SBAM: An Algorithm for Pair Matching

Peter Stephensen

February 2012

DREAM Working Paper 2012:1

DREAM, Danish Rational Economic Agents Model. Amaliegade 44, 1256 Copenhagen K
www.dreammodel.dk

SBAM: An Algorithm for Pair Matching*

Peter Stephensen, DREAM†

February 9, 2012

Abstract

This paper introduces a new algorithm for pair matching. The method is called SBAM (Sparse Biproportionate Adjustment Matching) and can be characterized as either *cross-entropy minimizing* or *matrix balancing.* The method is demonstrated in the context of a new Danish microsimulation model.

1 Introduction

Dynamic microsimulation finds increasing use in demographic and socioeconomic forecasting. A big advantage of the microsimulation approach is that it makes it possible to analyze family structure. In traditional population projections the goal is usually to forecast the population by age, gender and a few other characteristics (such as origin and/or geographical region). Introducing family structure into this approach is problematic, mainly because the size of the model increases tremendously when forecasting the population by family/household characteristics. These characteristics can alternatively be analyzed in a microsimulation model without loosing control over the size of the model.

*Financial support from the Knowledge Centre for Housing Economics, Realdania is gratefully acknowledged.

†The DREAM model. Amaliegade 44, 1256 Copenhagen K. www.dreammodel.dk

Modelling family structure demands some extra features compared to the traditional approach. To get the family composition right, two things are necessesary: Parity[1] must included in the fertility determination, and pair matching must be modelled. This paper deals with the latter subject and introduces a new algorithm for pair matching.

Usually two distinct methodologies are mentioned when it comes to matching: the stable marriage approach (previously used in the CORSIM and DYNACANE models) and the stochastic approach (used in the DYNASIM model). The method decribed in this paper cannot be categorized as either. The method is called SBAM (Sparse Biproportionate Adjustment Matching) and can be characterized as either *cross-entropy minimizing* or *matrix balancing* (defined below). The SBAM method is based on historical observations of pair mathings from one or more years, distributed on a set of types (age, gender, education, geographical region ect.). In a forecasted year, it is assumed that a *matching pool* of individuals has been formed. If the individuals in this pool are distributed on types as in the historical data, the matching problem is easy to solve: We simply distribute the pairs as in the historical data. If this is not the case (which it typically is not), the pairs must be distributed in a new way. A criteria could be to distribute the pairs such that the distribution deviates as little as possible from the historical distribution. This can be interpreted as a so-called matrix balancing problem (Schneider & Zenios, 1990): Change the original data (defined as a matrix) such that the row and column sums are given by predefined values. A number of solutions exist to this kind of problem. One such solution is called biproportionate adjustment (or RAS adjustment). This method has two advantageous properties: It is relatively easy to implement, and it has a nice interpretation. Using biproportionate adjustment, the outcome can be interpreted as the result of a so-called cross-entropy minimization problem (McDougall, 1999). The matching changes the distribution of pairs relative to the original distribution, so that the information loss is as small as possible. The information loss is defined by Shannon's Information Theory

[1]The number of children a woman already has.

(Shannon, 1948).

Section 2 decribes the methodology of the matching method. Section 3 shows an application of SBAM in a new Danish microsimulation model.

2 Methodology

There are assumed to be N individuals to be matched into pairs[2]. The individuals are divided into T types:

$$N = \sum_{j=1}^{T} N_j$$

A type could for example be defined on the basis of gender, age, origin and education. The number T can therefore be expected to be rather large[3].

The aim is to find real numbers $x_{i,j}$ $(i = 1, ..., T,\ j = 1, ..., T)$ such that

$$\sum_{j=1}^{T} x_{it} = N_t,\ t = 1, ..., T \tag{1}$$

and

$$x_{ij} = x_{ji},\ i = 1, ..., T,\ j = 1, ..., T \tag{2}$$

The matching is defined by (1). The variable x_{ij} indicates the number of individuals of type i that are paired with an individual of type j. If an individual of type i is paired with a person of type j, then the opposite is also the case: An individual of type j is paired with a person of type i. This gives rise to the symmetry assumption (2).

[2] N is assumed to be an even number.

[3] As an example, assume the types are defined on the basis of 2 genders, 50 ages (15-65), 5 education levels and 11 geographical regions. Then $T = 2 * 50 * 5 * 11 = 5.500$

2.1 Data

The algorithm is based on data from actual matchings. Let x_{ij}^0 be the number of individuals of type i that according to data is matched with an individual of type j. As mentioned above, the data set x_{ij}^0 is symmetric. This is ensured in the following way: When a pair of type (i, j) is added to data, it is done by following the procedure:

$$\begin{aligned} x_{ij}^0 &=: x_{ij}^0 + 1 \\ x_{ji}^0 &=: x_{ji}^0 + 1 \end{aligned}$$

where $=:$ is an algorithmic equal sign[4]. In the data set, individuals are distributed on T types:

$$N_t^0 = \sum_{i=1}^{T} x_{it}^0 = \sum_{j=1}^{T} x_{tj}^0 \tag{3}$$

and the total number of individuals is given by

$$N_0 = \sum_{j=1}^{T} N_j^0$$

It is advantageous to describe the problem in matrix notation. The data set x_{ij}^0 can be described as a $T \times T$ matrix, X^0. Define the vector

$$\vec{N}^0 = \left(N_1^0, ..., N_T^0\right)$$

According to (3), both the row and column sums of X^0 should be given by $\vec{N}^0$.

2.2 Biproportionate Adjustment

We are going to match N individuals, distributed on types according to $\vec{N} = (N_1, ..., N_T)$. We wish to find a $T \times T$ dimensional symmetric matrix X such that its row and column

[4] $x =: x + a$ means that x is increased with the value a.

sums add to $\vec{N}$. This should be done so that X deviates as little as possible from the original data X^0. In other words, we would like our matching X to reflect as much as possible of the matching information in the original (real world) matching X^0. This can be interpreted as a classical matrix balancing problem: *Given a rectangular matrix A, determine a matrix X that is close to A and satisfies a given set of linear restrictions on its entities* (Schneider & Zenios, 1990).

Algorithms for matrix balancing can be separated into two broad classes: scaling algorithms and optimization algorithms. Scaling algorithms multiply the rows and columns of the original matrix by positive constants until the matrix is balanced. Optimization algorithms minimize a penalty function that measures the deviation of a candidate balanced matrix from the original matrix. The balance conditions are constraints in the optimization model, so that the optimal solution is the balanced matrix closest to the original matrix.

We are going to use the scaling approach here. According to the biproportionate adjustment model (also called RAS adjustment), the balancing problem can be solved in the following iterative way: Start with the original matrix. Scale the rows such that the row sums are correct. Then scale the columns such that the column sums are correct. Repeat these two operations until a new stable matrix has emerged.

When using the optimization algorithms, it is obvious that the new matrix deviates as little as possible from the original matrix (that is part of the definition of the problem). This is less obvious when it comes to the scaling algorithms. In fact, it can be demonstrated that the biproportionate model is an entropy-theoretic model (McDougall, 1999). The new matrix can be characterized as the solution to a cross-entropy minimization model. Entropy should here be understood in an information theoretical context (Shannon, 1948). By using the biproportionate model we are actually minimizing the loss of information when changing from the type distribution $\vec{N}^0$ to $\vec{N}$.

2.3 Sparse algorithm

As mentioned above, the number of types T can be very large. Therefore, a $T \times T$ matrix can easily become so large that it gives rise to computational problems. As there at the same time often will be many zeros in the X^0 matrix, it will have obvious advantages to introduce a sparse matrix method. The method is implemented in C# and is based on so-called *linked lists*[5]. A $T \times T$ matrix can be represented by a *SBAMMatrix*. A SBAMMatrix is a C# object that essentially contains $2T$ linked lists: T linked lists for the rows and T linked lists for the columns. Each element in the linked list contains a pointer to data and a reference to the next element in the list. In this way, data is actually represented twice: as rows and as columns. The reason for this redundancy is that it makes biproportionate scaling much easier.

3 An application

The SBAM method has been used in the development of a new Danish microsimulation model. The purpose of the model is to forecast the evolution and composition of Danish households and their demand for housing. The model works with a full sample of the Danish population of approximately 5,5 million individuals and 2,5 million households divided into 11 (geographical?) regions. The model describes demography, education, socio-economic status and housing choice.

Each individual will in every period (every year) with a given probability be included in the so-called *matching pool*. This probability depends on the characteristics of the individual. For example, a young person that is single, will have a high probability, while an older person living in a relationship, will have a lower probability.

In this way, a matching pool containing approximately 120,000 individuals arises each

[5] A linked list is a data structure consisting of a group of nodes that together represent a sequence. Each node is composed of data and a reference (a link) to the next node in the sequence. This structure is memory space saving and allows for efficient insertion or removal of elements from any position in the sequence.

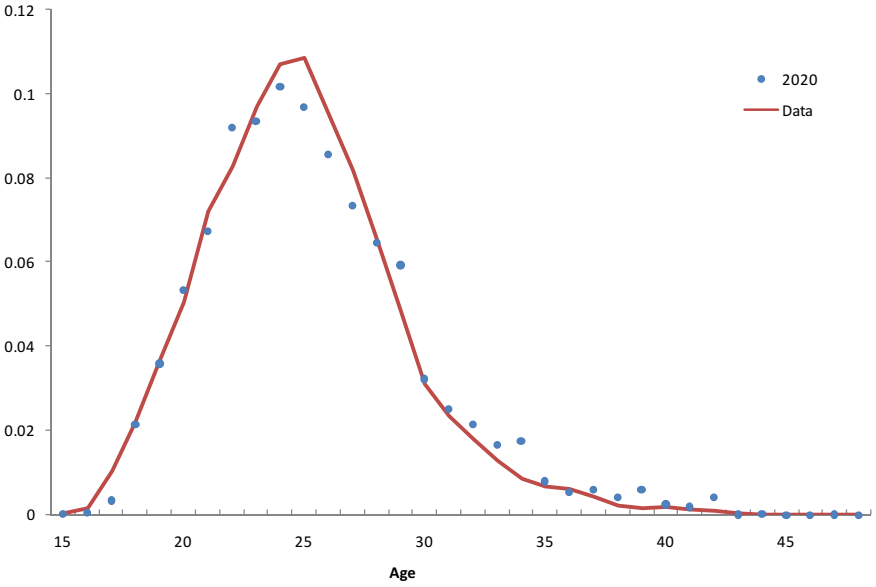
period. From this, the corresponding 60,000 pairs are formed. The SBAM algorithm is used for this. In the experiments reported in this paper, the individuals are divided into types on the basis of gender, age (15-65), 5 education levels and 11 regions. This results in 5,500 different types (=2*50*5*11). On a Windows-server (Intel Xeon CPU X5550, 2.67GHz), the matching takes approximately 20 seconds.

Figures 1-3 give examples of the results of the model in 2020. The figures show the distribution of newly formed pairs in the original data (from 2008) and in 2020. Figure 1 displays the age distribution of partners of 25-29 year old males. It is evident that SBAM is capable of generating an age distribution fairly consistent with data. The mean age of a partner is 25.0 in the data. In the forecast, the average is 25.4.

Figure 2 shows the educational distribution of partners for individuals with a vocational education. The SBAM algorithm finds it necessary to move the distribution slightly to ensure that the over-all matching is solved. In comparison to the original data, the proportions of partners with educational levels of "High school" and "Vocational" have thus fallen, while the proportions of "No education", "Medium" and "Long" have risen.

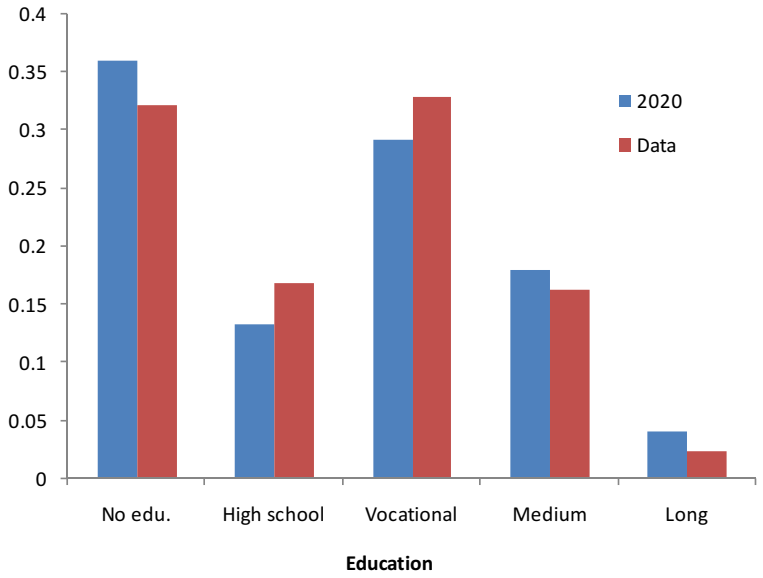
Lastly, Figure 3 displays the regional distribution of partners for individuals living in "Copenhagen, environs". The Copenhagen area is divided into two regions: "Copenhagen, environs" (7) and "Copenhagen, city" (6). It is seen that approximately 50 per cent of new partners also live in the environs of Copenhagen. In addition, Copenhagen city and North Sealand (8) account for a significant proportion of new partners. It is evident that the SBAM method produces a distribution that is fairly consistent with the original data.

Figure 1: Age distribution of partners. 25-29 year old males.



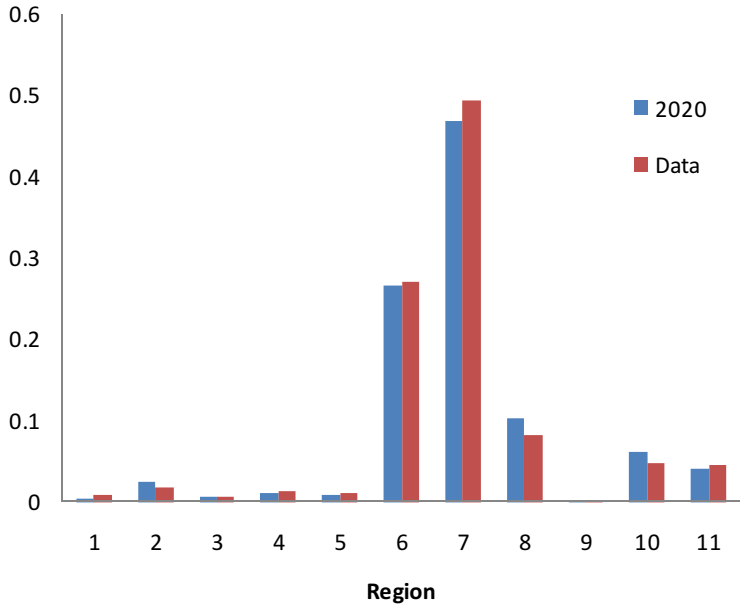
Source: Own calculations.

Figure 2: Educational distribution of partners. Vocational (?).



Source: Own calculations.

Figure 3: Regional distribution of partners. Copenhagen, environs.



Note: Copenhagen, city=6, Copenhagen, environs=7, North Sealand=8.
Source: Own calculations.

4 References

McDougall, Robert A. (1999) "Entropy Theory and RAS are Friends". GTAP Working Paper 5-14-1999

Schneider, Michael H. and Zenios, Stavros A. (1990) "A Comparative Study of Algorithms for Matrix Balancing". Operations Research, Vol. 38, No. 3 (May - Jun., 1990), pp. 439-455.

Shannon, C.E. (1948) "A mathematical theory of communication". Bell System Technical Journal, 27:379–423, 623–659.

Easther, Richard and Jan Vink. 2000. "A Stochastic Marriage Market for CORSIM." Strategic Forecasting Technical Paper.