The performance of Iterative Proportional Fitting for spatial microsimulation: applying new tests to an old technique

Iterative Proportional Fitting (IPF) is an established technique in Regional Science and has been used for a variety of purposes in the field, primary amongst these being the generation of small area microdata. The technique is mature, widely used and relatively straight-forward yet there have been few studies focussed on evaluation of its performance. An additional research problem is the tendency of researchers to ‘start from scratch’, resulting in a variety ad-hock implementations, with little evidence of the merits of differing approaches. Although IPF is well described mathematically, reproducible examples of the algorithm, written in modern programming language, are rare in the academic literature. These knowledge gaps mean that answers to methodological questions must be guessed: Under what conditions can a ‘perfect’ match be expected, and how can one test whether these conditions exist in the input datasets? How many iterations should be used for any given application? What is the impact of initial weights on the final results? And what impact will integerisation have on accuracy? This paper tackles such questions, using a systematic methodology based on publicly available code and input data. The results confirm IPF’s status as robust and widely applicable method for combining areal and individual-level data and demonstrate the importance of using various metrics of for determining its performace. We conclude with an agenda for future tests and offer more general guidance on how the the method can be optimised to maximise its utility for future research.

Keywords: Iterative proportional fitting, spatial microsimulation, modelling

JEL Code: C

# Notes for authors

Additonal papers to cite include:

# Introduction

The aggregation of survey data has a long history in the social sciences. Generally this involves translating the raw data from a ‘long’ format — the usual format of raw census data as it is digitised and entered into a database, where each row represents an individual and each column a separate variable — to a ‘wide’ data format . The wide data form — where each row represents a geographic area or other sub-group of the total population and each column represents a variable value or bin (see ) — generally occupies less disk space for storage but has some important disadvantages. Data loss usually occurs when survey data is aggregated into the wide form, especially when continuous variables are converted into discrete categories and when not cross-tabulations, such as age/income, are not provided.

The process of aggregation is performed in three steps: a) conversion of all variables into discrete categories (e.g. an individual with an income of £13,000 per year would be allocated to band of £10,000 to £15,000 per year), b) assigning a column to each category — and in some cases each category combination — and c) assigning integer counts to cells in each column. Frequently large national social survey datasets such as censuses are *only* available in the wide, geographically aggregated form, which is problematic for the analysis of spatial phenomena .

Iterative proportional fitting (IPF) can help overcome these issues when used in *spatial microsimulation*, which can be understood in simplistic terms as the reversal of geographic aggregation to approximate the original ‘long’ dataset (). As illustrated in this figure, the need for spatial microsimulation is driven by the tendency of statistical agencies to make the raw georeferenced microdata generated through large social surveys unnavailable to the research community : what geographers and regional scientists refer to as spatial microsimulation is generally known as *population synthesis* in transport modelling, which is and apt name: its core function is to simulate populations based on imperfect data. Spatial microsimulation also refers to the wider approach which harnesses spatial microdata in the analysis stage . In this paper we focus on the former meaning, which is also known as *static spatial microsimulation* .

image

## What is IPF and why use it?

IPF is a statistical procedure that enables the calculation of *maximum likelihood* of values that are constrained by multiple variables. IPF is a general purpose tool with multiple uses. In this paper we focus on its ability to *disaggregate* the geographically aggregated data typically provided by national statistical agencies to simulate *spatial microdata* (). But why would one want do this in the first place?

The main advantages of wide datasets are that they can summarise information about a very large number of individuals in a small number of rows, saving storage space, easing data analysis and, critically, ensuring the anonymity survey respondents. As alluded to above, the main disadvantage is that the process of aggregation is almost always associated with data loss: we do not know from a wide dataset if the hypothetical individual earning £13,000 is in fact on a salary of £10,000, £15,000 or anywhere in between. Also, when aggregation is done by geographical zones — as is usually the case — there is a loss of geographical resolution: instead of knowing person’s precise home location from the full dataset, we may only be able to pin them down to a relatively large administrative area such as a Local Authority. Finally, the cross-tabulations between variables are generally during the process of aggregation.

Both types of data representation are common in the physical sciences, and the raw long dataset is often available. In the social sciences, by contrast, the long format is often unavailable due to the long history of and confidentiality issues surrounding officially collected survey data. It worth emphasising that National Census datasets — in which characteristics of every individual are recorded and which constitute some of the best datasets in social science — are usually only available in wide form, however, posing major challenges to researchers in the sciences, posing problems to researchers interested in intra-zone variability at the local level (e.g. ).

A related long-standing research problem especially prevalent in geography and regional science is the scale of analysis. There is often a trade-off between the quality and resolution of the data, which can result in a compromise between detail and accuracy of results . A common scenario facing researchers investigating a spatial problem is being presented with two datasets: one long aspatial survey table where each row represents the characteristics of an individual — such as age, sex, employment status and income — and another of wide, geographically aggregated dataset containing count data for each category. IPF in spatial microsimulation tackles this problem by allocating weights to the individuals from the aspatial microdataset for each zone, based on *constraint variables* shared by both input datasets. It is these ‘linking variables’ that enable IPF to converge on the weights that best represent how representative each individual is of each zone, as described in the next section.

## IPF in spatial microsimulation

Spatial microsimulation tackles the afforementioned problems associated with the prevalent geographically aggregated wide data form by simulating the individuals within each administrative zone. The most detailed output of this process of spatial microsimulation is ‘spatial microdata’, a long table of individuals with associated attributes and weights for each zone (see for a simple example). Because these individuals correspond to observations in the survey dataset, the spatial microdata generated in this way, of any size and complexity, can be stored with only 3 columns: person identification number (ID), zone and weight. Weights are critical to spatial microdata generated in this way, as they determine how representative each individual is of each zone. Thus, an even more compact way of storing spatial microdata taken from a single dataset is as a ‘weight matrix’ (table 2).

Representing the data in this form offers a number of advantages (Clarke and Holm, 1987). These include:

* The ability to investigate intra-zone variation (by analysing a the variable after subsetting a single area from the data frame),and inter-zone variation in the same dataset.
* Cross-tabulations can be created for any combination of variables (e.g. age and employment status).
* The presentation of continuous variables such as income as real numbers, rather than as categorical count data, in which there is always data loss.
* The data is in a form that is ready to be passed into an individual-level model. For example individual behaviour could simulate traffic, migration or the geographical distribution of future demand for healthcare.

These benefits have not gone unnoticed by the spatial modelling community. Spatial microsimulation is now more than a methodology, and can be considered as a research field in its own right, with a growing body of literature, methods and results . In tandem with these methodological developments, a growing number of applied studies has emerged to take advantage of the new possibilities opened up by such small area individual-level data, with applications as diverse as the location of health services to the analysis of commuting . Various strategies for generating small area microdata are available, as described in recent overviews on the subject . However, it is not always clear which one is most appropriate, as highlighted in the conclusions of a recent Reference Guide to spatial microsimulation : “It would be desirable if the spatial microsimulation community were able to continue to analyse which of the various reweighting/synthetic reconstruction techniques is most accurate — or to identify whether one approach is superior for some applications while another approach is to be preferred for other applications”.

The longest-established and most straight-forward of these is Iterative Proportional Fitting (IPF), the performance of which is the subject of this paper...

## Applications

**Dimitris and Eveline: please fill in this section** Applications of IPF include using it ...

The application of greatest interest to regional scientists is as a method of spatial microsimulation: generating individual level data allocated to specific zones... Papers that have used IPF in this context (or a subsitutable method such as simulated annealing) have explored a range of regional economic problems that purely aggregate level analyses struggle to address. These include:

* using spatial microdata to analyses the wider impacts of major job losses at the local level .
* an investigation into to local impacts of the marine sector in Ireland
* ...
* ...

# IPF: theory and a worked example

Iterative Proportional Fitting has a long history in Statistics and the social sciences (especially in Economics and Geography). In statistical language, IPF is used to estimate the values of internal cells in a cross-tabulated table to fit known margins. The method is also known as matrix raking or scaling, biproportional fitting, RAS, and entropy maximising. There are a number of texts discussing the method in a mathematical formal way in some detail (for example ). provide a succinct presentation of the mathematics of IPF and their work is the foundation of the work presented here...

## IPF theory: a worked example

In most modelling texts there is a strong precedence of theory over application: the latter usually flows from the former. The location of this section after a description of the programming language R is therefore a little unconventional but there is a logic to this order. Having demonstrated the power and flexibility of the programming language in which the model is written, the next stage is to analyse the task to which it is to be set. More importantly for reproducible research, this theory section is illustrated with a simple worked example that culminates in a question to the reader, to test his or her understanding.

IPF is a simple statistical procedure, “in which cell counts in a contingency table containing the sample observations are scaled to be consistent with various externally given population marginals” . In other words, and in the context of *spatial* microsimulation, IPF produces maximum likelihood estimates for the frequency with which people appear in different areas. The method is also known as ‘matrix raking’ or the RAS algorithm, and has been described as one particular instance of a more general procedure of ‘entropy maximisation’ . The mathematical properties of IPF have been described in several papers (). Illustrative examples of the procedure can be found in , and . investigated the reliability of IPF and evaluated the importance of different factors influencing the its performance. Similar methodologies have since been employed by , and Ballas et al.  to investigate a wide range of phenomena.

To illustrate how IPF works in practice, a simplified example is described below. This is a modified version of a simpler demonstration from .[[1]](#footnote-28) Table [t:w] describes a hypothetical microdataset comprising 5 individuals, who are defined by two constraint variables, age and sex. Each has two categories. Table [t:s] contains aggregated data for a hypothetical area, as it would be downloaded from census dissemination portal Casweb. illustrates this table in a different form, which shows our ignorance of interaction between age and sex.

[h]

llll & Sex & Age-group & Weight  
1 & Male & Over-50 & 1  
2 & Male & Over-50 & 1  
3 & Male & Under-50 & **1**  
4 & Female & Over-50 & 1  
5 & Female & Under-50 & 1

[t:w]

[htbp]

cllll Constraint & &  
Category & & & &   
Area & Under-50 & Over-50 & Male & Female  
1 & 8 & 4 & 6 & 6

[t:s]

[htbp]

cllllMarginal totals& & &  
& Age/sex & Male & Female & T  
 & Under-50 & **?** & ? & 8  
& Over-50 & ? & ? &4  
& T & 6 & 6 &12

[t:s2]

Table [t:m] presents the hypothetical microdata in aggregated form, that can be compared directly to Table [t:s2].

[htbp]

cllllMarginal totals& & &  
& Age/sex & Male & Female & T  
 & Under-50 & **1** & 1 & 2  
& Over-50 & 2 & 1 &3  
& T & 3 & 2 &5

[t:m]

Using these data it is possible to readjust the weights of the hypothetical individuals, so that their sum would add up to the totals given in Table [t:s2] (12). In particular, the weights can be readjusted by multiplying them by the marginal totals, originally taken from Table [t:s] and then divided by the respective marginal total in [t:m]. Because the total for each small-area constraint is 12, this must be done one constraint at a time. This can be expressed, for a given area and a given constraint ( or age in this case), as follows:

where is the new weight for individuals with characteristics (age, in this case), and (sex), is the original weight for individuals with these characteristics, is element marginal total of the small area constraint, (Table [t:s]) and is the marginal total of category of the aggregated results of the weighted microdata, (Table [t:m]). represents the iteration number. Although the marginal totals of are known, its cell values are unknown. Thus, IPF estimates the interaction (or cross-tabulation) between constraint variables. (Follow the emboldened values in the tables to see how the new weight of individual 3 is calculated for the sex constraint.) Table [t:new-weights] illustrates the weights that result. Notice that the sum of the weights is equal to the total population, from the constraint variables.

[htbp]

lllll & Sex & age-group & Weight & New weight, w(2)  
1 & Male & Over-50 & 1 &   
2 & Male & Over-50 & 1 &   
3 & Male & Under-50 & 1 &   
4 & Female & Over-50 & 1 &   
5 & Female & Under-50 & 1 &

[t:new-weights]

After the individual level data have been re-aggregated (), the next stage is to repeat for the age constraint to generate a third set of weights, by replacing the in and with and incrementing the value of n:

To test your understanding of IPF, apply to the information above and that presented in . This should result in the following vector of new weights, for individuals 1 to 5:

As before, the sum of the weights is equal to the population of the area (12). Notice also that after each iteration the fit between the marginal totals of and improves. The total absolute error (TAE) from to improves from 14 to 6 in and above. TAE for (not shown, but calculated by aggregating ) improves even more, to 1.3. This number would eventually converge to 0 through subsequent iterations, as there are no empty cells in the input microdataset; a defining feature of IPF.

[htbp]

cllllMarginal totals& & &  
& Age/sex & Male & Female & T  
 & Under-50 & 4 & 4 & 8  
& Over-50 & & & 4  
& T & & 5 & 12

[t:m2]

The above process, when applied to more categories (e.g. socio-economic class) and repeated iteratively until a satisfactory convergence occurs, results in a series of weighted microdatasets, one for each of the small areas being simulated. This allows for the estimation of variables whose values are not known at the local level (e.g. income) . An issue with the results of IPF (absent from combinatorial optimisation methods), however, is that it results in non-integer weights: fractions of individuals appear in simulated areas. As described in the introduction, this is not ideal for certain applications. Integer weights allow the results of spatial microsimulation to be further processed using dynamic microsimulation and agent based modelling techniques .

Spatial microsimulation can also provide insight into the likely distribution of individual level variables about which only geographically aggregated statistics have been made available. An issue with the results of IPF (absent from combinatorial optimisation methods), however, is that it results in non-integer weights: fractions of individuals appear in simulated areas.

## Implementing IPF in R

The above example is best undertaken by hand, probably with a pen and paper to gain an understanding of IPF, before the process is automated for larger datasets. This section explains how the IPF algorithm described above is implemented in R, using a slightly more complex example .[[2]](#footnote-30)

# Evaluation techniques

To verify the integrity of any model, it is necessary to compare its outputs with empirical observations or prior expectations gleaned from theory. The same principles apply to spatial microsimulation, which can be evaluated using both internal and *external* validation methods . *Internal validation* is the process whereby the aggregate-level outputs of spatial microsimulation are compared with the input constraint variables. Internal validation typically tackle such issues as “do the simulated populations in each area microdata match the total populations implied by the constrain variables?” and “what is the level of correspondence between the cell values of different attributes in the aggregate input data and the aggregated results of spatial microsimulation?” As we shall see below, ideally we would hope for a perfect fit between the input and output datasets during internal evaluation. *External validation* is the process whereby the variables that are being estimated are compared to data from another source, external to the estimation process, so the output dataset is compared with another known dataset for those variables.

This section outlines the evaluation techniques that are commonly used in the field, with a strong focus on internal validation. The options for external validation are heavily context dependent so must generally be decided on a case-by-case basis, and are mentioned in passing with a focus on general principles rather than specific advice. The methods are presented in ascending order of complexity and roughly descending order of frequency of use, ranging from Pearson’s coefficient of correlation to entropy based measures. Before these methods are described, it is worth stating the the purpose of this section is not to find the ‘best’ evaluation metric: each has advantages and disadvantages, the importance of which will vary depending on the nature of the research. A final point is that the use of a variety of techniques in this paper is of interest in itself. The high degree of correspondence between them (presented in ), suggests that researchers need only to present one or two metrics (but should perhaps try more, for corroboration) to establish relative levels of goodness of fit between different model configurations.

## Scatter plots and Pearson’s coefficient of correlation

A scatter plot of cell counts for each category for the original and simulated variables is a basic but very useful preliminary diagnostic tool in spatial microsimulation . In addition marking the points, depending on the variable and category which they represent, can help identify which variables are particularly problematic, aiding the process of improving faulty code (sometimes referred to as ‘debugging’).

In these scatter plots each data point represents a single zone-category combination (e.g. 16-25 year old female in zone 001), with the x axis value corresponding to the number of individuals of that description in the input constraint table. The y value corresponds to the aggregated count of individuals in the same category in the aggregated (wide) representation of the spatial microdata output from the model. This stage can be conducted either before or after *integerisation* (more on this below). As taught in basic statistics courses, the closer the points to the 1:1 (45 degree, assuming the axes have the same scale) line, the better the fit.

Pearson’s coefficient of correlation (henceforth Pearson’s ) is a quantitative indicator of how much deviation there is from this ideal 1:1 line. It varies from 0 to 1, and in this context reveals how closely the simulated values fit the actual (census) data. An value of 1 represents a perfect fit; an value close to zero suggests no correspondence between then constraints and simulated outputs (i.e. that the model has failed). We would expect to see very high coefficient of determination for internal validation (the constraint) and strong positive correlation between target variables that have been estimated and external datasets (e.g. income).

## Standard error around identity (SEI)

here SEI is the standard error around identity, y are the estimated values for each area, y rel are the reliable estimates for each area from a census or other data source and y rel is the mean estimate for all areas where reliable data are available. This estimate has been used in validation by both Ballas and Tanton (see Ballas et al. 2007; Tanton et al. 2011).

## Total Absolute and Standardized Error (TAE and SAE)

The SAE addresses this issue by using the population size as the denominator, but generally, authors seem to use total population, rather than the population for that categorization of the variable, which may be deemed to understate the size of the errors. Also, this measure does not provide any information on whether any differences are statistically significant (Edwards and Tanton, 2013).

## Z scores

Here, the evaluation mainly takes place at the cell level (for a single constraint in a single area). However, there should also be attention for the overall picture like spatial concentration.

# Input data

Three input datasets form the basis of the scenarios tested in this paper, ‘simple’, ‘small-area’ and ‘sheffield’. Each consists of a set of aggregate constraints and corresponding individual-level survey tables. The latter are anonymous and have been ‘scrambled’ to enable their publication on the internet. This enables the results to be replicated by others, without breaching the user licenses of the data. To this end, all the input data (and code) used to generate the results reported in the paper are available online on the code-sharing site GitHub. The simplest scenario consists of 6 zones which must be populated by a sample survey dataset of only five individuals, constrained by two variables (age and sex). This scenario is purely hypothetical and is used to represent the IPF procedure and the tests in the simplest possible terms. Its small size allows the entirety of the input data to be viewed in a couple of tables that fit on a single page (see and ). As with all data used in this paper, these tables can be viewed and downloaded online.

[htbp]

rrl & & Sex  
1 & 59 & m  
2 & 54 & m  
3 & 35 & m  
4 & 73 & f  
5 & 49 & f

[t1]

[htbp]

rrrrr & &  
 & & & &  
1 & 8 & 4 & 6 & 6  
2 & 2 & 8 & 4 & 6  
3 & 7 & 4 & 3 & 8  
4 & 5 & 4 & 7 & 2  
5 & 7 & 3 & 6 & 4  
6 & 5 & 3 & 2 & 6

[t2]

The next smallest input dataset, and the most commonly used scenario in the paper, is ‘small-area’. The area constraints consist of 24 Output Areas (the smallest administrative geographical unit in the UK), with an average population of 184 people aged over 16. Three files describe the number of hours worked, the marital status and the tenancy of these individuals (although tenancy in fact counts houses, not people): the files can be accessed from the hyperlinks embedded in digital versions of the previous sentence. A sample of these constraint variables is presented in below.

[htbp]

rrrrr

[t3]

The individual-level table that compliments these geographically aggregated count constraints is taken from Understanding Society wave 1. These have been randomized to ensure anonymity but they are still roughly representative of real data. The table consists of 1768 rows of data and is stored in R’s own space-efficient filetype (see “ind.RData”, which must be opened in R to be viewed properly). As shown in below, this consists of the same 3 constraint variables as provided in but at aggregate level, in addition to one ‘target variable’: monthly pre-tax income in pounds.

[htbp]

rrrrr

[t4]

... More here on loading-up this data.

# Method: model experiments

## Baseline scenarios

In order to perform a wide range of model experiments, in which only one factor is altered at a time, it was necessary to clearly define baseline scenarios. These are model runs with sensible defaults, which can be referred back to as a reference point and re-used at any point in the future. The baseline ‘simple’ scenario consists of 3 iterations, leading to a total of 6 constraint-iteration permutations. The results of simplest baseline scenario are shown in . The results show that the model fit improves after every iteration and every constraint in every measure of model fit (except ‘p5’, which reaches zero in the second iteration) — continues to improve, even though the printed value, to 6 decimal places, is indistinguishable from 1 after two complete iterations.

[htbp]

rrrrr

[tblresults1]

The baseline ‘small-area’ scenario consists of three iterations of IPF, in which all three constraints were used. The results for all 9 constraint-iteration permutations of this baseline scenario are illustrated in and in the online ‘measures’ table. These results show that IPF successfully brings the simulated cell counts in-line with the census after each constraint is applied and that the fit improves with each iteration (only two iterations are shown here, as the fit improves very little beyond two, and to avoid clutter).

image

[fipfinac]

The baseline ‘Sheffield’ scenario consists of the same basic set-up, but with 4 constraints. As before, the results show convergence towards better model fit, although the fit does not always improve from one constraint to the next (see the online ‘measures’ table).

## Constraints and iterations

These were the simplest model experiments, requiring only the reporting of iterations that had already taken place, and re-arrangement of the IPF code such that constraints were applied in a different order than before. The re-ordered code for the ‘small-area’ scenario, for example can be seen in the script ‘etsim-reordered’ online.

## Initial weights

The impact of initial weights was tested in two ways: first, the impact of changing a sample of one or more initial weight values by a set amount was explored. Second, the impact of of the size of the initial weights was tested, by running the same model experiment several times, but with incrementally larger initial weights applied to a pre-determined set of individuals.

## Empty cells

Empty cells refer to cross-tabulations of individual-level attributes that are not represented by any single individual in the survey input data. An example of this would be if no female in the 16 to 49 year age bracket were present in . The method used to find-out if empty cells exist is to first select only individuals with unique characteristics (in the narrow terms of the constraint variables — of course every survey individual is unique). The column totals of the categorical (binary, 0 or 1) form of this dataset are compared: any categories with fewer than expected totals have missing cells.

Once the presence of empty cells is determined in the baseline scenarios, the next stage is to add or subtract empty cells by adding or subtracting individuals to the input data. After these changes were made, the baseline scenarios were run as before.

In the ‘simple’ scenario, this simply involves removing a single individual from the input data, which equates to deleting a single line of code (see ‘etsim-empty’). The question of which unique individual to remove is solved by random sampling.

## Integerisation

This was the next variable that was altered, following . In the simplest scenario, the initial weight of a single individual was altered repeatedly, in a controlled manner. The code enabling this model experiment is contained in the folder ‘small-area-weights’ (see ‘etsim.R’). The number of different initial weights can be controlled by the parameter ‘num.ws’ which is set to 5 as an arbitrary default. Initial weights are set to k, a vector of length ‘num.ws’ which are by default set as a function of ‘u’ which is in term defined as the sequence 1:num.ws:

k <- u/5 + 0.5 # initial weight of sample individuals for testing

In this case the resultant weights are set to 0.7, 0.9, 1.1, 1.3 and 1.5 for each model experiment. For factors of one million, for example, the above would be replaced by k <- u \* 10^6 yrm. To observe the impacts of altering the weights in this way the usual tests of model fit were conducted. In addition, the capacity of initial weights to influence an individual’s final weight/probability of selection was tested by tracing its weight into the future after each successive constraint and iteration. This enabled plotting original vs simulated weights under a range of conditions ().

# Results

## Baseline scenarios

As expected from above, the correlation of the baseline scenario improves over time, with each constraint and from one iteration to the next. Through the course of the first iteration, the correlation improves from 0.67 (the correlation between the frequencies of variables in the survey dataset and their frequencies in the census dataset, for each zone) to 0.9981. As illustrated in , the marginal improvement reduces from one iteration to the next: at the end of iteration 2, the correlation has improved to 0.999978.

image

[fnconscor]

## Constraints and iterations

As illustrated above, the number of iterations required to reach a very close fit between the census and simulated data is very low, with rapid reductions in the marginal accuracy return on additional computing power. After four iterations, the result in R is indistinguishable from 1, with its default accuracy of 7 decimal places. The decay in the rate of improvement with additional iterations and constraints was found to super-exponential.

## Initial weights

It was found that initial weights had little impact on the overall simulation. Taking the simplest case, a random individual (id = 228) was selected from the ‘small area’ data and his/her initial weight was set to 2 (instead of the default of 1). The impact of this change on the individual’s simulated weight decayed rapidly, tending to zero after only 5 iterations in the small-area scenario. It was found that altered initial weights affect the simulated weights of all other individuals; this is illustrated in , which shows the impact on individual 228 and a randomly selected individual whose initial weight was not altered, for all areas.

image

[ficweight]

shows that changing initial weights for individuals has a limited impact on the results that tends to zero from one iteration to the next. After iteration one the impact of a 100% increase in the initial weight of individual 228 had reduced to 0.3%; after 3 iterations the impact was negligible. The same pattern can be observed in the control individual 570, although the impact is smaller and in the reverse direction. The same pattern can be observed for other individuals, by altering the ‘analysis.min’.

To analyse the impacts of changing the initial weights of multiple individuals, the same experiment was conducted, but this time the initial weights were applied to the first five individuals. The results are illustrated in , which shows the relationship between initial and final weights for the first two individuals and another two individuals (randomly selected to show the impact of altering weights on other individual weights).

image

[finweight]

shows that the vast majority of the impact of altered starting weights occurs in iteration-constraint combinations 1.1 and 1.2, before a complete iteration has even occurred. After that, the initial weights continue to have an influence, of a magnitude that cannot be seen. For individual 1, the impact of a 53% difference in initial weights (from 0.7 to 1.5) shrinks to 2.0% by iteration 1.3 and to 1.1% by iteration 2.3.

These results suggest that initial weights have a limited impact on the results of IPF, that declines rapidly with each iteration.

## Empty cells

## Integerisation

# Conclusions

1. In the interaction between the age and sex constraints are assumed to be known. (Their equivalent of contains data for every cell, not question marks.) This results in IPF converging instantly. However, in Census data, such cross-tabulation is often absent, and IPF must converge over multiple constraints and iterations. This latter scenario is assumed in the worked example below. Other worked examples of the principles are provided in (for entropy maximisation), and (using the proprietary statistical software SPSS). [↑](#footnote-ref-28)
2. This tutorial is available from Rpubs, a site dedicated to publishing R analyses that are reproducible. It uses the RMarkdown mark-up language, which enables R code to be run and presented within documents. See http://rpubs.com/RobinLovelace/5089 [fnrpub] . [↑](#footnote-ref-30)