Data Min Knowl Disc (2011) 23:300–321
DOI 10.1007/s10618-010-0205-7

# Matching samples of multiple views

**Abhishek Tripathi · Arto Klami · Matej Orešič · Samuel Kaski**

Received: 23 July 2009 / Accepted: 28 October 2010 / Published online: 28 November 2010

**Abstract** Multi-view learning studies how several views, different feature representations, of the same objects could be best utilized in learning. In other words, multi-view learning is analysis of co-occurrence data, where the observations are co-occurrences of samples in the views. Standard multi-view learning such as joint density modeling cannot be done in the absence of co-occurrence, when the views are observed separately and the identities of objects are not known. As a practical example, joint analysis of mRNA and protein concentrations requires mapping between genes and proteins. We introduce a data-driven approach for learning the correspondence of the observations in the different views, in order to enable joint analysis also in the absence of known co-occurrence. The method finds a matching that maximizes statistical dependency between the views, which is particularly suitable for multi-view

Responsible editor: Johannes Gehrke.

A. Tripathi (✉)
Helsinki Institute for Information Technology HIIT, Department of Computer Science,
University of Helsinki, Helsinki, Finland
e-mail: abhishek.tripathi@cs.helsinki.fi

A. Klami
Helsinki Institute for Information Technology HIIT, Department of Information and Computer Science,
Aalto University, Helsinki, Finland
e-mail: arto.klami@tkk.fi

M. Orešič
Quantitative Biology and Bioinformatics, VTT Technical Research Centre of Finland, Espoo, Finland
e-mail: matej.oresic@vtt.fi

S. Kaski
Helsinki Institute for Information Technology HIIT, Aalto University and University of Helsinki,
Helsinki, Finland
e-mail: samuel.kaski@tkk.fi

methods such as canonical correlation analysis which has the same objective. We apply the method to translational metabolomics, to identify differences and commonalities in metabolic processes in different species or tissues. The metabolite identities and roles in the different species are not generally known, and it is necessary to search for a matching. In this paper we show, using different metabolomics measurement batches as the views so that the ground truth is known, that the metabolite identities can be reliably matched by a consensus of several matching solutions.

**Keywords** Bipartite matching · Canonical correlation · Consensus matching · Co-occurrence data · Multi-view learning

## 1 Introduction

Multi-view learning considers the task of learning from two or more data sets with co-occurring observations. Intuitively, using all of the views for learning is beneficial, regardless of the learning task. The basic approach for using the views would be to build a model for the joint representations of all views, for example a hierarchical Bayesian model, but recently more targeted multi-view learning methods have also been introduced. Bickel and Scheffer (2005) maximize a consensus between models learned from each view, and Klami and Kaski (2008); Rogers et al. (2010) build hierarchical models specifically designed to capture statistical dependencies between the views in a latent variable representation. The latter line of work provides a generative alternative for canonical correlation analysis that is also applicable for the same task. Examples of multi-view learning applications include cross-lingual text mining and machine translation (Li and Shawe-Taylor 2006), multimodal information retrieval (Farquhar et al. 2006), modeling joint collections of text and images (Blei and Jordan 2003), and integration of mRNA and protein expression measurements in systems biology (Rogers et al. 2008).

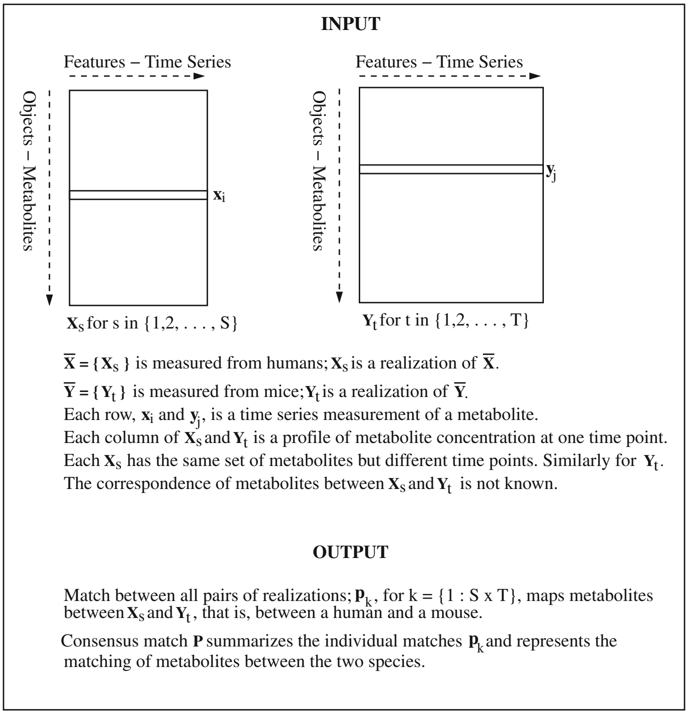
The traditional multi-view learning methods require strict co-occurrence. That is, the views must have known one-to-one matching of samples. Text analysis can be done for sentence-aligned corpora, images must be paired with their captions in multimodal retrieval, and mRNA and protein expressions can be analyzed jointly only if we know which protein corresponds to which mRNA sequence. Strict co-occurrence holds for many applications, but not for all. While the amount of bilingual text is abundant, for instance through the world wide web, the alignment of such corpora is not always known. Aligning such documents in two languages, for instance at sentence-level (Melamed 1999; Barzilay and Elhadad 2003), would provide useful resources for machine translation. In systems biology, the views may be measurements made with different technologies, for instance, different brands of microarrays use different probe sets to measure the activities of the same set of genes. The correspondence of probes between different microarray brands is not always known. Nevertheless, it can be assumed that the views do in fact measure mostly the same objects, we just do not know which object corresponds to which in the other view. It would be useful to use various multi-view learning techniques for these kinds of applications as well, which requires learning the correspondence from the data. In this paper we propose an algorithm for learning the correspondence or match between the objects of two views.

In a specific simplified setting solving the matching is easy in principle: If the two views are known to be independent replications of the same measurements, then we can directly compare the vectorial representations of the objects in the two views. A natural solution is to match objects whose representations are similar. The only complication then is in satisfying constraints of the matching, such as one-to-one property: each object in one view should match exactly one object in the other. With this particular restriction the task reduces to weighted bi-partite graph matching or linear assignment problem with costs stemming from a distance measure between the representations. Kuhn (1955) provides a classical algorithm that solves the problem, and Burkard et al. (2009) give a very recent extensive treatment on assignment problems in general.

The matching problem gets considerably more challenging when the views are not replications and do not share a common representation. They can either consist of different sets of features used to characterize the objects, or even be completely different domains: one view could be a time series of gene expression, while the other could be the gene sequence. This type of applications are the primary motivations for modeling dependencies between multiple views, and are routinely addressed in analysis of co-occurrence data (for example, Vinokourov et al. (2003) used kernelized CCA for learning a mapping between image descriptions and content that have very different representations), but tools for applications without co-occurrence are missing. We introduce methods for this general case, based on finding a representation where the views can be compared. If such a representation can be extracted for each of the views, then we can use the simple bi-partite graph matching solution for solving the correspondence. The problem is thus reduced to finding a good representation, a representation where comparing the measurements is sensible.

In our preliminary work (Tripathi et al. 2008, 2009), done simultaneously and independently of a similar method presented by Haghighi et al. (2008), the representations were sought using the following idea: The two views can be compared in a representation that maximally captures the statistical dependencies between the two views. The underlying idea is to capture all the variation in a view that is shared with the other view, and use this shared representation for comparisons, whereas variation independent of the other view may be ignored. Measures of dependency between the views, such as correlation or mutual information, require matched sets of objects, which leads naturally to an iterative alternating algorithm. Given any matching we can find the best possible representation by maximizing the dependency, and given the representation we can find the best matching by solving the assignment problem. In this paper we present this approach in detail, with its connection to mutual information, and extend it.

Both Tripathi et al. (2008, 2009) and Haghighi et al. (2008) learn the matching from a single observation of the objects in each of the views. This is a well-defined optimization problem, yet the solution is likely to involve uncertainty; another realization of the feature representations would typically lead to a different match. It is infeasible to expect that a purely data-driven solution could learn an accurate and reliable matching given just a single observation. We further extend the methods to more realistic applications where each view is represented by multiple independent realizations, and learn the matching given all the available data, as a consensus of the individual matches.



$\overline{\mathbf{X}} = \{\mathbf{X}_s\}$ is measured from humans; $\mathbf{X}_s$ is a realization of $\overline{\mathbf{X}}$.
$\overline{\mathbf{Y}} = \{\mathbf{Y}_t\}$ is measured from mice; $\mathbf{Y}_t$ is a realization of $\overline{\mathbf{Y}}$.
Each row, $\mathbf{x}_i$ and $\mathbf{y}_j$, is a time series measurement of a metabolite.
Each column of $\mathbf{X}_s$ and $\mathbf{Y}_t$ is a profile of metabolite concentration at one time point.
Each $\mathbf{X}_s$ has the same set of metabolites but different time points. Similarly for $\mathbf{Y}_t$.
The correspondence of metabolites between $\mathbf{X}_s$ and $\mathbf{Y}_t$ is not known.

Match between all pairs of realizations; $\mathbf{p}_k$, for k = {1 : S x T}, maps metabolites between $\mathbf{X}_s$ and $\mathbf{Y}_t$, that is, between a human and a mouse.

Consensus match $\mathbf{P}$ summarizes the individual matches $\mathbf{p}_k$ and represents the matching of metabolites between the two species.

**Fig. 1** Mapping between the abstract terminology used in the paper and the application of translational metabolomics. The task is to learn a *consensus match* between metabolite identities of humans and mice, *pairing* each human metabolite with one mouse metabolite. The consensus match is found by combining *individual matches* of several *realizations* of the two species. Each realization is a data matrix measuring the metabolic activity of a single individual, human or mouse. The rows of the data matrices correspond to the *objects* being matched, in this case the metabolites. The columns, in turn, are *features* that are used for learning the match, and they are the metabolic concentrations at different time points

To clarify the terminology, we use the term *realization* to represent a data matrix, and a *match* or *matching* refers to a one-to-one correspondence computed for the samples of two such data matrices. The terminology is illustrated in Fig. 1 that maps the terms to the application introduced below. Each row of the data matrix is an *object* and the columns are *features* that are typically different for the views and also for different realizations. Furthermore, we use the term *pair* to represent a single pair of objects, one for each view. In other words, a match consists of a set of pairs, and a consensus can be learned from a collection of matches computed from several realizations.

One of the most promising application fields of the matching methods is in translation of findings from model organisms, such as mice, to men. There it is crucial to know which properties of the model organisms generalize to men and which do not. Metabolites, i.e. small molecules in cells and biofluids, are common across species and thus provide a best chance to find translational biomarkers, as has been previously demonstrated in metabolic syndrome (Damian et al. 2007). Comparative metabolome analysis is commonly performed by mass spectrometry, and comparisons of metabolic profiles require solving the matching problem between the metabolites in the two profiles for two reasons: the identities are not clear due to various technical reasons of the measurement process, and the functions of the metabolites may be different in the different tissues or species.

In a typical metabolomic experiment we will have metabolite concentrations for a collection of humans and mice, and the task would be to learn a global match between the two organisms, not only between any two individuals. The individual matching solutions between any two individuals are samples of the complete problem. We introduce a computationally feasible solution by learning the individual matching solutions for sufficiently many realizations of both views, and then combining the matching results to find a global consensus. Coincidentally, the global consensus can be found by again solving an assignment problem, this time applied to a matrix of co-occurrences in individual matching solutions instead of comparison costs. In addition to finding the global match, we suggest how alternative matchings can be inferred based on the collection of the independent matching results.

We start by solving a translational metabolomics problem where we know the ground truth. By matching the metabolites of two populations of the *same* species, here humans, we can evaluate how well the known matching is revealed. Even this problem is far from trivial because of differences in individuals (and their diets), and technical differences in measurement batches. We can then move to the biologically more interesting problem of translation between humans and mice in follow-up works.

It is worth noticing that our solution for combining the individual matching solutions does not make any assumptions about how the individual matchings have been computed; the matching algorithm could be replaced with any other method. One recent alternative would be the kernelized sorting method by Quadrianto et al. (2009), which translates within-view distances to between-view distances using the Hilbert-Schmidt independence criterion (HSIC; Smola et al. 2007). This results in a quadratic assignment problem, which Quadrianto et al. (2009) address by iteratively solving a linear assignment problem. The main difference is that our approach explicitly finds the representations that can then be analyzed to understand the relationship between the views, whereas kernelized sorting is less transparent but it can more straightforwardly use non-linear kernels.

The paper is organized as follows: Sect. 2 describes the matching problem and our dependency-based solution for it, whereas Sect. 3 discusses alternative approaches for the same problem. Section 4 then introduces the concept of consensus matching and presents an algorithm for finding it based on any of the alternatives for finding individual matchings. Finally, in Sect. 5 we apply the algorithm to a simplified translational metabolomic experiment with known ground truth, and conclude the article by discussion in Sect. 6.

## 2 The matching problem

Given two data sets or views $\mathbf{X} \in \mathbb{R}^{N \times D_x}$ and $\mathbf{Y} \in \mathbb{R}^{M \times D_y}$, $M \geq N$, we want to find a permutation $\mathbf{p}$ of the objects in $\mathbf{Y}$ such that the $i$th row (object) $\mathbf{x}_i$ in $\mathbf{X}$ is paired with the row $\mathbf{y}_{\mathbf{p}(i)}$ in $\mathbf{Y}$. In other words, we assume that each object or observation in $\mathbf{X}$ is paired with exactly one object in $\mathbf{Y}$, while $\mathbf{Y}$ may also have objects that will be left without a pair. The matching will be primarily based on the actual data vectors, although prior information on matchings can be included as will be explained later.

If the two views can be assumed to be replicates (e.g., repeated measurements with the same sensor, or biological replicates in a gene expression study) of each other, the observations $\mathbf{x}$ and $\mathbf{y}$ lie in the same data space. Then it makes sense to assume that the distance $d(\mathbf{x}_i, \mathbf{y}_j)$ between $\mathbf{x}_i$ and $\mathbf{y}_j$ is a measure on the likelihood of $\mathbf{y}_j$ matching $\mathbf{x}_i$; the smaller the distance the more likely the two objects correspond to each other. Finding the complete match then reduces to the optimization problem

$$\arg\min_{\mathbf{p}} \sum_{i=1}^{N} d(\mathbf{x}_i, \mathbf{y}_{\mathbf{p}(i)}). \tag{1}$$

That is, the goal is to find a permutation such that the total distance between the two views of all matched objects is minimized. The problem (1) is the so-called *assignment problem*, for which a global optimum can be found with the well-known Hungarian algorithm (Kuhn 1955; Burkard et al. 2009). Note that the distance measure $d(\cdot, \cdot)$ affects the result, and care should be taken in choosing it. We will not discuss the choice further for the special case of replicate measurements, since our main goal is in extending the solution to non-replicate views.

When the views are not replicate measurements, the above approach does not work. This is because there is no easy way to define the distance measure. Even if both views had vectorial representations of equal length, the features themselves may not be directly comparable, making standard measures like Euclidean distance inapplicable. Below, we will give a justified choice of metric for such cases.

A new principle is needed, and we build that on *statistical dependency*. If the permutation $\mathbf{p}$ is random the views will necessarily be statistically independent, that is, $p(\mathbf{X}, \mathbf{Y}(\mathbf{p})) = p(\mathbf{X})p(\mathbf{Y}(\mathbf{p}))$. Hence, if the views are not independent, some systematic approach has been exercised in choosing the matching and, vice versa, systematic structure in the views can be found by maximizing the dependency with respect to $\mathbf{p}$. Assuming that the systematic structure is maximized if the permutation pairs the two views of the same object, which makes sense at least if the views are informative enough, then maximizing dependency should be a good solution to the matching problem.

We conjecture that maximizing the dependency, measured as the mutual information

$$I(\mathbf{X}, \mathbf{Y}(\mathbf{p})) = \int p(\mathbf{x}, \mathbf{y}_{\mathbf{p}(i)}) \log \frac{p(\mathbf{x}, \mathbf{y}_{\mathbf{p}(i)})}{p(\mathbf{x})p(\mathbf{y}_{\mathbf{p}(i)})} d\mathbf{x} d\mathbf{y},$$

with respect to the permutation $\mathbf{p}$ finds a good matching. In practice the mutual information cannot be directly used as the cost function, but it can be approximated with various levels of detail.

### 2.1 Matching in a subspace

Information processing can only lose information, and hence $I(\mathbf{X}, \mathbf{Y}(\mathbf{p})) \geq I(\mathbf{f}(\mathbf{X}), \mathbf{g}(\mathbf{Y}(\mathbf{p}))$ for any functions $\mathbf{f}$ and $\mathbf{g}$. This can be utilized in maximizing the mutual information, since it is practically not feasible to directly estimate mutual information in high-dimensional spaces. For dependency search any transformation hence gives us a lower bound, which we may be able to maximize in practice instead of the complete mutual information,

$$\max_{\mathbf{p},\mathbf{f},\mathbf{g}} \; I\left(\mathbf{f}(\mathbf{X}), \mathbf{g}(\mathbf{Y}(\mathbf{p}))\right) \leq \max_{\mathbf{p}} I\left(\mathbf{X}, \mathbf{Y}(\mathbf{p})\right). \tag{2}$$

In practice estimation of mutual information from finite data faces two serious problems: over-fitting and computational complexity. Both can be reduced by using simple transformations that reduce dimensionality, and simple estimates of mutual information. In practice we will use linear projections. To estimate mutual information in the lower-dimensional space, we will use (canonical) correlations, which will not detect dependencies in higher-order moments but are faster to compute than for instance the non-parametric estimates used in (Klami and Kaski 2005). For normally distributed data, there is a monotonous relationship between correlations and mutual information (Gretton et al. 2003), and hence finding projections that maximize correlations will maximize mutual information as well; for other distributions the relationship is only approximative and chosen because of computational reasons.

### 2.2 Technical details

For linear projections let $\mathbf{f}(\mathbf{x}) = \mathbf{x}\mathbf{w}_x^T$ where $\mathbf{x}, \mathbf{w}_x \in \mathbb{R}^{1\times D_x}$, and $\mathbf{g}(\mathbf{y}) = \mathbf{y}\mathbf{w}_y^T$ where $\mathbf{y}, \mathbf{w}_y \in \mathbb{R}^{1\times D_y}$. When using correlation as the dependency measure, the optimization problem becomes

$$\max_{\mathbf{p},\mathbf{w}_x,\mathbf{w}_y} \operatorname{corr}\left(\mathbf{X}\mathbf{w}_x^T, \mathbf{Y}(\mathbf{p})\mathbf{w}_y^T\right). \tag{3}$$

This has a direct relationship to the assignment problem (1) which can be seen as follows.

Given fixed projections, we can write the cost with the sample estimate of correlation as

$$\max_{\mathbf{p}} \frac{\mathbf{w}_x\mathbf{X}^T\mathbf{Y}(\mathbf{p})\mathbf{w}_y^T}{\|\mathbf{X}\mathbf{w}_x^T\|\,\|\mathbf{Y}(\mathbf{p})\mathbf{w}_y^T\|}. \tag{4}$$

The numerator can be expressed as

$$\mathbf{w}_x\mathbf{X}^T\mathbf{Y}(\mathbf{p})\mathbf{w}_y^T = \frac{1}{2}\left(\|\mathbf{X}\mathbf{w}_x^T\|^2 + \|\mathbf{Y}(\mathbf{p})\mathbf{w}_y^T\|^2 - \|\mathbf{X}\mathbf{w}_x^T - \mathbf{Y}(\mathbf{p})\mathbf{w}_y^T\|^2\right). \tag{5}$$

Assuming we are looking for a full one-to-one match, that is, $N = M$ and each object in $\mathbf{Y}$ is being matched to one object in $\mathbf{X}$, the first two terms in (5) as well as the denominator in (4) are constants with respect to $\mathbf{p}$; the order of objects does not affect the norm. Ignoring the constant terms gives

$$\min_{\mathbf{p}} \|\mathbf{X}\mathbf{w}_x^T - \mathbf{Y}(\mathbf{p})\mathbf{w}_y^T\|^2 = \min_{\mathbf{p}} \sum_{i=1}^{N} \|\mathbf{x}_i\mathbf{w}_x^T - \mathbf{y}_{\mathbf{p}(i)}\mathbf{w}_y^T\|^2 \tag{6}$$

as the optimization problem for $\mathbf{p}$. The important difference compared to (4) is that the cost has now been factorized over the objects, which allows treating the objects as independent samples in the minimization task. Given the constraint of one-to-one mapping, we have in fact arrived at an assignment problem (1) where the distance $d$ is the Euclidean (squared) distance between the projected values.

Now we propose an iterative algorithm to solve the optimization problem, by alternating between two steps: optimizing the matching $\mathbf{p}$ and learning the projections $\mathbf{w}_x$ and $\mathbf{w}_y$. Above, the former turned out to be the classical *assignment problem* where the cost of assignments was defined by the distance in the projection space, and which can be solved exactly with for instance the Hungarian algorithm.

For the second step of the alternating algorithm, the projections are computed assuming a fixed matching of objects. The cost function is still (3), but the optimization is with respect to the projection vectors $\mathbf{w}_x$ and $\mathbf{w}_y$. This is the familiar canonical correlation analysis (CCA) problem, which can be solved exactly with linear algebra (see Hardoon et al. 2004). CCA is a classical method for finding linear relationship between two multidimensional variables. It finds a set of basis vectors for two multidimensional variables such that the projections of variables onto these basis vectors are maximally correlated. The correlation between the projections is called *canonical correlation*. Each CCA component is associated with the corresponding canonical correlation $\rho_i$ that characterizes the strength of dependency.

In this case, given the matched data matrices $\mathbf{X}$ and $\mathbf{Y}(\mathbf{p})$, CCA will find the basis vectors $\mathbf{w}_x$ and $\mathbf{w}_y$ such that $corr\left(\mathbf{X}\mathbf{w}_x^T, \mathbf{Y}(p)\mathbf{w}_y^T\right)$ is maximal. The CCA solution is not, however, restricted to one-dimensional projections, but instead we can search for projection matrices $\mathbf{W}_x \in \mathbb{R}^{D\times D_x}$ and $\mathbf{W}_y \in \mathbb{R}^{D\times D_y}$ such that all the components are uncorrelated: $corr\left(\mathbf{X}\mathbf{W}_x^{(i)T}, \mathbf{X}\mathbf{W}_x^{(j)T}\right) = 0 \;\; \forall j \neq i$, using the notation $\mathbf{W}_x^{(i)}$ to denote the $i$th row of the matrix $\mathbf{W}_x$. Here, $D = min(D_x, D_y)$ is the maximal number of CCA components, and the additional components extracted by CCA can naturally be used also when solving the matching as well, by extending the distance measure in (6) to use multi-dimensional projections.

As correlation is scale-invariant, the different CCA projections can be re-scaled for maximal informativeness. For normal distributions $I(\mathbf{X}, \mathbf{Y}) = -1/2\sum_i \log\left(1 - \rho_i^2\right)$, showing that mutual information decomposes additively over the components, with the canonical correlation $\rho_i$ signifying the contribution of that particular component. Since we have no guarantee of normality we will not fixate on this specific functional form, but we use the same idea of using canonical correlations for weighting the contributions. In particular, we re-scale each dimension of $\mathbf{X}\mathbf{W}_x^T$ and $\mathbf{Y}\mathbf{W}_y^T$ with the corresponding canonical correlation, giving

$$\min_{\mathbf{p}} \sum_{i=1}^{N} \sum_{j=1}^{D} \rho_j^2 \left\| \mathbf{x}_i \mathbf{W}_x^{(j)T} - \mathbf{y}_{\mathbf{p}(i)} \mathbf{W}_y^{(j)T} \right\|^2 \tag{7}$$

as the final cost function to be used when learning the match.

The two steps explained above are combined together as a single alternating optimization algorithm: first a matching of objects is given as initialization and CCA is used to find optimal projections based on the matching. A new matching is then solved via the assignment problem using distances computed in the feature space. These two steps are repeated until convergence, detected as a step that leaves both the permutation and the projections intact. The matching is initialized randomly; given possibly completely different feature representations for the views it would be hard to use joint modeling to improve on random initializations. By starting from several random initializations it is possible to avoid local optima; the algorithm can be guaranteed to converge to a local but not a global optimum.

It is worth noticing that possible prior information on the match, typically obtained from yet another data source, can be taken into account as additional (hard or soft) constraints for the permutation matrix. The simplest approach is to use hard constraints that exclude certain matches from the set of possible solutions. We formulate this through the concept of *candidate sets*. For each observation $\mathbf{x}$ we define a subset of observations in $\mathbf{Y}$ as candidates. Given reliable information on the candidates, the constraints can trivially be added by modifying the cost matrix used for the assignment problem. Instead of filling the matrix with the computed distances, the excluded candidates are given an infinite distance, making it impossible for the assignment problem solver to match them. This not only improves the accuracy of the match, but also decreases the computational cost dramatically for large sample sizes when using relatively small candidate sets. Then the assignment cost matrix will be sparse, enabling efficient algorithms such as those of Jonker and Volgenant (1987); Duff and Koster (2001).

As a technical detail, the optimization problems (4) and (6) are equivalent only for $M = N$. For $M > N$ the terms containing $\left\|\mathbf{Y}(\mathbf{p})\mathbf{w}_y^T\right\|$ have to be included, since the set of objects in $\mathbf{Y}$ may change when changing the permutation. This means that the assignment problem cost matrix cannot be filled with the object-wise distances. In practice, however, the factorized cost gives a good approximation also for the cases where $M$ is slightly larger than $N$ (Tripathi et al. 2009). To counter the potential bias of favoring objects with small norm (*a priori* the Euclidean distance between $\mathbf{x}$ and $\mathbf{y}$ is positively correlated with $\|\mathbf{x}\|$ and $\|\mathbf{y}\|$), we suggest dividing the distances in (7) by $\sum_{j=1}^{D} \rho_j \left\|\mathbf{x}_i \mathbf{W}_x^{(j)T}\right\| \left\|\mathbf{y}_{\mathbf{p}(i)} \mathbf{W}_y^{(j)T}\right\|$ when $M > N$. This particular form of normalization is motivated by the form of (4). Even though we perform all the experiments in this article with $M = N$, we empirically study in Sect. 5.3 the effect of the normalization. The experiments suggest that the normalization should be used even when $M = N$.

```
Input: Matrices $\mathbf{X} \in \mathbb{R}^{N \times D_x}$ and $\mathbf{Y} \in \mathbb{R}^{M \times D_y}$. Candidate sets $S_i$ for each row $\mathbf{x}_i$ of $\mathbf{X}$, consisting
       of sets of indices for the objects in $\mathbf{Y}$. Each element in $S_i$ is an index from 1 to $M$.
Output: A match between the objects in $\mathbf{X}$ and $\mathbf{Y}$, given as a vector $\mathbf{p} \in [1..M]^N$. All the elements
        in $\mathbf{p}$ must be unique and $\mathbf{p}_i \in S_i \; \forall i$.
Initialization: Choose random $\mathbf{p}$ that satisfies the candidate set constraints.
repeat
    Find the projection matrices $\mathbf{W}_x$ and $\mathbf{W}_y$ and the canonical correlations $\{\rho_j\}_{j=1}^{D}$, where
    $D = \min(D_x, D_y)$, by maximizing the correlation between $\mathbf{X}$ and $\mathbf{Y}(\mathbf{p})$.
    Compute pair-wise distances $d(i,k)$ between objects in $\mathbf{X}$ and objects in $\mathbf{Y}$ where
    $k \in S_i, d(i,k) = \left\{\sum_{j=1}^{D} \rho_j^2 \left\|\mathbf{x}_i \mathbf{W}_x^{(j)T} - \mathbf{y}_k \mathbf{W}_y^{(j)T}\right\|^2\right\} / \left\{\sum_{j=1}^{D} \rho_j \left\|\mathbf{x}_i \mathbf{W}_x^{(j)T}\right\| \left\|\mathbf{y}_k \mathbf{W}_y^{(j)T}\right\|\right\}$.
    Set $d(i,k) = \infty$ for all pairs $(i,k)$ for which $k \notin S_i$.
    Find the match in the subspace defined by $\mathbf{W}_x$ and $\mathbf{W}_y$ by optimizing $\min_{\mathbf{p}} \sum_{i=1}^{N} d(i, \mathbf{p}(i))$,
    taking into account the constraint of unique values for the elements of $\mathbf{p}$.
until $\mathbf{p}$, $\mathbf{W}_x$, and $\mathbf{W}_y$ do not change;
```

**Algorithm 1**: Summary of the matching algorithm.

The final algorithm with all the technical details is summarized in Algorithm 1. The input for the algorithm consists of the data matrices and the candidate sets chosen based on application domain knowledge, and the output consists of the learned match $\mathbf{p}$ together with the CCA projections $\mathbf{W}_x$ and $\mathbf{W}_y$. The computational complexity at each iteration of the matching algorithm can be described as follows: (1) Perform canonical correlation analysis which is a generalized eigenvector problem. (2) Compute the distance matrix in the latent space which is of the order $O(N|c|)$, where $|c|$ is the size of the candidate sets. (3) Solve Assignment Problem which takes $O(NM^2)$. The Assignment Problem can be solved faster if the sparsity of the distance matrix due to candidate sets is utilized.

## 3 Related work

A number of other works have addressed the problem of making two data matrices commensurable when no co-occurrences are known. These works can be divided into two categories: methods that seek a match between the objects of the two views, like our method, and methods that aim at computing distances between the objects represented by different views. Even though the latter works typically only provide the distance measure, they can be converted to achieve also the former by solving a bi-partite matching problem for costs stemming from the distances. That is, our viewpoint makes it obvious on how the actual matching problem can be solved given a distance measure.

Haghighi et al. (2008) introduced an algorithm very closely resembling ours, developed independently of our initial works (Tripathi et al. 2008, 2009). The main difference is that Haghighi et al. (2008) use a slightly different, heuristic cost for the assignment problem, whereas we derived the cost by connecting the distances to the original cost function. Another difference is that Haghighi et al. (2008) started from a probabilistic formulation of canonical correlation analysis, searching for the maximum likelihood estimate of the probabilistic CCA. This difference is, however, superficial, since the maximum likelihood solution has been shown to be equivalent to the solution of classical CCA (Bach and Jordan 2005), and hence in this step the algorithms are identical.

Quadrianto et al. (2009) provide an alternative approach motivated along the same lines: objects should be matched to maximize statistical dependency between the views. Instead of correlation they use Hilbert-Schmidt independence criterion (HSIC; Smola et al. 2007) for measuring the dependency. HSIC can be computed directly for the original representations, and hence their algorithm does not provide explicit low-dimensional representations. The algorithm is still iterative in the same way as ours, since the HSIC results in a quadratic assignment problem instead of a linear one and they solve it by iteratively applying linear assignment problem solvers.

Wang and Mahadevan (2009) propose a method for aligning different manifolds, by constructing a distance measure between objects of the two different views. The measure is based on aligning local neighborhoods within each of the views, and in a sense converts within-view distances into between-view distances. In brief, a distance from $\mathbf{x}_i$ to $\mathbf{y}_j$ is based on alignment of the neighborhoods of the $k$ nearest neighbors in both spaces. The objects are deemed to be close to each other if the set of distances from $\mathbf{x}_i$ to its closest neighbors is close to the set of the distances from $\mathbf{y}_i$ to its neighbors. The algorithm is not iterative in the same sense as the earlier approaches, but on the other hand is computationally very complex: Determining the distance between two neighborhoods requires going through all $k!$ permutations of the neighboring objects. The approach does not directly return one-to-one mappings between the objects but merely allows measuring distances between the objects in the different views.

Some methods have also been presented for the related problem of matching the objects given a small training data for which the matching is already known. This problem is considerably simpler since the seed matching provides information on which features are informative of the match. Wang and Mahadevan (2008) apply “procrustes analysis” for creating a distance measure for the rest of the objects, whereas Tripathi et al. (2010) study the semi-supervised problem for solving the actual match. As expected, providing some supervision clearly increases the accuracy.

In Sect. 5.2 we empirically compare our approach with the above alternatives, excluding the almost identical method of Haghighi et al. (2008) and the semi-supervised variants, in the application of metabolite matching. To ensure fair comparison we adopt parts of our algorithm for the others as well, including the candidate sets and coupling the manifold alignment of Wang and Mahadevan (2009) with the assignment problem.

## 4 Consensus match

As far as we know, all solutions proposed to the matching problem, including our solution to the generalized problem for different data domains, find the match for a single realization of two data matrices. In a realistic matching application the task is, however, to find an underlying match between several realizations of the object collections. For example, when matching metabolic identities between two species we often have measurements for several individuals of both species. The main task is to find the match between the species, not between the individuals. By assuming the metabolic activity of a single individual to be a noisy example of generic species-specific activity, the match between the species is obtained by averaging over the matches of different individuals.

Let $\bar{\mathbf{X}} = \{\mathbf{X}_s\}, s \in \{1 : S\}$ and $\bar{\mathbf{Y}} = \{\mathbf{Y}_t\}, t \in \{1 : T\}$ be the $T$ and $S$ realizations of the two views, where $\mathbf{X}_s \in \mathbb{R}^{N \times D_{xs}}$ and $\mathbf{Y}_t \in \mathbb{R}^{M \times D_{yt}}$. Again $M \geq N$. The task is to learn a single permutation $\mathbf{p}$ to match the objects. Algorithm 1 provides a solution between any two realizations $\mathbf{X}_s$ and $\mathbf{Y}_t$, and now we would like to utilize this existing pairwise algorithm to solve the global matching problem. In brief, the basic idea is to find the matches between sufficiently many realizations, and then to find a consensus of all these matches.

We will make two simplifying assumptions. First, the included pairs of realizations are assumed to be independent samples, which holds approximatively assuming the total number of realizations is large. Second, when combining the matchings we will neglect some of the constraints as detailed below; this is necessary to keep the computations manageable. The quality of the solution needs to be evaluated empirically, which we will do in Sect. 5.

Let $\mathbf{p}_k, k \in \{1 : S \times T\}$ be a match between the objects of any two $\mathbf{X}_s$ and $\mathbf{Y}_t$, obtained by solving (3). We combine the individual solutions $\mathbf{p}_k$ by creating a contingency table $\mathbf{C} \in \mathbb{N}^{N \times M}$, where cell $\mathbf{C}(i, j)$ is the count of solutions where the $i$th observation of $\mathbf{X}_s$ has been paired with the $j$th observation of $\mathbf{Y}_t$. Intuitively, if two objects are paired with each other in many individual matching solutions the corresponding cell value will have a high count.

Now we make the simplifying assumption of using only the information provided in the contingency table. Then the problem reduces to solving a maximum weight bipartite matching between the rows and columns of the contingency table $\mathbf{C}$, where the cost (or weight) of matching comes from the cell frequency $\mathbf{C}(i, j), i \in \{1 : N\}, j \in \{1 : M\}$, and $\sum_{j=1}^{M} \mathbf{C}(i, j) = S \times T, \forall i$. Let $\mathbf{P}$ be the consensus matching based on the contingency table, obtained as solution to the optimization problem

$$\max_{\mathbf{P}} \sum_{i=1}^{N} \mathbf{C}(i, \mathbf{P}(i)). \tag{8}$$

Coincidentally, this problem can be solved using the Hungarian algorithm used above for solving the individual matchings as well. The only difference is that instead of minimizing the total distance we now maximize the counts.

After solving the consensus match, the individual pairs can be ordered according to decreasing count. That is, the rows and columns of $\mathbf{C}(i, \mathbf{P}(i))$ are re-ordered so that the found pairs are on the diagonal in decreasing order. This corresponds to a crude measure or reliability of any given pair; those occurring in the beginning of the list are more likely to be correct than those at the end.

Finally, we propose a simple way to characterize potential alternative pairs for each object. This may be useful for applications where one-to-many correspondences are also possible, or simply to provide more information on the match. This is done by comparing the counts of all possible pairs to the simple null distribution of all matches being *a priori* equally likely. We use $\mathbf{C}(r, i)$ as a test statistic for all $i \in [1, M]$, and estimate the p-values for each pair of objects as the proportion where $\mathbf{Z}(r, i) > \mathbf{C}(r, i)$. The null distribution $\mathbf{Z}(r, i)$ is generated by drawing 1000 random matches that satisfy the candidate sets, and counting in how many of those each of the potential pairs occurs. That is, the distribution is constructed in the same way as the matching algorithm is being initialized. The pairs with low p-values are given as a list of potential alternative matches for any particular object in $\mathbf{X}$, complementing the original consensus match for real biological use cases.

## 5 Application: translational metabolomics

In metabolomics studies concentrations of metabolic products in tissue samples are measured, typically with mass spectrometry combined with a chromatography method such as Liquid Chromatography. The result is a set of peaks in the mass spectrum, in the two-dimensional plane of retention time vs. mass-to-charge ratio. Some of these peaks can be identified to stem from specific metabolites but many cannot, and uncertainties remain in the metabolic profiles even after sophisticated preprocessing methods.

A typical goal in the studies is to find differences in the metabolic profiles of two populations, for instance diabetic and healthy (Orešič et al. 2008b) or males and females (Nikkilä et al. 2008). Also similarities in populations are interesting, in particular when comparing metabolic profiles of different organisms, such as mice and men, in order to find out which properties of the model organisms generalize to humans and which do not. The goal of translating findings from one study or population to another can be called translational metabolomics.

A main problem in comparing two populations is that, due to the measurement and preprocessing process, the matching between the metabolites between the populations is not completely known. This is obviously problematic for the unidentified metabolites but also the roles of identified metabolites may be different in different organisms or tissues. A matching needs to be found, and it is an attractive idea to compute the matching in a data-driven way, in order to both circumvent imperfectness in preprocessing methods and to be able to find non-trivial correspondences where the functional role of the metabolites may be different.

Due to the nature of the measurement process, there is a translational problem even within the same tissue in the same organism. In large scale studies involving several hundreds of samples, the metabolic profile measurements are commonly made in multiple analytical batches. Within a single batch the metabolic profiles are comparable, but across the batches there is a possibility of larger variation due to instrumental drift. For us this is an excellent opportunity to test the method, because for the identified metabolites within a single tissue and single organism, the ground truth of the matching is known. We will compute a data-driven matching and compare it to the ground truth, in order to estimate how reliable the method is, and then in future studies apply it to translation.

## 5.1 Data

The data consist of measurements of lipids from a recent large birth cohort study of Type 1 diabetes (DIPP; Orešič et al. 2008b). In this study, lipidomic profiles of healthy human patients and patients developing into type 1 diabetes were measured at variable intervals. The data corresponding to each individual is a time series of metabolic expression for a given set of metabolites. The length of time series for each individual is different, ranging from 2 to 30. Also the sampling points are different, implying that each individual forms a different data domain. We use below altogether 126 individuals, with time series of more than two points, and 53 metabolites which have been measured and identified for all individuals, so the true matching is known.

We randomly partition the set of individuals into two sets, $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}}$, each consisting of 63 individuals. Each set ($\bar{\mathbf{X}}$ or $\bar{\mathbf{Y}}$) represents a view. An individual in any view is called a realization, which is a data matrix with metabolites as observations and time points as variables. The task is to find a matching of metabolites between the two views, $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}}$. Any two realizations, one from each view, can be used to match the metabolites. There are altogether $63 \times 63 = 3969$ such combinations of realizations, and hence the final match can be computed as a consensus of 3969 matching solutions.

The lipids are traditionally categorized into known classes (Orešič et al. 2008a) which we can take as prior information. In practice, matches should occur only within a class, and the idea is to search within a functional class to find a pair for any given metabolite. This also speeds up the process by restricting the search space.

Further, it is in general a good idea to restrict the search to only a subset of the possible pairs if possible. We call such subsets *candidate sets*, and for each metabolite in any of the $\mathbf{X}_s$, a candidate set is a subset of the corresponding metabolites in $\mathbf{Y}_t$. In the metabolomics application we use mass-to-charge ratio (MZ) and retention time (RT) of metabolites to create the candidate sets. For a given metabolite, we pick a fixed number of closest metabolites based on MZ and RT values weighted by corresponding thresholds (0.001 for MZ and 10 seconds for RT, coming from field experts). In our experiments, the size of the candidate sets was 10 for those functional classes having more than 10 metabolites. For smaller functional classes the whole class formed the candidate set.

## 5.2 Experiments

In this section we will study empirically the accuracy of the matching results for a data set where the ground truth is known. The experiments are designed to answer the following questions: (i) How exactly should the match be computed? (ii) How does finding the consensus of several matches help and to what extent? (iii) How much data do we need to get a good matching solution? We also compare the performance of our matching algorithm with manifold alignment and kernelized sorting.

First, we study the performance of our algorithm on a single pair of realizations and study the effect of normalization on the distances (7) between the objects when learning the match. The purpose of this experiment is to both illustrate the accuracy of a match learned from a single realization, and also to show how normalizing the distances in the projection space to emphasize data points with high variation improves the accuracy.

We then compare our CCA-based matching algorithm to manifold alignment (Wang and Mahadevan 2009) and kernelized sorting (Quadrianto et al. 2009) by matching the objects between a single pair of realizations. In order to make the comparison fair, the concept of *candidate sets* is incorporated also for the manifold alignment and kernelized sorting; the information about functional classes of lipids is used to restrict the search space for matching as in our algorithm. Manifold alignment is used for matching in two settings. (i) We choose the true match for each object to be the closest object in the other data set as suggested in the original publication. (ii) The distance provided by the manifold alignment method is used as a cost for the assignment problem, and a one-to-one match is learned with the Hungarian algorithm as in our method. In case of Kernelized sorting, we used both of the initialization alternatives suggested by Quadrianto et al. (2009), namely initialization with kernel-PCA and random initialization. For both methods we use Gaussian kernels and choose the kernel parameters as in (Quadrianto et al. 2009) for both, since Wang and Mahadevan (2009) do not give suggestions on how the choice should be made. The neighborhood parameter $k$ of the manifold alignment was chosen to be 4, following the suggestion in the original publication.

Next, we move to studying the consensus match. The consensus is computed using our CCA-based matching algorithm. We show how the accuracy is considerably improved already when learning the match from two realizations of both views instead of one, and continue with experiments on increasing number of realizations. At the same time, we study the effect of initialization of the individual matches, and conclude by studying the choice of distance normalization in a larger experiment. We also study how many realizations (that is, individual measurements) are needed for learning a good match between the metabolites, using the optimal settings learned in the first experiment. In a real biological experiment the measurement resources are always constrained, and knowing in advance how much data is needed for reliable results would be beneficial. We seek to answer this question for the data studied in this paper, expecting the result to roughly generalize to experiments of similar nature.

## 5.3 Results

We start by assessing the quality of the match learned from a single realization of views. As we have access to a number of realizations, we measure the performance over 1,000 random choices of the humans used for matching the metabolites. We learn the match using both the distance measure of (7) and a normalized variant shown in Algorithm 1, in order to compare these two approaches. Figure 2 shows the distribution of the proportion of correct matches, with average percentage at 35.7% for the normalized variant and 26.3% for the un-normalized variant. The normalized variant is hence considerably more accurate, and is to be preferred at least for individual matches.

Both variants are more accurate than a baseline provided by the hard constraints of the candidate sets (average accuracy of random matches satisfying the constraints

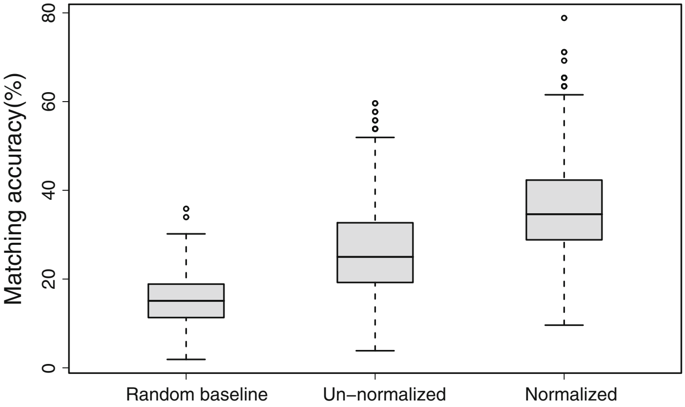


**Fig. 2** Matching accuracy in the task of matching metabolites given a single realization of each view. The matching accuracy is computed against the known ground truth. The boxplots represent the matching accuracies of 1,000 randomly selected realization pairs. The matching given by the proposed algorithm is more accurate than the baseline given by the random matching satisfying the candidate sets, showing it is possible to infer the match from the data. Furthermore, using the normalized distance for optimizing the match is superior to the un-normalized variant by a margin of almost 10%

is 15.2%), indicating that the algorithm has found useful information already from a single realization. For a practical biological application the accuracy of the method would not be acceptable, however. The poor accuracy is primarily due to the difficulty of the task itself—some humans have only three-point time series, and the two views may have very different time-scales and dimensionalities. It would be unfeasible to expect high accuracy for such a matching problem.

Figure 3 shows the comparison of our algorithm with manifold alignment and kernelized sorting. The average matching accuracy of our CCA-based matching algorithm (35.7%) is clearly better than the average matching accuracy of manifold alignment (23.4%), and both the variants of kernelized sorting; KPCA-based initialization gives 27% and random initialization reaches 30.7%. All the differences are statistically significant ($p$-value below $1e - 14$) based on *Welch Two Sample t*-test. When coupled with the assignment problem, the average accuracy of manifold alignment (36.08%) is comparable to our CCA-based matching approach. In this case, the difference is not statistically significant ($p$-value 0.37). The manifold alignment method is, however, orders of magnitude slower than both of the alternative methods. In summary, we observe that the proposed CCA-based matching method is, for this application, more accurate than kernelized sorting, and comparable but much faster than the manifold alignment solution complemented with an assignment problem solver. This warrants using the proposed algorithm for learning the consensus match, but it is worth keeping in mind that the consensus could also be learned from the solutions of the other alignment methods—or from a combination of solutions learned with different algorithms.

Next we illustrate how the accuracy is improved when more realizations are used for learning the match, using the consensus solution of Sect. 4. As shown in Fig. 4, already the consensus learned from two realizations (that is, a total of 4 individual matches) gives a notable increase in accuracy. The real power of the consensus, however, is in

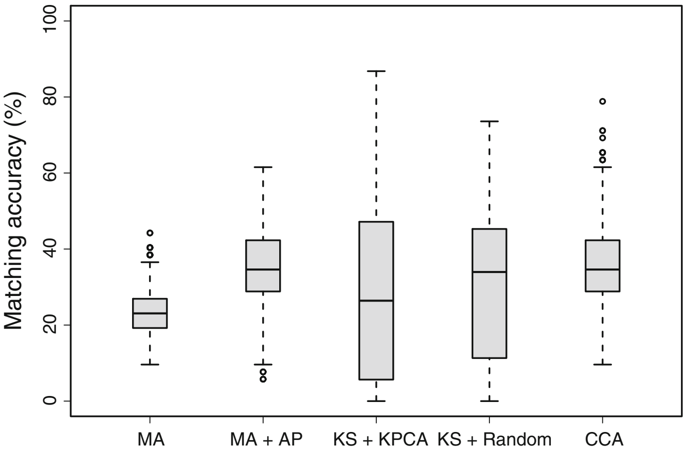


**Fig. 3** Comparison of Manifold alignment, Kernelized sorting and our CCA-based matching. MA represents Manifold alignment as proposed in the original publication, MA + AP represents MA with Assignment problem. KS + KPCA represents Kernelized sorting with KPCA-based initialization, KS + Random represents Kernelized sorting with random initialization. CCA represents our CCA-based matching algorithm. In all the methods, the same candidate sets and information about metabolite functional classes is used. Matching accuracy is based on 1,000 randomly selected single realizations

combining a large number of individual matches, learned from tens of realizations. We illustrate this by adding more and more realizations. We start again from the case of $1 + 1$ humans that gives one matching solution, and progressively increase the number of humans on both sides, always computing the consensus match over all possible matching solutions that can be computed based on the given data. For each data collection size we average the results over 100 random choices of individuals for both collections. Note that we use separate humans for the $\mathbf{X}$ and $\mathbf{Y}$, since this study is a proxy for translation studies where the populations would indeed be different.

We also study the effect of initialization of the individual matches on the consensus matching. The individual matching solutions solved by maximizing (3) are not global optima, but only local ones affected by the initial matching. Intuitively, it makes sense to start each individual matching from a different random initialization to maximize the independence of the solutions. In Fig. 5 we show the accuracy of the consensus matches for two different initialization strategies, one using different random initialization for each individual matching solution and one using the same initial matching for all. The result confirms the intuition, showing that maximizing the independence between the individual matches through different initializations is crucial for good matching accuracy. While random initialization might not be optimal for a single match, it guarantees sufficient diversity for the consensus.

In the above experiment we used the normalized distances, as suggested by the results of the experiment with one realization. To verify whether it is the better choice also for consensus matches, we run a similar experiment over the whole range of possible data collections. Figure 6 shows that the normalized variant is more accurate for all data collection sizes, confirming the initial observation. In conclusion, we have shown that the choices made in Algorithm 1 are superior to the possible alternatives.

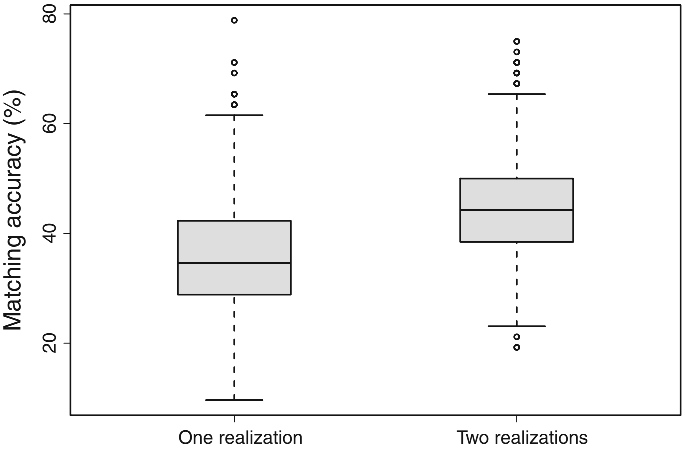


**Fig. 4** Illustration of the gain in performance from using more realizations for matching the objects between two views. The two boxplots correspond to: (i) the matching of metabolites based on single realization for each view, and (ii) the matching accuracy based on the consensus of matchings using two realizations for each view. The gain from adding the second realization is clear, on average around 9%. The normalized-distance variant is used in both the cases

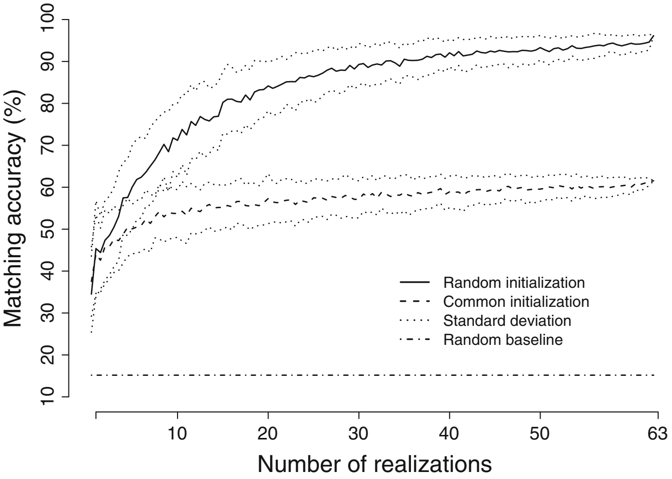


**Fig. 5** Accuracy of the consensus matching as a number of realizations, combining all possible individual matches between the humans in both collections. The matching accuracies of two alternative initialization strategies are averaged over 100 random selections of individuals to the two the data collections, and the dotted curves show the standard deviation over the randomization. The variant having different random initializations (*solid line*) for each individual match is shown to be clearly more accurate than the consensus computed from individual matches with identical initializations (*dashed line*). Both variants clearly surpass the baseline accuracy given by random matches (*dash-dotted line*)

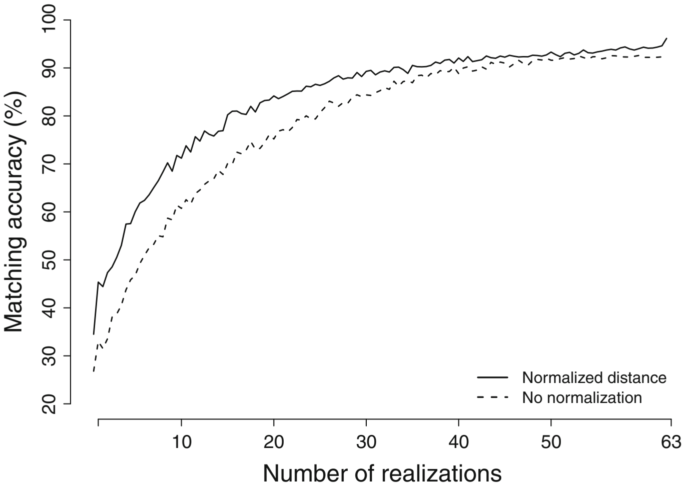


**Fig. 6** Comparison of the normalized and un-normalized distance variants in a consensus match. The normalized variant is slightly more accurate for all data collection sizes, and is hence to be preferred. The final accuracy of the better variant reaches 96%, and already with 30 realizations the accuracy is at 90%, which could be considered as a sufficient accuracy for a practical translational metabolomics application

In particular, the distances in (7) should be normalized by the norms of the projections, and giving a different random initialization for each individual match is a good strategy.

Finally, we turn our attention to the actual matching results. Both Figs. 5 and 6 show that the accuracy grows rather monotonously as a function of available data, eventually reaching 96.15% accuracy for the case with 63 realizations (3,969 individual matches). This means that only 2 metabolites out of 52 are matched incorrectly, which is the second best achievable result—a single mistake in one-to-one matching naturally implies that another metabolite must also have been matched incorrectly. Furthermore, the statistical test for finding alternative pairs indicates that for both of the erroneously paired metabolites the true pair would be the most likely alternative pair. In summary, it is possible to learn the match between the metabolites of the two data collections in a data-driven way, given enough data.

The remaining question is, how much data is actually needed. This can be studied by looking at the curve in Fig. 6. We know from Fig. 2 that the accuracy with just one realization is not sufficient. However, the accuracy rises steeply when more data is added. Already at 10 realizations (consensus of 100 individual matches) the accuracy is at 70%. This represents the kind of accuracy obtainable with small-scale metabolomics data sets. Going further, we notice that 30 realizations (900 individual matches) are enough for roughly 90% accuracy, which would typically already be sufficient in a biological experiment. While the results do not directly generalize to other metabolomics measurements, we can still conclude that measurement collections of feasible size (tens of realizations) are sufficient for learning sufficiently accurate matches, at least for collections where the number of time points is comparable to our data (on average around 10 time points).

## 6 Discussion

In this paper we introduced the problem of learning the match between sets of objects that have not been measured as co-occurring data. The goal is to make it possible to combine different views even when the co-occurrence of the objects is not known, to enable joint analysis of the views or application of various multi-view learning methods.

We presented an algorithm that matches the objects of the two views by repeatedly solving linear assignment problems. If the feature spaces of the two views can be directly compared, then the matching can be solved as a single assignment problem. When the views can not be directly compared, we need an iterative algorithm that alternatingly solves the match and finds a better representation for comparing the objects. The representations are learned to maximize the statistical dependency (mutual information) between the views, with the intuitive idea that a representation capturing the dependency between the views gives the most reliable measure of similarity.

We also showed how several independent matching results, obtained by applying the algorithm for several realizations of the views, can be combined to improve the accuracy of the match. The consensus is again learned by solving an assignment problem, which makes the whole approach computationally straightforward: The whole process only requires a standard assignment problem solver, linear algebra to solve the canonical correlation analysis problem to find the representations, and simple bookkeeping. For larger problems the computational efficiency of the assignment problem can be improved by using prior information to introduce sparsity in the cost matrix. The algorithm used for finding the consensus is general, and can be applied on top of any other algorithm for finding the individual matching solutions, such as those presented by Quadrianto et al. (2009) or Wang and Mahadevan (2009). In this work we showed that the accuracy of our individual alignments is better than one and comparable to the other, and computationally much faster than the comparable one.

We used the problem of translational metabolomics, that is, study of differences and commonalities between metabolic activity of different species, as a prototype application. Model organisms such as mice are being used for more extensive biological experiments, and it is crucial to be able to generalize the findings to humans. However, the metabolic profiles, measured by mass spectrometry, are not directly comparable between the species, and hence the match needs to be inferred from the data. In this article we applied the method to a partially artificial problem of matching the metabolites between two collections of the *same* species, in order to evaluate the performance of the method in a setting having a gold standard baseline. We showed that the accuracy learned from a single realization of noisy objects is not sufficient for further analysis, but that with large enough measurement collections near 100% accuracy can be achieved. For the particular metabolomic data in question around 30 examples of both species would be needed for adequate 90% matching accuracy, which is within the limits of current measurement practices. The algorithm is being applied at the moment in lipidomics studies, for matching lipids between humans and mice.

**Acknowledgements** AK and SK belong to Finnish Center of Excellence in Adaptive Informatics Research of the Academy of Finland. AK was supported by the Academy of Finland decision number 133818, and AT, AK and SK were partially supported by EU Network of Excellence PASCAL2, ICT 216886. This work was partially supported by TEKES (grant no. 40101/07).

## References

Bach FR, Jordan MI (2005) A probabilistic interpretation of canonical correlation analysis. Technical Report 688, Department of Statistics, University of California, Berkeley

Barzilay R, Elhadad N (2003) Sentence alignment for monolingual comparable corpora. In: Proceedings of the 2003 conference on empirical methods in natural language processing. Association for Computational Linguistics, Morristown, NJ, USA, pp 25–32

Bickel S, Scheffer T (2005) Estimation of mixture models using Co-EM. In: Proceedings of the European conference on machine learning, Lecture Notes in Computer Science, vol 3720/2005. Springer, Berlin, Heidelberg, pp 35–46. doi:10.1007/11564096

Blei DM, Jordan MI (2003) Modeling annotated data. In: Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press, New York, NY, USA, pp 127–134

Burkard R, Dell'Amico M, Martello S (2009) Assignment problems. Society for Industrial and Applied Mathematics, Philadelphia

Damian D, Orešič M, Verheij E, Meulman J, Friedman J, Adourian A, Morel N, Smilde A, van der Greef J (2007) Applications of a new subspace clustering algorithm (COSA) in medical systems biology. Metabolomics 3:69–77

Duff IS, Koster J (2001) On algorithms for permuting large entries to the diagonal of a sparse matrix. SIAM J Matrix Anal Appl 22(4):973–996. doi:10.1137/S0895479899358443

Farquhar JDR, Hardoon DR, Meng H, Shawe-Taylor J, Szedmak S (2006) Two view learning: SVM-2K, theory and practice. In: Weiss Y, Schölkopf B, Platt J (eds) Advances in neural information processing systems, vol 18. MIT Press, Cambridge, MA pp 355–362

Gretton A, Herbrich R, Smola A (2003) The kernel mutual information. In: Proceedings of ICASSP'03, IEEE international conference on acoustics, speech, and signal processing, IEEE, pp IV-880–IV-883

Haghighi A, Liang P, Berh-Kirkpatrick T, Klein D (2008) Learning bilingual lexicons from monolingual corpora. In: Proceedings of ACL-08: HLT. Association for Computational Linguistics, Columbus, Ohio, pp 771–779

Hardoon DR, Szedmak S, Shawe-Taylor J (2004) Canonical correlation analysis: an overview with application to learning methods. Neural Comput 16(12):2639–2664

Jonker R, Volgenant A (1987) A shortest augmenting path algorithm for dense and sparse linear assignment problems. Computing 38(4): 325–340. doi:10.1007/BF02278710

Klami A, Kaski S (2005) Non-parametric dependent components. In: Proceedings of ICASSP'05, IEEE international conference on acoustics, speech, and signal processing, IEEE, pp V-209–V-212

Klami A, Kaski S (2008) Probabilistic approach to detecting dependencies between data sets. Neurocomputing 72(1–3):39–46

Kuhn HW (1955) The Hungarian method for the assignment problem. Naval Res Logist Quart 2(1–2): 83–97

Li Y, Shawe-Taylor J (2006) Using KCCA for Japanese-English cross-language information retrieval and document classification. J Intel Inf Syst 27(2):117–133. doi:10.1007/s10844-006-1627-y

Melamed D (1999) Bitext maps and alignment via pattern recognition. Comput Linguist 25(1):107–130

Nikkilä J, Sysi-Aho M, Ermolov A, Seppänen-Laakso T, Simell O, Kaski S, Orešič M (2008) Gender dependent progression of systemic metabolic states in early childhood. Mole Syst Biol 4:197. doi:10.1038/msb.2008.34

Orešič M, Hänninen V, Vidal-Puig A (2008) Lipidomics: a new window to biomedical frontiers. Trends Biotechnol 26(12):647–652. doi:10.1016/j.tibtech.2008.09.001

Orešič M, Simell S, Sysi-Aho M, Nanto-Salonen K, Seppänen-Laakso T, Parikka V, Katajamaa M, Hekkala A, Mattila I, Keskinen P, Yetukuri L, Reinikainen A, Lähde J, Suortti T, Hakalax J, Simell T, Hyöty H, Veijola R, Ilonen J, Lahesmaa R, Knip M, Simell O (2008) Dysregulation of lipid and amino acid metabolism precedes islet autoimmunity in children who later progress to type 1 diabetes. J Exp Med 205(13):2975–2984

Quadrianto N, Song L, Smola A (2009) Kernelized sorting. In: Koller D, Schuurmans D, Bengio Y, Bottou L (eds) Advances in neural information processing systems, vol 21. MIT Press, Cambridge, MA, pp 1289–1296

Rogers S, Girolami M, Kolch W, Waters KM, Liu T, Thrall B, Wiley HS (2008) Investigating the correspondence between transcriptomic and proteomic expression profiles using coupled cluster models. Bioinformatics 24(24):2894–2900. doi:10.1093/bioinformatics/btn553

Rogers S, Klami A, Sinkkonen J, Girolami M, Kaski S (2010) Infinite factorization of multiple non-parametric views. Mach Learn 79(1-2):201–226. doi:10.1007/s10994-009-5155-1

Smola AJ, Gretton A, Song L, Schölkopf B (2007) A Hilbert space embedding for distributions. In: Takimoto E (ed) Algorithmic learning theory, Lecture Notes on Computer Science, invited paper. Springer, Berlin, Heidelberg, pp 13–31

Tripathi A, Klami A, Kaski S (2008) Using dependencies to pair samples for multi-view learning. TKK reports in information and computer science TKK-ICS-R8, Helsinki University of Technology, Espoo, Finland

Tripathi A, Klami A, Virpioja S (2010) Bilingual sentence matching using kernel CCA. In: Kaski S, Miller DJ, Oja E, Honkela A (eds) Proceedings of IEEE International Workshop on Machine Learning for Signal Processing (MLSP), IEEE, pp 130–135. doi:10.1109/MLSP.2010.5589249

Tripathi A, Klami A, Virpioja S (2010) Bilingual sentence matching using kernel CCA. In: Kaski S, Miller DJ, Oja E, Honkela A (eds) Proceedings of IEEE international workshop on machine learning for signal processing (MLSP), IEEE, pp 130–135. doi:10.1109/MLSP.2010.5589249

Vinokourov A, Hardoon DR, Shawe-taylor J (2003) Learning the semantics of multimedia content with application to web image retrieval and classification. In: In proceedings of fourth international symposium on independent component analysis and blind source separation

Wang C, Mahadevan S (2008) Manifold alignment using Procrustes analysis. In: Proceedings of the 25th international conference on machine learning, pp 1120–1127

Wang C, Mahadevan S (2009) Manifold alignment without correspondence. In: IJCAI'09: Proceedings of the 21st international joint conference on artifical intelligence. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp 1273–1278