Computational and Systems Biology
Cancer Biology

Optimal transport for automatic alignment of untargeted metabolomic data

Reviewed Preprint

Revised by authors after peer review.

About eLife's process

Reviewed preprint version 2
April 9, 2024 (this version)

Reviewed preprint version 1
December 11, 2023

Posted to preprint server
September 13, 2023

Sent for peer review
September 11, 2023

Marie Breeur, George Stepaniants, Pekka Keski-Rahkonen, Philippe Rigollet, Vivian Viallon

Nutrition and Metabolism Branch, International Agency for Research on Cancer, Lyon, France • Massachusetts Institute of Technology, Department of Mathematics, Cambridge MA, United States

Abstract

Untargeted metabolomic profiling through liquid chromatography-mass spectrometry (LC-MS) measures a vast array of metabolites within biospecimens, advancing drug development, disease diagnosis, and risk prediction. However, the low throughput of LC-MS poses a major challenge for biomarker discovery, annotation, and experimental comparison, necessitating the merging of multiple datasets. Current data pooling methods encounter practical limitations due to their vulnerability to data variations and hyperparameter dependence. Here we introduce GromovMatcher, a flexible and user-friendly algorithm that automatically combines LC-MS datasets using optimal transport. By capitalizing on feature intensity correlation structures, GromovMatcher delivers superior alignment accuracy and robustness compared to existing approaches. This algorithm scales to thousands of features requiring minimal hyperparameter tuning. Applying our method to experimental patient studies of liver and pancreatic cancer, we discover shared metabolic features related to patient alcohol intake, demonstrating how GromovMatcher facilitates the search for biomarkers associated with lifestyle risk factors linked to several cancer types.

eLife assessment

The authors describe an **important** tool, GromovMatcher, that can be used to compare proteomic data from various experimental approaches. The underlying method is innovative, the algorithm is clearly described, and the validation that is presented is **convincing**.

Introduction

Untargeted metabolomics is a powerful analytical technique used to identify and measure a large number of metabolites in a biological sample without preselecting targets Patti (2011). This approach allows for a comprehensive overview of an individual's metabolic profile, provides insights into the biochemical processes involved in cellular and organismal physiology Wishart (2019); Pirhaji et al. (2016), and allows for the exploration of how environmental factors

impact metabolism Rappaport et al. (2014); Bedia (2022). It creates new opportunities to investigate health-related conditions, including diabetes Wang et al. (2011), inflammatory bowel diseases Franzosa et al. (2019), and various cancer types Loftfield et al. (2021); Li et al. (2020). However, a major challenge in biomarker discovery, metabolic signature identification and other untargeted metabolomic analyses lies in the low throughput of experimental data, necessitating the development of efficient pooling algorithms capable of merging datasets from multiple sources Loftfield et al. (2021).

A common experimental technique in untargeted metabolomics is liquid chromatography-mass spectrometry (LC-MS) which assembles a list of thousands of unlabeled metabolic features characterized by their mass-to-charge ratio (*m/z*), retention time (RT) Zhou et al. (2012), and intensity across all biological samples. Combining LC-MS datasets from multiple experimental studies remains challenging due to variation in the *m/z* and RT of a feature from one study to another Zhou et al. (2012); Ivanisevic and Want (2019). This problem is further compounded by differing instruments and analytical protocols across laboratories, resulting in seemingly incompatible metabolomic datasets.

Manual matching of metabolic features can be a laborious and error-prone task Loftfield et al. (2021). To address this challenge, several automated methods have been developed for metabolic feature alignment. One such method is MetaXCMS, which matches LC-MS features based on user-defined *m/z* and RT thresholds Tautenhahn et al. (2011). More advanced tools use information on feature intensities measured in samples. For instance, PAIRUP-MS uses known shared metabolic features to impute the intensities of all features from one dataset to another Hsu et al. (2019). MetabCombiner Habra et al. (2021) and M2S Climaco Pinto et al. (2022) compare average feature intensities, along with their *m/z* and RT values, to align datasets without requiring extensive knowledge of shared features. These automated alignment methods have accelerated our ability to pool and annotate datasets as well as extract biologically meaningful biomarkers. However, they demand substantial fine-tuning of user-defined parameters and ignore correlations among metabolic features which provide a wealth of additional information on shared features.

Here we introduce GromovMatcher, a user-friendly flexible algorithm which automates the matching of metabolic features across experiments. The main technical innovation of GromovMatcher lies in its ability to incorporate the correlation information between metabolic feature intensities, building upon the powerful mathematical framework of computational optimal transport (OT) Peyré et al. (2019); Villani (2021). OT has proven effective in solving various matching problems and has found applications in multiomics analysis Demetci et al. (2022), cell development Schiebinger et al. (2019); Yang et al. (2020), and chromatogram alignment Skoraczyński et al. (2022). Here we leverage the Gromov-Wasserstein (GW) method Mémoli (2011); Solomon et al. (2016), which matches datasets based on their distance structure and has been seminally applied to spatial reconstruction problems in genomics Nitzan et al. (2019). GromovMatcher builds upon the GW algorithm to automatically uncover the shared correlation structure among metabolic feature intensities while also incorporating *m/z* and RT information in the final matching process.

To assess the performance of GromovMatcher, we systematically benchmark it on synthetic data with varying levels of noise, feature overlap, and data normalizations, outperforming prior state-of-the-art methods of metabCombiner Habra et al. (2021) and M2S Climaco Pinto et al. (2022). Next we apply GromovMatcher to align experimental patient studies of liver and pancreatic cancer to a reference dataset and associate the shared metabolic features to each patient's alcohol intake. Through these efforts, we demonstrate how GromovMatcher data pooling improves our ability to discover biomarkers of lifestyle risk factors associated with several types of cancer.

Results

GromovMatcher algorithm

GromovMatcher uses the mathematical framework of OT to find all matching metabolic features between two untargeted metabolomic datasets (**Fig. 1**). It accepts two LC-MS datasets with possibly different numbers of metabolic features and samples. Each feature, fx_i, in Dataset 1 and fy_j in Dataset 2, is identified by its *m/z*, RT, and vector of feature intensities across samples (**Fig. 1a**). The primary tenet of GromovMatcher is that shared metabolic features have similar correlation patterns in both datasets and can be matched based on the distance/correlations between their feature intensity vectors. Specifically, GromovMatcher computes the pairwise distances between the feature intensity vectors of each metabolic feature in a dataset and saves them into a distance matrix, one per dataset (**Fig. 1b**). In practice, we use either the Euclidean distance or the cosine distance (negative of correlation) to perform this step (Methods). The resulting distance matrices contain information about the feature intensity similarity within each study. Using optimal transport, we can deduce shared subsets of metabolic features in both datasets which have corresponding feature intensity distance structures.

OT was originally developed to optimize the transportation of soil for the construction of forts Monge (1781) and was later generalized through the language of probability theory and linear programming Kantorovich (2006), leading to efficient numerical algorithms and direct applications to planning problems in economics. The ability of OT to efficiently match source to target locations found applications in data science for the alignment of distributions Courty et al. (2017); Alvarez-Melis et al. (2019) and was generalized by the Gromov-Wasserstein (GW) method Peyré et al. (2016); Alvarez-Melis and Jaakkola (2018) to align datasets with features of differing dimensions.

In practice, a sizeable fraction of the metabolic features measured in one study may not be present in the other. Hence, in most cases only a subset of features in both datasets can be matched. Recent GW formulations for unbalanced matching problems Sejourne et al. (2021) allow for matching only subsets of metabolic features with similar intensity structures (**Fig. 1c**). To incorporate additional feature information, we modify the optimization objective of unbalanced GW to penalize feature matches whose *m/z* differences exceed a fixed threshold (Methods, **Appendix 1**). The optimization of this objective computes a *coupling matrix* $\tilde{\Pi}$ where each entry $\tilde{\pi}_{ij} \geq 0$ indicates the level of confidence in matching metabolic feature fx_i in Dataset 1 to fy_j in Dataset 2.

Differences in experimental conditions can induce variations in RT between datasets that can be nonlinear and large in magnitude Zhou et al. (2012); Climaco Pinto et al. (2022); Habra et al. (2021). In the spirit of previous methods for LC-MS batch or dataset alignment Smith et al. (2006); Brunius et al. (2016); Liu et al. (2020); Vaughan et al. (2012); Habra et al. (2021); Climaco Pinto et al. (2022); Skoraczynski et al. (2022), the learned coupling $\tilde{\Pi}$ is used to estimate a nonlinear map (drift function) between RTs of both datasets by weighted spline regression, which allows us to filter unlikely matches from the coupling matrix to obtain a refined coupling matrix $\hat{\Pi}$ (**Fig. 1d**, Methods). An optional thresholding step removes matches with small weights from the coupling matrix. The final output of GromovMatcher is a binary matching matrix M where M_{ij} is equal to 1 if features fx_i and fy_j are matched and 0 otherwise. Throughout the paper, we refer to the two variants of GromovMatcher, with and without the optional thresholding step as GMT and GM respectively.

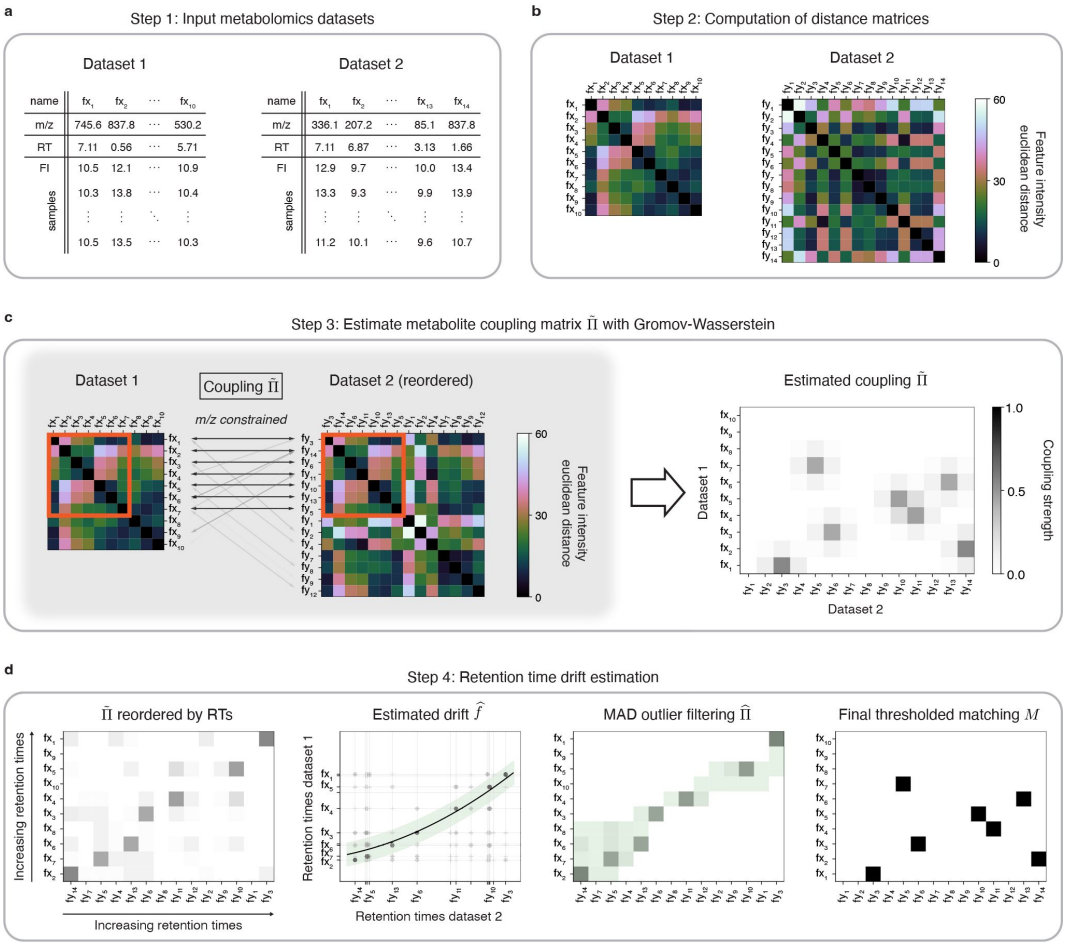


Figure 1.

An optimal transport approach for combining untargeted metabolomics datasets (GromovMatcher).

(**a**) Inputs are two LC-MS datasets of unlabeled metabolic features (rows) identified by their *m/z*, RT, and feature intensities across biospecimen samples. Both studies can have differing numbers of metabolic features and samples. (**b**) In both datasets, the intensities across samples of each metabolic feature are formed into a vector and Euclidean distances between these feature vectors are computed and stored in a distance matrix. (**c**) Based on the technique of optimal transport, the unbalanced GW algorithm learns a coupling matrix $\tilde{\Pi}$ that places large weights $\tilde{\Pi}_{ij} \geq 0$ when fx_i and fy_y likely correspond to the same metabolic feature. It optimizes $\tilde{\Pi}$ to match features with similar pairwise distances (red outlined boxes) whose *m/z* ratios are close. (**d**) The final step of GromovMatcher plots the retention times of features from both datasets against each other and fits a spline interpolation *f* weighted by the estimated coupling weights n. This retention time drift function is then used to set all entries $\tilde{\Pi}_{ij}$ to zero for those outlier pairs (fx_i, fy_y) which exceed twice the median absolute deviation (MAD) around $\hat{f}$ (green highlighted region). Finally, the coupling matrix $\tilde{\Pi}$ is filtered and/or thresholded to obtain a refined coupling $\hat{\Pi}$ which is then binarized to obtain a one-to-one matching *M* between a subset of metabolite pairs in both datasets.

Validation on ground-truth data

We first evaluate the performance of GromovMatcher using a real-world untargeted metabolomics study of cord blood across 499 newborns containing 4,712 metabolic features characterized by their *m/z*, RT, and feature intensities Alfano et al. (2020). To generate ground-truth data, we randomly divide the initial dataset into two smaller datasets sharing a subset of features (**Fig. 2**). We simulate diverse acquisition conditions by adding noise to the *m/z* and RT of dataset 2, and to the feature intensities in both datasets. Moreover, we introduce an RT drift in dataset 2 to replicate the retention time variations observed in real LC-MS experiments (Methods and Materials). For comparison, we also test M2S Climaco Pinto et al. (2022) and metabCombiner Habra et al. (2021), both of which use *m/z*, RT, and median or mean feature intensities to match features (**Fig. 3**). MetabCombiner is supplied with 100 known shared metabolic features to automatically set its hyperparameters, while M2S parameters are manually fine-tuned to optimize the F1-score in each scenario (**Appendix 2**). We assess the performance of GM, GMT, metabCombiner, and M2S across 20 randomly generated dataset pairs in terms of their precision (fraction of true matches among the detected matches) and recall/sensitivity (fraction of true matches detected) averaged across 20 dataset pairs.

To investigate how the number of shared features affects dataset alignment, we generate pairs of LC-MS datasets with low, medium, and high feature overlap (25%, 50%, and 75%), while maintaining a medium noise level (Methods). Here we find that GM and GMT generally outperform existing alignment methods, with a recall above 0.95 while metabCombiner and M2S tend to be less sensitive (**Fig. 3b**). All methods drop in precision as the feature overlap is decreased, with GM and GMT still maintaining an average precision above 0.8.

Next we evaluate all four methods at low, moderate, and high noise levels for pairs of datasets with 50% overlap in their features (Methods). Our results show that GMT, GM, and M2S maintain an average recall above 0.89, while metabCombiner's recall drops below 0.6 for high noise. At large noise levels, RT drift estimation becomes more challenging, leading to a higher rate of false matches between metabolites (lower precision) for all four methods (**Fig. 3 - figure supplement 1**). Nevertheless, GMT obtains a high average precision and recall of 0.86 and 0.92 respectively.

A notable difference between GM, metabCombiner, and M2S lies in their use of feature intensities. MetabCombiner expects that the mean feature intensity rankings are identical across studies, while M2S assumes that shared features have similar median intensities. In contrast, GM uses both the mean feature intensities and their variances and covariances. In practice, differences in experimental assays or study populations can lead to greater variation in feature intensities, making matchings based on these statistics less reliable. Centering and scaling the feature intensities to unit variance avoids potential biases arising from inconsistent feature intensity magnitudes, but preserves correlations that GM leverages.

Exploring this further, we test how sensitive all four methods are to centering and scaling of feature intensities. MetabCombiner and M2S are tuned using the same methodology as for non-centered and non-scaled data. For M2S, we match features solely based on their *m/z* and RT. In this experiment (**Figure 3 - figure supplement 2**), the absence of intensity magnitude information significantly affects metabCombiner's performance and, to a lesser extent, M2S. GM and GMT still obtain accurate matchings, due to their use of correlation structures which are preserved under centering and scaling.

Application to EPIC data

Next, we apply GM, metabCombiner and M2S to align datasets from the European Prospective Investigation into Cancer and Nutrition (EPIC) cohort, a prospective study conducted across 23 European centers. EPIC comprises more than 500,000 participants who provided blood samples at

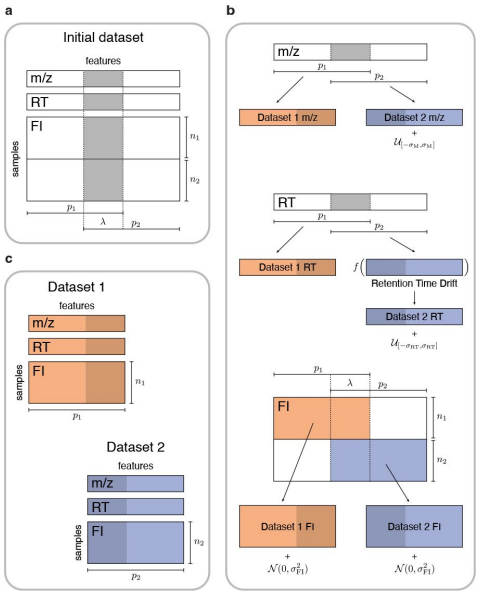


Figure 2.

Simulated data for testing untargeted metabolomics alignment methods.

(**a**) Initial LC-MS dataset taken from the EXPOsOMICS project with m/z, RT, and feature intensities of $p = 4,712$ metabolites identified in cord blood across $n = 499$ newborns. (**b**) Newborns (rows) are split into two disjoint groups of sizes $n_1 = 249$ and $n_2 = 250$ respectively and metabolic features (columns) are split into two equal groups of size $p_1 = p_2$ with overlap λ_p where $\lambda = 0.25, 0.5, 0.75$ (Methods). Datasets are perturbed by additive noise of magnitude (σ_M, σ_{RT}, σ_{FI}) and a nonlinear drift $f(x)$ is applied to the RTs of dataset 2. (**c**) The two resulting datasets share $\lambda = 25\%, 50\%$, or 75% of the original dataset's metabolic features.

recruitment Riboli et al. (2002). Untargeted metabolomics data were successively acquired in several studies nested within the full cohort.

In the present work, we use LC-MS data from the EPIC cross-sectional (CS) study Slimani et al. (2003) and two matched case-control studies nested within EPIC, on hepatocellular carcinoma (HCC) Stepien et al. (2016, 2021) and pancreatic cancer (PC) Gasull et al. (2019). LC-MS untargeted metabolomic data were acquired at the International Agency for Research on Cancer, making use of the same platform and methodology (Methods). The number of samples and features in each study is displayed in **Fig. 4a**.

Loftfield et al. Loftfield et al. (2021) previously matched features from the CS, HCC, and PC studies in EPIC for alcohol biomarker discovery. The authors first identified 205 features (163 in positive and 42 in negative mode) associated with alcohol intake in the CS study. These features were then manually matched by an expert to features in both the HCC and PC studies (Methods). In our analysis, we use these features as a validation set and compare each method's matchings to the expert manual matchings on this subset. Due to the imbalance between the number of positive and negative mode features in the validation subset, our main analysis focuses on the alignment results of CS with HCC and CS with PC in positive mode. We delegate the matching results between the negative mode studies to Appendix 4.

In this section, we use the same settings for GM as in our simulation study, and do not apply an additional thresholding step. The parameters of metabCombiner and M2S are calibrated using the validation subset as prior knowledge (Appendix 2).

Preliminary analysis of the validation subset reveals inconsistencies in the mean feature intensities (**Figure 4 - figure supplement 1**), but Figure 4b shows that on centered and scaled data, the 90 expert matched features shared between the CS and HCC studies have similar correlation structures. Hence, to avoid potential errors we center and scale the feature intensities which improves the performance of all three methods tested below (**Appendix 4**, **Appendix 4 - Table 1**).

Hepatocellular carcinoma

Here we analyze the quality of the matchings obtained by GM, M2S, and metabCombiner between the CS and HCC datasets in positive mode. Both GM and M2S identify approximately 1000 shared features while metabCombiner finds a smaller number of about 700 shared features. We refer the reader to **Figure 4- figure supplement 2a** for the precise matched feature sizes and details on the agreement between the feature matchings of all three methods.

We evaluate the performance of metabCombiner, M2S, and GM on the validation subset in positive mode (**Fig. 4c**), which consist of 90 features from the CS study manually matched to features from the HCC study and 73 features specific to the CS study. MetabCombiner demonstrates precise matching but lacks sensitivity. M2S's precision and recall are comparable with GM, in contrast to its performance on simulated data. This can be attributed to the RT drift shape between the CS and HCC studies (**Appendix 2**), which is estimated to be close to linear (**Figure 4 - figure supplement 3**). Because the parameters of M2S are fine-tuned in the validation subset, it is able to learn this linear drift and apply tight RT thresholds to achieve accurate matchings. In contrast to metabCombiner and M2S, the GM algorithm is not given any prior knowledge of the validation subset, and nevertheless demonstrates the highest precision and recall rates of the three methods (**Fig. 4c**). Figure 4b shows how GM recovers the majority of the expert matched pairs by leveraging the shared correlations.

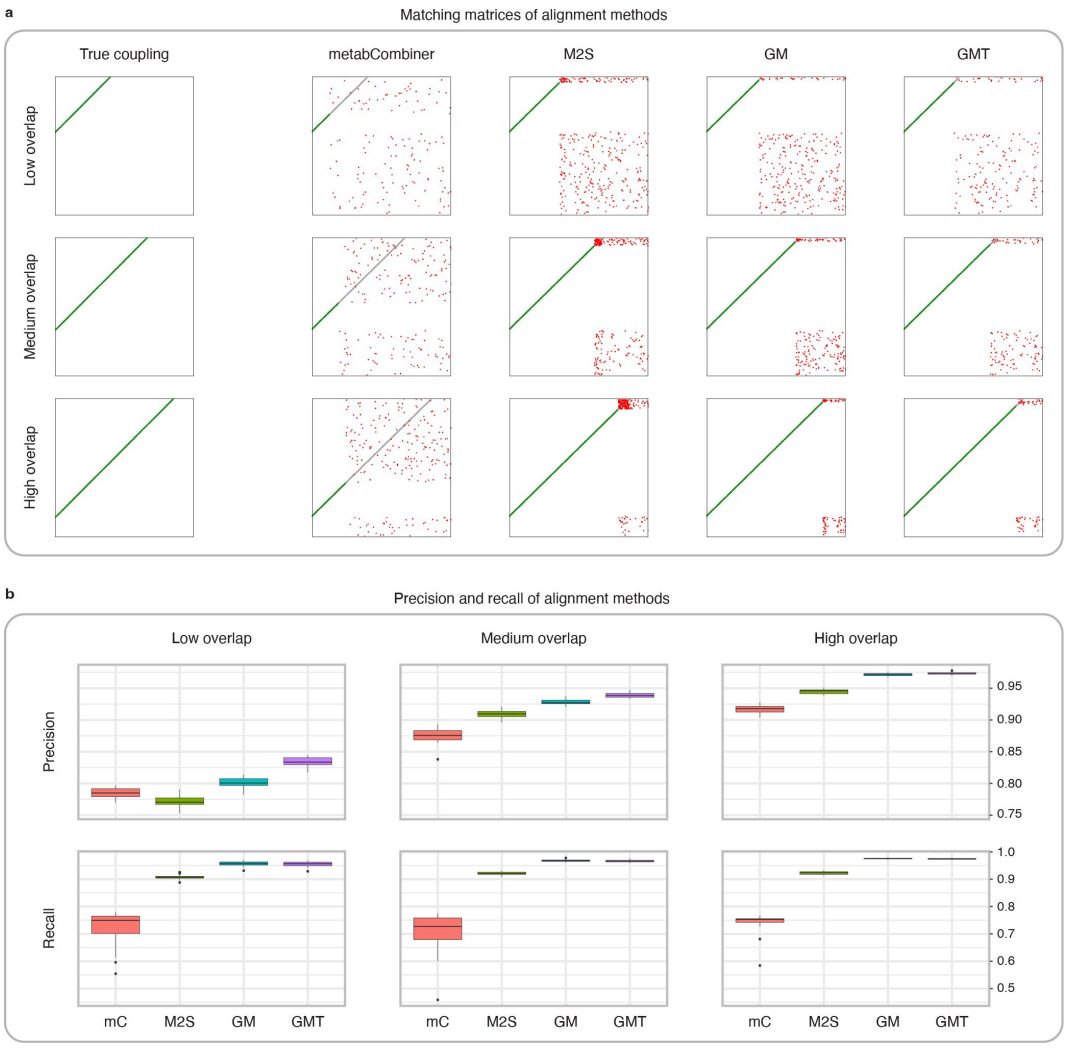


Figure 3.

Comparison of MetabCombiner, M2S, and GromovMatcher on simulated data.

(**a**) Ground-truth matchings, and matchings inferred by metabCombiner, M2S, GM, and GMT. Pairs of datasets are generated for three levels of overlap (low, medium and high), with a medium noise level (Methods). Matches correctly recovered (true positives) are represented in green. True matches that are not recovered (false negatives) are highlighted in grey. Incorrect matches (false positives) are plotted in red. Features in rows and columns of matching matrices are reordered for visual clarity. (**b**) Average precision and recall on 20 randomly generated pairs of datasets, for three levels of overlap (low, medium and high) with a medium noise level.

Pancreatic cancer

Matching features between the CS and PC studies in positive mode, GM and M2S identify approximately 1000 common features, while metabCombiner detects approximately 600 matches (**Figure 4- figure supplement 2b**). We examine the performance of all three methods on the validation subset consisting of 66 manually matched features between CS and PC along with 97 features specific to the CS study. As before, GM and M2S have high recall while the recall of metabCombiner is less than 0.5.

A decrease in precision is observed for both GM and M2S compared to the previous CS-HCC matchings. We therefore manually inspect the false positive matches; the set of CS features matched by the method to the PC study but explicitly examined and left unmatched in the expert manual matching. Assessing the GM results, we identify 7 false positive feature matches. Upon secondary inspection, 3 pairs are revealed as correct matches that were not initially identified in the expert matching. M2S finds 11 false positive matches which include the 7 false positives recovered by GM. Manual examination of the 4 remaining pairs reveals 2 clear mismatches. These results highlight the advantage of using automated methods for data alignment, as both GM and M2S detect correct matches that were not identified by experts, with GM being more precise than M2S.

Illustration for alcohol biomarker discovery

Loftfield et al. Loftfield et al. (2021) identified biomarkers of habitual alcohol intake by first performing a discovery step, where they examined the relationship between alcohol intake and metabolic features in the CS study. They then manually matched the significant features in CS to features from the HCC and PC studies, and repeated the analysis with samples from the HCC and PC studies to determine whether the association with alcohol intake persisted. This led to the identification of 10 features possibly associated with alcohol intake (**Fig. 5a**).

To extend this analysis and illustrate the benefit of GM automatic matching for biomarker discovery, we use GM to pool features from the CS, HCC, and PC studies, and examine the relationship between metabolic features and alcohol intake in the pooled study (Methods and **Fig. 5b**).

Applying an FDR correction on the pooled study, we identify 243 features associated with alcohol intake, including 185 features consistent with the discovery step of Loftfield et al. Loftfield et al. (2021), and 55 newly discovered features (**Fig. 5c**). Using the more stringent Bonferroni correction on the pooled data, we identify 36 features shared by all three studies that are significantly associated with alcohol intake. These features include all 10 features identified in Loftfield et al. (**Fig. 5c**). These findings highlight the potential benefits of using GM automatic matching for biomarker discovery in untargeted metabolomics data. Additional information regarding the methodology and findings of our GM and Loftfield et al. analyses can be found in Methods and **Appendix 4**.

Discussion

LC-MS metabolomics has emerged as an increasingly powerful tool for biological and biomedical research, offering promising opportunities for epidemiological and clinical investigations. However, integrating data from different sources remains challenging. To address this issue, we introduce GromovMatcher, a method based on optimal transport that automatically aligns LC-MS

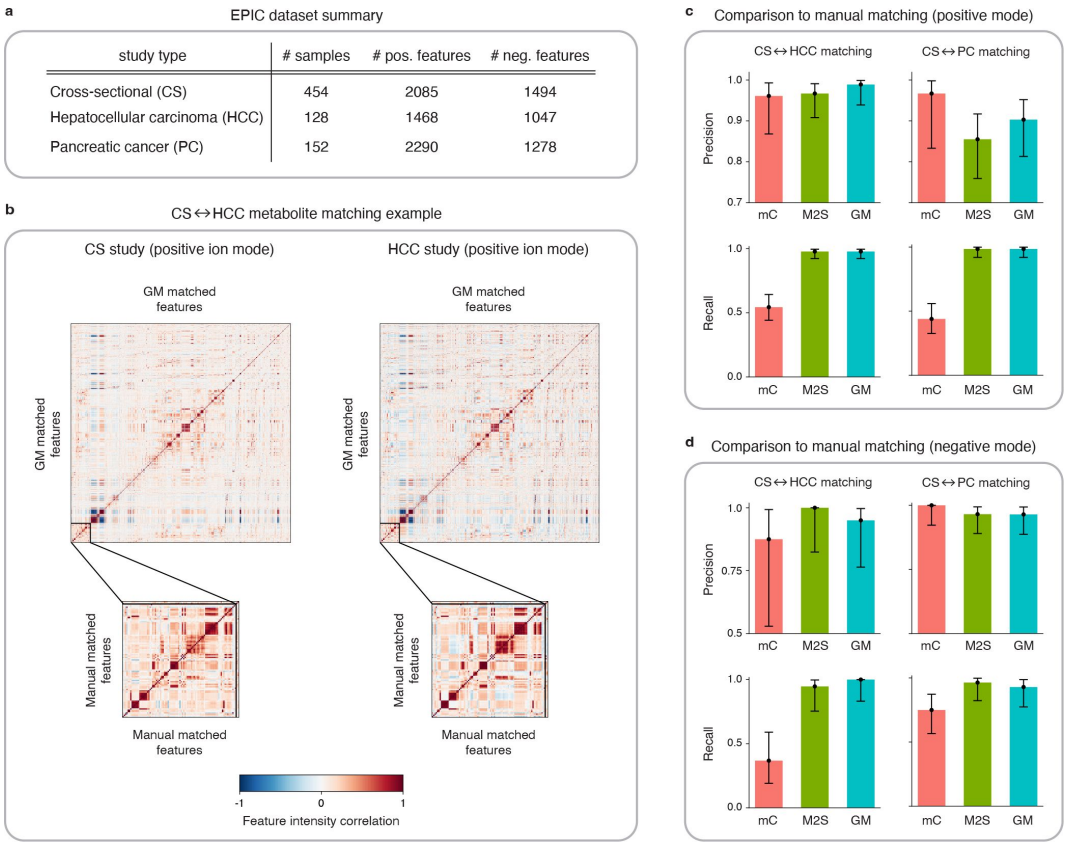


Figure 4.

Application of GromovMatcher and comparison to existing methods on EPIC dataset.

(**a**) Dimensions of the three EPIC studies used. For each ionization mode, the cross-sectional (CS) study is aligned successively with the hepatocellular carcinoma (HCC) study and the pancreatic cancer (PC) study. (**b**) Demonstration of expert manual matching and GromovMatcher (GM) matching between the CS and HCC studies in positive mode. Experts manually match 90 features from Loftfield et al. (2021) and the correlation matrices of these features in both datasets have similar structure (bottom two matrices). GM discovers 996 shared features between the CS and HCC datasets which have similar correlation structure (top two matrices). We validate that 88 of the 90 features from the manually expert matched subset are contained in the set of features matched by GM. (**c**) Performance of metabCombiner (mC), M2S and GM in positive mode. Precisions and recalls are measured on a validation subset of 163 manually examined features, and 95% confidence intervals are computed using modified Wilson score intervals. (**d**) Performance of mC, M2S and GM in negative mode. Precision and recall are measured on a validation subset of42 manually examined features, and 95% confidence intervals are computed using modified Wilson score intervals. See **Table 2** and **Table 3** for exact precisions, recalls, and confidence intervals in positive and negative mode respectively.

Study	Manual matches found in positive mode	Manual matches found in negative mode
Hepatocellular carcinoma (HCC)	90	19
Pancreatic cancer (PC)	66	28

Table 1

Results from the manual matching conducted for Loftfield et al. *Loftfield et al. (2021)*.

Features from the CS study (163 features in positive mode, 42 features in negative mode) were manually investigated for matches in the HCC and PC studies.

	CS ⟷ HCC		CS ⟷ PC	
Method	**Precision**	**Recall**	**Precision**	**Recall**
GromovMatcher	0.989 (0.939, 0.999)	0.978 (0.923, 0.996)	0.903 (0.813, 0.952)	0.985 (0.919, 0.999)
M2S	0.967 (0.908, 0.991)	0.978 (0.923, 0.996)	0.855 (0.759, 0.917)	0.985 (0.919, 0.999)
metabCombiner	0.961 (0.868, 0.993)	0.544 (0.442, 0.643)	0.967 (0.833, 0.998)	0.439 (0.326, 0.559)

Table 2

Precision and recall on the EPIC validation subset in positive mode.

95% confidence intervals were computed using modified Wilson score intervals Brown et al. (2001); Agresti and Coull (1998).

	CS ⟷ HCC		CS ⟷ PC	
Method	**Precision**	**Recall**	**Precision**	**Recall**
GromovMatcher	0.950 (0.764, 0.997)	1.000 (0.832, 1.000)	0.929 (0.774, 0.987)	0.929 (0.774, 0.987)
M2S	1.000 (0.824, 1.000)	0.947 (0.754, 0.997)	0.931 (0.780, 0.988)	0.964 (0.823, 0.998)
metabCombiner	0.875 (0.529, 0.993)	0.368 (0.191, 0.590)	1.000 (0.845, 1.000)	0.750 (0.566, 0.873)

Table 3

Precision and recall on the EPIC validation subset in positive mode.

95% confidence intervals were computed using modified Wilson score intervals Brown et al. (2001); Agresti and Coull (1998).

data from pairs of studies. Our method exhibits superior performance on both simulated and real data when compared to existing approaches. Additionally, it presents a user-friendly interface with few hyperparameters.

While GromovMatcher is robust to noise and variations in data, it may face limitations when aligning LC-MS studies from populations with different characteristics, where the correlation structures between features may be inconsistent across studies. In this case, the base assumption of GromovMatcher can be relaxed by focusing on subsamples with similar characteristics, as exemplified in a recent study Gomari et al. (2022).

A current limitation is that GromovMatcher does not account for more than two datasets simultaneously, although this can be overcome by aligning multiple studies to a chosen reference dataset, as demonstrated in our biomarker experiments. The extension of Gromov-Wasserstein to multiple distributions Beier et al. (2022) is another promising approach for generalizing GromovMatcher to multiple dataset alignment. Further improvements can be made by incorporating existing knowledge about the studies being matched, such as known shared features, samples in common, or MS/MS data.

The results obtained from GromovMatcher are highly promising, opening the door for various analyses of metabolomic datasets acquired in different experimental laboratories. Here we demonstrated the potential of GromovMatcher in expediting the combination and meta-analysis of data for biomarker and metabolic signature discovery. The matchings learned by GromovMatcher also allow for comparison between experimental protocols by assessing the drift in *m/z*, RT, and feature intensities across studies. Finally, inter-institutional annotation efforts can directly benefit from incorporating this method to transfer annotations between aligned datasets. Bridging the gap between otherwise incompatible LC-MS data, GromovMatcher enables seamless comparison of untargeted metabolomics experiments.

Methods and Materials

GromovMatcher method overview

GromovMatcher accepts as input two feature tables from separate LC-MS untargeted metabolomics studies. Each feature table for dataset 1 and dataset 2 consists of n_1, n_2 biospecimen samples respectively and p_1, p_2 metabolic features respectively detected in the study. Features in dataset 1 are given the label fx_i for $i = 1,\ldots,p_1$. Every feature is characterized by a mass-to-charge ratio (*m/z*) denoted by m_i^x, a retention time (RT) denoted by RT_i^x, and a vector of intensities across all samples written as $X_i \in \mathbb{R}^{n_1}$. Similarly, features in dataset 2 are labeled as fy_j for $j = 1,\ldots,p_2$ and are characterized by their $m/z\, m_j^y$, retention time RT_j^y, and a vector of intensities across all samples $Y_i \in \mathbb{R}^{n_2}$.

Our goal is to identify pairs of indexes (i, j) with $i \in \{1, \ldots, p_1\}$ and $j \in \{1, \ldots, p_2\}$, such that fx_i and fy_j correspond to the same metabolic feature. More formally, we aim to identify a *matching matrix* $M \in \{0,1\}^{p_1 \times p_2}$ such that $M_{ij} = 1$ if fx_i and fy_j correspond to the same feature, hereafter referred to as *matched* features. Otherwise, we set $M_{ij} = 0$.

Because the *m/z* and RT values of metabolomic features are often noisy and subject to experimental bias, our matching algorithm leverages metabolite feature intensities X_i, Y_j to produce accurate dataset alignments. The GromovMatcher method is based on the idea that signal

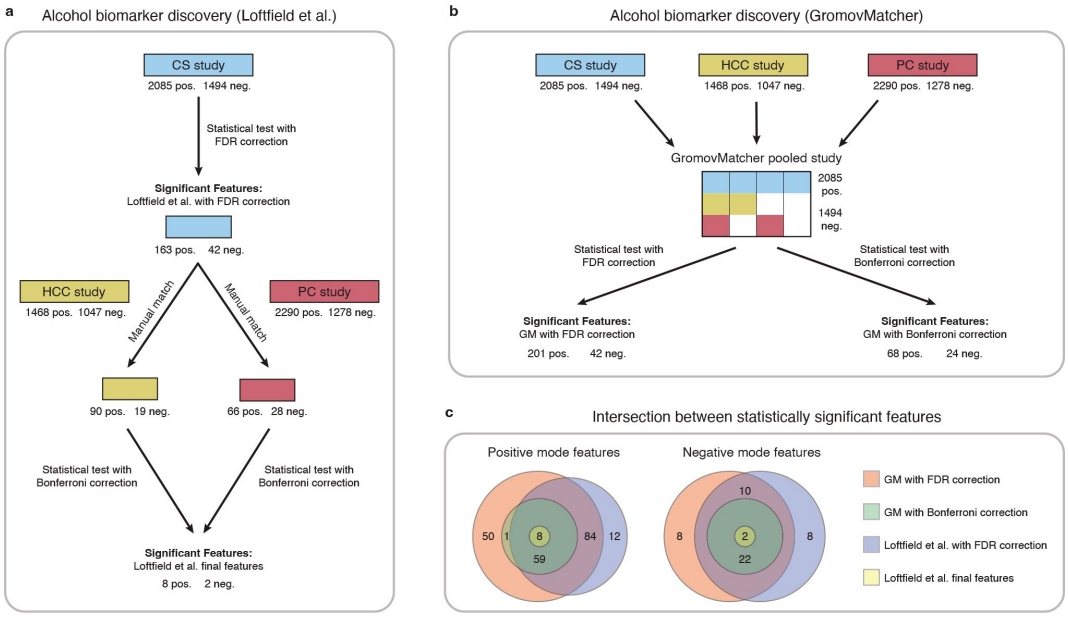


Figure 5.

Comparison of GromovMatcher and Loftfield et al. Loftfield et al. (2021) analysis for alcohol biomarker discovery on EPIC data.

(**a**) Loftfield study implemented a discovery step, examining the relationship between alcohol intake and metabolic features in the CS study. The significant features in CS were manually matched to features from the HCC and PC and the analysis was repeated using samples from the HCC and PC studies. After this step, 10 features associated with alcohol intake were identified. (**b**) GromovMatcher analysis begins by matching features from CS study to HCC and PC studies respectively (top blue, yellow, and red boxes). Samples corresponding to each CS feature are combined with the samples of its matched feature in the HCC study, PC study, or both. This generates a larger pooled data matrix with the same number of features as the CS study but with more samples pooled across the three original studies (center matrix). Because some features in the CS study may not have matches in HCC or PC, the corresponding entries in the pooled matrix are set to NaN/missing values (white regions in matrix). Each column/feature in this matrix is statistically tested for association with alcohol intake (ignoring missing values) and an FDR or a stricter Bonferroni correction is performed to retain only a subset of features from the pooled study that have a strong association. (**c**) Venn diagrams show intersection of feature sets (in positive and negative mode) found to be associated with alcohol intake by one of the four different analyses.

intensities of the same metabolites measured in two different studies should exhibit similar correlation structures, in addition to having compatible *m/z* and RT values. Here we define the Pearson correlation for vectors $u, v, \epsilon\, \mathbb{R}^n$ as

$$\mathrm{corr}(u, v) = \frac{\langle u - \bar{u}, v - \bar{v} \rangle}{\|u - \bar{u}\| \|v - \bar{v}\|} \tag{1}$$

where we define

$$\bar{u} = \frac{1}{n} \sum_{i=1}^{n} u_i, \quad \|u\| = \sqrt{\sum_{i=1}^{n} u_i^2}, \quad \langle u, v \rangle = \sum_{i=1}^{n} u_i v_i \tag{2}$$

as the mean value, Euclidean norm and inner product respectively. If measurements X_i, Y_j correspond to the same underlying feature, and similarly, measurements X_k, Y_l share the same an underlying feature, we expect that

$$\mathrm{corr}(X_i, X_k) \approx \mathrm{corr}(Y_j, Y_l). \tag{3}$$

This idea that the feature intensities of shared metabolites have the same correlation structure in both datasets also holds more generally for distances, under a suitable choice of distance. For example, the correlation coefficient corr(*u, V*) can be turned into a dissimilarity metric by defining

$$d^{\cos}(u, v) = \sqrt{1 - \mathrm{corr}(u, v)} \tag{4}$$

commonly referred to as the *cosine distance*. Preservation of feature intensity correlations then trivially amounts to the preservation of cosine distances.

Another classical notion of distance between vectors $u, v, \epsilon\, \mathbb{R}^n$ is the normalized Euclidean distance

$$d^{\mathrm{euc}}(u, v) = \frac{1}{\sqrt{n}} \|u - v\| = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (u_i - v_i)^2} \tag{5}$$

which is equal to the cosine distance (up to constants) when the vectors *u, v* are centered and scaled to have zero mean and a standard deviation of one. The Euclidean distance depends on the magnitude or mean intensity of metabolic features, and hence is a useful metric for matching metabolites as long as these mean feature intensities are reliably collected.

To summarize, the main tenant of GromovMatcher is that if measurements X_i, Y_j correspond to the same feature and X_k, Y_l correspond to the same feature, then for suitably chosen distances $d_x : \mathbb{R}^{n_1} \times \mathbb{R}^{n_1} \to \mathbb{R}$ and $d_y : \mathbb{R}^{n_2} \times \mathbb{R}^{n_2} \to \mathbb{R}$, these distances are preserved

$$d_x(X_i, X_k) \approx d_y(Y_j, Y_l) \tag{6}$$

across both datasets. In this paper, the distances d_x, d_y are taken to be the normalized Euclidean distances in (5). We take care to specify those experiments where the metabolic features *X* and *Y* are centered and scaled. In these cases, implicitly the Euclidean distance between normalized feature vectors becomes the cosine distance (4) between the original (unnormalized) feature vectors.

Unbalanced Gromov–Wasserstein

The goal of GromovMatcher is to learn a matching matrix $M \in \{0,1\}^{p_1 \times p_2}$ that gives an alignment between a subset of metabolites in both datasets. However, searching over the combinatorially large set of binary matrices would be an inefficient approach for dataset alignment. The mathematical framework of optimal transport Peyré et al. (2019) instead enlarges this space of binary matrices to the set of *coupling matrices* with real nonnegative entries $\Pi \in \mathbb{R}_+^{p_1 \times p_2}$. The entries $\prod_{ij}$ with large weights indicate that feature fx_i in dataset 1 and feature fy_j in dataset 2 are a likely match. Taking inspiration from (6), we minimize the following objective function

$$\mathcal{E}(\Pi) = \sum_{i,k=1}^{p_1} \sum_{j,l=1}^{p_2} \Pi_{ij} \Pi_{kl} \left| d_x(X_i, X_k) - d_y(Y_j, Y_l) \right| \tag{7}$$

to estimate the coupling matrix $\prod$.

A standard approach is to optimize this objective over all coupling matrices $\prod$ under exact marginal constraints $\Pi \mathbf{1}_{p_2} = \frac{1}{p_1} \mathbf{1}_{p_1}, \Pi^T \mathbf{1}_{p_1} = \frac{1}{p_2} \mathbf{1}_{p_2}$. Here we define $\mathbf{1}_n$ is the ones vector of length n, and $\Pi_1 = \Pi \mathbf{1}_{p_2}, \Pi_2 = \Pi^T \mathbf{1}_{p_1}$ denote the column and row sums of the coupling matrix. Objective (7) under these exact marginal constraints defines a distance between the two sets of metabolic feature vectors $\{X_i\}_{i=1}^{p_1}, \{Y_i\}_{i=1}^{p_2}$ known as the Gromov–Wasserstein distance Mémoli (2011), a generalization of optimal transport to metric spaces. Note that for pairs X_i, Y_j and X_k, Y_j, for which $d_x(X_i, X_k) \approx d_y(Y_j, Y_l)$, the entries $\prod_{lj}$, $\prod_{kl}$, are penalized less and hence matches between features fx_i, fy_j and features fx_k, fy_l, are more favored. In our optimization, we avoid enforcing exact marginal constraints on the marginal distributions $\prod \mathbf{1}_{p2}$ and $\prod \mathbf{1}_{p1}$ of our coupling matrix as this would enforce that all metabolites in both datasets are matched (Appendix 1). However, without any marginal constraints on the coupling $\prod$, the objective function (7) is trivially minimized by $\prod = 0$, leaving all metabolites in both datasets unmatched.

To account for this, we follow the ideas of unbalanced Gromov–Wasserstein (UGW) Sejourne et al. (2021) and add three regularization terms to our objective

$$\begin{aligned} \mathcal{L}_{\rho,\varepsilon}(\Pi) = \mathcal{E}(\Pi) &+ \rho D_{\mathrm{KL}} \left(\Pi_1 \otimes \Pi_1, a \otimes a \right) \\ &+ \rho D_{\mathrm{KL}} \left(\Pi_2 \otimes \Pi_2, b \otimes b \right) \\ &+ \varepsilon D_{\mathrm{KL}} (\Pi \otimes \Pi, (a \otimes b)^{\otimes 2}) \end{aligned} \tag{8}$$

where $\rho, \epsilon > 0$ and we define $a = \mathbf{1}_{p1}$, $b = \mathbf{1}_{p2}$. Here $\otimes$ denotes the Kronecker product. We define D_{KL} as the Kullback-Leibler (KL) divergence between two discrete distributions $\mu, \nu \in \mathbb{R}_+^p$ by

$$D_{\mathrm{KL}}(\mu, \nu) = \sum_{i=1}^{p} \mu_i \ln \left(\frac{\mu_i}{\nu_i} \right) - \sum_{i=1}^{p} \mu_i + \sum_{i=1}^{p} \nu_i \tag{9}$$

which measures the closeness of probability distributions.

The first two regularization terms in (8) enforce that the row sums and column sums of the coupling matrix $\prod$ do not deviate too much from a uniform distribution, leading our optimization to match as many metabolic features as possible. The magnitude of the regularizer ρ roughly enforces the fraction of metabolites in both datasets that are matched where large ρ implies most metabolites are matched across datasets. The final regularization term ϵ in (8) controls the smoothness (entropy) of the coupling matrix $\prod$ where larger values of ϵ encourage $\prod$ to put uniform weights on many of its entries, leading to less precision in the metabolite matches. However, increasing ϵ also leads to better numerical stability and a significant speedup of the

alternating minimization algorithm used to optimize the objective function (Appendix 1). In our implementation, we set ρ and ϵ to the lowest possible values under which our optimization converges, with ρ = 0.05 and ϵ = 0.005.

Our full optimization problem can now be written as

$$\mathrm{UGW}_{\rho,\varepsilon} = \min_{\Pi \in \mathbb{R}_+^{p_1 \times p_2}} \mathcal{L}_{\rho,\varepsilon}(\Pi). \tag{10}$$

The UGW objective function is optimized through alternating minimization based on the code of Sejourne et al. (2021) using the unbalanced Sinkhorn algorithm Séjourné et al. (2019) from optimal transport (Appendix 1).

Constraint on *m*/*z* ratios

Matched metabolic features must have compatible m / z so we enforce that $\prod_{ij}$ =0 when $|m_i^x - m_j^y| > m_{\mathrm{gap}}$ where m_{gap} is a user-specified threshold. Based on prior literature Loftfield et al. (2021); Hsu et al. (2019); Climaco Pinto et al. (2022); Habra et al. (2021); Chen et al. (2021) we set m_{gap} = 0.01ppm. Note that m_{gap} is not explicitly used in (10) but is rather enforced in each iteration of our alternating minimization algorithm for the UGW objective (Appendix 1).

Unlike the m/z ratios discussed above, RTs often exhibit a non-linear deviation (drift) between studies so we cannot enforce compatibility of RTs directly in our optimization. Instead, in the following step of our pipeline we ensure matched metabolite pairs have compatible RTs by estimating the drift function and subsequently using it to filter out metabolite matches whose RT values are inconsistent with the estimated drift.

Estimation of the RT drift and filtering

Estimating the drift between RTs of two studies is a crucial step in assessing the validity of metabolite matches and discarding those pairs which are incompatible with the estimated drift.

Let $\tilde{\Pi} \in \mathbb{R}_+^{p_1 \times p_2}$ be the minimizer of (10) obtained after optimization. We seek to estimate the RT drift function $f : \mathbb{R}_+ \to \mathbb{R}_+$ which relates the retention times of matched features between the two studies. Namely, if feature fx_i and feature fy_j correspond to the same metabolic feature, then we must have that $RT_j^y \approx f(RT_i^x)$.

We propose to learn the drift f through the weighted spline regression

$$\min_{f \in B_{n,k}} \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \tilde{\Pi}_{ij} \left| f\left(RT_i^x\right) - RT_j^y \right| \tag{11}$$

where $B_{n,k}$ is the set of n-order B-splines with k knots. All pairs (RT_i^x, RT_j^y) in objective (11) are weighted by the coefficients of $\tilde{\Pi}$ so that larger weights are given to pairs identified with high confidence in the first step of our procedure. The order of the B-splines was set to n = 3 by default, while the number of knots k was selected by 10-fold cross-validation.

Pairs identified as incompatible with the estimated RT drift are then discarded from the coupling matrix. To do this, we first take the estimated RT drift $\hat{f}$, and the set of pairs $S = \{i, j : \tilde{\Pi}_{i,j} \neq 0\}$ recovered in $\tilde{\Pi}$. We then define the residual associated with $(i, j) \in s$ as

$$r_{\hat{f}}(i,j) = |\hat{f}(RT_i^x) - RT_j^y|. \tag{12}$$

The 95% prediction interval and the median absolute deviation (MAD) of these residuals are given by

$$\begin{aligned} \text{PI} &= 1.96 \times \text{std}(\{r_{\hat{f}}(i,j), (i,j) \in \mathcal{S}\}) \\ \text{MAD} &= \text{median}(\{|r_{\hat{f}}(i,j) - \mu_r|, (i,j) \in \mathcal{S}\}) \\ \mu_r &= \text{median}(\{|r_{\hat{f}}(i,j)|, (i,j) \in \mathcal{S}\}) \end{aligned} \tag{13}$$

where $|S|$ is the size of S and the functions std and median denote the standard deviation and median respectively. Similar to the approach in Climaco Pinto et al. (2022), we create a new filtered coupling matrix $\hat{\Pi} \in \mathbb{R}_+^{p_1 \times p_1}$ given by

$$\hat{\Pi}_{ij} = \begin{cases} \tilde{\Pi}_{ij} & \text{if } r_{\hat{f}}(i,j) < \mu_r + r_{\text{thresh}} \\ 0 & \text{otherwise} \end{cases}. \tag{14}$$

where r_{thresh} is a given filtering threshold. Following Habra et al. (2021), the estimation and outlier detection step can be repeated for multiple iterations, to remove pairs that deviate significantly from the estimated drift and improve the robustness of the drift estimation. In our main algorithm, we use two preliminary iterations where estimate the RT drift and discard outliers outside of the 95% prediction interval by setting r_{thresh} = PI. We the re-estimate the drift and perform a final filtering step with the more stringent MAD by setting r_{thresh} = 2 × MAD.

At this stage, it is possible for $\hat{\Pi}$ to still contain coefficients of very small magnitude. As an optional postprocessing step, we discard these coefficients by setting all entries smaller than $\tau \max(\hat{\Pi})$ to zero, for some user-defined $\tau \in [0,1]$. Lastly, a feature from either study could have multiple possible matches, since $\hat{\Pi}$ can have more than one non-zero coefficient per row or column. Although reporting multiple matches can be helpful in an exploratory context, for the sake of simplicity in our analysis, the final output of GromovMatcher returns a one-to-one matching, as we only keep those metabolite pairs (*i,j*) where the entry $\hat{\Pi}_{ij}$ is largest in its corresponding row and column. All nonzero entries of $\hat{\Pi}$ which do not satisfy this criterion are set to zero. Finally, we convert $\hat{\Pi}$ into a binary matching matrix $M \in \{0,1\}^{p_1 \times p_2}$ with ones in place of its nonzero entries and this final output is returned to the user.

As a naming convention, we use the abbreviation GM for our GromovMatcher method, and use the abbreviation GMT when running GromovMatcher with the optional τ -thresholding step with τ = 0.3.

Metrics for dataset alignment

Every alignment method studied in this paper returns a binary *partial matching* matrix $M \in \{0,1\}^{p_1 \times p_1}$ which has at most one nonzero entry in each row and column. Specifically, M_{ij} = 1 if metabolic features *i* and *j* in both datasets correspond to each other and M_{ij} = 0 otherwise. In our simulated experiments, we compare the partial matching M to a known ground-truth partial matching matrix $M^* \in \{0,1\}^{p_1 \times p_2}$.

To do this, we first compute the number of true positives, false positives, true negatives, and false negatives as

$$\begin{aligned}
\mathrm{TP} &= \sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \mathbf{1}_{M_{ij}=1}\mathbf{1}_{M^*_{ij}=1} \\
\mathrm{FP} &= \sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \mathbf{1}_{M_{ij}=1}\mathbf{1}_{M^*_{ij}=0} \\
\mathrm{TN} &= \sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \mathbf{1}_{M_{ij}=0}\mathbf{1}_{M^*_{ij}=0} \\
\mathrm{FN} &= \sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \mathbf{1}_{M_{ij}=0}\mathbf{1}_{M^*_{ij}=1}
\end{aligned} \tag{15}$$

where **1** denotes the indicator function. Then we use these values to compute the precision and recall as

$$\begin{aligned}
\mathrm{Precision} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \\
\mathrm{Recall} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}.
\end{aligned} \tag{16}$$

Precision measures the fraction of correctly found matches out of all discovered metabolite matches, while recall, also know as sensitivity, measures the fraction of correctly matched pairs out of all truly matched pairs. These two statistics can be summarized into one metric called the F1-score by taking their harmonic mean

$$\mathrm{F1} = 2 \cdot \frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{17}$$

These three metrics, precision, recall, and the F1-score, are used throughout the paper to assess the performance of dataset alignment methods, both on simulated data where the ground-truth matching is known, and on the validation subset in EPIC, using results from the manual examination as the ground-truth benchmark.

Validation on simulated data

To assess the performance of GromovMatcher and compare it to existing dataset alignment methods, we simulate realistic pairs of untargeted metabolomics feature with known ground-truth matchings. This allows us to analyze the dependence of alignment methods on the number of shared metabolites, dataset noise level, and feature intensity centering and scaling.

Dataset generation

Our pairs of synthetic feature tables are generated from one real untargeted metabolomics study of 500 newborns within the EXPOsOMICS project, which uses reversed phase liquid chromatography-quadrupole time-of-flight mass spectrometry (UHPLC-QTOF-MS) system in positive ion mode Alfano et al. (2020). The original dataset is first preprocessed following the procedure detailed in Alfano et al. Alfano et al. (2020), resulting in p = 4,712 features measured in n = 499 samples available for subsequent analysis. Features and samples from the original study are then divided into two feature tables of respective size (n_1 ,p_1) and (n_2,p_2), with $n_1 + n_2 = n$ and p_1 ,$p_2 \leq p$. In order to do this, $n_1 = \lfloor n/2 \rfloor$ randomly chosen samples from the original study are

placed into dataset 1 and the remaining $n_2 = [n/2]$ samples from the original study are placed into dataset 2. Here [.] and [.] denote integer floor and ceiling functions. The features of the original study are randomly assigned to dataset 1, dataset 2, or both, allowing the resulting studies to have both common and study-specific features (**Fig. 2**). Specifically, for a fixed overlap parameter $\lambda \in [0,1]$, we assign a random subset of $\approx \lambda p$ features into both dataset 1 and dataset 2 while the remaining $\approx (1 - \lambda p)$ features are divided equally between the two studies such that $p_1 = p_2$. We choose $\lambda \in \{0.25, 0.5, 0.75\}$ corresponding to low, medium and high overlap. For more detailed information on how the dataset split is performed and for additional validation experiments with unbalanced dataset splits (e.g. $n_1 \neq n_2$, $p_1 \neq p_2$) we refer the reader to **Appendix 3**.

After generating a pair of studies, random noise is added to the *m/z*, RT and intensity levels of features in dataset 2 to mimic variations in data acquisition across two different experiments. The noise added to each *m/z* value in study 2 is sampled from a uniform distribution on the interval $[-\sigma_M, \sigma_M]$ with $\sigma_M = 0.01$ Climaco Pinto et al. (2022). The RTs of dataset 2 are first deviated by the function $f(x) = 1.1x + 1.3\sin(1.2\sqrt{x})$, corresponding to a systematic inter-dataset drift Habra et al. (2021); Climaco Pinto et al. (2022); Brunius et al. (2016). A uniformly distributed noise on the interval $[-\sigma_{RT}, \sigma_{RT}]$ is added to the deviated RTs of dataset 2, with $\sigma_{RT} \in \{0.2, 0.5, 1\}$ (in minutes) corresponding to low, moderate and high variations Climaco Pinto et al. (2022); Habra et al. (2021); Vaughan et al. (2012). Finally, we add a Gaussian noise $\mathcal{N}(0, \sigma_{FI}^2)$ to the feature intensities of both studies where σ_{FI} is the scalar variance of the noise. This noise perturbs the correlation matrices of dataset 1 and dataset 2, making matching based on feature intensity correlations more challenging. We vary σ_{FI} over the set of values {0.1, 0.5, 1}.

Given this data generation process, we test the performance of the four alignment methods (M2S, metabCombiner, GM, and GMT) under the parameter settings described below.

Dependence on overlap

We first assess how the performance of the four methods is affected by the number of metabolic features shared in both datasets. For each value of $\lambda = 0.25, 0.5, 0.75$ (low, medium and high overlap), we randomly generate 20 pairs of datasets with noise on the *m/z*, RT and feature intensities set to $\sigma_M = 0.01$, $\sigma_{RT} = 0.5$, $\sigma_{FI} = 0.5$. The precision and recall of each method at low, medium, and high overlap is recorded for each of the repetitions.

Noise robustness

Next we test the robustness to noise of each method by fixing the metabolite overlap fraction at $\lambda = 0.5$ and generating 20 random pairs of datasets at low ($\sigma_{RT} = 0.2$, $\sigma_{FI} = 0.1$), medium ($\sigma_{RT} = 0.5$, $\sigma_{FI} = 0.5$), and high $\sigma_{RT} = 1$, $\sigma_{FI} = 1$) noise levels. Similarly, the precision and recall of each method is saved for each noise level across the 20 repetitions.

Feature intensity centering and scaling

In order to test how all four methods are affected when the mean feature intensities and variance are not comparable across studies, we assess their performance when the feature intensities in both studies are mean centered and standardized to have unit standard deviation across all samples. We again generate 20 random pairs of datasets with medium overlap and medium noise, normalize the feature intensities in each pair of datasets, and compute the precision and recall of each method across the 20 repetitions.

EPIC data

We also evaluate our method on data collected within the European Prospective Investigation into Cancer and Nutrition (EPIC) cohort, an ongoing multicentric prospective study with over 500,000 participants recruited between 1992 and 2000 from 23 centers in 10 European countries, and who

provided blood samples at the inclusion in the study Riboli et al. (2002). In EPIC, untargeted metabolomics data were successively acquired in several studies nested within the full cohort.

In the present work, we use untargeted metabolomics data acquired in three studies nested in EPIC, namely the EPIC cross-sectional (CS) study Slimani et al. (2003) and two matched case-control studies nested within EPIC, on hepatocellular carcinoma (HCC) Stepien et al. (2016, 2021) and pancreatic cancer (PC) Gasull et al. (2019), respectively. All data were acquired at the International Agency for Research on Cancer, making use of the same plateform and methodology: UHPLC-QTOF-MS (1290 Binary Liquid chromatography system, 6550 quadrupole time-of-flight mass spectrometer, Agilent Technologies, Santa Clara, CA) using reversed phase chromatography and electrospray ionization in both positive and negative ionization mode.

In a previous analysis aiming at identifying biomarkers of habitual alcohol intake in EPIC, the 205 features associated with alcohol intake in the CS study were manually matched to features in both the HCC and PC studies Loftfield et al. (2021). The results from this manual matching are presented in **Table 1**. This matching process was based on the proximity of *m/z* and RT, using a matching tolerance of ±15 ppm and ±0.2 min, and on the comparison of the chromatograms of features in a quality control samples from both studies.

Preprocessing

In the HCC and PC studies, samples corresponding to participants selected as cases in either study (i.e., participants selected in the study because of a diagnosis of incident HCC or PC) are excluded. Indeed, the metabolic profiles of participants selected as controls are expected to be more comparable across studies than those of cases, especially if certain features are associated with the risk of HCC or PC. Apart from this additional exclusion criterion, the untargeted metabolomics data of each study is pre-processed following the steps described in Loftfield et al. Loftfield et al. (2021), to eliminate unreliable features and samples, impute missing values and minimize technical variations in the feature intensity levels.

Alcohol biomarker discovery

Loftfield et al. Loftfield et al. (2021) used the untargeted metabolomics data of the CS, HCC and PC studies in their alcohol biomarker discovery study in EPIC, without being able to automatically match their common features and pool the 3 datasets. Instead, the authors first implemented a discovery step, examining the relationship between alcohol intake and metabolic features measured in the CS study and accounting for multiple testing using a false discovery rate (FDR) correction. This led to the identification of 205 features significantly associated with alcohol intake in the CS study. In order to gauge the robustness of these associations, the authors of Loftfield et al. Loftfield et al. (2021) then implemented a validation step using data from two independent test sets. The first test set was composed of data from the EPIC HCC and PC studies, while the second was derived from the Finnish Alpha-Tocopherol, Beta-Carotene Cancer Prevention (ATBC) study. The 205 features identified in the discovery step were manually investigated for matches in the EPIC test set, and 67 features were effectively matched to features in the HCC or PC study, or both. The authors then evaluated the association between alcohol intake and those 67 features, applying a more conservative Bonferroni correction to determine whether the association with alcohol intake persisted. This step led to the identification of 10 features associated with alcohol intake (Extended Data **Fig. 5a**). The second test set was then used to determine whether those 10 features were also significant in the ATBC population, which was indeed the case.

To conduct a more in-depth investigation of the matchings produced by the GromovMatcher algorithm, we build upon the analysis previously conducted by Loftfield et al. Loftfield et al. (2021) by exploring potential alcohol biomarkers using a pooled dataset created from the CS, HCC, and PC studies. Our goal is to assess whether pooling the data leads to increased statistical power and allows for the detection of more features associated with alcohol intake. Namely, we

generate the pooled dataset by aligning a chosen reference dataset (CS study) with the HCC and PC studies successively using the GM matchings computed in both positive and negative mode (Methods and Extended Data **Fig. 5b**). Features that are not detected in either the HCC or PC studies are designated as 'missing' in the final pooled dataset for samples belonging to the respective studies where the feature is not found.

To evaluate the potential relationship between alcohol consumption and pooled metabolic features, we use a methodology akin to that of Loftfield et al. Loftfield et al. (2021). The self-reported alcohol intake data is adjusted for various demographic and lifestyle factors (age, sex, country, body-mass-index, smoking status and intensity, coffee consumption, and study) via the residual method in linear regression models. Feature intensities are also adjusted for technical variables (plate number and position within the plate) via linear mixed effect models. The significance of the association is assessed using correlation coefficients computed from the residuals for both selfreported alcohol intake and feature intensities. P-values are corrected using either false discovery rate (FDR) or Bonferroni correction to account for multiple testing. Corrected p-values less than 5% are considered significant.

Acknowledgements

We thank Jörn Dunkel for helpful advice on our manuscript. We acknowledge the MIT Super-Cloud and Lincoln Laboratory Supercomputing Center Reuther et al. (2018) for providing HPC resources that have contributed to the research results reported within this paper. G.S. acknowledges support through a National Science Foundation Graduate Research Fellowship under Grant No. 1745302. P.R. is supported by NSF grants IIS-1838071, DMS-2022448, and CCF-2106377. We are grateful to the Principal Investigators of each of the EPIC centers for sharing the data for our experimental application.

Data availability

The LC-MS data used to generate our simulated validation experiments is located at the bottom of the "Files" section in *https://www.ebi.ac.uk/metabolights/MTBLS1684/files* under filename 'metabolomics_normalized_data.xlsx'. The EPIC data is not publicly available, but access requests can be submitted to the Steering Committee *https://epic.iarc.fr/access/index.php*.

All code for the data preprocessing, figure generation, as well as the GromovMatcher algorithm and its comparison to other methods are available at: *https://github.com/sgstepaniants/GromovMatcher*. Instructions and examples for how to run the GromovMatcher method are provided in the Github repository. The metabCombiner implementation written by the original authors was taken from their Github codebase: *https://github.com/hhabra/metabCombiner*. The M2S implementation of the original authors was taken from their Github codebase: *https://github.com/rjdossan/M2S*.

Author contributions

M.B. and G.S. contributed equally and are joint first authors. P.R. and V.V. conceived the project. M.B. and G.S. developed the algorithms as well as performed the comparison to other alignment methods. P.K.R. analyzed and postprocessed the EPIC data and the analysis of our method on this data was performed by M.B. The manuscript was prepared by M.B., G.S., P.K.R, P.R., and V.V.

Materials and Correspondence

All correspondence and material requests should be addressed to V.V.

IARC disclaimer

Where authors are identified as personnel of the International Agency for Research on Cancer/World Health Organization, the authors alone are responsible for the views expressed in this article and they do not necessarily represent the decisions, policy, or views of the International Agency for Research on Cancer/World Health Organization.

Appendix 1

In this paper, we study how to match metabolic features across two datasets where Dataset 1 has p 1 metabolic features measured across n_1 patients and Dataset 2 has p_2 metabolic features measured across n_2 patients. Our goal is to identify pairs of indexes (i,j) with i e {1,... ,p_1} and j e {1,... ,p_2}, such that feature i in Dataset 1 and feature j in Dataset 2 correspond to the same metabolic feature. More formally, we aim to identify a *matching matrix* $M^* \in \{0,1\}^{p_1 \times p_2}$ such that $M^*_{i,j} = 1$ if features i in Dataset 1 and feature j in Dataset 2 correspond to the same feature, hereafter referred to as *matched* features. Otherwise we set $M^*_{i,j} = 0$ otherwise. We emphasize that a matching matrix M^* can have at most one nonzero entry in each row and column.

Both of the datasets we aim to match are obtained from liquid chromatography-mass spectrometry (LC-MS) experiments. Hence, for Dataset 1 each metabolite i ϵ [p_1] is labeled with a mass-to-charge (m/z) ratio m_i^1 . as well as a retention time (RT) given by RT_i^1. Additionally, each metabolite has a vector of intensities across patients denoted by $X_i \in \mathbb{R}^{n_1}$. Similarly, each metabolite j ϵ [p_2] in Dataset 2 is labeled by its m/z ratio m_j^2 , its retention time RT_j^2 and its vector of intensities across samples $Y_j \in \mathbb{R}^{n_2}$.

Correlations and distances between metabolomic features

Features cannot be aligned based on their m/z and RT alone as they are often too inconsistent across studies. Our method is based on the idea that, in addition to their m/z and RT being compatible, the signal intensities of metabolites measured in two different studies should exhibit similar correlation structures, or more generally exhibit similar distances between their intensity vectors. In other words, if feature intensity vectors $X_i \in \mathbb{R}^{n_1}, Y_j \in \mathbb{R}^{n_2}$ correspond to the same underlying feature $(M^*_{ij} = 1)$ and similarly if $X_k \in \mathbb{R}^{n_1}, Y_l \in \mathbb{R}^{n_2}$ correspond to the same feature $(M^*_{kl} = 1)$, then we expect that

$$\mathrm{corr}(X_i, X_k) \approx \mathrm{corr}(Y_j, Y_l) \quad \text{if} \quad \Pi^*_{ij} = \Pi^*_{kl} = 1. \tag{18}$$

Here we define corr(u, v) to be the Pearson correlation coefficient between two feature intensity vectors u, v ϵ $\mathbb{R}^n$ by

$$\mathrm{corr}(u, v) = \frac{\langle u - \bar{u}, v - \bar{v} \rangle}{\|u - \bar{u}\| \|v - \bar{v}\|} \tag{19}$$

where we define

$$\bar{u} = \frac{1}{n}\sum_{i=1}^{n} u_i, \quad \|u\| = \sqrt{\sum_{i=1}^{n} u_i^2}, \quad \langle u, v\rangle = \sum_{i=1}^{n} u_i v_i \tag{20}$$

as the mean value, Euclidean norm and inner product respectively. More generally, with d_x and d_y denoting two given distances on $\mathbb{R}^{n_1}$ and $\mathbb{R}^{n_2}$ respectively, we expect that

$$d_x(X_i, X_k) \approx d_y(Y_j, Y_l) \quad \text{if} \quad \Pi^*_{ij} = \Pi^*_{kl} = 1. \tag{21}$$

Throughout this paper, we use the normalized Euclidean distance defined for any $u, v \in \mathbb{R}^n$ as

$$d^{\text{euc}}(u, v) = \frac{1}{\sqrt{n}}\|u - v\| \tag{22}$$

where for d_x and d_y we take $n = n_1, n_2$ respectively. If the signal intensity vectors u, v are mean centered and normalized by their standard deviation as

$$u \mapsto \sqrt{n} \cdot \frac{u - \frac{1}{n}\sum_{i=1}^{n} u_i}{\left\| u - \frac{1}{n}\sum_{i=1}^{n} u_i \right\|} \tag{23}$$

and likewise for v, then it follows that

$$d^{\text{euc}}(u, v) = \sqrt{2\Big(1 - \text{corr}(u, v)\Big)} = \sqrt{2} d^{\cos}(u, v) \tag{24}$$

where we denote $d^{\cos}(u, v) = \sqrt{1 - \text{corr}(u, v)}$ as the cosine distance. For the purposes of this paper, we will always assume that d_x and d_y denote the normalized Euclidean distance from (22). As shown above, this will be implicitly equal to the cosine distance from (24) on centered and scaled data.

The goal of metabolomic feature matching is to learn the binary matching matrix M^* that aligns the distances between pairs of features in the most consistent way possible as shown in (21). To formalize this notion into a practical algorithm, we use the mathematical theory of optimal transport Peyré et al. (2019) which we discuss next.

Optimal transport

Optimal transport (OT) applies in the setting when the points $\{X_i\}_{i=1}^{p_1}$ and $\{Y_j\}_{j=1}^{p_2}$ and $\{Y_j\}_{j=1}^{p_2}$ being matched live in the same dimensional space $n_1 = n_2 = n$. It aims to find a matching between each point X_i and its corresponding point Y_j such that the sum of distances between matches is minimized. Matches between each pair of points can be stored in a matching matrix $M \in \{0, 1\}^{p_1 \times p_2}$ such that $M_{ij} = 1$ if X_i and Y_j are matched, and $M_{ij} = 0$ otherwise. Again we note that M must have at most one nonzero entry in each row and column to be a valid matching matrix.

Instead of searching over this space of binary matching matrices, optimal transport places masses $a_i \geq 0$ at all points X_i for $i = 1,\ldots,p$ 1 and masses $b_j \geq 0$ at all points Y_y for $j = 1, \ldots, p_2$ and optimizes over the space of probabilistic *couplings* $\Pi \in \mathbb{R}_+^{p_1 \times p_2}$ which move a n_{ij} amount of mass from X_i to Y_j. We assume here for simplicity that the sum of masses in both datasets are equal to one $\sum_{i=1}^{p_1} a_i = \sum_{i=1}^{p_2} b_i = 1$ and that the coupling $\prod$ transports all mass from a into b. More formally, optimal transport optimizes over the constrained set of couplings

$$\mathbf{U}(a, b) = \left\{\Pi \in \mathbb{R}_+^{p_1 \times p_2} : \Pi \mathbf{1}_{p_2} = a \text{ and } \Pi^T \mathbf{1}_{p_1} = b\right\} \tag{25}$$

where $\mathbf{1}_p$ denotes the all ones vector of length p. In practice, the points X_i and Y_j in each dataset are all treated the same and the masses placed on the data are chosen to be uniform $a = \frac{1}{p_1}\mathbf{1}_{p_1}$ and $b = \frac{1}{p_2}\mathbf{1}_{p_2}$.

The cost function which optimal transport minimizes is the sum of squared distances of its transported mass

$$\mathcal{E}(\Pi) = \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}d(X_i, Y_j) \tag{26}$$

where $d(u, v) = d^{\text{euc}}(u, v) = \frac{1}{\sqrt{n}}\|u - v\|$ is the Euclidean distance. The distance matrix $d(X_i, Y_j)$ in the OT objective can be replaced more generally with a cost matrix $C \in \mathbb{R}^{p_1 \times p_2}$ that is not necessarily a distance matrix. In this case the cost function becomes

$$\mathcal{E}(\Pi) = \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}C_{ij} \tag{27}$$

When the transport cost C_{ij} is a distance, the OT optimization defines a valid distance metric known as the *optimal transport distance* between discrete distributions $\{(a_i, X_i)\}_{i=1}^{p_1}$ and $\{(b_j, Y_j)\}_{j=1}^{p_2}$ in $\mathbb{R}^n$ given by

$$\text{OT}(a, b) = \min_{\Pi \in \mathbf{U}(a,b)} \mathcal{E}(\Pi). \tag{28}$$

When $d(u, v)$ is Euclidean, this OT distance is also referred to as the L^1 optimal transport distance, the Wasserstein 1-distance, or the Earth mover's distance. As formulated, the computation of the optimal transport objective involves an optimization over coupling matrices $\prod$ which can be solved by linear programming Peyré et al. (2019). The OT optimization problem becomes time consuming for problems with many points $p_1, p_2 \gg 1$. We show in the next section how augmenting this distance with a regularization term leads to a more efficient algorithm for learning the optimal coupling $\prod$.

Entropic regularization

Define the Kullback-Leibler (KL) divergence between two positive vectors $\mu, \nu \in \mathbb{R}_+^p$ as

$$D_{\text{KL}}(\mu, \nu) = \sum_{i=1}^{p}\mu_i \ln\left(\frac{\mu_i}{\nu_i}\right) - \sum_{i=1}^{p}\mu_i + \sum_{i=1}^{p}\nu_i \tag{29}$$

Given fixed marginals $a \in \mathbb{R}^{p_1}$ and $b \in \mathbb{R}^{p_2}$ from the previous section, we can define the entropy of a coupling matrix $\Pi \in \mathbb{R}_+^{p_1 \times p_2}$ with respect to these fixed marginals as

$$D_{\text{KL}}(\Pi, a \otimes b) = \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}\ln\left(\frac{\Pi_{ij}}{a_i b_j}\right) - \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij} + \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}a_i b_j. \tag{30}$$

where $(a \otimes b)_{IJ} = a_i b_j$ denotes the outer product. This can be further simplified as

$$\begin{aligned} D_{\text{KL}}(\Pi, a \otimes b) &= \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}\ln\left(\frac{\Pi_{ij}}{a_i b_j}\right) - \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij} + \left(\sum_{i=1}^{p_1}a_i\right)\left(\sum_{j=1}^{p_2}b_j\right) \\ &= \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}\ln\left(\frac{\Pi_{ij}}{a_i b_j}\right) \\ &= H(\Pi) + \ln(p_1) + \ln(p_2) \end{aligned} \tag{31}$$

where we define $H(\prod)$ by

$$H(\Pi) = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \Pi_{ij} \ln(\Pi_{ij}). \tag{32}$$

In the second line of the derivation above, we used the fact that the entries of a, b, and $\prod$ summed to one, and in the third line we used the fact that the marginals a and b were uniform. Under these assumptions, we see that the KL divergence $D_{\mathrm{KL}}(\prod, a \otimes b)$ is independent of the values of the marginals a, b and is equal to $H(\prod)$ up to constants.

Although here the general definition of entropy through the KL divergence reduces to the simpler formula of $H(\prod)$, in the following sections we will need to extend our analysis to cases when a, b, and $\prod$ have positive values that do not sum to one (i.e. not distributions). In this context, we will no longer have that $D_{\mathrm{KL}}(\prod, a \otimes b) = H(\prod) + const$ but we will still be able to use $D_{\mathrm{KL}}(\prod, a \otimes b)$ as a general notion of entropy for n.

The entropy of a coupling $D_{\mathrm{KL}}(n, a \otimes b)$ is an important notion because it quantifies how uniform or smooth $\prod$ is with respect to the product distribution $a \otimes b$. In particular, if a and b are set to uniform distributions as commonly done in practice, then $D_{\mathrm{KL}}(\prod, a \otimes b)$ is small when $\prod$ has close to uniform entries and is large otherwise. This notion of smoothness allows us to use $D_{\mathrm{KL}}(\prod, a \otimes b)$ as a regularizer in our optimal transport distance as

$$\mathcal{L}_{\varepsilon}(\Pi) = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \Pi_{ij} C_{ij} + \varepsilon D_{\mathrm{KL}}(\Pi, a \otimes b) \tag{33}$$

where ϵ is a small regularization parameter. Note that here we have denoted the transport cost matrix by $C \in \mathbb{R}^{p_1 \times p_2}$ which is not necessarily a distance matrix. The introduction of the regularizer $\epsilon D_{\mathrm{KL}}(\prod, a \otimes b)$ gives us an efficient iterative algorithm known as the *Sinkhorn algorithm* for optimizing $\prod$ which we describe in the following sections.

Unbalanced optimal transport

Before we introduce the Sinkhorn algorithm, we introduce a final modification to our optimal transport distance that allows us to learn couplings between distributions $a, b \in \mathbb{R}_+^p$ that do not preserve mass. In other words, the coupling $\prod$ is not required to perfectly satisfy the marginal constraints $\Pi \mathbf{1}_{p_2} = a$ and $\Pi^T \mathbf{1}_{p_1} = b$. In our metabolite matching problem, this is particularly useful as not all metabolites in one dataset necessarily appear in the other dataset and hence should be left unmatched. This modification of optimal transport, known as unbalanced optimal transport (UOT) Chizat et al. (2018), optimizes the following cost function

$$\mathcal{L}_{\rho,\varepsilon}(\Pi) = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \Pi_{ij} C_{ij} + \rho D_{\mathrm{KL}}(\Pi \mathbf{1}_{p_2}, a) + \rho D_{\mathrm{KL}}(\Pi^T \mathbf{1}_{p_1}, b) + \varepsilon D_{\mathrm{KL}}(\Pi, a \otimes b) \tag{34}$$

where we have added two KL terms with regularization parameter p to enforce that the marginals of the coupling $\Pi \mathbf{1}_{p_2}, \Pi^T \mathbf{1}_{p_1}$ are approximately close to the prescribed marginals a, b respectively. We have also kept the smoothness/entropy regularizer $\epsilon D_{\mathrm{KL}}(\prod, a \otimes b)$ from the previous section.

Unbalanced Sinkhorn algorithm

Now we are ready to present the unbalanced Sinkhorn algorithm Peyré etal. (2019) for optimizing the unbalanced optimal transport cost defined above. First we rewrite our optimization as

$$\min_{\Pi\in\mathbb{R}_+^{p_1\times p_2}} \mathcal{L}_{\rho,\varepsilon}(\Pi) = \min_{\Pi\in\mathbb{R}_+^{p_1\times p_2}} \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}C_{ij} + \rho D_{\mathrm{KL}}(\Pi\mathbf{1}_{p_2}, a) + \rho D_{\mathrm{KL}}(\Pi^T\mathbf{1}_{p_1}, b) + \varepsilon D_{\mathrm{KL}}(\Pi, a\otimes b)$$

$$= \min_{u\in\mathbb{R}_+^{p_1},\, v\in\mathbb{R}_+^{p_2}} \min_{\substack{\Pi\in\mathbb{R}_+^{p_1\times p_2}\\ \Pi\mathbf{1}_{p_2}=u,\ \Pi^T\mathbf{1}_{p_1}=v}} \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}C_{ij} + \rho D_{\mathrm{KL}}(u,a) + \rho D_{\mathrm{KL}}(v,b) + \varepsilon D_{\mathrm{KL}}(\Pi, a\otimes b).$$

The inner minimization can be solved exactly by introducing dual variables $f\in\mathbb{R}^{p_1}, g\in\mathbb{R}^{p_2}$ and writing out the Lagrange dual problem

$$\min_{\substack{\Pi\in\mathbb{R}_+^{p_1\times p_2}\\ \Pi\mathbf{1}_{p_2}=u,\ \Pi^T\mathbf{1}_{p_1}=v}} \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}C_{ij} + \varepsilon D_{\mathrm{KL}}(\Pi, a\otimes b) - f^T(\Pi\mathbf{1}_{p_2}-u) - g^T(\Pi^T\mathbf{1}_{p_1}-v) =$$

$$\max_{f\in\mathbb{R}^{p_1},\, g\in\mathbb{R}^{p_2}} \min_{\Pi\in\mathbb{R}_+^{p_1\times p_2}} \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}C_{ij} + \varepsilon D_{\mathrm{KL}}(\Pi, a\otimes b) - f^T(\Pi\mathbf{1}_{p_2}-u) - g^T(\Pi^T\mathbf{1}_{p_1}-v).$$

where we have removed the terms $\rho D_{\mathrm{kl}}(u, a)$ and $\rho D_{\mathrm{kl}}(v, b)$ since they do not depend on $\prod$. Taking the gradient in $\prod$ in the inner minimization and setting it to zero we get

$$C_{ij} + \varepsilon\log\left(\frac{\Pi_{ij}}{a_i b_j}\right) - f_i - g_j = 0$$

which implies that

$$\Pi_{ij} = a_i b_j \exp\left(\frac{f_i + g_j - C_{ij}}{\varepsilon}\right)$$

Now we can substitute this expression for $\prod$ back into our Lagrange dual problem. First we compute

$$\varepsilon D_{\mathrm{KL}}(\Pi, a\otimes b) = \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}\left(f_i + g_j - C_{ij}\right) - \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij} + \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2}a_i b_j$$

which implies that

$$\sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}C_{ij} + \varepsilon D_{\mathrm{KL}}(\Pi, a\otimes b) - f^T(\Pi\mathbf{1}_{p_2}-u) - g^T(\Pi^T\mathbf{1}_{p_1}-v)$$

$$= u^T f + v^T g - \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2}a_i b_j\exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right) + \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2}a_i b_j.$$

Hence, the outer maximization in our Lagrange dual problem for f and g can now be written as

$$\max_{f\in\mathbb{R}^{p_1},\, g\in\mathbb{R}^{p_2}} u^T f + v^T g - \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2}a_i b_j\exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right)$$

where we have removed the last constant sum in $a_i b_j$. Finally we can rewrite our entire minimization from the start of this section as

$$\begin{aligned}\min_{\Pi\in\mathbb{R}_+^{p_1\times p_2}} \mathcal{L}_{\rho,\varepsilon}(\Pi) &= \min_{u\in\mathbb{R}_+^{p_1},\, v\in\mathbb{R}_+^{p_2}} \min_{\substack{\Pi\in\mathbb{R}_+^{p_1\times p_2}\\ \Pi\mathbf{1}_{p_2}=u,\ \Pi^T\mathbf{1}_{p_1}=v}} \sum_{i=1}^{p_1}\sum_{j=1}^{p_2}\Pi_{ij}C_{ij} + \rho D_{\mathsf{KL}}(u,a) + \rho D_{\mathsf{KL}}(v,b) + \varepsilon D_{\mathsf{KL}}(\Pi, a\otimes b)\\ &= \min_{u\in\mathbb{R}_+^{p_1},\, v\in\mathbb{R}_+^{p_2}} \max_{f\in\mathbb{R}^{p_1},\, g\in\mathbb{R}^{p_2}} u^T f + v^T g - \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} a_i b_j \exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right) + \rho D_{\mathsf{KL}}(u,a) + \rho D_{\mathsf{KL}}(v,b).\end{aligned}$$

By strong duality, we can interchange the minimum and maximum above to write

$$\begin{aligned}\min_{\Pi\in\mathbb{R}_+^{p_1\times p_2}} \mathcal{L}_{\rho,\varepsilon}(\Pi) &= \max_{f\in\mathbb{R}^{p_1},\, g\in\mathbb{R}^{p_2}} \min_{u\in\mathbb{R}_+^{p_1},\, v\in\mathbb{R}_+^{p_2}} u^T f + v^T g - \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} a_i b_j \exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right) + \rho D_{\mathsf{KL}}(u,a) + \rho D_{\mathsf{KL}}(v,b)\\ &= U^*(f) + V^*(g) - \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} a_i b_j \exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right)\end{aligned}$$

where we define the functions

$$\begin{aligned}U^*(f) &= \min_{u\in\mathbb{R}_+^{p_1}} u^T f + \rho D_{\mathsf{KL}}(u,a)\\ V^*(g) &= \min_{v\in\mathbb{R}_+^{p_2}} v^T g + \rho D_{\mathsf{KL}}(v,b).\end{aligned} \tag{35}$$

In fact, we can solve the minimizations in U^* and V^* in closed form to get the minimizers $u^* = a \odot \exp(-f/\rho)$ and $v^* = b \odot \exp(-g/\rho)$ which we can substitute back in to get

$$\begin{aligned}U^*(f) &= \sum_{i=1}^{p_1} u_i^* f_i + \rho\sum_{i=1}^{p_1} u_i^* \ln\left(\frac{u_i^*}{a_i}\right) - \rho\sum_{i=1}^{p_1} u_i^* + \rho\sum_{i=1}^{p_1} a_i\\ &= \sum_{i=1}^{p_1} u_i^* f_i - \sum_{i=1}^{p_1} u_i^* f_i - \rho\sum_{i=1}^{p_1} a_i \exp(-f_i/\rho) + \rho\sum_{i=1}^{p_1} a_i\\ &= -\rho\sum_{i=1}^{p_1} a_i \exp(-f_i/\rho) + \rho\sum_{i=1}^{p_1} a_i.\end{aligned}$$

Likewise we can see that

$$V^*(f) = -\rho\sum_{i=1}^{p_2} b_i \exp(-g_i/\rho) + \rho\sum_{i=1}^{p_2} b_i.$$

Thus, we can rewrite our full optimization as

$$\begin{aligned}\min_{\Pi\in\mathbb{R}_+^{p_1\times p_2}} \mathcal{L}_{\rho,\varepsilon}(\Pi) &= \max_{f\in\mathbb{R}^{p_1},\, g\in\mathbb{R}^{p_2}} -\rho\sum_{i=1}^{p_1} a_i \exp(-f_i/\rho) - \rho\sum_{i=1}^{p_2} b_i \exp(-g_i/\rho) - \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} a_i b_j \exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right)\\ &= \min_{f\in\mathbb{R}^{p_1},\, g\in\mathbb{R}^{p_2}} \rho\sum_{i=1}^{p_1} a_i \exp(-f_i/\rho) + \rho\sum_{i=1}^{p_2} b_i \exp(-g_i/\rho) + \varepsilon\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} a_i b_j \exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right)\end{aligned}$$

where we have removed the terms independent of f and g.

Note that now we can optimize the cost function above by performing an alternating minimization on the dual variables f and g. Taking the gradient in f and setting it to zero we see that

$$-a_i \exp(-f_i/\rho) + a_i\sum_{j=1}^{p_2} b_j \exp\left(\frac{f_i+g_j-C_{ij}}{\varepsilon}\right) = 0$$

which implies that

$$f_i = -\frac{\varepsilon\rho}{\varepsilon+\rho}\ln\Big(\sum_{j=1}^{p_2} b_j \exp\Big(\frac{g_j - C_{ij}}{\varepsilon}\Big)\Big).$$

Similarly, we can write out

$$g_j = -\frac{\varepsilon\rho}{\varepsilon+\rho}\ln\Big(\sum_{i=1}^{p_1} a_i \exp\Big(\frac{f_i - C_{ij}}{\varepsilon}\Big)\Big).$$

We are now ready to write out the full unbalanced Sinkhorn algorithm which performs an alternating minimization on the dual potentials *f, g* as outlined above. We remind the reader that the coupling matrix can be recovered from the dual potentials by the formula

$$\Pi_{ij} = a_i b_j \exp\Big(\frac{f_i + g_j - C_{ij}}{\varepsilon}\Big).$$

The unbalanced Sinkhorn algorithm proceeds as follows.

The final output of the Sinkhorn algorithm optimization is a real-valued coupling matrix $\Pi \in \mathbb{R}_+^{p_1\times p_2}$. In some cases, it is desirable to transform the coupling matrix into a binary-valued matching matrix $M \in \{0,1\}^{p_1\times p_2}$ with possibly an added restriction that there is at most one nonzero element in each row and column (to obtain a valid partial matching). This can be done by either thresholding the real matrix $\prod$ or by assigning all maximal entries in each row (or column) to one and setting the remaining entries to zero. For our metabolomics matching problem, we describe our procedure for transforming our real-valued coupling into a binary matching matrix in the section on the GromovMatcher algorithm below.

Gromov-Wasserstein

Now that we have introduced the general formulation of unbalanced optimal transport and its corresponding Sinkhorn algorithm, we can extend this formulation to matching problems between distributions of points that live in different dimensional spaces. In our metabolomics setting, we aim to match two datasets of p_1 and p_2 metabolic features respectively where each feature in a dataset is associated with a feature intensity vector $\{X_i\}_{i=1}^{p_1} \subset \mathbb{R}^{n_1}$ and $\{Y_j\}_{j=1}^{p_2} \subset \mathbb{R}^{n_2}$ respectively across samples. We assume that there exists a true matching matrix $M^* \in \{0,1\}^{p_1\times p_2}$ with at most one nonzero entry in each row and column such that two metabolites (*i, j*) are matched if $M_{ij}^* = 1$.

We make the further assumption that if feature vectors X_i, Y_j are matched and feature vectors X_k, Y_l, are matched under M^*, then we expect that

$$d_x(X_i, X_k) \approx d_y(Y_j, Y_l) \quad \text{if} \quad \mathrm{M}^*_{\mathrm{ij}} = \mathrm{M}^*_{\mathrm{kl}} = 1. \tag{36}$$

where d_x is a distance metric on $\mathbb{R}^{n_1}$ and d_y is a distance metric $\mathbb{R}^{n_2}$. In practice, we always choose these distance metrics to be the normalized Euclidean distance defined for any $u,v \in \mathbb{R}^n$ as

$$d^{\mathrm{euc}}(u,v) = \frac{1}{\sqrt{n}}\|u - v\| \tag{37}$$

which is equal to the cosine distance $d^{\cos}$ (i.e. one minus the correlation) for centered and scaled data. Given these two distance matrices $D^x = [d_x(X_i, X_k)]_{i,k=1}^{p_1} \in \mathbb{R}^{p_1\times p_1}$ and $D^y = [d_y(Y_j, Y_l)]_{j,l=1}^{p_2} \in \mathbb{R}^{p_2\times p_2}$ we would like to infer the true matching matrix M^* by solving an optimization problem.

Algorithm 1: UnbalancedSinkhorn

input : Transport cost C, marginals a, b, marginal relaxation ρ, entropic regularization ε

output : Return the coupling matrix Π

Initialize $g = 0$

while (f, g) has not converged **do**

  Set $f_i \leftarrow -\frac{\varepsilon\rho}{\varepsilon+\rho} \ln\left(\sum_{j=1}^{p_2} b_j \exp\left(\frac{g_j - C_{ij}}{\varepsilon}\right)\right)$ for $i \in [p_1]$

  Set $g_j \leftarrow -\frac{\varepsilon\rho}{\varepsilon+\rho} \ln\left(\sum_{i=1}^{p_1} a_i \exp\left(\frac{f_i - C_{ij}}{\varepsilon}\right)\right)$ for $j \in [p_2]$

Return the coupling matrix $\Pi_{ij} = a_i b_j \exp\left(\frac{f_i + g_j - C_{ij}}{\varepsilon}\right)$ for $i \in [p_1]$ and $j \in [p_2]$

Algorithm 1

UnbalancedSinkhorn

Consider the following objective function

$$\mathcal{E}(M) = \sum_{i,k=1}^{p_1} \sum_{j,l=1}^{p_2} M_{ij} M_{kl} \left| D^x_{ik} - D^y_{jl} \right|. \tag{38}$$

where the matching matrices $M \in \{0,1\}^{p_1 \times p_2}$ we optimize over are constrained to satisfy marginal constraints $\prod \mathbf{1}_{p2} > 0$ and $\Pi^T \mathbf{1}_{p_1} > 0$. These marginal constraints simply impose that there is at least one nonzero entry in each row and column (i.e. each metabolite in both datasets has at least one corresponding match). Searching for the $\prod$ minimizing $\epsilon_{X,Y}(\prod)$ consists of putting the nonzero entries in $\prod$ such that the distance profiles of the matched features are similar, so that the minimizer of this criterion provides a good candidate estimate of $\prod^*$. This is closely related to the Gromov–Hausdorff distance Gromov (2001), an extension of optimal transport to the case where the sets to be coupled do not lie in the same metric space.

In practice, it is often desirable to optimize over a different set of matrices in order to make the optimization problem more tractable. Here we take intuition from optimal transport, and search over the set of coupling matrices with marginal constraints

$$\mathbf{U}(a,b) = \{\Pi \in \mathbb{R}_+^{p_1 \times p_2} : \Pi \mathbf{1}_{p_2} = a \text{ and } \Pi^T \mathbf{1}_{p_1} = b\}. \tag{39}$$

where as before, $a \in \mathbb{R}_+^{p_1}$ and $b \in \mathbb{R}_+^{p_2}$ are desired marginals which are typically set to be uniform distributions $a = \frac{1}{p_1}\mathbf{1}_{p_1}$ and $b = \frac{1}{p_2}\mathbf{1}_{p_2}$. These marginal vectors can be interpreted as distributions of masses a_i and b_j on the feature vectors X_i and Y_j respectively for $i \in [p_1]$, $j \in [p_2]$.

Coupling matrices in $\mathbf{U}(a, b)$ transport the distribution of masses a in the first dataset to the distribution of masses b in the second dataset. Now we can formulate the Gromov-Wasserstein (GW) distance, introduced by Memoli Mémoli (2011), as

$$\mathrm{GW}(a,b) = \min_{\Pi \in \mathbf{U}(a,b)} \mathcal{E}(\Pi) \tag{40}$$

By optimizing this objective, each entry $\prod_{ij}$ now reflects the strength of the matched pair (X_i, Y_j). Optimizing $\mathrm{GW}(a, b)$ then amounts to placing larger entries in $\prod$ whose paired features have similar distance profiles. Before we develop an algorithm to optimize this objective, we first modify it to allow for unbalanced matchings where marginal constraints are not enforced exactly (e.g. features in both datasets can remain unmatched).

Unbalanced Gromov–Wasserstein

In an untargeted context, all features measured in one study are not necessarily observed in another, either because these features are truly not shared or because of measurement error. However, the constraint $\prod \in \mathbf{U}(a, b)$ in the original GW optimization criterion (40) ensures that all the mass is transported from one set to another, resulting in all features being matched across studies. In order to discard study-specific features during the GW computation, we use the unbalanced Gromov–Wasserstein (UGW) distance with an additional entropic regularization for computational purposes, described in Séjourné et al. Sejourne et al. (2021). The optimization problem therefore reads

$$\begin{aligned} \mathcal{L}_{\rho,\varepsilon}(\Pi) = \mathcal{E}(\Pi) &+ \rho D_{\mathrm{KL}}\left(\Pi \mathbf{1}_{p_2} \otimes \Pi \mathbf{1}_{p_2}, a \otimes a\right) \\ &+ \rho D_{\mathrm{KL}}\left(\Pi^T \mathbf{1}_{p_1} \otimes \Pi^T \mathbf{1}_{p_1}, b \otimes b\right) \\ &+ \varepsilon D_{\mathrm{KL}}(\Pi \otimes \Pi, (a \otimes b)^{\otimes 2}) \end{aligned} \tag{41}$$

$$\mathrm{UGW}_{\rho,\varepsilon} = \min_{\Pi \in \mathbb{R}_+^{p_1 \times p_2}} \mathcal{L}_{\rho,\varepsilon}(\Pi) \tag{42}$$

with $\rho, \epsilon > 0$. Here D_{KL} is the Kullback–Leibler divergence defined in the previous sections and we define the tensor product $(P \otimes P)_{i,j,k,l} = P_{i,j}P_{kl}$. Here we set the desired marginal constraints to $a = \frac{1}{p_1}\mathbf{1}_{p_1}$ and $b = \frac{1}{p_2}\mathbf{1}_{p_2}$ as before.

As in the case of unbalanced optimal transport Chizat et al. (2018), the regularization ρ times the Kullback–Leibler divergences allows for the relaxation of the marginal constraints $\Pi\mathbf{1}_{p_2} = a$ and $\Pi^T\mathbf{1}_{p_1} = b$. The value of $\rho > 0$ controls the extent to which we allow for mass destruction. Smaller values of ρ tend to lessen the constraint on the marginals of $\prod$, while balanced GW is recovered when $\rho \to +\infty$. As proposed in the original paper Séjourne et al. (2021), our UGW cost modifies the UOT formulation by using the quadratic Kullback-Leibler divergence in $\Pi\mathbf{1}_{p_2}\otimes\Pi\mathbf{1}_{p_2}$ and $\Pi^T\mathbf{1}_{p_1}\otimes\Pi^T\mathbf{1}_{p_1}$ instead, hence preserving the quadratic form of the GW cost function $\varepsilon(\prod)$.

The term $\epsilon D_{KL}(\Pi\otimes\Pi,(a\otimes b)^{\otimes 2})$ serves as an entropic regularization, inspired again by optimal transport. Adding such a penalty is a standard way to compute an approximate solution to the optimal transport problem using the Sinkhorn algorithm as we shall show in the following section. Here again, we modify the entropic penalty in UGW to have a quadratic form in $\prod \otimes \prod$ to agree with the quadratic form of the GW cost $\epsilon(\prod)$. The parameter e controls the smoothness (entropy) of the coupling matrix $\prod$ where larger values of ϵ encourage $\prod$ to put uniform weights on many of its entries, leading to less precision in the feature matches. However, increasing ϵ also leads to better numerical stability and a significant speedup of the alternating Sinkhorn algorithm used to optimize the objective function described below.

UGW optimization algorithm

Now we are ready to write out an algorithm to optimize the UGW objective in (42). First write our objective as

$$\begin{aligned}\mathcal{L}_{\rho,\varepsilon}(\Pi) = \sum_{i,k=1}^{p_1}\sum_{j,l=1}^{p_2}\Pi_{ij}\Pi_{kl}\left|D^x_{ik} - D^y_{jl}\right| &+ \rho D_{\mathrm{KL}}\left(\Pi\mathbf{1}_{p_2}\otimes\Pi\mathbf{1}_{p_2}, a\otimes a\right)\\ &+ \rho D_{\mathrm{KL}}\left(\Pi^T\mathbf{1}_{p_1}\otimes\Pi^T\mathbf{1}_{p_1}, b\otimes b\right)\\ &+ \varepsilon D_{\mathrm{KL}}(\Pi\otimes\Pi,(a\otimes b)^{\otimes 2}).\end{aligned} \tag{43}$$

Using the quadratic nature of our cost function, we aim to perform an alternating minimization in the two copies of $\prod$. For the moment, let's differentiate these two copies by $\prod$ and Γ and write the new cost

$$\begin{aligned}\mathcal{F}_{\rho,\varepsilon}(\Pi,\Gamma) = \sum_{i,k=1}^{p_1}\sum_{j,l=1}^{p_2}\Pi_{ij}\Gamma_{kl}\left|D^x_{ik} - D^y_{jl}\right| &+ \rho D_{\mathrm{KL}}(\Pi\mathbf{1}_{p_2}\otimes\Gamma\mathbf{1}_{p_2}, a\otimes a)\\ &+ \rho D_{\mathrm{KL}}(\Pi^T\mathbf{1}_{p_1}\otimes\Gamma^T\mathbf{1}_{p_1}, b\otimes b)\\ &+ \varepsilon D_{\mathrm{KL}}(\Pi\otimes\Gamma,(a\otimes b)^{\otimes 2}).\end{aligned} \tag{44}$$

Before we expand this cost, we introduce the notation $m(\pi)$ to denote the sum of the elements of π which can be a vector, matrix or tensor. In general, for four positive distributions $\pi, a \in \mathbb{R}_+^p$ and $\gamma, b \in \mathbb{R}_+^q$ we have that the KL satisfies the tensorization property

$$\begin{aligned}D_{\mathrm{KL}}(\pi\otimes\gamma, a\otimes b) &= \sum_{i=1}^{p}\sum_{j=1}^{q}\pi_i\gamma_j\ln\left(\frac{\pi_i\gamma_j}{a_ib_j}\right) - \sum_{i=1}^{p}\sum_{j=1}^{q}\pi_i\gamma_j + \sum_{i=1}^{p}\sum_{j=1}^{q}a_ib_j\\ &= m(\gamma)\sum_{i=1}^{p}\pi_i\ln\left(\frac{\pi_i}{a_i}\right) + m(\pi)\sum_{j=1}^{q}\gamma_j\ln\left(\frac{\gamma_j}{b_j}\right) - m(\pi)m(\gamma) + m(a)m(b)\\ &= m(\gamma)D_{\mathrm{KL}}(\pi,a) + m(\pi)D_{\mathrm{KL}}(\gamma,b) + \Big(m(\pi) - m(a)\Big)\Big(m(\gamma) - m(b)\Big).\end{aligned} \tag{45}$$

Specifically, if we remove those terms that do not depend on y we are left with

$$D_{\mathrm{KL}}(\pi \otimes \gamma, a \otimes b) = m(\pi) D_{\mathrm{KL}}(\gamma, b) + m(\gamma) \sum_{i=1}^{p} \pi_i \ln\left(\frac{\pi_i}{a_i}\right) + const. \tag{46}$$

This allows us to write for the marginal constraints $a \in \mathbb{R}_+^{p_1}, b \in \mathbb{R}_+^{p_2}$ and couplings $\Pi, \Gamma \in \mathbb{R}_+^{p_1 \times p_2}$ that

$$D_{\mathrm{KL}}(\Pi \mathbf{1}_{p_2} \otimes \Gamma \mathbf{1}_{p_2}, a \otimes a) = m(\Pi) D_{\mathrm{KL}}(\Gamma \mathbf{1}_{p_2}, a) + m(\Gamma) \sum_{i=1}^{p_1} (\Pi \mathbf{1}_{p_2})_i \ln\left(\frac{(\Pi \mathbf{1}_{p_2})_i}{a_i}\right) + const.$$

$$D_{\mathrm{KL}}(\Pi^T \mathbf{1}_{p_1} \otimes \Gamma^T \mathbf{1}_{p_1}, b \otimes b) = m(\Pi) D_{\mathrm{KL}}(\Gamma^T \mathbf{1}_{p_1}, b) + m(\Gamma) \sum_{j=1}^{p_2} (\Pi^T \mathbf{1}_{p_1})_j \ln\left(\frac{(\Pi^T \mathbf{1}_{p_1})_j}{b_j}\right) + const.$$

$$D_{\mathrm{KL}}(\Pi \otimes \Gamma, (a \otimes b)^{\otimes 2}) = m(\Pi) D_{\mathrm{KL}}(\Gamma, a \otimes b) + m(\Gamma) \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \Pi_{ij} \ln\left(\frac{\Pi_{ij}}{a_i b_j}\right) + const.$$

where in the expansions above we have removed all terms that are independent of Γ. Finally, expanding out $\mathcal{F}_{\rho,\epsilon}$ (∏,Γ) and keeping only those terms that depend on Γ we get

$$\mathcal{F}_{\rho,\varepsilon}(\Pi, \Gamma) = \sum_{k=1}^{p_1} \sum_{l=1}^{p_2} \Gamma_{kl} C_{kl}^{\Pi} + \rho m(\Pi) D_{\mathrm{KL}}(\Gamma \mathbf{1}_{p_2}, a) + \rho m(\Pi) D_{\mathrm{KL}}(\Gamma^T \mathbf{1}_{p_1}, b) + \varepsilon m(\Pi) D_{\mathrm{KL}}(\Gamma, a \otimes b) \tag{47}$$

where the cost matrix $C^{\Pi} \in \mathbb{R}^{p_1 \times p_2}$ is defined as

$$C_{kl}^{\Pi} = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \Pi_{ij} \left| D_{ik}^x - D_{jl}^y \right| + \rho \sum_{i=1}^{p_1} (\Pi \mathbf{1}_{p_2})_i \ln\left(\frac{(\Pi \mathbf{1}_{p_2})_i}{a_i}\right) + \rho \sum_{j=1}^{p_2} (\Pi^T \mathbf{1}_{p_1})_j \ln\left(\frac{(\Pi^T \mathbf{1}_{p_1})_j}{b_j}\right) + \varepsilon \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \Pi_{ij} \ln\left(\frac{\Pi_{ij}}{a_i b_j}\right). \tag{48}$$

where we have hidden the dependence of C^{Π} on the distance matrices D^x,D^y, the marginals a, b, and the regularization parameters ρ, ϵ for ease of notation.

Remarkably, the cost above in Γ for fixed ∏ is in the form of an unbalanced optimal transport problem which can be solved through unbalanced Sinkhorn iterations (Algorithm 1). Note that in our derivation above, it did not matter whether we optimized Γ with ∏ fixed or vice versa because the cost $\mathcal{F}_{\rho,\epsilon}$(∏,Γ) is symmetric in both of its arguments.

Our iterative algorithm for solving the unbalanced GW problem will proceed at each iteration by optimizing Γ to minimize the cost above using the unbalanced Sinkhorn method, setting ∏ equal to Γ and repeating. With each iteration, we expect this iterative procedure to make smaller and smaller updates to Γ until convergence. By definition, at the end of each iteration we assign ∏ = Γ so the minimizer of $\mathcal{F}_{\rho,\epsilon}$(∏,Γ) we converge to should also be a minimizer of the original UGW cost $\mathcal{L}_{\rho\varepsilon}$(∏) in the sense that the relaxation of $\mathcal{L}_{\rho,\epsilon}$(∏) to $\mathcal{F}_{\rho,\epsilon}$(∏,Γ) is tight. This is proven rigorously under strict mathematical assumptions in Sejourne et al. (2021). We state the full UGW optimization algorithm below.

Following the implementation of the UGW algorithm in Sejourne et al. (2021), we initialize both ∏ and Γ to be the product distribution of the marginals $a \otimes b / \sqrt{m(a) m(b)}$ before we begin the optimization. Also, we note that if (∏,Γ) is a minimizer of our UGW objective $\mathcal{F}_{\rho,\epsilon}$(∏,Γ), then so is $(\frac{1}{s}\Pi, s\Gamma)$ for any scale factor $s > 0$. Hence, we can set $m(\frac{1}{s}\Pi) = m(s\Gamma)$ by choosing $s = \sqrt{m(\Pi)/m(\Gamma)}$. This motivates the final step in the while loop of the UGW algorithm where the rescaling of Γ by the factor $\sqrt{m(\Pi)/m(\Gamma)}$ leads to mass equality m(∏) = m(Γ) and also stabilizes the convergence of the algorithm.

Algorithm 2: UnbalancedGromovWasserstein

input : Distance matrices D^x, D^y, marginals a, b, marginal relaxation ρ, entropic regularization ε

output : Return the coupling matrix Π

Initialize $\Pi = \Gamma = a \otimes b / \sqrt{m(a)m(b)}$

while (Π, Γ) has not converged **do**

  Update $\Pi \leftarrow \Gamma$

  Update $\Gamma =$ UnbalancedSinkhorn$(C^{\Pi}, a, b, \rho m(\Pi), \varepsilon m(\Pi))$

  Rescale $\Gamma \leftarrow \sqrt{m(\Pi)/m(\Gamma)}\Gamma$

Update $\Pi \leftarrow \Gamma$

Return Π.

Algorithm 2

UnbalancedGromovWasserstein

Returning to our metabolomics matching problem, we further guide our UGW optimization procedure by discouraging it from matching metabolic feature pairs whose mass-to-charge ratios are incompatible. Namely, we choose a value m_{gap} such that for all pairs (i,j) with $i \in [p_1]$, $j \in [p_2]$ and mass-to-charge ratios m_i^x, m_j^y we enforce that

$$|m_i^x - m_j^y| > m_{\text{gap}} \implies \Pi_{ij} = 0. \tag{49}$$

In practice, this is done by taking the optimal transport cost C^{Π} in every iteration of the UGW algorithm and premultiplying it elementwise by a factor $W \in \mathbb{R}_+^{p_1 \times p_2}$ given by

$$C^{\Pi} \to W \odot C^{\Pi}, \quad W_{ij} = 99 \cdot \mathbf{1}_{\{|m_i^x - m_j^y| > m_{\text{gap}}\}} + 1 \tag{50}$$

where $\mathbf{1}_\chi$ denotes the indicator function that is one when the condition χ is satisfied and zero otherwise. Such a prefactor changes the transport cost to be very large for feature matches with incompatible mass-to-charge ratio times, and hence, the entries of $\prod$ set small weights at these entries. Our weighted UGW algorithm is rewritten below.

As mentioned before, the coupling matrix returned by our weighted UGW algorithm is a realvalued matrix rather than a binary matching matrix. In the next section, we describe how we incorporate metabolite retention time information to filter out unlikely pairs in our coupling matrix and transform it into a valid one-to-one matching of features across two datasets.

Retention time drift estimation and filtering

To filter out unlikely matches from the coupling matrix returned by Algorithm 3 above, we use the retention times (RTs) of the metabolites in both datasets. We remind the reader that RTs were not incorporated into the weighted UGW algorithm since they often exhibit a non-linear deviation between datasets, and hence are not directly comparable. However, using the metabolite coupling $\tilde{\Pi} \in \mathbb{R}_+^{p_1 \times p_2}$ obtained from Algorithm 3, it is possible to estimate this RT drift. The estimated RT drift $\hat{f} : \mathbb{R}_+ \to \mathbb{R}_+$ allows us to assess the plausibility of the pairs recovered by the restricted UGW coupling $\tilde{\Pi}$, and discard pairs incompatible with the estimated drift.

We propose to learn the drift $\hat{f}$ through the weighted spline regression

$$\min_{f \in \mathcal{B}_{n,k}} \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \tilde{\Pi}_{ij} \left| f\left(RT_i^x\right) - RT_j^y \right| \tag{51}$$

where $\mathcal{B}_{n,k}$ is the set of n-order B-splines with k knots. All pairs (RT_i^x, RT_j^y) in objective (51) are weighted by the coefficients of $\tilde{\Pi}$ so that larger weights are given to pairs identified with high confidence in the first step of our procedure.

Pairs identified as incompatible with the estimated RT drift are then discarded from the coupling matrix. To do this, we first take the estimated RT drift $\hat{f}$, and the set of pairs $S = \{i,j : \tilde{\Pi}_{i,j} \neq 0\}$ recovered in $\tilde{\Pi}$ with nonzero entries. We then define the residual associated with $(i, j) \in \mathcal{S}$ as

$$r_{\hat{f}}(i,j) = |\hat{f}(RT_i^x) - RT_j^y|. \tag{52}$$

The 95% prediction interval and the median absolute deviation (MAD) of these residuals are given by

$$\begin{aligned} \text{PI} &= 1.96 \times \text{std}(\{r_{\hat{f}}(i,j), (i,j) \in \mathcal{S}\}) \\ \text{MAD} &= \text{median}(\{|r_{\hat{f}}(i,j) - \mu_r|, (i,j) \in \mathcal{S}\}) \\ \mu_r &= \text{median}(\{|r_{\hat{f}}(i,j)|, (i,j) \in \mathcal{S}\}) \end{aligned} \tag{53}$$

Algorithm 3: WeightedUnbalancedGromovWasserstein

input : Distance matrices D^x, D^y, marginals a, b, marginal relaxation ρ, entropic regularization ε,
mass-to-charge ratios m^x, m^y, mass-to-charge ratio gap m_{gap}

output : Return the coupling matrix Π

Initialize $\Pi = \Gamma = a \otimes b / \sqrt{m(a)m(b)}$

Set $W_{ij} = 99 \cdot \mathbf{1}_{\{|m_i^x - m_j^y| > m_{\text{gap}}\}} + 1$ for $i \in [p_1]$ and $j \in [p_2]$

while (Π, Γ) has not converged **do**

  Update $\Pi \leftarrow \Gamma$

  Update $\Gamma = \text{UnbalancedSinkhorn}(W \odot C^{\Pi}, a, b, \rho m(\Pi), \varepsilon m(\Pi))$

  Rescale $\Gamma \leftarrow \sqrt{m(\Pi)/m(\Gamma)}\Gamma$

Update $\Pi \leftarrow \Gamma$

Return Π.

Algorithm 3

WeightedUnbalancedGromovWasserstein

where $|\mathcal{S}|$ is the size of $\mathcal{S}$and the functions std, median denote the standard deviation and median respectively. Following Habra etal. (2021), we then create a new filtered coupling matrix $\hat{\Pi} \in \mathbb{R}_+^{p_1 \times p_1}$ given by

$$\hat{\Pi}_{ij} = \begin{cases} \tilde{\Pi}_{ij} & \text{if } r_{\hat{f}}(i,j) < \mu_r + r_{\text{thresh}} \\ 0 & \text{otherwise} \end{cases}. \tag{54}$$

where r_{thresh} is a given filtering threshold. The procedure of estimating the drift function $\hat{f}$ in (51) and filtering the coupling can be repeated for multiple iterations, to improve the drift and coupling estimation. In our main algorithm, we use two preliminary iterations where we estimate the RT drift and discard outliers with r_{thresh} = PI, defined as points falling outside of the 95% prediction interval. We the re-estimate the drift and perform a final filtering step with the more stringent MAD by setting r_{thresh} = 2 × MAD.

At this stage, it is possible for $\hat{\Pi}$ to still contain coefficients of very small magnitude. As an optional postprocessing step, we discard these coefficients by setting all entries smaller than $\tau \max(\hat{\Pi})$ to zero for some scaling constant $\tau \in [0, 1]$. Lastly, a feature from either study could have multiple possible matches, since $\hat{\Pi}$ can have more than one non-zero coefficient per row or column. Although reporting multiple matches can be helpful in an exploratory context, for the sake of simplicity in our analysis, the final output of GromovMatcher returns a one-to-one matching. Consequently, we only keep those metabolite pairs (i, j) where the entry $\hat{\Pi}_{ij}$ is largest in its corresponding row and column. All nonzero entries of $\hat{\Pi}$ which do not satisfy this criterion are set to zero. Finally, we convert $\hat{\Pi}$ into a binary matching matrix $M \in \{0, 1\}^{p_1 \times p_2}$ with ones in place of its nonzero entries and this final output is returned to the user.

As a naming convention, we use the abbreviation GM for our GromovMatcher method, and use the abbreviation GMT when running GromovMatcher with the optional τ -thresholding step.

GromovMatcher algorithm summary

In summary, our full GromovMatcher algorithm consists of (1) UGW optimization followed by (2) retention time drift estimation and filtering.

The tuning of ρ and ϵ was computationally driven and the two parameters were set as low as possible, with ρ = 0.05 and ϵ = 0.005. Based on literature Loftfield et al. (2021); Hsu et al. (2019); Climaco Pinto et al. (2022); Habra et al. (2021); Chen et al. (2021) and what is considered to be a plausible variation of a feature's m/z, we set m_{gap} = 0.01ppm. For RT drift estimation, the order of the B-splines was set to n = 3 by default, while the number of knots k was selected by 10-fold cross-validation. If the optional thresholding step was applied in GMT, we set τ = 0.3. Otherwise, we let τ = 0 which gives the unthresholded GM algorithm.

Appendix 2

Here we discuss existing metabolomic alignments methods and the hyperparameter experiments we perform on these methods. We consider two existing alignment methods for comparison, metabCombiner Habra et al. (2021) and M2S Climaco Pinto et al. (2022). Both of them take the same kind of input as GromovMatcher, i.e. feature tables with features identified with their m/z, RT, and intensities across samples.

Algorithm 4: GromovMatcher

input : Distance matrices D^x, D^y, marginals a, b, marginal relaxation ρ, entropic regularization ε,
mass-to-charge ratios m^x, m^y, mass-to-charge ratio gap m_{gap},
retention times RT^x, RT^y, B-spline order n, filtering threshold τ

output : Return the matching matrix M and the retention time drift $\hat{f}$

Step 1: Weighted UGW optimization
Compute $\tilde{\Pi} = \text{WeightedUnbalancedGromovWasserstein}(D^x, D^y, a, b, \rho, \varepsilon, m^x, m^y)$

# Step 2: Retention time drift estimation and filtering
for $i = 1 : 3$ **do**
  Perform weighted spline regression (51) for RT drift $\hat{f} \in \mathcal{B}_{n,k}$ where k is chosen by 10-fold cross validation
  Initialize $r_{\text{thresh}} = 0$
  if $i < 3$ **then**
    Set $r_{\text{thresh}} = \text{PI}$ from (53)
  else
    Set $r_{\text{thresh}} = 2 \times \text{MAD}$ from (53)
  Set $\tilde{\Pi} = \hat{\Pi}$

Compute $\mathcal{U} = \max(\hat{\Pi})$
Set $\hat{\Pi}_{ij} = 0$ if $\hat{\Pi}_{ij} < \tau\mathcal{U}$ for $i \in [p_1]$ and $j \in [p_2]$
Set $\hat{\Pi}_{ij} = 0$ if $i \neq \text{argmax}_k \hat{\Pi}_{k,j}$ or $j \neq \text{argmax}_k \hat{\Pi}_{i,k}$ for $i \in [p_1]$ and $j \in [p_2]$
Define the binarized matching $M_{ij} = \mathbf{1}_{\{\hat{\Pi}_{ij} > 0\}}$

Return M and $\hat{f}$.

Algorithm 4

GromovMatcher

MetabCombiner hyperparameter experiments

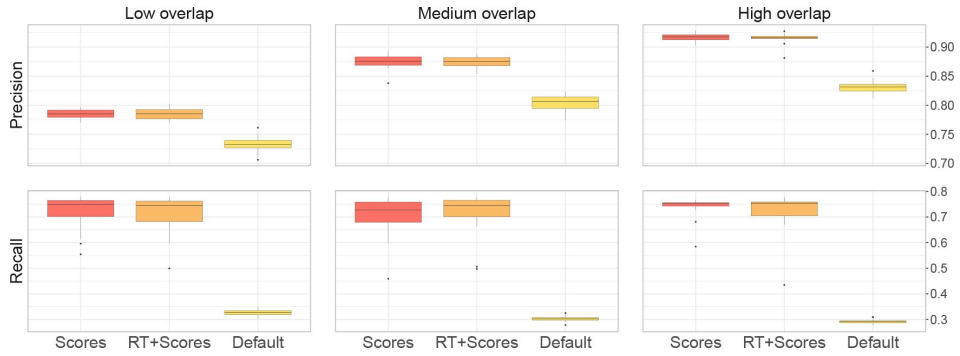
MetabCombiner Habra et al. (2021) is a three-step process that begins by grouping features based on their m/z within user-specified bins. This creates a search space for potential feature pairs. In the second step, MetabCombiner estimates the RT drift using the potential feature pairs identified in the first step, and eliminates outlying pairs over several iterations. This step can incorporate prior knowledge by identifying shared features and marking them as anchors, which are not discarded. In the final step, MetabCombiner scores the remaining feature pairs based on their m/z, RT, and relative intensity compatibility to discriminate between multiple matches for one feature. The scoring system relies on weights assigned to m/z, RT, and feature intensities, with the magnitude of those weights reflecting the reliability of the corresponding measurements across studies.

MetabCombiner Habra et al. (2021) includes adjustable parameters throughout the pipeline. We set most of them to default values unless otherwise stated. MetabCombiner first establishes candidate pairs by binning features in the m/z dimension with a width of binGap, and pairing the features sorted by relative intensities. The'binGap'parameter sets the m/z tolerance of metabCombiner, similar to m_{gap} in GromovMatcher. We used the same value of 0.01 as in GromovMatcher.

MetabCombiner then estimates the RT drift using basis splines, and removes pairs associated with a high residual (twice the mean model error) from the candidate set.

In our main experiment, the RT drift is estimated exclusively using candidate pairs selected by the pipeline. However, it is also possible to include known ground truth pairs as 'anchors' to estimate the RT drift. We choose not to rely on prior knowledge for drift estimation as Habra et al. Habra et al. (2021) show their drift estimation to be efficient and robust, even without prior knowledge. To confirm this claim, we conduct a sensitivity analysis comparing the results obtained in our main experiment with those obtained when supplying metabCombiner with known shared metabolites to anchor the RT drift estimation. We randomly select 100 anchors from the ground truth matching and compute the metabCombiner matchings with otherwise identical settings as in our main experiment. The results from this analysis (reported in **Appendix 2 - Figure 1**) show that the unsupervised RT drift estimation (using anchors selected by the pipeline only) performs as well as the supervised RT drift estimation, showing the drift estimation to be very consistent, with or without shared entities.

After establishing candidate pairs and filtering out those that contradict the estimated RT drift, metabCombiner discriminates between multiple matches using a scoring system that considers m/z, RT, and rankings of the median feature intensities. Each dimension has a specific weight that can be left at default, manually adjusted, or automatically tuned using known matched pairs. Habra et al. Habra et al. (2021) provide qualitative guidelines for tuning the weights manually, mainly based on the experimental conditions and visual inspection of the RT drift plot. Since this approach is difficult to implement in the various settings we consider for our simulation study, we rely on the quantitative tuning function included in the metabCombiner pipeline. This function takes into account known shared features and tunes the weights to optimize the scores of those known matches. We randomly select 100 known true matches to define the objective function metabCombiner maximizes. We search over the recommended range of values, with the m/z weight $A \in [50, 150]$, the RT weight $B \in [5, 20]$ and the feature intensities weight $C \in [0, 1]$. **Appendix 2 - Figure 1** presents the results obtained with the weights set at default values ($A = 100$, $B = 15$, $C = 0.5$), as a sensitivity analysis.



Appendix 2—figure 1.

Performance of metabCombiner with the different parameter settings.

The first setting, labelled ‘Scores’ correspond to the design of our main analysis, where 100 randomly selected true pairs are supplied to metabCombiner to set the scoring weights automatically, but are not otherwise used. In the second setting, labelled ‘Scores + RT’, metabCombiner is allowed to use the 100 true pairs not only to set the scoring weights, but also to estimate the RT drift. Finally, in the third ‘Default’ setting, we do not use any prior knowledge for the RT drift estimation and keep the scoring weights’ default values.

M2S hyperparameter experiments

Pinto et al. Climaco Pinto et al. (2022) introduce M2S as a more versatile alternative to metabCombiner, while still adhering to most of its core principles. Like metabCombiner, M2S follows a three-step process. First it searches for matches within user-defined thresholds for *m/z*, retention time, and mean feature intensity. Next, M2S estimates *m/z*, RT and feature intensity drifts between datasets and removes any outlier pairs. Finally, M2S selects the best match using a scoring system that weighs each measurement, similar to metabCombiner. M2S notably stands out by providing greater flexibility in the methods and measurements used at each step of the procedure, resulting however in a larger number of parameters that require manual fine-tuning. To address this, we adopt two different approaches for the simulation study and the EPIC study alignment. In the simulation study, we set the initial thresholds to oracle values and investigate technical parameters. For the EPIC study alignment, we use the combination of technical parameters with the best average F1-score in the simulation study and select the best threshold values based on the performance on the validation subset.

More precisely, M2S first matches all pairs of metabolic features whose absolute difference in *m/z*, RT, and median of log_{10} FI are within the user-defined thresholds 'MZ_intercept', 'RT_intercept' and 'log10FI_intercept'. On simulated data experiments, we set these thresholds to MZ_intercept = 0.01, RT_intercept = 3.5 and log10FI_intercept = 0.2 which are large enough to not exclude any true feature matches in any of the scenarios for our simulated data under low, medium, and high overlap/noise (see Methods). M2S also offers more detailed options to match features whose absolute difference stays within two lower and upper bound lines with a given slope where the intercepts of these lines are defined using the values above. In our analysis, we set the slopes of these linear boundaries to zero so as to not remove any true matches. Because the reference and target studies we are matching in the simulated analysis are on the same scale, we set the FI adjust method to 'none'.

The second step of M2S involves calculating penalization scores for every pair of matches which are used to determine the best set of matches between metabolic features of both datasets. This step depends on a set of hyperparameters which we perform a grid search over to optimize the performance of M2S. For estimating the *m/z*, RT, and FI drift, the hyperparameters are the percentage of neighbors 'nrNeighbors', the neighborhood shape 'neighMethod', and the LOESS span percentage 'pctPointsLoess' used to smooth the estimated drift functions. After the drifts are estimated, they are normalized using a method specified by 'residPercentile' that puts the *m/z*, RT, and FI residuals on the same scale. We always fix residPercentile = NaN which defaults to the standard 2 × MAD normalization. Next, for every remaining metabolic feature match, the residuals/drifts of the *m*/*z*, RT, and FI are added together by taking the weighted square root sum of squares. For unnormalized data where feature intensity magnitudes are important, we weight all three drifts equally using $W = (1, 1, 1)$ and for data with normalized feature intensities we set the FI drift weight to zero such that $W = (1, 1, 0)$. Finally, using these weighted penalization scores, M2S selects the best matched pair within a multiple match cluster to obtain a one-to-one matching between datasets.

The third and final step of M2S involves removing those remaining matches which have large differences in *m*/*z*, RT, or FI. This can be performed using several methods indicated by the hyperparameter 'methodType'. Each method excludes those matched pairs whose differences in *m*/*z*, RT, or FI exceed a certain number of median absolute deviations indicated by the parameter 'nr-MAD'. The remaining one-to-one metabolic feature matches are returned as the final result of the M2S algorithm.

Metric	Low overlap	Medium overlap	High overlap
Precision	0.831	0.917	0.947
Recall	0.934	0.933	0.939

Appendix 2—table 1.

Performance of M2S in a setting where the RT drift between studies is linear.

To optimally tune M2S on our simulated experiments, we determine the optimal M2S parameter combination for each individual simulation setting (low, medium, high overlap and noise) by performing a grid search over the product of parameter lists

- nrNeighbors = [0.01, 0.05, 0.1, 0.5, 1]
- neighMethod = ['cross', 'circle']
- pctPointsLoess = [0, 0.1, 0.5]
- methodType = ['none', 'scores', 'byBins', 'trend_mad', 'residuals_mad']
- nrMAD = [1, 3, 5]

Each parameter combination for M2S is tested across 20 randomly generated datasets at the same overlap and noise settings. For each setting, the combination of parameters above with the best average F1-score across these 20 trials is used as the optimal parameter choice.

M2S applies initial RT thresholds to search for candidate pairs, which may favor settings where the RT drift follows a linear trend. Therefore, as a sensitivity analysis, we apply M2S to simulated data with a linear drift. The simulation process is identical to that of our main simulation study, except for the deviation of the RT in dataset 2. Specifically, for a given overlap value, we divide the original real-world dataset into two smaller datasets and introduce random noise to the *m*/*z*, RT and intensities of the features, without introducing a systematic deviation to the RT in dataset 2. M2S parameters are kept identical to the ones used in our main analysis in comparable settings. The results obtained by M2S on three pairs of datasets generated for three overlap values (0.25, 0.5 and 0.75) and a medium noise level are reported in Appendix 2 - Table 1. While the results obtained in a high overlap setting are close to those obtained in our main analysis M2S demonstrates better performance in a low overlap setting when the RT drift is linear than in our main analysis. This observation is consistent with the results obtained by M2S on EPIC data, considering the relatively low estimated overlap between the aligned EPIC studies in our main analysis.

For the EPIC data, we select the parameter combination that yields the highest F1-score across all simulated settings. However, due to the unavailability of oracle values for setting initial thresholds, we perform a search over several MZ intercept values (0.01, 0.05, and 0.1), RT intercept values (0.1, 0.5, 1, and 5), and logFI intercept values (1, 10, and 100).

Appendix 3

In this section, we study the sensitivity of all three alignment methods GMT, M2S, and mC to the validation dataset split when creating two validation studies for matching. As described in the section "Validation on ground-truth data" and depicted in **Fig. 2** of the main text, we generate two datasets to be matched by splitting an initial LC-MS dataset with p features and n samples into two smaller overlapping datasets. The first dataset has p_1 features and n_1 samples while the second dataset has p_2 features and n_2 samples. The sets of samples in both datasets are disjoint such that $n_1 + n_2 = n$. However, the dataset split is constructed such that both datasets share $\approx \lambda p$ of their features where $\lambda \in [0,1]$ is an overlap fraction. Namely, this is done by defining the dataset feature sizes as

$$p_1 = \left\lfloor \left(\lambda + \lambda_f(1-\lambda)\right)p \right\rfloor, \quad p_2 = \left\lfloor \left(\lambda + (1-\lambda_f)(1-\lambda)\right)p \right\rfloor \tag{55}$$

and the dataset sample sizes as

$$n_1 = \lfloor \lambda_s n \rfloor, \quad n_2 = n - \lfloor \lambda_s n \rfloor. \tag{56}$$

As before, $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ denote integer floor and ceiling functions. Then taking the original LC-MS dataset and randomly permuting its samples and features, the first p_1 features and first n_1 samples are placed into dataset 1 while the last p_2 features and last n_2 samples are placed into dataset 2. It is indeed easy to check here that with such a splitting procedure, the feature overlap between both datasets is $p_1 + p_2 - p \approx \lambda p$.

Here $\lambda_f \in [0.5, 1]$ controls the fraction of features in dataset 1 that is not shared with dataset 2 and $\lambda_s \in (0, 1)$ controls the fraction of samples in dataset 1 vs. dataset 2. In particular, if $\lambda_f = 1$ then the features in dataset 2 are entirely a subset of those in dataset 1. In the experiments described in the main text, we always set $\lambda_f = \lambda_s = 0.5$ as to balance the number of features and samples in both resulting datasets.

Now we study how the performance of all three alignment methods changes when λ_f, λ_s and the feature overlap λ are varied. Here we vary the feature overlap $\lambda \in \{0.25, 0.5, 0.75\}$, the feature fraction $\lambda_s \in \{0.5,0.6, 0.7, 0.8,0.9\}$, and the sample fraction $\lambda_s \in \{0. 1, 0.2, \ldots ,0.9\}$. In Figs. 1, 2, 3 we show how the precision and recall of GMT, M2S, and mC depend on these parameters. Here we use the same unnormalized validation data and experimental setup as decribed in the main text section “Validation on ground-truth data” and in the Methods and Materials section “Validation on simulated data”. For each triple (λ, λ_f, λ_s) we randomly generate 20 dataset splits with these parameters and show the average precision and recall for each method over these trials. Our method GMT (thresholded GromovMatcher) is applied out-of-box with the default hyperparameter settings. The algorithm hyperparameters for mC and M2S are chosen optimally for each individual triple of dataset parameters (λ, λ_f, λ_s) to maximize the average F1 score in each setting. The hyperparameters searched over when optimizing mC and M2S are described in detail in **Appendix 2**.

Consistent with prior validation experiments, we find that GromovMatcher outperforms both mC and M2S in all dataset regimes, for low overlap and high overlap λ as well as for varying balances of features λ_f and samples λ_s. Remarkably, all three methods exhibit the same sensitivity to variations of (λ, λ_f, λ_s). All methods exhibit a monotonic decrease in their precision as λ_f drops from 0.9 to 0.5. In other words, the most challenging setting for matching both datasets is when dataset 1 and dataset 2 both have an equal number of unshared features (e.g. $\lambda_f = 0.5$). Likewise, the simplest setting for matching is when the features in dataset 2 are exactly a subset of the features in dataset 1 (e.g. $\lambda_f = 1$). Sensitivity to this parameter λ_f is most noticeable at low feature overlap $\lambda = 0.25$.

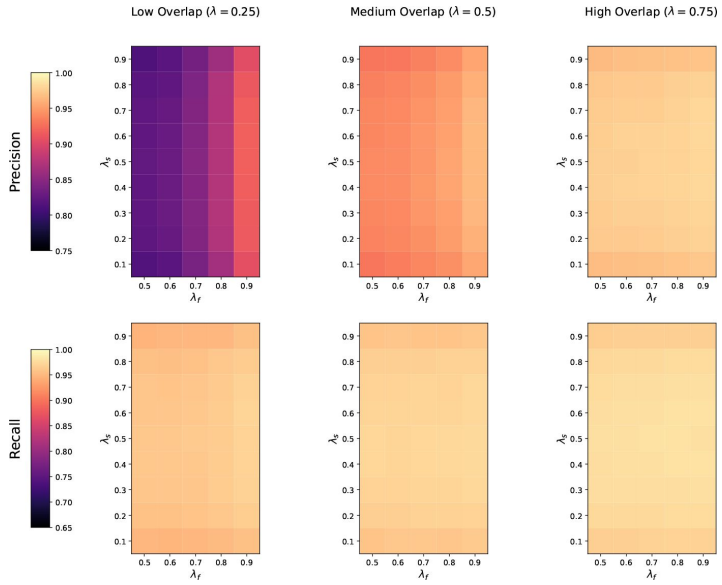
Appendix 4

Here we describe additional preprocessing details and analyses of the EPIC data.

Centered and scaled data - Negative mode

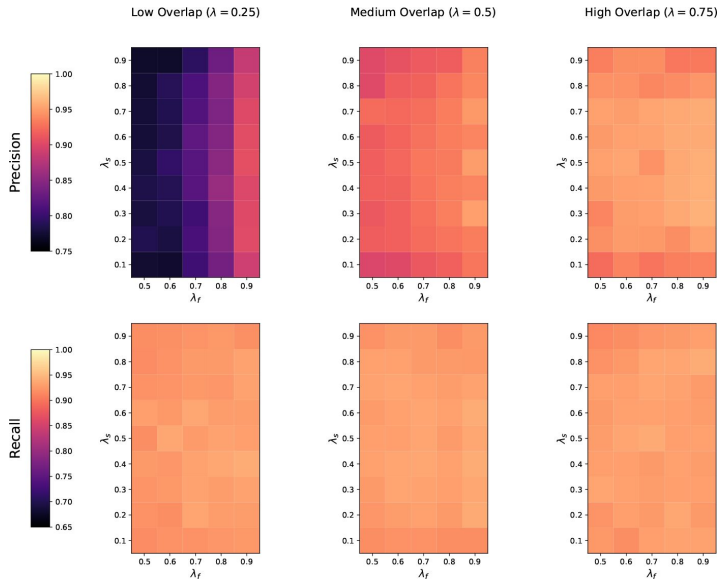
In this section, we present the results obtained on centered and scaled EPIC data in negative mode, shown in Figure 4 of our main paper. However, due to the smaller size of the validation subset (42 features examined in negative mode compared to 163 in positive mode), the evaluation of the performance of the three methods may be less reliable than in positive mode.

First, we align the CS and HCC studies in negative mode and detect a total of 449, 492, and 180 matches with GM, M2S, and metabCombiner, respectively. Similar to the positive mode analysis, we evaluate the precision and recall of the three methods on the 42 feature validation subset, of which 19 were manually matched. GM and M2S demonstrate identical F1-scores of 0.98, while metabCombiner performs poorly in comparison. GM is able to recover all 19 true matches and identified only 1 false positive, while M2S recovers no false positives but missed 1 true positive.



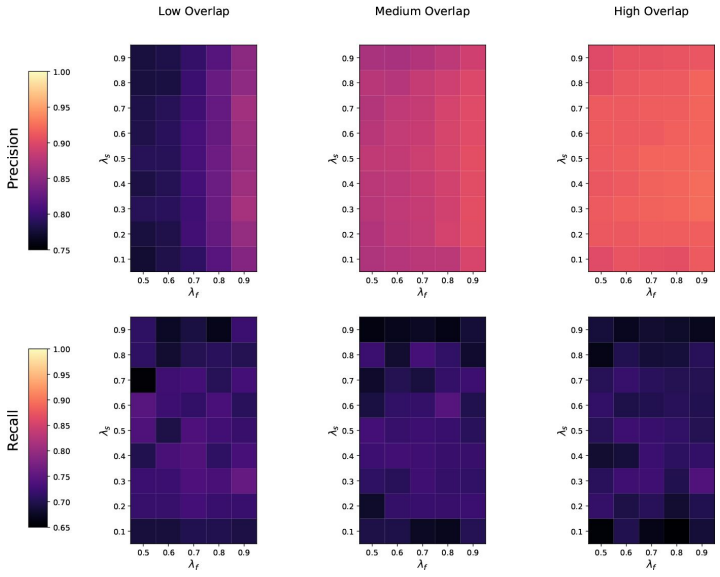
Appendix 3—figure 1.

Sensitivity of thresholded GromovMatcher (GMT) to feature overlap fraction λ, feature imbalance fraction λ_f, and sample imbalance fraction λ_s between two datasets being matched.



Appendix 3—figure 2.

Sensitivity of M2S to feature overlap fraction λ, feature imbalance fraction λ_f, and sample imbalance fraction λ_s between two datasets being matched.



Appendix 3—figure 3.

Sensitivity of metabCombiner (mC) to feature overlap fraction λ, feature imbalance fraction λ_f, and sample imbalance fraction λ_s between two datasets being matched.

Next, we align the CS and PC studies in negative mode and detect a total of 485, 569, and 314 matches with GM, M2S, and metabCombiner, respectively. Again, we evaluate the precision and recall of the three methods on the 42 feature validation subset, of which 26 were manually matched. MetabCombiner performs better than in the other EPIC pairings with an F1-score of 0.857, but is still outperformed by the other two methods. GM is slightly outperformed by M2S in this setting, with an almost identical precision of 0.93, but a slightly higher recall for M2S due to detecting 1 additional true positive. However, this remains a good performance for GM since M2S was optimally tuned using the validation subset itself.

Non-centered and non-scaled data

As a sensitivity analysis, we apply the three methods to EPIC data that has not been centered or scaled. The detailed results can be found in **Appendix 4 - Table 1**.

M2S was tuned manually on the validation subset to ignore feature intensities in both cases. As a result, it maintains its performance compared to our main experiment. On the other hand, the performance of GM and metabCombiner is affected by the lack of consistency in feature intensities. MetabCombiner's recall drops slightly but its precision remains comparable to that of our main experiment, with the method clearly favoring the latter. Although GM's recall decreases slightly in positive mode, it remains more precise than the optimally tuned M2S, and it balances precision and recall better than metabCombiner. Interestingly, GM's results in negative mode are improved compared to our main experiment, and it outperforms both mC and M2S. However, since the validation subset in negative mode is relatively small, these differences may not be significant. Nonetheless, GM maintains a good performance, similar to that of the optimally tuned M2S.

Similar to the analysis we conducted on centered and scaled data, we find a high number of false positives when aligning the CS study and the PC study in positive mode. Therefore, we manually examine the matches recovered by GM. Our examination reveals 2 false positives, 4 unclear matches, and 3 additional good matches that GM also identifies in our main analysis. This demonstrates that the lack of centering and scaling results in two additional false positives for GM that are not present in our main results.

Illustration for alcohol biomarker discovery

Loftfield et al. Loftfield et al. (2021) identified 205 features associated with alcohol intake in the CS study, using a false discovery rate (FDR) correction to account for multiple testing. By applying an FDR correction in our pooled analysis, we identify 243 features associated with alcohol intake. Out of those 243 features, 185 are consistent with the features identified in the discovery step of Loftfield et al. Loftfield et al. (2021), while 55 features are newly discovered (**Fig. 5c**). We examine the 20 features identified as significant in Loftfield et al.'s discovery analysis but that are not significant in our pooled analysis. Both manual and GM matching yield identical results for these features, indicating that the loss of significance is not due to incorrect matching. Upon further investigation, we find that these features do not demonstrate a meaningful association with alcohol intake in the HCC and PC studies. This observation is reinforced by the fact that none of these features are among the 10 features that persisted after the validation step in Loftfield et al.

Out of the 205 features initially discovered in Loftfield et al. Loftfield et al. (2021), 10 are replicated in the EPIC HCC and PC studies using the more stringent Bonferroni correction. When using a Bonferroni correction in our pooled analysis, we find significant association between alcohol intake and 92 features, 36 of which are effectively shared by the three studies. Notably, these features include all 10 features that were retained in Loftfield et al. (**Fig. 5c**).

	CS ⟷ HCC		CS ⟷ PC	
Method	**Precision**	**Recall**	**Precision**	**Recall**
GromovMatcher	0.988 (0.937, 0.999)	0.944 (0.876, 0.997)	0.873 (0.776, 0.932)	0.939 (0.854, 0.976)
M2S	0.967 (0.908, 0.991)	0.978 (0.923, 0.996)	0.855 (0.759, 0.917)	0.985 (0.919, 0.999)
metabCombiner	0.979 (0.889, 0.999)	0.511 (0.410, 0.612)	0.926 (0.766, 0.987)	0.379 (0.271, 0.499)

(a) Positive mode

	CS ⟷ HCC		CS ⟷ PC	
Method	**Precision**	**Recall**	**Precision**	**Recall**
GromovMatcher	0.950 (0.764, 0.997)	1.000 (0.832, 1.000)	0.964 (0.823, 0.998)	0.964 (0.823, 0.998)
M2S	1.000 (0.824, 1.000)	0.947 (0.754, 0.997)	0.931 (0.780, 0.988)	0.964 (0.823, 0.998)
metabCombiner	1.000 (0.566, 1.000)	0.263 (0.118, 0.488)	1.000 (0.785, 1.000)	0.500 (0.326, 0.674)

Appendix 4—table 1.

Precision and recall on the EPIC validation subset for unnormalized data in (a) positive mode, and (b) negative mode.

95% confidence intervals were computed using modified Wilson score intervals Brown et al. (2001); Agresti and Coull (1998).

This analysis illustrates how GromovMatcher can be used in the context of biomarker discovery, and its potential to allow for increased statistical power.

Appendix 5

Here we investigate how the choice of the reference dataset influences the discovery of metabolites shared across the CS, HCC and PC EPIC studies by GromovMatcher. All three methods considered in this paper, GromovMatcher, M2S, and metabCombiner, are limited to the comparison of two datasets. However, they can still be used to compare and pool multiple datasets using a multi-step procedure. Namely, this can be done by designating a 'reference' dataset and aligning all studies to it one by one. We take this exact approach in our analysis when aligning the CS, HCC, and PC studies of the EPIC data in positive mode. Namely, the HCC and PC studies are both aligned to the CS study (see main text **Fig. 5b**). However, this method raises two critical questions: *(i)* how does the use of a reference dataset affect matching results, and *(ii)* how is the matching affected by the choice of reference dataset.

To address these questions, we compare the features identified as common to the three studies using two different studies as references: the CS study used as reference in the main analysis, and the HCC study. For simplicity, let's denote $M_{\text{study1,study2}}$ the matching matrix obtained when aligning study 1 and study 2.

Changes in matching results when reference dataset is used

Concerning question (i), we compare two matchings: HCC to CS to PC (the matrix product $M^T_{\text{CS,HCC}} M_{\text{CS,PC}}$) which we will refer to as the reference matching, and the direct matching of PC to HCC ($M_{\text{HCC, PC}}$). Note that these matchings are not fully comparable as the former considers only features found in CS, potentially missing unique HCC and PC matches. We can however compare the two matchings on the subset of 706 features common to all three studies, as determined by the reference matching. We find that the direct matching supports 683 out of them, indicating that the matching via a reference still yields good results compared to the direct matching (see **Appendix 5 Fig. 1**).

Effect of reference dataset choice on matching results

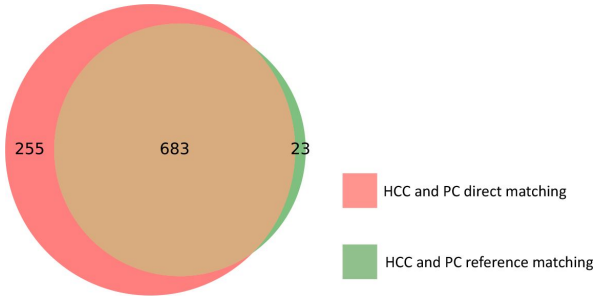
Concerning question *(ii)*, we compare the features identified as common to the three studies using two different studies as references: the CS study used in the paper, and the HCC study. We find that they identify 706 and 708 common features respectively, with an overlap of 640 features (see **Appendix 5 Fig. 2**). This highlights that the choice of reference does matter to some extent. In the paper, choosing CS as a reference was informed by CS's sample size, and study population.

References

Agresti A, Coull BA. (1998) **Approximate is better than "exact" for interval estimation of binomial proportions.** *Am Stat* **52**:119–126 https://doi.org/10.2307/2685469

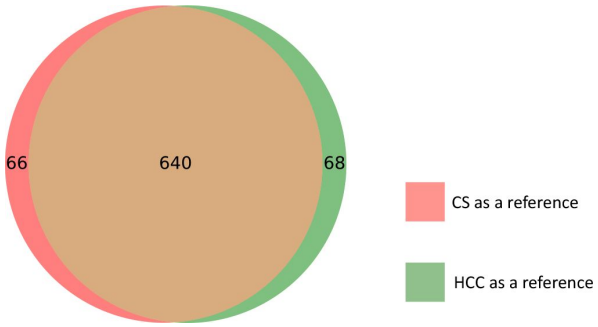
et al. *et al.* (2020) **A multi-omic analysis of birthweight in newborn cord blood reveals new underlying mechanisms related to cholesterol metabolism.** *Metabolism* **110**:154–292 https://doi.org/10.1016/j.metabol.2020.154292

Alvarez-Melis D, Jaakkola T. (2018) **Gromov-Wasserstein Alignment of Word Embedding Spaces.** *In: EMNLP Brussels* :1881–1890 https://doi.org/10.18653/v1/D18-1214.



Appendix 5—figure 1

Overlap between the 706 features common to the HCC and PC studies found via reference matching, and the 938 features common to HCC and PC found by direct matching



Appendix 5—figure 2.

Overlap between the features identified as common to the three EPIC studies using either the CS study or the HCC study as a reference.

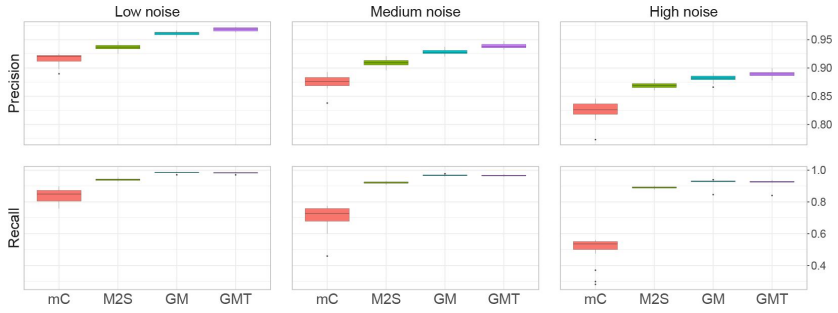


Figure 3—figure supplement 1.

Average precision and recall obtained on simulated data, with fixed overlap λ = 0.5.

The noise level corresponds to different values of σ_{RT} and σ_{FI}. High, medium and low noise level correspond to (σ_{RT}, σ_{FI}) = (0.2, 0. 1), (0.5, 0.5) and (1, 1) respectively. We run 20 simulations for each setting.

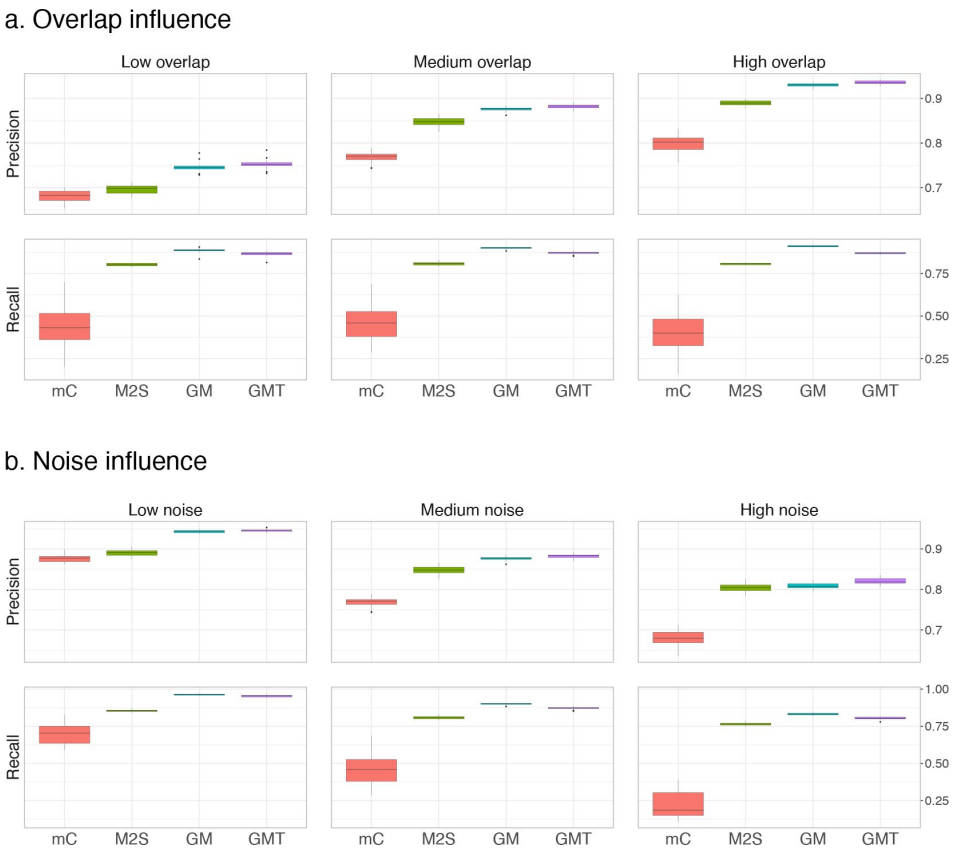


Figure 3—figure supplement 2.

Performance on centered and scaled data.

The feature intensities of both datasets are centered and scaled to have means of 0 and standard deviations of 1. The average precision and recall of the three methods are computed on 20 randomly generated pairs of datasets, for (a) three levels of overlap (low, medium and high corresponding to $\lambda = 0.25, 0.5$ and 0.75, respectively) with a medium noise level ((σ_{RT}, σ_{FI}) = (0.5, 0.5)), and (b) fixed medium overlap ($\lambda = 0.5$) and three different noise levels (low, medium and high corresponding to (σ_{RT}, σ_{FI}) = (0.2, 0.1), (0.5, 0.5) and (1, 1), respectively).

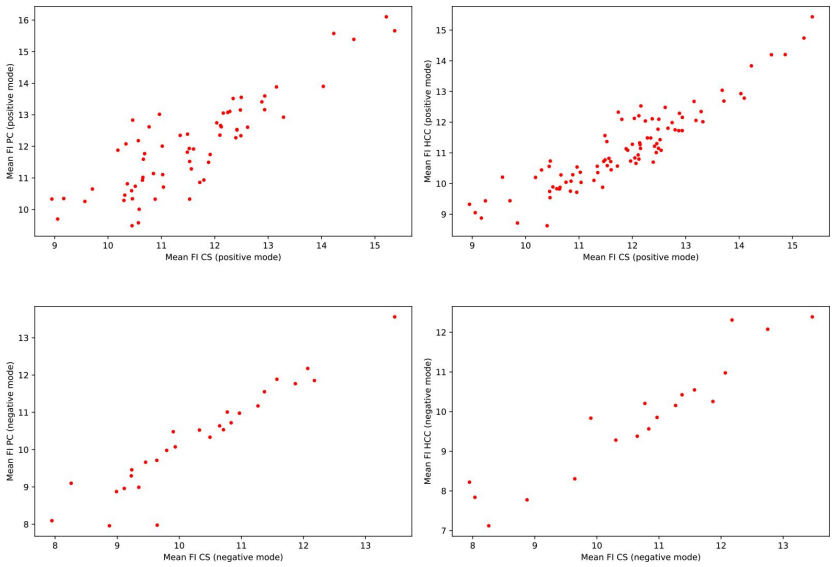


Figure 4—figure supplement 1.

Consistency of the mean feature intensities (FI) in EPIC.

Each scatter plot represents the mean feature intensities of manually matched features from the validation subset. Each dot represents a pair of manually matched features. The axis represent the mean feature intensities recorded in the two different studies.

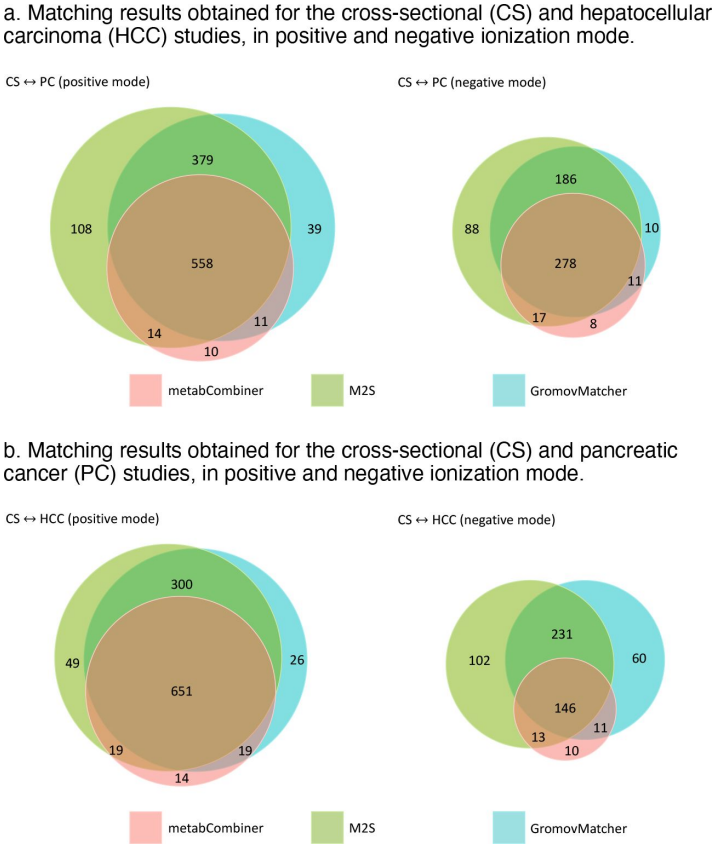


Figure 4—figure supplement 2.

Overlap between the matching results obtained by metabCombiner, M2S and GromovMatcher in EPIC. Venn diagrams are not up to scale.

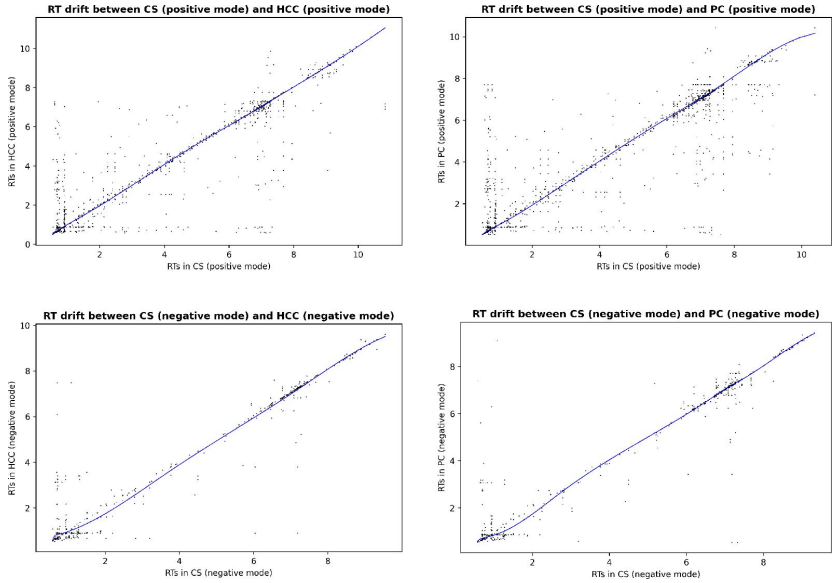


Figure 4—figure supplement 3.

Estimated RT drift between the EPIC studies aligned in the main experiment.

Each dot correspond to a candidate matched pair after the first step of GM (*m/z* constrained GW matching), before the RT drift estimation and RT based filtering.

Alvarez-Melis D, Jegelka S, Jaakkola TS. (2019) **Towards optimal transport with global invariances.** *In: AISTATS PMLR;* :1870–1879

Bedia C. (2022) **Metabolomics in environmental toxicology: Applications and challenges** *Trends Environ Anal Chem.* **34** https://doi.org/https://doi.org/10.1016/j.teac.2022.e00161

Beier F, Beinert R, Steidl G. (2022) **Multi-marginal Gromov-Wasserstein transport and barycenters.** *arXiv* https://doi.org/https://doi.org/10.48550/arXiv.2205.06725

Brown LD, Cai TT, DasGupta A. (2001) **Interval Estimation for a Binomial Proportion.** *Stat Sci* **16**:101–133 https://doi.org/10.1214/ss/1009213286

Brunius C, Shi L, Landberg R. (2016) **Large-scale untargeted LC-MS metabolomics data correction using between-batch feature alignment and cluster-based within-batch signal intensity drift correction.** *Metabolomics.* **12** https://doi.org/10.1007/s11306-016-1124-4

Chen L *et al.* (2021) **Metabolite discovery through global annotation of untargeted metabolomics data** *Nat Methods* **18**:1377–1385 https://doi.org/10.1038/s41592-021-01303-3

Chizat L, Peyré G, Schmitzer B, Vialard FX. (2018) **Unbalanced optimal transport: Dynamic and Kantorovich formulations** *J Funct Anal.* **274**:3090–3123 https://doi.org/10.1016/j.jfa.2018.03.008

Climaco Pinto R *et al.* (2022) **Finding Correspondence between Metabolomic Features in Untargeted Liquid Chromatography-Mass Spectrometry Metabolomics Datasets.** *Anal Chem* **94**:5493–5503 https://doi.org/10.1021/acs.analchem.1c03592

Courty N, Flamary R, Habrard A, Rakotomamonjy A. (2017) **Joint distribution optimal transportation for domain adaptation.** *NeurIPS*

Demetci P, Santorella R, Sandstede B, Noble WS, Singh R. (2022) **SCOT: Single-Cell Multi-Omics Alignment with Optimal Transport** *J Comput Biol.* **29**:3–18 https://doi.org/10.1089/cmb.2021.0446

et al. *et al.* (2019) **Gut microbiome structure and metabolic activity in inflammatory bowel disease.** *Nat Microbiol* **4**:293–305 https://doi.org/10.1038/s41564-018-0306-4

et al. *et al.* (2019) **Methodological issues in a prospective study on plasma concentrations of persistent organic pollutants and pancreatic cancer risk within the EPIC cohort.** *Environmental Research* **169**:417–433 https://doi.org/10.1016/j.envres.2018.11.027

Gomari DP, Schweickart A, Cerchietti L, Paietta E, Fernandez H, Al-Amin H, Suhre K, Krumsiek J. (2022) **Variational autoencoders learn transferrable representations of metabolomics data.** *Commun Biol.* **5** https://doi.org/10.1038/s42003-022-03579-3

Gromov M. (2001) **Metric Structures for Riemannian and Non-Riemannian Spaces.** https://doi.org/10.1007/978-0-8176-4583-0

Habra H, Kachman M, Bullock K, Clish C, Evans CR, Karnovsky A. (2021) **metabCombiner: Paired Untargeted LC-HRMS Metabolomics Feature Matching and Concatenation of Disparately Acquired Data Sets** *Anal Chem.* **93**:5028–5036 https://doi.org/10.1021/acs.analchem.0c03693

Hsu YHH *et al.* (2019) **PAIRUP-MS: Pathway analysis and imputation to relate unknowns in profiles from mass spectrometry-based metabolite data** *PLoS Comput Biol.* **15**:1–26 https://doi.org/10.1371/journal.pcbi.1006734

Ivanisevic J, Want EJ. (2019) **From Samples to Insights into Metabolism: Uncovering Biologically Relevant Information in LC-HRMS Metabolomics Data** *Metabolites.* **9** https://doi.org/10.3390/metabo9120308

Kantorovich LV. (2006) **On the translocation of masses.** *J Math Sci* **133**:1381–1382 https://doi.org/10.1007/s10958-006-0049-2

Li L, Zheng X, Zhou Q, Villanueva N, Nian W, Liu X, Huan T. (2020) **Metabolomics-Based Discovery of Molecular Signatures for Triple Negative Breast Cancer in Asian Female Population** *Sci Rep.* **10** https://doi.org/10.1038/s41598-019-57068-5

Liu Q, Walker D, Uppal K, Liu Z, Ma C, Tran V, Li S, Jones DP, Yu T. (2020) **Addressing the batch effect issue for LC/MS metabolomics data in data preprocessing.** *Sci Rep.* **10** https://doi.org/10.1038/s41598-020-70850-0

et al. *et al.* (2021) **Novel biomarkers of habitual alcohol intake and associations with risk of pancreatic and liver cancers and liver disease mortality.** *J Natl Cancer Inst* **113**:1542–1550 https://doi.org/10.1093/jnci/djab078

Mémoli F. (2011) **Gromov-Wasserstein Distances and the Metric Approach to Object Matching.** *Found Comput Math.* **11**:417–487 https://doi.org/10.1007/s10208-011-9093-5

Monge G. (1781) **Mémoire sur la théorie des déblais et des remblais.** *Mem Math Phys Acad Royale Sci* :666–704

Nitzan M, Karaiskos N, Friedman N, Rajewsky N. (2019) **Gene expression cartography.** *Nature.* **576**:132–137 https://doi.org/10.1038/s41586-019-1773-3

Patti GJ. (2011) **Separation strategies for untargeted metabolomics.** *J Sep Sci* **34**:3460–3469 https://doi.org/10.1002/jssc.201100532

Peyré G, Cuturi M, Solomon J. (2016) **Gromov-wasserstein averaging of kernel and distance matrices.** *In: ICML PMLR;* :2664–2672 https://doi.org/10.5555/3045390.3045671

et al., Peyré G, Cuturi M (2019) **Computational optimal transport: With applications to data science.** *Found Trends Mach Learn.* **11**:355–607 https://doi.org/10.1561/2200000073

Pirhaji L, Milani P, Leidl M, Curran T, Avila-Pacheco J, Clish CB, White FM, Saghatelian A, Fraenkel E. (2016) **Revealing disease-associated pathways by network integration of untargeted metabolomics.** *Nat Methods* **13**:770–776 https://doi.org/10.1038/nmeth.3940

Rappaport SM, Barupal DK, Wishart D, Vineis P, Scalbert A. (2014) **The Blood Exposome and Its Role in Discovering Causes of Disease.** *Environ Health Perspect* **122**:769–774 https://doi.org/10.1289/ehp.1308015

Reuther A *et al.* (2018) **>Interactive supercomputing on 40,000 cores for machine learning and data analysis.** *In: HPEC IEEE;* :1–6 https://doi.org/10.1109/HPEC.2018.8547629

et al. *et al.* (2002) **European Prospective Investigation into Cancer and Nutrition (EPIC): study populations and data collection.** *Public Health Nutr* **5**:1113–1124 https://doi.org/10.1079/PHN2002394

et al. *et al.* (2019) **Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming.** *Cell* **176**:928–943 https://doi.org/10.1016/j.cell.2019.01.006

Séjourné T, Feydy J, Vialard FX, Trouvé A, Peyré G. (2019) **Sinkhorn divergences for unbalanced optimal transport.** *arXiv preprint arXiv:191012958* https://doi.org/10.48550/arXiv.1910.12958

Sejourne T, Vialard FX, Peyré G. (2021) **The Unbalanced Gromov Wasserstein Distance: Conic Formulation and Relaxation. In: NeurIPS** *Curran Associates, Inc.* **34**:8766–8779

Skoraczynski G, Gambin A, Miasojedow B. (2022) **Alignstein: Optimal transport for improved LC-MS retention time alignment** *GigaScience* **11** https://doi.org/10.1093/gigascience/giac101

Slimani N *et al.* (2003) **Group level validation of protein intakes estimated by 24-hour diet recall and dietary questionnaires against 24-hour urinary nitrogen in the European Prospective Investigation into Cancer and Nutrition (EPIC) calibration study.** *Cancer Epidemiol Biomarkers Prev.* **12**:784–795

Smith CA, Want EJ, O'Maille G, Abagyan R, Siuzdak G. (2006) **XCMS: Processing Mass Spectrometry Data for Metabolite Profiling Using Nonlinear Peak Alignment, Matching, and Identification.** *Anal Chem.* **78**:779–787 https://doi.org/10.1021/ac051437y

Solomon J, Peyré G, Kim VG, Sra S. (2016) **Entropic Metric Alignment for Correspondence Problems.** *ACM Trans Graph* **35** https://doi.org/10.1145/2897824.2925903

et al. *et al.* (2016) **Alteration of amino acid and biogenic amine metabolism in hepatobiliary cancers: Findings from a prospective cohort stud** *Int J Cancer.* **138**:348–360 https://doi.org/10.1002/ijc.29718

et al. *et al.* (2021) **Metabolic perturbations prior to hepatocellular carcinoma diagnosis: Findings from a prospective observational cohort study** *Int J Cancer.* **148**:609–625 https://doi.org/10.1002/ijc.33236

Tautenhahn R, Patti GJ, Kalisiak E, Miyamoto T, Schmidt M, Lo FY, McBee J, Baliga NS, Siuzdak G. (2011) **metaX-CMS: second-order analysis of untargeted metabolomics data** *Anal Chem.* **83**:696–700 https://doi.org/10.1021/ac102980g

Vaughan AA, Dunn WB, Allwood JW, Wedge DC, Blackhall FH, Whetton AD, Dive C, Goodacre R. (2012) **Liquid Chromatography-Mass Spectrometry Calibration Transfer and Metabolomics Data Fusion** *Anal Chem.* **84**:9848–9857 https://doi.org/10.1021/ac302227c

Villani C. (2021) **Topics in optimal transportation** *American Mathematical Soc.*

Wang TJ *et al.* (2011) **Metabolite profiles and the risk of developing diabetes.** *Nat Med* **17**:448–453 https://doi.org/10.1038/nm.2307

Wishart DS. (2019) **Metabolomics for Investigating Physiological and Pathophysiological Processes.** *Physiol Rev* **99**:1819–1875 https://doi.org/10.1152/physrev.00035.2018

Yang KD, Damodaran K, Venkatachalapathy S, Soylemezoglu AC, Shivashankar G, Uhler C. (2020) **Predicting cell lineages using autoencoders and optimal transport.** *PLoS Comput Biol* **16**:1–20 https://doi.org/10.1371/journal.pcbi.1007828

Zhou B, Xiao JF, Tuli L, Ressom HW. (2012) **LC-MS-based metabolomics.** *Mol BioSyst.* **8**:470–481 https://doi.org/10.1039/C1MB05350G

Article and author information

Marie Breeur
Nutrition and Metabolism Branch, International Agency for Research on Cancer, Lyon, France

George Stepaniants
Massachusetts Institute of Technology, Department of Mathematics, Cambridge MA, United States

Pekka Keski-Rahkonen
Nutrition and Metabolism Branch, International Agency for Research on Cancer, Lyon, France

Philippe Rigollet
Massachusetts Institute of Technology, Department of Mathematics, Cambridge MA, United States

Vivian Viallon
Nutrition and Metabolism Branch, International Agency for Research on Cancer, Lyon, France
For correspondence: mailto:viallonv@iarc.who.int

Copyright

Editors

Reviewing Editor
Paula Fernandez
INTA, Buenos Aires, Argentina

Senior Editor
Murim Choi
Seoul National University, Seoul, Korea, the Republic of

Reviewer #1 (Public Review):

Summary:

The authors have implemented Optimal Transport algorithm in GromovMatcher for comparing LC/MS features from different datasets. This paper gains significance in the

proteomics field for performing meta-analysis of LC/MS data.

Strengths:

The main strength is that GromovMatcher acheives significant performance metrics compared to other existing methods. The authors have done extensive comparisons to claim that GromovMatcher performs well.

Weaknesses:

The authors might need to add the limitation of datasets and thus have tested/validated their tool using simulated data in the abstract as well.

https://doi.org/10.7554/eLife.91597.2.sa1

Reviewer #2 (Public Review):

Summary

The goal of untargeted metabolomics is to identify differences between metabolomes of different biological samples.

Untargeted metabolomics identifies features with specific mass-to-charge-ratio (m/z) and retention time (RT). Matching those to specific metabolites based on the model compounds from databases is laborious and not always possible, which is why methods for comparing samples on the level of unmatched features are crucial.
The main purpose of the GromovMatcher method presented here is to merge and compare untargeted metabolomes from different experiments. These larger datasets could then be used to advance biological analyses, for example, for identification of metabolic disease markers.

The main problem that complicates merging different experiments is that m/z and RT vary slightly for the same feature (metabolite).

The main idea behind the GromovMatcher is built on the assumption that if two features match between two datasets (that feature i from dataset 1 matches feature j from dataset 2, and feature k from dataset 1 matches feature l from dataset 2), then the correlations or distances between the two features within each of the datasets (i and k, and j and l) will be similar. The authors then use the Gromov-Wasserstein method to find the best matches matrix from these data.

The variation in m/z between the same features in different experiments is a user-defined value and it is initially set to 0.01 ppm. There is no clear limit for RT deviations, so the method estimates a non-linear deviation (drift) of RT between two studies. GromovMatcher estimates the drift between two studies, and then discards the matching pairs where the drift would deviate significantly from the estimate. It learns the drift from a weighted spline regression.

The authors validate the performance of their GromovMatcher method using a dataset of cord blood. They use 20 different splits and compare the GromovMatcher (both its GM and GMT iterations, whereby GMT version uses the deviation from estimated RT drift to filter the matching matrix) with two other matching methods: M2S and metabCombiner.

The second validation was done using a (scaled and centered) dataset of metabolics from cancer datasets from the EPIC cohort that were manually matched by an expert. This dataset was also used to show that using automated methods can identify more features that are associated with a particular group of samples than what was found by manual matching. Specifically, the authors identify additional features connected to alcohol consumption.

Strengths:

I see the main strength of this work in its combination of all levels of information (m/z, RT, and higher-order information on correlations between features) and using each of the types of information in a way that is appropriate for the measure. The most innovative aspect is using the Gromov-Wasserstein method to match the features based on distance matrices.

The authors of the paper identify two main shortcomings with previously established methods that attempt to match features from different experiments: a) all other methods require fine-tuning of user-defined parameters, and, more importantly, b) do not consider correlations between features. The main strength of the GromovMatcher is that it incorporates the information on distances between the features (in addition to also using m/z and RT).

Weaknesses:

The main weakness is that there seem not to be enough manually curated datasets that could be used for validation. It will, therefore, be important, for the authors, and the field in general to keep validating and improving their methods if more datasets become available.

The second weakness, as emphasized by the authors in the discussion is that the method as it is set up now can be directly used only to compare two datasets. I am confident that the authors will successfully implement novel algorithms to address this issue in the future.

https://doi.org/10.7554/eLife.91597.2.sa0

Author Response

The following is the authors' response to the original reviews.

> *eLife assessment*
>
> *This paper represents important findings when identifying untargeted metabolomics and its differences between metabolomes of different biological samples. GromovMatcher is the fantasy name for the soft development. The main idea behind it is built on the assumption of featuring and matching complex datasets. Although the manuscript reflects a solid analysis, it remains incomplete for validation with putative non-curated datasets.*

We are grateful to the eLife editor for taking the time and effort to assess our manuscript.

We are however unsure of what the editor means by "it remains incomplete for validation with putative non-curated datasets". As noted by Reviewer 2, manually curated datasets that could be used for validation are scarce. Most publicly available datasets do not contain sufficient information to establish a ground truth matching on which GromovMatcher, M2S, or metabCombiner can be tested. Even in the case where such a ground truth matching can be established, it must be performed by-hand through a manual matching process which is extremely time-consuming and requires very specific expertise. This, in our opinion, only highlights the need for automatic alignment methods such as metabCombiner, M2S or GromovMatcher.

We do agree that the performance of GromovMatcher (and its competitors) needs to be validated further, and we plan to continue validating GromovMatcher as additional data becomes available in EPIC and other cohorts. With that in mind, the lack of publicly available validation data is the reason why we conducted such an extensive simulation study, arguably

more comprehensive than previous validations, exploring challenging settings that we believe reflect real-life scenarios (main text “Validation on ground-truth data” and Appendix 3). We would like to stress that this allows us to highlight previously ignored limitations of the previously published methods, metabCombiner and M2S.

We wish to thank the editor and reviewers for their time and efforts in reviewing our manuscript which led to many significant additions to our paper. Namely we:

• Performed an additional sensitivity analysis (Appendix 3) exploring how an imbalance in the number of features or samples between two studies being matched (e.g. the dataset split), affects the quality of matchings found by GromovMatcher, metabCombiner, and M2S.

• Investigated how changing or removing the reference dataset (Appendix 5) in the EPIC study (main text “Application to EPIC data”), affects the results of GromovMatcher.

• Improved alignment matrix visualizations in Fig. 3a for all four methods tested on the validation data, to highlight more clearly which feature matches were correctly identified or missed.

The revised paper is uploaded as the file “main_elife_revision.pdf” where all revisions are highlighted in blue as well as a copy “main_elife_revision_nohighlights.pdf” where revisions are not highlighted.

Summary:

> *The authors have implemented the Optimal Transport algorithm in GromovMatcher for comparing LC/MS features from different datasets. This paper gains significance in the proteomics field for performing meta-analysis of LC/MS data.*
>
> *Strengths:*
>
> *The main strength is that GromovMatcher achieves significant performance metrics compared to other existing methods. The authors have done extensive comparisons to claim that GromovMatcher performs well.*
>
> *Weaknesses:*
>
> *There are two weaknesses.*
>
> *(1) When the number of features is reduced the precision drops to ~0.8.*

We would like to clarify that this drop in precision occurs in the challenging setting where only a small proportion of metabolites are shared between both datasets (e.g., the overlap – or proportion of shared features - was 25% in our simulation study). When two untargeted metabolic datasets share only 25% of their features, this is a challenging setting for any automated matching method as the vast majority 75% of the features in both datasets must remain unmatched.

In such settings, the reviewer correctly observes that the precision of GromovMatcher algorithms (GM and GMT) drops within the range of 0.80 - 0.85 (Figure 3b, top left panel). Such a precision of 0.8 or larger is still competitive compared with the alternative methods MetabCombiner (mC) and M2S whose precisions drop below 0.8 (see main text Fig. 3b, top left panel).

> ***Public Reviews:***

Summary:

> *The authors have implemented the Optimal Transport algorithm in GromovMatcher for comparing LC/MS features from different datasets. This paper gains significance in the proteomics field for performing meta-analysis of LC/MS data.*
>
> *Strengths:*
>
> *The main strength is that GromovMatcher achieves significant performance metrics compared to other existing methods. The authors have done extensive comparisons to claim that GromovMatcher performs well.*
>
> *Weaknesses:*
>
> *There are two weaknesses.*
>
> *(1) When the number of features is reduced the precision drops to ~0.8.*

We would like to clarify that this drop in precision occurs in the challenging setting where only a small proportion of metabolites are shared between both datasets (e.g., the overlap – or proportion of shared features - was 25% in our simulation study). When two untargeted metabolic datasets share only 25% of their features, this is a challenging setting for any automated matching method as the vast majority 75% of the features in both datasets must remain unmatched.

In such settings, the reviewer correctly observes that the precision of GromovMatcher algorithms (GM and GMT) drops within the range of 0.80 - 0.85 (Figure 3b, top left panel). Such a precision of 0.8 or larger is still competitive compared with the alternative methods MetabCombiner (mC) and M2S whose precisions drop below 0.8 (see main text Fig. 3b, top left panel).

Precision is measured as the number of metabolite pairs correctly matched divided by all matches identified by a method. In other words, even in the challenging setting when the number of shared features (true matches) between both datasets is small (e.g. low 25% overlap), upwards of 80% of the feature matches found by GromovMatcher are correct which is a very encouraging result.

> *(2) How applicable is the method for other non-human datasets?*

We thank the reviewer for raising this question. The crux of the matter concerning the application to animal data revolves around the hypothesis that correlations between metabolites in two different studies are preserved. Theoretically, the metabolome operates under similar principles in humans, governed by an underlying network of biochemical reactions. Consequently, in comparable human populations, the GM hypothesis is likely to hold to some extent.

However, in practice, application to animal data is more complicated. Animal studies tend to have smaller sample sizes and often stem from intervention-driven scenarios, such as mice subjected to specific diets or chemicals. This results in deliberate alterations in metabolic structures which makes finding two comparable animal studies less likely. To investigate the reviewer's question, we have searched through the two predominant LC-MS dataset repositories (MetaboLights and NIH Metabolomics Workbench) but did not find any pairs of

> ***Reviewer #1 (Public Review):***

Summary:

> *The authors have implemented the Optimal Transport algorithm in GromovMatcher for comparing LC/MS features from different datasets. This paper gains significance in the proteomics field for performing meta-analysis of LC/MS data.*
>
> *Strengths:*
>
> *The main strength is that GromovMatcher achieves significant performance metrics compared to other existing methods. The authors have done extensive comparisons to claim that GromovMatcher performs well.*
>
> *Weaknesses:*
>
> *There are two weaknesses.*
>
> *(1) When the number of features is reduced the precision drops to ~0.8.*

We would like to clarify that this drop in precision occurs in the challenging setting where only a small proportion of metabolites are shared between both datasets (e.g., the overlap – or proportion of shared features - was 25% in our simulation study). When two untargeted metabolic datasets share only 25% of their features, this is a challenging setting for any automated matching method as the vast majority 75% of the features in both datasets must remain unmatched.

In such settings, the reviewer correctly observes that the precision of GromovMatcher algorithms (GM and GMT) drops within the range of 0.80 - 0.85 (Figure 3b, top left panel). Such a precision of 0.8 or larger is still competitive compared with the alternative methods MetabCombiner (mC) and M2S whose precisions drop below 0.8 (see main text Fig. 3b, top left panel).

Precision is measured as the number of metabolite pairs correctly matched divided by all matches identified by a method. In other words, even in the challenging setting when the number of shared features (true matches) between both datasets is small (e.g. low 25% overlap), upwards of 80% of the feature matches found by GromovMatcher are correct which is a very encouraging result.

> *(2) How applicable is the method for other non-human datasets?*

We thank the reviewer for raising this question. The crux of the matter concerning the application to animal data revolves around the hypothesis that correlations between metabolites in two different studies are preserved. Theoretically, the metabolome operates under similar principles in humans, governed by an underlying network of biochemical reactions. Consequently, in comparable human populations, the GM hypothesis is likely to hold to some extent.

However, in practice, application to animal data is more complicated. Animal studies tend to have smaller sample sizes and often stem from intervention-driven scenarios, such as mice subjected to specific diets or chemicals. This results in deliberate alterations in metabolic structures which makes finding two comparable animal studies less likely. To investigate the reviewer's question, we have searched through the two predominant LC-MS dataset repositories (MetaboLights and NIH Metabolomics Workbench) but did not find any pairs of comparable animal studies due to the reasons mentioned above. One potential strategy to navigate this issue could entail regressing the metabolic intensities against the variables that

notably differ between the two animal populations and running GM using the residual intensities. This would be an interesting direction for future research and additional validation would be needed to test the robustness of GM in this setting.

> ***Reviewer #2 (Public Review):***
>
> *Summary:*
>
> *The goal of untargeted metabolomics is to identify differences between metabolomes of different biological samples. Untargeted metabolomics identifies features with specific mass-to-charge ratio (m/z) and retention time (RT). Matching those to specific metabolites based on the model compounds from databases is laborious and not always possible, which is why methods for comparing samples on the level of unmatched features are crucial.*
>
> *The main purpose of the GromovMatcher method presented here is to merge and compare untargeted metabolomes from different experiments. These larger datasets could then be used to advance biological analyses, for example, for the identification of metabolic disease markers. The main problem that complicates merging different experiments is m/z and RT vary slightly for the same feature (metabolite).*
>
> *The main idea behind the GromovMatcher is built on the assumption that if two features match between two datasets (that feature I from dataset 1 matches feature j from dataset 2, and feature k from dataset 1 matches feature l from dataset 2), then the correlations or distances between the two features within each of the datasets (i and k, and j and l) will be similar. The authors then use the Gromov-Wasserstein method to find the best matches matrix from these data.*
>
> *The variation in m/z between the same features in different experiments is a user-defined value and it is initially set to 0.01 ppm. There is no clear limit for RT deviations, so the method estimates a non-linear deviation (drift) of RT between two studies. GromovMatcher estimates the drift between the two studies and then discards the matching pairs where the drift would deviate significantly from the estimate. It learns the drift from a weighted spline regression.*
>
> *The authors validate the'performance of their GromovMatcher method by a validation experiment using a dataset of cord blood. They use 20 different splits and compare the GromovMatcher (both its GM and GMT iterations, whereby the GMT version uses the deviation from estimated RT drift to filter the matching matrix) with two other matching methods: M2S and metabCombiner.*
>
> *The second validation was done using a (scaled and centered) dataset of metabolics from cancer datasets from the EPIC cohort that was manually matched by an expert. This dataset was also used to show that using automatic methods can identify more features that are associated with a particular group of samples than what was found by manual matching. Specifically, the authors identify additional features connected to alcohol consumption.*
>
> *Strengths:*
>
> *I see the main strength of this work in its combination of all levels of information (m/z, RT, and higher-order information on correlations between features) and using each of the types of information in a way that is appropriate for the measure. The most innovative aspect is using the Gromov-Wasserstein method to match the features based on distance matrices.*

We thank the reviewer for acknowledging this strength of our proposed GromovMatcher method.

> *The authors of the paper identify two main shortcomings with previously established methods that attempt to match features from different experiments: a) all other methods require fine-tuning of user-defined parameters, and, more importantly, b) do not consider correlations between features. The main strength of the GromovMatcher is that it incorporates the information on distances between the features (in addition to also using m/z and RT).*
>
> *Weaknesses:*
>
> *The first, minor, weakness I could identify is that there seem not to be plenty of manually curated datasets that could be used for validation.*

We thank the reviewer for raising this issue concerning manually curated validation data.

Manually curated datasets available for validation purposes are indeed scarce. This stems from the laborious nature of matching features across diverse studies, hence the need for automatic matching methods. Our future strategy involves further validation of the GromovMatcher approach as more data becomes accessible in EPIC and other cohorts.

The scarcity of real-life publicly available datasets that can be used for validation purpose is the reason why we conducted an extensive simulation study (main text “Validation on ground-truth data” and Appendix 3). It is notably thorough, arguably more comprehensive than previous validations, utilizes real-life untargeted data, and imitates situations where data originates from distinct untargeted metabolomics studies, complete with realistic noise parameters encompassing RT, mz, and feature intensities. Our validation study comprehensively explores the performance of GromovMatcher, M2S, and metabCombiner, including in challenging realistic settings where there is a nonlinear drift in retention times, varying levels of feature overlaps between studies, normalizations of feature intensities, as well as imbalances in the number of features and samples present in the studies being matched.

> *The second is also emphasized by the authors in the discussion. Namely, the method as it is set up now can be directly used only to compare two datasets.*

This is indeed a limitation that is common to all three methods considered in this paper. However, all these methods, GromovMatcher, M2S, and metabCombiner, can still be used to compare and pool multiple datasets using a multi-step procedure. Namely, this can be done by designating a 'reference' dataset and aligning all studies to it one by one. We take this exact approach in our paper when aligning the CS, HCC, and PC studies of the EPIC data in positive mode (main text “Application to EPIC data”). Namely, the HCC and PC studies are both aligned to the CS study by running GromovMatcher twice, and after obtaining these matchings, our analysis is restricted to those features in HCC and PC that are present in the CS study.

After the reviewer’s comment, we have added an additional sensitivity analysis in Appendix 5, to compare the results produced by GromovMatcher depending on the choice of the reference study. Namely, setting the reference study to either the CS study or the HCC study, GromovMatcher identified 706 and 708 common features respectively, with an overlap of 640 features. This highlights that the choice of reference does matter to some extent. In our original analysis of the EPIC data, choosing CS as the reference was motivated by the fact that CS had the largest sample size (compared to HCC and PC) and a subset of features in HCC and

PC were already matched by experts to the CS study which we could use for validation (see Loftfield et al. (2021). J Natl Cancer Inst.).

As mentioned in the discussion section of our manuscript, the recently proposed multimarginal Gromov-Wasserstein algorithm (Beier, F., Beinert, R., & Steidl, G. (2023). Information and Inference) could potentially allow multiple metabolomic studies to be matched using one optimization routine (e.g. without the designation of a 'reference study' for matching). We have not explored this possibility in depth yet as fast numerical methods for multimarginal GW are still in their infancy. Also, such multimarginal methods rely on the computation and storage of coupling or matching matrices that are tensors where the number of dimensions is equal to the number of datasets being matched. Therefore, multimarginal methods have large memory costs, which currently precludes their application for the matching of multiple metabolomics datasets.

> ***Reviewer #2 (Recommendations For The Authors):***
>
> *(1) I was struggling with the representation used in Figure 3a. The gray points overlayed over the green points on a straight line are difficult to visually quantify. I found that my eyes mainly focused on the pattern of the red dots.*

Figure 3a has been modified to improve visual clarity. Namely we have consistently reordered the rows and columns of the coupling matrices such that the true positive matches (green points) are spatially separated from the false negative matches (red points). Now the fraction of true positive and false negative matches can be appreciated much more clearly by eye in Figure 3a.

> *(2) I would also like to add the caveat that I cannot judge whether the authors used the other two methods that they compare with GromovMatcher (the M2S and metabCombiner) optimally. But I also do not see any evidence that they did not. Hopefully one of the other reviewers can address that.*

We appreciate the reviewer for highlighting the comparison of our approach GromovMatcher to the other existing methods M2S and MetabCombiner (mC). Both M2S and mC depend on tens of hyperparameters each with a discrete or continuous set of values that must be properly optimized to infer accurate matchings between dataset features. We detail in Appendix 2 how the hyperparameters of the M2S and mC methods are optimally tuned to achieve the best possible performance on the validation ground-truth data. Namely, both in the simulation study and on EPIC data, we grid-search over all important hyperparameters in the M2S and mC methods and choose those parameter combinations that result in the highest F1 score, averaged over 20 random trials. We remark that no such hyperparameter optimization was performed for our GromovMatcher method. As shown in Figures 3 and 4 of the main text, we find that GromovMatcher outperforms M2S and mC even in these cases when the hyperparameters of M2S and mC are tuned to predict optimal feature matchings.

Given the large combinatorial space of hyperparameter choices, we believe we have thoroughly tested the important hyperparameter combinations that users of M2S and mC would be likely to explore in their own research.

> *(3) Validation*
>
> *(3a) The first validation is done on a split cord blood dataset. I could not clearly see from the paper how sensitive the result is to the dataset split.*

We are grateful for the reviewer's question and have included new experiments in Appendix 3 which show how the results of GromovMatcher, M2S, and MetabCombiner are affected by

the dataset split. In our original manuscript, our validation ground-truth experiment began with an untargeted metabolomic dataset consisting of n = 499 samples and p = 4,712 metabolic features which is split equally into two datasets consisting of an equal number of samples n1 = n2 and an equal number of metabolic features p1 = p2. The features of these equal-sized datasets would then be matched by our method.

Now in Appendix 3 (Figs. 1-3) we show the sensitivity of all three alignment methods (GromovMatcher, M2S, and MetabCombiner) when we vary the fraction of samples in dataset 1 over dataset 2 given by n1/ n2, the overlap in shared features between both datasets, and the fraction of metabolic features in dataset 1 that are not present in dataset 2 which affects the feature sizes of both datasets p1/ p2. We find that all alignment methods are able to maintain a consistent precision and recall score when these three dataset split parameters are varied. GromovMatcher achieves a higher precision and recall than M2S and MetabCombiner for all choices of dataset split, agreeing with the validation experiment results from the main text (see main text Fig. 3). All three methods tested decrease in precision (without dropping in recall) when dataset 1 and dataset 2 contain an equal number of unshared features (e.g. when p1 = p2). Therefore, these sensitivity experiments in Appendix 3 show that our results in the main text are performed in the most challenging setting for the dataset split.

> *(3b) The second validation was done using a (scaled and centered) dataset of metabolics from cancer datasets from the EPIC cohort that was manually matched by an expert. Here the authors observe that metabCombiner has good precision, but lags in recall. And M2S has a very similar performance to GromovMatcher. The authors explain this by the fact that the drift in RT between the two experiments is mostly linear and thus does not affect the M2S performance. Can the authors find a different validation dataset where the drift in RT is not linear? If yes, it would be interesting to add it to the paper.*

We thank the reviewer for raising this question. As mentioned above, curated validation datasets such as the EPIC study analyzed in our paper are very rare and we do not currently have a validation study with a nonlinear retention time drift.

Nevertheless, we performed an additional analysis of simulated data (reported in Appendix 2 – “M2S hyperparameter experiments” and Appendix 2 – Table 1) that demonstrates the decrease in M2S performance when the simulated drift is nonlinear. As presented in Appendix 2 – Table 1, in a low overlap setting with a linear drift which corresponds to the EPIC data, precision and recall were 0.831 and 0.934 respectively, instead of 0.769 and 0.905 in the main analysis where the drift was nonlinear.