocation problems, things get more difficult. A simple rounding of the continuous solution may lead to an infeasible solution and in general does not deliver the optimal solution. Therefore special algorithms have to be applied and in the case that the objective function is separable and convex, a simple greedy fashioned algorithm can be applied. As mentioned in Ibaraki and Katoh (1988), this type of algorithm is also called incremental or marginal allocation algorithm and proceeds as follows. Given an initial vector of the lower bounds, one unit of resource is assigned each iteration to the most favorable activity (in the sense of minimizing the increase of the current objective value) under consideration of the upper bounds and until $\sum_{k=1}^{n} x_k = T$. Although this method does not have a polynomial running time, there exist polynomial time algorithms for solving such allocation problems. Groenevelt (1991) studies allocation problems with separable objective functions over a polymatroid and apart from a marginal allocation algorithm also proposes a 'decomposition algorithm', which is especially suited for polymatroids that are implicitly defined by some generating structure, and a 'bottom up algorithm', which is useful when a polymatroid feasible region is defined by an explicit list of constraints. Hochbaum (1994) revisits the greedy idea for polymatroidal constraints and shows that a greedy algorithm can be applied with arbitrary increments, rather than unit increments. In each iteration, the given increment is assigned to the most favorable activity until no such increments are possible. Then, this process of scaled greedy increments is repeated with smaller increments. It is worth noting that a polynomial running time can be shown for all allocation problems.

As we will see later on, the methods presented by Groenevelt and Hochbaum can be applied to stratified sampling with integer constraints and deliver quite good results. Furthermore, Bretthauer et al. (1999) propose two branch and bound methods for solving integer stratified sampling problems. These algorithms mainly differ in the method utilized to solve the continuous subproblems in each tree, namely a pegging method and a Lagrange multiplier method.

We concentrate on optimal allocation problems in stratified sampling and propose a different derivation of the Lagrange method as well as a fixed point iteration for solving the continuous allocation problem. Further, we compare different root finding algorithms needed for the Lagrange method in terms of computing time and number of iterations. As in survey sampling you cannot select half a person or address, we also propose algorithms for solving the integer allocation problem. These algorithms are again applied to a numerical example so that computational aspects as well as the relation of the integer solution compared to the rounded continuous solution can be analyzed.

3.1 Allocation Problems in Statistics

In classical survey statistics the randomization of a random variable is introduced by the sample selection scheme. The standard estimator for the total t of a variable of interest y in a finite population U of size N is the Horvitz-Thompson estimator

$$\hat{t}_y^{HT} = \sum_{k \in s} d_k y_k,$$

where s is the sample with size $n^s = |s|$ and $d_k = \pi_k^{-1}$ denotes the first order inclusion probability (cf. Särndal et al., 2003). In general, one seeks a sampling design which minimizes the variance of the estimator $V(\hat{t}_y)$. One standard approach which avoids the use of sophisticated inclusion probabilities is stratified random sampling where all population units are uniquely split into H groups, the strata. This procedure is called stratification and is done for various reasons (cf. Cochran, 1977).

(i) If data of known precision are wanted for certain subdivisions of the population, it is advisable to treat each subdivision as a 'population' in its own right.

(ii) Administrative convenience may dictate the use of stratification; for example, the agency conducting the survey may have field offices, each of which can supervise the survey for a part of the population.

(iii) Sampling problems may differ markedly in different parts of the population. With human populations, people living in institutions are often placed in a different stratum from people living in ordinary homes because a different approach to the sampling is appropriate for the two situations.

(iv) Stratification may produce a gain in precision in the estimates of characteristics of the whole population. It may be possible to divide a heterogeneous population into subpopulations, each of which is internally homogeneous. This is suggested by the name 'strata', with its implication of a division into layers. If each stratum is homogeneous, in that the measurements vary little from one unit to another, a precise estimate of any stratum mean can be obtained from a small sample in that stratum. These estimates can then be combined into a precise estimate for the whole population.

Concerning stratified random sampling, the Horvitz-Thompson estimator then simplifies to

$$\hat{t}_y^{SRS} = \sum_{h=1}^{H} \frac{N_h}{N} \mu_{y,h},$$

where N_h is the number of units in stratum h and $\mu_{y,h}$ the corresponding sample mean of the variable of interest y. The variance of the estimator $\hat{t}_y^{SRS}$ is given by

$$V(\hat{t}_y^{SRS}) = \sum_{h=1}^{H} \frac{N_h^2 S_h^2}{n_h} \left(1 - \frac{n_h}{N_h}\right) \tag{3.1}$$

with inferential stratum variances S_h^2 from the universe. The total sample size will then have to be divided into H stratum specific sample sizes n_h which describes an allocation problem. Minimizing the variance of the estimator in equation (3.1) under the equality constraint

$$\sum_{h=1}^{H} n_h = n^s$$

yields the so-called optimal allocation by Neyman (1934) and Tschuprow (1923)

$$n_h = \frac{N_h S_h}{\sum_{k=1}^{H} N_k S_k} n^s \tag{3.2}$$

via a standard Lagrangian approach. In a recent paper, Choudhry et al. (2012) also investigate other optimal allocations in the context of small area estimation and propose a nonlinear programming method for obtaining optimal sample allocations to strata under stratified sampling. The gained solution minimizes the total sample size subject to specified tolerances on the coefficient of variation of estimators of strata means and the population mean.

Though optimal allocations minimize the variance of the estimator of interest, some peculiarities may occur. First, in sampling without replacement it may happen that the optimal allocation yields stratum-specific sample sizes n_h exceeding the number of available units N_h which by definition is not allowed. Furthermore, especially when a huge gain in efficiency is observed, some strata suffer from extremely low sample sizes, thus it is desirable to assure minimal stratum specific sample sizes or fractions.

Regarding business surveys, the optimal allocation often delivers stratum-specific sample sizes that lead to a large spread of the design weights. This spread makes it very hard to get

good estimates. Apart from survey statistics, a recent discussion on the use of survey weights d_k occurred in Gelman (2007b). It was pointed out that large variations of survey weights may not be compensated properly in statistical modeling, especially in Bayesian statistics. In stratified random sampling the design weights are given by $d_k = N_h/n_h$, if the observation k is in stratum h. The variation of design weights is then given by

$$\max_{k,l=1,\ldots,H} \frac{N_k \cdot n_l}{N_l \cdot n_k}. \tag{3.3}$$

For other sampling designs as well as the interplay of modeling and survey weighting we refer to Burgard et al. (2013).

In order to limit this variation or to enable minimal or maximal sampling fractions, bounds on the variables n_h are introduced, such as

$$m_h \leq n_h \leq M_h \quad \forall h = 1, \ldots, H.$$

In addition to that, the total sample size is restricted by n^s, so we get

$$\sum_{h=1}^{H} n_h \leq n^s.$$

3.2 Mathematical Formulation of the Allocation Problem

Rewriting the objective function from the previous section without constant terms yields the following optimization problem.

$$\begin{aligned} \min_{n \in \mathbb{R}_+^H} \; & \sum_{h=1}^{H} \frac{d_h^2}{n_h} \\ s.t. \; & \sum_{h=1}^{H} n_h \leq n^s \\ & m_h \leq n_h \leq M_h \quad \forall h = 1, \ldots, H, \end{aligned} \tag{3.4}$$

where $n := (n_1, \ldots, n_H)^T \in \mathbb{R}_+^H$ defines the sample size in the different strata $h \in \{1, \ldots, H\}$. Note that in this work $\mathbb{R}_+ = \{x \in \mathbb{R} : \; x > 0\}$. The whole sample size is given by n^s and d_h is defined as the product of the known stratum variance S_h^2 and the population size N_h of stratum h. The upper and lower bounds for the sample size of each stratum h are defined as $M_h > m_h > 0$. In shorter notation this problem leads to our model problem

$$\begin{aligned} \min_{n \in \mathbb{R}_+^H} \; & f(n) \\ s.t. \; & g(n) \leq 0 \\ & n \in U, \end{aligned} \tag{3.5}$$

where f is a separable function, i.e.,

$$f : \mathbb{R}_+^H \to \mathbb{R}_+, \quad n \mapsto f(n) = \sum_{h=1}^{H} f_h(n_h).$$

The components f_h are then defined as

$$f_h : \mathbb{R}_+ \to \mathbb{R}_+, \; n_h \mapsto \frac{d_h^2}{n_h} \quad \forall h = 1, \ldots, H,$$

with given $d_h > 0, \; e = (1, \ldots, 1)^T$ and

$$g : \mathbb{R}_+^H \to \mathbb{R}, \; n \mapsto g(n) = n^T e - n^s,$$

where $n^s > 0$ and

$$U = \{n \in \mathbb{R}^H : \; m_h \leq n_h \leq M_h \; \forall h = 1, \ldots, H\}.$$

Before discussing possible methods for solving the allocation problem, we will have a look at some assumptions that guarantee solvability.

Theorem 3.2.1. *Assume that*

$$d_h \neq 0, \quad \forall h = 1, \ldots, H, \tag{3.6}$$

$$0 < m_h < M_h < \infty, \quad \forall h = 1, \ldots, H, \tag{3.7}$$

$$\sum_{h=1}^{H} m_h \leq n^s < \sum_{h=1}^{H} M_h. \tag{3.8}$$

Then we have:

(i) The objective function f is strictly convex and in each component f is strictly monotonically decreasing.

(ii) The feasible set is non-empty and the solution n^* of the optimization problem (3.5) is unique.

(iii) The inequality constraint at the solution is active, i.e., $g(n^*) = 0$.

Proof. (i) The second derivative of f, i.e., the Hessian of f, is given by the diagonal matrix

$$\nabla^2 f(n) = 2 \cdot \operatorname{diag}(d_1^2/n_1^3, \ldots, d_H^2/n_H^3)$$

which is positive definite due to (3.6). Hence, the function f is strictly convex.
The gradient of f is given by

$$\nabla f(n) = -(d_1^2/n_1^2, \ldots, d_H^2/n_H^2)^T$$

and therefore in each component n_h the function f is strictly monotonically decreasing.

(ii) The non-emptiness holds due to (3.8). Furthermore, the feasible set is compact and f is continuous, so a solution of the optimization problem exists. It is unique because f is strictly convex.

(iii) If we assume that the optimal point satisfies $g(n^*) < 0$, then using assumption (3.8) we have

$$\sum_{h=1}^{H} n_h^* < n^s < \sum_{h=1}^{H} M_h.$$

Hence, there exists an index $\hat{h}$ with $n_{\hat{h}}^* < M_{\hat{h}}$. Let us choose $\alpha = \min\{M_{\hat{h}} - n_{\hat{h}}^*, -g(n^*)\} > 0$ and $\tilde{n} = (n_1^*, \ldots, n_{\hat{h}-1}^*, n_{\hat{h}}^* + \alpha, n_{\hat{h}+1}^*, \ldots, n_H^*)$. Since f is decreasing in the $\hat{h}$-th component, we obtain $f(\tilde{n}) < f(n^*)$, a lower value than for n^*. The point $\tilde{n}$ is also feasible, because $g(\tilde{n}) = g(n^*) + \alpha \leq 0$ and $\tilde{n}_{\hat{h}} = n_{\hat{h}}^* + \alpha \leq n_{\hat{h}}^* + M_{\hat{h}} - n_{\hat{h}}^* = M_{\hat{h}}$. Therefore, n^* is not optimal, which is a contradiction. Hence the inequality constraint is active at the optimal point.

□

3.3 Solution of the Continuous Allocation Problem

Regarding the continuous allocation problem in stratified sampling, we can apply the pegging method as mentioned in Sanathanan (1971) or a Lagrangian approach given in Srikantan (1963). The proof of the Lagrange method is usually done via Karush-Kuhn-Tucker conditions. We (Münnich, Sachs and Wagner, 2012c) present an approach using the normal cone instead of complementarity conditions for including the box constraint. This leads to a non-differentiable equation for which we compare different root finding algorithms. If the optimal multiplier is known, we can easily determine the solution vector n^* of the optimization problem. Furthermore, we derive a fixed point formulation for the optimal Lagrange multiplier and use a fixed point iteration based on this formula. This algorithm has the advantage that, if the Lagrange iterates are close enough to the solution, only one additional iteration is needed and the algorithm terminates with the solution.

In Stenger and Gabler (2005), they assume an ordering of the coefficients of the objective function and that the lower bounds are zero. Then they conclude that there is an index i^* such that the optimal solution is at the maximal value for all indices larger than i^*. The solution also guarantees an optimal allocation in without replacement sampling. An extension and generalization of this approach is presented in Gabler et al. (2012) where a problem similar to our case is considered and, under an ordering assumption on the coefficients, the existence of a set partition into active and non-active constraints with optimal value is given. Stefanov (2006) considers optimization problems of the type considered here from a general point of view. Hohnhold (2009) also minimizes a separable function under a box constraint and an equality constraint. He gives a necessary condition for the solution in dependence on the Lagrange multiplier and shows that this reduces to a root finding problem.

If one solves the optimization problem with standard optimization methods, like interior point methods or sequential quadratic programming (SQP) methods (cf. Nocedal and Wright, 2006), one will not use the special structure of this problem which we exploit below.

Solution via Lagrange multiplier method

The main goal in the following approach is to express n as a function depending on the Lagrange multiplier λ. Then, this expression $n(\lambda)$ is inserted into the function g which leads to a one-dimensional equation. The only disadvantage is that the resulting function is continuous but not necessarily differentiable, thus only basic root finding algorithms can be used. First, we define the normal cone which will be used in the Karush-Kuhn-Tucker conditions.

Definition 3.3.1 (Normal cone)**.** *The* **normal cone** *to U in n^* is defined as*

$$N_U(n^*) := \{y \in \mathbb{R}^H : \ y^T x \leq 0 \ \forall x = \gamma z : \ z \in (U - n^*), \ \gamma > 0\}.$$

This leads to the following Karush-Kuhn-Tucker conditions.

Theorem 3.3.2. *A vector $n^* \in \mathbb{R}^H$ is a minimum of problem* (3.5) *if and only if there exists a Lagrange multiplier $\lambda^* \in \mathbb{R}_+ \cup \{0\}$ such that*

$$0 \in \nabla f(n^*) + \lambda^* \nabla g(n^*) + N_U(n^*), \tag{3.9}$$

and furthermore

$$\lambda^* g(n^*) = 0. \tag{3.10}$$

Proof. See Theorem 3.25 in Ruszczynski (2006). Note that the constraint function g is affine and U is a convex polyhedron. Therefore, a constraint qualification condition is satisfied. Furthermore, the objective function f is strictly convex on $\mathbb{R}_+^H$ and the feasible set is convex due to Theorem 3.2.1,(i). Moreover, the necessary optimality condition is also sufficient. □

Regarding those Karush-Kuhn-Tucker conditions, we can easily check the equivalence of the first condition to the following equation.

Lemma 3.3.3. *Under the given assumptions of Theorem 3.3.2, equation* (3.9) *is equivalent to*

$$\begin{aligned} 0 &\geq -\tfrac{d_h^2}{{M_h}^2} + \lambda^* \quad \text{if } n_h^* = M_h, \\ 0 &= -\tfrac{d_h^2}{{n_h^*}^2} + \lambda^* \quad \text{if } n_h^* \in (m_h, M_h), \\ 0 &\leq -\tfrac{d_h^2}{{m_h}^2} + \lambda^* \quad \text{if } n_h^* = m_h. \end{aligned} \tag{3.11}$$

Proof. In this particular setting with the definition of U it is easy to show that

$$N_U(n^*) = \{y \in \mathbb{R}^H : \begin{cases} y_h \geq 0, & \text{if } n_h^* = M_h, \\ y_h = 0, & \text{if } n_h^* \in (m_h, M_h), \} \\ y_h \leq 0, & \text{if } n_h^* = m_h. \end{cases}$$

Taking this into consideration, there exists $\lambda^* \in \mathbb{R}_+ \cup \{0\}$ such that equation (3.9) can be reformulated as

$$-\nabla f(n^*) - \lambda^* \nabla g(n^*) \in \{y \in \mathbb{R}^H : \begin{cases} y_h \geq 0, & \text{if } n_h^* = M_h, \\ y_h = 0, & \text{if } n_h^* \in (m_h, M_h), \} \\ y_h \leq 0, & \text{if } n_h^* = m_h. \end{cases} \tag{3.12}$$

Note that $\nabla f(n^*)_h = -d_h^2/{n_h^*}^2$ and $\nabla g(n^*)_h = 1$ for all h. Then equation (3.12) is equivalent to the following three cases for $h = 1, ..., H$:

$$\begin{aligned} 0 &\geq -\tfrac{d_h^2}{{M_h}^2} + \lambda^* \quad \text{if } n_h^* = M_h, \\ 0 &= -\tfrac{d_h^2}{{n_h^*}^2} + \lambda^* \quad \text{if } n_h^* \in (m_h, M_h), \\ 0 &\leq -\tfrac{d_h^2}{{m_h}^2} + \lambda^* \quad \text{if } n_h^* = m_h, \end{aligned}$$

which completes the proof.

Furthermore, we can state that because of f being strictly monotonically decreasing in each component and $n^s < \sum_{h=1}^{H} M_h$, the Lagrange multiplier λ^* is strictly positive. Therefore, from $\lambda^* g(n^*) = 0$ follows $g(n^*) = 0$. □

As stated before, the inequality constraint holds with equality in the optimal solution. Therefore, the second condition of the Karush-Kuhn-Tucker conditions can be rewritten as

$${n^*}^T e - n^s = 0. \tag{3.13}$$

Revisiting equation (3.11), we can reformulate all conditions by using λ as a variable and then define n depending on the choice of λ. To achieve this we set

$$n : \mathbb{R}_+ \to \mathbb{R}_+^H, \ \lambda \mapsto n(\lambda),$$

with

$$n_h(\lambda) = \begin{cases} M_h, & \text{if } \lambda \leq \frac{d_h^2}{M_h^2}, \\ \frac{d_h}{\sqrt{\lambda}}, & \text{if } \frac{d_h^2}{M_h^2} < \lambda < \frac{d_h^2}{m_h^2}, \\ m_h, & \text{if } \lambda \geq \frac{d_h^2}{m_h^2}. \end{cases} \tag{3.14}$$

Theorem 3.3.4. *A vector $n^* \in \mathbb{R}^H$ is the unique solution of the optimization problem* (3.5) *if and only if there exists a multiplier $\lambda^* \in \mathbb{R}_+$ such that $n(\lambda^*)$ defined in* (3.14) *satisfies*

$$g(n(\lambda^*)) = 0. \tag{3.15}$$

Proof. Since we proved in Theorem 3.2.1,(iii) that the inequality constraint is always active at the solution, equation (3.10) of Theorem 3.3.2 is satisfied in the form of $g(n^*) = 0$. Therefore, (3.15) has to be satisfied. If (n^*, λ^*) is given such that (3.11) holds, then we define $n(\lambda^*)$ and with the three given cases it is easy to check that $n(\lambda^*) = n^*$. On the other hand, if for some λ^* the vector $n(\lambda^*)$ satisfies (3.15), then by a quick verification we see that $(n(\lambda^*), \lambda^*)$ also satisfies (3.11). This completes the proof. □

The last theorem states that the solution of the optimization problem is equivalent to solve the equation

$$\tilde{g}(\lambda) := g(n(\lambda)) = n(\lambda)^T e - n^s = 0.$$

Since $\tilde{g}$ is continuous but not differentiable, the equation has to be solved by methods which only require continuity. There exists a wide range of adequate methods which are mentioned after the following remarks. Further, detailed information concerning those methods can be found in Ralston and Rabinowitz (1978).

Remark 3.3.5. *All the arguments can also be applied to the following optimization problem:*

$$\begin{aligned} \min_{n \in \mathbb{R}_+^H} & \sum_{h=1}^{H} \frac{d_h^2}{n_h} \\ s.t.\ & n^T p - n^s \leq 0 \\ & n \in U, \end{aligned}$$

where p defines a vector of penalty, cost or weighting parameters.
Then the solution of the problem without a box constraint can be specified as

$$n_h^* = \frac{n^s}{\sum_{i=1}^{H} d_i \sqrt{p_i}} \cdot \frac{d_h}{\sqrt{p_h}},$$

and for all n_h as function depending on λ

$$n_h(\lambda) = \begin{cases} M_h, & \text{if } \lambda \leq \frac{d_h^2}{M_h^2 p_h}, \\ \frac{d_h}{\sqrt{\lambda p_h}}, & \text{if } \frac{d_h^2}{M_h^2 p_h} < \lambda < \frac{d_h^2}{m_h^2 p_h}, \\ m_h, & \text{if } \lambda \geq \frac{d_h^2}{m_h^2 p_h}. \end{cases}$$

Remark 3.3.6. *More general, we can also consider an optimization problem with a separable objective function. Therefore, we look at the following optimization problem.*

$$\begin{aligned} \min_{n\in\mathbb{R}_+^H} & \sum_{h=1}^{H} f_h(n_h) \\ s.t.\ & n^T e - n^s \leq 0 \\ & n \in U, \end{aligned}$$

where f_h is a differentiable, strictly monotonically decreasing, convex function. Then the solution n_h as function depending on λ satisfies

$$n_h(\lambda) = \begin{cases} M_h, & \text{if } \lambda \leq -f_h'(M_h), \\ (f_h')^{-1}(-\lambda), & \text{if } -f_h'(M_h) < \lambda < -f_h'(m_h), \\ m_h, & \text{if } \lambda \geq -f_h'(m_h). \end{cases}$$

The easiest method for solving a nonlinear equation $g(x) = 0$ is the bisection method (Algorithm 3.1) which is very robust and its convergence rate is linear with convergence factor $\frac{1}{2}$. Since λ has to be positive and g is defined on $\mathbb{R}_+$, the left bound always has to be positive.

Algorithm 3.1 Bisection method

Input: $g : \mathbb{R} \to \mathbb{R}$ continuous on $[a, b] \subset \mathbb{R}$ and $g(a)g(b) < 0$
  set initial iterate $x = a$
  while $|g(x)| \geq \epsilon$ **do**
    $x := \frac{a+b}{2}$
    if $g(a)g(x) > 0$ **then**
      set $a \leftarrow x$
    else
      set $b \leftarrow x$
    end if
  end while
  return root x

The secant method (Algorithm 3.2) is superlinearly convergent under certain differentiability conditions on the function g which unfortunately do not hold for our application. Nevertheless, as we will see later on, the numerical results for this method are satisfactory.

Algorithm 3.2 Secant method

Input: $g : \mathbb{R} \to \mathbb{R}$ continuous on $[a, b] \subset \mathbb{R}$ and $g(a)g(b) < 0$
Ensure: $x^0, x^1 \in (a, b)$ close to the optimal solution
  for $k = 1, 2, \ldots$ **do**
    if $|g(x^k)| \leq \epsilon$ **then**
      stop
    end if
    $x^{k+1} := x^k - \frac{x^k - x^{k-1}}{g(x^k) - g(x^{k-1})} g(x^k)$
  end for
  return root x^k

Another applicable method is the regula falsi (Algorithm 3.3). As the simple regula falsi does not perform as well as the secant method we look for extensions of the regula falsi.

Algorithm 3.3 Regula falsi

Input: $g : \mathbb{R} \to \mathbb{R}$ continuous on $[a, b] \subset \mathbb{R}$ and $g(a)g(b) < 0$
  set initial iterate $x = a$
  while $|g(x)| \geq \epsilon$ **do**
    $x := \frac{ag(b) - bg(a)}{g(b) - g(a)}$
    if $g(a)g(x) > 0$ **then**
      set $a \leftarrow x$
    else
      set $b \leftarrow x$
    end if
  end while
  return root x

Although one cannot prove the convexity of g, the graph in Figure 3.3 indicates that g has certain convex parts. For convex functions there exist some extensions called the Illinois method (Algorithm 3.4), Pegasus method and an alternative method, which all speed up the convergence.

Algorithm 3.4 Illinois method

Input: $g : \mathbb{R} \to \mathbb{R}$ continuous on $[a, b] \subset \mathbb{R}$ and $g(a)g(b) < 0$
  set initial iterate $x = a$
  while $|g(x)| \geq \epsilon$ **do**
    $x := \frac{ag(b)-bg(a)}{g(b)-g(a)}$
    if $g(a)g(x) > 0$ **then**
      set $a \leftarrow x$
    else
      $\alpha = \frac{1}{2}$
      set $b \leftarrow \frac{\alpha g(a)x}{\alpha g(a)-g(x)} + \frac{g(x)a}{g(x)-\alpha g(a)}$
    end if
  end while
  return root x

If we choose

$$\alpha = \frac{g(b)}{g(b) + g(x)}$$

we get the **Pegasus method**.
An **alternative method** uses the following choice for α.

$$\alpha = \begin{cases} \beta, & \text{if } \beta > 0, \\ \frac{1}{2}, & \text{if } \beta \leq 0, \end{cases}$$
$$\text{with } \beta = \frac{(g(x) - g(b))(b - a)}{(x - b)(g(b) - g(a))},$$

which has a better convergence behavior.

We implemented all methods for the problem under consideration in R and present numerical results in Section 3.5.

Solution via fixed point iteration

Aside from the presented search for a root of the function $\tilde{g}$, the solution of the optimization problem can also be found through a fixed point iteration. As stated before, the following equation has to be solved:

$$\tilde{g}(\lambda) = n(\lambda)^T e - n^s = 0.$$

The components $n_h(\lambda)$ take values M_h, m_h or their values lie in the interval (m_h, M_h). Accordingly, we partition the set $J = \{1, \ldots, H\}$ of indices into three subsets J_M^λ, J_m^λ and

J^λ. Using these subsets the equation is rewritten as

$$\sum_{h\in J^\lambda} \frac{d_h}{\sqrt{\lambda}} + \sum_{h\in J_M^\lambda} M_h + \sum_{h\in J_m^\lambda} m_h - n^s = 0.$$

If we solve this equation for λ, we obtain

$$\lambda = \left(\frac{\sum_{h\in J^\lambda} d_h}{n^s - \sum_{h\in J_M^\lambda} M_h - \sum_{h\in J_m^\lambda} m_h} \right)^2.$$

Then the following function ϕ can be defined as

$$\begin{aligned} &\phi : \mathbb{R}_+ \to \mathbb{R}_+ \\ &\qquad \lambda \mapsto \phi(\lambda) = \left(\frac{\sum_{h\in J^\lambda} d_h}{n^s - \sum_{h\in J_M^\lambda} M_h - \sum_{h\in J_m^\lambda} m_h} \right)^2. \end{aligned}$$

Now we can prove that solving $\tilde{g}(\lambda) = 0$ is equivalent to solving $\lambda = \phi(\lambda)$.

Lemma 3.3.7. *Solving $\tilde{g}(\lambda) = 0$ is equivalent to solving $\lambda = \phi(\lambda)$.*

Proof. One direction has been shown prior to this lemma. If $\lambda^* = \phi(\lambda^*)$, then $n(\lambda^*)$ defined in (3.14) also satisfies $\tilde{g}(\lambda^*) = 0$, which completes the proof. □

Note that ϕ is discontinuous and has jumps. However, since the function ϕ depends on λ only through the subsets of J, it has a remarkable property. Due to (3.14) we can rewrite J_m^λ, J_M^λ as

$$\begin{aligned} J_M^\lambda &= \left\{ h = 1, \dots, H : \ \lambda \le \frac{d_h^2}{M_h^2} \right\}, \\ J_m^\lambda &= \left\{ h = 1, \dots, H : \ \lambda \ge \frac{d_h^2}{m_h^2} \right\}. \end{aligned}$$

Consider the fixed point λ^* and assume that in J_m^λ, J_M^λ the inequalities are strict inequalities. Then it is conceivable that small perturbations of λ^* do not change these sets J_m^λ, J_M^λ and hence the value of ϕ does not change. This means that the optimal λ^* is already delivered as an output of ϕ when the input λ is close to the optimal λ^*.

Lemma 3.3.8. *Let λ^* be the solution of $g(\lambda) = 0$ and assume that*

$$\lambda^* \notin \left\{ \frac{d_h^2}{m_h^2}, \ \frac{d_h^2}{M_h^2} : \ h = 1, \dots, H \right\}.$$

Then there exists $\epsilon > 0$ such that

$$\lambda^* = \phi(\lambda) \quad \forall |\lambda - \lambda^*| < \epsilon.$$

Proof. Due to the assumption, the index sets J_M^λ and J_m^λ can be written as

$$J_M^{\lambda^*} = \left\{ h \in 1, \dots, H : \ \lambda^* < \frac{d_h^2}{M_h^2} \right\},$$
$$J_m^{\lambda^*} = \left\{ h \in 1, \dots, H : \ \lambda^* > \frac{d_h^2}{m_h^2} \right\}.$$

We define

$$\epsilon := \min \left\{ |\lambda^* - \frac{d_h^2}{M_h^2}|, \ |\lambda^* - \frac{d_h^2}{m_h^2}| \ : \ h = 1, \dots, H \right\}.$$

Therefore, for all $\lambda \in (\lambda^* - \epsilon, \lambda^* + \epsilon)$ holds

$$J_M^{\lambda^*} = J_M^\lambda, \ J_m^{\lambda^*} = J_m^\lambda, \ J^{\lambda^*} = J^\lambda.$$

Since the function ϕ depends on λ only through the index sets we have

$$\phi(\lambda) = \phi(\lambda^*) = \lambda^* \quad \forall |\lambda - \lambda^*| < \epsilon.$$

□

If the fixed point iteration converges, then the iteration terminates after finitely many steps with the exact solution λ^*, because the index sets do not change anymore. This yields to the following Algorithm 3.5.

Algorithm 3.5 Fixed point iteration

Input: Starting value $\lambda^0 > 0$ near to the optimal solution
  for k=0,1,2,... **do**
    determine index sets $J_M^{\lambda^k}, J_m^{\lambda^k}, J^{\lambda^k}$
    $\lambda^{k+1} := \left(\frac{\sum_{h \in J^{\lambda^k}} d_h}{n^s - \sum_{h \in J_M^{\lambda^k}} M_h - \sum_{h \in J_m^{\lambda^k}} m_h} \right)^2$
    if $\lambda^{k+1} = \lambda^k$ **then**
      stop
    end if
  end for
  return solution λ^k

In order to find a good starting point λ^0, the easiest way is to use the optimal allocation following Neyman (1934) and Tschuprow (1923) for the above optimization problem without box constraint. The solution of the problem without box constraint is given by

$$n_h = \frac{d_h}{\sum_{k=1}^H d_k} n^s \quad \forall h = 1, \dots, H.$$

From this equation we can derive the multiplier λ^0 as

$$\lambda^0 = \left(\frac{d_h}{n_h}\right)^2 = \frac{1}{(n^s)^2}\left(\sum_{k=1}^{H} d_k\right)^2 .$$

In case that λ^0 is not close enough to the optimal λ^* such that the fixed point iteration does not converge, one can start with some iterations of the methods presented before and afterwards continue with the fixed point iteration.

3.4 Solution of the Integer Allocation Problem

All approaches mentioned so far have one thing in common. The allocation problem is solved in continuous variables, i.e., the integrality conditions on the variables are relaxed. When considering integer allocation problems, things get more difficult. A simple rounding of the continuous solution does in general not deliver the optimal and may even lead to an infeasible solution. Therefore, special algorithms have to be applied.

Before we (Friedrich et al., 2013) will deal with different algorithms in detail, we introduce the concept of polymatroids. These combinatorial structure, which is a generalization of the more widely known matroid, leads to interesting approaches to integer optimization (cf. Schrijver (2003) for a general introduction on the topic). The observation that the feasible region of the allocation problem is a special type of polymatroid is the mathematical reason for the correctness of the methods.

Definition 3.4.1 (Polymatroid)**.** *A* **polymatroid** *is a set of the form*

$$P(\varphi) := \left\{ x \in \mathbb{R}^E : \; x \geq 0, \sum_{e \in A} x_e \leq \varphi(A) \quad \forall A \subseteq E \right\},$$

where E is a finite set and $\varphi : 2^E \to \mathbb{R}_+$ is a monotone, submodular function, i.e.,

$$\varphi(X \cap Y) + \varphi(X \cup Y) \leq \varphi(X) + \varphi(Y) \quad \forall X, Y \subseteq E,$$

and satisfies $\varphi(\emptyset) = 0$.

This definition looks rather complicated, but actually describes a polyhedron with 'nice' properties. In fact, every polymatroid is a bounded polyhedron and Figure 3.1 shows a general polymatroid in $\mathbb{R}^3$.

As already mentioned, greedy strategies are applicable for the minimization of a separable objective function with convex summands when the feasible region is a polymatroid. The easiest form is the simple greedy algorithm stated in Algorithm 3.6. It increases one variable per iteration with respect to the given constraints. Normally, $x = 0$ is chosen as feasible initial iterate, but other choices are possible as long as x is feasible and element-wise smaller than the optimal solution.

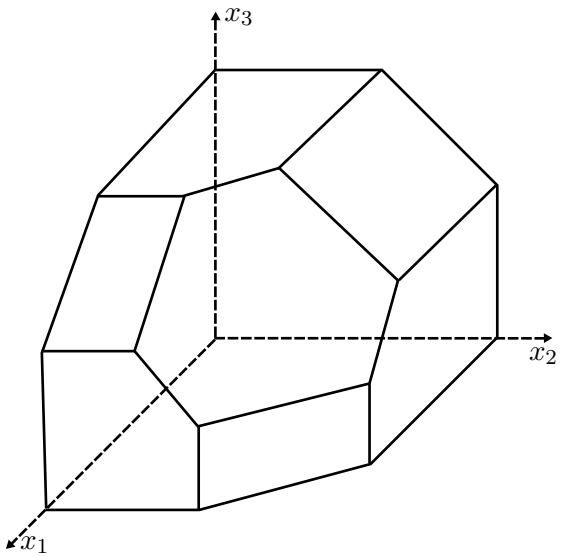


Figure 3.1: Example of a polymatroid

Algorithm 3.6 Simple greedy

Input: $H, n^s \in \mathbb{N},\ d \in \mathbb{R}^H,\ m, M \in \mathbb{N}^H,\ x$ feasible initial iterate
  set $I = \{1, \ldots, H\}$
  while $\sum_{h=1}^{H} x_h \neq n^s - \sum_{h=1}^{H} m_h$ **do**
    compute $\delta_h := \frac{d_h^2}{x_h + m_h + 1} - \frac{d_h^2}{x_h + m_h}$ for all $h \in I$
    determine index h^* such that $\delta_{h^*} = \min_{h \in I}\{\delta_h\}$
    if $x_{h^*} + 1 \leq M_{h^*} - m_{h^*}$ **then**
      set $x_{h^*} \leftarrow x_{h^*} + 1$
    else
      set $I \leftarrow I \setminus \{h^*\}$
    end if
  end while
  return optimal solution x

The correctness of the simple greedy strategy is a consequence of the following result due to Groenevelt (1991).

Theorem 3.4.2. *The simple greedy algorithm finds an integer solution of the problem*

$$\min\left\{ f(x) \mid x \geq 0, \sum_{e\in A} x_e \leq \varphi(A)\ \forall A \subseteq E \right\},$$

where

(i) E *is a finite set,*

(ii) $\varphi : 2^E \to \mathbb{R}_+$ *is submodular, monotone and satisfies* $\varphi(\emptyset) = 0$,

(iii) $f : \mathbb{R}_+^E \to \mathbb{R}$ *is separable with continuous convex components.*

Indeed, if we apply the shift $x_h := n_h - m_h$ for all $h = 1, \ldots, H$ and define φ above by $\varphi(A) = \min\{\sum_{e\in A}(M_e - m_e), n^s - \sum_{e\in E} m_e\}$, it is easy to see that the feasible set of the allocation problem 3.4 is a polymatroid. Furthermore, as f is separable with continuous convex components, the simple greedy algorithm can be used to solve the problem.

However, the major drawback of the simple greedy strategy is the fact that it needs a lot of (numerically cheap) iterations to find the optimum since only increments of one unit per iteration are possible. Hochbaum (1994) presents an elegant refinement that generally uses only a fraction of the number of iterations of the simple strategy. She shows that a greedy algorithm can be applied with arbitrary large increments, rather than unit increments. Starting with an increment of $s > 1$, the increment is assigned to the most favorable activity until no such increments are possible. Then, s is decreased and the process of scaled greedy increments is repeated with the successively smaller increments until the increment equals 1 as in Algorithm 3.6.

We call this procedure given in Algorithm 3.7 capacity scaling. The mathematical finesse of the algorithm lies in Theorem 4.1 of Hochbaum (1994), which guarantees that only the last increase of each variable with an increment higher than one might lead to non-optimal assignments. By canceling this last step (after the end of the inner while-loop of the algorithm), the iterate passed over to the simple greedy algorithm is not only guaranteed to be feasible, but is element-wise smaller than the optimal solution. Therefore, the simple greedy algorithm can reach the optimal solution from this starting point.

Algorithm 3.7 Capacity scaling

Input: $H, n^s \in \mathbb{N},\ d \in \mathbb{R}^H,\ m, M \in \mathbb{N}^H$

set $I = \{1, \dots, H\}$, initial iterate $x = 0$ and initial increment $s = \left\lceil \frac{n^s - \sum_{h=1}^{H} m_h}{2H} \right\rceil$.
while $s > 1$ **do**
  while $\sum_{i=1}^{n} x_h < n^s - \sum_{h=1}^{H} m_h$ and $I \neq \emptyset$ **do**
    compute $\delta_h = \frac{d_h^2}{x_h + m_h + 1} - \frac{d_h^2}{x_h + m_h}$ for all $h \in I$
    determine index h^* such that $\delta_{h^*} = \min_{h \in I}\{\delta_h\}$
    if $x_{h^*} + 1 \leq M_{h^*} - m_{h^*}$ **then**
      if $x_{h^*} + s \leq M_{h^*} - m_{h^*}$ **then**
        set $x_{h^*} \leftarrow x_{h^*} + s$
      else
        set $x_{h^*} \leftarrow x_{h^*} + 1$
        set $I \leftarrow I \setminus \{h^*\}$
      end if
    else
      set $I \leftarrow I \setminus \{h^*\}$
    end if
  end while
  set $x_h \leftarrow \max\{0, x_h - s\}$ for all $h \in \{1, \dots, H\}$
  set $I \leftarrow \{1, \dots, H\}$
  set $s \leftarrow \lceil \frac{s}{2} \rceil$
end while
call simple greedy with initial iterate x
return optimal solution x

We further present a third algorithm which also can be considered a greedy strategy, but abandons the concept of increasing only one variable per iteration. The underlying idea is again quite simple: Algorithm 3.6 above successively increases the locally best variables until the upper bounds are reached. The convexity of the objective implies that the marginal δ_h increases in each iteration of the algorithm.

The key observation is that the entire solution can be reconstructed from the value of the marginal δ_{last} in the last iteration of the algorithm. That is because by computing the marginal at an arbitrary value of an arbitrary variable $n_h = x_h + m_h$ and comparing it to δ_{last}, we can decide if n_h is above or below its value in the optimal solution. Hence, the optimization problem is equivalent to finding δ_{last}, which can easily be done by a binary search. Before we present the algorithm, it is helpful to restate the problem of finding δ_{last} in an illustrative form. Since the H variables n_h of the allocation problem can only attain finitely many different values, all possible values for δ_h can be written down in a (supposedly very large) matrix. Furthermore, these values can be arranged in such a way that each column of the matrix contains the possible marginals of one single variable n_h, starting from the marginal at $n_h = m_h$ in the first row to $n_h = M_h$. If necessary, the rest of each column is filled in with an arbitrary number that is larger than all marginals. Then, because of the convexity of the components of the objective function, all columns are sorted. The Problem

of finding δ_{last} can then be restated as finding the $(n^s - \sum_{h=1}^{H} m_h)$-smallest value among the entries of this matrix. By considering the problem in the way above, it is clear why the paper by Frederickson and Johnson (1982) motivated Algorithm 3.8.

Algorithm 3.8 Binary search

Input: $H, n^s \in \mathbb{N},\ d \in \mathbb{R}^H,\ m, M \in \mathbb{N}^H$
  set $I = \{1, \ldots, H\}$ and initial iterate $n = m$.
  while $\sum_{h=1}^{H} n_h \neq n^s$ **do**
    compute $u_h := \lfloor \frac{M_h + m_h + 1}{2} \rfloor$ and $c_h = \frac{d_h^2}{u_h} - \frac{d_h^2}{u_h - 1}$ for all $h \in I$
    compute $s :=$ lower median$\{c_h \ :\ h \in I\}$
    compute $n_h := \lfloor 0,5 + \sqrt{0,25 - d_h^2 s^{-1}} \rfloor$ for all $h \in I$
    if $n_h < m_h$ **then**
      set $n_h \leftarrow m_h$
    end if
    if $n_h > M_h$ **then**
      set $n_h \leftarrow M_h$
    end if
    if $\sum_{h=1}^{H} n_h < n^s$ **then**
      set $m_h \leftarrow n_h$ for all $h \in I$
    else
      set $M_h \leftarrow n_h$ for all $h \in I$
    end if
    if $m_h = M_h$ **then**
      set $n_h = m_h$
      set $I \leftarrow I \setminus \{h\}$
    end if
  end while
  return optimal solution n

Of course, the explicit construction of the matrix of marginal gains would be very costly in an implementation of this idea. Fortunately, it is not necessary to construct it. Since the objective function in Problem (3.4) has convex monotone summands, there exists an inverse function of the marginal costs of every summand, precisely

$$\frac{d_h^2}{n_h} - \frac{d_h^2}{n_h - 1} = \delta_{last},\ n_h \geq 0 \quad \Leftrightarrow \quad n_h = 0,5 + \sqrt{0,25 - d_h^2 \delta_{last}^{-1}}.$$

With the help of this simple formula, we are able to formulate Algorithm 3.8.

Note that, since the marginals per variable are sorted, the lower median of the remaining possible values for n_h is given by the simple formula $\lfloor \frac{M_h + m_h + 1}{2} \rfloor$. A key reason for the effectiveness of the algorithm is the approximation of the (lower) median of the possible values for all variables by computing the lower median of the lower medians per variable.

3.5 Application to the German Census Sampling and Estimation Research Project

Before applying the methods mentioned in Section 3.3 and 3.4, we will regard the setting and circumstances in the German Census Sampling and Estimation Research Project. Recalling Definition 2.0.4 'A survey is a systematic method for gathering information from (a sample of) entities for the purposes of constructing quantitative descriptors of the attributes of the larger population of which the entities are members.' In the case of observing the whole population, this special type of survey is called census (cf. Särndal et al., 2003).

The last German census was done in 1981 (German Democratic Republic) and 1987 (West Germany), where these were classic censuses. In contrast to this, the German Census 2011 is register-assisted and apart from analyzing the register of residents, a 10% sample is drawn for getting information of register errors (over- and undercounts) and other variables of interest that are not listed in the register. Thus, it is possible to determine the official population figure (goal 1) and get detailed figures on further information like educational background or status of employment (goal 2). As only the people in the sample are interviewed and data from the register is used, this method is cheaper than a classic survey while delivering comparable results. Nevertheless, precision is not only demanded for on state level but also on other levels like governmental units. Therefore, Germany is divided into disjunct domains, the so called sampling points (SMP), what from partial samples are drawn. This guarantees a nationwide distribution of the sample. These sampling points are divided into four types, whose distribution can be seen in Figure 3.2 where

(i) Type 0, (SDT): urban districts with more than 200,000 inhabitants belonging to communities with more than 400,000 inhabitants,

(ii) Type 1, (GEM): communities with at least 10,000 inhabitants as long as they do not belong to type 0,

(iii) Type 2, (VBG): small communities (less than 10,000 inhabitants) in an association of communities summing up to at least 10,000 inhabitants,

(iv) Type 3, (KRS): aggregation of the communities belonging to a district as long as they are not assigned to another type.

Apart from the devision into SMPs, a stratification into eight strata depending on the registered inhabitants at an address is done, where each stratum contains the same number of registered people.

The minimal and maximal sampling fraction yielding the box constraint were defined and are given in Table 3.1. Note that type 2* denotes the SMPs of type 2 belonging to Rhineland-Palatinate. They depend on the SMP type and the community size, where the latter one leads to a division into

(i) Type I: 0 to 10,000 inhabitants,

(ii) Type II: 10,000 to 30,000 inhabitants,

(iii) Type III: 30,000 to 100,000 inhabitants,

(iv) Type IV: more than 100,000 inhabitants.

These sampling fractions allow to gain reliable model estimates from rural vs. urban comparisons where the classical optimal allocation would yield an extremely high sampling fraction in large towns and very low sampling fraction in rural areas. Furthermore, they restrict the variation of design weights which in random sampling are given by $d_k = N_h/n_h$ if the observation k is in stratum h.

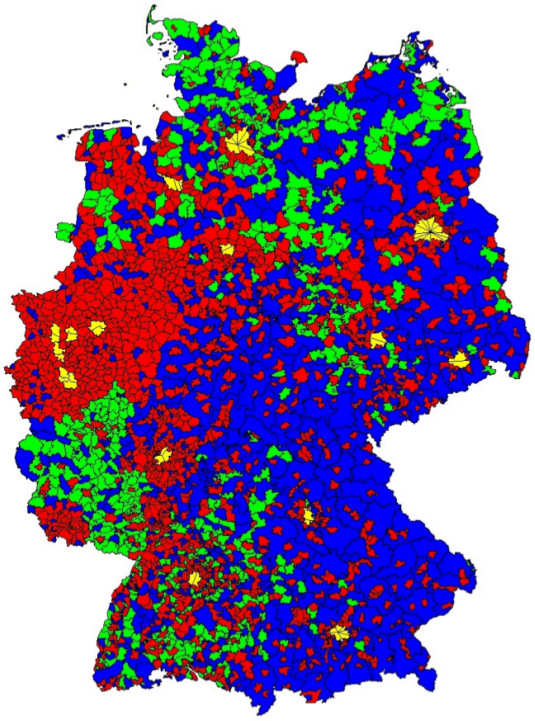


Figure 3.2: Distribution of the SMPs where SDT=yellow, GEM=red, VBG=green, KRS=blue. Source: Münnich, Gabler, Ganninger, Burgard and Kolb (2012)

	type 0		type 1		type 2*		type 2		type 3	
com. size	p_h	P_h	p_h	P_h	p_h	P_h	p_h	P_h	p_h	P_h
I	–	–	–	–	–	–	–	–	0.05	0.05
II	–	–	0.05	0.50	0.05	0.50	0.05	0.05	0.05	0.05
III	–	–	0.04	0.40	0.04	0.40	0.05	0.05	0.05	0.05
IV	0.02	0.40	0.02	0.40	0.02	0.40	0.05	0.05	0.05	0.05

Table 3.1: Sampling fraction in the different SMPs depending on community size

This leads to the following simplified allocation problem

$$\begin{aligned}
&\min_{n} \sum_{g=1}^{2391} \sum_{h=1}^{8} \frac{N_{g,h}^2 S_{g,h}^2}{n_{g,h}} \\
&s.t. \sum_{g=1}^{2391} \sum_{h=1}^{8} n_{g,h} = n^s \\
&\quad m_{g,h} \leq n_{g,h} \leq M_{g,h} \quad \forall g = 1, \ldots, 2391,\ h = 1, \ldots, 8,
\end{aligned}$$

where $S_{g,h}^2$ denotes the stratum and area specific variance. Further explanations and a detailed review can be found in Münnich, Gabler, Ganninger, Burgard and Kolb (2012).

Our simulation study was done with an artificially generated data set representing the German population. This data was also used for the simulation studies in Münnich, Gabler, Ganninger, Burgard and Kolb (2012). For further explanations concerning the generation of the data we refer to Kolb (2012). Instead of 2391 SMPs, our setting consists of 2393 SMPs and for the sake of simplicity our allocation problems is seen as vector valued, such that every subset depending on the SMPs and strata is referred to as a stratum. Therefore, it follows that $n \in \mathbb{R}_+^{19144}$ and each component n_h represents the number of addresses drawn in the stratum h ($h = 1, \ldots, 19144$). Further, the data is simplified in that way, that the lower and upper bounds as well as the desired overall sample size n^s are integer valued and no fixed partial samples exist.

First, we tried to solve our problem with an already implemented algorithm in order to get a benchmark. We chose the 'solnp' algorithm implemented in the 'Rsolnp' package but we were faced with storage problems. The algorithm needs to compute a diagonal matrix with a dimension higher than the number of strata which exceeds the available RAM on a common desktop PC. This problem is avoided by computationally solving the root problem in R or applying greedy-type methods because we do not need matrices or other memory absorbing arrays.

In order to solve the continuous root problem with the different algorithms, we had to choose the upper and lower bounds for λ, i.e., a and b. In case the bounds did not satisfy the sign constraint, they were adjusted by doubling or halving.

The computing effort of the algorithms in R for a problem with 19,144 variables executed on a common desktop PC with an Intel(R) Core(TM)2 Duo CPU with 3.00GHz and an internal memory of 4 GB can be seen in Table 3.2 and Table 3.5. GGM marks an improvement of the method presented by Gabler et al. (2012), which uses the sufficient optimality conditions in the proof in order to avoid ordering and is of linear complexity.

	time [sec]	iterations
bisection method	0.484	30
secant method	0.088	8
regula falsi normal	0.261	17
regula falsi Illinois	0.083	7
regula falsi Pegasus	0.120	7
regula falsi alternative	0.072	4
GGM	0.032	18
fixed point iteration	0.014	4

Table 3.2: Computing time and number of iterations of the continuous methods in R

We can state that the fixed point iteration needs less time and less iterations than the other methods, except for the regula falsi alternative that needs the same number of iterations as the fixed point iteration. However, the fixed point iteration is much faster which can be explained by a differing number of calculations in each iteration of the two methods. This shows a drawback of more sophisticated methods. They may need less or the same number of iterations but each iteration may require more time, which can also be seen in the case of the regula falsi Pegasus and the regula falsi Illinois.

Regarding Table 3.3 we can see that the fixed point iteration also works with other starting points, where $3.511786 \cdot 10^{-8}$ is the computed starting point derived from the optimal allocation following Neyman and Tschuprow.

λ^0	$1.000000 \cdot 10^{-9}$	$3.511786 \cdot 10^{-8}$	$1.000000 \cdot 10^{-5}$
λ^1	$3.413068 \cdot 10^{-8}$	$1.399724 \cdot 10^{-8}$	$5.235127 \cdot 10^{-9}$
λ^2	$1.399394 \cdot 10^{-8}$	$1.358768 \cdot 10^{-8}$	$1.441687 \cdot 10^{-8}$
λ^3	$1.358768 \cdot 10^{-8}$	$1.358735 \cdot 10^{-8}$	$1.358746 \cdot 10^{-8}$
λ^4	$1.358735 \cdot 10^{-8}$	$1.358735 \cdot 10^{-8}$	$1.358735 \cdot 10^{-8}$
λ^5	$1.358735 \cdot 10^{-8}$	–	$1.358735 \cdot 10^{-8}$

Table 3.3: Computed λ^k depending on different starting points λ^0 of the fixed point iteration

The bisection method needs the most iterations because of the special structure of the function g, which can be seen in Figure 3.3. The left bound stays very long at a constant value and in the beginning only the right bound is moving towards the root. We can also state that the function g takes function values between $1.5 \cdot 10^7$ and $-1 \cdot 10^6$ over the small interval of length $3 \cdot 10^{-8}$. Therefore, the function value is very sensitive concerning changes in λ.

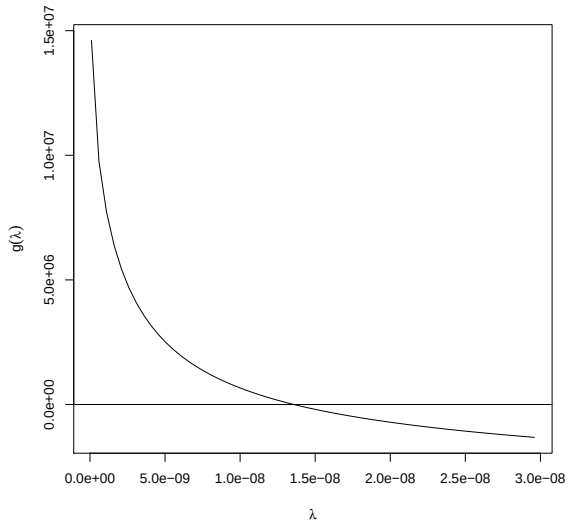


Figure 3.3: Plot of $g(\lambda)$

We also applied our algorithms to a higher dimensional problem, where the computational effort of the differing algorithms is pointed out even stronger than in the problem with 19,444 variables (cf. Table 3.4).

	number of variables			
	$2 \cdot 10^4$	$2 \cdot 10^5$	$2 \cdot 10^6$	$2 \cdot 10^7$
bisection method	0.484	6.617	57.096	960.782
secant method	0.088	1.152	9.518	132.942
regula falsi normal	0.261	4.422	38.694	475.119
regula falsi Illinois	0.083	1.158	10.356	158.492
regula falsi Pegasus	0.120	1.807	14.986	231.269
regula falsi alternative	0.072	0.992	8.192	105.409
GGM	0.032	0.371	3.460	52.236
fixed point iteration	0.014	0.226	1.653	61.860

Table 3.4: Computing time [sec] of the continuous methods in R for different problem sizes

Regarding the computing time and the number of iterations of the integer methods given in Table 3.5, we can state that the simple greedy is the slowest method and needs the most iterations. This is not surprising because in each iteration only one element is assigned. Therefore, the number of iterations equals the number of elements to be assigned, i.e., in the simulations study $n^s - \sum_{h=1}^{19144} m_h = 4567313$.

	time [sec]	iterations
simple greedy	1165.06	4,567,313
capacity scaling	81.80	226,369
binary search	11.64	23

Table 3.5: Computing time and number of iterations of the integer methods in R

Capacity scaling leads to a speed-up in the computation time of factor 14 and needs only 5% of the iteration steps of the simple greedy method. These 226,369 iterations are split into 203,785 iterations with an increment greater one and 22,584 iterations with an increment of one, i.e., 22,584 steps of the simple greedy are performed.

The fastest method concerning computation time and needed iterations is the binary search which only needs 11.64 seconds to solve the allocation problem. This means an enormous speed-up of factor 100 and regarding the number of iterations, the binary search only needs 23 iterations which is precious little.

Almost the same properties can be encountered regarding the implementations of the algorithms in `C++`. Further, due to the general advantage of being much faster than R and the use of sophisticated object structures like heaps, the computing time of the implementations in `C++` could be reduced to a minimum (cf. Table 3.6).

	time [sec]	iterations
simple greedy	3.82	4,567,313
capacity scaling	0.58	283,457
binary search	0.10	22

Table 3.6: Computing time and number of iterations of the integer methods in `C++`

3.6 Rounding Impacts

One possible solution to overcome the non-integrity of the continuous solution is simple rounding. However, the rounded solution does in general not coincide with the integer solution, which in fact is the optimal solution. In the simulation study, the rounded solution leads to an allocation with 25 elements less than the allocation determined by the greedy-type algorithms. This means a variation of 0.0003% to the equality constraint, which is quite small. Rounding up every partial sample leads to an allocation with 4816 elements more than the allowed 7,900,000 given by n^s, where rounding down leads to an allocation with 4876 elements missing. The problem of satisfying the equality constraint exactly can be tackled by an intelligent rounding procedure, whereas we will not deal with this topic in this work.

Having a look at the rounded solution compared to the integer solution, we can state that the difference is spread to 25 SMPs which are all missing exactly one element. Further, they are spread all over Germany as we can see in Figure 3.4.

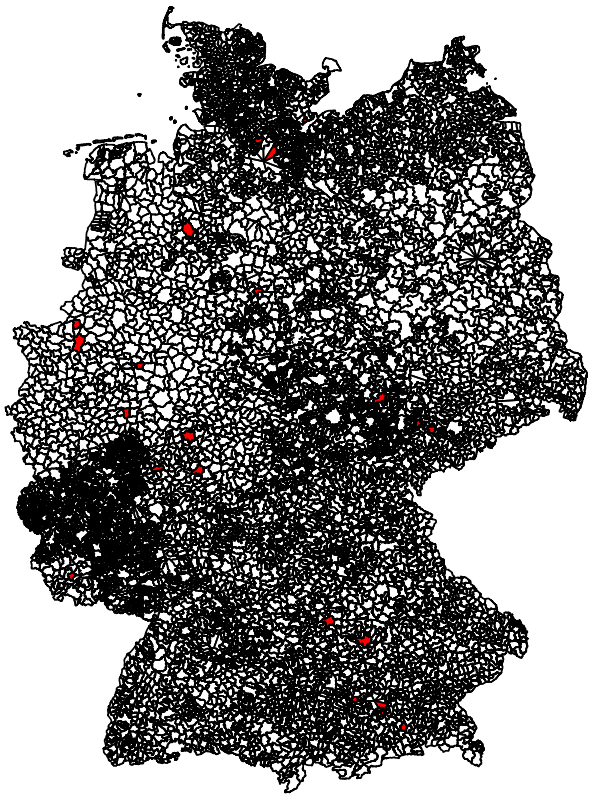


Figure 3.4: SMPs with difference between the partial samples computed by rounding and greedy-type algorithms

A clustering of these SMPs cannot be observed and Figure 3.5, showing Bavaria and Schleswig-Holstein, also indicates that they are more or less randomly distributed. Although there are 25 SMPs in which the rounded solution deviates from the exact integer solution, those deviations cannot be found in every state. In Baden-Wuerttemberg, Berlin, Bremen, Mecklenburg-Western Pomerania, North Rhine-Westphalia as well as Rhineland-Palatinate the rounded solution coincides with the computed integer solution.

(a) Bavaria (b) Schleswig-Holstein

Figure 3.5: SMPs with difference between the partial samples computed by rounding and greedy-type algorithms

It is also worth to mention that the differences do not only occur in some strata (the strata depending on the registered inhabitants at an address) but can be found in every stratum (cf. Table 3.7).

stratum	1	2	3	4	5	6	7	8
number of differences	3	2	2	2	4	4	5	3

Table 3.7: Differences depending on strata

Further, those differences occur almost only at SMPs of type 1, that are communities with at least 10,000 inhabitants as long as they are no urban districts with more than 200,000 inhabitants belonging to communities with more than 400,000 inhabitants. At SMPs of type 3 and type 2 (except for Rhineland-Palatinate) the sampling fraction is fixed to 5%, so no differences can occur.

In conclusion, the rounded continuous solution approximates the optimal solution rather good and because of their speed, the continuous methods are powerful methods. This is especially the case during first tests in which an optimal solution needs to be computed for different settings very often and therefore should be done very fast. After having found the optimal setting, the detailed solution for this setting should be computed by integer methods which are also relatively fast. Especially capacity scaling does not only determine the exact integer solution but it is also very fast and can be applied in practice almost without any limitations. However, all presented integer algorithms strongly rely on the special polymatroidal structure of the feasible region.

Chapter 4

Fundamentals of Nonsmooth Analysis

Just as 'nonlinear' is understood in mathematics to mean 'not necessarily linear' we intend the term 'nonsmooth' to refer to certain situations in which smoothness of the data is not necessarily postulated.

— Frank H. Clarke
Optimization and Nonsmooth Analysis

The first question that arises when reading this statement given in Clarke (1983) is: how often are we faced with nonsmooth data? The answer to this is rather easy: frequently! A well known example of nonsmooth functions is the distance function

$$d_C : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\},\ x \mapsto d_C(x) = \min_{c \in C} \|x - c\|_2,$$

where $C \subset \mathbb{R}^n$. Another example is the reformulation $f(x) = 0$ of the nonlinear variational inequality problem of finding an $x \in C$ such that

$$\langle F(x), y - x \rangle \geq 0 \quad \forall y \in C$$

with given $F : \mathbb{R}^n \to \mathbb{R}^n$. Here f is given by

$$f : \mathbb{R}^n \to \mathbb{R}^n,\ x \mapsto x - \mathrm{Pr}_C(x - F(x))$$

and $\mathrm{Pr}_C(\cdot)$ denotes the projection operator of $\mathbb{R}^n$ onto C.

Therefore, it is necessary to extend existing optimization methods for developing a theory that can be applied to nonsmooth cases.

Clarke mentions the generalized gradient and the generalized Jacobian for locally Lipschitz functions. He uses this theory for solving some nonsmooth optimization problems, for example by applying a Lagrange multiplier rule for nonsmooth functions. This generalized Jacobian was already used by Mifflin (1977) in order to define, with additional properties, semismoothness for functionals. Applying his theory to functions, semismooth functions can be described in an easy way as functions for which the generalized Jacobian defines a certain approximation scheme. Another group of nonsmooth functions are B-differentiable functions, which were first mentioned by Robinson (1987). The requirements for functions in order to be B-differentiable can be satisfied more easily than those for being semismooth. The disad-

vantage of this is that the results for certain methods in the B-differentiable case are not as strong as the ones in the semismooth case.

We will now have a short glance at some basics of functional analysis and the existing different types of (directional) derivatives. The definitions are mostly taken from Shapiro (1990), Werner (2007), Yamamuro (1974), Pang (1990) as well as Rudin (1991).

4.1 Topological Aspects

Before dealing with the different derivatives, we will revise some topological aspects. Although we will deal with normed vector spaces for our applications, we give the definitions in the original form and therefore need a short introduction to topological vector spaces.

Definition 4.1.1 (Vector space). *A* **vector space** *over a field K is a set V in which two operations, addition $+$ and scalar multiplication $\cdot$, are defined such that $(V,+)$ is an Abelian group and the scalar multiplication is compatible.*

Definition 4.1.2 (Topological space). *A* **topological space** *is a pair (X,T) of a set X and a collection T of subsets of X (called* **open** *sets), with the following properties:*

(i) $X, \emptyset$ *are open,*

(ii) U, V *open* $\Rightarrow$ $U \cap V$ *is open,*

(iii) I *index set,* U_i *open for all* $i \in I$ $\Rightarrow$ $\bigcup_{i\in I} U_i$ *is open.*

Such a collection T is called **topology** *on X. If no explicit specification of T is needed, the topological space is called X rather than (X,T).*

Definition 4.1.3 (Continuity). *Let $(X,T_X),(Y,T_Y)$ be topological spaces. Then a mapping $f: X \to Y$ is* **continuous**, *if $f^{-1}(U) \in T_X$ for all $U \in T_Y$.*

Definition 4.1.4 (Topological vector space). *A* **topological vector space** *X is a vector space which is endowed with a topology such that addition and scalar multiplication are continuous functions.*

Definition 4.1.5 (Banach space). *A* **Banach space** *X is a normed vector space which is complete in the metric defined by its norm; this means that every Cauchy sequence is required to converge.*

Definition 4.1.6 (Hilbert space). *A pair $(X, \langle\cdot,\cdot\rangle)$ with X being a vector space and $\langle\cdot,\cdot\rangle$ being a scalar product on X is called inner product space. If the resulting normed space is complete, it is called a* **Hilbert space**.

Remark 4.1.7. *(i) Banach spaces, Hilbert spaces and normed vector spaces are topological vector spaces.*

(ii) The following implications hold: Hilbert space $\Rightarrow$ Banach space $\Rightarrow$ normed vector space $\Rightarrow$ metric vector space $\Rightarrow$ topological vector space

4.2 Different Types of Derivatives

We will now revise some concepts of differentiability. First, we have a look at Shapiro (1990) who defines a class of directional derivatives corresponding to the topology of uniform convergence on a family of subsets. Under certain conditions, these directional derivatives coincide with the well known Fréchet and Gâteaux derivative as well as the Bouligand-derivative given in Robinson (1987), which is essential in the theory of nonsmooth analysis.

Definition 4.2.1 (Lipschitz continuity). *Let $(X, d_X), (Y, d_Y)$ be metric spaces. A function $f : X \to Y$ is called* **Lipschitz (continuous)** *if there exists a real constant $L \geq 0$ such that*

$$d_Y(f(x_1), f(x_2)) \leq L d_X(x_1, x_2) \quad \forall x_1, x_2 \in X.$$

L is called a Lipschitz constant for the function f. If $L \in (0, 1)$, the function is called a **contraction**.

Definition 4.2.2 (Locally Lipschitz continuity). *Let $(X, d_X), (Y, d_Y)$ be metric spaces. A function $f : X \to Y$ is called* **locally Lipschitz (continuous)** *if for every $x \in X$ there exists a neighborhood U of x such that f restricted to U is Lipschitz continuous.*

Definition 4.2.3 (Positive homogeneity). *A mapping $A : X \to Y$ is called* **positively homogeneous** *if*

$$A(th) = tA(h) \quad \forall t \geq 0,\ h \in X.$$

Definition 4.2.4 (σ-directionally differentiability). *Let X, Y be topological vector spaces over the topological field $\mathbb{R}$, $f : X \to Y$ and let Σ be a family of subsets of X. f is called* **σ-directionally differentiable** *at $x \in X$ if there exists a positively homogeneous mapping $A : X \to Y$ satisfying*

$$(i) \quad f(x + h) - f(x) = A(h) + r(h) \text{ and} \tag{4.1}$$

$$(ii) \quad \text{for all } S \in \Sigma : \ \frac{r(th)}{t} \xrightarrow[t \to 0+]{} 0 \text{ uniformly with respect to } h \in S. \tag{4.2}$$

Remark 4.2.5. *(i) If the family Σ consists of all finite subsets of X, the obtained σ-directional derivative coincides with the Gâteaux directional derivative, which will be defined later on.*

(ii) If the family Σ consists of sequentially compact subsets of X, we get the compact directional derivative.

(iii) If the family Σ consists of all bounded subsets of X, we obtain the bounded directional derivative.

(iv) If Σ consists of all bounded subsets of X and X, Y are normed spaces, (4.1) and (4.2) can be replaced by

$$\lim_{h \to 0} \frac{\|f(x + h) - f(x) - A(h)\|}{\|h\|} = 0. \tag{4.3}$$

(v) If in addition the mapping A in (4.3) is linear and continuous we get the Fréchet derivative, which will be defined later on.

(vi) For locally Lipschitz mappings in finite-dimensional spaces, an equivalent definition to (4.3) was given in Robinson (1987) under the name 'Bouligand-derivative'.

This concept of directional derivatives depending on certain sets is also mentioned in Clarke (1983) where the sets are required to be Banach spaces instead of being common topological spaces.

Definition 4.2.6. *Let X, Y be topological vector spaces. $L(X,Y)$ denotes the* **set of all continuous linear mappings** *of X to Y.*

Definition 4.2.7 (Directional derivative)**.** *Let f map X to another Banach space Y. The* **directional derivative** *of f at x in the direction h is defined as*

$$f'(x;h) := \lim_{t \to 0+} \frac{f(x+th) - f(x)}{t},$$

if the limit exists.

Definition 4.2.8 (Gâteaux, Hadamard, Fréchet derivative; Banach spaces)**.** *Let X, Y be Banach spaces and $x \in X$. A function f is said to admit a* **Gâteaux derivative** *at x, if there exists $Df(x) \in L(X,Y)$ such that for every every h in X, one has*

$$\lim_{t \to 0+} \frac{f(x+th) - f(x)}{t} = \langle Df(x), h \rangle,$$

and that the convergence is uniform with respect to h in finite sets.
If the word 'finite' in the preceding sentence is replaced by 'compact', the derivative is known as **Hadamard derivative***; for 'bounded' we obtain the* **Fréchet derivative**.

The concept of differentiability mentioned in Yamamuro (1974) is again quite similar to the concept mentioned in Shapiro (1990). He also defines a derivative depending on certain sets, the so called M-derivative.

Definition 4.2.9 (Directional derivative)**.** *If the* **directional derivative** *of f at x in direction h exists, we write: $f \in D(x, Y; \to h)$. Further, $D(x, Y; \to X) := \bigcap_{h \in X} D(x, Y; \to h)$.*

Definition 4.2.10 (M-differentiability)**.** *Let X, Y be topological vector spaces, $D \subset X$ open and M be a set of subsets of X such that every singleton belongs to M.*
$f : D \to Y$ is called **M-differentiable** *at x if there exists*

$$A \in L(X,Y) : \lim_{t \to 0+} \frac{f(x+th) - f(x) - A(th)}{t} = 0$$

uniformly with respect to h on each member of M. We write $f \in D_M(x,Y)$ and the continuous and linear mapping A is called **M-derivative**.

Remark 4.2.11 (Gâteaux, Hadamard, Fréchet derivative)**.** *Assume that* $f \in D_M(x, Y)$.

(i) If M *consists of all bounded subsets of* X, f *is* **Fréchet differentiable** *at* x. *We write:* $f \in D(x, Y)$.

(ii) If M *consists of all sequentially compact subsets of* X, f *is* **Hadamard differentiable** *at* x. *We write:* $f \in D_H(x, Y)$.

(iii) If M *consists of all single point subsets of* X, f *is* **Gâteaux differentiable** *at* x. *We write:* $f \in D_G(x, Y)$.

The following lemma shows the connection between Fréchet, Hadamard, Gâteaux and directional differentiable functions.

Lemma 4.2.12.

$$D(x, Y) \subset D_H(x, Y) \subset D_G(x, Y) \subset D(x, Y; \to X)$$

Proof. Refer to Yamamuro (1974). □

In the case of normed vector spaces, we get the following lemma for Fréchet differentiable functions.

Lemma 4.2.13. *Let* X *be a normed linear space and* Y *be a topological vector space. Then* $f \in D(x, Y)$ *if and only if*

$$\lim_{h \to 0} \frac{f(x+h) - f(x) - A(h)}{\|h\|} = 0.$$

Proof. Refer to Yamamuro (1974). □

We will now have a detailed look at the case in which X, Y are normed spaces, which is the case for our application. In contrast to the former definitions, these definitions do not make use of different sets for defining the derivatives. For further details we refer to Werner (2007).

Definition 4.2.14 (Gâteaux, Fréchet derivative; normed spaces)**.** *Let X, Y be normed spaces, $D \subset X$ be an open subset and $f : D \to Y$.*

(i) f is called **Gâteaux differentiable** *in $x \in D$ if there exists a continuous linear operator $A \in L(X, Y)$ such that*

$$\lim_{t \to 0} \frac{f(x + th) - f(x)}{t} = A(h) \quad \forall h \in X. \tag{4.4}$$

(ii) f is called **Fréchet differentiable** *in $x \in D$ if the convergence in (4.4) is uniform concerning $h \in B_X = \{y : \|y\| \leq 1\}$.*

Lemma 4.2.15. *Let X, Y be normed spaces, $D \subset X$ be an open subset and $f : D \to Y$. f is Fréchet differentiable in $x \in D$ if and only if there exits $A \in L(X, Y)$ such that*

$$(i) \quad f(x + h) - f(x) = A(h) + r(h) \text{ and} \tag{4.5}$$

$$(ii) \quad \frac{r(h)}{\|h\|} \xrightarrow[\|h\| \to 0]{} 0. \tag{4.6}$$

Proof. Refer to Werner (2007). □

4.3 B-differentiability

Now that we have seen this many different definitions of the derivatives one may ask how B(ouligand)-differentiability fits into these definitions. As noted in Remark 4.2.5, Shapiro (1990) mentions the Bouligand-differentiability for the case the family Σ consists of all bounded subsets of X. At first glance this seems to characterize Fréchet differentiability but in his definition he does not require A to be continuous and linear. He only requires A to be a positively homogeneous mapping. Pang (1990) gives a clear definition of B(ouligand)-differentiability for the case $f : \mathbb{R}^n \to \mathbb{R}^m$ and points out the difference between the derivatives mentioned in the last section.

Definition 4.3.1 (B-differentiability)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is said to be* **B-differentiable** *at $x \in \mathbb{R}^n$ if there exists a positively homogeneous function $A : \mathbb{R}^n \to \mathbb{R}^m$, called the* **B-derivative** *of f at x, such that*

$$\lim_{h \to 0} \frac{f(x + h) - f(x) - A(h)}{\|h\|} = 0.$$

If f is B-differentiable at all $x \in S$, then f is said to be B-differentiable on S.

Remark 4.3.2. *In the original definition of B-differentiability in Robinson (1987) the positive homogeneity of the B-derivative was expressed in terms of a cone property of its graph.*

Remark 4.3.3. *The one fundamental distinction between a B-differentiable function and a Fréchet differentiable function is the absence of linearity in the B-derivative.*

Shapiro (1990) shows that if X, Y in Definition 4.2.4 are normed finite dimensional spaces and $f : X \to Y$ is locally Lipschitz, all types of directional derivatives mentioned in Remark 4.2.5 coincide. Therefore we get the following easy to understand definition for B-differentiability in the case that $X = \mathbb{R}^n, Y = \mathbb{R}^m$.

Definition 4.3.4 (B-differentiability; locally Lipschitz)**.** *$f : \mathbb{R}^n \to \mathbb{R}^m$ is called* **B-differentiable** *in $x \in D$ if f is locally Lipschitz and all directional derivatives exist in $x \in D$.*

The local Lipschitz property is important because it is well known that there are functions that are directionally differentiable at a point without being continuous there. Another important consequence of the B-differentiability is that the limit of the directional derivative is uniform on compact sets of directions. This is stated in the following lemma.

Lemma 4.3.5. *Let $D \subset \mathbb{R}^n$, $f : D \to \mathbb{R}^m$ be B-differentiable in $x \in D$. Then*

$$\lim_{h \to 0} \frac{f(x+h) - f(x) - f'(x;h)}{\|h\|} = 0.$$

Proof. Refer to Kanzow (2005), Ito and Kunisch (2009), Facchinei and Pang (2003a) or Qi (1993). □

This property will be used in Chapter 6 and can be rewritten as follows.

Lemma 4.3.6. *Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a B-differentiable function and $t \geq 0$. Then it holds:*

(i) $f(x+h) - f(x) - f'(x;h) = o(\|h\|)$,

(ii) $f(x+th) - f(x) \leq t f'(x;h) + \phi(t\|h\|)t\|h\|$,

where ϕ is a function

$$\phi : \mathbb{R} \to \mathbb{R}_+ \cup \{0\}, \; u \mapsto \phi(u)$$

satisfying $\phi(u) \xrightarrow[u \to 0]{} 0$.

Proof. (i) Refer to Qi (1993).

(ii) Follows from (i) and $f'(x;\cdot)$ being positively homogeneous.

□

4.4 Generalized Jacobian

As a nonsmooth function $f : \mathbb{R}^n \to \mathbb{R}^m$ is not differentiable in the usual Fréchet sense for all $x \in D$, we need to define an analogon for the usual Jacobian. Furthermore, if we want to develop algorithms for solving nonsmooth equations, we will encounter many difficulties when sticking solely to directional derivatives. Therefore, we make use of the generalized Jacobian which is discussed in detail in Clarke (1983) and Kanzow (2005). This generalized Jacobian is needed for the generalized Newton's method, which we will mention in Chapter

6, and the semismooth Newton method from Qi and Sun (1993), which will be mentioned in Chapter 5. Furthermore, we give some important theorems and lemmata.

Definition 4.4.1. *D_f denotes the set of all elements in which f is differentiable.*

$$D_f := \{x \in \mathbb{R}^n : f \text{ is differentiable in } x\}$$

Definition 4.4.2 (B-subdifferential, generalized Jacobian)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be locally Lipschitz, $x \in \mathbb{R}^n$ and let $J_f(x) \in \mathbb{R}^{m\times n}$ denote the Jacobian of f in $x \in D_f$. Then*

$$\partial_B f(x) := \{V \in \mathbb{R}^{m\times n} : \exists (x^k)_{k\in\mathbb{N}} \subset D_f : x^k \to x \text{ and } J_f(x^k) \to V\}$$

is called **B-subdifferential** *of f in x and*

$$\partial f(x) := \operatorname{conv} \partial_B f(x)$$

is called the **generalized Jacobian** *of f in x.*

Remark 4.4.3. *The existence of the sequence $(x^k)_{k\in\mathbb{N}}$ is guaranteed due to Rademacher (1919), who shows that a locally Lipschitz function f is differentiable almost everywhere so $\mathbb{R}^n \setminus D_f$ has Lebesgue measure zero.*

For understanding and getting used to the generalized Jacobian we will have a look at the following examples.

Example 4.4.4. *(i) If $f : \mathbb{R}^n \to \mathbb{R}^m$ is a continuously differentiable function then*

$$\partial_B f(x) = \partial f(x) = \{J_f(x)\} \quad \forall x \in \mathbb{R}^n.$$

(ii) If $f : \mathbb{R} \to \mathbb{R}$, $x \mapsto |x|$, we get

$$\partial_B f(x) = \partial f(x) = \{J_f(x)\} \quad \forall x \neq 0$$
$$\partial_B f(0) = \{-1, +1\}, \quad \partial f(0) = [-1, +1].$$

(iii) If $f : \mathbb{R} \to \mathbb{R}$, $x \mapsto \max\{0, x\}$, we deduce

$$\partial_B f(x) = \partial f(x) = \{J_f(x)\} \quad \forall x \neq 0$$
$$\partial_B f(0) = \{0, +1\}, \quad \partial f(0) = [0, +1].$$

(iv) If $f : \mathbb{R}^n \to \mathbb{R}$, $x \mapsto \|x\|_2$, we obtain

$$\partial_B f(x) = \partial f(x) = \left\{\frac{x}{\|x\|_2}\right\} \quad \forall x \neq 0$$
$$\partial_B f(0) = \{x \in \mathbb{R}^n \ : \ \|x\|_2 = 1\}, \quad \partial f(0) = \{x \in \mathbb{R}^n \ : \ \|x\|_2 \leq 1\}.$$

Now we will consider some useful properties of B-differentiable and locally Lipschitz functions, which will be helpful for the next chapters.

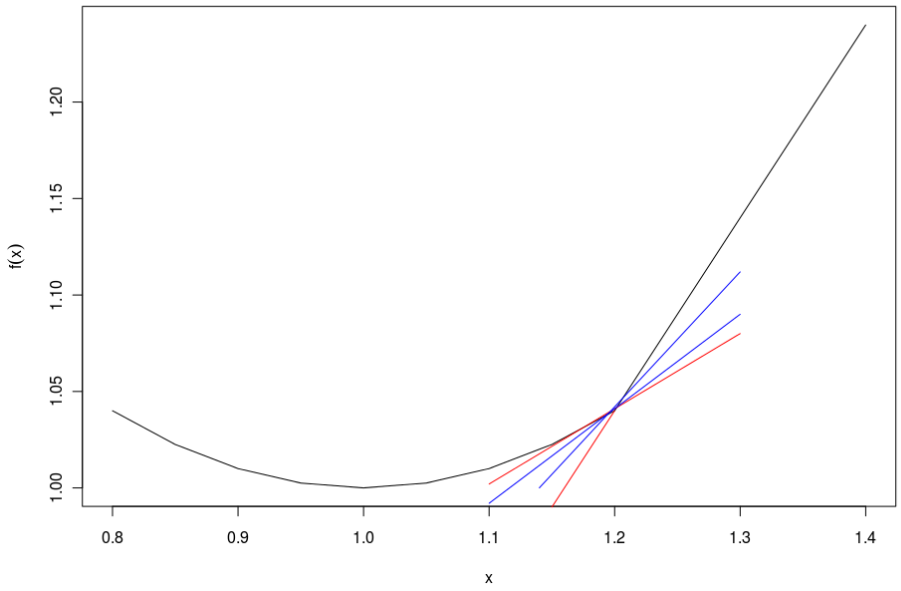


Figure 4.1: Example of the B-subdifferential (red) and elements of the generalized Jacobian (blue)

Theorem 4.4.5 (Mean value theorem)**.** *Let $D \subset \mathbb{R}^n$ be open, $f : D \to \mathbb{R}^m$ be locally Lipschitz and $x, y \in D$. Then there exist $m+1$ points z^k on $[x, y]$ as well as $m+1$ scalars $\lambda_k \geq 0$ satisfying $\sum_{k=1}^{m+1} \lambda_k = 1$ and $V_k \in \partial f(z^k)$ such that*

$$f(y) - f(x) = \sum_{k=1}^{m+1} \lambda_k V_k (y - x).$$

Proof. Refer to Clarke (1983). □

The former theorem reflects a mean value theorem which is frequently not known even for smooth functions. Furthermore, we get the following corollary of Theorem 4.4.5 which shows the connection of the generalized Jacobian with B-differentiable functions.

Corollary 4.4.6. *Let $D \subset \mathbb{R}^n$ be open and $f : D \to \mathbb{R}^m$ be B-differentiable in $x \in D$. Then for every $h \in \mathbb{R}^n$ there exists $V \in \partial f(x)$ such that*

$$f'(x; h) = Vh.$$

Proof. Refer to Qi and Sun (1993). □

4.5 Semismoothness

The before mentioned generalized Jacobian helps to extend many results from smooth analysis to locally Lipschitz functions. However, a straightforward extension of Newton's method to general nonsmooth equations by using the generalized Jacobian will not work so easily, so we need another important notion of nonsmooth analysis, namely semismoothness. Semismoothness was originally introduced for functionals by Mifflin (1977) and extended by Qi and Sun (1993), who make use of 'p-order semismoothness'. We will concentrate on the basic semismoothness and strong semismoothness, which in some cases also combine B-differentiability with the generalized Jacobian. As we can see in the next definition, semismooth functions are locally Lipschitz functions for which the generalized Jacobians define a certain approximation scheme. This makes it possible to get almost the same results for Newton's method in the nonsmooth case as in the smooth case. For a detailed discussion on generalized Newton's methods we refer to Chapter 5 and 6.

Definition 4.5.1 (Semismoothness)**.** *Let $D \subset \mathbb{R}^n$ and $f : D \to \mathbb{R}^m$ be a B-differentiable function. Then f is called*

(i) **semismooth** *in $x \in D$, if* $\lim\limits_{h^k \to 0, V_k \in \partial f(x+h^k)} \dfrac{V_k h^k - f'(x;h^k)}{\| h^k \|} = 0,$

(ii) **strongly semismooth** *in $x \in D$, if* $\limsup\limits_{h^k \to 0, V_k \in \partial f(x+h^k)} \dfrac{V_k h^k - f'(x;h^k)}{\| h^k \|^2} < +\infty,$

(iii) (strongly) semismooth on D, if f is (strongly) semismooth in every $x \in D$.

This definition is rather complicated but as we will see now, many common functions are semismooth functions.

Lemma 4.5.2. *Let $D \subset \mathbb{R}^n$ be open, $x \in D$ and $f : D \to \mathbb{R}^m$ be Lipschitz. Then it holds:*

(i) If f is continuously differentiable in x, then f is semismooth in x.

(ii) If f is differentiable and f' locally Lipschitz in x, then f is strongly semismooth in x.

Proof. Refer to Kanzow (2005) or Qi and Sun (1993). □

Theorem 4.5.3. *Let $D \subset \mathbb{R}^n$ be open, convex, and $f : D \to \mathbb{R}$ be convex. Then f is semismooth on D.*

Proof. Refer to Kanzow (2005) or Qi and Sun (1993). □

Example 4.5.4. *(i) The minimum-function*

$$f : \mathbb{R}^2 \to \mathbb{R},\ (x, y) \mapsto \min\{x, y\}$$

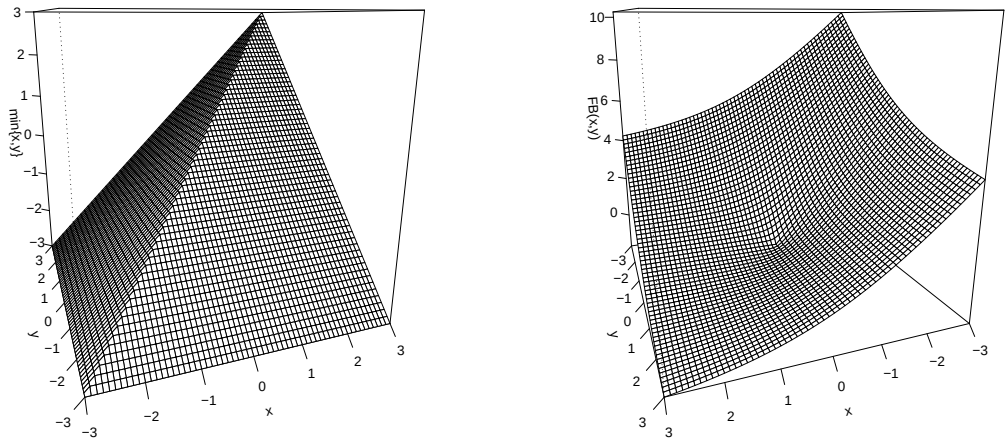
is strongly semismooth on $\mathbb{R}^2$.
As f is differentiable and f' is locally Lipschitz for all $x \neq y$, f is strongly semismooth

for all $x \neq y$ (Lemma 4.5.2). The strong semismoothness in $x = y$, that are the points in the kink of the surface of f, can be proved by a simple calculation.

(ii) The Fischer-Burmeister function (Fischer, 1992)

$$FB : \mathbb{R}^2 \to \mathbb{R},\ (x, y) \mapsto \sqrt{x^2 + y^2} - x - y$$

is strongly semismooth on $\mathbb{R}^2$. This function plays an important role in solving variational inequality problems because it serves as NCP-function and the corresponding merit function is smooth, which simplifies gaining convergence statements. A short overview of this topic will be given in Chapter 6.
As FB is convex on $\mathbb{R}^2$, FB is semismooth on $\mathbb{R}^2$ (Theorem 4.5.3). For all $(x, y) \neq (0, 0)$, FB is strongly semismooth because of Lemma 4.5.2. The strong semismoothness in $(x, y) = (0, 0)$ can be proved by simple calculation.



(a) Minimum-function

(b) Fischer-Burmeister function

Figure 4.2: Perspective plot of the minimum-function (a) and the Fischer-Burmeister function (b)

As the sum of Fréchet differentiable functions is again Fréchet differentiable, we are interested whether such rules also exist for semismooth functions.

Lemma 4.5.5. *Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a locally Lipschitz function. If each component of f is semismooth at x, then f is semismooth at x.*

Proof. Refer to Qi and Sun (1993). □

Lemma 4.5.6. *(i) Scalar products of semismooth functions are semismooth functions.*
(ii) The sum of semismooth functions is semismooth.

Proof. Refer to Mifflin (1977). □

Lemma 4.5.7 (Chain rule)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be semismooth at $x \in \mathbb{R}^n$ and $g : \mathbb{R}^m \to \mathbb{R}^n$ be semismooth at $f(x) \in \mathbb{R}^m$. Then, the composite function $g \circ f$ is semismooth at x.*

Proof. Refer to Fischer (1997). □

Semismoothness positively affects the properties of the directional derivatives as we will see in the following estimates.

Lemma 4.5.8. *Let $D \subset \mathbb{R}^n$ be open, $x \in D$ and $f : D \to \mathbb{R}^m$ be B-differentiable in x. Then it holds:*

(i) If f is semismooth in x, then

$$\|f'(x+h;h) - f'(x;h)\| = o(\|h\|) \quad \forall h \to 0.$$

(ii) If f is strongly semismooth in x, then

$$\|f'(x+h;h) - f'(x;h)\| = \mathcal{O}(\|h\|^2) \quad \forall h \to 0.$$

Proof. Refer to Kanzow (2005). □

The next lemma describes an estimate for a kind of a Taylor expansion.

Lemma 4.5.9. *Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be B-differentiable and $x \in \mathbb{R}^n$. Then it holds:*

(i) If f is semismooth in x, then

$$\|f(x+h) - f(x) - Vh\| = o(\|h\|) \quad \forall h \to 0,\ V \in \partial f(x+h).$$

(ii) If f is strongly semismooth in x, then

$$\|f(x+h) - f(x) - Vh\| = \mathcal{O}(\|h\|^2) \quad \forall h \to 0,\ V \in \partial f(x+h).$$

Proof. Refer to Kanzow (2005). □

Chapter 5

Calibration via Semismooth Newton Method

Calibration has established itself as an important methodological instrument in large-scale production of statistics.

— Carl-Erik Särndal
The Calibration Approach in Survey Theory and Practice

Calibration is a widely used term which in general may have different meanings. Having a look at `http://www.thefreedictionary.com/calibrating` we get a military based definition like 'to measure the caliber of (a gun, mortar, etc.)' or 'to determine or check the range and accuracy of (a piece of artillery)' as well as definitions based in physics like 'to mark (the scale of a measuring instrument) so that readings can be made in appropriate units' or 'to determine the accuracy of (a measuring instrument, etc.)'. Furthermore, the more general definition 'to check, adjust, or standardize a measuring instrument, usually by comparing it with an accepted model' is given. This fits to the usage of calibration in mathematics, like 'adjoint-based calibration of local volatility models' where the parameter 'local volatility' of the measuring instrument 'local volatility model' is adjusted by comparing the computed call prices to given market data. Apart from these applications, statisticians make use of calibration in the context of calibrating design weights for certain estimators and as mentioned in Särndal (2007), 'calibration has established itself as an important methodological instrument in large-scale production of statistics.'

5.1 Calibration in Statistics

As seen before, calibration has different meanings so we will recall the definition and arguments given in Särndal (2007) to state our point of view.

Definition 5.1.1 (Calibration approach)**.** *The* **calibration approach** *to estimation for finite populations consists of*

(i) a computation of weights that incorporate specified auxiliary information and are restrained by calibration equation(s),

(ii) the use of these weights to compute linearly weighted estimates of totals and other finite population parameters: weight times variable value, summed over a set of observed units,

(iii) an objective to obtain nearly design unbiased estimates as long as nonresponse and other non sampling errors are absent.

This definition shows the advantages of calibration. Leading national statistical agencies fix on weighting methods because they are easy to explain to users and stakeholders and are, not only because of Horvitz and Thompson (1952), widely accepted. Furthermore, using auxiliary information allows to improve the accuracy of survey estimates and can deal effectively with surveys where auxiliary information exists at different levels, which is the case in the German Census Sampling and Estimation Research Project. In addition to that, if the gained weights are applied to a variable used for calibration, they deliver the known estimates or true values. In case of a sample based census that uses different kinds of estimators, this is very important because consistency with known aggregates is a desire to promote credibility. As most addressee of census results want to get one tool which can be applied to different variables, calibration delivers a unique weighting system, applicable to many study variables.

One standard approach to include these weights is calibration estimation according to Deville and Särndal (1992). However, it has to be kept in mind that the calibration approach can also deal with complex sampling designs, adjustments for nonresponse and frame errors. Since we are only interested in the computation of the weights we assume that we have single phase sampling and full response.

A simple way of weighting the study variable values y_k by the inverse of their inclusion probabilities π_k was introduced by Horvitz and Thompson (1952)

$$\hat{t}_y^{HT} = \sum_{k=1}^{n} d_k y_k,$$

where $d_k = \pi_k^{-1}$ $(k = 1, \ldots, n)$ denote the (original) design weights which are the reciprocal of the inclusion probabilities from a survey of size n.

It is often the case that some totals are known and shall be reproduced by Horvitz-Thompson estimation. These so-called calibration benchmarks

$$t_x = \sum_{k=1}^{n} w_k x_k$$

require the introduction of new calibrated weights $w_k = g_k d_k$, which should be close to the design weights d_k.

Since the calibrated weights $g_k d_k$ should not differ too much from the design weights, one could - in addition - minimize the distance between g_k and 1 for all $k = 1, \ldots n$, which leads to the following calibration problem.

$$\min_{g \in \mathbb{R}^n} \frac{1}{2} \sum_{k=1}^{n} d_k (g_k - 1)^2 \tag{5.1}$$

$$s.t. \sum_{k=1}^{n} w_k x_k = \sum_{k=1}^{n} g_k d_k x_k = t_x \tag{5.2}$$

where x_k and t_x are vector-valued quantities.

This method is called the minimum distance method (cf. Särndal, 2007) and was developed further by Deville and Särndal (1992) who dealt with different distance functions like

$$\sum_{k=1}^{n} d_k(g_k \cdot \ln(g_k) - g_k + 1)$$

and showed that they generate asymptotically equivalent calibration estimators. Other distance functions were considered by Deville et al. (1993), Singh and Mohl (1996) as well as Stukel et al. (1996). An efficient variance estimation method is given by Demnati and Rao (2004).

Apart from this method there also exists the instrumental vector method considered in Estevao and Särndal (2006) and Kott (2006), which we will not consider in this work. A recent overview of the developments in calibration methodology can be drawn from Kim and Park (2010).

Furthermore, it might be necessary in some instances, to limit the size of the weights in order to avoid a huge spread of the values. This can be seen as a hard way of circumventing a large variation of the calibrated survey weights as demanded in Gelman (2007b). Hence, in addition to the linear system of equations, we add a box constraint for the variables g. Therefore, we denote

$$U = \{g \in \mathbb{R}^n : \ m_k \leq g_k \leq M_k, \quad k = 1, ..., n\},$$

with $0 \leq m_k \leq 1 \leq M_k$. This leads to our formulation of the calibration problem as a quadratic program with linear equality and inequality constraints.

There are several approaches in the literature that deal with the solution of the constrained optimization using the special structure of the constraints. One approach followed by Deville et al. (1993) uses a penalty function formulation for the box constraint. Here we extend their approach to our setting with componentwise different bounds. Then the objective function is given by

$$\sum_{k=1}^{n} d_k f_3^k(g_k)$$

where

$$f_3^k(g_k) = \begin{cases} \left((g_k - m_k)\ln(\frac{g_k - m_k}{1 - m_k}) + (M_k - g_k)\ln(\frac{M_k - g_k}{M_k - 1})\right)\alpha, & \text{if } m_k < g_k < M_k, \\ \infty & \text{if } g_k \leq m_k, \ g_k \geq M_k. \end{cases}$$

with $\alpha = \frac{(M_k - 1)(1 - m_k)}{M_k - m_k}$. The resulting calibration problem is a minimization problem with only equality constraints. This can be tackled by solving with the Lagrange multiplier rule the necessary optimality conditions, a system of nonlinear equations in g and the multiplier λ. Although this method is somewhat related to interior point methods, it does not change

the penalty parameter as the iteration progresses. Therefore, the solution of such a problem cannot have components with values on the boundary of the box constraint U.

One of the most recent methods proposed in statistics for solving the constrained calibration problem with a quadratic objective function in the statistics software R from the R Development Core Team (2012) is the solver 'calib' created by the group of Yves Tillé (cf. Tillé and Matei, 2009). It can be interpreted as a pegging algorithm, that means in every iteration it computes the solution of the optimization problem with equality constraints for the index set $I \subset \{1, \ldots, n\}$ of all inactive indices and afterwards projects this solution on the set determined by the box constraint. Then the set of inactive indices is updated and the calibration benchmarks t are reduced by the value of all g_k which became active in this iteration. Then the iteration starts again and the optimization problem with equality constraints is solved for the new index set I of all inactive indices. In optimization, see Gill et al. (1981), it is well known, that active set strategies like the one explained above are in general not sufficient to ensure convergence of the iterates. However, in practice this method works quite well.

Vanderhoeft (2001) proposes a Newton-type method where he uses the technique of the Lagrange multiplier for the simplified problem without box constraint and sets up the necessary optimality conditions. This equation can be solved explicitly where the solution depends on the Lagrange multiplier. Then the Lagrange multiplier has to be chosen such that the equality constraint is satisfied. This technique is modified and the projection is included in this procedure. The resulting nonlinear equation is solved by a Newton-type method, where the derivative is replaced by some approximation. The disadvantage of this approach is that it does not guarantee convergence and that it may break down.

Our goal is to transform the calibration problem for general functions f into a nonlinear equation ψ depending on the Lagrange multiplier similar to the approach given in Chapter 3 dealing with the optimal allocation problem. Since - due to the projection - this mapping ψ is no longer differentiable in the classical sense, Newton's method cannot be applied. However, the nonlinearity is such that it is possible to apply the 'semismooth Newton method' given in Qi and Sun (1993) in order to solve this equation numerically. This algorithm can be applied to the calibration problem with box constraint and yields a fast and efficient numerical method. Furthermore, the analysis of this method is well understood and local quadratic convergence will be shown.

5.2 Mathematical Formulation of the Calibration problem

Let $0 < p < n < \infty$ and let $x_k = (x_{k1}, ..., x_{kp})^T \in \mathbb{R}^p$ for $k = 1, \ldots, n$ be the calibration variables. Measurements of these variables are available for all sample elements. Furthermore, let x_{ki} be the value of the i-th calibration variable for the k-th sample element. The variable d_k denotes the reciprocal of the inclusion probability for all x_k, $(k = 1, ..., n)$, so we can define $d = (d_1, ..., d_n)^T \in \mathbb{R}^n$ as the vector of the design weights. The calibration factors, which will be determined by the algorithm, are denoted by g_k $(k = 1, ..., n)$ and form the vector of g-weights $g = (g_1, ..., g_n)^T \in \mathbb{R}^n$. There also exist calibration benchmarks t_{x_i} $(i = 1, ..., p)$ forming the vector of the calibration totals $t_x = (t_{x_1}, ..., t_{x_p})^T \in \mathbb{R}^p$.

We further define the following matrices:

$$\bar{X}^T := \begin{pmatrix} | & & | \\ \xi_1 & \cdots & \xi_n \\ | & & | \end{pmatrix} = \begin{pmatrix} \xi_{11} & \cdots & \xi_{n1} \\ \vdots & & \vdots \\ \xi_{1p} & \cdots & \xi_{np} \end{pmatrix} = \begin{pmatrix} x_{11}d_1 & \cdots & x_{n1}d_n \\ \vdots & & \vdots \\ x_{1p}d_1 & \cdots & x_{np}d_n \end{pmatrix} \in \mathbb{R}^{p\times n},$$
$$D := \operatorname{diag}(d_1, ..., d_n),$$

where $\bar{X}^T$ is called the design matrix and consists of the values of the i-th calibration variable for the k-th sample element multiplied with the design weights.

As already mentioned, we may want to avoid negative or widely spread calibration factors so we define the box constraint

$$U = \{g \in \mathbb{R}^n : \ m_k \leq g_k \leq M_k, \quad k = 1, ..., n\},$$

qualifying a convex, closed set. Here we assume $0 \leq m_k \leq 1 \leq M_k$.

Let f be a strictly convex, nonnegative, twice continuously differentiable function

$$f : \mathbb{R}_+ \to \mathbb{R}_+ \cup \{0\},$$

which satisfies $f(1) = 0,\ f'(1) = 0$ and $f''(1) = 1$. Then we define

$$F : \mathbb{R}_+^n \to (\mathbb{R}_+ \cup \{0\})^n,\ g \mapsto F(g) = (f(g_1), ..., f(g_n))^T,$$

with the Jacobian

$$F' : \mathbb{R}_+^n \to \mathbb{R}^{n\times n},\ g \mapsto F'(g) = \operatorname{diag}(f'(g_1), ..., f'(g_n)),$$

and its inverse

$$F'^{-1} : \mathbb{R}^n \to \mathbb{R}^{n\times n},\ u \mapsto F'^{-1}(u) = \operatorname{diag}(f'^{-1}(u_1), ..., f'^{-1}(u_n)).$$

In the literature, special cases are being considered. The truncated linear method means calibration with a box constraint concerning the function

$$f_1 : \mathbb{R} \to \mathbb{R}_+ \cup \{0\},\ g_k \mapsto f_1(g_k) = \frac{(g_k - 1)^2}{2},$$

and the multiplicative method with additional box constraint where

$$f_2 : \mathbb{R}_+ \to \mathbb{R}_+ \cup \{0\},\ g_k \mapsto f_2(g_k) = g_k \cdot \ln(g_k) - g_k + 1.$$

In this setting we consider the following general calibration problem:

$$\begin{aligned} &\min_{g\in\mathbb{R}^n} \ d^T F(g) \\ &\quad s.t.\ \bar{X}^T g - t_x = 0 \\ &\qquad g \in U. \end{aligned} \tag{5.3}$$

Furthermore, we assume that the feasible set is non-empty. Since it is a convex compact set and $d^T F(\cdot)$ is continuous, there exists a solution of the calibration problem (5.3) which is due to the strict convexity of $d^T F(\cdot)$ unique.

5.3 Solution of the Calibration Problem

The main goal in the following approach (cf. Münnich, Sachs and Wagner, 2012b) is to express g as a function depending on the Lagrange multiplier λ. Then, this expression $g(\lambda)$ is inserted into the function

$$h : \mathbb{R}^n \to \mathbb{R}^p,\ g \mapsto h(g) = \bar{X}^T g - t,$$

which leads to a p-dimensional nonsmooth equation $h(g(\lambda)) = 0$. This approach is a generalization of the approach presented in Münnich, Sachs and Wagner (2012c) which dealt with a different objective function but only one equality constraint.

For a standard Lagrangian approach with complementarity conditions we rewrite the calibration problem (5.3) with equality and inequality constraints as:

$$\begin{aligned} &\min d^T F(g) \\ &s.t.\ h(g) := \bar{X}^T g - t_x = 0 \\ &\quad u(g) := g - M \leq 0 \\ &\quad v(g) := m - g \leq 0. \end{aligned} \tag{5.4}$$

The corresponding optimality criteria read as follows.

Theorem 5.3.1. *A vector $g^* \in \mathbb{R}^n$ is a solution of problem (5.4) if and only if there exists a Lagrange multiplier $\lambda^* \in \mathbb{R}^p,\ \mu^* \in \mathbb{R}^n_+,\ \kappa^* \in \mathbb{R}^n_+$ such that*

$$\nabla(d^T F(g^*)) + \sum_{i=1}^{p} \lambda_i^* \nabla h_i(g^*) + \sum_{j=1}^{n} \mu_j^* \nabla u_j(g^*) + \sum_{k=1}^{n} \kappa_k^* \nabla v_k(g^*) = 0, \tag{5.5}$$

$$h(g^*) = 0, \tag{5.6}$$

$$\mu_j^* u_j(g^*) = 0 \quad (j = 1, \ldots, n), \tag{5.7}$$

$$\kappa_k^* v_k(g^*) = 0 \quad (k = 1, \ldots, n). \tag{5.8}$$

Proof. See Theorem 3.8 in Horst (1979). A constraint qualification condition is satisfied because of all constraint functions being affine. Furthermore, the objective function $d^T F(g)$ is separable and strictly convex and the feasible set is convex, so the necessary optimality condition is also sufficient. □

This system of equalities can be reformulated in a more compact form.

Lemma 5.3.2. *Under the given assumptions of Theorem 5.3.1 equation* (5.5) *is equivalent to*

$$\left.\begin{array}{ll} 0 \geq d_k f'(M_k) + \xi_k^T \lambda^* & \text{if } g_k^* = M_k, \\ 0 = d_k f'(g_k^*) + \xi_k^T \lambda^* & \text{if } g_k^* \in (m_k, M_k), \\ 0 \leq d_k f'(m_k) + \xi_k^T \lambda^* & \text{if } g_k^* = m_k, \end{array}\right\} \quad (k = 1, \ldots, n). \tag{5.9}$$

Proof. A closer look at equations (5.7) and (5.8) reveals the following equivalences for the optimal solution g^*:

$$\begin{aligned} & g_k^* = M_k \quad \Leftrightarrow \quad u_k(g^*) = 0,\ \mu_k^* \geq 0,\ v_k(g^*) < 0,\ \kappa_k^* = 0, \\ & g_k^* = m_k \quad \Leftrightarrow \quad u_k(g^*) < 0,\ \mu_k^* = 0,\ v_k(g^*) = 0,\ \kappa_k^* \geq 0 \quad (k = 1, \ldots, n), \\ & g_k^* \in (m_k, M_k) \quad \Leftrightarrow \quad u_k(g^*) < 0,\ \mu_k^* = 0,\ v_k(g^*) < 0,\ \kappa_k^* = 0. \end{aligned}$$

As $\nabla(d^T F(g^*))_k = d_k f'(g_k^*),\ \nabla h_k(g^*) = (\xi_{1k}, \ldots, \xi_{nk})^T,\ \nabla u_k(g^*) = e^k$ and $\nabla v_k(g^*) = -e^k$ for all $k = 1, \ldots, n$ equation (5.5) is equivalent to

$$d_k f'(g_k^*) + \xi_k^T \lambda^* + \mu_k - \kappa_k = 0 \quad (k = 1, \ldots, n).$$

This is again equivalent to the following three cases for $k = 1, ..., n$:

$$\begin{array}{ll} 0 \geq d_k f'(M_k) + \xi_k^T \lambda^* & \text{if } g_k^* = M_k, \\ 0 = d_k f'(g_k^*) + \xi_k^T \lambda^* & \text{if } g_k^* \in (m_k, M_k), \\ 0 \leq d_k f'(m_k) + \xi_k^T \lambda^* & \text{if } g_k^* = m_k, \end{array}$$

which completes the proof. □

Another approach which leads to the same three equations uses the normal cone and is presented below. It is based on the approach given in Chapter 3 and extends this idea. Based on this, the necessary optimality conditions for solutions of the calibration problem (5.3) are as follows.

Theorem 5.3.3. *A vector $g^* \in \mathbb{R}^n$ is a minimum of problem* (5.3) *if and only if there exists a Lagrange multiplier $\lambda^* \in \mathbb{R}^p$ such that*

$$0 \in \nabla(d^T F(g^*)) + \sum_{i=1}^{p} \lambda_i^* \nabla h_i(g^*) + N_U(g^*), \tag{5.10}$$

and

$$h(g^*) = 0. \tag{5.11}$$

Proof. See Theorem 3.25 in Ruszczynski (2006). Since all constraint functions are affine and U is a convex polyhedron, a constraint qualification condition is satisfied. Furthermore, the objective function f is strictly convex and the feasible set U is convex, so the necessary optimality condition is also sufficient. □

If we apply this result to our calibration problem we get the following lemma.

Lemma 5.3.4. *Under the given assumptions of Theorem 5.3.3 equation* (5.10) *is equivalent to*

$$\left.\begin{array}{ll} 0 \geq d_k f'(M_k) + \xi_k^T \lambda^* & \text{if } g_k^* = M_k, \\ 0 = d_k f'(g_k^*) + \xi_k^T \lambda^* & \text{if } g_k^* \in (m_k, M_k), \\ 0 \leq d_k f'(m_k) + \xi_k^T \lambda^* & \text{if } g_k^* = m_k, \end{array}\right\} \quad (k = 1, \ldots, n). \tag{5.12}$$

Proof. In this particular setting with the definition of U it is easy to show that

$$N_U(g^*) := \{y \in \mathbb{R}^n : \begin{cases} y_k \geq 0, & \text{if } g_k^* = M_k, \\ y_k = 0, & \text{if } g_k^* \in (m_k, M_k), \}. \\ y_k \leq 0, & \text{if } g_k^* = m_k, \end{cases}$$

Taking this into consideration, there exists $\lambda^* \in \mathbb{R}^m$, such that equation (5.10) can be reformulated as

$$-\nabla(d^T F(g^*)) - \sum_{i=1}^{p} \lambda_i^* \nabla h_i(g^*) \in \{y \in \mathbb{R}^n : \begin{cases} y_k \geq 0, & \text{if } g_k^* = M_k, \\ y_k = 0, & \text{if } g_k^* \in (m_k, M_k), \}. \\ y_k \leq 0, & \text{if } g_k^* = m_k, \end{cases} \tag{5.13}$$

Note that $\nabla(d^T F(g^*))_k = d_k f'(g_k^*)$ and $\nabla h_k(g^*) = (\xi_{1k}, \ldots, \xi_{nk})^T$ for all $k = 1, \ldots, n$. Then equation (5.13) is equivalent to the following three cases for $k = 1, ..., n$:

$$\begin{array}{ll} 0 \geq d_k f'(M_k) + \xi_k^T \lambda^* & \text{if } g_k^* = M_k, \\ 0 = d_k f'(g_k^*) + \xi_k^T \lambda^* & \text{if } g_k^* \in (m_k, M_k), \\ 0 \leq d_k f'(m_k) + \xi_k^T \lambda^* & \text{if } g_k^* = m_k, \end{array}$$

which completes the proof. □

If we revisit equation (5.12) or equation (5.9) we can reformulate all the conditions by using λ as a variable and then define g depending on the choice of λ. To achieve this we define a map $g : \mathbb{R}^p \to (\mathbb{R}_+ \cup \{0\})^n$ componentwise as

$$g_k(\lambda) = \mathrm{Pr}_{[m_k, M_k]} \left(f'^{-1}(-\frac{\xi_k^T \lambda}{d_k}) \right) \tag{5.14}$$

$$= \begin{cases} M_k, & \text{if } -\frac{\xi_k^T \lambda}{d_k} \geq f'(M_k), \\ f'^{-1}(-\frac{\xi_k^T \lambda}{d_i}), & \text{if } f'(m_k) < -\frac{\xi_k^T \lambda}{d_k} < f'(M_k), \quad (k = 1, \ldots, n) \\ m_k, & \text{if } -\frac{\xi_k^T \lambda}{d_k} \leq f'(m_k). \end{cases} \tag{5.15}$$

We use this definition to state another optimality criteria for our optimization problem which leads to the desired equation in λ.

Theorem 5.3.5. *A vector $g^* \in \mathbb{R}^n$ is the unique solution of the optimization problem* (5.3) *if and only if there exists a multiplier $\lambda^* \in \mathbb{R}^p$ such that $g(\lambda^*)$ defined in* (5.14) *satisfies*

$$h(g(\lambda^*)) = 0. \tag{5.16}$$

Proof. If (g^*, λ^*) are given such that (5.12) holds, we define $g(\lambda^*)$ and it is easy to check with the three given cases that $g(\lambda^*) = g^*$.
On the other hand, if for some λ^* the vector $g(\lambda^*)$ satisfies (5.16), then by a quick verification we see that $(g(\lambda^*), \lambda^*)$ also satisfies (5.12). This completes the proof. □

The last theorem states that finding a solution of the calibration problem (5.3) is equivalent to solve the equation

$$\psi(\lambda) = 0,$$

where

$$\psi : \mathbb{R}^p \to \mathbb{R}^p,\ \lambda \mapsto \bar{X}^T g(\lambda) - t_x$$

with

$$g_k(\lambda) = \Pr_{[m_k, M_k]}\left(f'^{-1}(-\frac{\xi_k^T \lambda}{d_k})\right) \quad (k = 1, \ldots, n).$$

Due to the nonsmoothness of the projection, ψ is not continuously differentiable and the standard Newton's method cannot be applied. A more general method that can be applied is the semismooth Newton method, which requires that ψ has to be semismooth.

Lemma 5.3.6. *ψ is (strongly) semismooth, if f'^{-1} is (strongly) semismooth.*

Proof. Let f'^{-1} be (strongly) semismooth. The projection $\Pr_{[m_i, M_i]} : \mathbb{R} \to \mathbb{R}$, $y \mapsto \Pr_{[m_i, M_i]}(y)$ can be written as a composition of the (strongly) semismooth functions min and max (cf. Kanzow, 2005). $\Pr_{[m_i, M_i]}(y) = \min\{M_i, \max\{m_i, y\}\}$, so we can apply the chain rule (Lemma 4.5.7) and deduce that the projection is (strongly) semismooth. Further, g is a composition of the projection and f'^{-1}, so g is (strongly) semismooth. t_{x_k}, ξ_{ki} are constant and as

$$\psi_i(\lambda) = \sum_{k=1}^{n} \xi_{ki} g_k(\lambda) - t_{x_k}, \quad i = 1, \ldots, p,$$

it follows with Lemma 4.5.5 (ii) that all ψ_k are (strongly) semismooth. As each component function is (strongly) semismooth, we deduce with Lemma 4.5.5 that ψ is (strongly) semismooth. □

We will now have a closer look for which functions f it holds that f'^{-1} and therefore ψ is (strongly) semismooth. As semismoothness in this case requires f'^{-1} to be locally Lipschitz

we cannot deduce general requirements for f but if $f : D \to \mathbb{R}$ is twice continuously differentiable and strictly convex it follows that $f' : D \to \mathbb{R}$ is continuously differentiable and strictly monotonically increasing. Therefore there exists an inverse function $f'^{-1} : f'(D) \to \mathbb{R}$ which is continuous and strictly monotonically increasing. If further $f''(x) \neq 0$ for all $x \in D$ the inverse function f'^{-1} is differentiable on $f'(D)$. Furthermore, if f'^{-1} as well as $(f'^{-1})'$ are locally Lipschitz it follows that f'^{-1} is strongly semismooth.

Now we regard the two given functions and check whether their inverse derivatives are strongly semismooth.

Lemma 5.3.7. *(i) $f_1'^{-1}$ is strongly semismooth.*

(ii) $f_2'^{-1}$ is strongly semismooth.

Proof. (i) Since $f_1(g_i) = \frac{(g_i-1)^2}{2}$ we obtain $f_1'^{-1}(y) = y + 1$ which is continuously differentiable and, following Lemma 4.5.2, therefore strongly semismooth.

(ii) Since $f_2(g_i) = g_i \cdot \ln(g_i) - g_i + 1$ it is obvious that $f_2'^{-1}(y) = e^y$ which is continuously differentiable and, following Lemma 4.5.2, therefore strongly semismooth.

□

We sum up the former statements to show the strong semismoothness of ψ.

Lemma 5.3.8. *ψ is strongly semismooth for the cases that $f = f_1$ and $f = f_2$.*

Proof. Follows because of Lemma 5.3.7 as well as Lemma 5.3.6. □

As ψ is strongly semismooth we can apply the semismooth Newton method for solving the nonsmooth equation $\psi(\lambda) = 0$, which will be presented in the following section.

5.4 Semismooth Newton Method

In each iteration of Newton's method for a smooth functions $F : \mathbb{R}^n \to \mathbb{R}^n$ a linear system of equations

$$J_F(x^k)s^k = -F(x^k)$$

with $J_F(x^k)$ denoting the Jacobian of F in x^k is solved in s^k. The resulting next iterate is computed as

$$x^{k+1} = x^k + s^k.$$

This method is locally quadratically convergent if the initial value $x^0 \in \mathbb{R}^n$ is close enough to the solution x^* satisfying $F(x^*) = 0$ and the inverse of the Jacobian exists in x^* and satisfies $\|J_F(x^*)^{-1}\| \leq \beta$, where $\beta > 0$. Furthermore, the Jacobian is required to be Lipschitz (cf. Dennis and Schnabel, 1983) and the regularity in x^* ensures the regularity of the Jacobian for all x near x^* (cf. Banach's lemma).

Regarding nonsmooth functions G, the semismooth Newton method by Qi (1993) reads as follows.

Algorithm 5.1 Semismooth Newton method

Input: $G : \mathbb{R}^n \to \mathbb{R}^n$ locally Lipschitz, $x^0 \in \mathbb{R}^n$ initial iterate
  while $\| G(x^k) \| \geq \epsilon$ **do**
    choose $H_k \in \partial_B G(x^k)$
    solve $H_k s^k = -G(x^k)$
    $x^{k+1} = x^k + s^k$
    $k \leftarrow k + 1$
  end while
  return solution x^k

The original version given in Qi and Sun (1993) choses $H_k \in \partial G(x^k)$ instead of $\partial_B G(x^k)$ and the local convergence theorem assumes all $H \in \partial G(x^*)$ to be nonsingular. As this condition is too strong, consider $G(x) = |x|$ where the Newton method converges trivially, we stick to the version given in Qi (1993) requiring $H_k \in \partial_B G(x^k)$. Then the assumption that all $H \in \partial G(x^*)$ have to be nonsingular is replaced by another regularity condition which we will mention below.

Definition 5.4.1 (Strongly BD-regular)**.** *Let $G : \mathbb{R}^n \to \mathbb{R}^n$ be locally Lipschitz in x. Then G is* **strongly BD-regular** *at x, if all $H \in \partial_B G(x)$ are nonsingular.*

The following lemma shows that the BD-regularity guaranties the regularity of all elements $H \in \partial_B G(x)$ for all x near x^*. This will be needed in the convergence theorem.

Lemma 5.4.2. *Let $G : \mathbb{R}^n \to \mathbb{R}^n$ be locally Lipschitz and strongly BD-regular in x^*. Then holds:*

$$\exists \epsilon > 0, c > 0 : \| H^{-1} \| \leq c \quad \forall H \in \partial_B G(x),\ x \in B_\epsilon(x^*).$$

Proof. Refer to Qi (1993). □

Theorem 5.4.3 (Superlinear convergence)**.** *Suppose that x^* is a solution of $G(x) = 0$, and that $G : \mathbb{R}^n \to \mathbb{R}^n$ is semismooth and strongly BD-regular at x^*. Then the semismooth Newton method is well defined and converges to x^* superlinearly in a neighborhood of x^*.*

Proof. Refer to Qi (1993). □

Furthermore, quadratic convergence can be proven for the case that G is strongly semismooth.

Theorem 5.4.4 (Quadratic convergence)**.** *Suppose that x^* is a solution of $G(x) = 0$, and that $G : \mathbb{R}^n \to \mathbb{R}^n$ is strongly semismooth and strongly BD-regular at x^*. Then there exists $\epsilon > 0$ such that for all $x^0 \in B_\epsilon(x^*)$ the semismooth Newton method is well defined and generates a sequence $(x^k)_{k \in \mathbb{N}}$ converging quadratically to x^*.*

Proof. Refer to Kanzow (2005). □

In order to ensure global convergence Qi and Sun (1993) give an extension of the classical Newton-Kantorovich theorem whereas Qi (1993) proposes a hybrid method. We will discuss these approaches in detail in Chapter 6 and will now focus on the local behavior concerning our calibration problem.

Theorem 5.4.5. *Let ψ be BD-regular in λ^* satisfying $\psi(\lambda^*) = 0$.*

(i) The semismooth Newton method converges quadratically to λ^* for all starting points λ^0 close to λ^*.

(ii) $g(\lambda^*)$ solves the calibration problem (5.3).

Proof. (i) Following Lemma 5.3.8, ψ is strongly semismooth. Hence Theorem 5.4.4 can be applied which states the proof.

(ii) Since λ^* is solution of $\psi(\lambda) = 0$, following Theorem 5.3.5, $g(\lambda^*)$ is the solution of the calibration problem (5.3).

□

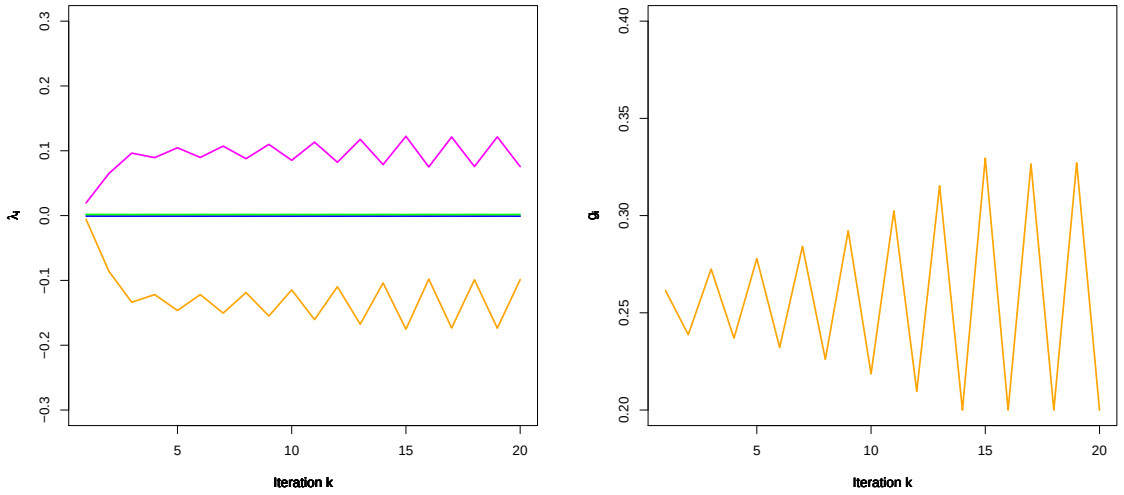
The last theorem makes clear that the semismooth Newton method can be applied for solving the standard calibration problem. In the following section we will study the behavior of the semismooth Netwon method and other competing methods.

5.5 Numerical Aspects

The algorithms were tested on an example included in the 'sampling' package in R. It contains approximately 200 calibration variables and eight calibration benchmarks. Nevertheless, the algorithms can also be applied to higher dimensional problems. The algorithm were implemented in R and tests were run on a common desktop PC with an Intel(R) Core(TM)2 Duo CPU with 3.00 GHz and an internal memory of 4 GB.

As already pointed out, the Newton-type method with projection proposed by Vanderhoeft (2001) may lead to convergence problems. During our tests we noticed that in a few instances some indices switched from being active into being inactive after every iteration. This effect, also called zig-zagging, is well known in constrained optimization and is due to an improper active set strategy. In Figure 5.1 we can see the switching values of certain components of the Lagrange multiplier and a sample variable. As the values of g directly depend on λ the switching λ results in switching weights g. This was not encountered when using the semismooth Newton method.

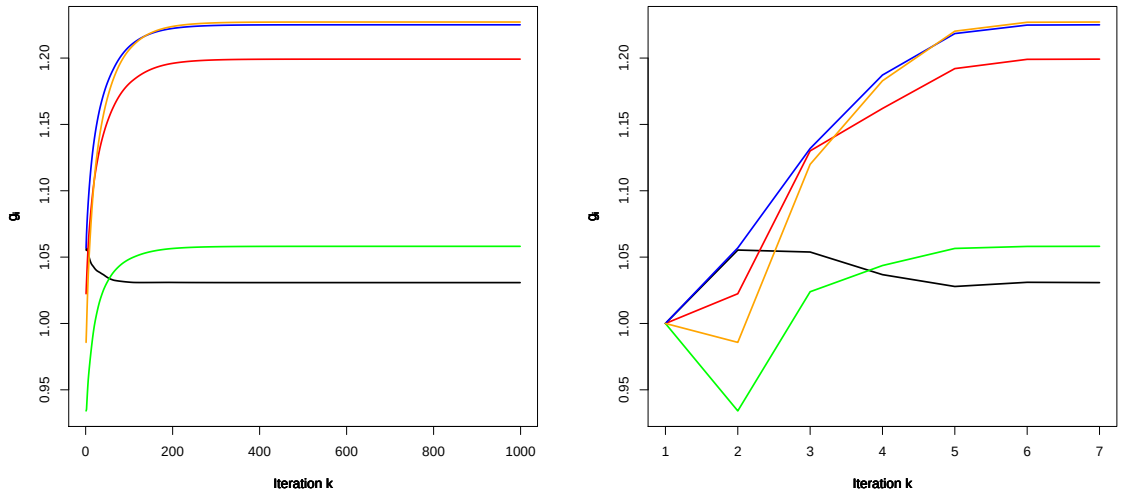
Figure 5.2 shows the convergence of some calibration factors of the Newton-type method with projection and the nonsmooth Newton method for a feasible example. We can see that the Newton-type method with projection delivers a rough approximation of the optimal weights after an acceptable number of iterations but needs much more iterations to reach the optimal



(a) Lagrange multiplier (b) Sample calibration factor

Figure 5.1: Zig-zagging of certain components of the Lagrange multiplier (a) and a sample calibration factor (b) using the Newton-type method with projection

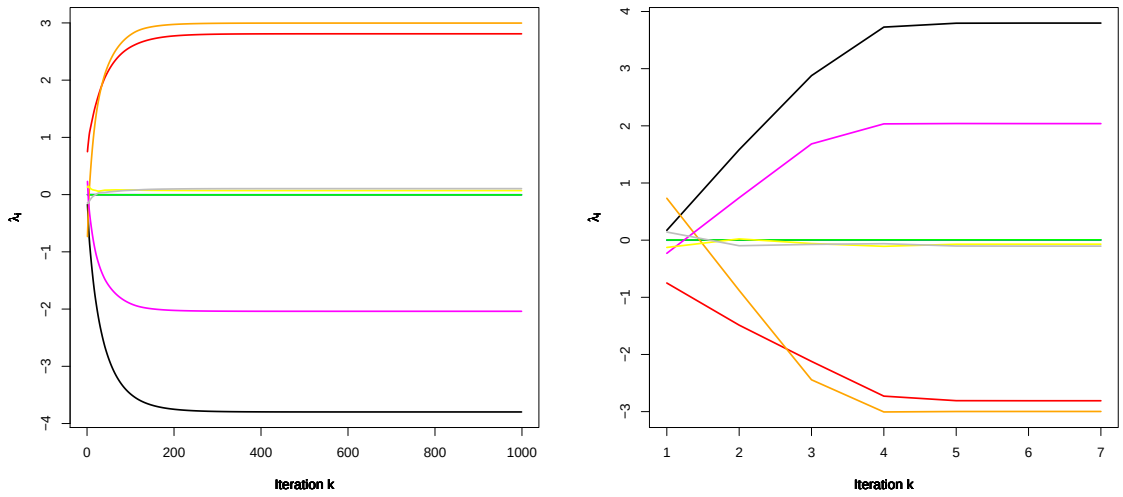
weights exactly. In contrast to this, the semismooth Newton method determines the optimal weights after only few steps.



(a) Newton-type method with projection (b) Semismooth Newton method

Figure 5.2: Convergence of some calibration factors g_i using the Newton-type method with projection (a) and the semismooth Newton method (b)

The convergence of the components of the Lagrange multiplier is shown in Figure 5.3. We can state that the behavior of the Lagrange multiplier can be compared to the behavior of the calibration factors. As the Newton-type method with projection uses kind of a 'wrong' Lagrange multiplier leading to a weight that afterwards is projected on the upper and lower bound, the limits are different from the ones of the semismooth Newton method.



(a) Newton-type method with projection

(b) Semismooth Newton method

Figure 5.3: Convergence of certain components of the Lagrange multiplier using the Newton-type method with projection (a) and the semismooth Newton method (b)

As indicated by the convergence theorem, we can observe the quadratic convergence of the semismooth Newton method also numerically (cf. Table 5.1). In contrast, the Newton-type method with projection shows a fairly slow convergence locally. Keep in mind that we did not apply any step size rule because, as we will see in Chapter 6, those methods need a more sophisticated treatment and further assumptions.

	$\|\lambda^k - \lambda^*\|$	
k	semismooth Newton method	Newton-type method with projection
1	4.546454	0.132
2	1.197537	1.197537
3	1.646565	1.256575
4	0.498095	1.187136
5	$5.1583 \cdot 10^{-3}$	1.099326
6	$3.2433 \cdot 10^{-6}$	1.020406
7	$6.5050 \cdot 10^{-11}$	0.950875
8	0	0.890555
⋮	-	⋮
290	-	$3.1508 \cdot 10^{-11}$
291	-	$2.8353 \cdot 10^{-11}$
292	-	$1.6328 \cdot 10^{-11}$
293	-	$2.1859 \cdot 10^{-11}$
294	-	0

Table 5.1: Convergence for f_2

We also made a comparison of the following three methods using the quadratic function f_1, the so-called truncated linear case, as distance function.

(i) 'calib' by Tillé and Matei (2009) as listed in the R package 'sampling',

(ii) Newton-type methods with projections according to Vanderhoeft (2001),

(iii) semismooth Newton method.

	calib		Newton-type w. proj.		semismooth Newton		
n	it.	time[sec]	it.	time[sec]	it.	time[sec]	ϵ
18,500	3	0.058	269	4.573	6	0.387	10^{-5}
18,500	2	0.052	40	0.720	5	0.452	10^{-5}
18,500	1	0.027	221	3.956	4	0.335	10^{-5}
18,500	1	0.032	41	0.898	4	0.454	10^{-5}
18,500	2	0.046	140	2.742	5	0.545	10^{-5}
18,500	3	0.063	181	3.595	6	0.622	10^{-5}

Table 5.2: Computing effort for differing data of same problem size using f_1

When using the truncated linear method we can state that the 'calib' function is the fastest algorithm with regard to the number of iterations and also time. The semismooth Newton method needs more iterations than the calib algorithm. The advantage of 'calib' is a very aggressive detection of the active indices which leads quickly to small dimensional problems to be solved in the following iterations (cf. pegging algorithms). To our knowledge, there does not exist a convergence statement. In Table 5.2, we list the results for various data sets of the same size.

The computing effort for examples with an increasing number of the same variables can be found in Table 5.3. The termination criterion $\|\bar{X}^T g - t_x\| \leq \epsilon$ and g being feasible was adjusted depending on the number of variables. As a starting point we used the optimal

Lagrange multiplier of the optimization problem without box constraint. Table 5.3 also shows a linear increase of the computing time with regard to the size of the variables. This is plausible, because the linear system to be solved is in the dual variable λ, where the dimension stays constant.

	calib		Newton-type w. proj.		semismooth Newton		
n	it.	time[sec]	it.	time[sec]	it.	time[sec]	ϵ
185	3	0.003	253	0.120	4	0.006	10^{-6}
1,850	3	0.007	279	0.590	4	0.027	10^{-6}
18,500	3	0.057	279	4.936	4	0.233	10^{-5}
185,000	3	0.667	270	51.815	4	2.775	10^{-4}
1,850,000	3	7.093	242	479.225	4	29.456	$2 \cdot 10^{-2}$

Table 5.3: Computing effort for different problem sizes with f_1

We also ran the test set on a different distance function f_2, the truncated multiplicative case. In this case, the 'calib' algorithm can no longer be applied in the existing software because we have a more complicated objective function. The Newton-type method with projection also cannot be applied due to convergence problems. The semismooth Newton method can still be used and delivers excellent results. Table 5.4 shows that the truncated multiplicative case needs more iterations and time compared to the truncated linear case. This can be explained by the more complicated objective function which takes more time to be evaluated.

	semismooth Newton		
n	it.	time[sec]	ϵ
185	8	0.012	10^{-6}
1,850	8	0.060	10^{-6}
18,500	8	0.479	10^{-5}
185,000	8	4.977	10^{-4}
1,850,000	9	61.946	10^{-3}

Table 5.4: Computing effort for the truncated multiplicative method

Overall, the numerical results confirm the theoretical statements from the previous sections and show that the semismooth Newton method is a fast and reliable method to solve the calibration problem under investigation.

Chapter 6

Nonmonotone Step Size Rules for B-Differentiable Functions

The first step, my son, which one makes in the world, is the one on which depends the rest of our days.

— Voltaire

Suppose we want to solve a nonsmooth equation $f(x) = 0$ in $\mathbb{R}^n$. Such equations may occur in calibration of estimator weights, as mentioned in Chapter 5, when solving the calibration problem

$$\begin{aligned} \min_{g \in \mathbb{R}^n} \; & d^T F(g) \\ s.t. \; & \bar{X}^T g - t = 0 \\ & g \in [m, M]. \end{aligned}$$

This can be done by solving the equation

$$f(\lambda) = 0,$$

with

$$f : \mathbb{R}^n \to \mathbb{R}^n, \; \lambda \mapsto \bar{X}^T g(\lambda) - t$$

and

$$g_k(\lambda) = \Pr_{[m_k, M_k]} \left(h'^{-1}(-\frac{\xi_k^T \lambda}{d_k}) \right) \quad (k = 1, \ldots, n)$$

is a semismooth function. Another application comes from nonlinear variational inequality problems (VIP). We want to find $x \in C$ such that

$$\langle F(x), y - x \rangle \geq 0 \quad \forall y \in C,$$

where C denotes a closed convex subset of $\mathbb{R}^n$ and $F : \mathbb{R}^n \to \mathbb{R}^n$.

As mentioned in Facchinei and Pang (2003a), x_* solves the variational inequality problem (VIP) if and only if x_* solves the nonsmooth normal equation (NE)

$$f(x) = x - \mathrm{Pr}_C(x - F(x)) = 0,$$

with $\mathrm{Pr}_C(\cdot)$ denoting the projection operator of $\mathbb{R}^n$ onto C. Although being a large research area, we will neither deal with the applications of VIPs nor will we talk about the history of VIPs in detail. As we are especially interested in solving nonsmooth equations, we have a closer look at solution methods based on the nonsmooth formulation of VIPs. Such nonsmooth equations also occur when dealing with nonlinear complementarity problems (NCP), which are special cases of VIPs, or its reformulation using Karush-Kuhn-Tucker conditions.

In the smooth case the vector valued equation $f(x) = 0$ can be solved by Newton's method, where in each iteration the system of linear equalities

$$J_f(x_k)d_k = -f(x_k)$$

is solved in d_k. The new iterate is given by

$$x_{k+1} = x_k + \alpha_k d_k$$

with an appropriate step size α_k. In the nonsmooth case the Jacobian does not exist everywhere, so the first equation is replaced by

$$f(x_k) + A(x_k)(d_k) = 0,$$

where $A(x_k)(d_k)$ is a substitute for the Jacobian in x_k applied to d_k. The existing methods reviewed below mainly differ in the choice of $A(x_k)(d_k)$ and the assumptions on f.

An early work on solving nonsmooth equations was done by Kummer (1988), who proposes a Newton type method for solving general non-differentiable functions and proves local convergence. In the special case of having a locally Lipschitz function and using the generalized Jacobian of Clarke (1983), that is $V_k d_k = -f(x_k)$, $V_k \in \partial f(x_k)$, he proves local superlinear convergence of the iterates x_k to x_*, under the assumption that $|\partial f(x_*)| = 1$ has to be satisfied.

The assumption mentioned before implies that f is also semismooth at x_*, but not vice versa. Qi and Sun (1993) investigate the convergence property for the more general case of having semismooth functions and applying a method similar to the one presented in Kummer (1988). They show local convergence and when assuming further Kantorovich-type conditions they can even prove global convergence. A detailed discussion on this semismooth Newton method is given in Chapter 5 and in the case of choosing the 'right' element of the generalized Jacobian, the iterates of the therein mentioned semismooth Newton method coincide with the iterates of the generalized Newton's method given in Pang (1990). There, the function f is assumed to be B-differentiable and $A(x_k)(d_k)$ is replaced by the B-derivative of f in x_k applied to d_k. Apart from this, Martínez and Qi (1995), Facchinei et al. (1996), Facchinei and Kanzow (1997) and Facchinei and Pang (2003b) adapt the extensions of Newton's method in the smooth case and derive a semismooth inexact Newton method as well as a semismooth

inexact Lenvenberg-Marquardt Newton method which are, under certain conditions, both locally convergent.

In order to obtain global convergence of the generalized Newton's method for B-differentiable functions, a line search is added in Pang (1990) yielding global superlinear convergence. This method is called damped Newton method and was motivated by the need for a computationally robust method with guaranteed convergence. It is also the first work on global Newton methods for solving nonsmooth equations. However, it has a theoretical drawback: the convergence requires a Fréchet differentiability assumption at a limit point of the produced sequence.

A modification of the damped Newton method in Pang (1990) is given in Pang (1991) where the search direction d_k is determined by $f(x_k) + f'(x_k; d_k) = 0$ with $f'(x_k; d_k)$ being the directional derivative of the B-differentiable function f in x_k along d_k. Furthermore, applications to NCPs and VIPs are discussed in detail. Another modification is given in Han et al. (1992), where a generalization of the damped Newton method for locally Lipschitz functions is presented. Nevertheless, the direction-generation step in this method calls for the solution of a nonlinear program that in general is neither smooth nor convex. Pang and Qi (1993) propose a variant of this method in which the direction-generation subproblems are convex quadratic programs that are always solvable. An alternative trust region approach is given in Qi and Sun (1994) and compared to the method of Han et al. (1992).

Ralph (1994) proposes a path search algorithm as an alternative to Pang's line search algorithm. The path search replaces the traditional line search and leads to the same convergence properties. As line search methods are more common and more reliable, we will not discuss this algorithm in this work and refer to the original work for further details.

Qi (1993) seizes the ideas of Pang (1991), Han et al. (1992) and Qi and Sun (1993) to study the convergence of these methods in combination with certain regularity conditions, e.g., BD-regularity. Furthermore, he proposes a hybrid method of the methods given in Han et al. (1992) and Qi and Sun (1993). In the latter case the search direction d_k is determined by $V_k d_k = -f(x_k)$, $V_k \in \partial_B f(x_k)$. It should be noted that V_k is taken from the B-subdifferential of f in x_k instead of the generalized Jacobian. Using a line search and taking further assumptions ensures global and local quadratic convergence. This idea of combining two methods is also taken into account in Ito and Kunisch (2009), where they show global convergence and in the case of f being semismooth they show local superlinear convergence.

In contrast to the globalization methods above, that use a **monotone** line search relying on the merit function

$$\theta : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\},\ x \mapsto \theta(x) = \|f(x)\|^2,$$

the method presented in Ferris and Lucidi (1994) uses a **nonmonotone** line search based on the nonmonotone line search for smooth functions introduced in Grippo et al. (1986) to obtain global convergence. Furthermore, they do not assume an explicit rule for calculating the search direction.

Fischer (1992) uses the Fischer-Burmeister-function to reformulate the Karush-Kuhn-Tucker

conditions arising from an inequality constrained optimization problem as a system of nonsmooth equations. The advantage of this approach concerning global convergence lies in the feature that the gained merit function is smooth, so the standard convergence proof can be applied as it is done by De Luca et al. (1996) for NCPs.

Harker and Xiao (1990) do not make use of the Fischer-Burmeister-function and therefore get a nonsmooth merit function for their damped Newton method. They show global convergence under certain conditions and also add a numerical example. A derivative free algorithm for complementarity problems is given in Fischer (1997) and Kanzow (2004) also proposes an inexact Newton method for large scale complementarity problems. Furthermore, there also exist trust region approaches and in the case of examining mixed complementarity problems (MCP) Ferris et al. (1998) consider a general algorithmic framework which is applied to the PATH solver by Dirkse and Ferris (1995) relying on the path search presented in Ralph (1994). Kanzow (2000) also studies the theoretical and numerical properties of incorporating global optimization algorithms, namely a tunneling and a filled function method, to a standard semismooth Newton-type method for solving MCPs.

Most of the methods using line search require the decrease of the function values to be monotone. However, as known in smooth optimization, for certain problems a nonmonotone line search such as the ones given in Grippo et al. (1986) or Zhang and Hager (2004) may deliver better results. In nonsmooth optimization, so far only the nonmonotone line search of Ferris and Lucidi (1994) was studied. We generalize the approaches for B-differentiable functions mentioned above by using nonmonotone step size rules deriving from Armijo (1966) and apply the nonmonotone step size rule of Zhang and Hager (2004) to nonsmooth optimization. Note that for the sake of readability the iteration indices in this chapter are written as subscript instead of superscript to circumvent a mix up with the exponent of a scalar.

6.1 Preliminary Results

We now consider some theorems and lemmata which will be useful for the following approaches.

Lemma 6.1.1. *Let* $(a_k)_{k\in\mathbb{N}}, (\epsilon_k)_{k\in\mathbb{N}}$ *be sequences with*

$$a_k \geq a_{min},\ \epsilon_k \geq 0,\ \sum_{k=1}^{\infty} \epsilon_k < \infty.$$

If

$$a_{k+1} \leq a_k + \epsilon_k,$$

then there exists $a_* \geq a_{min}$ *such that*

$$a_k \xrightarrow[k\to\infty]{} a_*.$$

Proof. Refer to Tichatschke and Kaplan (1994). □

Theorem 6.1.2. *Let $\theta : \mathbb{R}^n \to \mathbb{R}$ be bounded below by 0, $\rho : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\}$, $(x_k)_{k\in\mathbb{N}}, (d_k)_{k\in\mathbb{N}} \subset \mathbb{R}^n$, $(\alpha_k)_{k\in\mathbb{N}}, (\lambda_k)_{k\in\mathbb{N}}, (\nu_k)_{k\in\mathbb{N}} \subset \mathbb{R}_+ \cup \{0\}$,*

$$x_{k+1} = x_k + \alpha_k d_k$$

with

$$\lambda_k + \theta(x_k) \geq \rho(x_k) \geq 0 \tag{6.1}$$

and for given $\sigma > 0$ it holds

$$\theta(x_{k+1}) - \theta(x_k) \leq -\sigma\alpha_k\theta(x_k) + \nu_k. \tag{6.2}$$

If

$$\sum_{k=1}^{\infty} \lambda_k\alpha_k < \infty \quad \text{and} \quad \sum_{k=1}^{\infty} \nu_k < \infty \tag{6.3}$$

we have:

(*i*) there exists $\theta_* \geq 0$ such that $\theta(x_k) \xrightarrow[k\to\infty]{} \theta_*$,

(*ii*) $\sum_{k=1}^{\infty} \alpha_k\rho(x_k) < \infty$.

Proof. The proof is in line with the proof given in Sachs and Sachs (2011), which is slightly adapted.

(*i*) It holds due to (6.1) and (6.2) that

$$\begin{aligned}\theta(x_{k+1}) - \theta(x_k) &\leq -\sigma\alpha_k\theta(x_k) + \nu_k \\ &\leq \sigma\lambda_k\alpha_k + \nu_k.\end{aligned}$$

If we define $a_k := \theta(x_k)$ and $\epsilon_k := \sigma\lambda_k\alpha_k + \nu_k$ this leads to:

$$\begin{aligned}&a_{k+1} \leq a_k + \epsilon_k,\ \epsilon_k \geq 0 \quad \forall\ k, \\ &\sum_{k=1}^{\infty} \epsilon_k = \sigma \sum_{k=1}^{\infty} \lambda_k\alpha_k + \sum_{k=1}^{\infty} \nu_k < \infty, \\ &a_k = \theta(x_k) \text{ bounded below by 0.}\end{aligned}$$

Therefore, we can apply Lemma 6.1.1 so there exists $a_* \geq 0$ such that

$$a_k \xrightarrow[k\to\infty]{} a_*.$$

As $a_k = \theta(x_k)$ there exists $\theta_* \geq 0$ and we get

$$\theta(x_k) \xrightarrow[k\to\infty]{} \theta_*.$$

(*ii*) Using (6.1) and (6.2) we deduce that

$$\alpha_k \rho(x_k) \leq \alpha_k \lambda_k + \alpha_k \theta(x_k) \leq \alpha_k \lambda_k + \frac{1}{\sigma}(\nu_k + \theta(x_k) - \theta(x_{k+1})).$$

Taking the sum over $k = 1, \ldots, j$ yields

$$\begin{aligned}\sum_{k=1}^{j} \alpha_k \rho(x_k) &\leq \sum_{k=1}^{j} \alpha_k \lambda_k + \frac{1}{\sigma} \sum_{k=1}^{j} \nu_k + \frac{1}{\sigma} \sum_{k=1}^{j} (\theta(x_k) - \theta(x_{k+1})) \\ &= \sum_{k=1}^{j} \alpha_k \lambda_k + \frac{1}{\sigma} \sum_{k=1}^{j} \nu_k + \frac{1}{\sigma}(\theta(x_1) - \theta(x_{j+1})).\end{aligned}$$

If $j \to \infty$ we conclude with (6.3) that

$$\sum_{k=1}^{\infty} \alpha_k \rho(x_k) \leq \sum_{k=1}^{\infty} \alpha_k \lambda_k + \frac{1}{\sigma} \sum_{k=1}^{\infty} \nu_k + \frac{1}{\sigma}(\theta(x_1) - \theta_*) < \infty.$$

□

For given nonsmooth functions $f : \mathbb{R}^n \to \mathbb{R}^n$ and $\theta : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\},\ x \mapsto \theta(x) = \|f(x)\|^2$, we obtain the following statements simplifying the application of the derivative to the search direction.

Lemma 6.1.3. *Let $x \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}^n$ be B-differentiable and $\theta : \mathbb{R}^n \to \mathbb{R},\ x \mapsto \theta(x) = f(x)^T f(x)$. $A_f(x)(d)$ denotes the B-derivative of f in x applied to d and $A_\theta(x)(d)$ denotes the B-derivative of θ in x applied to d. Then θ is also B-differentiable and if d is chosen such that $A_f(x)(d) = -f(x)$ it holds*

$$A_\theta(x)(d) = 2f(x)^T A_f(x)(d) = -2f(x)^T f(x) = -2\theta(x).$$

Lemma 6.1.4. *Let $x \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}^n$ be locally Lipschitz and $\theta : \mathbb{R}^n \to \mathbb{R},\ x \mapsto \theta(x) = f(x)^T f(x)$. $V_f(x) \in \partial f(x)$ denotes an element of the generalized Jacobian of f in x and $V_\theta(x) \in \partial\theta(x)$ denotes an element of the generalized Jacobian of θ in x. Then θ is also locally Lipschitz and if d is chosen such that $V_f(x)d = -f(x)$ it holds*

$$V_\theta(x)d = 2f(x)^T V_f(x)d = -2f(x)^T f(x) = -2\theta(x).$$

Remark 6.1.5. *If f is continuously differentiable and d is chosen such that $J_f(x)d = -f(x)$ we get*

$$\nabla\theta(x)^T d = 2f(x)^T J_f(x)d = -2f(x)^T f(x) = -2\theta(x).$$

6.2 Convergence of a Nonmonotone Step Size Rule for B-differentiable Functions

As mentioned at the beginning of the chapter, we want to solve $f(x) = 0$ by minimizing the merit function $\theta(x) = \|f(x)\|^2$. In case of having smooth functions $\tilde{f}$ and $\tilde{\theta}$, the well known Armijo's rule for smooth functions guarantees a sufficient decrease in the smooth objective function $\tilde{\theta} : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\}$ by determining an appropriate step size $\alpha_k > 0$ for the iteration given by

$$x_{k+1} = x_k + \alpha_k d_k,$$

where d_k is a certain descent/search direction. A common notation can be found in Nocedal and Wright (2006) or in the original work by Armijo (1966), which requires α_k to satisfy

$$\tilde{\theta}(x_k + \alpha_k d_k) - \tilde{\theta}(x_k) \leq c_1 \alpha_k \nabla\tilde{\theta}(x_k)^T d_k$$

for some constant $c_1 \in (0, 1)$. Before we are able to formulate a nonmonotone Armijo's rule, which is based on the formulation given in Sachs and Sachs (2011), we have to make some assumptions. Also, keep in mind that, if $V_k d_k = -\tilde{f}(x_k)$ with $V_k \in \partial_B \tilde{f}(x_k)$ the expression $\nabla\tilde{\theta}(x_k)^T d_k$ in the continuously differentiable case can be replaced by $-2\tilde{\theta}(x_k)$. For a detailed discussion refer to De Luca et al. (1996) as well as Ito and Kunisch (2009).

Throughout the following, we assume that $f : \mathbb{R}^n \to \mathbb{R}^n$ and therefore $\theta : \mathbb{R}^n \to \mathbb{R} \cup \{0\}$ are B-differentiable functions and from Algorithm 6.1 mentioned below we get sequences $(x_k)_{k\in\mathbb{N}}, (d_k)_{k\in\mathbb{N}} \subset \mathbb{R}^n$. Furthermore there exist sequences $(\nu_k)_{k\in\mathbb{N}}, (\alpha_k)_{k\in\mathbb{N}} \subset \mathbb{R}_+ \cup \{0\}$ and the assumptions (A.1)-(A.3) hold.

(A.1) $S = \{x \in \mathbb{R}^n : \ \theta(x) \leq \theta(x_0) + \sum_{k=1}^{\infty} \nu_k\}$ is bounded.

(A.2) There exists $\bar{\sigma} \in (0, 1)$ such that for all $x_k \in S$ there exists a search direction $d_k \in \mathbb{R}^n$ such that

$$\theta'(x_k; d_k) \leq -\bar{\sigma}\theta(x_k). \tag{6.4}$$

(A.3) $(d_k)_{k\in\mathbb{N}}$ is bounded.

Assumption (A.2) is motivated by the work of Ito and Kunisch (2009) and is used to guarantee that the nonmonotone Armijo's rule is well defined. We are now able to formulate the following globalization algorithm.

Algorithm 6.1 Globalized generalized Newton's method

Input: initial iterate $x_0 \in \mathbb{R}^n,\ \theta(x_0) \neq 0,\ k = 0,\ \epsilon > 0$
  while $\theta(x_k) > \epsilon$ **do**
    determine search direction d_k satisfying (6.4)
    determine step size α_k with an appropriate step size rule
    $x_{k+1} = x_k + \alpha_k d_k$
    $k \leftarrow k + 1$
  end while
  return solution x_k

The choice of an appropriate step size rule plays an essential role in the further examination of the globalized generalized Newton's method. Ito and Kunisch (2009) formulate a monotone Armijo based step size rule, but sometimes a nonmonotone step size rule leads to better convergence results. We now present a general nonmonotone Armijo based step size rule whose convergence behavior will be discussed in detail.

Algorithm 6.2 Nonmonotone Armijo's rule

Input: $\beta \in (0,1),\ \sigma \in (0, \bar{\sigma}),\ x_k, d_k \in \mathbb{R}^n,\ \nu_k \in \mathbb{R}_+ \cup \{0\},\ \alpha_{max} > 0,\ \theta(x_k) \neq 0$
Ensure: d_k is chosen such that (6.4) holds.
  if $\theta(x_k + \alpha_{max} d_k) - \theta(x_k) \leq -\sigma \alpha_{max} \theta(x_k) + \nu_k$ **then**
    $\alpha_k = \alpha_{max}$
  else
    determine smallest $l_k \in \mathbb{N}$ such that
    $\theta(x_k + \alpha_{max}\beta^{l_k} d_k) - \theta(x_k) \leq -\sigma \alpha_{max} \beta^{l_k} \theta(x_k) + \nu_k$
    $\alpha_k = \alpha_{max}\beta^{l_k}$
  end if
  return step size α_k

Before we check whether the nonmonotone Armijo's rule is well-defined we reconsider Lemma 4.3.6 and deduce the following.

Remark 6.2.1. *For the B-differentiable function θ it holds that*

$$\lim_{d \to 0} \frac{\theta(x+d) - \theta(x) - \theta'(x;d)}{\|d\|} = 0,$$

where $\theta'(x;d)$ denotes the directional derivative of f in direction d. Therefore

$$\theta(x_k + \alpha_{max}\beta^{l_k} d_k) - \theta(x_k) \leq \alpha_{max}\beta^{l_k}\theta'(x_k; d_k) + \phi(\alpha_{max}\beta^{l_k}\|d_k\|)\alpha_{max}\beta^{l_k}\|d_k\|, \quad (6.5)$$

where $\phi : \mathbb{R} \to \mathbb{R}_+ \cup \{0\},\ u \mapsto \phi(u)$ guarantees $\phi(u) \xrightarrow[u \to 0]{} 0$ and $\alpha_{max}\beta^{l_k} > 0$.

It is now easy to prove that the nonmonotone Armijo's rule is well-defined.

Lemma 6.2.2. *Under the assumptions mentioned before the nonmonotone Armijo's rule (Algorithm 6.2) is well-defined.*

Proof. We have to show that for all $x_k \in S$ there exists $\bar{\alpha}_k > 0$ such that

$$\theta(x_k + \alpha d_k) - \theta(x_k) \leq -\sigma\alpha\theta(x_k) + \nu_k \quad \forall \alpha \in [0, \bar{\alpha}_k].$$

Assume that for a given x_k with $\theta(x_k) \neq 0$ and for $\alpha_{max}\beta^{l_j} \to 0+$, which is equivalent to $l_j \to \infty$, it holds that

$$\theta(x_k + \alpha_{max}\beta^{l_j} d_k) - \theta(x_k) > -\sigma\alpha_{max}\beta^{l_j}\theta(x_k) + \nu_k \geq -\sigma\alpha_{max}\beta^{l_j}\theta(x_k).$$

Using (6.5) and $\phi(u) \xrightarrow[u\to 0]{} 0$, we get

$$\begin{aligned}\alpha_{max}\beta^{l_j}\theta'(x_k; d_k) + \phi(\alpha_{max}\beta^{l_j}\|d_k\|)\alpha_{max}\beta^{l_j}\|d_k\| &\geq \theta(x_k + \alpha_{max}\beta^{l_j} d_k) - \theta(x_k) \\ &> -\sigma\alpha_{max}\beta^{l_j}\theta(x_k)\end{aligned}$$

which is equivalent to

$$\theta'(x_k; d_k) + \phi(\alpha_{max}\beta^{l_j}\|d_k\|)\|d_k\| \geq -\sigma\theta(x_k).$$

Following (6.4), it holds that $\theta'(x_k; d_k) \leq -\bar{\sigma}\theta(x_k)$ and we deduce

$$-\bar{\sigma}\theta(x_k) + \phi(\alpha_{max}\beta^{l_j}\|d_k\|)\|d_k\| \geq -\sigma\theta(x_k).$$

Because of $\phi(\alpha_{max}\beta^{l_j}\|d_k\|)\|d_k\| \xrightarrow[l_j\to\infty]{} 0$ we get

$$0 \geq (\bar{\sigma} - \sigma)\theta(x_k) \geq 0.$$

As $(\bar{\sigma} - \sigma) > 0$ it follows that $\theta(x_k) = 0$, hence a contradiction. □

We are now able to formulate a convergence statement of the function values.

Theorem 6.2.3. *Let θ be bounded below by 0. Furthermore be $\sum_{k=1}^{\infty} \nu_k < \infty, \nu_k \geq 0$ for all k and $(x_k)_{k\in\mathbb{N}}, (d_k)_{k\in\mathbb{N}}$ generated by the generalized Newton's method (Algorithm 6.1) satisfy the nonmonotone Armijo's rule (Algorithm 6.2). Then it holds*

$$\theta(x_k) \xrightarrow[k\to\infty]{} 0.$$

Proof. Because of Lemma 6.2.2, the nonmonotone Armijo's rule is well defined and for all k it holds that

$$\theta(x_{k+1}) - \theta(x_k) \leq -\sigma\alpha_k\theta(x_k) + \nu_k.$$

If we set $\lambda_k = 0$ and $\rho(x_k) = \theta(x_k)$ with $\theta(x_k) \geq 0$ for all k, condition (6.1) from Theorem 6.1.2 is satisfied. Furthermore, θ is bounded below and (6.2) and (6.3) are as well satisfied.

Therefore, we can apply Theorem 6.1.2 and we conclude that

$$(i) \quad \text{there exists } \theta_* \geq 0 \text{ such that } \theta(x_k) \xrightarrow[k\to\infty]{} \theta_*, \tag{6.6}$$

$$(ii) \quad \sum_{k=1}^{\infty} \alpha_k \theta(x_k) < \infty, \tag{6.7}$$

which also ensures that

$$\alpha_k \theta(x_k) \xrightarrow[k\to\infty]{} 0. \tag{6.8}$$

(a) If $\lim_{k\to\infty} \alpha_k = 0$, we can state due to $\alpha_k = \alpha_{max}\beta^{l_k}$ and $\alpha_k > 0$ for all k that

$$\theta(x_k + \frac{\alpha_k}{\beta}\|d_k\|) - \theta(x_k) > -\sigma\frac{\alpha_k}{\beta}\theta(x_k) + \nu_k \geq -\sigma\frac{\alpha_k}{\beta}\theta(x_k).$$

Using (6.5) and $\phi(u) \xrightarrow[u\to 0]{} 0$ we get

$$\begin{aligned}\frac{\alpha_k}{\beta}\theta'(x_k; d_k) + \phi(\frac{\alpha_k}{\beta}d_k)\frac{\alpha_k}{\beta}\|d_k\| &\geq \theta(x_k + \frac{\alpha_k}{\beta}d_k) - \theta(x_k) \\ &> -\sigma\frac{\alpha_k}{\beta}\theta(x_k)\end{aligned}$$

which is equivalent to

$$\theta'(x_k; d_k) + \phi(\frac{\alpha_k}{\beta}\|d_k\|)\|d_k\| > -\sigma\theta(x_k).$$

Applying (6.4) leads to

$$-\bar{\sigma}\theta(x_k) + \phi(\frac{\alpha_k}{\beta}\|d_k\|)\|d_k\| > -\sigma\theta(x_k).$$

Therefore, we obtain

$$\Leftrightarrow \phi(\frac{\alpha_k}{\beta}\|d_k\|)\|d_k\| > (\bar{\sigma} - \sigma)\theta(x_k).$$

Following (A.3), $(d_k)_{k\in\mathbb{N}}$ is bounded and by assumption on α_k we get

$$\frac{\alpha_k}{\beta}d_k \xrightarrow[k\to\infty]{} 0 \;\Rightarrow\; \phi(\frac{\alpha_k}{\beta}d_k) \xrightarrow[k\to\infty]{} 0 \;\Rightarrow\; \phi(\frac{\alpha_k}{\beta}d_k)\|d_k\| \xrightarrow[k\to\infty]{} 0.$$

For $k \to \infty$ we obtain

$$0 \geq (\bar{\sigma} - \sigma)\theta_* \geq 0,$$

and as $(\bar{\sigma} - \sigma) > 0$ it follows that $\theta_* = 0$ so we get

$$\theta(x_k) \xrightarrow[k\to\infty]{} 0.$$

(b) Otherwise, there exists a subsequence $(\alpha_{k_j})_{j\in\mathbb{N}}, \delta > 0$ such that $\alpha_{k_j} > \delta$ for all $j \in \mathbb{N}$. Using (6.8) we get

$$\alpha_{k_j}\theta(x_{k_j}) \xrightarrow[j\to\infty]{} 0. \tag{6.9}$$

As $\alpha_{k_j} > \delta$ for all $j \in \mathbb{N}$, we deduce that $\theta(x_{k_j}) \xrightarrow[j\to\infty]{} 0$ so the subsequence $(\theta(x_{k_j}))_{j\in\mathbb{N}}$ converges to zero. Following (6.6) we know that $(\theta(x_k))_{k\in\mathbb{N}}$ converges to θ_*. As the limits have to be equal, we deduce $\theta(x_k) \xrightarrow[k\to\infty]{} 0$.

□

Before we will have a look at certain nonmonotone step size rules, we will mention a little disadvantage of our algorithm compared to the one presented by Ito and Kunisch (2009). Under some additional assumptions they are able to prove the convergence of the iterates x_k to x_*. This proof makes use of the monotonicity of the normed function values $\|f(x_k)\|$, which of course is not the case in our nonmonotone setting.

Our setting allows only to prove that the iterates of the function values $\theta(x_k)$ converge to 0. Nevertheless, because of (A.1), S is bounded and we can deduce that there exists at least one subsequence $(x_{k_j})_{j\in\mathbb{N}}$ converging to an accumulation point $x_* \in S$ satisfying $\theta(x_*) = 0$.

This gets clearer if we have a look at the following example. Let $f : \mathbb{R} \to \mathbb{R}$, $x \mapsto f(x) = x^2 - 1$ and $\theta : \mathbb{R} \to \mathbb{R}_+ \cup \{0\}$, $x \mapsto \theta(x) = 0.5 \cdot |x^2 - 1|^2$. There are two accumulation points in

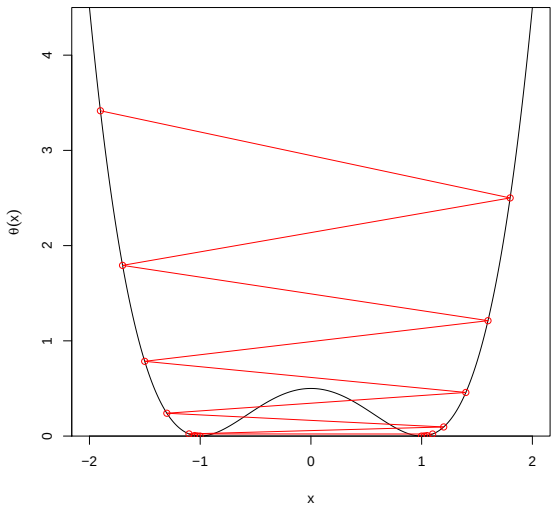


Figure 6.1: Zigzagging of possible iterates of the merit function

-1 and 1 which we will denote by x_* and x_{**}. If the iterates zigzag between a neighborhood of x_* and x_{**}, the function values $\theta(x_k)$ converge to 0 but we cannot define a limit of the sequence $(x_k)_{k\in\mathbb{N}}$.

However, we are able to define two subsequences $(x_{k_j})_{j\in\mathbb{N}}$ and $(x_{k_l})_{l\in\mathbb{N}}$. The first one consists of the iterates in the neighborhood of x_* and the second one of the iterates in the neighborhood of x_{**}. Hence, we get two converging subsequences $x_{k_j} \xrightarrow[j\to\infty]{} x_*$ and $x_{k_l} \xrightarrow[l\to\infty]{} x_{**}$.

Our approach can also be applied to hybrid methods similar to the ones given in Ito and Kunisch (2009) or Qi (1993). Those methods first try a full step with Newton search direction which is tested with kind of a 'watchdog step'. If this step is 'good' a Newton step is performed. Otherwise, a search direction satisfying (6.4) is determined and a monotone line search is performed. This makes it also possible to propose a theorem determining the convergence rate.

Algorithm 6.3 Hybrid generalized Newton's method

Input: $x_0 \in \mathbb{R}^n,\ \theta(x_0) \neq 0,\ k = 0,\ \epsilon > 0$
  while $\theta(x_k) > \epsilon$ **do**
    if existing **then**
      determine search direction d_k satisfying $V_k d_k = -f(x_k),\ V_k \in \partial_B f(x_k)$
      if $\theta(x_k + d_k) - \theta(x_k) \leq -\sigma\theta(x_k)$ **then**
        $x_{k+1} = x_k + d_k$
      else
        determine search direction d_k satisfying (6.4)
        determine step size α_k with the monotone step size rule 6.4
        $x_{k+1} = x_k + \alpha_k d_k$
      end if
    else
      determine search direction d_k satisfying (6.4)
      determine step size α_k with the monotone step size rule 6.4
      $x_{k+1} = x_k + \alpha_k d_k$
    end if
    $k \leftarrow k + 1$
  end while
  return solution x_k

The corresponding Armijo based monotone step size rule reads as follows.

Algorithm 6.4 Monotone step size rule

Input: $\beta \in (0,1),\ \sigma \in (0, \bar{\sigma}),\ x_k, d_k \in \mathbb{R}^n,\ \theta(x_k) \neq 0$
Ensure: d_k is chosen such that (6.4) holds.
  determine smallest $l_k \in \mathbb{N} \cup \{0\}$ such that
  $\theta(x_k + \beta^{l_k} d_k) - \theta(x_k) \leq -\sigma\beta^{l_k}\theta(x_k)$
  $\alpha_k = \beta^{l_k}$
  return step size α_k

Lemma 6.2.4. *Under the assumptions mentioned before the monotone step size rule (Algorithm 6.4) terminates after a finite number of iterations.*

Proof. We have to show that for all $x_k \in S$ there exists $\bar{\alpha}_k > 0$ such that

$$\theta(x_k + \alpha d_k) - \theta(x_k) \leq -\sigma\alpha\theta(x_k) \quad \forall \alpha \in [0, \bar{\alpha}_k].$$

Assume that for a given x_k with $\theta(x_k) \neq 0$ and for $\beta^{l_j} \to 0+$, which is equivalent to $l_j \to \infty$, it holds that

$$\theta(x_k + \beta^{l_j} d_k) - \theta(x_k) > -\sigma\beta^{l_j}\theta(x_k).$$

Using (6.5) and $\phi(u) \xrightarrow[u\to 0]{} 0$ we get

$$\begin{aligned}\beta^{l_j}\theta'(x_k; d_k) + \phi(\beta^{l_j}\|d_k\|)\beta^{l_j}\|d_k\| &\geq \theta(x_k + \beta^{l_j} d_k) - \theta(x_k) \\ &> -\sigma\beta^{l_j}\theta(x_k),\end{aligned}$$

which is equivalent to

$$\theta'(x_k; d_k) + \phi(\beta^{l_j}\|d_k\|)\|d_k\| \geq -\sigma\theta(x_k).$$

Following (6.4), it holds that $\theta'(x_k; d_k) \leq -\bar{\sigma}\theta(x_k)$ and we deduce

$$-\bar{\sigma}\theta(x_k) + \phi(\beta^{l_j}\|d_k\|)\|d_k\| \geq -\sigma\theta(x_k).$$

Because of $\phi(\beta^{l_j}\|d_k\|)\|d_k\| \xrightarrow[l_j\to\infty]{} 0$, we get

$$0 \geq (\bar{\sigma} - \sigma)\theta(x_k) \geq 0.$$

As $(\bar{\sigma} - \sigma) > 0$, it follows that $\theta(x_k) = 0$, hence a contradiction. □

After having shown that the monotone step size rule is well-defined, we are again able to state the convergence of the function values.

Theorem 6.2.5. *Let θ be bounded below by 0. Furthermore assume that $(x_k)_{k\in\mathbb{N}}, (d_k)_{k\in\mathbb{N}}$ are generated by the hybrid generalized Newton's method (Algorithm 6.3) and satisfy the monotone step size rule (Algorithm 6.4) where necessary. Then it holds*

$$\theta(x_k) \xrightarrow[k\to\infty]{} 0.$$

Proof. The proof proceeds analogously to the proof of Theorem 6.2.3 with slight adaptations. Because of Lemma 6.2.4 for all k it holds that

$$\theta(x_{k+1}) - \theta(x_k) \leq -\sigma\alpha_k\theta(x_k).$$

If we set $\lambda_k = 0,\ \rho(x_k) = \theta(x_k)$ with $\theta(x_k) \geq 0$ and $\nu_k = 0$ for all k condition (6.1) from Theorem 6.1.2 is satisfied. Furthermore θ is bounded below and (6.2) and (6.3) are as well satisfied. Therefore we can apply Theorem 6.1.2 and we conclude that

$$(i) \quad \text{there exists } \theta_* \geq 0 \text{ such that } \theta(x_k) \xrightarrow[k\to\infty]{} \theta_*, \tag{6.10}$$

$$(ii) \quad \sum_{k=1}^{\infty} \alpha_k\theta(x_k) < \infty, \tag{6.11}$$

which also ensures that

$$\alpha_k\theta(x_k) \xrightarrow[k\to\infty]{} 0. \tag{6.12}$$

(a) If $\lim_{k\to\infty} \alpha_k = 0$, we can state due to $\alpha_k = \beta^{l_k}$ and $\alpha_k > 0$ for all k that

$$\theta(x_k + \frac{\alpha_k}{\beta}\|d_k\|) - \theta(x_k) > -\sigma\frac{\alpha_k}{\beta}\theta(x_k).$$

Using (6.5) and $\phi(u) \xrightarrow[u\to 0]{} 0$ we get

$$\begin{aligned}\frac{\alpha_k}{\beta}\theta'(x_k; d_k) + \phi(\frac{\alpha_k}{\beta}d_k)\frac{\alpha_k}{\beta}\|d_k\| &\geq \theta(x_k + \frac{\alpha_k}{\beta}d_k) - \theta(x_k) \\ &> -\sigma\frac{\alpha_k}{\beta}\theta(x_k),\end{aligned}$$

which is equivalent to

$$\theta'(x_k; d_k) + \phi(\frac{\alpha_k}{\beta}\|d_k\|)\|d_k\| > -\sigma\theta(x_k).$$

Applying (6.4) leads to

$$-\bar{\sigma}\theta(x_k) + \phi(\frac{\alpha_k}{\beta}\|d_k\|)\|d_k\| > -\sigma\theta(x_k).$$

Therefore, we obtain

$$\phi(\frac{\alpha_k}{\beta}\|d_k\|)\|d_k\| > (\bar{\sigma} - \sigma)\theta(x_k).$$

Following (A.3), $(d_k)_{k\in\mathbb{N}}$ is bounded and by assumption on α_k we get

$$\frac{\alpha_k}{\beta}d_k \xrightarrow[k\to\infty]{} 0 \ \Rightarrow\ \phi(\frac{\alpha_k}{\beta}d_k) \xrightarrow[k\to\infty]{} 0 \ \Rightarrow\ \phi(\frac{\alpha_k}{\beta}d_k)\|d_k\| \xrightarrow[k\to\infty]{} 0.$$

For $k \to \infty$ we obtain

$$0 \geq (\bar{\sigma} - \sigma)\theta_* \geq 0,$$

and as $(\bar{\sigma} - \sigma) > 0$ it follows that $\theta_* = 0$ so we get

$$\theta(x_k) \xrightarrow[k\to\infty]{} 0.$$

(b) Otherwise, there exists a subsequence $(\alpha_{k_j})_{j\in\mathbb{N}}, \delta > 0$ such that $\alpha_{k_j} > \delta$ for all $j \in \mathbb{N}$. Using (6.12) we get

$$\alpha_{k_j}\theta(x_{k_j}) \xrightarrow[j\to\infty]{} 0. \tag{6.13}$$

As $\alpha_{k_j} > \delta$ for all $j \in \mathbb{N}$, we deduce that $\theta(x_{k_j}) \xrightarrow[j\to\infty]{} 0$ so the subsequence $(\theta(x_{k_j}))_{j\in\mathbb{N}}$ converges to zero. Following (6.10) we know that $(\theta(x_k))_{k\in\mathbb{N}}$ converges to θ_*. As the limits have to be equal, we deduce $\theta(x_k) \xrightarrow[k\to\infty]{} 0$.

□

If we further assume that there exists x_* such that $x_k \xrightarrow[k\to\infty]{} x_*$, we can give a statement concerning the convergence rate. The following lemma is also needed.

Lemma 6.2.6. *Suppose that $f : \mathbb{R}^n \to \mathbb{R}^n$ is semismooth at the solution x_* of $f(x) = 0$ and that all $V \in \partial_B f(x_*)$ are nonsingular. Then there exists $\delta, C > 0$ and $\epsilon : \mathbb{R}_+ \to \mathbb{R}_+$ with $\epsilon(t) \xrightarrow[t\to 0+]{} 0$ such that $\|V^{-1}\| \leq C$ for all $V \in \partial_B f(x),\ x \in B_\delta(x_*)$ and*

$$(i) \quad \|x - V^{-1}f(x) - x_*\| \leq \epsilon(\|x - x_*\|)\|x - x_*\|,$$
$$(ii) \quad \|f(x - V^{-1}f(x)\| \leq \epsilon(\|x - x_*\|)\|f(x)\|,$$

for all $V \in \partial_B f(x),\ x \in B_\delta(x_*)$.

Proof. Refer to Qi (1993). □

Theorem 6.2.7. *Assume that there exists x_* such that $x_k \xrightarrow[k\to\infty]{} x_*$ and f is semismooth as well as strongly BD-regular in x_*. Then $x_k \xrightarrow[k\to\infty]{} x_*$ superlinearly.*

Proof. As $x_k \xrightarrow[k\to\infty]{} x_*$ the iterates enter into the region of attraction of Theorem 5.4.3. Following Lemma 6.2.6, there exists an index $\bar{k}$ such that

$$\|f(x_{\bar{k}} - V^{-1}f(x_{\bar{k}})\| \leq \epsilon(\|x_{\bar{k}} - x_*\|)\|f(x_{\bar{k}})\|.$$

Therefore, there also exists an index k_δ such that

$$\theta(x_{k_\delta} - V^{-1}f(x_{k_\delta})) \leq \epsilon^2(\|x_{k_\delta} - x_*\|)\theta(x_{k_\delta}) \leq (1 - \sigma)\theta(x_{k_\delta}).$$

This means that for all $k \geq k_\delta$ the watchdog step is satisfied and a full step is performed, which allows to apply Theorem 5.4.3 to get superlinear convergence. □

Unfortunately, a refinement with a nonmonotone step size rule is not possible. Due to the added ν_k we cannot guarantee that a full step $x_{k+1} = x_k + d_k$ with d_k satisfying $V_k d_k = -f(x_k),\ V_k \in \partial_B f(x_k)$ is performed.

6.3 Nonmonotone Step Size Rule by Zhang and Hager

Zhang and Hager (2004) introduce a special nonmonotone step size rule which makes use of a convex combination of the function values of the former iterates.

Algorithm 6.5 Nonmonotone step size rule by Zhang and Hager

Input: $\beta \in (0,1),\ \sigma \in (0, \bar{\sigma}),\ x_k, d_k \in \mathbb{R}^n,\ c_k \in \mathbb{R}_+ \cup \{0\},\ \alpha_{max} > 0,\ \theta(x_k) \neq 0$
$c_0 = \theta(x_0),\ q_0 = 1,\ 0 \leq \eta_{min} \leq \eta_{max}$
Ensure: d_k is chosen such that (6.4) holds and $\eta_k \in [\eta_{min}, \eta_{max}]$
if $\theta(x_k + \alpha_{max} d_k) \leq c_k - \sigma \alpha_{max} \theta(x_k)$ **then**
$\alpha_k = \alpha_{max}$
else
determine smallest $l_k \in \mathbb{N}$ such that
$\theta(x_k + \alpha_{max} \beta^{l_k} d_k) \leq c_k - \sigma \alpha_{max} \beta^{l_k} \theta(x_k)$
$\alpha_k = \alpha_{max} \beta^{l_k}$
end if
$q_{k+1} = \eta_k q_k + 1$
$c_{k+1} = \frac{\eta_k q_k c_k + \theta(x_{k+1})}{q_{k+1}}$
return step size α_k

If we set

$$\nu_k = c_k - \theta(x_k) \quad \forall k,$$

we are in the setting of our nonmonotone Armijo's rule in Algorithm 6.2 . We are now able to use the findings of the former section to prove convergence of the globalized generalized Newton's method if using the nonmonotone step size rule by Zhang and Hager (2004) given above.

Theorem 6.3.1. *Let the assumptions of Theorem 6.2.3 hold. Furthermore we assume that* $\eta_{max} < 1$. *Then it holds*

$$\theta(x_k) \xrightarrow[k \to \infty]{} 0.$$

Proof. In order to apply Theorem 6.2.3 we have to show that

$$(i) \quad \nu_k = c_k - \theta(x_k) \geq 0,$$
$$(ii) \quad \sum_{k=1}^{\infty} \nu_k < \infty.$$

As $-\sigma\alpha_{max}\beta^{l_k}\theta(x_k) \leq 0$ for all k, we deduce that

$$\theta(x_{k+1}) - c_k \leq 0. \tag{6.14}$$

Furthermore, as

$$q_{k+1} = \eta_k q_k + 1 \Leftrightarrow \eta_k = \frac{q_{k+1} - 1}{q_k}$$

we get

$$c_{k+1} = \frac{\eta_k q_k c_k + \theta(x_{k+1})}{q_{k+1}} = \frac{(q_{k+1} - 1)c_k + \theta(x_{k+1})}{q_{k+1}} = c_k + \frac{\theta(x_{k+1}) - c_k}{q_{k+1}}. \tag{6.15}$$

Using (6.14) we derive that

$$c_k - c_{k+1} = c_k - c_k - \frac{\theta(x_{k+1}) - c_k}{q_{k+1}} = \frac{c_k - \theta(x_{k+1})}{q_{k+1}} \geq 0,$$

which also shows that $(c_k)_{k\in\mathbb{N}}$ is monotonically decreasing. This leads to

$$0 \leq \sum_{k=0}^{j} \frac{c_k - \theta(x_{k+1})}{q_{k+1}} = \sum_{k=0}^{j} c_k - c_{k+1} = c_0 - c_{j+1} \leq c_0 - \theta(x_{j+2}) \leq c_0. \tag{6.16}$$

Due to (6.15) we deduce that

$$\nu_{k+1} = c_{k+1} - \theta(x_{k+1}) = (1 - \frac{1}{q_{k+1}})(c_k - \theta(x_{k+1})) = (q_{k+1} - 1)\frac{c_k - \theta(x_{k+1})}{q_{k+1}} \geq 0, \tag{6.17}$$

which shows (i).

As $\eta_k \leq \eta_{max} < 1$, the sequence $(q_k)_{k\in\mathbb{N}}$ is bounded and

$$\begin{aligned} q_{k+1} &= \eta_k q_k + 1 \leq \eta_{max} q_k + 1 \leq \eta_{max}(\eta_{max} q_{k-1} + 1) + 1 \\ &\leq \ldots \leq \sum_{j=0}^{k+1} \eta_{max}^j \leq \sum_{j=0}^{\infty} \eta_{max}^j = \frac{1}{1 - \eta_{max}}. \end{aligned} \tag{6.18}$$

Due to (6.16),(6.17) and (6.18), this yields

$$0 \leq \sum_{k=1}^{\infty} \nu_k = \sum_{k=1}^{\infty} (1 - \frac{1}{q_k})(c_{k-1} - \theta(x_k)) = \sum_{k=1}^{\infty} (q_k - 1) \frac{c_{k-1} - \theta(x_k)}{q_k}$$
$$\leq \frac{\eta_{max}}{1 - \eta_{max}} \sum_{k=1}^{\infty} \frac{c_{k-1} - \theta(x_k)}{q_k} \leq \frac{\eta_{max}}{1 - \eta_{max}} c_0 < \infty,$$

which shows (ii). □

Since the requirements make it very hard to find an appropriate and easy to compute example, we will not conclude with a numerical study. Nevertheless, Chapter 6 shows that techniques from smooth optimization can under certain additional assumptions and requirements be extended to nonsmooth optimization. In practice, smooth methods often deliver good results, even when applied to nonsmooth optimization problems.

Chapter 7

Generalized Calibration for Coherent Small Area Estimation

Survey weighting is a mess.
— Andrew Gelman
Struggles with Survey Weighting and Regression Modeling

Survey weighting is a mess. This controversial opening line in Gelman (2007b) led to an extensive discussion about survey weighting. But what is behind all this? In some of the classical statistical theory of sampling, survey weights are equal to the reciprocal of the inclusion probabilities. In fact, they are typically constructed based on a combination of probability calculation and nonresponse adjustment. Calibration or regression modeling tries to overcome this misery by adding information and therefore adjusting the weights. In fact, this does not overcome all problems and criticism but makes regression modeling a mess with which Gelman (2007a) is comfortable.

We extend the standard calibration approach, which is a special type of regression modeling, by requesting the weights to fulfill certain additional conditions and make the weights 'less messy'.

7.1 Extending Classical Calibration

Consider a finite population $U = \{1, 2, \ldots, N\}$ whose elements are denoted by integers k. Assume the inclusion probabilities $\pi_k > 0$ are known and the design weights are given by $d_k = \pi_k^{-1}$ for each k. Further a finite sample s with cardinality n is drawn from the population U. We want to estimate the total $t_y = \sum_{k \in U} y_k$ of the study variable y by using the Horvitz-Thompson estimator $\hat{t}_y^{HT} = \sum_{k \in s} w_k y_k$ with calibrated weights $w_k = d_k g_k$. For calibrating those weights, additional auxiliary information forming the vector $x_k = (x_{k1}, \ldots, x_{kp})^T$ is available for each element k. Here we have to distinguish whether the auxiliary information is available for every $k \in U$ or only for every $k \in s$. In the first case, which is the case of the adjusted index data in the German Census Sampling and Estimation Research Project, we have complete auxiliary information and can calculate the total exactly. We could even construct a new variable x_k^2 and calculate the total $\sum_{k \in U} x_k^2$, where x_k squared stands for a component wise multiplication. In the second case, which

is in our case the ISCED (international standard classification of education) data in the German Census Sampling and Estimation Research Project, the total forming the benchmark, has to be imported from an outside source, for example from sample census or small area estimations.

The resulting calibration problem forms as follows: find weights $w_k = d_k g_k$ for all $k \in s$ such that a distance function, e.g., the 'chi-square distance'

$$f : \mathbb{R}^n \to \mathbb{R}_+, \; g \mapsto \sum_{k \in s} d_k \frac{(g_k - 1)^2}{2},$$

is minimized with additional calibration benchmarks

$$\sum_{k \in s} x_{ki} d_k g_k = t_{x_i} \quad \forall i = 1, \ldots, p,$$

where $t_{x_i} = \sum_{k \in U} x_{ki}$ in the case of complete auxiliary information or t_i given in the second case. Furthermore, range restrictions

$$m_k \leq g_k \leq M_k \quad \forall k \in s,$$

are given, where $0 \leq m_k < 1 < M_k$ for all $k \in s$, so no fixed values exist. For a detailed discussion on standard calibration problems and its solution methods we refer to Chapter 5. Although these calibration methods and its modifications are used very often in practice, they are lacking some important aspects.

(i) A regulation of the spread of the calibrated weights w can only be done by the range restriction. However, this does not take the ratio of the largest to the smallest calibrated weight into account. Following Little et al. (2009), this ratio should not exceed 10 and is unacceptable beyond 100.

(ii) Further, if we regard the German Census Sampling and Estimation Research Project, there often exist many estimates on different levels. These estimates are gained by different estimators which leads to coherence problems, so we have to allow the benchmarks to be fulfilled with a slight perturbation.

(iii) The methods using penalization do not allow to analyze which calibration benchmarks are restrictive and might be relaxed in order to get overall better estimates.

These aspects were the main reason for developing the classical calibration approach into a multicriterial calibration approach. In case of the German Census 2011, the original design weights' ratio of the largest to the smallest calibrated weight is 25. This is gained by limiting the sampling fraction in the different SMPs between 2% and 50%. We call the maximally permitted spread ratio **Gelman-bound** and because of consistence, the ratio of the calibrated weights should also be limited by a Gelman-bound near to 25.

The problem of missing coherence is also often encountered at table estimates in official statistics, which are based on different estimation methods. This problem may also occur in census that use sample information. Estimates on a lower level are gained by small area estimation methods (cf. Rao, 2003, or the estimator presented in You and Rao, 2002), whereas

estimates on higher levels are gained by classical estimators such as the general regression estimator (cf. Definition 2.2.2). As the sum of estimates on areas on a lower level usually differs from the estimated value of the corresponding higher level, it is not possible to get weights such that all calibration benchmarks are satisfied exactly. Therefore, we have to allow the benchmarks to be relaxed.

In literature, Chambers (1996) mentions ridge weighting to incorporate the relaxation of the benchmarking constraints to get positive weights in the context of robust weighting for multipurpose established surveys. Ridge weighting adds the benchmarking constraints as penalty term in the objective function and delivers a closed form solution for the weights depending on the penalty parameter. However, these weights are not always positive or range restricted so the optimal penalty parameter has to be determined such that the range restriction or box constraint is satisfied. This is often done by examining different plots. Further, costs of the weighted estimator not satisfying the calibration constraints have to be defined. For a detailed discussion of penalty methods we refer to, e.g., Gill et al. (1981) or Nocedal and Wright (2006).

Rao and Singh (1997) propose a method that relaxes some benchmarking constraints while satisfying range restrictions and maintaining design consistency. In fact, it is an iterative ridge regression method with projection of the weights into the given range restriction and updating the maximum tolerance of the perturbation of the benchmarking constraints. The existence of solutions to ridge regression methods with range restrictions is discussed in Théberge (2000), where he makes use of an adjustment of the penalty parameter. Chen et al. (2002) make use of a relaxation of the benchmarking constraints in order to make the calibrated weights fulfill the range restriction, where they concentrate on obtaining weights in pseudo-empirical likelihood and model-calibrated empirical likelihood estimators. Beaumont and Bocci (2008) discuss ridge calibration and the method of Chen et al. (2002) and propose an alternative method which boils down to be equivalent to ridge calibration and delivers satisfying results when staying close to the given benchmarks is desired. Multiple and ridge model calibration is studied in Montanari and Ranalli (2009) and allows to obtain a single set of weights for more than one survey variable and estimates that are coherent with census data or aligned with those produced by another survey. In Rao and Singh (2009) an iterative method (ridge shrinkage) is proposed to generalize the ridge regression method in a manner similar to the iterative modifications of generalized regression, to meet range restrictions. It forces convergence for a specified number of iterations by using a build-in tolerance specific procedure to relax benchmarking constraints while satisfying range restrictions and maintaining design consistency.

Nevertheless, all these methods require to determine user-specified costs associated with not satisfying the benchmarks in advance. The method we will present below is based on standard optimization procedures, for which very efficient algorithms and software exists, and makes it also possible to get information about the restricting effect of the calibration benchmarks by regarding the Lagrange multiplier. Further, apart from adding the Gelman-bound constraint, the method delivers a single set of weights for estimating other variables of interest leading to a one number census. Early discussions and applications of this method were recently given by Münnich, Sachs and Wagner (2012a).

7.2 Mathematical Formulation of the Census Problem

Consider the finite population $U = \{1, 2, \ldots, N\}$ of addresses in Germany whose elements are denoted by integers k. As mentioned in Chapter 3, this population is divided into K disjunct subsets K_j, the so called 'Kreise' or districts, i.e.,

$$K_j \text{ with } |K_j| = k_j,\ K_j \subset U,\ \bigcup_{j=1}^{K} K_j = U,\ K_j \cap K_k = \emptyset\ \forall j \neq k,$$

which are again divided into the so called 'smallest sampling points' SMP_{jl}, i.e.,

$$SMP_{jl} \text{ with } |SMP_{jl}| = p_{jl},\ \sum_{l=1}^{G_j} p_{jl} = k_j,$$

where G_j denotes the number of SMPs forming a Kreis/district K_j and

$$SMP_{jl} \subset K_j,\ \bigcup_{l=1}^{G_j} SMP_{jl} = K_j,\ SMP_{jl} \cap SMP_{ji} = \emptyset\ \forall l \neq i.$$

Now, a sample s with cardinality n is drawn consisting of partial samples s_{jl} of the SMPs mentioned above, such that

$$\sum_{j=1,\ldots,K} \sum_{l=1,\ldots,G_j} |s_{jl}| = n,\ s_{jl} \subset SMP_{jl},\ \bigcup_{j=1,\ldots,K,\ l=1,\ldots,G_j} s_{jl} = s.$$

The way the partial samples are drawn is discussed in Chapter 3 and Münnich, Sachs and Wagner (2012c). To illustrate the situation mentioned above, we have a look at Figure 7.1, where the partitioning of the population into three districts/Kreise and four SMPs in district 1 with corresponding partial samples is shown. For the sake of simplicity we will only regard the elements sampled, so we assume that

$$s = \{s_1, \ldots, s_n\} = \bigcup_{j=1,\ldots,K,\ l=1,\ldots,G_j} s_{jl}.$$

This leads to a simple vector whose elements are arranged as shown in figure 7.2. Further, assume that the inclusion probabilities $\pi_k > 0$ are known and the design weights are given by $d_k = \pi_k^{-1}$ for each k. We want to estimate the total $t_y = \sum_{k \in U} y_k$ of the study variable y by using the calibrated Horvitz-Thompson estimator

$$\hat{t}_y = \sum_{k \in s} w_k y_k$$

with calibrated weights $w_k = d_k g_k$. For calibrating those weights, additional auxiliary information are available for each element k. On the one hand, benchmarks for districts and SMPs concerning known register data, on the other hand benchmarks for districts and SMPs

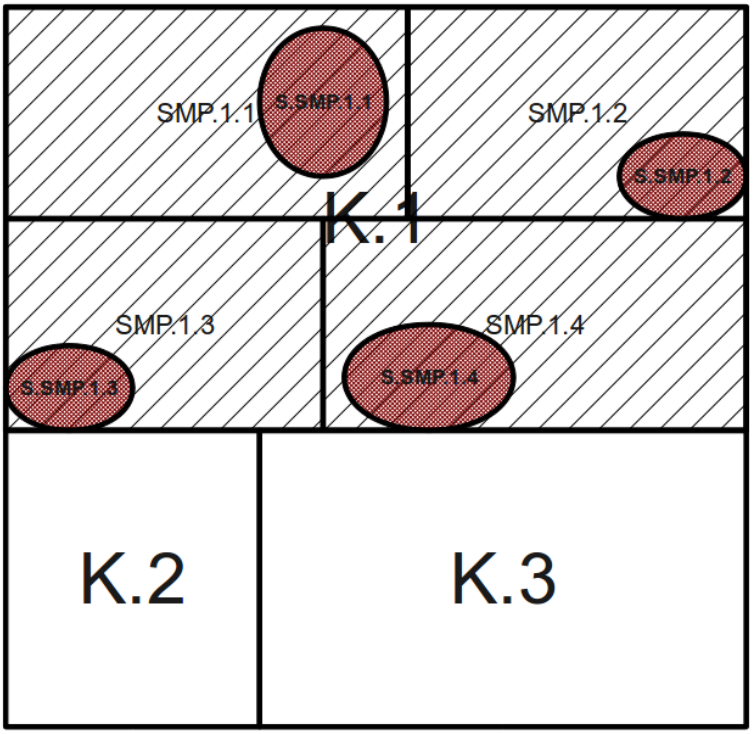


Figure 7.1: Partitioning of the population

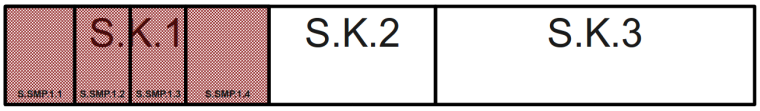


Figure 7.2: Partitioning of the sample

concerning another attribute, e.g., employment or educational background. The benchmarks concerning known register data on SMP and district level are estimated by a combined GREG estimator (cf. Definition 2.2.2 or Münnich, Gabler, Ganninger, Burgard and Kolb, 2012). As this estimator is vertically coherent, we only have to calibrate on SMP level. If the benchmarks are fulfilled on SMP level they are also fulfilled on district level. Concerning the other benchmarks, different estimators are applied on different levels. On district level, a combined GREG estimator is applied, whereas on SMP level the YOURAO estimator from Definition 2.4.4 is taken.

We are now able to build design matrices $X^T \in \mathbb{R}^{1\times n}$ consisting of known register data and $\bar{X}^T \in \mathbb{R}^{r\times n}$ consisting of other attributes mentioned above. Note that for the sake of readability the notation slightly differs from the one used in Chapter 5.

$$X^T := (\xi_1, \ldots, \xi_n) = (x_1 d_1, \ldots, x_n d_n) \in \mathbb{R}^{1\times n},$$

$$\bar{X}^T := \begin{pmatrix} | & & | \\ \bar{\xi}_1 & \cdots & \bar{\xi}_n \\ | & & | \end{pmatrix} = \begin{pmatrix} \bar{\xi}_{11} & \cdots & \bar{\xi}_{n1} \\ \vdots & & \vdots \\ \bar{\xi}_{1r} & \cdots & \bar{\xi}_{nr} \end{pmatrix} = \begin{pmatrix} \bar{x}_{11}d_1 & \cdots & \bar{x}_{n1}d_n \\ \vdots & & \vdots \\ \bar{x}_{1r}d_1 & \cdots & \bar{x}_{nr}d_n \end{pmatrix} \in \mathbb{R}^{r\times n}.$$

Further, we define

$$\bar{X}_j^T := (\bar{\zeta}_{qr})_{qr}, \quad \bar{\zeta}_{qr} = \begin{cases} \bar{\xi}_{qr}, & \text{if } s_q \in K_j, \\ 0, & \text{else,} \end{cases}$$

representing the 'partial matrices' for the districts j, $(j = 1, \ldots, K)$. Regarding an example with two different calibration variables and two districts, this leads to

$$\bar{X}^T = \left(\begin{array}{cccc||ccc} * & * & * & * & * & * & * \\ * & * & * & * & * & * & * \end{array}\right),$$
$$\bar{X}_1^T = \left(\begin{array}{cccc||ccc} * & * & * & * & 0 & 0 & 0 \\ * & * & * & * & 0 & 0 & 0 \end{array}\right), \quad \bar{X}_2^T = \left(\begin{array}{cccc||ccc} 0 & 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & 0 & * & * & * \end{array}\right).$$

Defining

$$\bar{X}_{jl}^T := (\bar{\zeta}_{qr})_{qr}, \quad \bar{\zeta}_{qr} = \begin{cases} \bar{\xi}_{qr}, & \text{if } s_q \in S_{jl}, \\ 0, & \text{else,} \end{cases} \quad (j = 1, \ldots, K,\ l = 1, \ldots, G_j),$$

leads to the 'partial matrices' for the SMPs and the matrices of the example form as follows:

$$\bar{X}^T = \left(\begin{array}{cc|cc||c|cc} * & * & * & * & * & * & * \\ * & * & * & * & * & * & * \end{array}\right),$$
$$\bar{X}_{11}^T = \left(\begin{array}{cc|cc||c|cc} * & * & 0 & 0 & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 & 0 & 0 \end{array}\right), \quad \bar{X}_{12}^T = \left(\begin{array}{cc|cc||c|cc} 0 & 0 & * & * & 0 & 0 & 0 \\ 0 & 0 & * & * & 0 & 0 & 0 \end{array}\right),$$
$$\bar{X}_{21}^T = \left(\begin{array}{cc|cc||c|cc} 0 & 0 & 0 & 0 & * & 0 & 0 \\ 0 & 0 & 0 & 0 & * & 0 & 0 \end{array}\right), \quad \bar{X}_{22}^T = \left(\begin{array}{cc|cc||c|cc} 0 & 0 & 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & 0 & * & * \end{array}\right).$$

The given benchmarks for the calibration variables $\bar{x}$ on district level K_j are denoted by $t_{\bar{x}_j}$ and $t_{\bar{x}_{jl}}$ on SMP level SMP_{jl}. X_j^T, X_{jl}^T as well as t_{x_j} and $t_{x_{jl}}$ are defined analogously.

Let us now have a look at the spread ratio mentioned in the introduction, which is defined as follows.

Definition 7.2.1 (Spread ratio)**.** *The* **spread ratio** *SR of weights* $w_k = d_k g_k,\ k = 1, \ldots, n$ *is defined as*

$$SR(w) := \frac{\max_{k=1,\ldots,n} w_k}{\min_{k=1,\ldots,n} w_k}.$$

At first glance, a restriction of this ratio by the Gelman-bound GB could be done by adding $SR(w) - GB \leq 0$ as additional constraint. However, in contrast to the calibration benchmarks which are linear constraints, the ratio constraint is nonlinear and non-differentiable. Therefore, solution methods like barrier methods, augmented Lagrangian methods, SQP methods

or interior point methods, which are discussed in detail in Gill et al. (1981) or Nocedal and Wright (2006), are not applicable and would, if applied, lead to numerical difficulties during the minimization process. Another possibility rewrites the nonlinear constraint $SR : \mathbb{R}^n \to \mathbb{R}$ as a set of $2n + 1$ linear constraints.

Lemma 7.2.2. *The nonlinear inequality $SR(w) - GB \leq 0$ can be rewritten as the set of linear inequalities*

$$\begin{aligned} -w_k + \alpha &\leq 0 \quad \forall k = 1, \ldots, n, \\ w_k - \beta &\leq 0 \quad \forall k = 1, \ldots, n, \\ -GB\alpha + \beta &\leq 0. \end{aligned}$$

Proof. It holds

$$SR(w) - GB \leq 0 \Leftrightarrow \frac{\max_{k=1,\ldots,n} w_k}{\min_{k=1,\ldots,n} w_k} - GB \leq 0.$$

Let $\alpha, \beta \in \mathbb{R}_+$ such that

$$\begin{aligned} -w_k + \alpha &\leq 0 \quad \forall\ k = 1, \ldots, n, \\ w_k - \beta &\leq 0 \quad \forall\ k = 1, \ldots, n. \end{aligned}$$

Then it holds that $\alpha \leq \min_{k=1,\ldots,n} w_k$ and $\beta \geq \max_{k=1,\ldots,n} w_k$ as well as $\frac{\max_{k=1,\ldots,n} w_k}{\min_{k=1,\ldots,n} w_k} \leq \frac{\beta}{\alpha}$. Further

$$-GB\alpha + \beta \leq 0 \Leftrightarrow \frac{\beta}{\alpha} - GB \leq 0 \Rightarrow \frac{\max_{k=1,\ldots,n} w_k}{\min_{k=1,\ldots,n} w_k} - GB \leq 0,$$

which completes the proof. □

After having introduced the necessary notations, we are able to formulate the calibration problem. We want to gain calibration factors g_k near to 1 such that the Horvitz-Thompson estimator forms as follows.

$$\hat{t}_y^{HT} = \sum_{k \in s} w_k y_k = \sum_{k \in s} g_k d_k y_k.$$

Furthermore, there exists an objective function $f : \mathbb{R}^n \to \mathbb{R}$ with $f(g) = \sum_{k \in s} d_k \frac{(g_k - 1)^2}{2}$ so we can formulate the following calibration problem.

$$\begin{aligned} \min_{g \in \mathbb{R}^n} \ & \sum_{k \in s} d_k \frac{(g_k - 1)^2}{2} \\ s.t.\ & X_{jl}^T g - t_{x_{jl}} = 0 \quad \forall j = 1, \ldots, K,\ l = 1, \ldots, G_j \ (SMPs) \\ & \bar{X}_j^T g - t_{\bar{x}_j} = 0 \quad \forall j = 1, \ldots, K \ (districts) \\ & \bar{X}_{jl}^T g - t_{\bar{x}_{jl}} = 0 \quad \forall j = 1, \ldots, K,\ l = 1, \ldots, G_j \ (SMPs) \end{aligned} \tag{7.1}$$

In case of different estimators on district and SMP level this optimization problem is in general infeasible. As already mentioned, the German Census Sampling and Estimation Research Project recommends to use a combined GREG estimator and the YOURAO estimator. This leads to coherence problems, so we have to relax our problem and the benchmark constraints may be perturbed. Nevertheless, the weights and the perturbation shall be bounded, thus an additional box constraint is added. Furthermore, we want to limit the spread ratio $SR(w)$ by the Gelman-bound GB, so we get the following calibration problem.

$$\begin{aligned}
\min_{(g,\epsilon,\alpha,\beta)\in\mathbb{R}^{n+u+2}} \quad & \sum_{k\in s} d_k \frac{(g_k-1)^2}{2} + \sum_k \delta_k \frac{(\epsilon_{dis,k}-1)^2}{2} + \sum_k \delta_k \frac{(\epsilon_{SMP,k}-1)^2}{2} \\
s.t.\ & X_{jl}^T g - t_{x_{jl}} = 0 \quad \forall j = 1,\dots,K,\ l = 1,\dots,G_j \ (SMPs) \\
& \bar{X}_j^T g - \mathrm{diag}(t_{\bar{x}_j})\epsilon_{dis} = 0 \quad \forall j = 1,\dots,K \ (districts) \\
& \bar{X}_{jl}^T g - \mathrm{diag}(t_{\bar{x}_{jl}})\epsilon_{SMP} = 0 \quad \forall j = 1,\dots,K,\ l = 1,\dots,G_j \ (SMPs) \\
& -d_k g_k + \alpha \le 0 \quad \forall k = 1,\dots,n \\
& d_k g_k - \beta \le 0 \quad \forall k = 1,\dots,n \\
& -GB\alpha + \beta \le 0. \\
& (g, \epsilon_{dis}, \epsilon_{SMP}) \in [m, M] \times [m_{\epsilon_{dis}}, M_{\epsilon_{dis}}] \times [m_{\epsilon_{SMP}}, M_{\epsilon_{SMP}}]
\end{aligned} \tag{7.2}$$

If we set $z = (g, \epsilon_{dis}, \epsilon_{SMP}, \alpha, \beta) \in \mathbb{R}^{n+u+2}, q = (d, \delta, 0, 0) \in \mathbb{R}^{n+u+2}$, $Q = \mathrm{diag}(q) \in \mathbb{R}^{(n+u+2)\times(n+u+2)}$, minimizing the objective function of (7.2) is equivalent to

$$\min_{z\in\mathbb{R}^{n+u+2}} \ \frac{1}{2} z^T Q z - q^T z.$$

Furthermore, the calibration constraints can be written as linear constraints

$$Az \leqq t,$$

where $A \in \mathbb{R}^{(p+u+2n+1)\times(n+u+2)}$, $t \in \mathbb{R}^{p+u+2n+1}$ and '$\leqq$' means that the first $p+u$ equations have to be fulfilled with '$=$' and the last $2n+1$ equations have to be fulfilled with '$\le$'.
If we further set $L = (m, m_{\epsilon_{dis}}, m_{\epsilon_{SMP}}, -\infty, -\infty) \in \mathbb{R}^{n+u+2}$ and
$U = (M, M_{\epsilon_{dis}}, M_{\epsilon_{SMP}}, \infty, \infty) \in \mathbb{R}^{n+u+2}$ we get the following quadratic problem.

$$\begin{aligned}
\min_{z\in\mathbb{R}^{n+u+2}} \quad & z^T Q z - q^T z \\
s.t.\ & Az \leqq t \\
& z \in [L, U].
\end{aligned} \tag{7.3}$$

7.3 Computational Aspects

Before dealing with the solution and statistical analysis of the calibration problem, we will have a look at some computational aspects. If we consider the following example with $n = 12,\ r = 2,\ u = 18$, where three districts are each divided into two SMPs, we get

the structure of the matrix A seen in Figure 7.3. The first block (blocks are separated by solid lines) derives from the benchmarks, that have to be fulfilled exactly. The second and the third block derive from the relaxed benchmarks and the last, which is also the largest block, derives from the Gelman-bound constraint. Obviously, the matrix is sparse and only

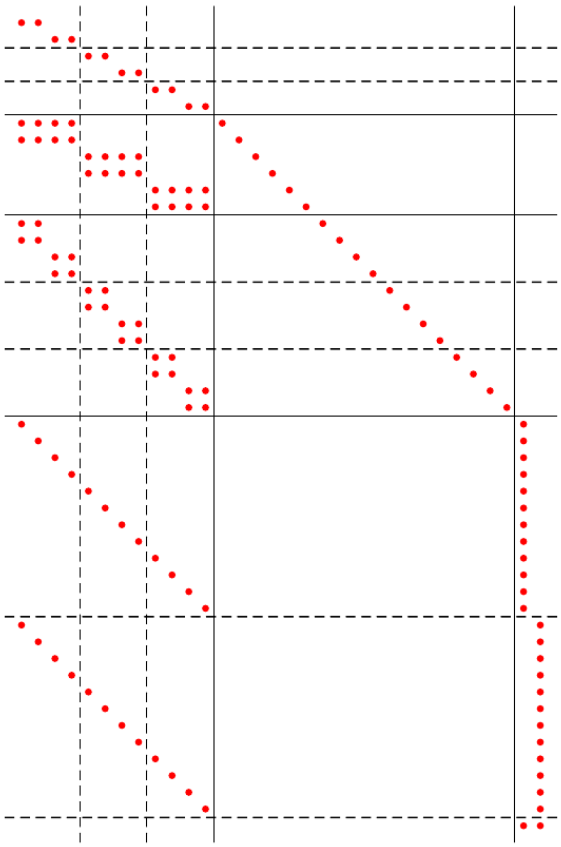


Figure 7.3: Sparse structure of the matrix A

$u + (2r + 5)n + 2 = 128$ of the $(6 + u + 2n + 1) \cdot (n + u + 2) = 1568$ entries are nontrivial (8.1%). In the more realistic case of $n = 1{,}409{,}620$, $r = 3$, $u = 8460$, only $u + (2r + 5)n + 2 = 15{,}514{,}282$ of the $(2391 + u + 2n + 1) \cdot (n + u + 2) \approx 4.01 \cdot 10^{12}$ entries are nontrivial (0.00038%). Therefore it is inevitable to use a solver that supports sparsity.

If necessary for applying a special solver, the inequality constraints can be transformed into equality constraints by adding slack variables. For small dimensions, the resulting calibration problem

$$\begin{aligned} &\min_{z} \ \bar{z}^T \bar{Q} \bar{z} - \bar{q}^T \bar{z} \\ &s.t. \ \bar{A}\bar{z} = t \\ &\quad \bar{z} \in [\bar{L}, \bar{U}] \end{aligned} \tag{7.4}$$

can then be solved by well known algorithms implemented in R packages such as ‘calib’ by

Tillé and Matei (2009) or 'quadprog' by Turlach and Weingessel (2011), whose algorithm is based on the methods of Goldfarb and Idnani (1982) or Goldfarb and Idnani (1983).

However, if dimensions raise we get storage problems because the algorithms do not support sparsity. We decided to use the commercial solver 'IBM ILOG CPLEX Optimization Studio' because it has several advantages. At first, it is a sophisticated solver and supports sparsity, which is crucial for our given census problem. Secondly, there exits an R-package called 'cplexAPI' by Gelius-Dietrich (2012) which passes the sparse notation of the optimization problem to the `C` codes from CPLEX and returns the solution to R. Nevertheless, the sparse notation has to be constructed which, because of a great amount of loops, was done by `C` routines called from R. These routines exploit the special structure of A and therefore are cheap in terms of computing time but expensive in time needed for coding.

7.4 Application to the German Census Sampling and Estimation Research Project

Apart from calibrating the weights in order to get weights for estimating other variables of interest, the method presented previously has another advantage. The German Federal Statistical Office would like to have a single set of weights that, if applied to the variables of interest by using the Horvitz-Thompson estimator, led to the already gained estimates. On the one hand, this so called one number census has the advantage, that the estimates are vertically coherent and on the other hand it also delivers an easy to understand and easy to communicate estimation approach. Further, because of the weighting approach, the estimates can be easily included into tables leading to coherent results.

We did a simulation study with an artificially generated data set representing the German population. In order to get reliable results, 990 samples were drawn and the corresponding estimates were done. The benchmarks, that had to be fulfilled exactly, were the totals of the corrected register data (REG). As they were estimated by the combined GREG estimator, the estimates on SMP and district level were coherent so it was sufficient to take the SMP totals as benchmarks. Another variable for calibration was the ISCED (international standard classification of education) data, where three classes, namely ISCEDA, ISCEDB and ISCEDC were chosen and therefore led to three benchmarks in every domain. Their totals on district level were estimated by the GREG estimator whilst the totals on SMP level were estimated by the YOURAO estimator, so these benchmarks had to be relaxed.

As mentioned in Chapter 3 there exist 2391 SMPs summing up to 429 districts. Every sample has a cardinality of 1,409,620, so the quadratic problem (7.3) optimizes over $z \in \mathbb{R}^{1,418,082}$ and the matrix Q in the objective function is of dimension 1,418,082 $\times$ 1,418,082. The matrix of the constraints is much larger, namely $A \in \mathbb{R}^{2,830,092 \times 1,418,082}$.

After several pretests, we decided to concentrate on four different settings which can be seen in Table 7.1. In setting 1 only the benchmarks of the corrected register data have to be fulfilled. Further, no Gelman-bound is preset and as $\delta = 0$, no penalty is imposed on not fulfilling the ISCED benchmarks. This setting is chosen in order to see, how difficult it will be to fulfill the ISCED benchmarks because we can see, how much the given benchmarks differ

on district and on SMP level. Setting 2 imposes a penalty on the variance of the ISCED benchmarks and the spread of the calibrated weights w is limited by the Gelman-bound 35 which is chosen because the initial weights had a spread of 25. Further, the calibration factors g are restricted by a box constraint whereas the variance of the ISCED benchmarks is not restricted. This setting corresponds to a penalty approach and is chosen to see how the spread changes and how a penalty approach would handle the ISCED benchmarks.

		setting 1		setting 2		setting 3		setting 4	
GB		∞		35		35		35	
δ		0		1000		1000		1000	
m	M	0	∞	0.1	10	0.1	10	0.1	10
$m_{\epsilon_{dis}}$	$M_{\epsilon_{dis}}$	$-\infty$	∞	$-\infty$	∞	0.7	1.3	0.8	1.2
$m_{\epsilon_{SMP}}$	$M_{\epsilon_{SMP}}$	$-\infty$	∞	$-\infty$	∞	0.7	1.3	0.7	1.3

Table 7.1: Test settings

Settings 3 and 4 impose a box constraint on the perturbation of the ISCED benchmarks. Setting 3 allows a maximum perturbation of 30% on district and SMP level, where because of the penalty term in the objective function, the maximum should only be reached by few domains. In setting 4, the perturbation on district level is narrowed to 20%.

It is worth to note that when setting $GB = \infty$, $m_{\epsilon_{dis}} = -\infty$, $M_{\epsilon_{dis}} = \infty$, $m_{\epsilon_{SMP}} = -\infty$, $M_{\epsilon_{SMP}} = \infty$ and differing δ for every benchmark, our approach includes the so called 'ridge calibration' approach and the 'CSW method' mentioned in Beaumont and Bocci (2008), but has the advantage that bounds for the calibrated weights can be easily imposed.

The computations of the simulation were processed on a compute server with 48 CPU cores and an internal memory of 264 GB. Each CPU is an AMD Opteron(TM) with 1.7 GHz. As the samples are independent from each other, we were able to use almost all CPUs at the same time. Reordering the date, building the sparse notation of the matrices and computing a solution took about 6 hours, so the whole simulation took about 4 weeks.

Regarding the resolvability depending on the test settings in Table 7.2, we can state that when the ISCED benchmarks are ignored, which corresponds to setting one, there exists a solution for every sample. If we add the Gelman-bound, a box constraint for g and a penalty parameter, only one sample leads to an unsolvable calibration problem, whereas the remaining 989 samples form a resolvable calibration problem. This indicates that the Gelman-bound constraint seems to play less a role concerning resolvability. We will further have a look at two explicit samples, namely sample 651 which leads to resolvable calibration problems for every setting, and sample 394 whose calibration problems for setting 3 and 4 are unsolvable.

	setting 1	setting 2	setting 3	setting 4
resolvable	990	989	848	755
[%]	100	99.9	85.7	76.3

Table 7.2: Resolvability depending on test settings

Regarding the target and actual values of the solution of sample 651 in Table 7.3, we can state that a calibration concerning only the corrected register data without imposing a Gelman-bound and without a penalty on the ISCED constraints leads to a spread ratio of 63, which is far above the initial value of 25. Fortunately, the calibration factors g lie between 0.13 and 1.56 which is near 1 so 0.1 and 10 as upper and lower bound in the other settings seems to be well-chosen. On district level, the maximum perturbation concerning the ISCED benchmarks is 78% upwards and 22% downwards and on SMP level 330% upwards and 52% downwards. Keep in mind, that because of $\delta = 0$ in setting one, the ISCED benchmarks are not considered during calibration so the strong perturbation is not surprising. Nevertheless, the greater variation on SMP level indicates, that fulfilling the benchmarks on SMP level is not as easy as on district level.

In setting 2, the Gelman-bound $GB = 35$ and the penalty parameter $\delta = 1000$ as well as a box constraint for g are added. This leads to a reduction of the variation to the ISCED benchmarks on district and SMP level, where the biggest reduction can be seen on SMP level to a maximum of 44%. Nevertheless, 44% are far too much for being an acceptable variation and it is also likely that most of the calibration constraints are already fulfilled with less variation. Therefore, an upper and a lower bound of 30% perturbation is added in setting 3. It is also interesting to see, that this limitation does not affect $\min\{\epsilon_{dis}\}$, which is the smallest value of the downward perturbation on district level. As already mentioned, it is quite likely that fulfilling the benchmarks on SMP level is not as easy as on district level. Therefore, in setting 4 the range restriction on district level is limited to 20% which also leads to a solution satisfying all constraints.

	setting 1		setting 2		setting 3		setting 4	
	target	actual	target	actual	target	actual	target	actual
GB	∞	63	35	35	35	35	35	35
m	0	0.13	0.1	0.1	0.1	0.1	0.1	0.1
M	∞	1.56	10	2.49	10	2.49	10	2.49
$m_{\epsilon_{dis}}$	$-\infty$	0.78	$-\infty$	0.81	0.7	0.81	0.8	0.81
$M_{\epsilon_{dis}}$	∞	1.78	∞	1.56	1.3	1.3	1.2	1.2
$m_{\epsilon_{SMP}}$	$-\infty$	0.48	$-\infty$	0.64	0.7	0.7	0.7	0.7
$M_{\epsilon_{SMP}}$	∞	4.30	∞	1.44	1.3	1.3	1.3	1.3

Table 7.3: Target and actual values of the solution of sample 651

However, this good results are not always achieved (cf. Table 7.2) and depend on the sample. Regarding the results of sample 394 given in Table 7.4, we can see that setting 1 leads to a very large spread ratio of 105, which fortunately can be reduced to 35 in setting 2. The range of g and the maximum perturbation of the ISCED variable on district level can be compared to those of sample 651. On a less positive note, the perturbation on SMP level in setting 1 amounts to unpleasant 2270%, which indicates that severe coherence problems of the ISCED variable are rather likely. Further, this indicates that the estimation of the corrected register data variable as well as the estimation of the ISCED variable on SMP level differ widely. Adding a penalty (setting 2) makes it possible to lower the value of $\max\{\epsilon_{SMP}\}$ to 2.34, but 134% perturbation is still far too much. That is the reason why setting 3 and setting 4 are unsolvable. The high perturbation on SMP level can be explained by a suboptimal small area estimation which was already encountered in some pretests. However, it is not the case that certain SMPs or samples with a certain number of sampled elements in the SMPs lead to the suboptimal estimates, where suboptimal means negative estimates or estimates far away from the estimates gained for the other samples. Although we did not generate the estimates and will not deal with this topic in detail (for further information refer to Kolb, 2012), we will mention a possible approach to handle extreme benchmarks or unfavorable SMPs concerning calibration. The easiest possibility is to chose a 'small' δ_k for the corresponding SMP and to vary the bounds allowing a greater perturbation. Then, this benchmark is of inferior importance in the optimization process and does not inference the other benchmarks. However, this violates the desired vertical coherence in the superior district.

	setting 1		setting 2	
	target	actual	target	actual
GB	∞	105	35	35
m	0	0.12	0.1	0.1
M	∞	1.54	10	2.14
$m_{\epsilon_{dis}}$	$-\infty$	0.78	$-\infty$	0.83
$M_{\epsilon_{dis}}$	∞	1.66	∞	1.49
$m_{\epsilon_{SMP}}$	$-\infty$	0.50	$-\infty$	0.67
$M_{\epsilon_{SMP}}$	∞	23.70	∞	2.34

Table 7.4: Target and actual values of the solution of sample 394

We will now revisit sample 651. Regarding the box plots (Figure 7.4) of the variable g depending on the setting, we can see that most calibration factors g_k are near to one. In setting 2, 3 and 4 the spread between the first and the third quartile increases and the whiskers also slightly drift outwards. However, they all stay close to one. The maximum of g increases to 2.5 which is caused by the imposition of a penalty on violating the ISCED benchmarks. As $M = 10$, $\max\{g\}$ could be larger, but it is likely that g is limited by the given Gelman-bound.

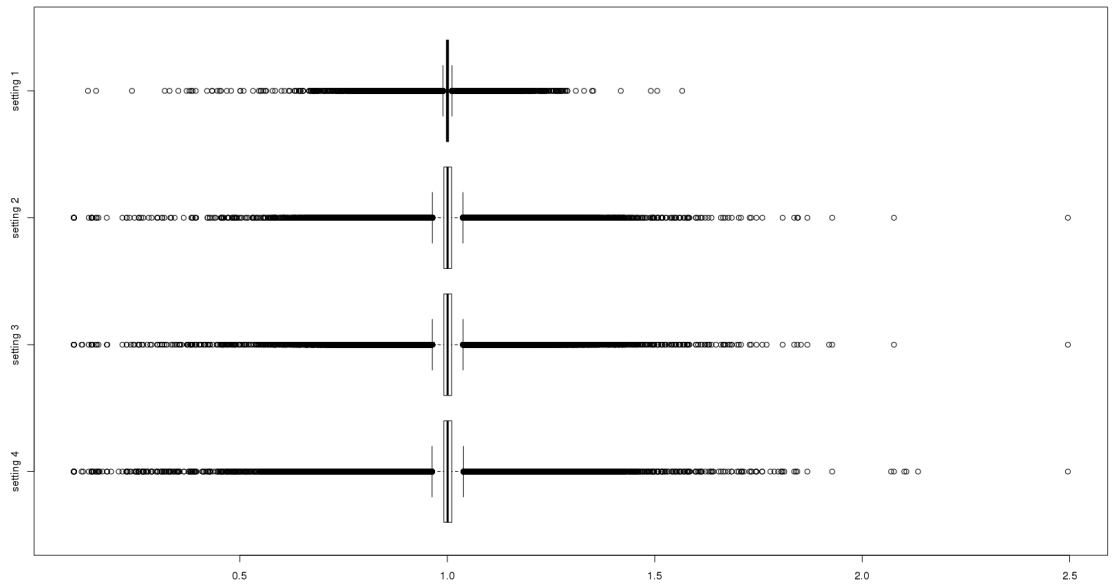


Figure 7.4: Box plots of calibration factors g for sample 651

Figure 7.5 shows the values of ϵ_{dis} for the ISCEDA benchmark for every setting and 7.6 the corresponding histograms. Almost every value of ϵ_{dis} is very close to 1 except for three outliers taking values between 1.25 and 1.30. As setting 3 imposes an upper bound of 1.3, the ISCEDA benchmark on district level does not lead to any problems. Further, a bound of 1.2 is also unproblematic as we can see in setting 4. However, keep in mind that all benchmarks and all levels interact, so a small change on the upper level may cause a large change on the lower level.

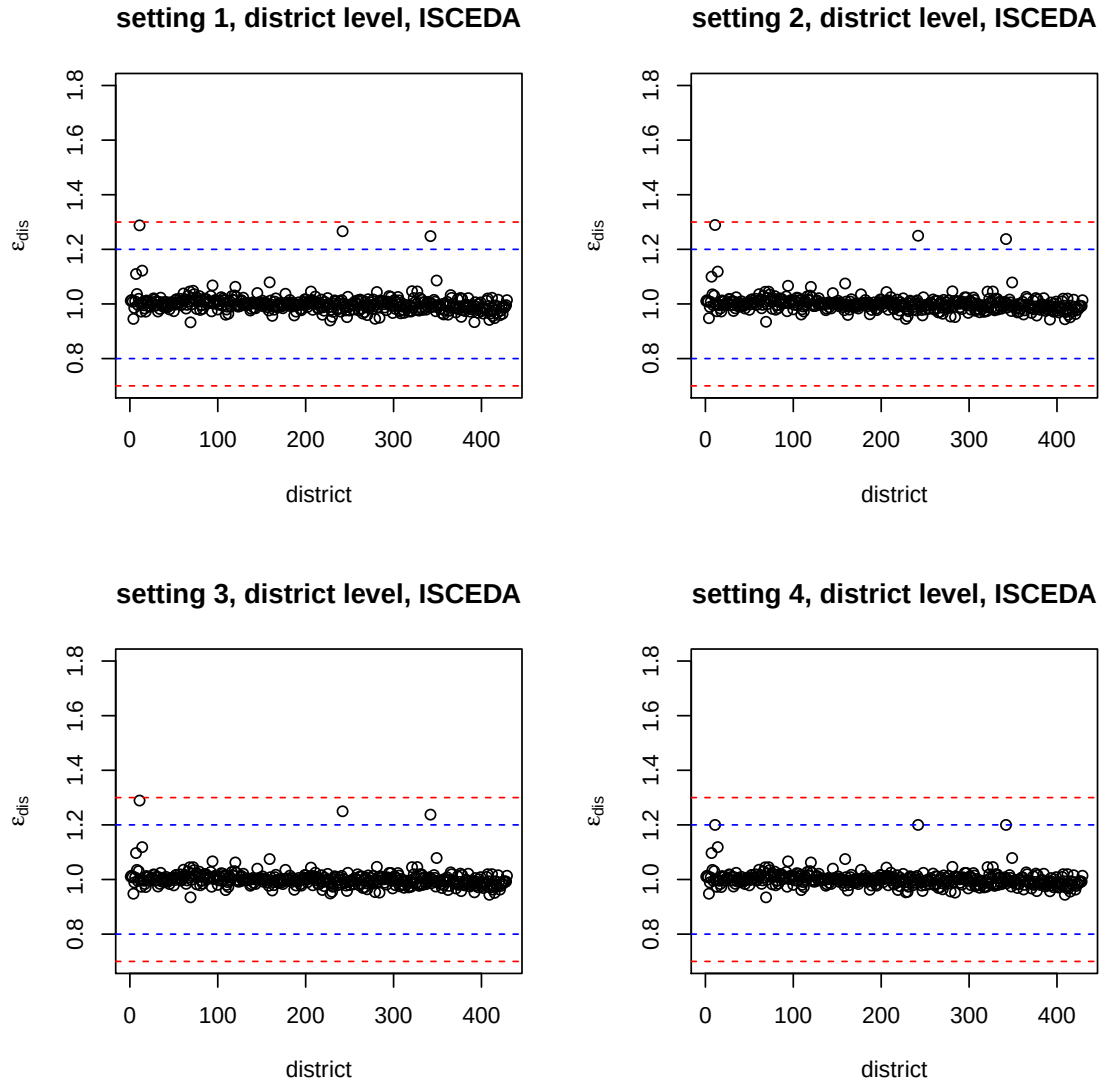


Figure 7.5: Perturbation ϵ_{dis} for benchmark ISCEDA and sample 651

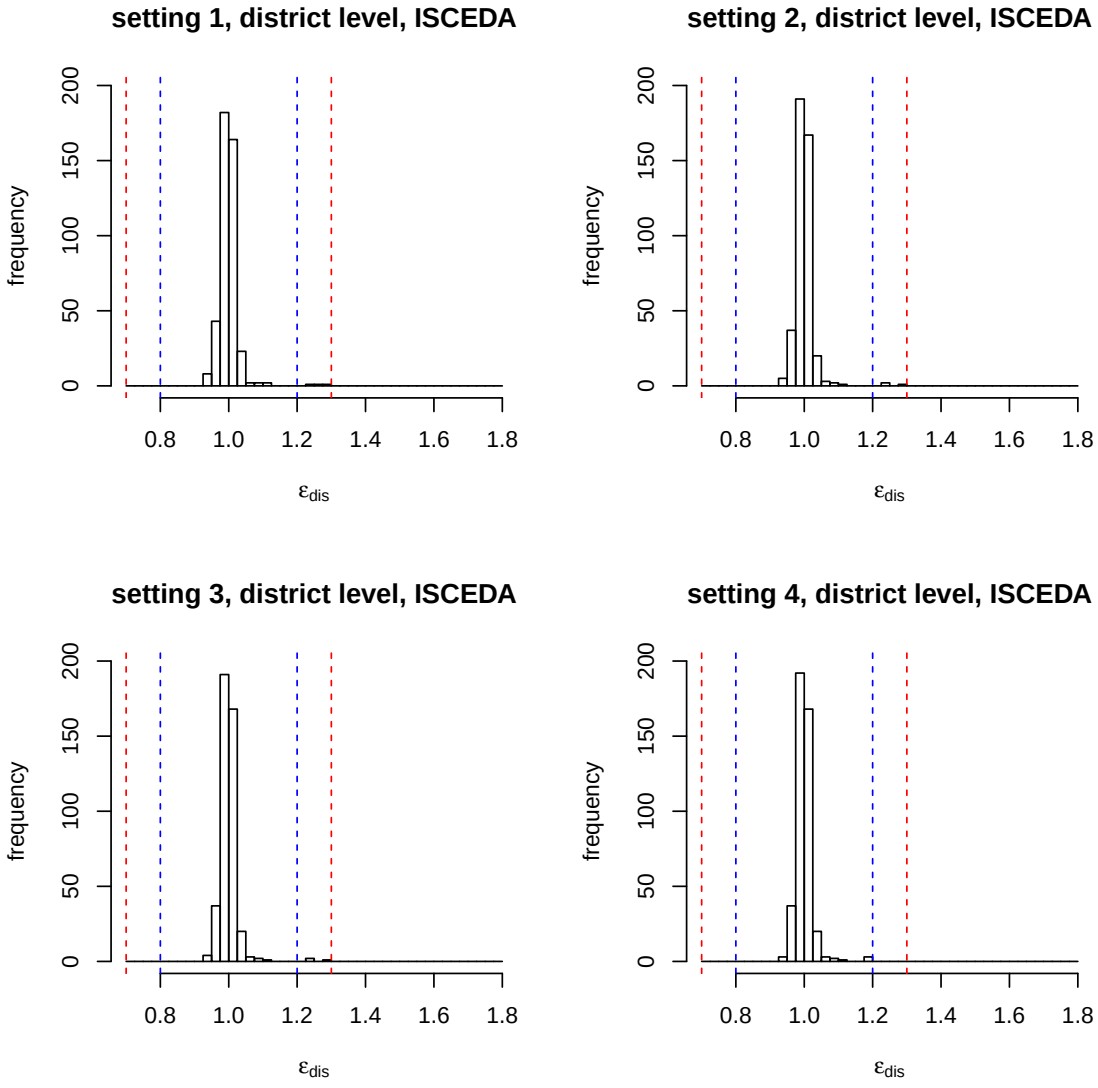


Figure 7.6: Histogram with absolute frequencies of the perturbation ϵ_{dis} for benchmark ISCEDA and sample 651

The ISCEDB variable is more problematic than ISCEDA. We can see that the variation is larger, i.e., not only more districts differ from the benchmark, but also differ with a greater amplitude. In setting one, the maximum variation is approximately 80%, which can be lowered by a penalty to less than 60%. This is very interesting, because it shows that a sole penalty approach, as presented in Beaumont and Bocci (2008) or Chambers (1996), may yield unsatisfying results. Our approach adds a box constraint on the perturbation and at the same time puts a penalty on the relaxation of the benchmark. This has the advantage, that we are able to limit the maximum perturbation. Figure 7.7 clearly shows that in setting 3 and 4 the maximum perturbation is limited by 30% and 20%, respectively, and that the

limitation affected many districts. This is supported by Figure 7.8 where we can clearly see, that the variance of approximately 20 districts is fixed on 20%. While the ISCEDB variable imposes some difficulties, the ISCEDC variable behaves like the ISCEDA variable and its benchmarks are more or less easy to handle.

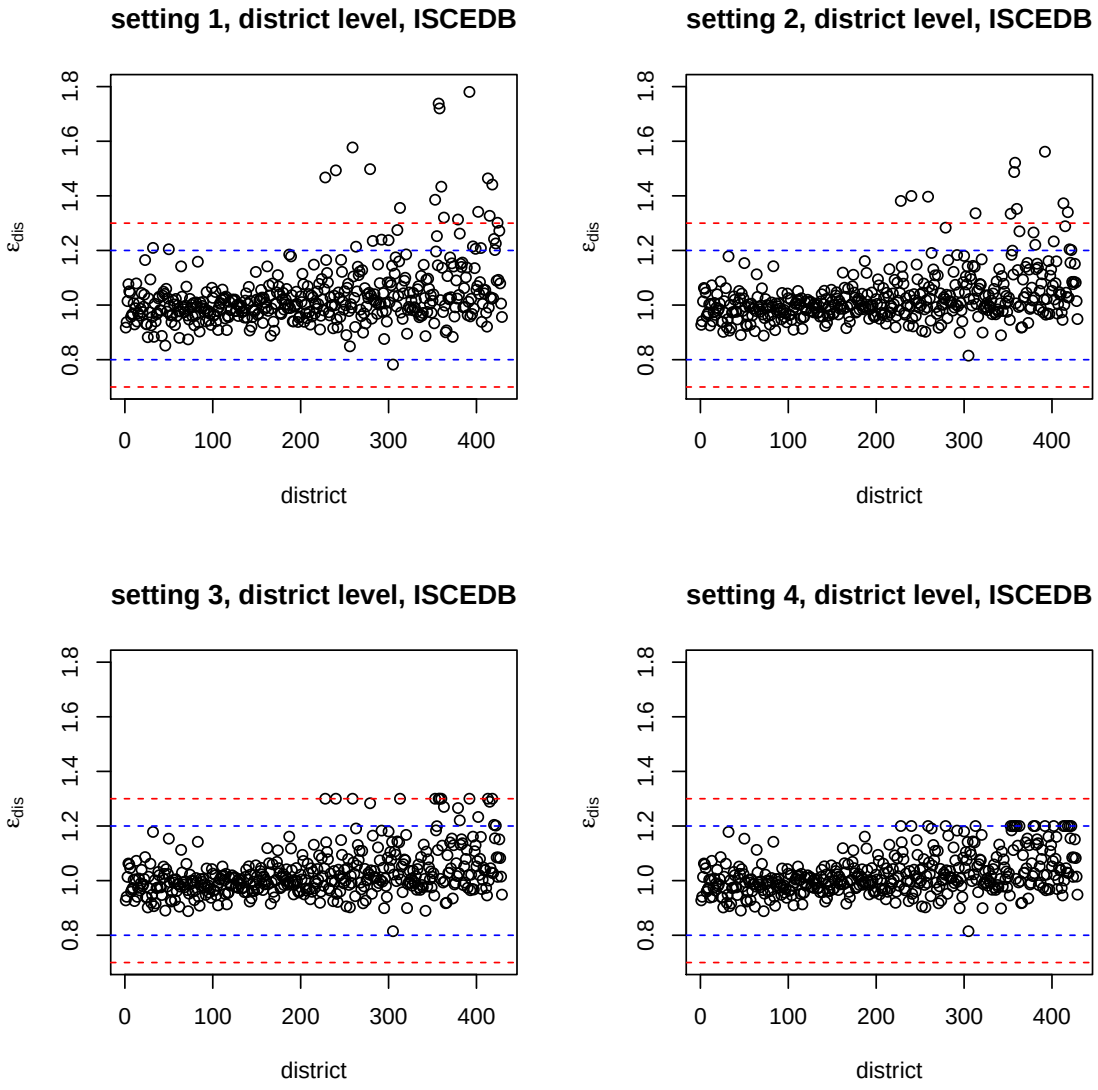


Figure 7.7: Perturbation ϵ_{dis} for benchmark ISCEDB and sample 651

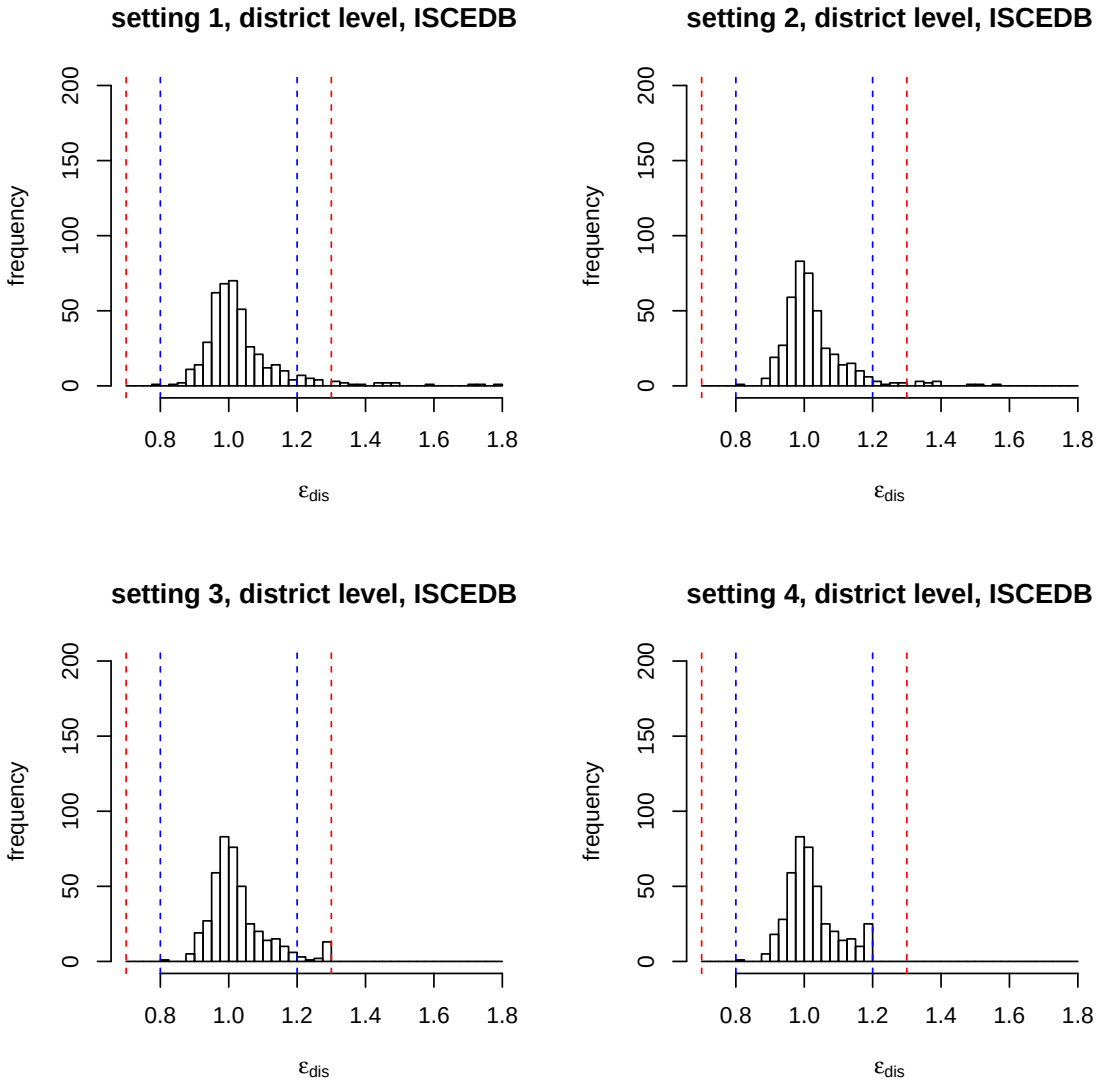


Figure 7.8: Histogram with absolute frequencies of the perturbation ϵ_{dis} for benchmark ISCEDB and sample 651

Regarding the ISCED variable on SMP level, we can state that ϵ_{SMP} behaves like ϵ_{dis} on district level. A closer look at the perturbation ϵ_{SMP} for benchmark ISCEDB given in Figure 7.9 reveals that in setting 1 there are many outliers. This number of outliers is reduced successfully by the penalty in setting 2, except for one SMP. Here we can again see the importance of the box constraint on ϵ, because in setting 3 and 4 the maximum perturbation of 30% is satisfied. We can also state, that a maximum perturbation of 20% can hardly be satisfied, because there are too many SMPs having a perturbation greater than 20%. In Figure 7.10 we can clearly see, that the penalty narrows the histograms of ϵ_{SMP}.

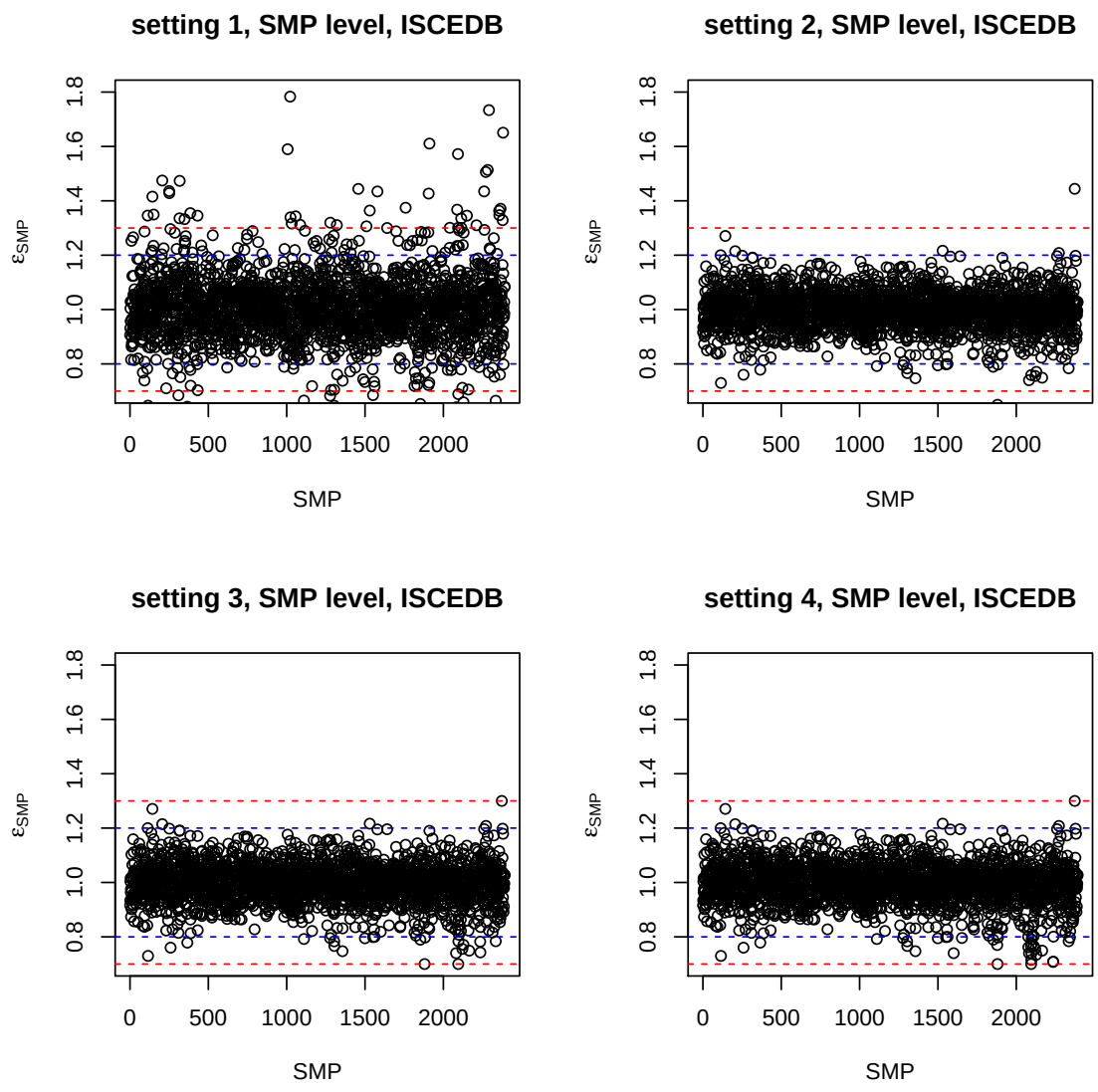


Figure 7.9: Perturbation ϵ_{SMP} for benchmark ISCEDB and sample 651

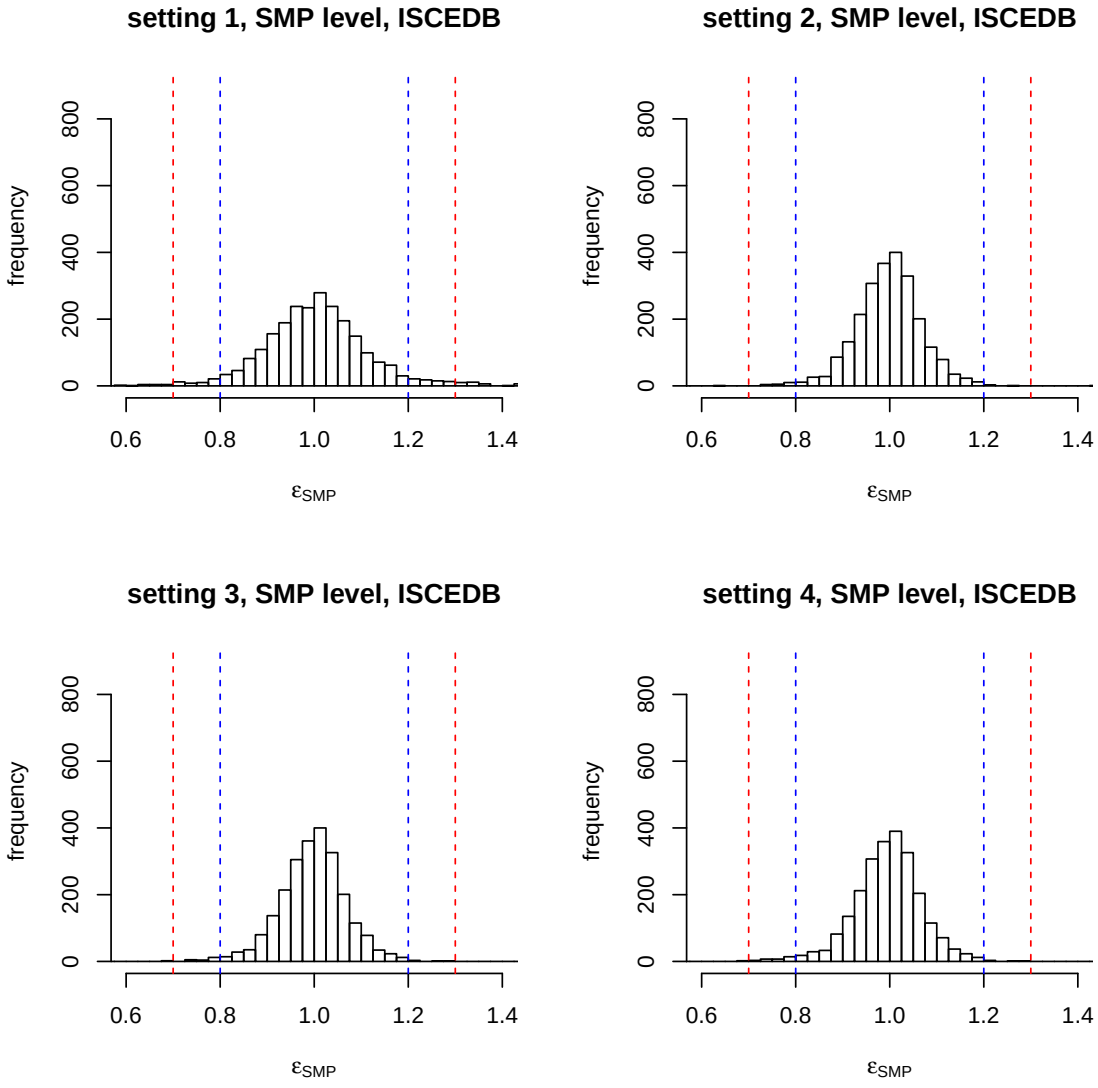


Figure 7.10: Histogram with absolute frequencies of the perturbation ϵ_{SMP} for benchmark ISCEDB and sample 651

As already mentioned, setting 3 and 4 are unsolvable for sample 394 due to one extremely large benchmark on SMP level. The benchmarks on district level are similar to the benchmarks of sample 651 and regarding Figure 7.11, it is most likely that the ISCED variable would not cause any problems concerning a limitation of the perturbation to 30% or even 20%.

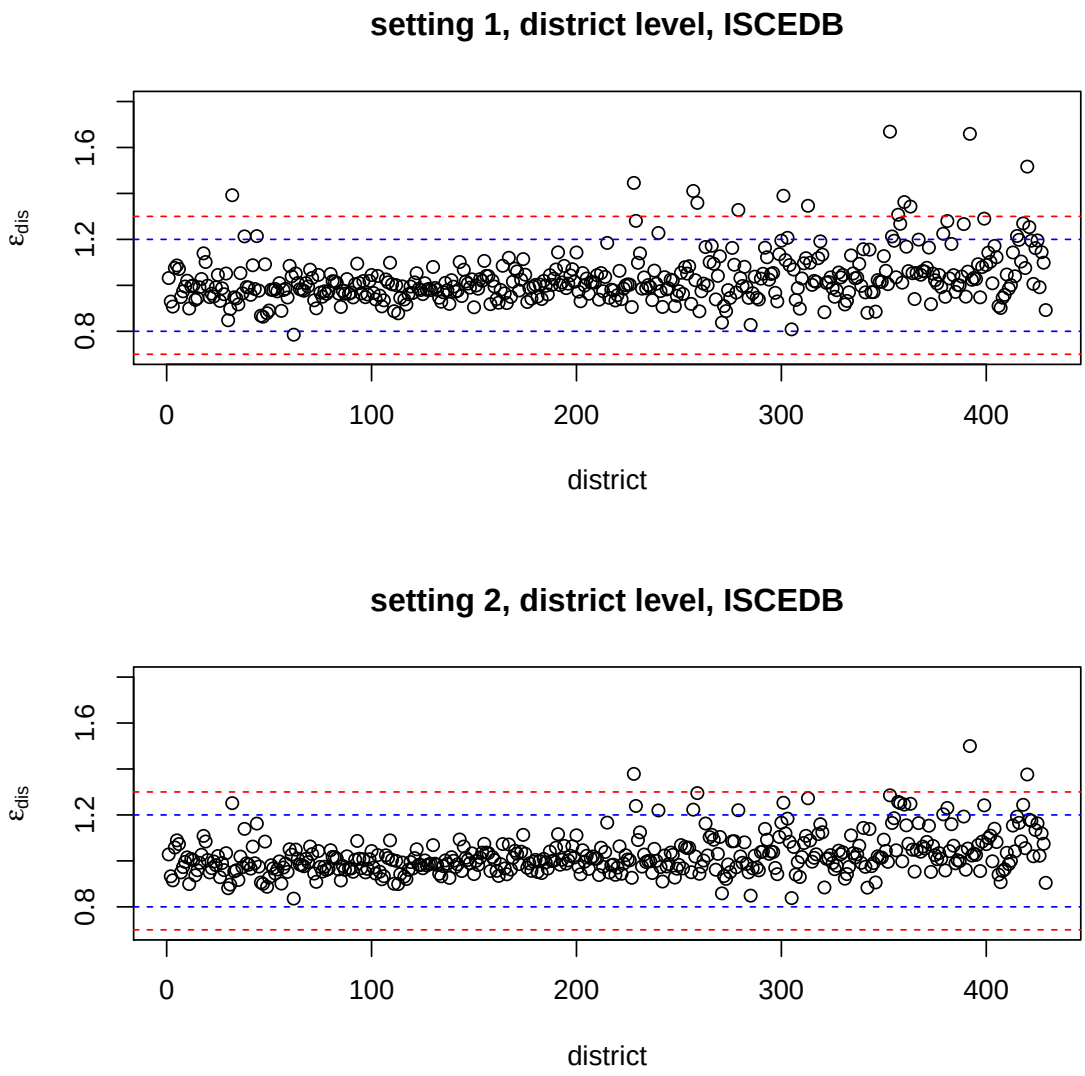
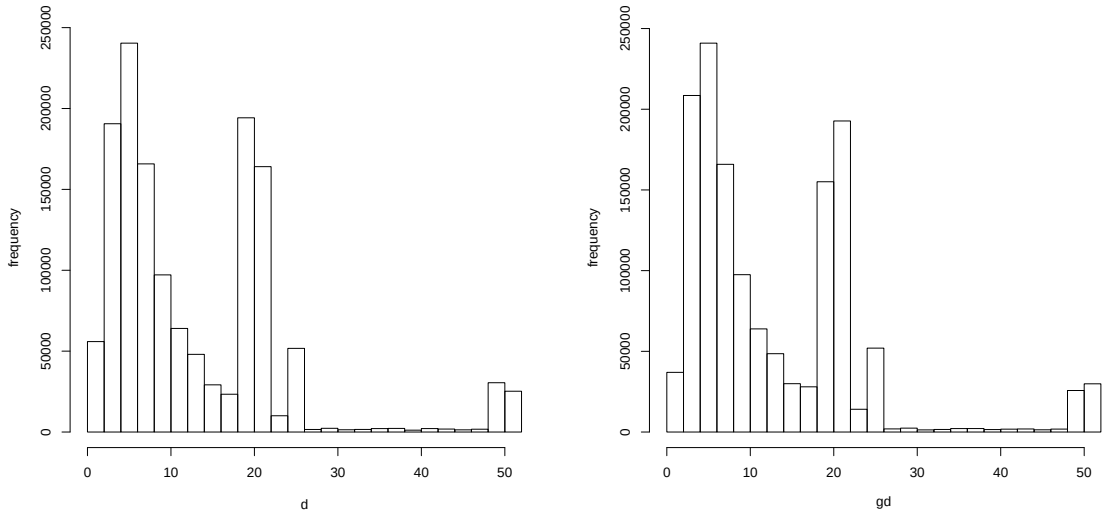


Figure 7.11: Perturbation ϵ_{dis} for benchmark ISCEDB and sample 394

Figure 7.12 shows the histogram of the initial weights d and the calibrated weights dg from setting 4 and sample 651. We can see that the distribution before calibration is roughly the same as the distribution after calibration.



(a) Initial weights d

(b) Calibrated weights dg

Figure 7.12: Histogram with absolute frequencies of the initial weights (a) and the calibrated weights (b)

However, a detailed look at the scatter plot in Figure 7.13 reveals, that the values of small initial weights slightly increase. This can be explained by comparing the given benchmarks and the simple Horvitz-Thompson estimator. The given benchmarks are usually larger than the estimates gained by the Horvitz-Thompson estimator, so this difference is counterbalanced by increasing the weights which leads to the seen results. Another interesting observation is that although the distribution of the weights around 20 has only minor changes, the amount of change of certain values of the weights is rather large, e.g., some weights switch from 19.82 to 41.66 whereas others switch from 21 to 2.1.

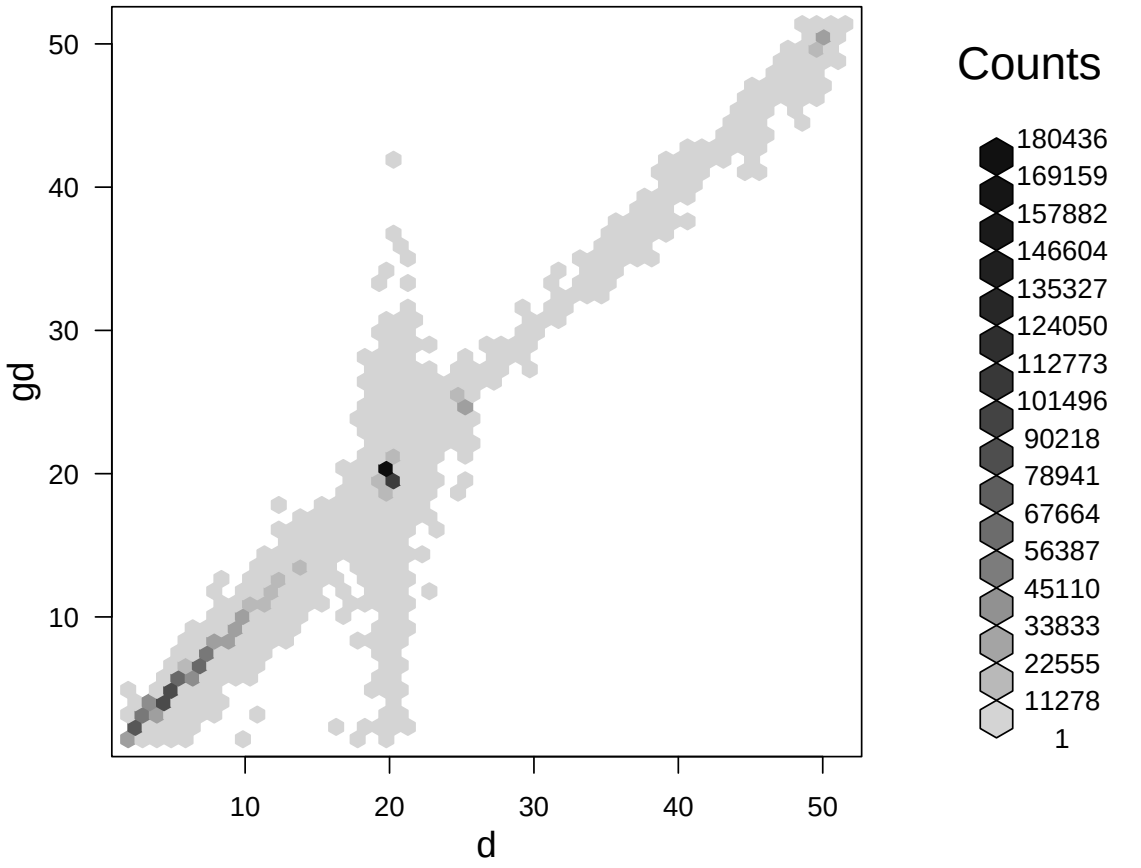


Figure 7.13: Scatter plot of the initial weights and the calibrated weights for sample 651

Another question arising is the role of the Lagrange multiplier. According to amount, the larger the Lagrange multiplier the greater the importance of the constraint during optimization. This property can also be examined at our simulation study. Regarding setting 2 for sample 394, the Lagrange multiplier λ_{9911} amounts approximately -71 where $\lambda_i \in (-2.1, 0.36)$ for all $i \in \{1, \ldots, 2830092\} \setminus \{9911\}$. Regarding the structure of the calibration matrix A, it is obvious that λ_{9911} is exactly the Lagrange multiplier belonging to the ISCEDB constraint in SMP 2078, which has a variation of 2270%. Having a look at the Lagrange multiplier for sample 561, no outliers can be noticed which is reasonable because regarding ϵ the estimates are 'good' and a satisfaction of the benchmarks can easily be reached.

In order to compare the optimal values of the optimal solutions we define the following functions of $z = (g, \epsilon_{dis}, \epsilon_{SMP}, \alpha, \beta)$:

$$
\begin{aligned}
OV_g : \mathbb{R}^{n+u+2} &\to \mathbb{R}, \\
z &\mapsto \sum_{k \in s} d_k \frac{(g_k - 1)^2}{2}, \\
OV_\epsilon : \mathbb{R}^{n+u+2} &\to \mathbb{R}, \\
z &\mapsto \sum_{k} \delta_k \frac{(\epsilon_{dis,k} - 1)^2}{2} + \sum_{k} \delta_k \frac{(\epsilon_{SMP,k} - 1)^2}{2}, \\
OV : \mathbb{R}^{n+u+2} &\to \mathbb{R}, \\
z &\mapsto \sum_{k \in s} d_k \frac{(g_k - 1)^2}{2} + \sum_{k} \delta_k \frac{(\epsilon_{dis,k} - 1)^2}{2} + \sum_{k} \delta_k \frac{(\epsilon_{SMP,k} - 1)^2}{2}.
\end{aligned}
$$

Regarding Table 7.5, we can see that the optimal value OV increases, that means getting worse, the more sophisticated a setting gets. This makes sense, because more constraints are added from one setting to the next. However, it is interesting to see that whilst OV_g increases, OV_ϵ decreases from setting 2 to setting 4. The increase of OV_g is easily explained: The more constraints are added, the more calibration factors move away from one. This can also be seen in Figure 7.4. In contrast to this, OV_ϵ decreases from setting 2 to setting 3 and slightly increases from setting 3 to setting 4. The decrease can be explained by the add of a box constraint limiting the values of ϵ. Nevertheless, as all ϵ_k depend on each other a stricter restriction may cause other ϵ_k to drift away from 1 such that OV_ϵ increases, which is the case in setting 4.

	setting 1	setting 2	setting 3	setting 4
OV_g	977	6,939	7,759	8,995
OV_ϵ	0	13,193	12,852	12,887
OV	977	20,132	20,612	21,883

Table 7.5: Rounded optimal values of sample 651

The objective values of sample 394 and setting 1 as well as setting 2 given in Table 7.6 can be compared to those of sample 651, where due to the unsolvability no optimal solution and therefore no objective values exist for setting 3 and setting 4.

	setting 1	setting 2	setting 3	setting 4
OV_g	921	6,457	-	-
OV_ϵ	0	13,159	-	-
OV	921	19,617	-	-

Table 7.6: Rounded optimal values of sample 394

As mentioned in Chapter 2 it is also desirable to know how 'good' an estimator is. In this case, we are interested in two aspects. The first one is how the perturbed estimates for the

ISCED variables gained by the calibrated Horvitz-Thompson estimator for domain d, that is

$$\hat{t}_{y,d}^{HTcal} = \sum_{k \in s_d} g_k d_k y_k,$$

behave compared to the YOURAO estimator. This is important because it gives us some information on the loss of accuracy in large tables and a one number census. The second point is how good calibrated Horvitz-Thompson estimates of further variables, e.g., EF117, are compared to the true values. Therefore, we regard the RRMSE and the RBias (cf. Chapter 2).

Table 7.7 shows the mean ($RRMSE_{mean}$), the minimum ($RRMSE_{min}$) as well as the maximum ($RRMSE_{max}$) of the RRMSE of the YOURAO estimator for the different ISCED variables. Here, mean/minimum/maximum means mean/minimum/maximum of the RRMSE of all SMPs. Note, that the RRMSE is computed for the estimates of setting 4 and that in order to get comparable results, only the estimates deriving from samples leading to a feasible calibration problem were considered in the computation.

	$RRMSE_{mean}$	$RRMSE_{min}$	$RRMSE_{max}$
ISCEDA	0.0206	0.0086	0.1963
ISCEDB	0.0728	0.0145	0.2409
ISCEDC	0.0530	0.0162	0.1993

Table 7.7: RRMSE of the YOURAO estimator for all SMPs and the different ISCED variables

Table 7.8 shows the same as Table 7.7 but instead of the YOURAO estimator, it is based on the calibrated Horvitz-Thompson estimator. We can see that for both estimators, the mean and the maximum of the RRMSE takes the highest values for the ISCEDB variable. This is in accord with the observations of ϵ because its variance is the largest in case of ISCEDB. For all ISCED variables, the mean of the RRMSE of the calibrated Horvitz-Thompson estimator lies below the mean of the YOURAO estimator, which is very positive. This property can also be seen regarding the maximum RRMSE, where for instance in the case of the ISCEDA variable the maximum RRMSE was lowered from 0.1963 to 0.1005.

	$RRMSE_{mean}$	$RRMSE_{min}$	$RRMSE_{max}$
ISCEDA	0.0201	0.0093	0.1005
ISCEDB	0.0565	0.0145	0.2575
ISCEDC	0.0435	0.0169	0.1348

Table 7.8: RRMSE of the calibrated Horvitz-Thompson estimator for all SMPs and the different ISCED variables

Regarding the RBias, we can state that the calibrated Horvitz-Thompson estimator leads to better results than the YOURAO estimator. As we can see in the histograms given in Figure 7.14 ISCEDA shows the best results because the estimators are almost unbiased. Further, the bias of the ISCEDB and ISCEDC variables is reduced because the anticipated density function is narrowed so most RBiases lie near to zero.

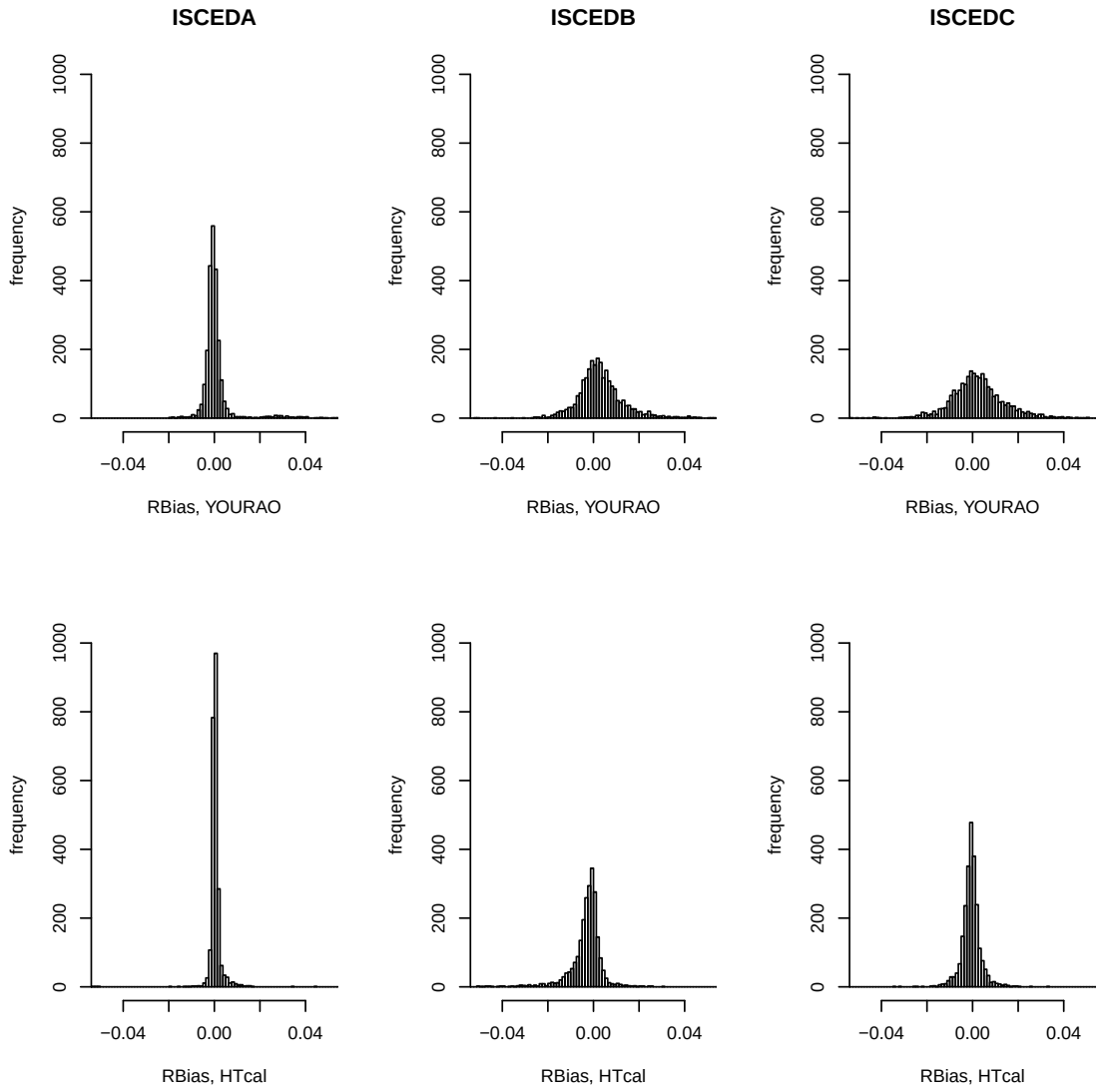


Figure 7.14: Histogram of the RBias of the YOURAO estimator and the calibrated Horvitz-Thompson estimator for all SMPs and the different ISCED variables

We will now have a look at the calibrated Horvitz-Thompson estimates of the EF117 variables. This variables classify the professional status, e.g., whether a person is an employee (EF117A), an official (EF117B) or a freelancer (EF117S), and we are interested in the total value over the whole population. Table 7.9 clearly shows that the RRMSE is very low leading to very good estimates. Further, as the RBias is extremely near to zero, the calibrated Horvitz-Thompson estimator is unbiased for the EF117 variables.

	RRMSE	*RBias*
EF117A	0.0010	−0.0001
EF117B	0.0029	−0.0005
EF117S	0.0019	−0.0004

Table 7.9: RRMSE and RBias of the calibrated Horvitz-Thompson estimator for the whole population and the different EF117 variables

In conclusion, the simulation study shows that a sole penalty approach as proposed in Beaumont and Bocci (2008) is not sufficient to deliver good results. Outliers can only be captured by adding a maximum admissible perturbation as it is done in our approach. Further, the results are quite promising and, with small reservations, our approach makes it possible to realize a one number census. However, a good data basis in terms of good estimates/benchmarks is inevitable for delivering good results.

Chapter 8

Conclusion and Outlook

In this thesis, we dealt with different optimization problems arising in survey statistics, such as an optimal allocation problem and calibration problems. Different methods for solving this allocation problem in continuous and integer variables were developed. The methods for the continuous case lead to one-dimensional root finding problems and for solving the integer allocation problem we used greedy type methods and developed a method based on a binary search. We further applied the algorithms to simulation data of the German Census Sampling and Estimation Research Project and compared the computing time and the number of iterations. Differences between the gained continuous and integer solution were also analyzed and pointed out.

For solving the general calibration problem we refined the optimality conditions leading to a nonsmooth equation. We proved the semismoothness of the obtained function so its root could be computed by the semismooth Newton method. This method was applied to simulation data and we showed that the theoretically proved quadratic convergence can also be encountered numerically. In the context of nonsmooth algorithms, we developed a general approach to step size rules. Here, we focused on nonmonotone, Armijo based line search methods but also showed that our approach can as well be applied to monotone line search methods. We also extended the classical calibration approach by adding further constraints on the calibrated weights and allowed some benchmarks to be relaxed in order to get vertical coherence. A solution method for this high dimensional quadratic program was implemented and a simulation study was conducted. In this study, we analyzed the simulation data of the German Census Sampling and Estimation Research Project and worked out the advantages and limits of the approach.

Main fields of application of the optimization methods mentioned in this thesis are sample based censuses, e.g., the German Census 2011 as well as future censuses. As a census is mandatory for every member state of the European Union every ten years, a lot of simulation studies have to be done in advance. By using our optimal allocation algorithms, a lot of simulations can be done in a short amount of time, such that different sampling fractions can be easily tested. Further, the already gained estimates of a sample based census may have to be delivered to Eurostat. For the use of those estimates in the Eurostat tables, the estimates have to be vertically coherent. Due to the use of small are methods, this coherence is not given and has to be artificially generated, which can be done by the presented extended calibration problem. In this context, the given R code should be generalized such that the method can be easily applied to other benchmarks or domain arrangements.

List of Tables

List of Figures

List of Algorithms

Bibliography

Armijo, L. (1966). Minimization of functions having lipschitz continuous first partial derivatives, *Pacific Journal of Mathematics* **16**(1): 2–5.

Battese, G. E., Harter, R. M. and Fuller, W. A. (1988). An error-components model for prediction of county crop areas using survey and satellite data, *Journal of the American Statistical Association* **83**(401): 28–36.

Beaumont, J.-F. and Bocci, C. (2008). Another look at ridge calibration, *Metron - International Journal of Statistics* **LXVI**(1): 5–20.

Bretthauer, K. M., Ross, A. and Shetty, B. (1999). Nonlinear integer programming for optimal allocation in stratified sampling, *European Journal of Operational Research* **116**(3): 667–680.

Burgard, J. P. and Münnich, R. T. (2012). Modelling over and undercounts for design-based monte carlo studies in small area estimation: An application to the german register-assisted census, *Computational Statistics and Data Analysis* **56**(10): 2856 – 2863.

Burgard, J. P., Münnich, R. T. and Zimmermann, T. (2013). *Small area modelling under complex survey designs for business data.* Available online at `http://www.blue-ets.istat.it`, visited 07/04/2013.

Chambers, R. (1996). Robust case-weighting for multipurpose establishment surveys, *Journal of Official Statistics* **12**(1): 3–32.

Chen, J., Sitter, R. R. and Wu, C. (2002). Using empirical likelihood methods to obtain range restricted weights in regression estimators for surveys, *Biometrika* **89**: 230–237.

Choudhry, G. H., Rao, J. and Hidiroglou, M. A. (2012). On sample allocation for efficient domain estimation, *Survey Methodology* **38**(1): 23–29.

Clarke, F. H. (1983). *Optimization and Nonsmooth Analysis*, Wiley, New York.

Cochran, W. (1977). *Sampling Techniques*, second edn, John Wiley and Sons, Ltd, New York, NY.

De Luca, T., Facchinei, F. and Kanzow, C. (1996). A semismooth equation approach to the solution of nonlinear complementarity problems, *Mathematical Programming* **75**: 407–439.

Demnati, A. and Rao, J. (2004). Linearization variance estimators for survey data, *Survey Methodology* **30**: 17–26.

Dennis, J. and Schnabel, R. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall.

Deville, J.-C. and Särndal, C.-E. (1992). Calibration estimators in survey sampling, *Journal of the American Statistical Association* **87**(418): 376–382.

Deville, J.-C., Särndal, C.-E. and Sautory, O. (1993). Generalized raking procedures in survey sampling, *Journal of the American Statistical Association* **88**(423): 1013–1020.

Dirkse, S. P. and Ferris, M. C. (1995). The path solver: a nommonotone stabilization scheme for mixed complementarity problems, *Optimization Methods and Software* **5**(2): 123–156.

Estevao, V. M. and Särndal, C.-E. (2006). Survey estimates by calibration on complex auxiliary information, *International Statistical Review* **74**(2): 127–147.

Facchinei, F., Fischer, A. and Kanzow, C. (1996). Inexact newton methods for semismooth equations with applications to variational inequality problems.

Facchinei, F. and Kanzow, C. (1997). A nonsmooth inexact newton method for the solution of large-scale nonlinear complementarity problems, *Mathematical Programming* **76**: 493–512.

Facchinei, F. and Pang, J.-S. (2003a). *Finite-Dimensional Variational inequalities and Complementarity Problems, Volume I*, Springer, New York, Berlin, Heidelberg.

Facchinei, F. and Pang, J.-S. (2003b). *Finite-Dimensional Variational inequalities and Complementarity Problems, Volume II*, Springer, New York, Berlin, Heidelberg.

Fay, R. E. and Herriot, R. A. (1979). Estimates of income for small places: An application of james-stein procedures to census data, *Journal of the American Statistical Association* **74**(366): 269–277.

Ferris, M., Kanzow, C. and Munson, T. (1998). Feasible descent algorithms for mixed complementarity problems, *Mathematical Programming* **86**: 475–497.

Ferris, M. and Lucidi, S. (1994). Nonmonotone stabilization methods for nonlinear equations, *Journal of Optimization Theory and Applications* **81**: 53–71.

Fischer, A. (1992). A special newton-type optimization method, *Optimization* **24**(3-4): 269–284.

Fischer, A. (1997). Solution of monotone complementarity problems with locally lipschitzian functions, *Mathematical Programming* **76**.

Frederickson, G. and Johnson, D. (1982). The complexity of selection and ranking in $X + Y$ and matrices with sorted columns, *Journal of Computer and System Sciences* **24**(2): 197–208.

Friedrich, U., Münnich, R. T. and Wagner, M. (2013). Integer optimization in stratified sampling, *in preparation* .

Gabler, S., Ganninger, M. and Münnich, R. T. (2012). Optimal allocation of the sample size to strata under box constraints, *Metrika* **75**: 151–161.

Gelius-Dietrich, G. (2012). *cplexAPI*. R package version 1.2.2, `http://CRAN.R-project.org/package=cplexAPI`.

Gelman, A. (2007a). Rejoinder: Struggles with survey weighting and regression modeling, *Statistical Science* **22**(2): 184–188.

Gelman, A. (2007b). Struggles with survey weighting and regression modeling, *Statistical Science* **22**(2): 153–164.

Gill, P. E., Murray, W. and Wright, M. H. (1981). *Practical Optimization*, Academic Press, inc.

Goldfarb, D. and Idnani, A. (1982). Dual and primal-dual methods for solving strictly convex quadratic programs, *in* J. Hennart (ed.), *Numerical Analysis*, Vol. 909 of *Lecture Notes in Mathematics*, Springer Berlin Heidelberg, pp. 226–239.

Goldfarb, D. and Idnani, A. (1983). A numerically stable dual method for solving strictly convex quadratic programs, *Mathematical Programming* **27**: 1–33.

Grippo, L., Lampariello, F. and Lucidi, S. (1986). A nonmonotone line search technique for newton's method, *SIAM Journal on Numerical Analysis* **23**(4): 707–716.

Groenevelt, H. (1991). Two algorithms for maximizing a separable concave function over a polymatroid feasible region, *European Journal of Operational Research* **54**(2): 227–236.

Groves, R. M., Fowler Jr., F. J., Couper, M. P., Lepkowski, J. M., Singer, E. and Tourangeau, R. (2004). *Survey Methodology*, John Wiley and Sons, Ltd.

Han, S.-P., Pang, J.-S. and Rangaraj, N. (1992). Globally convergent newton methods for nonsmooth equations, *Mathematics of Operations Research* **17**(3): 586–607.

Harker, P. T. and Xiao, B. (1990). Newton's method for the nonlinear complementarity problem: A b-differentiable equation approach, *Mathematical Programming* **48**: 339–357.

Harris, J. W. and Stocker, H. (1998). *Handbook of Mathematics and Computational Science*, Springer, New York, Berlin, Heidelberg.

Hochbaum, D. S. (1994). Lower and upper bounds for the allocation problem and other nonlinear optimization problems, *Mathematics of Operations Research* **19**(2): 390–409.

Hohnhold, H. (2009). Variants of optimal allocation in stratified sampling, *Technical report*, Statistisches Bundesamt, Wiesbaden.

Horst, R. (1979). *Nichtlineare Optimierung*, Carl Hanser Verlag.

Horvitz, D. G. and Thompson, D. J. (1952). A generalization of sampling without replacement from a finite universe, *Journal of the American Statistical Association* **47**(260): 663–685.

Ibaraki, T. and Katoh, N. (1988). *Resource allocation problems: algorithmic approaches*, MIT Press, Cambridge, MA, USA.

Ito, K. and Kunisch, K. (2009). On a semi-smooth newton method and its globalization, *Mathematical Programming* **118**: 347–370.

Kanzow, C. (2000). Global optimization techniques for mixed complementarity problems, *Journal of Global Optimization* **16**: 1–21.

Kanzow, C. (2004). Inexact semismooth newton methods for large-scale complementarity problems, *Optimization Methods and Software* **19**(3-4): 309–325.

Kanzow, C. (2005). Nichtglatte analysis mit anwendungen. Available online at `http://www.mathematik.uni-wuerzburg.de/~kanzow/opt/Semi_05.pdf`, visited 10/29/2012.

Kim, J. K. and Park, M. (2010). Calibration estimation in survey sampling, *International Statistical Review* **78**(1): 21–39.

Kolb, J.-P. (2012). *Methoden zur Erzeugung synthetischer Simulationsgesamtheiten*, PhD thesis, University of Trier.

Kott, P. S. (2006). Using calibration weighting to adjust for nonresponse and coverage errors, *Survey Methodology* **32**(2): 133–142.

Kummer, B. (1988). Newton's method for non-differentiable functions, *in* J. Guddat, B. Bank, H. Hollatz, P. Kall, D. Klatte, B. Kummer, K. Lommatzsch, K. Tammer, M. Vlach and K. Zimmermann (eds), *Advances in mathematical optimization: invited papers dedicated to Prof. Dr. Dr. h.c. F. Nožička on occasion of his 70th birthday*, Mathematical research, Akademie-Verlag, pp. 114–125.

Lehtonen, R. and Veijanen, A. (2009). Chapter 31 - design-based methods of estimation for domains and small areas, *in* C. Rao (ed.), *Handbook of Statistics Sample Surveys: Inference and Analysis*, Vol. 29, Part B of *Handbook of Statistics*, Elsevier, pp. 219–249.

Little, R., Meng, X.-L. and Gelman, A. (2009). Invited paper session, *Joint Statistical Meeting 2009, Washington* .

Markowitz, H. (1952). Portfolio selection, *The Journal of Finance* **7**(1): 77–91.

Martínez, J. M. and Qi, L. (1995). Inexact newton methods for solving nonsmooth equations, *Journal of Computational and Applied Mathematics* **60**(1-2): 127–145.

Mifflin, R. (1977). Semismooth and semiconvex functions in constrained optimization, *SIAM Journal on Control and Optimization* **15**: 959–972.

Montanari, G. E. and Ranalli, M. G. (2009). Multiple and ridge model calibration for sample surveys, *Proceedings of the Workshop in Calibration and Estimation in Surveys, Ottawa, October 2007*, Statistics Canada.

Münnich, R. T., Burgard, J. and Vogt, M. (2013). Small area-statistik: Methoden und anwendungen, *AStA Wirtschafts- und Sozialstatistisches Archiv* **6**(3): 149–191.

Münnich, R. T., Gabler, S., Ganninger, M., Burgard, J. P. and Kolb, J.-P. (2012). *Statistik und Wissenschaft: Stichprobenoptimierung und Schätzung im Zensus 2011*, Vol. 21, Statistisches Bundesamt, Wiesbaden.

Münnich, R. T., Sachs, E. W. and Wagner, M. (2012a). Calibration benchmarking for small area estimates: An application to the german census 2011, *Symposium on the Analysis of Survey Data and Small Area Estimation in Honour of the 75th Birthday of J.N.K Rao, Ottawa, Invited Paper Session* .

Münnich, R. T., Sachs, E. W. and Wagner, M. (2012b). Calibration of estimator-weights via semismooth newton method, *Journal of Global Optimization* **52**: 471–485.

Münnich, R. T., Sachs, E. W. and Wagner, M. (2012c). Numerical solution of optimal allocation problems in stratified sampling under box constraints, *AStA Advances in Statistical Analysis* **96**: 435–450.

Neyman, J. (1934). On the two different aspects of the representative method: The method of stratified sampling and the method of purposive selection, *Journal of the Royal Statistical Society* **97**: 558–606.

Nocedal, J. and Wright, S. (2006). *Numerical Optimization*, second edn, Springer, Berlin.

Pang, J.-S. (1990). Newton's method for b-differentiable equations, *Mathematics of Operations Research* **15**: 311–341.

Pang, J.-S. (1991). A b-differentiable equation-based, globally and locally quadratically convergent algorithm for nonlinear programs, complementarity and variational inequality problems, *Math. Program.* **51**(1): 101–131.

Pang, J.-S. and Qi, L. (1993). Nonsmooth equations: motivation and algorithms, *SIAM Journal of Optimization* **3**.

Patriksson, M. (2008). A survey on the continuous nonlinear resource allocation problem, *European Journal of Operational Research* **185**(1): 1–46.

Qi, L. (1993). Convergence analysis of some algorithms for solving nonsmooth equations, *Mathematics of Operations Research* **18**(1): 227–244.

Qi, L. and Sun, J. (1993). A nonsmooth version of newton's method, *Mathematical Programming* **58**.

Qi, L. and Sun, J. (1994). A trust region algorithm for minimization of locally lipschitzian functions, *Mathematical Programming* **66**: 25–43.

R Development Core Team (2012). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. `http://www.R-project.org/`.

Rademacher, H. (1919). über partielle und totale differenzierbarkeit von funktionen mehrerer variabeln und über die transformation der doppelintegrale, *Mathematische Annalen* **79**: 340–359.

Raj, D. (1968). *Sampling theory*, McGraw-Hill series in probability and statistics, McGraw-Hill.

Ralph, D. (1994). Global convergence of damped newton's method for nonsmooth equations via the path search, *Mathematics of Operations Research* **19**(2): 352–389.

Ralston, A. and Rabinowitz, P. (1978). *A First Course in Numerical Analysis*, McGraw-Hill.

Rao, J. (2003). *Small Area Estimation*, John Wiley and Sons, Ltd.

Rao, J. and Singh, A. (1997). A ridge-shrinkage method for range-restricted weight calibration in survey sampling, *Proceedings of the Survey Research Methods Section*, American Statistical Association, pp. 57–65.

Rao, J. and Singh, A. (2009). Range restricted weight calibration for survey data using ridge regression, *Pakistan Journal of Statistics* **25**: 371–384.

Robinson, S. M. (1987). Local structure of feasible sets in nonlinear programming, part iii: Stability and sensitivity, *Nonlinear Analysis and Optimization*, Vol. 30 of *Mathematical Programming Studies*, Springer Berlin Heidelberg, pp. 45–66.

Rossi, P., Wright, J. and Anderson, A. (1983). *Handbook of survey research*, Quantitative studies in social relations, Academic Press.

Rudin, W. (1991). *Functional analysis*, McGraw-Hill Inc., New York.

Ruszczynski, A. (2006). *Nonlinear Optimization*, Princeton University Press, Princeton, NJ, USA.

Sachs, E. W. and Sachs, S. M. (2011). Nonmonotone line searches for optimization algorithms, *Control and Cybernetics* **40**(4): 1059 – 1075.

Sanathanan, L. (1971). On an allocation problem with multistage constraints, *Operations Research* **19**(7): 1647–1663.

Särndal, C.-E. (2007). The calibration approach in survey theory and practice, *Survey Methodology* **33**(2): 99–119.

Särndal, C.-E., Swensson, B. and Wretman, J. (2003). *Model Assisted Survey Sampling*, Springer.

Schrijver, A. (2003). *Combinatorial optimization: polyhedra and efficiency. Volume B*, Springer Verlag.

Shapiro, A. (1990). On concepts of directional differentiability, *Journal of Optimization Theory and Applications* **66**: 477–487.

Singh, A. and Mohl, C. (1996). Understanding calibration estimators in survey sampling, *Survey Methodology* **22**(2): 107–115.

Srikantan, K. S. (1963). A problem in optimum allocation, *Operations Research* **11**(2): 265–273.

Stefanov, S. (2006). Minimization of a convex linear-fractional separable function subject to a convex inequality constraint or linear inequality constraint and bounds on the variables, *Applied Mathematics Research eXpress* **2006**: 1–24.

Stenger, H. and Gabler, S. (2005). Combining random sampling and census strategies. Justification of inclusion probabilities equal to 1, *Metrika* **61**: 137–156.

Stukel, D., Hidiroglou, M. and Särndal, C.-E. (1996). Variance estimation for calibration estimators: A comparison of jackknifing versus taylor linearization, *Survey Methodology* **22**: 117–125.

Théberge, A. (2000). Calibration and restricted weights, *Survey Methodology* **26**: 99–107.

Tichatschke, R. and Kaplan, A. (1994). *Stable Methods for Ill-Posed Variational Problems*, Akademie-Verlag, Berlin.

Tillé, Y. and Matei, A. (2009). *sampling: Survey Sampling*. R package version 2.3, `http://CRAN.R-project.org/package=sampling`.

Tschuprow, A. (1923). On the mathematical expectation of the moments of frequency distributions in the case of correlated observations, *Metron* **2**: 461–493.

Turlach, B. A. and Weingessel, A. (2011). *quadprog*. R package version 1.5-4, `http://CRAN.R-project.org/package=quadprog`.

Vanderhoeft, C. (2001). Statistics belgium working papers - generalised calibration at statistics belgium, *Technical report*, STATBEL.

Werner, D. (2007). *Funktionalanalysis*, Springer-Verlag, Berlin Heidelberg.

Yamamuro, S. (1974). *Differential Calculus in Topological Linear Spaces*, Springer-Verlag, Berlin Heidelberg New York.

You, Y. and Rao, J. N. K. (2002). A pseudo-empirical best linear unbiased prediction approach to small area estimation using survey weights, *The Canadian Journal of Statistics / La Revue Canadienne de Statistique* **30**(3): 431–439.

Zhang, H. and Hager, W. W. (2004). A nonmonotone line search technique and its application to unconstrained optimization, *SIAM Journal on Optimization* **14**: 1043–1056.