Statistical Matching using KCCA and Super-OM*

Hugues Annoye (Corresponding Author)[a,b,d], Alessandro Beretta[a,e], Cédric Heuchenne[a,c,b,f]

[a] *Université Saint-Louis - Bruxelles, Boulevard du Jardin Botanique 43, 1000 Brussels, Belgium*
[b] *Université catholique de Louvain, Place de l'Université 1, 1348 Louvain-la-Neuve, Belgium*
[c] *Université de Liège, Place du 20 Août 7, 4000 Liège, Belgium*
[d] *Email : hugues.annoye@usaintlouis.be*
[e] *Email : alessandro.beretta@usaintlouis.be*
[f] *Email : cedric.heuchenne@usaintlouis.be*

Abstract

The potential to study and improve different aspects of our lives is ever growing thanks to the abundance of data available in today's modern society. Scientists and researchers often need to analyze data from different sources; the observations, which only share a subset of the variables, cannot always be paired to detect common individuals. This is the case, for example, when the information required to study a certain phenomenon is coming from different sample surveys. Statistical matching is a common practice to combine these data sets. In this paper, we investigate and extend to statistical matching two methods based on Kernel Canonical Correlation Analysis (KCCA) and Super-Organizing Map (Super-OM). These methods are designed to deal with various variable types, sample weights and incompatibilities among categorical variables. We use the 2017 Belgian Statistics on Income and Living Conditions (SILC) and we compare the performance of the proposed statistical matching methods by means of a cross-validation technique, as if the data were available from two separate sources.

Keywords: Statistical matching, Canonical Correlation Analysis, Kernel Canonical Correlation Analysis, Super-Organizing Map

1. Introduction

In the era of Big Data, a huge volume of information is continuously gathered through a wide range of activities. However, in many scientific and commercial applications, the availability of data still remains constrained due to the information of interest often being fragmented and coming from many different sources. Various private and public entities collect data by means of sample surveys, which are analyzed for a variety of reasons, but sometimes the required data are not available from a single source. In these situations, the set up of a new survey containing all of the required information would be impractical due to time and financial constraints. A more feasible alternative is to combine the already existing data using *statistical matching* (D'Orazio et al., 2006b), a technique which is also known as *data fusion*, *synthetical matching* or *statistical record-linkage*.

Let's consider the case of two data sets sharing some but not all variables referring to the same target population. An exact matching (record linkage) of the two data sets is likely to be impossible either because the observed individuals in the two data sets do not overlap (often the case with independent sample surveys conducted on large populations) or because a common variable for the identification of the individuals (e.g. social security number) is not available. In these cases, statistical matching can be used to merge the two data sets (D'Orazio et al., 2006b). The result will be a *synthetic* data set with both common and non-common variables jointly displayed, which can be used to carry out further statistical analysis. Statistical matching can be viewed as a missing data problem and several imputation methods can be used to fill the missing values. These techniques were first introduced in (Anderson, 1957); for a literature review, see (Kim and Shao, 2013; Van Buuren, 2018). The use of statistical matching started growing during the 1960s and has gained particular prominence in Europe in the context of media analysis in marketing, see (Rässler, 2002) for a detailed history and (Fosdick et al., 2016; Conti et al., 2017) for a literature review of statistical matching. Among others, a comparison of exposure to media and purchase behavior has been made in (Kamakura and Wedel, 1997) merging data coming from two separate surveys. Statistical matching has been widely adopted also in economics, where (Okner, 1972) was the first main contribution. A typical situation, where it plays a key role, is the study of the relationship between household income and consumption expenditure (Tonkin and Webber, 2012; Donatiello et al., 2014; Serafino and Tonkin, 2017; López-Laborda et al., 2020). The national statistical institutes of many European countries collect this kind of information, but subdivide it into two separate surveys: the Statistics on Income and Living Conditions (SILC) and the Household Budget Survey (HBS). In this case, the goal of statistical matching is to create a synthetic data set where income and expenditure variables are jointly displayed. Similar methods are also used by national institutions

*This work was funded by Innoviris in the Prospective Research for Brussels program (project 2019-PRB-8). We gratefully acknowledge data in the form of the Survey on Income and Living Conditions (2016 and 2017) from Eurostat and the Household Budget Survey (2016) from Statistics Belgium.

as in (Saverio et al., 2008), where they integrate two Italian surveys (Labour force and Time use) to have one unique data set to avoid the costs of doing a new survey to collect both information together.

The objective of this paper is to propose new statistical matching procedures based on extensions of two machine learning techniques that are adapted to statistical matching for tabular data with sample weights: Kernel Canonical Correlation Analysis (KCCA), and Super-Organizing Map (Super-OM). We compare these methods with more trivial extensions using Canonical Correlation Analysis (CCA) as well as with more classical econometric methods based on distance hot-deck (HD) and multivariate linear and multinomial logistic regressions (REG). KCCA has been used for data fusion in (Mitsuhiro and Hoshino, 2020). However, it is extended here to deal with further issues as the two other techniques, namely the ability to deal with mixed variable types, sample weights and problems of incompatibility between categorical variables, as pointed out by (D'Orazio et al., 2006a). This latter case occurs, for example, when we have two variables that indicate the city and the country of residency: if Paris corresponds to the variable city, then the imputed country should be France and not another country.

Canonical Correlation Analysis (CCA) (Hotelling, 1936) is a method of dimensionality reduction. We can extend it in a non-linear way with the KCCA proposed by (Lai and Fyfe, 2000) and (Akaho, 2001). KCCA is a machine learning technique that uses the kernel trick to map data into higher dimensional spaces where classical CCA is performed to extract non linear relationships from the data. In cross-domain matching, KCCA has also been used to perform matching between text and images using information about the relationship between both data sources (Shimodaira, 2014). The use of KCCA for statistical matching is first proposed by (Mitsuhiro and Hoshino, 2020). That type of matching is in the class of kernel matching, because it assigns new values by kernelized means of the observed ones.

The second approach is based on the Super-Organizing Map (Super-OM) (Wehrens and Buydens, 2007), a more flexible extension of the Self-Organizing Map (SOM) introduced by (Kohonen, 1982). The SOM is an unsupervised machine learning method used to produce a low-dimensional representation of a high-dimensional input space, which relies on an artificial neural network architecture. In the literature, SOMs have been already adopted to impute missing values, a common issue in many practical applications. For example, they have been used to impute missing values in the French National Personal Transport Survey (1993-1994) (Fessant and Midenet, 2002). A similar technique has been applied to socioeconomic data on housing in Ile-de-France and government spending (Cottrell and Letrémy, 2007) and, more recently, to surface water data with observations on different physicochemical variables (Folguera et al., 2015). In this paper, we propose a statistical matching technique based on a Super-OM, an extension of the classical SOM that allows for linking two sets of variables. To the best of our knowledge there is no other article proposing a statistical matching technique based on either SOMs or Super-OMs.

The article is structured as follows. First, section 2 provides background information about statistical matching. In section 3, we present the two methodologies and their theoretical foundations. In section 4, we use the SILC 2017 survey data for Belgium; we treat them as if they were available from two separate sources and by means of a cross-validation technique we compare the performance of the proposed statistical matching methods. Finally, in Section 5, we provide an application where we merge the SILC 2016 and HBS 2016 surveys for Belgium, similar to the one in (Tonkin and Webber, 2012; Serafino and Tonkin, 2017).

2. Statistical matching : Background

We consider two data sets from two independent sample surveys (A and B) containing n_A and n_B individuals, respectively. Each data set is composed of a matrix of common variables $\mathbf{X}$ and two matrices of non-common variables $\mathbf{Y}$ and $\mathbf{Z}$. In Figure 1, we summarize all the matrices and their dimensions, where J_q, for $q \in \{x, y, z\}$, denotes the number of columns (i.e. variables). Note that all the values in $\mathbf{Y}^B$ and $\mathbf{Z}^A$ are missing. In general, we denote the values of the i-th individual in $\mathbf{X}^A$ by $\mathbf{x}_i^A$. Finally, we use $\mathbf{w}^A$ and $\mathbf{w}^B$ to indicate the vectors of individual weights (bounded between zero and one) of length n_A and n_B, respectively, whereas $\mathbf{W}^A$ and $\mathbf{W}^B$ correspond to their diagonal matrices.

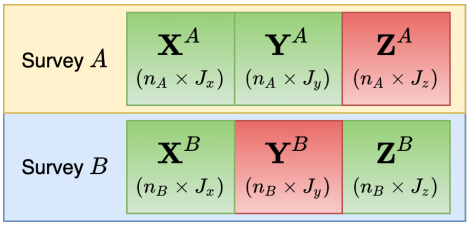

Figure 1: Sample survey data.

The purpose of the methodologies described in this section is to create a synthetic data set, where $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ are jointly displayed. For the sake of simplicity, in the rest of the paper, we will focus our attention on the creation of the synthetic data set $(\mathbf{X}^B, \mathbf{Y}^B, \mathbf{Z}^B)$, but the same procedure can be used to create $(\mathbf{X}^A, \mathbf{Y}^A, \mathbf{Z}^A)$.

In the statistical matching literature, in order to reconstruct the missing variables $\mathbf{Y}^B$ in data set B, there are three main traditional approaches (Aluja-Banet et al., 2007): parametric multivariate distributions, regressions and hot-deck. The first consists in estimating a full joint parametric distribution of $(\mathbf{X}, \mathbf{Y})$ from data set A and then use it to impute $\mathbf{Y}^B$ in B. The second approach investigates the relationship between $\mathbf{X}^A$ and $\mathbf{Y}^A$ in data set A using regression models (not necessarily linear), which are then applied to reconstruct $\mathbf{Y}^B$ in data set B. The third approach, hot-deck, is a fully non-parametric method, which consists in finding for each entity in data set B a similar entity in data set A from which we can borrow the values of $\mathbf{Y}^A$. Furthermore, mixed methods exist, which combine two of the previous approaches. Other non-traditional methods based on Bayesian modeling (D'Orazio et al., 2006b) and machine learning algorithms exist in the literature. The Bayesian techniques

use posterior distributions to draw values for the parameters of the joint distribution of X and Y and generate data from it. For the machine learning ones, a two-step procedure has been proposed in (Spaziani et al., 2019), where machine learning techniques (naive Bayes, random forests or boosting algorithms) are first applied to predict the non-common variables in both data sets and, then, a hot-deck method is used to improve these predictions. However, these procedures do not take the survey weights into account and do not consider a unique relationship between all common variables and all non-common variables.

2.1. Sample Weights

In general, sampling is not done in a uniform way on the whole population. This means that two households do not have the same probability of being drawn in the sample. In fact, draws are usually done using stratification methods. For example, for SILC data in Belgium, see Section 4, a two-stage stratification method is used. First, a certain number of municipalities are drawn in each region (e.g. province in Belgium), and then, in each municipality, a certain number of households will be selected according to a second criterion (e.g. a systematic selection on the age of the reference person or the tax quantile).

Such a construction of our database forces us to use weights to take into account the fact that two households do not have the same chances to be selected and do not represent the same number of people in the population studied.

First, some design weights are calculated. They are set equal to the inverse probability of selection of an individual in the population (see Appendix A for more details). Next, the sample weights are created by adjusting these design weights for non-responses and to reproduce characteristics from the population, such as gender, age, size of the household, region, working status, etc (see Appendix B.2 for more details). Finally, the individual weights are the household weights that are assigned to each member of the household.

In a large number of surveys, these weights are available. So, it is fundamental to take them into account during all statistical analysis methods and therefore during statistical matching to get the best possible match between our final results and the population.

3. Proposed statistical matching technique

3.1. Kernel Canonical Correlation Analysis (KCCA)

In this section, we present an extension of the statistical matching technique based on the Kernel Canonical Correlation Analysis (KCCA) proposed by (Mitsuhiro and Hoshino, 2020). First, we provide a brief introduction to the classical Canonical Correlation Analysis (CCA) and how it can be used to perform statistical matching. Then, we present its non-linear extension: KCCA. In order to simplify the notation, we consider $\mathbf{X}$ and $\mathbf{Y}$, two centered matrices with n rows. In the context of statistical matching, these matrices can be centered version of $\mathbf{X}^A$ and $\mathbf{Y}^A$, or $\mathbf{X}^B$ and $\mathbf{Z}^B$.

3.1.1. Canonical Correlation Analysis (CCA)

Canonical Correlation Analysis (CCA) is based on the following maximization problem:

$$\max_{\mathbf{a},\mathbf{b}} \mathbf{a}^T\mathbf{X}^T\mathbf{W}\mathbf{Y}\mathbf{b},$$

under the constraints that:

$$\mathbf{a}^T\mathbf{X}^T\mathbf{W}\mathbf{X}\mathbf{a} = 1 \text{ and } \mathbf{b}^T\mathbf{Y}^T\mathbf{W}\mathbf{Y}\mathbf{b} = 1,$$

where $\mathbf{a}$ and $\mathbf{b}$ are known as canonical vectors and $\mathbf{U} = \mathbf{X}\mathbf{a}$ and $\mathbf{V} = \mathbf{Y}\mathbf{b}$ as canonical variables. This consists of maximizing the correlation between $\mathbf{X}\mathbf{a}$ and $\mathbf{Y}\mathbf{b}$ under constraints on their variance to ensure uniqueness of the solution.

The same problem can be formulated as a generalized eigenvalue problem:

$$\begin{pmatrix} 0 & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & 0 \end{pmatrix}\begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{C}_{xx} & 0 \\ 0 & \mathbf{C}_{yy} \end{pmatrix}\begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix},$$

where, $\mathbf{C}_{xx} = \mathbf{X}^T\mathbf{W}\mathbf{X}$, $\mathbf{C}_{yy} = \mathbf{Y}^T\mathbf{W}\mathbf{Y}$ $\mathbf{C}_{xy} = \mathbf{X}^T\mathbf{W}\mathbf{Y}$ and $\mathbf{C}_{yx} = \mathbf{Y}^T\mathbf{W}\mathbf{X}$.

Statistical matching: imputation of the missing values.. After centering $\mathbf{X}^A$ and $\mathbf{Y}^A$, the $\mathbf{a}$ obtained from the CCA procedure is used to impute the missing values in $\mathbf{Y}^B$. For this purpose, we compute $\mathbf{U}^A = \mathbf{X}^A\mathbf{a}$ and $\mathbf{U}^B = \mathbf{X}^B\mathbf{a}$, where $\mathbf{X}^B$ is centered, and we use a kernel function, e.g. Gaussian

$$K_h(x, y) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-y)^2}{2h^2}\right),$$

to measure the distance between all components of $\mathbf{U}^A$ and $\mathbf{U}^B$. The bandwidth parameter h can be chosen by a cross-validation procedure. Finally, we obtain $\widehat{\mathbf{Y}}^B = \mathbf{\Omega}\mathbf{Y}^A$, a weighted mean of the variables in $\mathbf{Y}^A$, where:

$$\begin{aligned} \mathbf{\Omega} &= (\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_{n_B})^T \\ \boldsymbol{\omega}_i &= \left(\frac{w_1^A \varpi_{i1}}{\sum_{j=1}^{n_A} w_j^A \varpi_{ij}}, \frac{w_2^A \varpi_{i2}}{\sum_{j=1}^{n_A} w_j^A \varpi_{ij}}, \cdots, \frac{w_{n_A}^A \varpi_{in_A}}{\sum_{j=1}^{n_A} w_j^A \varpi_{ij}} \right), \end{aligned} \quad (1)$$

where $\varpi_{ij} = K_h\left(u_i^B, u_j^A\right)$ for $i \in \{1, \cdots, n_B\}$ and $j \in \{1, \cdots, n_A\}$, and u_i^B (resp. u_j^A) is an element of $\mathbf{U}^B$ (resp. $\mathbf{U}^A$).

Statistical matching: multiple canonical variables. In general, it is possible to consider the first κ canonical variables. If $\mathbf{X}$ and $\mathbf{Y}$ contain zero-mean variables and κ is the rank of the matrix X^TY, the i-th canonical variable, for $i = 2, \cdots, \kappa$, can be calculated by solving the following problem:

$$\max_{\mathbf{a},\mathbf{b}} \mathbf{a}_i^T\mathbf{X}^T\mathbf{W}\mathbf{Y}\mathbf{b}_i$$

under the constraints

$$\mathbf{a}_i^T\mathbf{X}^T\mathbf{W}\mathbf{X}\mathbf{a}_j = \begin{cases} 0 & j < i \\ 1 & j = i \end{cases} \text{ and } \mathbf{b}_i^T\mathbf{Y}^T\mathbf{W}\mathbf{Y}\mathbf{b}_j = \begin{cases} 0 & j < i \\ 1 & j = i \end{cases},$$

where a_1 and b_1 are the first canonical vectors.

In order to perform the statistical matching with more than one canonical variable it is possible to use a product kernel.

3.1.2. Kernel canonical correlation analysis (KCCA)

Kernel canonical correlation analysis (KCCA) is a non-linear extension of CCA proposed by (Lai and Fyfe, 2000) and (Akaho, 2001). Contrary to the classical CCA, the matrices $\mathbf{X}$ and $\mathbf{Y}$ are transformed into

$$\boldsymbol{\Phi}_x(\mathbf{X}) = (\boldsymbol{\phi}(\mathbf{x}_1), \boldsymbol{\phi}(\mathbf{x}_2), \cdots, \boldsymbol{\phi}(\mathbf{x}_n)) \in H_x$$
and
$$\boldsymbol{\Phi}_y(\mathbf{Y}) = (\boldsymbol{\phi}(\mathbf{y}_1), \boldsymbol{\phi}(\mathbf{y}_2), \cdots, \boldsymbol{\phi}(\mathbf{y}_n)) \in H_y,$$

where H_x and H_y are Hilbert spaces. To simplify the notations, we suppose in the rest of the paper that mapped data are centered. The case of non-centered mapped data can be easily treated by the technique used for Kernel Principal Component Analysis (KPCA) (see (Schölkopf et al., 1998), appendix B). We denote the inner products in the Hilbert spaces by $\mathbf{U} = \langle \mathbf{a} | \boldsymbol{\Phi}_x(\mathbf{X}) \rangle$ and $\mathbf{V} = \left\langle \mathbf{b} \middle| \boldsymbol{\Phi}_y(\mathbf{Y}) \right\rangle$, where $\mathbf{a} \in H_x$ and $\mathbf{b} \in H_y$.

The aim is to find $\mathbf{a}$ and $\mathbf{b}$ that maximize the correlation between $\mathbf{U}$ and $\mathbf{V}$ under variance constraints to insure the uniqueness of the solution.

For this purpose, as in (Akaho, 2001), we can write

$$\mathbf{a} = \sum_{i=1}^{n} \alpha_i \boldsymbol{\phi}_x(\mathbf{x}_i) \qquad \mathbf{b} = \sum_{i=1}^{n} \beta_i \boldsymbol{\phi}_y(\mathbf{y}_i)$$
$$\mathbf{U} = \sum_{i=1}^{n} \alpha_i \langle \boldsymbol{\phi}_x(\mathbf{x}_i) | \boldsymbol{\phi}_x(\mathbf{X}) \rangle \qquad \mathbf{V} = \sum_{i=1}^{n} \beta_i \left\langle \boldsymbol{\phi}_y(\mathbf{y}_i) \middle| \boldsymbol{\phi}_y(\mathbf{Y}) \right\rangle,$$

where α_i and β_i are scalars. In practice, thanks to the Mercer theorem, we do not need an explicit form for $\boldsymbol{\phi}_x$ and $\boldsymbol{\phi}_y$. In fact, the inner product $\left\langle \boldsymbol{\phi}_x(\mathbf{x}_i) \middle| \boldsymbol{\phi}_x(\mathbf{x}_j) \right\rangle$, resp. $\left\langle \boldsymbol{\phi}_y(\mathbf{y}_i) \middle| \boldsymbol{\phi}_y(\mathbf{y}_j) \right\rangle$, can be replaced by a symmetric positive definite kernel $(\mathbf{K}_x)_{ij} = K_{h_x}(\mathbf{x}_i, \mathbf{x}_j)$, resp. $(\mathbf{K}_y)_{ij} = K_{h_y}(\mathbf{y}_i, \mathbf{y}_j)$, where $\mathbf{K}_x$ and $\mathbf{K}_y$ are known as Gramian matrices.

We can calculate $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_n)^T$ and $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_n)^T$ as a solution of the following generalized eigenvalue problem (Bach and Jordan, 2002):

$$\begin{pmatrix} 0 & \mathbf{K}_x\mathbf{K}_y \\ \mathbf{K}_y\mathbf{K}_x & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{K}_x\mathbf{K}_x & 0 \\ 0 & \mathbf{K}_y\mathbf{K}_y \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix}. \quad (2)$$

Note that if we want to take into account the individual weights, we need to rewrite equation (2) as follows:

$$\begin{pmatrix} 0 & \mathbf{K}_x\mathbf{W}\mathbf{K}_y \\ \mathbf{K}_y\mathbf{W}\mathbf{K}_x & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{K}_x\mathbf{W}\mathbf{K}_x & 0 \\ 0 & \mathbf{K}_y\mathbf{W}\mathbf{K}_y \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix}. \quad (3)$$

If the Gramian matrices are not full rank, the matrices

$$\begin{pmatrix} \mathbf{K}_x\mathbf{K}_x & 0 \\ 0 & \mathbf{K}_y\mathbf{K}_y \end{pmatrix} \text{ or } \begin{pmatrix} \mathbf{K}_x\mathbf{W}\mathbf{K}_x & 0 \\ 0 & \mathbf{K}_y\mathbf{W}\mathbf{K}_y \end{pmatrix}$$

will be singular, as explained by (Melzer et al., 2001) and (Kuss and Graepel, 2003). Thus, we need a regularization parameter γ, which avoids singularities and guarantees uniqueness of the solution:

$$\begin{pmatrix} 0 & \mathbf{K}_x\mathbf{W}\mathbf{K}_y \\ \mathbf{K}_y\mathbf{W}\mathbf{K}_x & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{K}_x\mathbf{W}\mathbf{K}_x + \gamma I_n & 0 \\ 0 & \mathbf{K}_y\mathbf{W}\mathbf{K}_y + \gamma I_n \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix},$$

or alternatively:

$$\begin{pmatrix} 0 & \mathbf{K}_x\mathbf{W}\mathbf{K}_y \\ \mathbf{K}_y\mathbf{W}\mathbf{K}_x & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} = \lambda \cdot$$
$$\begin{pmatrix} \frac{1}{2}\left((\mathbf{K}_x + \gamma I_n)\mathbf{W}\mathbf{K}_x + \mathbf{K}_x\mathbf{W}(\mathbf{K}_x + \gamma I_n)\right) & 0 \\ 0 & \frac{1}{2}\left((\mathbf{K}_y + \gamma I_n)\mathbf{W}\mathbf{K}_y + \mathbf{K}_y\mathbf{W}(\mathbf{K}_y + \gamma I_n)\right) \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix},$$

where I_n is the $n \times n$ identity matrix. The bandwidth parameters of the kernels $K_{h_x}(\cdot,\cdot)$ and $K_{h_y}(\cdot,\cdot)$, as well as the regularization parameter γ, can be chosen using a cross-validation procedure.

Similarly to CCA, KCCA can be extended to the case of more than one canonical variable.

Statistical matching: imputation of the missing values. Once we fit a KCCA model on $\mathbf{X}^A$ and $\mathbf{Y}^A$, we calculate $\boldsymbol{\alpha}$. We can use it to impute the missing values in $\mathbf{Y}^B$. For this purpose, we compute $\mathbf{U}^A = \mathbf{K}_x^A\boldsymbol{\alpha}$ and $\mathbf{U}^B = \mathbf{K}_x^B\boldsymbol{\alpha}$, where $\left(\mathbf{K}_x^A\right)_{ij} = K_{h_x}\left(\mathbf{x}_i^A, \mathbf{x}_j^A\right)$ and $\left(\mathbf{K}_x^B\right)_{ij} = K_{h_x}\left(\mathbf{x}_i^B, \mathbf{x}_j^A\right)$. Finally, we obtain $\widehat{\mathbf{Y}}^B = \boldsymbol{\Omega}\mathbf{Y}^A$, a weighted mean of the variables in $\mathbf{Y}^A$, where $\boldsymbol{\Omega}$ is defined in equation 1 and the corresponding bandwidth is chosen by cross-validation.

Algorithm 1 KCCA

Input: $\mathbf{X}^A$ and $\mathbf{Y}^A$, the centered common and non-common variables in data set A and $\mathbf{X}^B$, the common variables in data set B. $\mathbf{w}^A$ and $\mathbf{w}^B$ the weights in both data sets.
Output: A data set B with the centered $\mathbf{X}^B$ and $\widehat{\mathbf{Y}}^B$

```
Calculate α and β of the KCCA between X^A and Y^A using w^A.
Calculate the kernel canonical variables U^A = K_x^A α and U^B = K_x^B α.
for i = 1, ..., n^B do
    for j = 1, ..., n^A do
        Calculate ϖ_ij = K_h(u_i^B, u_j^A) where K_h is a Gaussian kernel.
    end for
end for
Calculate Ŷ^B = ΩY^A.
```

3.2. Super-Organizing Map

The Super-Organizing Map (Super-OM) is an extension of the Self-Organizing Map (SOM). It is composed of separate layers for different sets of input variables with distinct weights, which make the Super-OM a more flexible model compared to the classical SOM. In this article, we will focus on a Super-OM with two input layers: $\mathbf{X}^A$ and $\mathbf{Y}^A$, the two sets of *common* and *non-common* variables, respectively.

We consider a Super-OM with a two-dimensional rectangular grid of $\mathcal{K}$ neurons (other types of grids can also be used, e.g. hexagonal). The two input layers are connected to the neurons in the corresponding hidden layers, which are characterised by weight vectors $\mathbf{m}_{k,1} \in \mathbb{R}^{J_x}$ and $\mathbf{m}_{k,2} \in \mathbb{R}^{J_y}$, for $k = 1, \cdots, \mathcal{K}$. The output layer consists of the $\mathcal{K}$ neurons with weight vectors $\mathbf{m}_k = (\mathbf{m}_{k,1}, \mathbf{m}_{k,2})$, for $k = 1, \cdots, \mathcal{K}$. In Figure 2, we provide an example of Super-OM architecture with two layers and $\mathcal{K} = 9$ neurons.

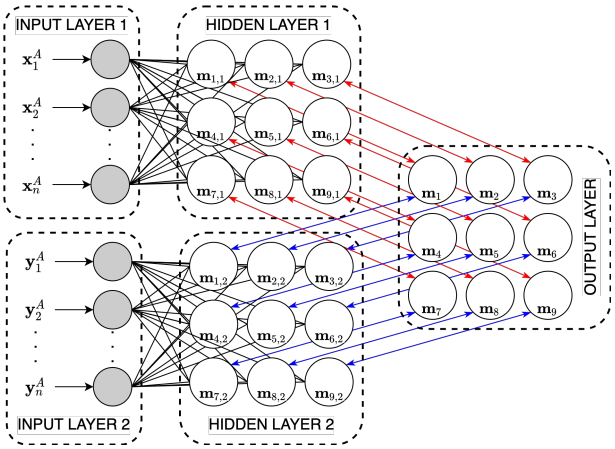

Figure 2: Architecture of a Super-OM with 2 layers and 9 neurons.

The vectors of weights $\mathbf{m}_k$ are estimated by means of a batch algorithm similar to the one used for the classical SOM. The main difference lies in the calculation of the distance measure, which now takes into account the weight of each layer. The vectors of weights are initialized with random values and, at each iteration $t \in \{1, \cdots, T\}$ the algorithm follows the steps described hereafter:

1. For all input vectors $\{(\mathbf{x}_i^A, \mathbf{y}_i^A) : i = 1, \cdots, n_A\}$, find the Best Matching Unit (BMU), i.e. the neuron with the lowest distance from the input vector $(\mathbf{x}_i^A, \mathbf{y}_i^A)$:

$$\tilde{k}_i^{(t)} = \arg\min_k d_{i,k}^{(t)};$$

where $\tilde{k}_i^{(t)}$ denote the index of the BMU and

$$d_{i,k}^{(t)} = \delta \sqrt{\sum_{j=1}^{J_x} \left[x_{i,j}^A - m_{k,1,j}^{(t)}\right]^2} + (1-\delta) \sqrt{\sum_{j=1}^{J_y} \left[y_{i,j}^A - m_{k,2,j}^{(t)}\right]^2} \tag{4}$$

is a weighted Euclidean distance (other distance measures can be used) and δ is the weight of the first layer;

2. Update the vectors of weights (for $k = 1, \cdots, \mathcal{K}$):

$$\mathbf{m}_k^{(t+1)} = \frac{\sum_{i=1}^{n_A} (\mathbf{x}_i^A, \mathbf{y}_i^A) w_i^A K_{\tilde{k}_i,k}^{(t)}}{\sum_{i=1}^{n_A} w_i^A K_{\tilde{k}_i,k}^{(t)}}$$

where w_i^A is the sample weight, $(\mathbf{x}_i^A, \mathbf{y}_i^A)$ is a row vector and $K_{\tilde{k}_i,k}^{(t)}$ is a neighborhood function centered in the BMU, e.g. a Gaussian kernel

$$K_{\tilde{k}_i,k}^{(t)} = \exp\left(-\frac{d_{\tilde{k}_i,k}^{*(t)}}{2\sigma^2(t)}\right),$$

where $d_{\tilde{k}_i,k}^{*(t)}$ is the distance between the neurons $\tilde{k}_i^{(t)}$ and k in the topological map space; and $\sigma(t)$ is the width of the kernel, a monotonically decreasing function [1]

$$\sigma(t) = \sigma_0 - (\sigma_0 - \sigma_1)\frac{t}{T};$$

3. Set $t = t + 1$ and repeat from step 1 while $t < T$.

Statistical matching: imputation of the missing values. The imputation of the missing values in $\mathbf{Y}^B$ is performed in three steps:

1. The hyperparameters (i.e. the size of the topological map and the weight δ) of the Super-OM are tuned on data set A using cross-validation;
2. Use the selected hyperparameters to train the Super-OM on data set A;
3. For $i = 1, \cdots, n_B$, find the Best Matching Unit (BMU), the neuron with the lowest distance between $\mathbf{x}_i^B$ and the weight vectors $\hat{\mathbf{m}}_{k,1}$, for $k = 1, \cdots, \mathcal{K}$, and compute the prediction $\hat{\mathbf{y}}_i^B$ as the weighted average of $\mathbf{Y}^A$. The latter is computed with weights, for $j = 1, \cdots, n_A$, equal to w_j^A if $\mathbf{x}_j^A$ belongs to the BMU and 0, otherwise.

3.3. How to deal with categorical variables

In order to avoid incompatibility problems in the categorical variables (e.g. predicting that a person lives in Paris, while he/she is living in Germany), we propose to use a two-step procedure. First, we impute the missing categorical variables in $\mathbf{Y}^B$ (or $\mathbf{Z}^A$) and then the continuous ones.

In the first step, the imputation of the categorical variables is based on all the common variables ($\mathbf{X}^A$ and $\mathbf{X}^B$) and the compatibility matrix $\mathbf{\Theta}$, where Θ_{ij}, for $i \in \{1, \cdots, n_B\}$ and $j \in \{1, \cdots, n_A\}$, takes value one if the ith row of categorical common variables in $\mathbf{X}^B$ is the same as the jth row of categorical common variables in $\mathbf{X}^A$ and zero, otherwise. Note that if an individual in data set B does not have a compatible counterpart in A, we reduce the number of variables on which we verify the equality until there is at least one compatible individual. Once the compatibility matrix is computed, for each individual i of the receiver data set, we draw at random individuals from data set A with probabilities:

$$\tilde{\omega}_i = \left(\frac{w_1^A \varpi_{i1}\Theta_{i1}}{\sum_{j=1}^{n_A} w_j^A \varpi_{ij}\Theta_{ij}}, \frac{w_2^A \varpi_{i2}\Theta_{i2}}{\sum_{j=1}^{n_A} w_j^A \varpi_{ij}\Theta_{ij}}, \cdots, \frac{w_{n_A}^A \varpi_{in_A}\Theta_{in_A}}{\sum_{j=1}^{n_A} w_j^A \varpi_{ij}\Theta_{ij}}\right), \tag{5}$$

where $\varpi_{ij} = K_h\left(u_i^B, u_j^A\right)$, for the techniques based on CCA. Whereas, for the Super-OM technique, $\varpi_{ij} = K_{h_s}(\mathbf{x}_j^A, \mathbf{x}_i^B)\varkappa_{ij}$, where $\varkappa_{ij}$ is a dummy variable equal to one if $\mathbf{x}_j^A$ has the same best matching unit as $\mathbf{x}_i^B$ and zero, otherwise, and a simple "rule of thumb" is used to select h_s (i.e. $h_s = 1.06\hat{\sigma}n^{1/5}$).

In the second step, the categorical variables imputed in the first step are added to the common variables and we impute

[1] where σ_0 is the quantile 2/3 of the distances between neurons in the topological map and σ_1 is equal to zero, as chosen in (Wehrens and Buydens, 2007) and (Wehrens and Kruisselbrink, 2018).

Algorithm 2 Super-OM

Input: $\mathbf{X}^A$ and $\mathbf{Y}^A$, the centered common and non-common variables in data set A and $\mathbf{X}^B$, the common variables in data set B. $\mathbf{w}^A$ and $\mathbf{w}^B$ the weights in both data sets.

Output: A data set B with the centered $\mathbf{X}^B$ and $\widehat{\mathbf{Y}}^B$.

Initialize the weights $\mathbf{m}_k$.
$t = 1$
while $t < T$ **do**
  $\forall i \in \{1, \cdots, n_A\}$ find the Best Matching Unit of $\{(\mathbf{x}_i^A, \mathbf{y}_i^A)\}$:

$$\tilde{k}_i^{(t)} = \arg\min_k d_{i,k}^{(t)};$$

  for $k = 1, \cdots, \mathcal{K}$ **do** Update :

$$\mathbf{m}_k^{(t+1)} = \frac{\sum_{i=1}^{n_A} (\mathbf{x}_i^A, \mathbf{y}_i^A) w_i^A K_{\tilde{k}_i,k}^{(t)}}{\sum_{i=1}^{n_A} w_i^A K_{\tilde{k}_i,k}^{(t)}};$$

  end for
  Set $t = t + 1$
end while
for $i = 1, \cdots, n_B$ **do**
  Find the BMU of $\mathbf{x}_i^B$
  for $j, 1 \cdots, n_A$ **do**
    Calculate $\mathbf{1}_{ij}^{BMU}$ that is equal to 1 if i and j have the same BMU and 0 otherwhise.
  end for
end for
Calculate $\widehat{\mathbf{Y}}^B = \mathbf{\Omega}\mathbf{Y}^A$, where:

$$\mathbf{\Omega} = (\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_{n_B})^T$$

$$\boldsymbol{\omega}_i = \left(\frac{w_1^A \mathbf{1}_{i1}^{BMU}}{\sum_{j=1}^{n_A} w_j^A \mathbf{1}_{ij}^{BMU}}, \frac{w_2^A \mathbf{1}_{i2}^{BMU}}{\sum_{j=1}^{n_A} w_j^A \mathbf{1}_{ij}^{BMU}}, \cdots, \frac{w_{n_A}^A \mathbf{1}_{in_A}^{BMU}}{\sum_{j=1}^{n_A} w_j^A \mathbf{1}_{ij}^{BMU}} \right).$$

the continuous variables in $\mathbf{Y}^B$ (or $\mathbf{Z}^A$) using the methodology presented in the previous Sections 3.1.1, 3.1.2 and 3.2.

This two-step procedure helps preserve joint distributions of subsets of common and non-common variables, as well as subsets of non-common variables only. Let's consider the simple case of two dummies (e.g., living in Belgium and in Brussels, where the first can be a common or a non-common variable and the second is a non-common variable) and one continuous non-common variable (e.g., gross income). We could predict the non-common variables using a one-step procedure not adjusting for compatibility constraints. However, this would introduce a bias in the estimation of the proportion of inhabitants in Belgium who live in Brussels. Our two-step procedure will lead to a better estimate of this proportion because we will not predict incompatible individuals (e.g. inhabitants in Belgium who live in Paris). Besides, we could apply ex-post the compatibility constraints in the one-step procedure. However, this would lead to a poorer estimate of, for example, the mean gross income given the dummy variables because the ex-post adjustment would only affect the dummies. This kind of bias will not appear in our two-step methodology because the predictions of the categorical variables in the first step are used to predict the continuous ones.

4. Comparison of Methodologies

In this section, we analyze the performance of the two statistical matching methods based on KCCA and Super-OM. In particular, we compare their performance with the ones of the more trivial extensions based on CCA, but also with common methods applied in econometrics (HD and REG).

For this purpose, we use the 2017 Belgian Statistics on Income and Living Conditions (SILC), from which we selected two subsets of common and non-common variables (see Tables 1 and 2, respectively). We divide the data set in five folds and, in each of them, we impute the non-common variables, as if they are missing. The remaining four-fifths are used to tune, by a 5-fold cross-validation procedure (see algorithm 4), the hyperparameters associated to the different machine learning algorithms.

All the methodologies applied in this Section (except HD) are implemented using the two-step procedure presented in Section 3.3. In the first step, we minimize the sum of the weighted Misclassification Rates (wMCR) of the categorical non-common variables, which have been transformed into dummies. While, in the second step, we minimize the sum of the Root weighted standardized Mean Squared Errors (RwsMSE) of the continuous non-common variables. These error measures are defined as:

$$\text{wMCR}\left(\mathbf{y}^d, \hat{\mathbf{y}}^d\right) = \sum_{i=1}^{n} w_i I\left(y_i^d \neq \hat{y}_i^d\right) \tag{6}$$

and

$$\text{RwsMSE}\left(\mathbf{y}^c, \hat{\mathbf{y}}^c\right) = \sqrt{\sum_{i=1}^{n} w_i \frac{\left(y_i^c - \hat{y}_i^c\right)^2}{\hat{\sigma}^2}}, \tag{7}$$

Algorithm 3 Final algorithm with KCCA

Input: $\mathbf{X}^A$ and $\mathbf{Y}^A$, the centered common and non-common variables in data set A and $\mathbf{X}^B$, the common variables in data set B. $\mathbf{w}^A$ and $\mathbf{w}^B$ the weights in both data sets.

Output: A data set B with the centered $\mathbf{X}^B$ and $\widehat{\mathbf{Y}}^B$.

for $i = 1, \cdots, n^B$ **do**
  for $j = 1, \cdots, n^A$ **do**
    Calculate $\Theta_{ij} = \mathbf{1}\left(\left(X^A_{ca}\right)_i = \left(X^B_{ca}\right)_j\right)$.
  end for
  Set $k = 1$
  while $\Theta_{i\cdot} = \overrightarrow{0}$ **do**
    for $j = 1, \cdots, n^A$ **do**
      Calculate $\Theta_{ij} = \mathbf{1}\left(\left(X^A_{ca}\right)_i \overset{d_{ca}-k}{=} \left(X^B_{ca}\right)_j\right)$ where $\overset{d_{ca}-k}{=}$ denotes the fact that the equality must be satisfied on $d_{ca}-k$ components.
    end for
    Set $k = k + 1$.
  end while
end for
Calculate $\boldsymbol{\alpha}_{ca}$ and $\boldsymbol{\beta}_{ca}$ of the KCCA between X^A and Y^A_{ca} using $\mathbf{w}^A$.
Calculate the kernel canonical variables $\mathbf{U}^A = \mathbf{K}^A_x\boldsymbol{\alpha}_{ca}$ and $\mathbf{U}^B = \mathbf{K}^B_x\boldsymbol{\alpha}_{ca}$.
for $i = 1, \cdots, n^B$ **do**
  for $j = 1, \cdots, n^A$ **do**
    Calculate $\varpi_{ij} = K_h\left(u^B_i, u^A_j\right)$ where K_h is a Gaussian kernel.
  end for
end for
for $i = 1, \cdots, n^B$ **do**
  Draw a $\left(\widehat{\mathbf{Y}}^B_{ca}\right)_i$ from $\mathbf{Y}^A_{ca}$ with the probability $\tilde{\boldsymbol{\omega}}_i$.
end for
Create the matrices $\widetilde{\mathbf{X}^A} = \left(\mathbf{X}^A, \mathbf{Y}^A_{ca}\right)$ and $\widetilde{\mathbf{X}^B} = \left(\mathbf{X}^B, \widehat{\mathbf{Y}}^B_{ca}\right)$.
Calculate $\boldsymbol{\alpha}_{co}$ and $\boldsymbol{\beta}_{co}$ of the KCCA between $\widetilde{\mathbf{X}^A}$ and $\mathbf{Y}^A_{co}$ using $\mathbf{w}^A$.
Calculate the kernel canonical variables $\widetilde{\mathbf{U}^A} = \widetilde{\mathbf{K}^A_x}\boldsymbol{\alpha}_{co}$ and $\widetilde{\mathbf{U}^B} = \widetilde{\mathbf{K}^B_x}\boldsymbol{\alpha}_{co}$.
for $i = 1, \cdots, n^B$ **do**
  for $j = 1, \cdots, n^A$ **do**
    Calculate $\varpi_{ij} = K_h\left(\widetilde{u}^B_i, \widetilde{u}^A_j\right)$ where K_h is a Gaussian kernel.
  end for
end for
Calculate $\widehat{\mathbf{Y}}^B_{co} = \boldsymbol{\Omega}\mathbf{Y}^A_{co}$.
$\widehat{\mathbf{Y}}^B = \left(\widehat{\mathbf{Y}}^B_{ca}, \widehat{\mathbf{Y}}^B_{co}\right)$.

Algorithm 4 Algorithm with 5-fold cross-validation

Input: $\mathbf{X}^A$ and $\mathbf{Y}^A$, the centered common and non-common variables in data set A and $\mathbf{X}^B$, the common variables in data set B. $\mathbf{w}^A$ and $\mathbf{w}^B$ the weights in both data sets.

Output: A data set B with the centered $\mathbf{X}^B$ and $\widehat{\mathbf{Y}}^B$.

Split A in five folds noted $A_i, i = 1, \cdots, 5$.
for $p = 1, \cdots, n^p_{ca}$ where n^p is the number of hyperparameters to test **do**
  for $i = 1, \cdots, 5$ **do**
    Use the four other folds to predict $\mathbf{Y}^{A_i}$
  end for
  $wMCR_p = \text{wMCR}\left(\mathbf{Y}^{A_i}_{ca}, \widehat{\mathbf{Y}}^{A_i}_{ca}\right)$
end for
$k_{ca} \leftarrow min\left\{wMCR_p\right\}$
Use the combination k_{ca} of hyperparameters to predict the categorical variable $\widehat{\mathbf{Y}}^B_{ca}$
for $p = 1, \cdots, n^p$ where n^p_{co} is the number of hyperparameters to test **do**
  for $i = 1, \cdots, 5$ **do**
    Use the four other folds to predict $\mathbf{Y}^{A_i}_{co}$ as $\widehat{\mathbf{Y}}^{A_i}_{co}$.
  end for
  $RwsMS\,E_p = \text{RwsMSE}\left(\mathbf{Y}^{A_i}_{co}, \widehat{\mathbf{Y}}^{A_i}_{co}\right)$
end for
$k_{co} \leftarrow min\left\{RwsMS\,E_p\right\}$
Use the combination k_{co} of hyperparameters to predict the categorical variable $\widehat{\mathbf{Y}}^B_{co}$
$\widehat{\mathbf{Y}}^B = \left(\widehat{\mathbf{Y}}^B_{ca}, \widehat{\mathbf{Y}}^B_{co}\right)$.

where n is the total number of observations, w_i is the sample weight (bounded between zero and one), $I(.)$ is an indicator function, $\hat{y}^d_i$ is the imputed value of a true dummy variable y^d_i, $\hat{y}^c_i$ is the imputed value of a true continuous variable y^c_i with $i = 1, \cdots, n$, and

$$\hat{\sigma}^2 = \sum_{i=1}^{n} w_i \left[y^c_i - \left(\sum_{i=1}^{n} w_i y^c_i\right)\right]^2 .$$

In order to tune the different hyperparameters used by the machine learning algorithms (CCA, KCCA and Super-OM), we mainly used a grid search, as described hereafter.

CCA and KCCA. In both procedures, we consider two-dimensional canonical variables. Their hyperparameters are tuned by 5-fold cross-validation and grid search. For CCA, we have only one hyperparameter in each phase, the bandwidth h; we use in both phases a grid of forty-nine values on the interval $[0.01, 0.25]$. In Table 3, we list the different hyperparameters for KCCA and the intervals used to find their optimal values[2].

Super-OM. In Steps 1 and 2, we consider two Super-OM models, both with a square grid of neurons. Their hyperparameters $\mathcal{K}$ (the number of neurons) and δ (as in formula 4) are tuned by 5-fold cross-validation. In Table 4, we list the possible values used to find the optimal ones in the two steps.

[2]Some of these intervals have been chosen after several trials.

Variable	Description	Type
RB090	Gender	Categorical
RX010	Age	Continuous
DB040	Region	Categorical
PE040	Educational level attained (ISCED level)	Categorical
PB190	Marital status	Categorical
PB220A	Country of citizenship	Categorical
PB210	Country of birth	Categorical
PL031	Activity status	Categorical
PL031	Number of hours worked per week	Categorical
PL140	Type of work contract	Categorical
PL111	Economic activities in employment	Categorical
PL040	Status in employment	Categorical
PL051	Occupation status (ISCO-08)	Categorical
HX060	Type of the household	Categorical
HX040	Size of the household	Continuous
HS110	Car ownership	Categorical

Table 1: Common variables

Variable	Description	Type
PY010N	Employee cash or near cash income (Net income)	Continuous
PY030G	Employer's social insurance contribution (Income)	Continuous
PY100N	Old-age benefits (Net income)	Continuous
RX050	Low work intensity status	Categorical
PB200	Consensual union	Categorical
PH030	Limitation in activities because of health problems	Categorical

Table 2: Non-common variables

HD. We use the distance hot-deck algorithm implemented in the R package *StatMatch*. The donation classes are created using three variables: Gender, Region and Marital status, whereas the distance is calculated using Age and Size of the household. For each individual in the receiver data set, the Manhattan distances to all the individuals in the donor data set that are in the same class are calculated. Then, the non-common variables of one of the $\sqrt{N_D + 1}$ closest individuals, where N_D is the number of available donors, is picked up at random and imputed in the receiver data set.

4.1. Results

The performance of the different methodologies used to impute the missing values is measured by the weighted standardized Mean Absolute Error (wsMAE) and Root weighted standardized Mean Squared Error (RwsMSE), for the continuous variables, and wMCR, for the categorical variables. The wMCR and the RwsMSE are defined in equation 6 and 7 respectively, while

$$\text{wsMAE}\left(\mathbf{y}^c, \hat{\mathbf{y}^c}\right) = \sum_{i=1}^{n} w_i \left| \frac{y_i^c - \hat{y}_i^c}{\hat{\sigma}} \right|,$$

where n is the total number of observations, w_i is the sample weight (bounded between zero and one), $\hat{y}_i^c$ is the imputed value of a true continuous variable y_i^c with $i = 1, \cdots, n$, and

$$\hat{\sigma}^2 = \sum_{i=1}^{n} w_i \left[y_i^c - \left(\sum_{i=1}^{n} w_i y_i^c \right) \right]^2 .$$

As a measure of the overall quality of the matching, we computed the mean of the RwsMSE of all variables (after the transformation of the categorical ones into dummies).

We also compute two multivariate coefficients of determination. The first is the one defined by (Cohen et al., 2002), based on the generalized variance (i.e. determinant of covariance matrix):

$$\text{mult-}R_1^2 = 1 - \frac{\det\left(R_{Y\hat{Y}}\right)}{\det\left(R_Y\right)\det\left(R_{\hat{Y}}\right)}$$

where $R_{Y\hat{Y}}$ is the full correlation matrix of the variables in Y and $\hat{Y}$ and R_Y (resp. $R_{\hat{Y}}$) is the correlation matrix of the variables in Y (resp. $\hat{Y}$). The second, with a geometric interpretation, is the one defined by (Jones, 2019):

$$\text{mult-}R_2^2 = 1 - \frac{SSE}{SST},$$

where $\text{SST} = \sum_{i=1}^{n} w_i \left[d\left(y_i, \bar{y}\right)\right]^2$, $\text{SSE} = \sum_{i=1}^{n} w_i \left(d\left(y_i, \hat{y}_i\right)\right)^2$ and $d(p,q) = \sqrt{\sum_{j=1}^{J} \left(p_j - q_j\right)^2}$.

Step	Hyperparameter	Interval
1	Bandwidth h of the kernel (prediction)	$5 \cdot 10^{-3} - 3.5 \cdot 10^{-2}$
	Bandwidth $0.5 \cdot h_x^{-2}$ of the kernel in $\mathbf{K}_x$	$4 \cdot 10^{-4} - 1.2 \cdot 10^{-3}$
	Bandwidth $0.5 \cdot h_y^{-2}$ of the kernel in $\mathbf{K}_y$	$4 \cdot 10^{-4} - 1.2 \cdot 10^{-3}$
	Regularization parameter γ	$1 \cdot 10^{-5} - 3 \cdot 10^{-5}$
2	Bandwidth h of the kernel (prediction)	$1 \cdot 10^{-2} - 7 \cdot 10^{-2}$
	Bandwidth $0.5 \cdot h_x^{-2}$ of the kernel in $\mathbf{K}_x$	$1.4 \cdot 10^{-3} - 2.2 \cdot 10^{-3}$
	Bandwidth $0.5 \cdot h_y^{-2}$ of the kernel in $\mathbf{K}_y$	$1 \cdot 10^{-4} - 1.6 \cdot 10^{-3}$
	Regularization parameter γ	$1 \cdot 10^{-5} - 3 \cdot 10^{-5}$

Table 3: Hyperparameters for KCCA

Step	Parameter	Values
1 & 2	Grid size ($\mathcal{K}$)	$3 \times 3,\ 4 \times 4,\ \ldots,\ 20 \times 20$
	Layer weight (δ)	$0.05,\ 0.1,\ \ldots,\ 0.9,\ 0.95$

Table 4: Hyperparameters for Super-OM.

Finally, we use the Cramér–von Mises criterion to assess the goodness of fit of all bivariate distributions. Theoretically, it is defined as

$$\text{CVM} = \int_{\mathbb{R}} \int_{\mathbb{R}} \left[\hat{F}_{n_A}(x,y) - \hat{G}_{n_B}(x,y) \right]^2 \mathrm{d}H_{n_A+n_B}(x,y)$$

where $\hat{F}_{n_A}(x,y)$, $\hat{G}_{n_B}(x,y)$ and $H_{n_A+n_B}(x,y)$ are the bivariate empirical distributions of two variables X and Y in data set A, B and $A+B$, respectively.

In practice, we use an empirical Cramér–von Mises criterion where the double integrals are approximated by a sum,

$$\widetilde{\text{CVM}} = \frac{n_A + n_B}{2} \sum_{j \in \{A,B\}} \sum_{i=1}^{n_j} w_i^j \left[\hat{F}_{n_A}\left(x_i^j, y_i^j\right) - \hat{G}_{n_B}\left(x_i^j, y_i^j\right) \right]^2, \tag{8}$$

where w_i^A (resp. w_i^B) denotes the standardized weights such that the sum is equal to one of observations in data set A (resp. B).

The aforementioned criteria are used to assess the quality of the imputations obtained with the different statistical matching techniques in each of the five folds. Their averages over the five folds are provided in Table 5. In terms of wsMAE, RwsMSE, wMCR and the overall error measures (Total RwsMSE), HD underperforms all the other methods, as expected, due to its random nature. The errors of the continuous variables are in many cases twice as large as those produced by our proposed methodologies. The best performance is achieved by REG, followed by KCCA. In terms of the two multivariate coefficients of determination (mult-R^2), these conclusions are mitigated. However, if we look at the averages of the empirical Cramér–von Mises ($\widetilde{\text{CVM}}$) criteria, which measure the distances between empirical distributions (including dependencies between the considered variables), the conclusions are different. In Table 5, we provide the average of the $\widetilde{\text{CVM}}$ criteria for all couples of non-common variables, as well as all couples of one common and one non-common variable[3].

The statistical matching methods based on machine learning techniques (KCCA, Super-OM and CCA) exhibit lower values, meaning that they have a better performance than REG, with KCCA providing the best results.

HD also exhibits low values because it simply reproduces the dependencies of the non-common variables in the donor data set; KCCA provides the best performance if we consider the average of the $\widetilde{\text{CVM}}$ criteria between all combinations of one common and one non-common variable.

Figures 3–5 contain density plots comparing the original distributions of the continuous variables and the imputed ones(for the first fold, the graphs being similar for the others). In line with previous results, the distributions obtained from KCCA seem to be very close to the original ones; in particular, they seem to outperform CCA and REG.

[3]In the computation of $\widetilde{\text{CVM}}$, data set A and B (in eq. 8) contain the true data and the one obtained through statistical matching, respectively, for a given fold.

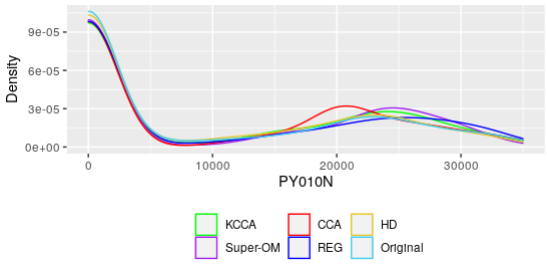

Figure 3: Density plot of variable PY010N for the first fold

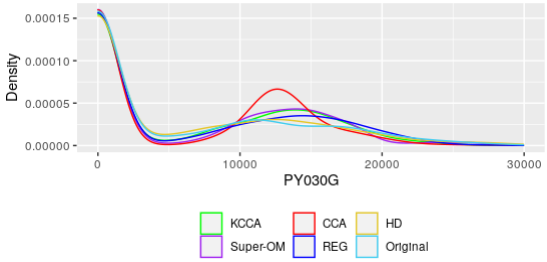

Figure 4: Density plot of variable PY030G for the first fold

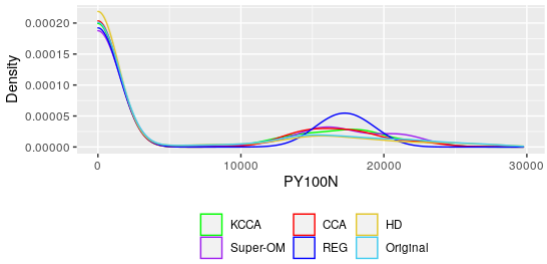

Figure 5: Density plot of variable PY100N for the first fold

In conclusion, the results indicate that our methodologies combine the advantages of both REG and HD. They seem better than REG in preserving the joint distribution, as measured by the empirical Cramér–von Mises criteria. At the same time, they also outperform HD in terms of prediction error.

Variable	Measure	KCCA	Super-OM	CCA	HD	REG
PY010N	wsMAE	0.23	0.25	0.25	0.64	0.22
PY010N	RwsMSE	0.59	0.63	0.62	1.46	0.58
PY030G	wsMAE	0.27	0.28	0.30	0.64	0.24
PY030G	RwsMSE	0.63	0.67	0.68	1.48	0.61
PY100N	wsMAE	0.20	0.20	0.21	0.29	0.20
PY100N	RwsMSE	0.58	0.57	0.60	0.77	0.57
RX050	wMCR	0.13	0.14	0.13	0.16	0.12
PB200	wMCR	0.14	0.15	0.13	0.19	0.13
PH030	wMCR	0.33	0.31	0.33	0.37	0.26
Total (cont.)	RwsMSE	0.60	0.62	0.63	1.23	0.58
Total (cat.)	RwsMSE	0.95	0.95	0.94	1.03	0.87
Total	RwsMSE	0.87	0.87	0.86	1.08	0.80
Total	Multivariate R_1^2 (Cohen, 2013)	0.99	0.99	0.99	0.97	0.99
Total	Multivariate R_2^2 (Jones, 2019)	0.63	0.59	0.59	-1.05	0.64
Total	Average $\widetilde{\text{CVM}}$ Bivariate Non-Common	1.18	2.43	1.78	0.67	9.85
Total	Average $\widetilde{\text{CVM}}$ Bivariate Mixed	0.67	1.39	0.94	0.82	5.42

Table 5: Results (average over five folds) of the different statistical matching techniques applied to the SILC data set only. wsMAE, weighted standardized Mean Absolute Error; RwsMSE, Root weighted standardized Mean Squared Error; wMCR, weighted Misclassification Rates; Total, all variables; Total (cont.), all continuous variables; Total (cat.), all categorical variables; Bivariate Non-Common, average of the $\widetilde{\text{CVM}}$ criteria for all couples of non-common variables; Bivariate Mixed, average of the $\widetilde{\text{CVM}}$ criteria for all couples of one common and one non-common variable.

5. Application

In this section, we apply the proposed statistical matching methods to integrate the Statistics on Income and Living Conditions (SILC) and the Household Budget Survey (HBS) data sets for Belgium in 2016. The objective is to impute the consumption variables (grouped into 10 macro-categories, see Table 7) from HBS into SILC. For this purpose, we employ 12 common variables which are available in both data sets (see Table 6) and the two-step procedure in Section 3.3. In the first step, we impute the binary variables indicating whether the associated expenditure is equal to zero or not. Then, in the second step, we impute the continuous variables taking into account the corresponding dummies imputed in the previous step.

Since we do not know the true values of the non-common variables in the receiver data set (SILC), we cannot calculate wsMAE, wsMAE, RwsMSE, wMCR and the multivariate R^2. For the $\widetilde{\text{CVM}}$ criteria, we do not know the empirical bivariate distributions of the common and non-common variables in the receiver data set, but for illustration purposes, we approximate them with the donor data set (HBS). In Table 8, we provide the averages of the $\widetilde{\text{CVM}}$ criteria for all couples of non-common variables, as well as all couples of one common and one non-common variable[4]. The results are consistent with the ones in the previous section (at least for non-common variables). KCCA provides the best results, and all the other proposed methods outperform REG. HD results are not displayed since our formula for the $\widetilde{\text{CVM}}$ criteria overrates their quality (roughly the difference between two similar distributions calculated on the same data set – donor).

Variable	Type
Gender	Categorical
Age	Continuous
Region	Categorical
Educational level attained (ISCED level)	Categorical
Marital status	Categorical
Activity status	Categorical
Type of work contract	Categorical
Status in employment	Categorical
Size of the household	Continuous
Number of Children	Continuous
Monthly imputed rent	Continuous
Total net income	Continuous

Table 6: Common variables

6. Conclusion

In this paper, we have extended several machine learning techniques to statistical matching: Kernel Canonical Correlation Analysis (KCCA), Super-Organizing Map (Super-OM).

[4]In the computation of $\widetilde{\text{CVM}}$, data set A and B (in eq. 8) are the donor (HBS) and the receiver (SILC), respectively.

	Variable	Type
1	Food products and non-alcoholic beverages	Continuous
2	Alcoholic beverages, tobacco, narcotics	Continuous
3	Clothing and footwear	Continuous
4	Housing, water, electricity, gas and other fuels	Continuous
5	Furnishing, household equipment and routine maintenance of the house	Continuous
6	Health	Continuous
7	Transport	Continuous
8	Communications	Continuous
9	Recreation and culture	Continuous
10	Education	Continuous
11	Restaurants and hotels (horeca)	Continuous
12	Miscellaneous goods and services	Continuous

Table 7: Non-common variables

	Non-Common	Mixed
KCCA	431.37	500.89
Super-OM	705.27	683.81
CCA	561.81	588.68
REG	962.63	918.48

Table 8: Fusion SILC-HBS. Average of the empirical Cramér-von Mises criteria ($\widetilde{\text{CVM}}$), as in eq. 8, for the different statistical matching techniques. Non-Common, average of the $\widetilde{\text{CVM}}$ criteria for all couples of non-common variables; Mixed, average of the $\widetilde{\text{CVM}}$ criteria for all couples of one common and one non-common variable.

First, we include sample weights in all the methodologies. Then, we propose a two-step procedure to deal with mixed data and incompatibilities between categorical variables. Finally, we compare the performance of the proposed methodologies with more trivial extensions such as Canonical Correlation Analysis (CCA), as well as more traditional econometric methods such as distance hot-deck (HD) and multivariate and multinomial regression (REG).

In a first exercise, the 2017 Belgian Statistics on Income and Living Conditions (SILC) data set is divided into five folds. In each fold, a set of variables is imputed as if they were missing on hand of the four remaining folds. This application has shown that the proposed methodologies are able to render competitive results, in particular for KCCA, and to harness the different advantages of both HD and REG methods: preserving the joint distributions and having small prediction errors. In a second application, we illustrate a set of consumption variables from the Household Budget Survey (HBS) into the SILC data set for Belgium in 2016. Once again, we observe that KCCA seems to perform very well.

The investigation of an efficient iterative procedure to integrate multiple data sets and the generation of fictitious data sets from the trained models (e.g. in case of confidential data) are promising paths for future research. In addition, given the high computational costs of KCCA in case of very large data sets, a bootstrap version of this procedure is worthy of further re-

search.

Compliance with ethical standards

Conflict of interest. The authors declare that they have no conflict of interest.

Appendix A. Initial weights in a two-stage stratification

The probability of selection of a new household can be calculated in a two-stage stratification procedure. It is the probability that a household is drawn given the primary sample unit (PSU, e.g. the municipality in Belgium) is drawn multiplied by the probability that the PSU is drawn. A PSU can be drawn several times so it leads us to the following formula :

$$\begin{aligned} P_h &= P\left(h \text{ drawn}\right) \\ &= P\left(h \text{ drawn}|X \text{ drawn}\right) \cdot P\left(X \text{ drawn}\right) \\ &= \frac{n_h}{N_X}\left(1-\left(1-\frac{N_X}{N_h}\right)^{g_h}\right) \end{aligned}$$

where :

- X denotes the PSU of h.
- n_h the number of households to be drawn in the (selected) PSU (40 by PSU for SILC in Belgium),
- N_X the number of households in the PSU,
- N_h the number of households in the stratum,
- g_h the number of PSU drawn in the stratum.

Because N_X is much smaller compared to N_h, a first order Taylor approximation can be used and :

$$\begin{aligned} P_h &= \frac{n_h}{N_X}\left(1-\left(1-\frac{N_X}{N_h}\right)^{g_h}\right) \\ &\approx \frac{n_h}{N_X}\frac{N_X}{N_h}g_h \\ &\approx \frac{n_h}{N_h}g_h. \end{aligned}$$

This calculation does not take into account the fact that when a household is drawn, it is immunized to be drawn again. Generally, given the number of households per PSU, the difference is negligible and neglected in practice. The initial weight of a household is just equal to the inverse of this probability.

Appendix B. Adjustment of the weights

Appendix B.1. Adjustment for non-response

The weights are then adjusted for non-response in one of following ways.

The classical idea is, if we have a household h that belongs to a group k, :

$$w_h^{(n)} = w_h \frac{1}{R_k}$$

where

$$R_k = \frac{\text{sum of design weights of responding units in cell k}}{\text{sum of design weights of selected units in cell k}}.$$

A second possibility is to use a logit regression where response propensities R_h is estimated using some variables that are available for every individuals. If a large number of variables are available for all households, i.e. both those who responded and those who did not, this type of model will give better results. For instance, in Belgium for SILC, Statbel uses a multiple logit regression model based on the province, the household size, the household type, the urbanity of PSU, and the fiscal income quantile of the household. Then the adjusted weights using that second technique become :

$$w_h^{(n)} = w_h \frac{1}{R_h}.$$

Appendix B.2. Final Adjustment

Then, the final weights are obtained by adjusting the weights $w_h^{(n)}$ to reproduce characteristics from the sample population. We suppose that there exist J auxiliary variables $x_1, \cdots x_j \cdots x_J$ (age, gender, region,... where categorical variables are transformed in dummies). The individual variables are aggregated at household level (number of men, number of women, ...)

For SILC, the calibration of (Deville and Särndal, 1992) is used with only dummy variables. It consists in minimizing:

$$\min_{w_k^{(f)}} \sum_{k=1}^{S} w_k^{(n)} G\left(\frac{w_k^{(f)}}{w_k^{(n)}}\right) \text{ such that } \sum_{k=1}^{S} w_k^{(f)} x_{jk} = X_j \forall j \in \{1, \cdots, J\},$$

where X_j is the total of the variable x_j in the population and S is the sample size.

Several functions can be chosen for G. For example, Statbel in Belgium, uses for SILC :

$$G(r) = \begin{cases} \frac{(1-L)(U-1)}{U-L}\left[(r-L)\log\left(\frac{r-L}{1-L}\right)+(U-r)\log\left(\frac{U-r}{U-1}\right)\right] & \text{if } r \in]L, U[\\ +\infty & \text{otherwise} \end{cases}$$

where they have to fix a lower bound L and an upper bound U.

The choice of this function is adapted to dummy variables. It is a truncated raking ratio. Raking ratio is developed by (Deming and Stephan, 1940) and is a method that is equivalent to the method when $G(r) = r\log(r) - r + 1$ as explained in (Deville and Särndal, 1992). In truncated raking ratio, we impose that

$$L \leq \frac{w_k^{(f)}}{w_k^{(n)}} \leq U.$$

The purpose of making this truncated version is to restrict the weights to avoid extremely large values compared to original weights while maintaining the advantages of the raking ratio, i.e. avoid having negative weights (if $L \geq 0$.). Moreover, this method always leads to a solution as the raking ratio.

References

Akaho, S., 2001. A kernel method for canonical correlation analysis. arXiv:0609071. arXiv preprint arXiv:0609071.

Aluja-Banet, T., Daunis-i Estadella, J., Pellicer, D., 2007. Graft, a complete system for data fusion. Comput Stat Data Anal 52, 635–649. doi:https://doi.org/10.1016/j.csda.2006.11.029.

Anderson, T.W., 1957. Maximum likelihood estimates for a multivariate normal distribution when some observations are missing. J Am Stat Assoc 52, 200–203. doi:https://doi.org/10.1080/01621459.1957.10501379.

Bach, F.R., Jordan, M.I., 2002. Kernel independent component analysis. J Mach Learn Res 3, 1–48. doi:https://doi.org/10.1109/ICASSP.2003.1202783.

Cohen, J., Cohen, P., West, S.G., Aiken, L.S., 2002. Applied multiple regression/correlation analysis for the behavioral sciences. Routledge. doi:https://doi.org/10.4324/9780203774441.

Conti, P.L., Marella, D., Scanu, M., 2017. How far from identifiability? a systematic overview of the statistical matching problem in a non parametric framework. Commun. Stat. - Theory Methods 46, 967–994. doi:https://doi.org/10.1080/03610926.2015.1010005.

Cottrell, M., Letrémy, P., 2007. Missing values : processing with the Kohonen algorithm, in: Applied Stochastic Models and Data Analysis 2005, p. math/0701152. arXiv:math/0701152.

Deming, W.E., 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, 427–444.

Deville, J., Särndal, C., 1992. Calibration approach estimators in sampling.

Donatiello, G., D'Orazio, M., Frattarola, D., Rizzi, A., Scanu, M., Spaziani, M., 2014. Statistical matching of income and consumption expenditures. Int J Econ Sci 3, 50.

D'Orazio, M., Di Zio, M., Scanu, M., 2006a. Statistical matching for categorical data: Displaying uncertainty and using logical constraints. J Off Stat - Stockh 22, 137.

D'Orazio, M., Di Zio, M., Scanu, M., 2006b. Statistical matching: Theory and practice. John Wiley & Sons. doi:https://doi.org/10.1002/0470023554.

Fessant, F., Midenet, S., 2002. Self-organising map for data imputation and correction in surveys. Neural Comput Appl 10, 300–310. doi:https://doi.org/10.1007/s005210200002.

Folguera, L., Zupan, J., Cicerone, D., Magallanes, J.F., 2015. Self-organizing maps for imputation of missing data in incomplete data matrices. Chemom Intell Lab Syst 143, 146 – 151. URL: http://www.sciencedirect.com/science/article/pii/S016974391500060X, doi:https://doi.org/10.1016/j.chemolab.2015.03.002.

Fosdick, B.K., DeYoreo, M., Reiter, J.P., et al., 2016. Categorical data fusion using auxiliary information. Ann Appl Stat 10, 1907–1929. doi:https://doi.org/10.1214/16-AOAS925.

Hotelling, H., 1936. Relations between two sets of variates. Biometrika doi:https://doi.org/10.2307/2333955.

Jones, T., 2019. A coefficient of determination for probabilistic topic models. arXiv:1911.11061. arXiv preprint arXiv:1911.11061.

Kamakura, W.A., Wedel, M., 1997. Statistical data fusion for cross-tabulation. J Mark Res 34, 485–498. doi:https://doi.org/10.2307/3151966.

Kim, J.K., Shao, J., 2013. Statistical methods for handling incomplete data. Chapman and Hall/CRC, New York. doi:https://doi.org/10.1201/b13981.

Kohonen, T., 1982. Self-organized formation of topologically correct feature maps. Biol Cybern 43, 59–69. doi:https://doi.org/10.1007/BF00337288.

Kuss, M., Graepel, T., 2003. The geometry of kernel canonical correlation analysis. Technical Report. Max Planck Institute for Biological Cybernetics.

Lai, P.L., Fyfe, C., 2000. Kernel and nonlinear canonical correlation analysis. Int J Neural Syst 10, 365–377. doi:https://doi.org/10.1142/s012906570000034x.

López-Laborda, J., Marín-González, C., Onrubia-Fernández, J., 2020. Estimating engel curves: a new way to improve the silc-hbs matching process using glm methods. J Appl Stat 47, 1–18. doi:https://doi.org/10.1080/02664763.2020.1796933.

Melzer, T., Reiter, M., Bischof, H., 2001. Nonlinear feature extraction using generalized canonical correlation analysis, in: International Conference on Artificial Neural Networks, Springer. pp. 353–360. doi:https://doi.org/10.1007/3-540-44668-0_50.

Mitsuhiro, M., Hoshino, T., 2020. Kernel canonical correlation analysis for data combination of multiple-source datasets. Jpn J Stat Data Sci 3, 1–18. doi:https://doi.org/10.1007/s42081-020-00074-z.

Okner, B., 1972. Constructing a new data base from existing microdata sets: the 1966 merge file, in: Annals of Economic and Social Measurement, Volume 1, number 3. NBER, pp. 325–362.

Rässler, S., 2002. Statistical matching: A frequentist theory, practical applications, and alternative Bayesian approaches. volume 168. Springer Science & Business Media. doi:https://doi.org/10.1007/978-1-4613-0053-3.

Saverio, G., Romano, M.C., Gianni, C., Di Zio, M., Marcello, D., Federica, P., Mauro, S., TORELLI, N., 2008. Time Use and Labour Force: a proposal to integrate the datathrough statistical matching. Technical Report. Istat-Produzione libraria e centro stampa.

Schölkopf, B., Smola, A., Müller, K.R., 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural computation 10, 1299–1319.

Serafino, P., Tonkin, R., 2017. Statistical matching of European Union statistics on income and living conditions (EU-SILC) and the household budget survey. Technical Report. Eurostat: Statistical Working Papers. Luxembourg: Publications Office of the European Union. doi:https://doi.org/10.2785/933460.

Shimodaira, H., 2014. A simple coding for cross-domain matching with dimension reduction via spectral graph embedding. ArXiv preprint arXiv:1412.8380.

Spaziani, M., Frattarola, D., D'Orazio, M., 2019. Integration of survey data in r based on machine learning. Romanian Stat Rev 2019, 5–16. doi:https://doi.org/10.13140/RG.2.2.14022.93762.

Tonkin, R., Webber, D., 2012. Statistical matching of eu-silc and household budget survey to compare poverty estimates using income, expenditures and material deprivation, in: EU-SILC International Conference, Vienna, pp. 6–7. doi:https://doi.org/10.2785/4151.

Van Buuren, S., 2018. Flexible Imputation of Missing Data. Chapman and Hall/CRC, New York. doi:https://doi.org/10.1201/9780429492259.

Wehrens, R., Buydens, L., 2007. Self- and super-organizing maps in r: The kohonen package. J Stat Softw 21, 1–19. doi:https://doi.org/10.18637/jss.v021.i05.

Wehrens, R., Kruisselbrink, J., 2018. Flexible self-organizing maps in kohonen 3.0. J Stat Softw 87, 1–18. doi:https://doi.org/10.18637/jss.v087.i07.