Statistisk sentralbyrå
Statistics Norway

Indirect Weight Benchmarking

Vemund Rundberget

TALL SOM FORTELLER

NOTATER / DOCUMENTS

2026/26

In the series Documents, documentation, method descriptions, model descriptions and standards are published.

Published: July 3rd, 2026

ISBN 978-82-587-2120-5 (electronic)
ISSN 2535-7271 (electronic)

Symbols in tables	Symbol
Category not applicable Figures do not exist at this time, because the category was not in use when the figures were collected.	.
Not available Figures have not been entered into our databases or are too unreliable to be published.	..
Confidential Figures are not published to avoid identifying persons or companies.	:
Decimal punctuation mark	.

Preface

This document describes an indirect method for benchmarking a matrix of weights by adjusting the weights by the smallest change that reproduces newly observed levels when the underlying cells cannot be observed directly.

The method was utilised as a part of Statistics Norway's benchmark revision of the national accounts in order to update its information on consumer behaviour. It was updated the consumption matrix used in Statistics Norway's national accounts as a part of the benchmark revision of the national accounts 2024. This was possible due to new information on household expenditures from Statstics Norway's Survey of consumer expenditure and detailed retail sales information from public registers without observing the link directly. This is both valuable for the updating the information in the national accounts themselves and for updating the weights in the consumer price index that follows from the national accounts.

The note is written to make the procedure as transparent as possible and to facilitate reuse in similar applications. It sets out the motivation, states the problem as a convex quadratic program characterised by its Kuhn–Tucker conditions, and includes a complete Python implementation using the cvxpy package.

Statistics Norway, June 30th, 2026

Linda Nøstbakken

Abstract

The national accounts of Norway uses a matrix of weights, the consumption matrix, to calculate the development of household consumption groups based off of retail-trade sales. However, the observation of both these weights and household consumption values are based on infrequent surveys. With new information on the target, household expenditures, this method allows updating the weights without observing them directly, given the values of retail-trade sales.

This is the indirect method for benchmarking a matrix of weights that links one vector to another through a matrix product. The method makes the smallest necessary changes to the weights in order to produce the target vector with the matrix product of the input vector and the new and endogenous weights. For reasonability, requirements are also put on the weights that they can individually only be between 0 and 1, as well as all the weights for a given industry summing up to 1. This allows for more updated information in intermittent estimates of the target vector.

Table of contents

Preface 3

Abstract 4

1. **The Weight Benchmarking Technique** 6
2. **Related Literature** 6
3. **Mathematical Problem** 7
4. **Python Code** 9

1. The Weight Benchmarking Technique

This document provides an indirect method of benchmarking a matrix of weights used in estimating a given vector of values based on some other vector values.

For the purposes of the Norwegian National Accounts, this approach is used as a part of the calculation of the Household's consumption of goods from retail trade across different consumption groups. Based on infrequent questionnaires, the National Accounts calculates a weight distribution of household consumption of goods across consumption groups from respective retail trade industries. This turns into what's called the consumption matrix. With only questionnaires that provide level information of household consumption of goods without the connection to which retail industries that provided the goods, the weights in the consumption matrix cannot directly be updated.

This method of benchmarking provides a means of indirectly updating the weights using only information about the sales in retail trade industries and another source for the level of household expenditures without information about which industries the consumption expenditures are from. This is possible due to the selection criterion that out of all the possible matrices that reproduce the expenditure levels, the matrix with the smallest sum of squared deviations is the one chosen. Notably, a pre-existing set of weights is also required.

If there were no errors in the calculations using the old weights, the result is simply the old weights. However, this is unlikely as the purchasing patterns within retail industries probably aren't constant over time. This necessitates updating them if new information about levels of household expenditures is available. This is also important as the shares of household consumption are crucial input in the weighting of consumer price indexes. These are affected by the weights in the consumption matrix, and thus this provides an improvement on the intermittent calculations of expenditure shares between direct updates to the consumption weights whenever new information on expenditure levels are available.

2. Related Literature

The benchmarking problem here is an instance of matrix balancing, adjusting the entries of a matrix to satisfy known margins while staying close to a prior. The classical multiplicative approach is iterative proportional fitting, also known as the RAS method, which scales rows and columns alternately to hit target margins (Deming and Stephan, 1940, Bacharach, 1970). Least-squares adjustment of national-accounts estimates subject to accounting constraints goes back to Stone et al. (1942), and movement-preserving benchmarking of time series to annual totals to Denton (1971), whose quadratic-minimisation formulation is close in spirit to the objective used here. The GRAS extension accommodates matrices with both positive and negative entries (Junius and Oosterhaven, 2003). The method in this note differs in minimising an additive squared change in the weights subject to a single set of level constraints together with row-sum (stochasticity) constraints, rather than fitting both row and column margins multiplicatively.

3. Mathematical Problem

The problem can be formulated as an optimisation problem. Given retail-trade sales, benchmark expenditure totals, and an existing weight matrix, the objective is to find the smallest adjustment to the weights that reproduces the benchmark totals while preserving the properties of a valid weight matrix.

Let $\mathbf{X}$ denote the updated weight matrix and let the sales vector, benchmark vector, and existing weights be given exogenously; indicated by the notation $(^*)$ throughout.

The exogenous values to benchmark against are the elements of the vector $\vec{V}$, in our case new levels of household consumption, with the vector $\vec{K}$, in our case the retail trade sales, and the former weights $\mathbf{W}^*$ as the input parameters. For the range of retail trade industries I and consumption groups J, the respective sales and consumption expenditure values are the row vectors

$$\vec{K}^* \equiv (K_1^*, K_2^*, \cdots, K_I^*) \in \mathbb{R}^{1\times I} \quad \text{and} \quad \vec{V}^* \equiv (V_1^*, V_2^*, \cdots, V_J^*) \in \mathbb{R}^{1\times J},$$

Our new weights $\mathbf{X}$ are solved for from the endogenous weight matrix

$$\mathbf{X} \equiv \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1J} \\ x_{21} & x_{22} & \cdots & x_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ x_{I1} & x_{I2} & \cdots & x_{IJ} \end{bmatrix}$$

where for each industry i, the retail industry's sales are distributed across the consumption groups $1, ..., J$. The objective function is the sum of squared deviations of the new weights from the previous ones, equivalently the squared Frobenius norm of the change in weights,

$$f(\mathbf{X}) \;=\; \|\mathbf{X} - \mathbf{W}^*\|_F^2 \;=\; \sum_{i=1}^{I}\sum_{j=1}^{J}\left(x_{ij} - w_{ij}^*\right)^2 .$$

which is a squared loss function for changing any given weight. The objective is to minimise the loss from changing the weights as much as possible subject to the Kuhn-Tucker conditions. The Kuhn-Tucker conditions provide the restrictions required to solve for new weights that when multiplied with the retail industries' sales will result in the given levels of household consumption expenditures. The constraints define the set of admissible weight matrices, while the Kuhn–Tucker conditions can be used to characterise the optimal solution.

We also require in this problem for reasonableness, as these are weights, that each individual weight be between 0 and 1 as well as the sum of the weights for every industry being 1.

Thus, the Kuhn–Tucker optimization problem in matrix form is

$$\min_{\mathbf{X}} \|\mathbf{X} - \mathbf{W}^*\|_F^2 \quad \text{s.t.} \quad \vec{K}^*\mathbf{X} = \vec{V}^*, \quad \mathbf{X}\,\vec{1}_J = \vec{1}_I, \quad \mathbf{0} \le \mathbf{X} \le \mathbf{1}_{IJ}.$$

The level constraint $\vec{K}^*\mathbf{X} = \vec{V}^*$ is a set of J equations, one per consumption group. The row-sum constraint $\mathbf{X}\,\vec{1}_J = \vec{1}_I$ is a set of I equations, one per industry. For the two equality constraints to be mutually consistent the totals must be equal, $\vec{K}^*\vec{1}_I = \vec{V}^*\vec{1}_J$[1].

We attach a multiplier to each constraint, a column vector $\vec{\lambda} \in \mathbb{R}^{J\times 1}$ to the level constraints, a row vector $\vec{\eta} \in \mathbb{R}^{1\times I}$ to the row-sum constraints, and matrices $\mathbf{M} \ge \mathbf{0}$ and $\mathbf{N} \ge \mathbf{0}$ (both $I \times J$) to the element-wise bounds $\mathbf{X} \ge \mathbf{0}$ and $\mathbf{X} \le \mathbf{1}_{IJ}$. Writing $\langle \mathbf{A}, \mathbf{B} \rangle \equiv \mathrm{tr}(\mathbf{A}^\top \mathbf{B}) = \sum_{i,j} A_{ij}B_{ij}$ for the Frobenius matrix inner product, the Lagrangian is the scalar

$$\mathcal{L} = \|\mathbf{X} - \mathbf{W}^*\|_F^2 - \left(\vec{K}^*\mathbf{X} - \vec{V}^*\right)\vec{\lambda} - \vec{\eta}\left(\mathbf{X}\vec{1}_J - \vec{1}_I\right) - \langle \mathbf{M}, \mathbf{X} \rangle + \langle \mathbf{N}, \mathbf{X} - \mathbf{1}_{IJ} \rangle.$$

Setting the matrix gradient $\nabla_{\mathbf{X}}\mathcal{L}$ to zero gives the stationarity condition

$$2\,(\mathbf{X} - \mathbf{W}^*) - (\vec{K}^*)^\top \vec{\lambda}^\top - \vec{\eta}^\top \vec{1}_J^\top - \mathbf{M} + \mathbf{N} = \mathbf{0},$$

where $(\vec{K}^*)^\top \vec{\lambda}^\top$ and $\vec{\eta}^\top \vec{1}_J^\top$ are outer products (rank-one $I \times J$ matrices); the transposes that the constraint avoided reappear here, as they must. Solving for the new weights gives

$$\mathbf{X} = \mathbf{W}^* + \tfrac{1}{2}\left((\vec{K}^*)^\top \vec{\lambda}^\top + \vec{\eta}^\top \vec{1}_J^\top + \mathbf{M} - \mathbf{N}\right).$$

This shows the structure of the adjustment. Every weight is nudged from its old value depending on whether the constraints require it. The solution to this problem is programmed numerically with the *cvxpy* package in Python, avaliable in the Python code chapter.

[1] This is forced by the two equality constraints alone. If $\vec{K}^*\mathbf{X} = \vec{V}^*$ and $\mathbf{X}\vec{1}_J = \vec{1}_I$, then $\vec{K}^*\vec{1}_I = \vec{V}^*\vec{1}_J$, i.e. $\sum_i K_i^* = \sum_j V_j^*$, since $\vec{V}^*\vec{1}_J = \vec{K}^*\mathbf{X}\vec{1}_J = \vec{K}^*(\mathbf{X}\vec{1}_J) = \vec{K}^*\vec{1}_I$.

4. Python Code

```
# # Indirect Weight Benchmarking
# This example shows the use of an indirect method of benchmarking
# a matrix of weights used in estimating a given vector of values
# based on some other vector values and a former set of weights.
# See the readme for further details.

import cvxpy as cp
import numpy as np

np.set_printoptions(suppress=True)
np.set_printoptions(formatter={'float': lambda x: "{0:0.2f}".format(x)})

rng = np.random.default_rng(42)   # seed for a reproducible example

# Example dimensions
I, J = 4, 3                       # industries (I), consumption groups (J)

# Input parameters
W = rng.random((I, J))            # previous weights
W = W / W.sum(axis=1, keepdims=True)  # rows MUST sum to 1
V = rng.random(J) * 100           # new benchmark levels by group
K_pre = rng.random(I) * 100       # retail sales by industry

# Guard the inputs (after scaling, sum(K) == sum(V) holds by construction).
if (K_pre < 0).any() or (V < 0).any():
    raise ValueError("K and V must be non-negative.")
if K_pre.sum() == 0:
    raise ValueError("Sum of K is zero; cannot scale.")

# Scale K so its total matches V's total (required by the constraints).
delta = V.sum() / K_pre.sum()
K = delta * K_pre
assert abs(V.sum() - K.sum()) < 1e-9

# Decision variables
X = cp.Variable((I, J))

# Objective: minimize squared deviation from old weights (scalar)
```

```
objective = cp.Minimize(cp.sum_squares(X - W))

# Constraints (X <= 1 is implied by row sums = 1 and X >= 0, so omitted):
constraints = [
    K @ X == V,              # reproduce benchmark (J equations)
    cp.sum(X, axis=1) == 1, # each industry’s shares sum to 1
    X >= 0,
]

# Solve and check the solver actually succeeded
problem = cp.Problem(objective, constraints)
problem.solve()
if problem.status not in ("optimal", "optimal_inaccurate"):
    raise RuntimeError(f"Benchmarking failed: status = {problem.status}")

# Results
X_opt = X.value
print("Old weights:\n", W)
print("New weights:\n", X_opt)
print("Change in weights:\n", X_opt - W)
print("Row sums (should be 1):", X_opt.sum(axis=1))
print("Result with new weights:", K @ X_opt)
print("Benchmark target:       ", V)
print("Difference to benchmark:", K @ X_opt - V)
print("Dual values (K X = V):  ", constraints[0].dual_value)
```

References

Bacharach, M. (1970). *Biproportional Matrices and Input–Output Change*. Cambridge University Press, Cambridge.

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

Denton, F. T. (1971). Adjustment of monthly or quarterly series to annual totals: An approach based on quadratic minimization. *Journal of the American Statistical Association*, 66(333):99–102.

Junius, T. and Oosterhaven, J. (2003). The solution of updating or regionalizing a matrix with both positive and negative entries. *Economic Systems Research*, 15(1):87–96.

Stone, R., Champernowne, D. G., and Meade, J. E. (1942). The precision of national income estimates. *The Review of Economic Studies*, 9(2):111–125.