An Unconstrained Optimization Approach to the Inconsistent Matrix Balancing Problem

Kiyoshi Yoneda *

Abstract This paper proposes and experiments with a new method to solve a variation of the matrix balancing problem such that the given marginals may be inconsistent and the matrix elements take real, possibly negative, values. The formulation is by the weighted least squares accessible to anyone familiar with elementary statistics.
Keywords: matrix balancing, weighted least squares, input-output analysis, origin-destination tables

1 Introduction

This paper proposes and experiments with a new method to solve a variant of the well-known matrix balancing problem:

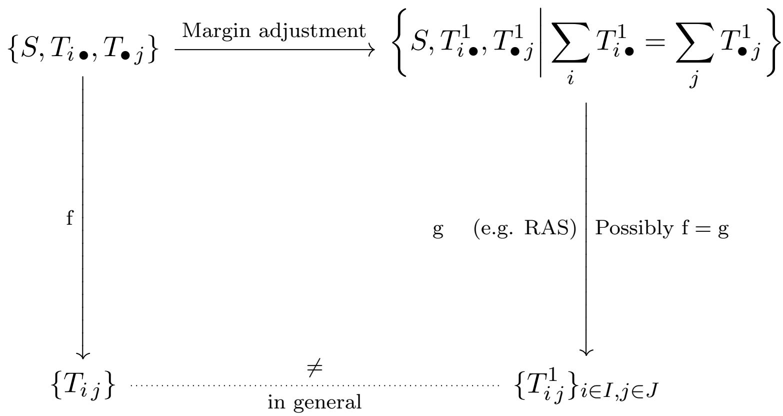
> **The inconsistent matrix balancing problem**
> is the *matrix balancing problem*, described in Appendix A, in which the row sums $\sum_j T_{i\bullet}$, the column sums $\sum_i T_{\bullet j}$, and the grand total $T_{\bullet\bullet}$ may differ.

If the marginals may be inconsistent then it necessarily implies that the marginal entries involve errors, which in turn implies that the solution T_{ij} should be allowed to add up to values close to but different from the given marginals. Thus a solution method for

*Faculty of Economics, Fukuoka University, Fukuoka, Japan

the inconsistent matrix balancing problems is capable of solving the consistent matrix balancing problems but may modify the marginal sums even when dealing with consistent problems.

A common practice in dealing with inconsistent matrix balancing problem is to preprocess the marginal sums to ensure consistency and then apply any algorithm based on constrained optimization pretending as if the marginals were accurate and consistent to begin with. The fallacy of this approach is obvious. Suppose f is a method permitting inconsistency and g a method that works when the problem is consistent. Then the result by f and the result obtained by applying g after forcing the marginals to appear consistent generally differ even when f = g:



A solution method applicable to the inconsistent matrix balancing problem, which is a candidate for f in the above diagram, is proposed in this paper based on *unconstrained* optimization. The idea is to modify the well-known constrained optimization formulation by Deming and Stephan [8, 4] into an unconstrained optimization by adding to the objective function the constraint violations multiplied by suitable weights. Ideally the weights should equal the Lagrangean multipliers in case the constraints are consistent.

The problem then reduces to the unconstrained optimization, the weighted least squares in this paper. The elements of T are assumed to be *real numbers that may be negative* with the usual data structure assumed in linear regression, i.e., independent normally distributed additive errors with different variances.

The motivation of the proposal has been that since the matrix balancing is a problem that pops up in various applications it seems beneficial to have a method available to anyone who understands the least squares. The major advantage of the method is in accessibility rather than in other requirements such as theoretical soundness, accuracy, or computational speed, even though those properties turn out to be defendable.

The remainder of this paper is organized as follows. Section 2 explains the proposed method, in which the method of weight assignment is the essential contribution. Section 3 experiments with the 2006 and the 2007 world trade data to illustrate the performance of the method. Section 5 concludes the paper with remarks on the desirable properties of the method mentioned in the previous paragraph. Appendix A outlines the standard matrix balancing problem and its variations. Appendix B is a cheat sheet for the method of weighted least squares. Appendix C contains programs in R [20] language used to process the example in Sections 3 and 4.

2 Formulation

2.1 Equations

The equations desired to satisfy are

$$\forall i, j \qquad T_{i\,j} = S_{i\,j} \tag{1}$$

$$\forall i \quad \sum_j T_{i\,j} = T_{i\,\bullet} \tag{2}$$

$$\forall j \quad \sum_i T_{i\,j} = T_{\bullet\,j} \tag{3}$$

$$\sum_{i\,j} T_{i\,j} = T_{\bullet\bullet} \tag{4}$$

If the system were to be assumed consistent, then (2), (3), and (4) would be considered constraints and (1) the equations to satisfy as closely as possible. In this paper all the equations (1), (2), (3), and (4) are to be satisfied as closely as possible rather than exactly. Since there are more equations than variables possibly contradicting each other, in general there is no solution that exactly satisfies all the equations. However, they may be satisfied approximately by various methods such as the least squares.

Now the equations (1) are much less credible than the other equations (2), (3), and (4). Hence the least squares should be weighted, with (1) bearing lighter weights than the rest of the equations. Thus the problem reduces to the assignment of weights. The method of weighted least squares is outlined in Appendix B for convenience.

2.2 Weights

Equations (2), (3), and (4) may be weighted by the method in [24] according to the accuracy of $T_{i\,\bullet}$, $T_{\bullet\,j}$, and $T_{\bullet\bullet}$, i.e.,

$$w_i = 12 \times 10^{-2k} \tag{5}$$

where k is the decimal digit up to which the numbers are significant. For instance, if only the integer parts of the numbers are significant, then $k = 0$; if only one digit may be trusted to the right of the decimal point, then $k = -1$; if only the "kilo" part is useful then $k = 3$, etc.

The idea behind (5) described in [24] is as follows. Let number n be a number whose last significant digit is k. Then the true value of the number may be considered to be uniformly distributed in the interval $[n - 4 \times 10^{-k-1}, n + 5 \times 10^{-k-1}[$. The normal distribution that best approximates this uniform distribution, measured in terms of Kullback-Leibler information, has mean n and variance $\frac{1}{12} \times 10^{2k}$. Hence it is reasonable to approximate n, the last significant digit of which is k, by the normal distribution with mean n and variance $\frac{1}{12} \times 10^{2k}$. Then the weight to be assigned, viz. the inverse of the variance, is 12×10^{-2k}.

The weights for (1) should be much smaller than the weights for the other equations but large enough to ensure the solution stability: a small perturbation in b should not drastically change the value of the solution. The weights for (1) do not greatly affect the solution so long as the weights for (1) are orders of magnitude smaller than the weights for equations (2), (3), and (4), as will be illustrated in Section 4. Informally, information the marginals provide should be amply respected but supplemented by S just enough to avoid instability.

3 Experiments with world trade data

The example adopted for numerical experiments is:

> **The world trade problem**
> Given the 2006 trade table among the seven geographical regions of the world, fill the 2007 trade table whose row and column sums are known but the individual entries are unknown.

Actually, both 2006 and 2007 tables are available [18, 19] and hence the performance of the proposed method may be evaluated against the factual 2007 table. The World is divided into the following seven geopolitical regions.

Index	*In extenso*
`N.Am`	North America
`SC.Am`	South and Central America
`Europe`	
`CIS`	Commonwealth of Independent States
`Africa`	
`M.East`	Middle East
`Asia`	Asia-Pacific

This is a convenient size to see the effect of the information provided by the marginal sums of T. If the table were too big, most information would be supplied by the priors (1) rather than by the marginal sums (2), (3), and (4); it would be the other way around if the table were too small.

All amounts in this example's tables are in billion (10^9) US dollars. The "World" row entries of Table 2 have no decimal points and digits below because the original table in [19] does not include them. Note that neither $\sum_i T_{i\bullet}$ nor $\sum_j T_{\bullet j}$ exactly equals to $T_{\bullet\bullet}$, meaning that the problem is inconsistent. Computer programs in R [20] used to process the problem are listed in Appendix C.

Table 1 is the 2006 data, from which S is calculated. Table 2 is the table T to fill in, where $\forall i, j\ T_{ij} = NA$ (Not Available). Table 3 shows S, which is obtained by inflating 1 proportionally by the factor

$$\frac{\text{Total for 2007}}{\text{Total for 2006}} = \frac{13619}{11783.0} = 1.1558$$

where the numerator and the denominator are the "World-World" entries of Tables 2 and 1, respectively.

Table 4 is for the weights w. The weights for the marginals have been set by (5) with $k = -1$ for $T_{\bullet j}$, $\forall j$, and with $k = 0$ for $T_{i\bullet}$, $\forall i$, and $T_{\bullet\bullet}$. The weights for S have been set by assuming a subjective and exogenous 30% standard deviation to the entry values of S_{ik}. For instance, since $S_{\text{CIS M.East}} = 15.4$, its 30% is 4.62, and the weight is $1/4.62^2 = 4\text{e-}3$. The numbers were truncated rather than rounded to one digit since only the exponents really matter.

Table 5 is the solution, T filled in by the method proposed in Section 2. Note that the marginals have been modified from Table 2. This is a consequence of assuming that the marginals involve errors.

Table 6 is the factual 2007 world trade statistics. Comparing Figure 5 against Figure 6 evaluates how useful the proposed method is. Figure 1 illustrates the original error contained in S, obtained by subtracting Table 6 from Table 3. Figure 2 illustrates the error in T, obtained by subtracting Table 6 from Table 5. Figure 2 shows a higher concentration around zero compared to Figure 1. The improvement from Figure 1 to Figure 2 is due to the information the marginal sums in Table 2 contain, extracted by the method in Section 2.

For the sake of completeness, Table 7 and Figure 3 demonstrate that the purely proportional allocation

$$T_{ij} = \frac{T_{i\bullet} T_{\bullet j}}{T_{\bullet\bullet}}$$

neglecting S performs poorly.

4 Sensitivity to the weights

This section intends to illustrate that the solution is not sensitive to the weights.

Table 8, corresponding to Table 4 in Section 3 sets the weights of the marginals (2), (3), and (4) to one. The weights for (1) are

Table 1: World trade 2006 statistics

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	2355.0	378.0	5118.0	290.0	283.0	381.0	2839.0	11783.0
N.Am	905.3	107.3	279.3	8.3	21.7	42.1	314.1	1678.3
SC.Am	135.0	111.5	86.4	6.1	11.3	7.9	61.8	429.9
Europe	430.3	66.6	3651.5	141.6	120.2	128.9	366.4	4963.0
CIS	24.2	7.6	246.5	80.3	5.7	13.3	45.6	425.6
Africa	79.8	11.3	148.1	1.4	32.8	6.3	72.6	363.3
M.East	72.3	4.4	102.8	3.0	20.9	71.6	339.6	645.5
Asia	708.3	69.5	603.8	49.7	69.9	111.4	1638.5	3277.8

Table 2: World trade 2007 marginals: $T_{i\bullet}, T_{\bullet,j}, T_{\bullet\bullet}$

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	2517	451	5956	397	355	483	3294	13619
N.Am	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	1853.5
SC.Am	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	499.2
Europe	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	5772.2
CIS	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	510.3
Africa	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	424.1
M.East	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	759.9
Asia	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	3799.7

Table 3: World trade 2006 extrapolated to 2007: S_{ij}

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	2722.0	436.9	5915.5	335.2	327.1	440.4	3281.4	13619.0
N.Am	1046.4	124.0	322.8	9.6	25.1	48.7	363.0	1939.8
SC.Am	156.0	128.9	99.9	7.1	13.1	9.1	71.4	496.9
Europe	497.3	77.0	4220.5	163.7	138.9	149.0	423.5	5736.3
CIS	28.0	8.8	284.9	92.8	6.6	15.4	52.7	491.9
Africa	92.2	13.1	171.2	1.6	37.9	7.3	83.9	419.9
M.East	83.6	5.1	118.8	3.5	24.2	82.8	392.5	746.1
Asia	818.7	80.3	697.9	57.4	80.8	128.8	1893.8	3788.5

Table 4: Weights: w

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	1e3	1e3	1e3	1e3	1e3	1e3	1e3	1e3
N.Am	9e-7	7e-5	1e-5	1e-2	2e-3	4e-4	8e-6	1e3
SC.Am	4e-5	6e-5	1e-4	2e-2	6e-3	1e-2	2e-4	1e3
Europe	4e-6	2e-4	6e-8	4e-5	5e-5	5e-5	6e-6	1e3
CIS	1e-3	1e-2	1e-5	1e-4	2e-2	4e-3	4e-4	1e3
Africa	1e-4	6e-3	3e-5	4e-1	7e-4	2e-2	1e-4	1e3
M.East	1e-4	4e-2	7e-5	8e-2	2e-3	1e-4	6e-6	1e3
Asia	1e-6	2e-4	2e-6	3e-4	2e-4	6e-5	3e-7	1e3

Table 5: World trade 2007 filled in with prior and marginals: T_{ij}

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	2538.2	472.0	5977.0	418.1	376.1	504.1	3315.1	13600.6
N.Am	941.0	136.1	326.0	9.8	26.2	51.7	360.2	1851.0
SC.Am	153.6	141.8	100.1	7.2	13.4	9.2	71.3	496.6
Europe	466.3	81.4	4238.9	220.4	171.6	176.7	414.3	5769.6
CIS	27.9	8.8	285.0	111.0	6.7	15.7	52.6	507.7
Africa	93.6	13.2	179.4	1.6	40.7	7.4	85.5	421.4
M.East	84.4	5.1	122.4	3.5	25.3	93.0	423.5	757.2
Asia	771.4	85.6	725.2	64.6	92.2	150.4	1907.7	3797.1

Table 6: World trade 2007 statistics: true T

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	2517	451	5956	397	355	483	3294	13619
N.Am	951.2	130.7	328.7	12.4	27.3	50.1	352.1	1853.5
SC.Am	151.3	122.0	105.6	6.4	13.7	9.1	80.2	499.2
Europe	458.5	80.4	4243.6	189.0	147.7	152.9	433.7	5772.2
CIS	23.6	6.3	287.5	103.2	6.9	16.2	59.6	510.3
Africa	91.9	14.6	167.5	0.9	40.5	10.5	80.9	424.1
M.East	83.9	4.4	108.3	4.8	27.5	93.4	397.3	759.9
Asia	756.4	92.3	714.6	79.8	91.4	150.4	1889.8	3799.7

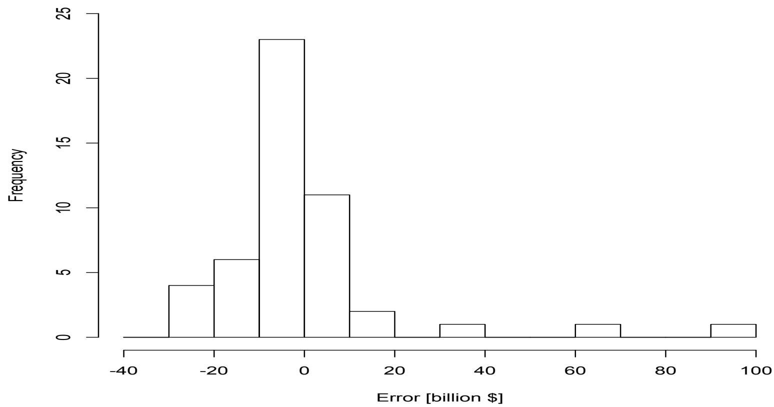


Figure 1: Error distribution for prior S only: no marginals

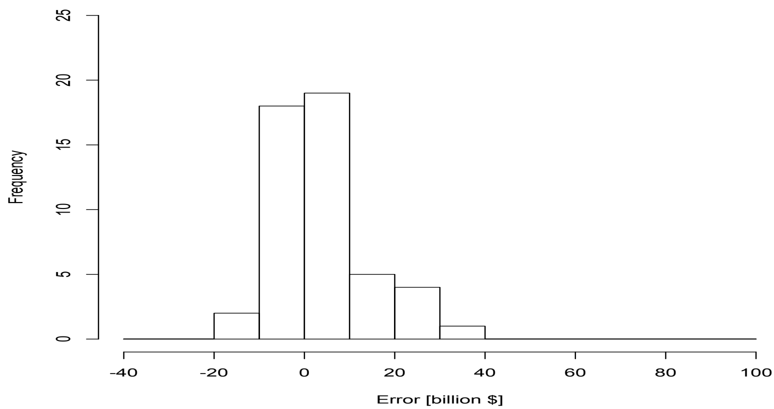


Figure 2: Error distribution for T: with prior S and marginals

Table 7: T without S: marginals only, uniform prior

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	2538.2	472.0	5977.0	418.1	376.1	504.1	3315.1	13600.6
World	1916.9	490.9	5668.7	486.1	415.0	737.3	3743.8	13458.7
N.Am	387.7	99.3	1146.5	98.3	83.9	149.1	757.2	2722.0
SC.Am	62.2	15.9	184.0	15.8	13.5	23.9	121.5	436.8
Europe	842.6	215.8	2491.6	213.7	182.4	324.1	1645.6	5915.8
CIS	47.7	12.2	141.2	12.1	10.3	18.4	93.2	335.1
Africa	46.6	11.9	137.8	11.8	10.1	17.9	91.0	327.1
M.East	62.7	16.1	185.5	15.9	13.6	24.1	122.5	440.4
Asia	467.4	119.7	1382.1	118.5	101.2	179.8	912.8	3281.5

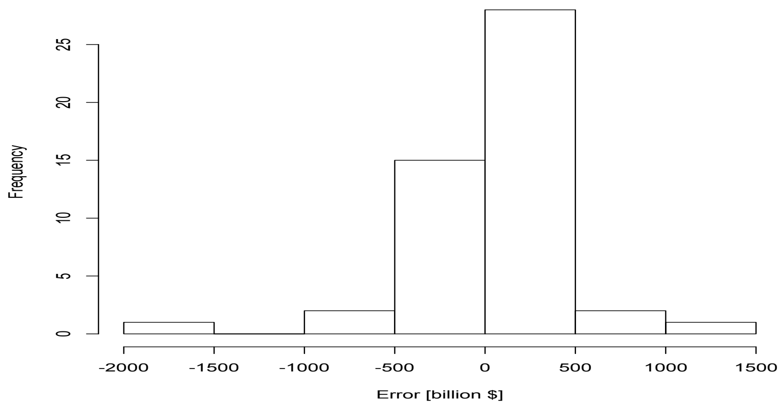


Figure 3: Error distribution for T without S: marginals only, uniform prior

set to

$$w_{ij} := \frac{1}{S_{ij}^2}$$

according to the original suggestion in [8]. Note that in case the table entries of S are nonzero natural numbers following Poisson distributions these weights are justifiable as the maximum likelihood estimates of the inverse variances. The weights are not justifiable if the assumed data structure (11) holds. However, the two tables 4 and 8 do not differ greatly.

Figure 4 is the result obtained by setting the above weights. Comparing this to Figure 2 no difference in performance is noticeable, although more extensive experiments would be necessary for a quantitative statement.

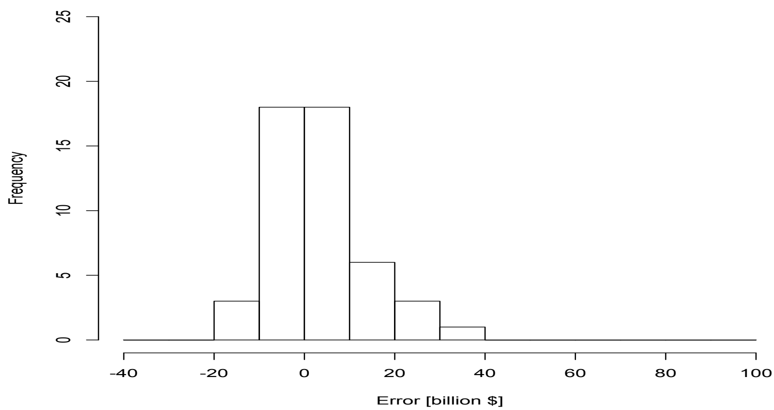


Figure 4: Error distribution for T: with Deming-Stephan weights

Table 8: Deming-Stephan Weights: w

OD	N.Am	SC.Am	Europe	CIS	Africa	M.East	Asia	World
World	1	1	1	1	1	1	1	1
N.Am	7e-5	1e-5	1e-2	2e-3	4e-4	8e-6	3e-7	1
SC.Am	6e-5	1e-4	2e-2	6e-3	1e-2	2e-4	4e-6	1
Europe	2e-4	6e-8	4e-5	5e-5	5e-5	6e-6	3e-8	1
CIS	1e-2	1e-5	1e-4	2e-2	4e-3	4e-4	4e-6	1
Africa	6e-3	3e-5	4e-1	7e-4	2e-2	1e-4	6e-6	1
M.East	4e-2	7e-5	8e-2	2e-3	1e-4	6e-6	2e-6	1
Asia	2e-4	2e-6	3e-4	2e-4	6e-5	3e-7	7e-8	1

5 Concluding remarks

The major advantage of the proposed method was stated, in Section 1, to be in accessibility to anyone who knows the least squares, such as beginning students. Additionally, other desirable properties are also present. The theoretical soundness is clear so long as the marginals contain error and the data structure (11) holds, an important requirement being that the table entries permit negative values. The accuracy is ensured by the Gauss-Markov theorem. Since the least squares is one of the best explored areas in numerical computation [14, 3], its efficiency is guaranteed.

Since in many important applications the table entries are assumed nonnegative, extensions in that direction would be useful. The most obvious way to accomplish this is by adopting the log-linear model.

References

[1] Jens Henrik Badsberg. Efficient methods for estimation in log-linear models. In *Bulletin of the International Statistical Institute*, 1999.

[2] P. Bickel, Y. Ritov, and J. Wellner. Efficient estimation of linear functionals of a probability measure p with known marginal distributions. *Annals of Statistics*, 19:1316–1346, 1991.

[3] A. Björck. *Numerical Methods for Least Squares Problems.* Society for Industrial and Applied Mathematics, 1996.

[4] Barbara A. Carothers. Matrix balancing – a comparative study. Master's thesis, Youngstown State University, July 2000.

[5] Raymond J. Carroll and David Ruppert. *Transformation and Weighting in Regression.* Chapman & Hall, 1988.

[6] Yair Censor and Stavros A. Zenios. Interval-constrained matrix balancing. *Linear Algebra Appl.*, 150:393–421, 1991.

[7] J.N. Darroch and D Ratcliff. Generalized iterative scaling for log-linear models. *Annals of Mathematical Statistics*, 43:1470–1480, 1972.

[8] Edwards W. Deming and Frederick F. Stephan. On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. *Annals of Mathematical Statistics*, 11(4):427–444, 1940.

[9] B. C. Eaves, A. J. Hoffman, U. G. Rothblum, and H. Schneider. Line-sum-symmetric scaling of square nonnegative matrices. *Mathematical Programming Study*, 25:124–141, 1985.

[10] Koichi Genma, Yutaka Kato, and Kazuyuki Sekitani. Matrix balancing problem and binary AHP. *Journal of the Operations Research Society of Japan*, pages 515–539, 2007.

[11] S. Haberman. Adjustment by minimum discriminant information. *Annals of Statistics*, 12:971–988, 1984.

[12] C. Ireland and S. Kullback. Contingency tables with given marginals. *Biometrika*, 55:179–188, 1968.

[13] Philip A. Knight and Daniel Ruiz. A fast algorithm for matrix balancing. In *Dagstuhl Seminar Proceedings on Web Information Retrieval and Linear Algebra Algorithms*, 2007.

[14] Charles L. Lawson and Richard J. Hanson. *Solving Least Squares Problems.* Society for Industrial and Applied Mathematics, 1987.

[15] Robert A. McDougall. Entropy theory and RAS are friends. Working paper in agricultural economics, Center for Global Trade Analysis, May 1999.

[16] Douglas J. Miller and Wei-han Liu. On the recovery of joint distributions from limited information. *Journal of Econometrics*, 107:259–274, 2002.

[17] Clyde L. Monma and T. Carpenter. Variations of matrix balancing for telecommunication demand forecasts. Technical report, Telcordia, 1997.

[18] World Trade Organization. International trade statistics 2007. Technical report, World Trade Organization, 2007.

[19] World Trade Organization. International trade statistics 2008. Technical report, World Trade Organization, 2008.

[20] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0.

[21] Michael H. Schneider and Stravros A. Zenios. A comparative study of algorithms for matrix balancing. *Operations Research*, 38:439–455, 1989.

[22] Thomas Wiedmann, Richard Wood, Manfred Lenzen, Rocky Harris, Dabo Guan, and Jan Minx. Application of a novel matrix balancing approach to the estimation of UK input-output tables. In *Proc. Sixteenth International Input-Output Conference*, July 2007.

[23] Kiyoshi Yoneda. Integer estimation of origin-destination tables. *Trans. Institute of Electric Engineers of Japan*, 114-C(4):483–490, 1994.

[24] Kiyoshi Yoneda. Optimal number of digits to report. *Journal of the Operations Research Society of Japan*, 39(3):428–434, September 1996.

[25] Kiyoshi Yoneda. Elevator trip distribution for inconsistent passenger input-output data. *Decision Making in Manufacturing and Services*, 1(1–2):175–190, December 2007.

APPENDIXES

A Matrix balancing problems

This section describes the standard matrix balancing problem and its variations.

The matrix balancing problem
Let I and J be the two finite sets and $|I|$ and $|J|$ their respective sizes. An $|I|$ by $|J|$ table

$$T = \begin{bmatrix} & \vdots & \\ \cdots & T_{ij} & \cdots \\ & \vdots & \end{bmatrix}$$

is presented as empty

$$= \begin{bmatrix} & \vdots & \\ \cdots & NA & \cdots \\ & \vdots & \end{bmatrix},$$

where

$$NA := \text{Not Available},$$

but its row sums

$$\begin{bmatrix} \vdots \\ T_{i\bullet} \\ \vdots \end{bmatrix} := \begin{bmatrix} \vdots \\ \sum_j T_{ij} \\ \vdots \end{bmatrix} \tag{6}$$

and column sums

$$\left[\cdots\ T_{\bullet j}\ \cdots\right] := \left[\cdots\ \sum_i T_{i\,j}\ \cdots\right] \tag{7}$$

are supplied with. Determine the entries $T_{i\,j}$ for $\forall(i,j) \in I \times J$ so that T is as *close* as possible to a given table S of the same size.

Since as early as the 1930s this problem appeared in various fields of applied mathematics such as contingency tables in statistics, Markov chains in probability, input-output (IO) analysis in economics, and orgin-destination (OD) tables in transportation. The problem comes under different titles such as the matrix balancing problem, data balancing, matrix scaling, matrix filling, trip distribution, joint distribution recovery, disaggregation, and the tables with given marginals, to name only some. See for instance [15] or [16] for history and references. Exact specification of the problem has many variants [17]:

- The table entries take only nonnegative values.
- A table entry is nonzero $T_{i\,j} \neq 0$ only if $S_{i\,j} \neq 0$ [6].
- The table entries are whole numbers [23].
- Some of the table entries may be known [25].
- The table is assumed symmetric and hence square [9, 13]. Rather than specifying the row and the column sums, each row sum is required to equal the corresponding column sum.

The problem has many variants in mathematical formulation as well since the solution depends on the notion of closeness of T to S which may differ among applications [10]; information theoretic measures [16] have been used extensively. For an application in IO tables see [22].

A mathematically important criterion in classifying those variants is whether or not the row sums $T_{i\bullet}$ and the column sums $T_{\bullet j}$ are *consistent* in the sense that

$$\sum_j T_{i\bullet} = \sum_i T_{\bullet j} \ . \tag{8}$$

If the problem is consistent, then the *marginals*, i.e. the row and the column sums and the grand total

$$T_{\bullet\bullet} := \sum_{i,j} T_{ij} \tag{9}$$

may be dealt with as constraints since the system of linear equations describing the marginals (6) and (7) and the grand total (9) usually has an infinite number of solutions. If both the nonnegativity and the consistency conditions hold, the *RAS algorithm* [21] has been the favored solution procedure due mainly to its computational efficiency [1]. The same algorithm is also known as the *iterative (proportional) fitting* [8], *iterative (proportional) scaling* [7] or *biproportionate adjustment* [15]. An important property of the RAS algorithm is that it permits an interpretation as a *Kullback-Leibler information minimization* method; see for instance [15]. With this observation other algorithms may be designed in such a way that some closeness criterion between T and S, such as the K-L information [12, 11] or the χ^2 [2], is minimized under the constraints that T satisfies the marginal conditions, (6) and (7).

A problem with this class of constrained optimization formulations is that it does not work if the problem is *inconsistent*: the space of feasible solutions may collapse to null since the constraining equations, (6), (7), and (9) may contradict each other. The RAS algorithm, for instance, will oscillate and fail to converge in such cases.

B Weighted least squares

Let the matrix representation of equations (1) to (4) be

$$Ax = b$$

where A is the design matrix consisting of 0s and 1s, x the vector of unknowns $T_{i\,j}$ for $\forall i, j$, and b the given data $S_{i\,j}$, $T_{i\,\bullet}$, $T_{\bullet\,j}$, and $T_{\bullet\bullet}$. Then the system of near-equations to solve is

$$W^{1/2}Ax \approx W^{1/2}b \tag{10}$$

where $W := \text{diag}w$ is the diagonal matrix of weights whose diagonal elements are w_i. The square root $W^{1/2}$ of W is the diagonal matrix whose diagonal elements are $w_i^{1/2}$, the positive square roots of elements of w. Assuming additive independent normal errors

$$Ax + \epsilon = b \tag{11}$$

the weights w_ℓ should be set equal to the inverses of the variances of ϵ_ℓ [14, 5]. A way to solve (10) by the weighted least squares is by solving the system of linear equations

$$A'WAx = A'Wb.$$

where the primes are for transposition.

C Programs in R

```
# file trade.R
# Fill 2007 trade data from its marginals and 2006 data.

regions <- c("OD", "N.Am", "SC.Am", "Europe", "CIS",
             "Africa", "M.East", "Asia", "World")
OD      <- "origin-destination table"
N.Am    <- "North America"
SC.Amm  <- "South and Central America"
CIS     <- "Commonwealth of Independent States"
M.East  <- "Middle East"

# Read data
trd06   <- read.table("trade06",col.names=regions,fill=TRUE)
comment(trd06) <- "values in billion dollars;
  2006 data from WTO International Trade Statistics 2007; Table I.4"
trd07   <- read.table("trade07",col.names=regions,fill=TRUE)
comment(trd07) <- "values in billion dollars;
  2007 data from WTO International Trade Statistics 2008; Table I.4"
trd07na <- read.table("trade07na",col.names=regions,fill=TRUE)
comment(trd07na) <- "values in billion dollars;
   2007 data from WTO International Trade Statistics 2008; Table I.4"

# Design matrix builder
tblmat <- function(o.n,d.n){ # I, rowSums, colSums, total
    o1 <- function(i){
        .r0 <- rep(0,d.n); .r1 <- rep(1,d.n); r<-c()
        for(j in 1:o.n){
            if(i==j){r <- c(r,.r1)} else {r <- c(r,.r0)}}
        r}
    d1 <- function(j){
        .r1 <- rep(0,o.n); .r1[j] <- 1; r<-c()
        for(i in 1:d.n){r <- c(r,.r1)}
        r}
    a <- diag(1,o.n*d.n)                  # I
    for(i in 1:o.n){a <- rbind(a,o1(i))} # rowSums
    for(j in 1:o.n){a <- rbind(a,d1(j))} # colSums
    rbind(a,rep(1,o.n*d.n))              # total
    }

# Arrange data for lsfit()
tbl.size <- dim(trd06)
a        <- tblmat(tbl.size[1]-1,tbl.size[2]-2)
m06      <- as.matrix(trd06[1:8,2:9])
b0       <- as.vector(t(m06[2:8,1:7]))
infl     <- trd07na[1,9]/trd06[1,9]
```

```
b1       <- infl*b0
b        <- c(b1,trd07na[,9][2:8],
         as.vector(trd07na[1,][2:8],mode="numeric"),
         trd07na[1,9])
trd07s   <- round(infl*trd06[,-1],1) # prior

# Weights according to Demming-Stephan but UNCONSTRAINED
w        <- c(1/b0^2,rep(1,tbl.size[1]-1),rep(1,tbl.size[2]-1))
trd07wD          <- trd06
trd07wD[2:8,2:8] <- 1/(trd07s[2:8,2:8])^2
trd07wD[1,2:8]   <- trd07wD[1:8,9]   <- 1
trd07wD[1:8,2:9] <- signif(trd07wD[1:8,2:9],digits=1)

# Result Demming-Stephan UNCONSTRAINED
x                <- lsfit(a,b,w,intercept=FALSE)
trd07eD          <- trd07na; trd07eD[1:8,2:9] <- NA
trd07eD[2:8,2:8] <- matrix(round(x$coeff,1),7,7,byrow=TRUE)
trd07eD[1,2:8]   <- round(colSums(trd07eD[2:8,2:8]),1)
trd07eD[2:8,9]   <- round(rowSums(trd07eD[2:8,2:8]),1)
trd07eD[1,9]     <- sum(trd07eD[1,2:8])

# Weights according to Yoneda
wf       <- function(k){12*10^(2*k)}
w        <- c(1/((infl-1)*b0)^2,
         rep(wf(0),tbl.size[1]-1),rep(wf(-1),tbl.size[2]-1))
trd07wY          <- trd06
trd07wY[2:8,2:8] <- 1/(infl * trd06[2:8,2:8])^2
trd07wY[1,2:8]   <- wf(0)
trd07wY[1:8,9]   <- wf(-1)
trd07wY[1:8,2:9] <- signif(trd07wY[1:8,2:9],digits=1)

# Result Yoneda
x                <- lsfit(a,b,w,intercept=FALSE)
trd07eY          <- trd07na; trd07eY[1:8,2:9] <- NA
trd07eY[2:8,2:8] <- matrix(round(x$coeff,1),7,7,byrow=TRUE)
trd07eY[1,2:8]   <- round(colSums(trd07eY[2:8,2:8]),1)
trd07eY[2:8,9]   <- round(rowSums(trd07eY[2:8,2:8]),1)
trd07eY[1,9]     <- sum(trd07eY[1,2:8])

# Proportional allocation
trd07sm          <- as.matrix(trd07s)
trd07eP          <- trd07na;
trd07eP[2:8,2:8] <- round((trd07sm[1,1:7] %o% trd07sm[2:8,8])/
                           trd07sm[1,8],1)
trd07eP[1,2:8]   <- round(colSums(trd07eP[2:8,2:8]),1)
trd07eP[2:8,9]   <- round(rowSums(trd07eP[2:8,2:8]),1)
trd07eP[1,9]     <- sum(trd07eP[1,2:8])
```

```
# Evaluate results
d0 <- as.vector(as.matrix((trd07s [2:8,1:7]-trd07[2:8,2:8])))
dD <- as.vector(as.matrix((trd07eD[2:8,2:8]-trd07[2:8,2:8])))
dY <- as.vector(as.matrix((trd07eY[2:8,2:8]-trd07[2:8,2:8])))
dP <- as.vector(as.matrix((trd07eP[2:8,2:8]-trd07[2:8,2:8])))
bp <- seq(-40,100,10)
hist(d0,bp,ylim=c(0,25),xlab="Error [billion $]",main=NULL)
hist(dD,bp,ylim=c(0,25),xlab="Error [billion $]",main=NULL)
hist(dY,bp,ylim=c(0,25),xlab="Error [billion $]",main=NULL)
hist(dP,xlab="Error [billion $]",main=NULL)
```

File `trade06` :

```
"World" 2355 378 5118 290 283 381 2839 11783
"N.Am" 905.3 107.3 279.3 8.3 21.7 42.1 314.1 1678.3
"SC.Am" 135.0 111.5 86.4 6.1 11.3 7.9 61.8 429.9
"Europe" 430.3 66.6 3651.5 141.6 120.2 128.9 366.4 4963.0
"CIS" 24.2 7.6 246.5 80.3 5.7 13.3 45.6 425.6
"Africa" 79.8 11.3 148.1 1.4 32.8 6.3 72.6 363.3
"M.East" 72.3 4.4 102.8 3.0 20.9 71.6 339.6 645.5
"Asia" 708.3 69.5 603.8 49.7 69.9 111.4 1638.5 3277.8
```

File `trade07na` :

```
"World" 2517 451 5956 397 355 483 3294 13619
"N.Am" NA NA NA NA NA NA NA 1853.5
"SC.Am" NA NA NA NA NA NA NA 499.2
"Europe" NA NA NA NA NA NA NA 5772.2
"CIS" NA NA NA NA NA NA NA 510.3
"Africa" NA NA NA NA NA NA NA 424.1
"M.East" NA NA NA NA NA NA NA 759.9
"Asia" NA NA NA NA NA NA NA 3799.7
```

File `trade07` :

```
"World" 2517 451 5956 397 355 483 3294 13619
"N.Am" 951.2 130.7 328.7 12.4 27.3 50.1 352.1 1853.5
"SC.Am" 151.3 122.0 105.6 6.4 13.7 9.1 80.2 499.2
"Europe" 458.5 80.4 4243.6 189.0 147.7 152.9 433.7 5772.2
"CIS" 23.6 6.3 287.5 103.2 6.9 16.2 59.6 510.3
"Africa" 91.9 14.6 167.5 0.9 40.5 10.5 80.9 424.1
"M.East" 83.9 4.4 108.3 4.8 27.5 93.4 397.3 759.9
"Asia" 756.4 92.3 714.6 79.8 91.4 150.4 1889.8 3799.7
```