MOTL: enhancing multi-omics matrix factorization with transfer learning

David Hirst[1,*], Morgane Térézol[1], Laura Cantini[2], Paul Villoutreix[1], Matthieu Vignes[3], Anaïs Baudot[1, 4, 5,*]

1 Aix Marseille Univ, INSERM, MMG, Marseille, France
2 Institut Pasteur, Université Paris Cité, CNRS UMR 3738, Machine Learning for Integrative Genomics Group, F-75015 Paris, France
3 School of Mathematical and Computational Sciences, College of Science, Massey University, Palmerston North, New Zealand
4 CNRS, Marseille, France
5 Barcelona Supercomputing Center, Spain

* Correspondence:
David Hirst, david.hirst@univ-amu.fr; Anaïs Baudot, anais.baudot@univ-amu.fr

Abstract

Joint matrix factorization is a popular method for extracting lower dimensional representations of multi-omics data. It disentangles underlying mixtures of biological signals, facilitating efficient sample clustering, disease subtyping, or biomarker identification, for instance. However, when a multi-omics dataset is generated from only a limited number of samples, the effectiveness of matrix factorization is reduced. Addressing this limitation, we introduce MOTL (Multi-Omics Transfer Learning), a novel framework for multi-omics matrix factorization with transfer learning based on MOFA (Multi-Omics Factor Analysis). MOTL infers latent factors for a small multi-omics dataset, with respect to those inferred from a large heterogeneous learning dataset. We designed two protocols to evaluate transfer learning approaches, based on simulated and real multi-omics data. Using these protocols, we observed that MOTL improves the factorization of multi-omics datasets, comprised of a limited number of samples, when compared to factorization without transfer learning. We showcase the usefulness of MOTL on a glioblastoma dataset comprised of a small number of samples, revealing an enhanced delineation of cancer status and subtype thanks to transfer learning.

1 Introduction

Omics data have transformed the study of biology and medicine by enabling high-throughput measurements of the activity and abundance of biological molecules and processes (Conesa and Beck, 2019; Dermitzakis, 2008; Manzoni et al., 2018) In recent years, the fields of biology and medicine have been revolutionized by the increased availability of multi-omics datasets (Conesa and Beck, 2019; Subramanian et al., 2020; Huang et al., 2017). A multi-omics dataset is comprised of multiple data matrices, each containing a different type of omics data (e.g., mRNA transcript counts, genomic mutations, DNA methylation prevalence). The integrated analysis of multi-omics data can provide a better understanding of a biological system than that obtained from the analysis of a single omics data matrix (Rohart et al., 2017; Huang et al., 2017; Rappoport and Shamir, 2018; Subramanian et al., 2020; Chauvel et al., 2020; Pierre-Jean et al., 2020; Cantini et al., 2021). Indeed, different omics contain complementary information that contribute to a more comprehensive overview of the underlying biological system. Additionally, using multiple omics can reveal insights into relationships between the different biological layers they represent. Combining omics is also expected to reduce the impact of noise. However, multi-omics data poses further analysis challenges beyond those encountered in single omics data analysis. These challenges include increased dimensionality, the presence of multiple data types, diverse sources of technological noise, and diverse ranges of variability. In this context, there has been an increased need for methods able to carry out integrated analysis of multiple omics.

The development of multi-omics analysis tools is an active area of research. A category of multi-omics analysis tools that has proven effective is matrix factorization (Rappoport and Shamir, 2018; Tini et al., 2019; Chauvel et al., 2020; Pierre-Jean et al., 2020; Cantini et al., 2021). Matrix factorization infers a lower dimensional representation of the observed data, in which a sufficiently informative proportion of the original signal is retained (Stein-O'Brien et al., 2018). Most classical matrix factorization approaches were designed for the analysis of a single data matrix. Applying matrix factorization to a single omics matrix produces a score matrix and a weight matrix, both of which contain values for latent factors that are potentially associated with different sources of underlying biological signal. The values in the weight matrix ideally represent signal across the assayed biological features, and the values in the score matrix represent the signal across the samples. For a multi-omics dataset, one of the strategies is to jointly factorize multiple omics data matrices. Various methods are now available for this purpose (Cantini et al., 2021). Multi-omics joint matrix factorization methods typically produce a weight matrix for each omics, and either a shared score matrix or a combination of shared and omics specific score matrices. Many multi-omics matrix factorization methods are extensions of classical methods designed for single omics. For example, intNMF (Chalise and Fridley, 2017) extends non-negative matrix factorization to the multi-omics setting, and allows the user to determine the relative contribution of each omics to the extraction of joint signal. JIVE (Lock et al., 2013) extends principal component analysis to model both joint and omics specific signal. MOFA (Argelaguet et al., 2018), which is an extension of Factor Analysis, uses a Bayesian framework to account for the presence of multiple data types and to distinguish between joint and omics specific signal. Overall, the factors inferred by multi-omics matrix factorization can be used for clustering samples to reveal disease sub-types, for identifying molecular profiles and biomarkers associated with diseases, as well as for prediction of outcomes such as drug response and survival (Rappoport and Shamir, 2018; Taroni et al., 2019; Pierre-Jean et al., 2020; Cantini et al., 2021; Banerjee et al., 2023). A challenge for matrix factorization is that it requires a large amount of observed data to produce a meaningful representation. However, there are cases where omics are measured from only a small number of samples, due to the rareness or cost of obtaining the data, and so there is a need for methods which help mitigate this challenge (Weiss et al., 2016; Stein-O'Brien et al., 2018; Banerjee et al., 2023).

For a dataset generated from a small number of samples, transfer learning is a potential solution to the limited effectiveness of matrix factorization. Transfer learning is a machine learning approach in which information extracted from a large learning domain is used to improve the performance of a task applied to a smaller target domain (Weiss et al., 2016; Stein-O'Brien et al., 2019; Taroni et al., 2019; Banerjee et al., 2023). It is assumed that the two domains share an overlapping latent space, allowing knowledge from the learning domain to be transferred to the application of the task to the target domain. Transfer learning has been successfully used in various machine learning applications, including image classification, text sentiment classification and recommendation systems (Weiss et al., 2016; Veeramachaneni et al., 2019; Dong et al., 2021; Banerjee et al., 2023). In a transfer learning approach to omics matrix factorization, information inferred from the prior factorization of a learning dataset, comprised of a large number of samples from a heterogeneous set of biological conditions, is incorporated into the factorization of a small target dataset (Peng et al., 2021; Banerjee et al., 2023). It is assumed that if the latent factors inferred from the learning dataset represent common underlying biological processes, they should help improve the factorization of the target dataset (Stein-O'Brien et al., 2019).

The usefulness of transfer learning approaches to matrix factorization, for omics data analysis, has been demonstrated in contexts in which both the target and learning datasets were comprised of a single matrix of omics data. In these cases, transfer learning was used to infer a score matrix for the target dataset by projecting it onto a weight matrix inferred from a learning dataset. In one study, Stein-O'Brien et al. factorized a mouse single cell RNA-seq learning dataset with the Bayesian non-negative matrix factorization algorithm CoGAPS (Fertig et al., 2010). Then, they used the transfer learning tool projectR (Sharma et al., 2020) to infer a score matrix for a human time course bulk RNA-seq dataset. The resulting factors were associated with known spatiotemporal differences across the samples. In another example, Davis-Marcisak et al. factorized a mouse single cell RNA-seq learning dataset with CoGAPS, and then used projectR to infer a score matrix for bulk RNA-seq data from human cancer samples. They observed an association between a particular projectR factor and outcomes in metastatic melanoma. Taroni et al. developed MultiPLIER , a transfer learning framework, which they demonstrated by firstly applying the non-negative matrix factorization algorithm PLIER (Mao et al., 2019) to a subset of Recount2 to infer a weight matrix. Recount2 is a compendium of RNA-seq data obtained from 70,000+ human samples taken across more than 2,000 studies (Collado-Torres et al., 2017). Taroni et al. then used a blood cell compendium of microarray gene expression data as the target dataset, for which they inferred a score matrix with MultiPLIER, as well as factorizing the target dataset directly with PLIER. For counts of a cell type of interest, MultiPLIER inferred a more highly correlated factor than was inferred by direct factorization of the target dataset. They also used microarray gene expression data for 79 samples from a rare disease group called antineutrophil cytoplasmic autoantibody associated vasculitis (AAV) as a target dataset. There are no AAV samples in the Recount2 compendium, yet the MultiPLIER factors were positively correlated with their best match from factors inferred by direct factorization of the target dataset.

It has thus been demonstrated that the application of matrix factorization to a large, heterogeneous learning dataset can yield factors containing transferable information, that are biologically relevant to target datasets from different organisms, diseases, cell types and omics platforms. However, existing transfer learning approaches to matrix factorization have been designed for, and demonstrated on, datasets comprised of single omics data only. To the best of our knowledge, transfer learning approaches to joint multi-omics matrix factorization are currently lacking.

We here introduce MOTL (Multi-Omics Transfer Learning), a novel Bayesian transfer learning algorithm for multi-omics matrix factorization. MOTL is based on MOFA, a popular tool for integrative multi-omics analysis (Argelaguet et al., 2018). We first present the statistical framework and implementation of MOTL. Next, we propose two protocols, that we designed based on simulated and real multi-omics datasets, for evaluating the performance of transfer learning approaches. We used these protocols to evaluate MOTL, and observed that, for a target multi-omics dataset comprised of a small number of samples, our transfer learning approach to matrix factorization is more effective than matrix factorization without transfer learning. Lastly, we showcase a practical use case of MOTL on a limited glioblastoma sample set, revealing an enhanced delineation of cancer status and subtype thanks to transfer learning.

2 Results

2.1 MOTL: A new transfer learning framework for multi-omics matrix factorization

We propose MOTL, a transfer learning approach to multi-omics matrix factorization. MOTL is based on MOFA (Argelaguet et al., 2018), which uses variational Bayesian inference (Blei et al., 2017). Consider a multi-omics target dataset, $\boldsymbol{T}$, consisting of omics matrices, $\boldsymbol{T}^{(m)}$, $m = 1, ..., M$. Each $\boldsymbol{T}^{(m)} = \left[t_{nd}^{(m)}\right] \in \mathbb{R}^{N_t \times D_m}$ contains data for N_t samples (rows) and D_m features (columns), where $t_{nd}^{(m)}$ is the value for the nth sample and the dth feature from the mth matrix. The features depend on which molecules were assayed to generate a given omics matrix; for example the features for mRNA counts are genes, while those for DNA methylation are CpG sites.

We wish to jointly factorize $\boldsymbol{T}^{(m)}$ into a matrix of sample scores, $\boldsymbol{Z} = [z_{nk}] \in \mathbb{R}^{N_t \times K}$, and an omics specific matrix of feature weights, $\boldsymbol{W}^{(m)} = \left[w_{kd}^{(m)}\right] \in \mathbb{R}^{K \times D_m}$. The resulting lower dimensional representation is based on K factors, which ideally represent underlying biological signals associated with some biological condition(s) of interest. z_{nk} is the score for the nth sample and the kth factor, while $w_{kd}^{(m)}$ is the weight for the kth factor and the dth feature from the mth matrix. The kth column vector of $\boldsymbol{Z}$, denoted by $\boldsymbol{z}_{:k}$, contains scores for factor k, while the nth row vector, $\boldsymbol{z}_{n:}$, contains scores for sample n. The kth row vector of $\boldsymbol{W}^{(m)}$, denoted by $\boldsymbol{w}_{k:}^{(m)}$, contains weights for factor k, while the dth column vector, $\boldsymbol{w}_{:d}^{(m)}$, contains weights for feature d.

We are concerned with the situation in which N_t is small, exacerbating the curse of dimensionality, and therefore we expect to improve the factorization of $\boldsymbol{T}$ by employing a transfer learning approach (see Figure 1). We do this transfer learning by incorporating values that have already been inferred from the prior factorization of a learning dataset, $\boldsymbol{L}$, and we assume that the Bayesian matrix factorization algorithm MOFA (Argelaguet et al., 2018) was used for factorizing $\boldsymbol{L}$.

The learning dataset consists of omics matrices $\boldsymbol{L}^{(m)}$, $m = 1, ..., M$. Each $\boldsymbol{L}^{(m)} = \left[l_{nd}^{(m)}\right] \in$

$\mathbb{R}^{N_l \times D_m}$ contains data for the same D_m features as $\boldsymbol{T}^{(m)}$, but for a different set of $N_l > N_t$ samples. We hypothesise that if $\boldsymbol{L}$ is comprised of samples from a heterogeneous set of biological conditions, then the factorization of $\boldsymbol{L}$ will yield information that is relevant for the factorization of $\boldsymbol{T}$.

MOTL is based on the variational Bayesian inference methodology used by MOFA (Methods 4.2). We have modified the MOFA algorithm to enable us to supplement the factorization of $\boldsymbol{T}$ by incorporating values already inferred from the prior factorization of $\boldsymbol{L}$. For MOTL, we assume that each observed $t_{nd}^{(m)}$ is a random variable, with a likelihood that is conditional on vectors $\boldsymbol{z}_{n:}$ and $\boldsymbol{w}_{:d}^{(m)}$. We model continuous, counts and binary data with the same likelihoods and link functions that MOFA uses. For observed continuous data we thus assume a Gaussian likelihood, into which we include a feature-wise precision parameter, $\tau_d^{(m)}$, for each feature d from matrix m. For observed binary data we assume a Bernoulli likelihood, and for observed counts data we assume a Poisson likelihood. In contrast to MOFA, MOTL doesn't center the input data during factorization fitting, as we want to incorporate an intercept that is compatible with the factorization of $\boldsymbol{L}$. We therefore replace $\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(m)}$ with $a_d^{(m)} + \boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(m)}$ in the likelihood, where $a_d^{(m)}$ is the feature-wise intercept for feature d, from matrix m. We infer $a_d^{(m)}$ values based on the MOFA factorization of $\boldsymbol{L}$ (Methods 4.7). MOTL accepts missing $t_{nd}^{(m)}$ values, and therefore it is not necessary to remove features with missing values, or perform imputation, before using MOTL.

In order to carry out a transfer learning approach to matrix factorization, MOTL uses the matrix of feature weights, $\boldsymbol{W}^{(m)}$, vector of feature-wise intercepts, $\boldsymbol{a}^{(m)} = \left[a_d^{(m)}\right] \in \mathbb{R}^{D_m}$, and vector of feature-wise precision parameter values, $\boldsymbol{\tau}^{(m)} = \left[\tau_d^{(m)}\right] \in \mathbb{R}^{D_m}$, inferred for each $\boldsymbol{L}^{(m)}$ with a prior MOFA factorization of $\boldsymbol{L}$. Instead of modelling these as random variables, we treat them as constants. We aim to obtain point estimates of z_{nk} values, for which we assume the same joint prior distribution as MOFA does,

$$p(\boldsymbol{Z}) = \prod_{n=1}^{N_t} \prod_{k=1}^{K} \text{Normal}(z_{nk}|0, 1) \tag{1}$$

MOTL obtains point estimates of z_{nk} values by approximating the joint posterior distribution $p(\boldsymbol{Z}|\boldsymbol{T})$ with a variational distribution:

$$q(\boldsymbol{Z}) = \prod_{n=1}^{N_t} \prod_{k=1}^{K} q(z_{nk}) = \prod_{n=1}^{N_t} \prod_{k=1}^{K} \text{Normal}(z_{nk}|\mu_{nk}, \sigma_{nk}) \tag{2}$$

MOTL infers $q(\boldsymbol{Z})$ iteratively. At each iteration, the value of each parameter is updated while all other parameter values are held fixed. MOTL optimizes the joint variational distribution by iterating until convergence. For each z_{nk}, the expected value, $\mathbb{E}_q\left[z_{nk}\right] = \mu_{nk}$, is used as the point estimate throughout and after model fitting. MOTL uses the same update equations for the parameters of $q(z_{nk})$ as MOFA, but with the inclusion of intercepts:

$$\sigma_{nk}^2 = \left(\sum_{m=1}^{M} \sum_{d=1}^{D_m} \tau_{nd}^{(m)} \left(w_{kd}^{(m)}\right)^2 + 1 \right)^{-1} \tag{3}$$

$$\mu_{nk} = \sigma^2_{nk} \sum_{m=1}^{M} \sum_{d=1}^{D_m} \tau^{(m)}_{nd} w^{(m)}_{kd} \left(\hat{t}^{(m)}_{nd} - a^{(m)}_{d} - \sum_{j \neq k} z_{nj} w^{(m)}_{jd} \right) \tag{4}$$

where $\tau^{(m)}_{nd}$ is the precision for the nth sample and dth feature from the mth matrix, and $\hat{t}^{(m)}_{nd}$ denotes a (possibly) transformed observed data point (Methods 4.2). For observed data with a Gaussian assumed likelihood, a feature-wise precision, $\tau^{(m)}_{d}$, is used instead of $\tau^{(m)}_{nd}$, and there is no transformation, meaning $\hat{t}^{(m)}_{nd} = t^{(m)}_{nd}$. For observed data with a non-Gaussian assumed likelihood, MOTL transforms the data to yield Gaussian pseudo-data values, which it does not center. The transformation to Gaussian pseudo-data allows updates of $q(\boldsymbol{Z})$ to be based on the assumption of Gaussian observed data. When MOFA transforms observed data with a Bernoulli assumed likelihood, it derives and uses a precision parameter, $\tau^{(m)}_{nd}$, for each sample and feature. For observed data with a Poisson assumed likelihood, it derives and uses a feature-wise precision, $\tau^{(m)}_{d}$. Thus for Bernoulli observed data, MOTL initializes $\tau^{(m)}_{nd}$ values with $\tau^{(m)}_{d}$ values, which are averages of the $\tau^{(m)}_{nd}$ values returned by the factorization of $\boldsymbol{L}$, and these are subsequently updated at each iteration of the algorithm. For Poisson observed data, MOTL uses the $\tau^{(m)}_{d}$ values obtained from the prior factorization of $\boldsymbol{L}$, and holds them fixed.

To monitor convergence we calculate the evidence lower bound (ELBO), which can be used to evaluate how well a variational distribution approximates a posterior distribution of interest. We calculate the ELBO with respect to $\boldsymbol{Z}$:

$$\text{ELBO}(\boldsymbol{Z}) = \mathbb{E}_q \left[\log p(\boldsymbol{T}|\boldsymbol{Z})\right] + \mathbb{E}_q \left[\log p(\boldsymbol{Z})\right] - \mathbb{E}_q \left[\log q(\boldsymbol{Z})\right] \tag{5}$$

For $\boldsymbol{T}^{(m)}$ with a non-Gaussian assumed likelihood, we use the same lower bound for $\log\, p\left(t^{(m)}_{nd}|\boldsymbol{Z}\right)$ as MOFA does. Maximizing this lower bound, coupled with the use of $\hat{t}^{(m)}_{nd}$ values, allows updates of $q(\boldsymbol{Z})$ based on the assumption of Gaussian observed data (Jaakkola and Jordan, 2000; Seeger and Bouchard, 2012). We calculate the ELBO at regular intervals, and the number of iterations between each calculation is a user defined parameter. We check for convergence based on the absolute change in the ELBO (from the previous check) as a percentage of the initial ELBO. The algorithm is deemed to have converged when a specified number of changes in the ELBO are consecutively below a threshold. Both the threshold, and the required number of consecutive changes falling below this threshold, are user defined parameters.

We allow factors to be dropped during training, based on the fraction of variance explained:

$$R^2_{mk} = 1 - \frac{\sum_{n=1}^{N_t} \sum_{d=1}^{D_m} \left(\hat{t}^{(m)}_{nd} - a^{(m)}_{d} - z_{nk} w^{(m)}_{kd} \right)^2}{\sum_{n=1}^{N_t} \sum_{d=1}^{D_m} \left(\hat{t}^{(m)}_{nd} - a^{(m)}_{d} \right)^2} \tag{6}$$

We drop the factor with the lowest R^2_{mk} that does not have any R^2_{mk} above the threshold. We assess factors in this way after each round of updates. After convergence the algorithm returns $\boldsymbol{Z}$ and $\boldsymbol{W}^{(m)}$ matrices for the factors that have not been dropped.

MOTL is available as an open source R implementation (Methods 4.9)

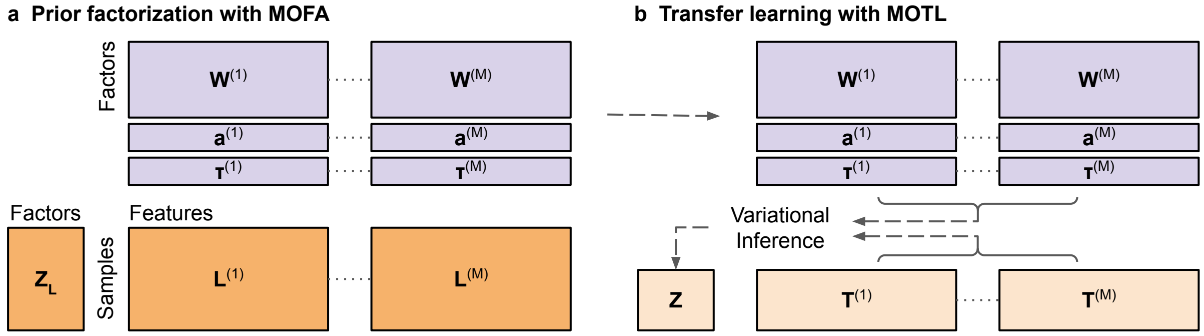

Figure 1: Overview of MOTL, our transfer learning approach to joint multi-omics matrix factorization based on variational Bayesian inference. **a** A multi-omics learning dataset, $\boldsymbol{L}$, consisting of M omics matrices, $\boldsymbol{L}^{(m)}$, $m = 1, ..., M$, is factorized with MOFA to infer a matrix of feature weights, $\boldsymbol{W}^{(m)}$, vector of feature-wise intercepts, $\boldsymbol{a}^{(m)}$, and a vector of feature-wise precision parameter values, $\boldsymbol{\tau}^{(m)}$, for each $\boldsymbol{L}^{(m)}$. **b** The feature weight, intercept, and precision parameter values, inferred from the factorization of $\boldsymbol{L}$, are incorporated into the factorization of a multi-omics target dataset, $\boldsymbol{T}$, for which MOTL infers a matrix of sample scores, $\boldsymbol{Z}$, with variational inference.

2.2 Evaluation protocol using simulated multi-omics data

We first designed and implemented a transfer learning evaluation protocol based on simulated multi-omics datasets, which we generated from groundtruth factors (Methods 4.3). In brief, in each simulation instance, we generated a multi-omics dataset $\boldsymbol{Y}$, which we subsequently split into a target dataset, $\boldsymbol{T}$, and a learning dataset, $\boldsymbol{L}$. $\boldsymbol{Y}$ consisted of matrices of counts, continuous, and binary data. We generated each matrix, $\boldsymbol{Y}^{(m)}$, from a statistical distribution conditional on random matrices $\boldsymbol{Z}$ and $\boldsymbol{W}^{(m)}$, which each contained values for K groundtruth factors. The kth column vector of $\boldsymbol{Z}$ contained sample scores for the kth groundtruth factor. The kth row vector of $\boldsymbol{W}^{(m)}$ contained feature weights for that same factor. We varied the number of groundtruth factors across configurations, using $K \in \{20, 30\}$. We generated $\boldsymbol{Z}$ based on the group membership of samples. In each instance, we created two groups of five samples for the target dataset. The learning dataset samples belonged to either 20 or 40 differently sized groups of randomly selected sizes. For each groundtruth factor and group, the sample scores were generated using a mean parameter value that was common to all samples in the group. We induced heterogeneity by allowing the means to vary across groups and factors, randomly selecting each group mean, for a given groundtruth factor, from a pool of three possible values. We split each $\boldsymbol{Y}^{(m)}$ into $\boldsymbol{T}^{(m)}$ and $\boldsymbol{L}^{(m)}$, based on the sample groups used to generate $\boldsymbol{Z}$. In each instance $\boldsymbol{T}$ contained data for 10 samples, while the expected number of samples for $\boldsymbol{L}$ was $\in \{400, 1000\}$.

For each simulation instance, we factorized $\boldsymbol{L}$ with MOFA (Methods 4.6). We then factorized $\boldsymbol{T}$ with our transfer learning method MOTL (Methods 4.7), incorporating output from the factorization of $\boldsymbol{L}$. To benchmark the performance of MOTL, we also performed direct MOFA factorizations (i.e., factorization without transfer learning) of $\boldsymbol{T}$ datasets. We evaluated both the MOTL and direct MOFA factorization of each $\boldsymbol{T}$, and compared the overall performance of each approach. We evaluated factorizations of each $\boldsymbol{T}$ by calculating an F1 score (Methods 4.8), to measure how well the factorization allowed us to uncover differentially active groundtruth factors underlying $\boldsymbol{T}$. The kth groundtruth factor was differentially active for $\boldsymbol{T}$ if the mean parameter values used to simulate the sample scores, for that factor, differed between the two groups of target dataset samples. Factorization with MOTL led to higher F1 scores than direct MOFA factorization, indicating that the MOTL factorizations were more effective in uncovering differentially active latent signal from $\boldsymbol{T}$ datasets (Figure 2). This was observed across all simulation configurations, and the overall uplift in mean F1 score for MOTL, when compared to direct MOFA factorization, was 0.21 (p-value $<$ 0.01, Methods 4.8). We thus observed that transfer learning with MOTL was more effective in uncovering differentially active latent signal, when compared to direct MOFA factorization (without transfer learning) of $\boldsymbol{T}$.

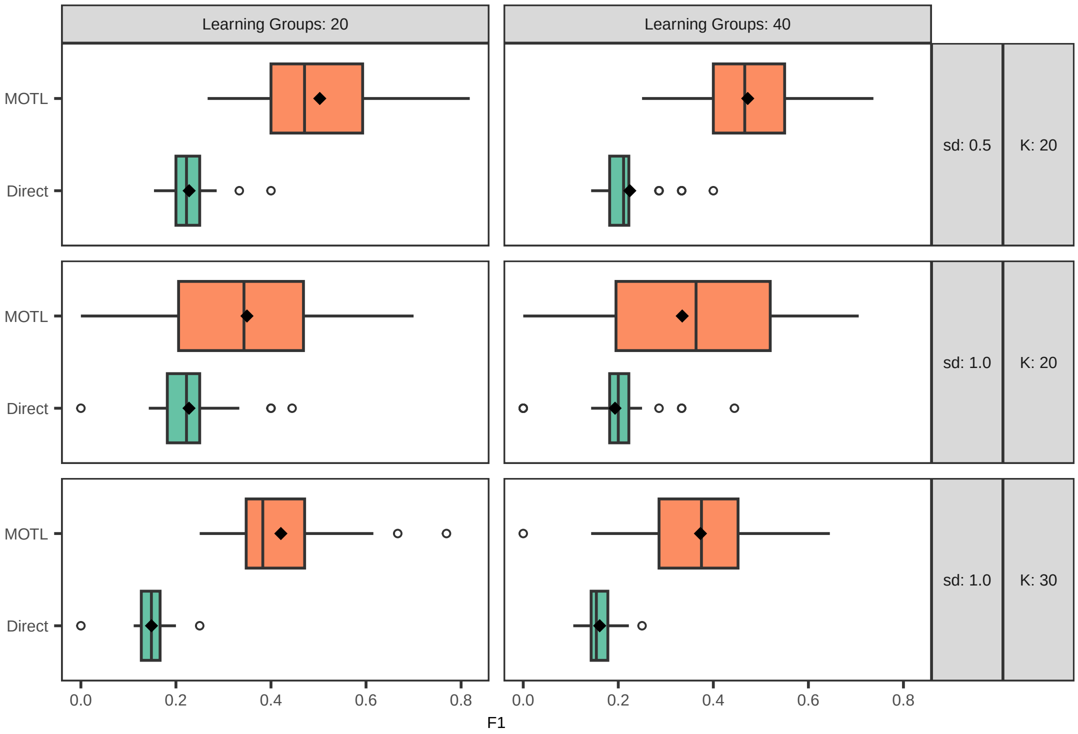

Figure 2: Evaluation of factorizations of small simulated multi-omics target datasets with and without MOTL transfer learning. The boxplots represent the F1 scores obtained for different factorization approaches and simulation configuration settings. Simulation configurations varied in the number of groups of samples used for the learning dataset (*Learning Groups*), the number of groundtruth factors (K), or the standard deviation used to simulate z_{nk} values (sd). F1 scores take a value between 0 and 1, and higher values indicate better factorizations. Each boxplot is based on 30 F1 scores. The hinges of the boxes are the 25^{th} and 75^{th} percentiles, the middle lines are medians, the diamonds are the mean values, and the whiskers are either extreme values or extend 1.5 times the inter-quartile range from the hinge.

2.3 Evaluation protocol using TCGA multi-omics data

We next designed, and implemented, a second transfer learning evaluation protocol, based on TCGA multi-omics data (Methods 4.4). We used four types of omics data: log2 transformed mRNA counts, log2 transformed miRNA counts, DNA methylation M-values, and single nucleotide variation (SNV) binary data, which we obtained for 32 different cancer types. We created target datasets using data from three cancer types; acute myeloid leukemia (LAML), pancreatic adenocarcinoma (PAAD) and skin cutaneous melanoma (SKCM). We created these target datasets by firstly creating four reference datasets.

Each reference dataset, $\boldsymbol{R}$, contained multi-omics data for all samples from either two, or all three of the cancer types. We then randomly split every $\boldsymbol{R}$ into non-overlapping target datasets which each contained only five samples per cancer type (Figure 3**a**). We merged data from the remaining 29 cancer types into a learning dataset, $\boldsymbol{L}$, which contained multi-omics data for 7,217 samples.

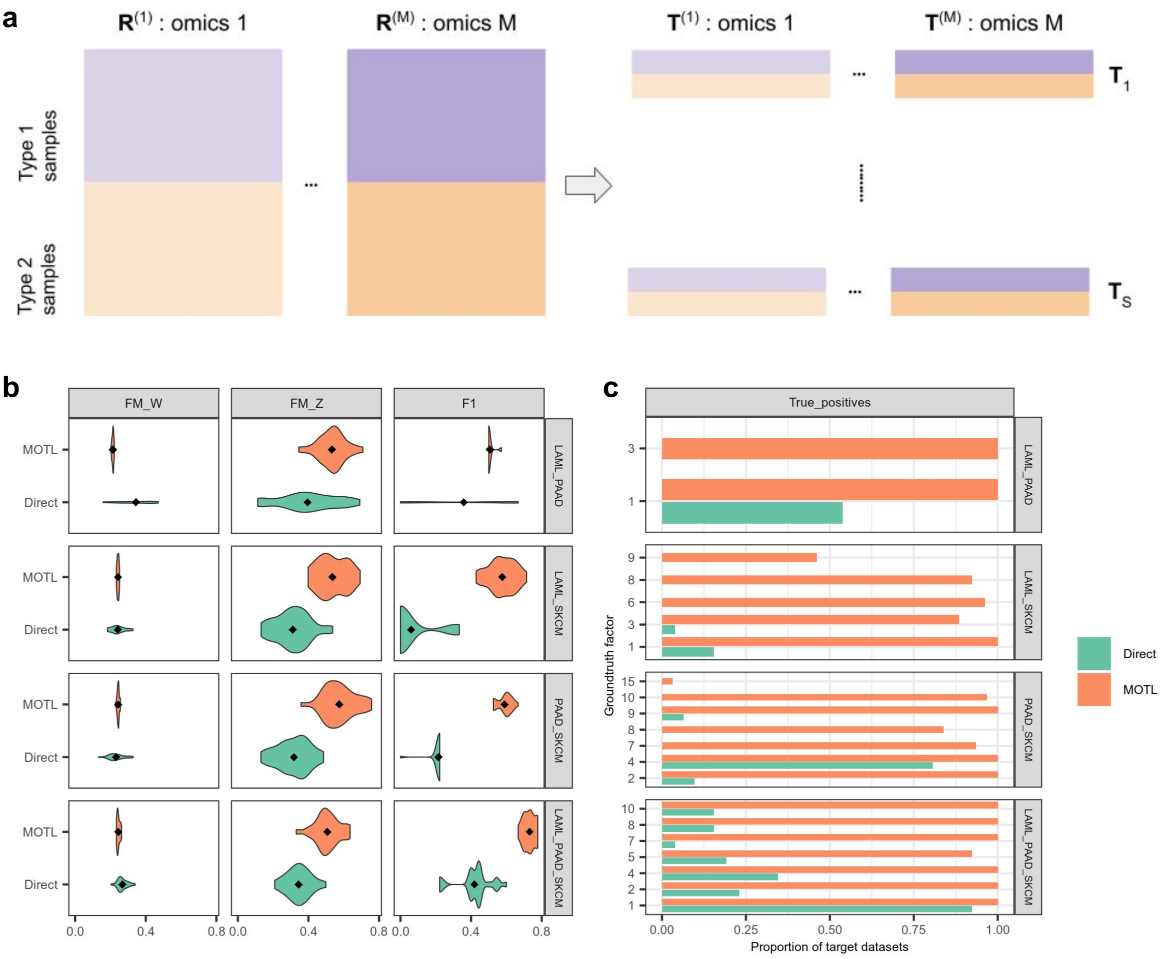

Figure 3: Evaluations using TCGA multi-omics data. **a** TCGA target datasets are created from two or three cancer types: We created a reference dataset, $\boldsymbol{R}$, containing multi-omics data for all samples from the selected cancer types. We then randomly split $\boldsymbol{R}$ into non-overlapping target, $\boldsymbol{T}$, datasets, containing multi-omics data for five samples per cancer type. We did this for subsets of the set of cancer types {LAML, PAAD, SKCM}; in total we created, and split, four reference datasets, each of which contained multi-omics data for all samples from either two (LAML and PAAD, LAML and SKCM, PAAD and SKCM), or three (LAML, PAAD and SKCM) cancer types. **b** Comparison of factorization approaches applied to TCGA multi-omics datasets. Violin plots of F-measure values for weight matrix factors (*FM_W*), F-measure values for score matrix factors (*FM_Z*), and F1 scores (*F1*). For each evaluation score, higher values indicate better factorizations. Scores are plotted by factorization method and by the cancer types characterizing the target dataset samples. **c** Frequency with which differentially active groundtruth TCGA factors were true positives. A differentially active groundtruth factor was a true positive if it was predicted as being differentially active based on a factorization of a target dataset. Each bar represents the proportion of target datasets, for which the factorization led to the differentially active groundtruth factor being a true positive. Proportions are plotted by factorization method, and by the cancer types characterizing the reference and target dataset samples.

We factorized $\boldsymbol{L}$ with MOFA (Methods 4.6), based on which we used MOTL to factorize each $\boldsymbol{T}$ (Methods 4.7). To benchmark the performance of MOTL, we also performed direct MOFA factorizations (without transfer learning) of $\boldsymbol{T}$ datasets. In order to evaluate the factorizations of $\boldsymbol{T}$ datasets, we factorized $\boldsymbol{R}$ datasets with MOFA and treated the resulting score, $\boldsymbol{Z}$, and weight, $\boldsymbol{W}^{(m)}$, matrices as groundtruth factor matrices (Methods 4.8). We were interested in how well the factorizations of $\boldsymbol{T}$ datasets uncovered the groundtruth, and we used F-measure values and F1 scores to evaluate this.

We calculated F-measure values to assess the correlation between factors inferred from each $\boldsymbol{T}$, and the groundtruth factors obtained from the factorization of the corresponding $\boldsymbol{R}$ dataset (Methods 4.8). We calculated F-measure values for weight matrices (FM_W), as well as for score matrices (FM_Z) . The overall mean FM_W for MOTL was slightly lower (0.03 reduction, p-value < 0.01) than for direct MOFA factorizations of $\boldsymbol{T}$ datasets (Figure 3**b**, column 1), which is the result of lower average relevance counterbalancing higher average recovery. We concluded from this that groundtruth $\boldsymbol{W}^{(m)}$ factors were more easily uncovered with those transferred from $\boldsymbol{L}$ than with direct MOFA factorization. However, despite factor trimming during MOTL factorization, some remaining transferred factors were less associated with groundtruth factors than those obtained with direct MOFA factorization. It is of note that the difference in average FM_W is attributable to the datasets containing LAML and PAAD samples only. If we exclude these, there is no difference in FM_W (p-value 0.59). The overall mean FM_Z for MOTL was 0.20 higher (p-value < 0.01, Methods 4.8) than for direct MOFA factorizations (Figure 3**b**, column 2). We thus observed that the $\boldsymbol{Z}$ factors obtained with MOTL, from $\boldsymbol{T}$ datasets, were more correlated with groundtruth factors, overall, than those obtained with direct MOFA factorization.

We also calculated F1 scores to measure how well the factorizations of $\boldsymbol{T}$ datasets uncovered differentially active groundtruth factors (Methods 4.8). For each $\boldsymbol{T}$ dataset, the groundtruth factors were the factors obtained from factorization of the corresponding $\boldsymbol{R}$ dataset. We considered the kth groundtruth factor to be differentially active if the distribution of scores in the kth column vector, of groundtruth $\boldsymbol{Z}$, differed between the cancer types. We can simultaneously evaluate the $\boldsymbol{Z}$ and $\boldsymbol{W}^{(m)}$ factors, and assess the overall quality of factorizations, by checking for an uplift in F1 scores (Figure 3**b**, column 3). MOTL (0.34 uplift, p-value < 0.01) yielded higher F1 scores than direct MOFA factorization, meaning it was more effective in uncovering latent activity that varied across cancer types.

We next examined differentially active groundtruth factors, with an initial focus on the frequency with which these factors were true positives (Figure 3**c**). For each factorization of a $\boldsymbol{T}$ dataset, a differentially active groundtruth factor was a true positive if it was predicted as being differentially active based on the factorization of $\boldsymbol{T}$. The unique count of true positives was a component of each F1 score value (Methods 4.8). We further performed a gene set enrichment analysis to identify the pathways and processes associated with differentially active groundtruth factors that were true positives (Methods 4.8, Supplementary Table 1), and that explained at least 1% of the mRNA variance in $\boldsymbol{R}$ (Supplementary Table 2).

The factorization of the $\boldsymbol{R}$ dataset containing all LAML and PAAD samples yielded six groundtruth factors, of which two were differentially active; Factor 1 and Factor 3. Both of these factors were true positives for 100% of MOTL factorizations of $\boldsymbol{T}$ datasets containing subsets of five LAML and PAAD samples (Figure 3**c**). In contrast, only one of these factors was a true positive for direct MOFA factorizations, and for just over half of the same $\boldsymbol{T}$

datasets (Figure 3**c**). Factor 1 is significantly associated with developmental processes, cell communication and immunity signaling. Factor 3 displays similar enrichments, with an additional specific enrichment related to the regulation of gene expression in beta cells (Supplementary Table 1).

The factorization of the $\boldsymbol{R}$ dataset containing all LAML and SKCM samples yielded 12 groundtruth factors, of which five were differentially active. Four out these five factors were true positives for MOTL factorizations for more than 80% of the $\boldsymbol{T}$ datasets; one factor was a true positive for just under half of the $\boldsymbol{T}$ datasets. Importantly, only two of these five groundtruth factors were true positives for direct MOFA factorizations, and only for a small proportion of the $\boldsymbol{T}$ datasets. Four out of these five differentially active groundtruth factors, that were true positives, explained at least 1% of the mRNA variance in $\boldsymbol{R}$: Factor 1, Factor 3, Factor 8, and Factor 9. We performed gene set enrichment analysis on these four factors. Factor 1 is significantly associated with extracellular organisation, developmental processes, cell communication signaling, and Fc Receptor mediated immune processes (Supplementary Table 1). Factor 3 is significantly associated with hematopoeitic cell lineage, Pi3K/AKT and G protein-coupled signaling, and chemokine, interleukin, interferon signaling. Both Factors 1 and 3 are associated with keratinisation and formation of the cornified envelope. Factors 8 and 9 do not present significant pathway enrichments beyond keratinisation and processes already associated with the first two factors.

The factorization of the $\boldsymbol{R}$ dataset containing all PAAD and SKCM samples yielded 17 groundtruth factors, of which eight were differentially active. Seven of these eight factors were true positives for MOTL factorizations, six with high frequency (i.e., identified for more than 80% of the $\boldsymbol{T}$ target datasets). Only one differentially active groundtruth factor is frequently a true positive for direct MOFA factorization of the $\boldsymbol{T}$ target datasets. Factor 2 is related to B cell receptor signaling and Fc Receptor mediated immune processes, drug metabolism by cytochrome p450 and other metabolism-related processes. This Factor 2 is rarely uncovered by direct MOFA factorization. Contrarily, Factor 4 is a true positive for MOTL for 100% of the target datasets and for direct MOFA factorizations for more than 75% of the target datasets. This factor is associated with developmental processes and cytokine-cytokine receptor interactions. Factor 7 is associated with keratinisation and formation of the cornified envelope, Factor 9 with cell adhesion and migration, and Factor 10 with cytokine and chemokine signaling.

The factorization of the $\boldsymbol{R}$ dataset containing all LAML, PAAD and SKCM samples yielded 13 groundtruth factors, of which seven were differentially active. All seven of these differentially active groundtruth factors were true positives for MOTL factorization with high frequency, compared to one factor for direct MOFA factorization. These groundtruth factors, differentially active between all three cancer types, are associated with the same cellular processes and pathways identified when comparing the cancer types pairwise (Supplementary Table 1).

Overall, the factors that were differentially active when comparing two or the three cancer types reflect the different embryonic origins of the cancerous tissues, and highlight the importance of immunity and microenviroment in cancer pathophysiology and response to treatments. In conclusion, matrix factorization with transfer learning using MOTL better uncovers differentially active groundtruth factors from target datasets containing only a small number of samples.

2.4 Application of MOTL to glioblastoma

Glioblastoma is a rare, heterogeneous, and aggressive cancer type. Multi-omics datasets offer an important opportunity to better characterize glioblastoma subtypes, identify biomarkers, and propose novel therapeutic options (Santamarina-Ojeda et al., 2023). However, large collections of glioblastoma tissue samples are difficult to obtain due to the relative scarcity of the disease and the challenges involved in acquiring samples via invasive biopsies.

In Santamarina-Ojeda et al. (2023), the authors conducted a multi-omics profiling (mRNA expression, DNA methylation) for four normal brain samples and nine patient-derived glioblastoma stem cell (pd-GBSC) cultures. The nine cancer samples had been previously classified into three subtypes thanks to transcriptome-based signatures: classical (CL), proneural (PN), and mesenchymal (MS). Given the small number of samples, the authors devised a strategy based on analyzing this dataset in parallel with datasets gathered from the literature. We illustrate here how MOTL could help in analysing such a dataset comprised of a limited number of samples.

We first applied a direct MOFA factorization (Methods 4.6) to a target dataset comprised of the four normal and nine pd-GBSC samples (Methods 4.5), revealing eight factors. Heatmap clustering of the samples, based on these factors, does not demonstrate clear grouping with respect to either cancer status or subtype (Figure 4a). Next, we applied MOTL to the same target dataset (Methods 4.7). In this case, we first created a TCGA learning dataset containing mRNA expression, miRNA expression, DNA methylation, and SNV data for samples from all 32 cancer types (Methods 4.4). It is noteworthy that this learning dataset did not contain data for glioblastoma, as there were no TCGA glioblastoma samples fulfilling our selection criteria (i.e., with complete 4-layer multi-omics profiles). We factorized this learning dataset with MOFA (Methods 4.6), based on which we applied MOTL to the target dataset. MOTL transfer learning factorization revealed 19 factors. In this case, the heatmap clustering of the MOTL factors separates cancer and normal samples and also displays subgroups partially matching subtypes previously defined based on transcriptome signatures (Figure 4b).

Further statistical tests revealed that only one of the eight factors identified by direct MOFA factorization was differentially active between normal and cancer samples, whereas 12 of the 19 factors obtained by MOTL transfer learning factorization were differentially active (Methods 4.8). Gene set enrichment analysis, focusing on differentially active factors that explained at least 1% of mRNA variance, revealed 715 processes and pathways associated with the direct MOFA factor, and 1061 processes and pathways associated with the 12 MOTL factors. The overlap between the two sets of associated processes and pathways was 318, which is statistically significant (hypergeometric test, one-sided p-value < 0.01, Supplementary Table 3).

We also applied both direct MOFA, and MOTL factorizations, to target datasets comprised of normal samples and samples from just a single cancer subtype. In all cases, the direct MOFA factorization yielded only one differentially active factor, whereas MOTL yielded 11 (CL vs normal), 16 (MS vs normal), and 12 (PN vs normal) differentially active factors. Focusing on the subset of differentially active factors that explained at least 1% of mRNA variance, we performed gene set enrichment analyses. We identified processes and pathways that were associated with only a single cancer subtype, such as fatty acid metabolism enrichment for the MS subtype and clathrin-mediated endocytosis for the PN subtype (Supplementary Table 4).

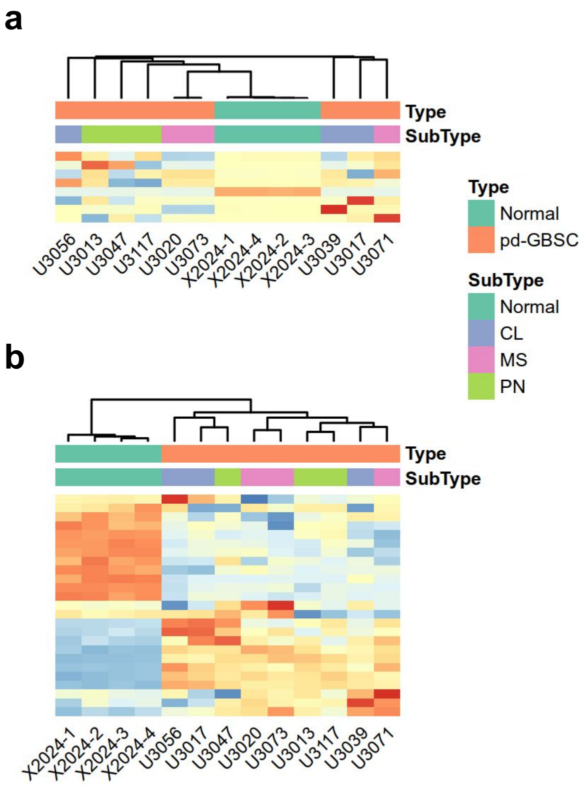

Figure 4: Heatmaps of factorizations of glioblastoma data: Each heatmap is based on the score matrix, $\boldsymbol{Z}$, inferred with a factorization of the target dataset comprised of multi-omics data (mRNA expression, DNA methylation) for all normal and patient-derived glioblastoma stem cell (pd-GBSC) samples. The multi-omics data was obtained from Santamarina-Ojeda et al., who also provided subtypes for the cancer samples, previously defined from transcriptomics signatures: CN (classical), PN (proneural), MS (mesenchymal). The rows of each heatmap are the factors, the columns are the samples, and the cells are the row-wise centered and scaled factor values. The rows and columns have been order with hierarchical clustering (complete-linkage). **a** Heatmap of $\boldsymbol{Z}$ matrix inferred with Direct MOFA factorization of the target dataset. **b** Heatmap of $\boldsymbol{Z}$ matrix inferred with MOTL factorization of the target dataset.

3 Discussion

We presented MOTL, an approach for multi-omics matrix factorization with transfer learning, which infers latent factor values for a multi-omics target dataset comprised of a small number of samples. MOTL factorizes the target dataset by incorporating latent factor values already inferred from the factorization of a learning dataset. We designed two protocols, based on simulated and real multi-omics datasets, for evaluating the performance of multi-omics matrix factorization with transfer learning. We implemented these protocols to evaluate MOTL, and observed that MOTL was more effective in uncovering differentially active groundtruth latent factors than direct matrix factorization without transfer learning. In addition, in the evaluation protocol based on TCGA data, the factors identified by MOTL are associated with biological processes and pathways consistent with the disease and tissue types represented by the target datasets. Finally, the application of MOTL to a glioblastoma dataset, comprised of a small number of samples, revealed an enhanced delineation of cancer status and subtype thanks to transfer learning. We thus demonstrated, in the case of a multi-omics dataset comprised of a small number of samples, that MOTL can enhance the discovery of biological processes and pathways associated with a biological condition of interest.

In the work presented here, we are concerned with the situation in which we wish to analyse a target dataset comprised of a limited number of samples. In our evaluations, the target datasets never contained data for more than 15 samples, as our primary concern was with target datasets considered too small for useful factor analysis. It would be insightful to extend the evaluations by using a larger range of sizes for target datasets, in order to identify a crossover point at which transfer learning no longer enhances matrix factorization. In addition, with target datasets containing few samples, the learning dataset is likely to represent a set of biological conditions that does not fully overlap with the set of biological conditions represented by the target dataset. It would be relevant to investigate how similar the learning and target datasets need to be in order for shared factors to exist. For instance, it would be beneficial to use a measure of similarity, between a given learning and target dataset, which would predict the effectiveness of using a transfer learning approach to apply matrix factorization to the target dataset. It could also be interesting to quantify how heterogeneous (i.e., representing a large diversity of tissues, diseases, experimental conditions ...) a learning dataset needs to be, in order to yield factors which can be relevant for a given target dataset.

To evaluate MOTL, we designed two evaluation protocols, based on simulated and real data. Importantly, these protocols can be reused to evaluate other transfer learning strategies for multi-omics data integration. For the evaluation protocol based on simulated datasets, we created groups of samples, and distinguished between groups by assigning different mean parameters when simulating sample scores. We intended to simulate the situation in which the factors are biological processes whose activity varies across different biological conditions. We used the same set of feature weights for all groups of samples, meaning they shared the same underlying latent factors, albeit with varying levels of activity. A future extension to this would be to make some factors completely inactive for the learning dataset. We could then evaluate how well MOTL is able to uncover groundtruth latent structure, for the target dataset, when not all latent factors are shared by both datasets. For the evaluation protocol based on real data, we used TCGA, as it is a public repository of multi-omics data with a large number of samples, representing various cancer and tissue types. We selected three different cancer types as references from which to build target datasets, and did not include samples from these cancer types in the learning dataset. In this setting, it was unknown, prior to evaluation, whether there were latent factors common to the learning and target datasets. Yet MOTL was effective in uncovering differentially active latent factors, demonstrating that latent factors can be shared across different cancer types. We envisage MOTL as being a helpful tool in the study of rare diseases in general. Therefore a future extension of our work would be to evaluate the application of MOTL to a target dataset with non-cancer rare disease samples, using factors inferred from the TCGA learning dataset.

With MOTL we have designed a transfer learning framework that is compatible with a prior learning dataset factorization carried out with the MOFA Bayesian approach (Argelaguet et al., 2018). The appeal of a Bayesian framework is the flexibility with regard to the incorporation of prior information, and variational inference serves as a fast alternative to sampling methods. In the future, MOTL could be extended to allow information to be incorporated at other levels of the assumed hierarchy of latent variables. For example, instead of fixing the feature weight values, they could be treated as random variables by MOTL, with priors informed by the factorization of the learning dataset.

In addition to MOFA, there are numerous methods available for multi-omics matrix factorization (Chalise and Fridley, 2017; Lock et al., 2013; Rohart et al., 2017; Mo et al., 2018), meaning a future extension of our work could be a transfer learning framework matched to a different multi-omics matrix factorization method. Similarly, we foresee value in extending an existing transfer learning method that has been designed for single omics data (Sharma et al., 2020; Taroni et al., 2019), so it can be applied in the multi-omics context. The evaluation of such a method, based on the factorization of a learning dataset with a variety of matrix factorization methods, would be informative.

Finally, a limitation with MOTL is that we are restricted to a factorization based on features that were retained for factorization of the learning dataset. A consequence is that some features which are highly variable in the target dataset may not contribute to the MOTL factorization. Therefore a future extension could be to add flexibility into the MOTL workflow, so that all features that are highly variable in the target dataset contribute to the factorization, even if they were not retained for the factorization of the learning dataset.

4 Methods

4.1 Mathematical Notation

- We denote matrices and datasets with bold capital letters: $\boldsymbol{Y}$
- If $\boldsymbol{Y}$ denotes a matrix, we introduce it as $\boldsymbol{Y} = [y_{nd}] \in \mathbb{R}^{N \times D}$ for which:
  - there are N rows and D columns
  - y_{nd} denotes the value in the nth row and the dth column
  - $\boldsymbol{y}_{n:}$ denotes the nth row vector, and $\boldsymbol{y}_{:d}$ denotes the dth column vector
- If $\boldsymbol{Y}$ denotes a dataset comprised of multiple matrices, we specify this, and denote each of the matrices as $\boldsymbol{Y}^{(m)} = \left[y_{nd}^{(m)}\right] \in \mathbb{R}^{N \times D_m}$
- We denote parameters for statistical distributions as non-bold, lower case letters. If the parameter is for a random variable stored in a matrix, we add indices. For example, $\tau_{nd}^{(m)}$ is a parameter for a random variable in the nth row and dth column of the mth matrix of some dataset. If $\tau_{d}^{(m)}$ is a parameter for the same matrix, then it is used for all values in the dth column.

4.2 The MOFA model

Consider a multi-omics dataset $\boldsymbol{Y}$ consisting of omics matrices $\boldsymbol{Y}^{(m)}$, $m = 1, ..., M$. Each $\boldsymbol{Y}^{(m)} = \left[y_{nd}^{(m)}\right] \in \mathbb{R}^{N \times D_m}$ contains data for N samples (rows) and D_m features (columns), where $y_{nd}^{(m)}$ is the value for the nth sample and the dth feature from the mth matrix. MOFA (Argelaguet et al., 2020), assumes the existence of latent factors, and jointly factorizes each $\boldsymbol{Y}^{(m)}$ into a shared matrix of sample scores $\boldsymbol{Z} = [z_{nk}] \in \mathbb{R}^{N \times K}$, and an omics specific matrix of feature weights $\boldsymbol{W}^{(m)} = \left[w_{kd}^{(m)}\right] \in \mathbb{R}^{K\times D_m}$. The kth column of $\boldsymbol{Z}$ contains scores for the kth factor, and the kth row of $\boldsymbol{W}^{(m)}$ contains corresponding weights for that factor.

MOFA assumes that each observed $y_{nd}^{(m)}$ is a random variable, characterised by a probability distribution conditional on a set of latent random variables $\boldsymbol{\beta}$. It is assumed that the joint likelihood $p(\boldsymbol{Y}|\boldsymbol{\beta})$ is equal to $\prod_{m=1}^{M}\prod_{n=1}^{N}\prod_{d=1}^{D_m} p(y_{nd}^{(m)} \mid \boldsymbol{\beta})$, and the choice of probability distribution depends on the type of observed data. A Gaussian likelihood is assumed for observed continuous data:

$$p(y_{nd}^{(m)} \mid \boldsymbol{\beta}) = \text{Normal}\left(y_{nd}^{(m)} \mid \boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(m)}, 1/\tau_d^{(m)}\right) \tag{7}$$

where $\boldsymbol{z}_{n:}$ is the vector of scores for the nth sample, $\boldsymbol{w}_{:d}^{(m)}$ is the vector of weights the dth feature from the mth matrix, and $\tau_d^{(m)}$ is the precision for that feature. A Bernoulli likelihood is assumed for observed binary data and the logistic link function $\pi(x) = (1+e^{-x})^{-1}$ is used:

$$p(y_{nd}^{(m)} \mid \boldsymbol{\beta}) = \text{Bernoulli}\left(y_{nd}^{(m)} \mid \pi\left(\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(m)}\right)\right) \tag{8}$$

A Poisson likelihood is assumed for observed counts data and the link function $\lambda(x) = \log(1+e^{x})$ is used:

$$p(y_{nd}^{(m)} \mid \boldsymbol{\beta}) = \text{Poisson}\left(y_{nd}^{(m)} \mid \lambda\left(\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(m)}\right)\right) \tag{9}$$

The assumed joint prior distribution, $p(\boldsymbol{\beta})$, is comprised of independent priors: $z_{nk} \sim \text{Normal}(0,\ 1)$, $w_{kd}^{(m)} = \hat{w}_{kd}^{(m)} s_{kd}^{(m)}$, $\hat{w}_{kd}^{(m)} \sim \text{Normal}(0,\ 1/\alpha_k^{(m)})$, $\alpha_k^{(m)} \sim \text{Gamma}(1e^{-14},\ 1e^{-14})$, $s_{kd}^{(m)} \sim \text{Bernoulli}(\theta_k^{(m)})$, $\theta_k^{(m)} \sim \text{Beta}(1,\ 1)$, $\tau_d^{(m)} \sim \text{Gamma}(1e^{-14},\ 1e^{-14})$.

MOFA uses mean-field variational inference (Grimmer, 2011; Fox and Roberts, 2012; Blei et al., 2017) to approximate the joint posterior distribution, $p(\boldsymbol{\beta}|\boldsymbol{Y})$, with a joint variational distribution factorized over J disjoint groups of variables:

$$\begin{aligned} q(\boldsymbol{\beta}) &= \prod_{j=1}^{J} q(\boldsymbol{\beta}_j) \\ &= \prod_{n=1}^{N}\prod_{k=1}^{K} q(z_{nk}) \prod_{m=1}^{M}\prod_{k=1}^{K} q(\alpha_k^{(m)})\ q(\theta_k^{(m)}) \prod_{m=1}^{M}\prod_{d=1}^{D_m} q(\tau_d^{(m)}) \prod_{m=1}^{M}\prod_{d=1}^{D_m}\prod_{k=1}^{K} q(\hat{w}_{kd}^{(m)}, s_{kd}^{(m)}) \end{aligned} \tag{10}$$

MOFA infers $q(\boldsymbol{\beta})$ iteratively until convergence. At each iteration, each $q(\boldsymbol{\beta}_j)$ is updated as

$$q(\boldsymbol{\beta}_j) \propto exp\{\mathbb{E}_{q_{-j}}[\log\ p(\boldsymbol{\beta}, \hat{\boldsymbol{Y}})]\} \tag{11}$$

where $\mathbb{E}_{q_{-j}}$ denotes an expectation with respect to the joint variational distribution, after removing $q(\boldsymbol{\beta}_j)$. The dataset $\hat{\boldsymbol{Y}}$ is derived by transformation of $\boldsymbol{Y}$. Observed data with a Gaussian assumed likelihood are transformed with feature-wise centering, which avoids the need to estimate intercepts. Observed data with a non-Gaussian assumed likelihood are transformed to derive Gaussian pseudo-data. The derivation of Gaussian pseudo-data occurs at each iteration, and is based on a new parameter, $\zeta_{nd}^{(m)}$, which is derived for each sample n and feature d from matrix m. For observed data with a Bernoulli assumed likelihood, a precision parameter, $\tau_{nd}^{(m)}$, is introduced for each sample and feature as part of the transformation:

$$\hat{y}_{nd}^{(m)} = \frac{2y_{nd}^{(m)} - 1}{2\tau_{nd}^{(m)}} \tag{12}$$

$$\tau_{nd}^{(m)} = 2\lambda\left(\zeta_{nd}^{(m)}\right) \tag{13}$$

$$\left(\zeta_{nd}^{(m)}\right)^2 = \mathbb{E}_q\left[\left(\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(m)}\right)^2\right] \tag{14}$$

For observed data with a Poisson assumed likelihood, a precision parameter, $\tau_d^{(m)}$, is introduced for each feature as part of the transformation:

$$\hat{y}_{nd}^{(m)} = \zeta_{nd}^{(m)} - \frac{\pi\left(\zeta_{nd}^{(m)}\right)\left(1 - y_{nd}^{(m)}/\lambda\left(\zeta_{nd}^{(m)}\right)\right)}{\tau_d^{(m)}} \tag{15}$$

$$\zeta_{nd}^{(m)} = \mathbb{E}_q\left[\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(m)}\right] \tag{16}$$

$$\tau_d^{(m)} = 0.25 + 0.17 \times max\left(\boldsymbol{y}_{:d}^{(m)}\right) \tag{17}$$

where $\boldsymbol{y}_{:d}^{(m)}$ is the vector of observed values for the dth feature from the mth matrix. For both Bernoulli and Poisson observed data, the $\hat{y}_{nd}^{(m)}$ values are centered at each iteration, and $\zeta_{nd}^{(m)}$ values are derived using the factorization fit from the preceding iteration. MOFA monitors convergence with the evidence lower bound (ELBO), which is used to evaluate how well the variational distribution approximates the posterior distribution. The ELBO is calculated as:

$$\text{ELBO}(\boldsymbol{\beta}) = \mathbb{E}_q\left[\log p(\boldsymbol{Y}|\boldsymbol{\beta})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{\beta})\right] - \mathbb{E}_q\left[\log q(\boldsymbol{\beta})\right] \tag{18}$$

For $\boldsymbol{Y}^{(m)}$ with non-Gaussian assumed likelihood, MOFA uses a lower bound for each $\log p\left(y_{nd}^{(m)}|\boldsymbol{\beta}\right)$. Maximizing this lower bound, coupled with the use of $\hat{y}_{nd}^{(m)}$ values, allows updates of $q(\boldsymbol{\beta})$ based on the assumption of Gaussian observed data (Jaakkola and Jordan, 2000; Seeger and Bouchard, 2012). MOFA assesses convergence at regular intervals, based on the percentage change in ELBO after. MOFA allows factors to be dropped during training, based on the fraction of variance explained for each matrix. After each iteration, MOFA identifies factors that do not explain a fraction of variance, for any omics matrix, over a threshold. MOFA then drops one of the identified factors.

4.3 Multi-omics data simulated with groundtruth factors

We simulated multi-omics datasets, from groundtruth factors, with various configurations. For each simulation configuration we generated 30 instances of a multi-omics dataset, $\boldsymbol{Y}$, consisting of matrices, $\boldsymbol{Y}^{(m)}, m = 1, 2, 3$. We split each $\boldsymbol{Y}$ into a target dataset, $\boldsymbol{T}$, and a learning dataset, $\boldsymbol{L}$. Each $\boldsymbol{Y}^{(m)} = \left[y_{nd}^{(m)}\right] \in \mathbb{R}^{N \times D_m}$ contained data for $N = N_t + N_l$ samples (rows) and $D_m = 2000$ features (columns), where $y_{nd}^{(m)}$ is the value for the nth sample and the dth feature from the mth matrix. N_t is the number of samples subsequently belonging to $\boldsymbol{T}$, and N_l is the number of samples belonging to $\boldsymbol{L}$. We generated each $\boldsymbol{Y}^{(m)}$ from a different statistical distribution, conditional on a random matrix of sample scores, $\boldsymbol{Z} = [z_{nk}] \in \mathbb{R}^{N \times K}$, and a random matrix of feature weights, $\boldsymbol{W}^{(m)} = \left[w_{kd}^{(m)}\right] \in \mathbb{R}^{K \times D_m}$. The kth column vector of $\boldsymbol{Z}$ contained sample scores for the kth groundtruth factor. The kth row vector of $\boldsymbol{W}^{(m)}$ contained feature weights for that same factor. We varied the number of groundtruth factors across configurations, using $K \in \{20, 30\}$.

We generated $\boldsymbol{Z}$ based on each sample being a member of a group. In each instance we created two groups of five samples belonging to $\boldsymbol{T}$, meaning N_t was always equal to 10 samples. We allowed N_l to vary across instances, with samples belonging to $\boldsymbol{L}$ being in differently sized groups of randomly selected sizes. We used either 20 learning groups of size $\in \{10, 20, 30\}$, or 40 groups of size $\in \{10, 25, 40\}$. For the nth sample, and kth groundtruth factor, we generated the score as $z_{nk} \sim \text{Normal}(\mu_{g(n)k},\ \sigma_z)$, where $\mu_{g(n)k}$ is the mean parameter for groundtruth factor k, for the group that sample n belonged to, $g(n)$. In each instance we selected $\mu_{g(n)k}$ randomly for each group and groundtruth factor, with probabilities $Pr(3) = 1/8$, $Pr(5) = 3/4$, $Pr(7) = 1/8$. The kth groundtruth factor was differentially active for $\boldsymbol{T}$ if $\mu_{g(n)k}$ differed between the two target dataset groups. For all instances of a given simulation configuration, the same standard deviation parameter, σ_z, was shared by all groups and groundtruth factors. We varied the latent noise-to-signal ratio across our simulation configurations by using $\sigma_z \in \{0.5, 1.0\}$

For the kth groundtruth factor, and the dth feature from the mth matrix, we generated the weight as $w_{kd}^{(m)} = \hat{w}_{kd}^{(m)} s_{kd}^{(m)}$. As such, each $w_{kd}^{(m)}$ was the product of a continuous random variable, $\hat{w}_{kd}^{(m)} \sim \text{Normal}(\mu^{(m)},\ \sigma_k^{(m)})$, and a binary random variable, $s_{kd}^{(m)} \sim \text{Bernoulli}(\theta_k^{(m)})$. We specified $\mu^{(m)}$, the mean parameter for the mth matrix, with $\mu^{(1)} = 5$; $\mu^{(2)} = 0$; $\mu^{(3)} = 0$. We generated , $\sigma_k^{(m)}$, the standard deviation parameter for the kth groundtruth factor and the mth matrix, with $\sigma_k^{(1)} \sim \text{Uniform}(0.5, 1.5)$; $\sigma_k^{(2)} \sim \text{Uniform}(0.5, 1.5)$; $\sigma_k^{(3)} \sim \text{Uniform}(0.1, 0.2)$. We generated the sparsity for the kth groundtruth factor and the mth matrix, $1 - \theta_k^{(m)}$, with $\theta_k^{(m)} \sim \text{Uniform}(0.15, 0.25)$.

We generated the values in each $\boldsymbol{Y}^{(m)}$ as:

$$\begin{aligned}
y_{nd}^{(1)} &\sim \text{Poisson}\left(\log\left(1 + \exp\left(\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(1)}\right)\right)\right) \\
y_{nd}^{(2)} &\sim \text{Normal}\left(\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(2)},\ \sigma_d \sim \text{Uniform}\left(0.25, 0.75\right)\right) \\
y_{nd}^{(3)} &\sim \text{Bernoulli}\left(1/\left(1 + \exp\left(-\boldsymbol{z}_{n:}\boldsymbol{w}_{:d}^{(3)}\right)\right)\right)
\end{aligned} \tag{19}$$

where $\boldsymbol{z}_{n:}$ is the vector of scores for the nth sample, $\boldsymbol{w}_{:d}^{(m)}$ is the vector of weights the dth feature from the mth matrix, and σ_d is the standard deviation for the dth feature.

We split each $\boldsymbol{Y}^{(m)}$ into $\boldsymbol{T}^{(m)} = \left[t_{nd}^{(m)}\right] \in \mathbb{R}^{N_t \times D_m}$, which contained values for the target group samples, and $\boldsymbol{L}^{(m)} = \left[l_{nd}^{(m)}\right] \in \mathbb{R}^{N_l \times D_m}$, which contained values for the learning group samples.

Before direct factorization with MOFA we pre-processed simulated $\boldsymbol{T}$ and $\boldsymbol{L}$ datasets by removing features with 0 variance across samples. Before factorization with transfer learning with MOTL, we pre-processed simulated $\boldsymbol{T}$ datasets by removing features that had 0 variance across samples or that had been removed from the corresponding $\boldsymbol{L}$ datasets.

4.4 TCGA multi-omics data acquisition and pre-processing

We used the R packages *TCGAbiolinks* (v.2.25.3) and *SummarizedExperiment* (v.1.28.0) to download and save TCGA mRNA expression, miRNA expression, DNA methylation, and single nucleotide variation (SNV) data (Silva et al., 2016; Hutter and Zenklusen, 2018; Mounir et al., 2019). The mRNA and miRNA expression data consisted of raw counts. The DNA methylation data consisted of CpG site β-values, which had been derived from HM450 array intensities with R package *SeSAMe* (v.1.16.0) (Zhou et al., 2018). The SNV data consisted of masked somatic mutation files.

We created four reference datasets, using data from three cancer types; acute myeloid leukemia (LAML), pancreatic adenocarcinoma (PAAD) and skin cutaneous melanoma (SKCM). Each reference dataset, $\boldsymbol{R}$, contained multi-omics data for all samples from either two, or all three of the cancer types. We did not include SNV data in $\boldsymbol{R}$ datasets containing LAML samples, due to the sparsity of SNV data for LAML. We only used samples that had data for all omics of interest, and only included one sample per study participant. We thus had multi-omics data for 134 LAML samples, 157 PAAD samples and 435 SKCM samples. We then randomly split each $\boldsymbol{R}$ into non-overlapping target datasets. Each resulting target dataset, $\boldsymbol{T}$, contained multi-omics data for five samples per cancer type.

For the evaluation protocol based on TCGA multi-omics data, we merged data from the remaining 29 cancer types into a learning dataset, $\boldsymbol{L}$. For this $\boldsymbol{L}$ we only used samples that had data for all four omics, and only included one sample per study participant. This $\boldsymbol{L}$ contained multi-omics data (mRNA, miRNA, DNA methylation and SNV) for 7,217 samples.

For the application of MOTL to the pd-GBSC target datasets, we created a new learning dataset by merging data from all 32 cancer types. This new learning dataset contained multi-omics data (mRNA, miRNA, DNA methylation and SNV) for 7,866 samples.

Before direct factorization with MOFA, we pre-processed $\boldsymbol{R}$, $\boldsymbol{T}$ and $\boldsymbol{L}$ datasets in the same way. For mRNA data we removed genes that map to the Y chromosome. For both mRNA and miRNA we removed genes if they had a count of zero in $\geq 90\%$ of samples, or had zero variance across samples. We normalized mRNA and miRNA counts with the *DESeq2* (v.1.38.0) R package (Love et al., 2014), and $\log_2(x+1)$ transformed the normalized counts. For DNA methylation data, we removed CpG sites that map to the X or Y chromosome, were masked during SeSAMe quality control, had missing values in $\geq 20\%$ of samples, or had zero variance across samples. We converted DNA methylation β-values to M-values (Du et al., 2010). We included SNV records whose variant classification was either *Frame_Shift_Del*, *Frame_Shift_Ins*, *In_Frame_Del*, *In_Frame_Ins*, *Missense_Mutation*, *Nonsense_Mutation*, *Nonstop_Mutation*, *Splice_Site* or *Translation_Start_Site*. We then created binary SNV matrices aggregated by gene and sample. We removed genes from SNV matrices if the mutation rate across samples was $\leq 1\%$. We filtered all omics to include only the 5,000 most variable features. We did not perform any batch effect correction on $\boldsymbol{L}$ datasets in order to preserve biological signal (Lee et al., 2020). We checked each $\boldsymbol{R}$ for batch effects with visualizations of UMAP co-ordinates (McInnes et al., 2020). We used the R package *uwot* (v.0.1.14) to derive UMAP coordinates from MOFA factorizations, and we did not observe the need to correct $\boldsymbol{R}$ datasets for batch effects.

Before factorization with transfer learning with MOTL, we pre-processed $\boldsymbol{T}$ datasets by removing all omics features that had zero variance across samples, or that had been removed from $\boldsymbol{L}$ during pre-processing. We used *DESeq2* to normalize mRNA and miRNA counts with the geometric means from $\boldsymbol{L}$, and then $\log_2(x+1)$ transformed the normalized counts. We converted DNA methlyation β-values to M-values, and converted SNV data to binary matrices after filtering on variant classification, as described previously.

4.5 Glioblastoma target dataset acquisition and pre-processing

We created four pd-GBSC target datasets, based on multi-omics profiling conducted by Santamarina-Ojeda et al. for four normal brain samples and nine patient-derived glioblastoma stem cell (pd-GBSC) cultures. The nine cancer samples had been previously classified into three subtypes thanks to transcriptome-based signatures: classical (CL), proneural (PN), and mesenchymal (MS). Each pd-GBSC target dataset contained mRNA expression and DNA methylation data for the four normal brain cortex samples, as well as either all nine cancer samples or just the samples from a subtype.

For factorization with transfer learning with MOTL, the pd-GBSC target datasets initially consisted of mRNA expression raw counts and DNA methylation β-values. Before factorization with MOTL, we pre-processed a pd-GBSC target dataset by removing all omics features that had zero variance across samples, or that had been removed from $\boldsymbol{L}$ during pre-processing. We used *DESeq2* to normalize mRNA counts with the geometric means from $\boldsymbol{L}$, and then $\log_2(x+1)$ transformed the normalized counts. We converted DNA methlyation β-values to M-values.

For direct MOFA factorization, without transfer learning, the pd-GBSC target datasets initially consisted of the same mRNA expression data, but already normalized and transformed by Santamarina-Ojeda et al., and DNA methylation β-values. Before direct MOFA factorization, we pre-processed mRNA data by removing genes that map to the Y chromosome, if they had a count of zero in $\geq 90\%$ of samples, or had zero variance across samples. We pre-processed DNA methylation data by removing CpG sites that had missing values in $\geq 20\%$ of samples, or had zero variance across samples. We converted DNA methylation β-values to M-values. We filtered both omics to include only the 5,000 most variable features.

4.6 Application of MOFA to simulated, TCGA and glioblastoma multi-omics datasets

We factorized simulated target, $\boldsymbol{T}$, and learning, $\boldsymbol{L}$, datasets with the MOFA Python implementation *mofapy2* (v.0.6.4). The number of factors we used for each MOFA factorization was equal to the lesser of the number of samples and the number of groundtruth factors that were differentially active when simulating the dataset. The kth groundtruth factor was differentially active for a dataset if the mean parameter, $\mu_{g(n)k}$, for the sample scores for that factor, was not the same for all groups of samples in the dataset. We specified observed data likelihoods corresponding to those used for simulating $\boldsymbol{Y}^{(m)}$ matrices. We set the maximum number of iterations to 10,000 to ensure convergence. For the remaining settings we used the *mofapy2* defaults, meaning that all datasets were feature-wise centered during factorization fitting.

We factorized pre-processed TCGA reference, $\boldsymbol{R}$, target, $\boldsymbol{T}$, and learning, $\boldsymbol{L}$, datasets with the MOFA Python implementation *mofapy2* (v.0.7.0). We specified Gaussian as the observed data likelihood for mRNA, miRNA and DNA methylation data, and specified Bernoulli as the likelihood for SNV data. For the $\boldsymbol{L}$ datasets, we started the factorization with 100 factors and allowed factors to be dropped based on the fraction of variance explained, for which we set the threshold to 0.001. We set the threshold so low in order to retain factors that explained little of the variance in $\boldsymbol{L}$, yet could be potentially relevant for transfer learning. For $\boldsymbol{R}$ datasets we also started with 100 factors. For $\boldsymbol{T}$ datasets, we started with the maximum number of factors allowed by MOFA, which was either 10 factors (two cancer types) or 15 factors (three cancer types). For $\boldsymbol{R}$ and $\boldsymbol{T}$ datasets we dropped factors based on a threshold of 0.01, in order to only retain relevant factors. For all TCGA datasets, we set the maximum number of iterations to 10,000, to ensure convergence, and the frequency of convergence checking to five, to ensure that the algorithm had stopped dropping factors before converging.

When saving the factorizations of simulated and TCGA $\boldsymbol{L}$ datasets, we set the expectations argument to *all*. We did this to ensure that the point estimate for each precision parameter was saved in addition to those that are saved by default.

We factorized the pre-processed pd-GBSC target datasets with the MOFA Python implementation *mofapy2* (v.0.7.0). We specified Gaussian as the observed data likelihood for the mRNA and the DNA methylation data. We started with the maximum allowable number of factors and dropped factors based on a threshold of 0.01.

4.7 Application of MOTL to simulated, TCGA and glioblastoma multi-omics datasets

We applied MOTL to simulated, TCGA and pd-GBSC multi-omics target datasets. For each target dataset, we used point estimates of feature weight and precision values saved from the MOFA factorization of the corresponding learning, $\boldsymbol{L}$, dataset. For observed data with a Gaussian or Poisson assumed likelihood, the transferred value of the precision for each feature, $\tau_d^{(m)}$, was held fixed throughout iterations of MOTL updates. For observed data with a Bernoulli assumed likelihood, we initialized the value of the precision for each sample and feature, $\tau_{nd}^{(m)}$, with a feature-wise average, $\tau_d^{(m)}$, of the $\tau_{nd}^{(m)}$ values from the factorization of $\boldsymbol{L}$. The precisions for Bernoulli observed data were then iteratively updated by MOTL. We estimated intercepts using likelihoods assumed for $\boldsymbol{L}$, combined with outputs from the MOFA factorization of $\boldsymbol{L}$. For Gaussian observed data we calculated the intercept for the dth feature, from the mth matrix, as $a_d^{(m)} = \frac{1}{N_l}\sum_{n=1}^{N_l} l_{nd}^{(m)}$, where $l_{nd}^{(m)}$ denotes an uncentered learning dataset value after pre-processing. For Poisson and Bernoulli observed data we obtained maximum likelihood estimates of $a_d^{(m)}$ values, for which we used the *mle* function from the R package *stats4* (v.4.2.0). For Poisson observed data we initialized each estimate with $a_d^{(m)} = \log\left(-1 + \exp\left(\frac{1}{N_l}\sum_{n=1}^{N_l} l_{nd}^{(m)}\right)\right)$, and for Bernoulli observed data we initialized it with $a_d^{(m)} = \log\left(\left(\frac{1}{N_l}\sum_{n=1}^{N_l} l_{nd}^{(m)}\right)\left(1 - \frac{1}{N_l}\sum_{n=1}^{N_l} l_{nd}^{(m)}\right)^{-1}\right)$.

When checking the ELBO for convergence, we used 0.0005% as the threshold, which is the default for MOFA. The algorithm was stopped when the absolute change in ELBO was under this threshold for two consecutive checks, and we set the maximum number of iterations to 10,000 to be consistent with our application of MOFA. For TCGA and pd-GBSC target datasets, we allowed factors to be dropped based on a threshold of 0.01 for the fraction of variance explained. We checked the ELBO after every five iterations, to ensure that the algorithm had stopped dropping factors before converging.

4.8 Evaluation methods

Groundtruth factors: For each simulated $\boldsymbol{T}$ (Methods 4.3), the groundtruth factor values were contained in the corresponding simulated $\boldsymbol{Z}$ and $\boldsymbol{W}^{(m)}$ matrices. The sample scores for the kth groundtruth factor were contained in $\boldsymbol{z}_{:k}$, the kth column vector of simulated $\boldsymbol{Z}$. The feature weights for the mth matrix, for that same groundtruth factor, were contained in $\boldsymbol{w}_{k:}^{(m)}$, the kth row vector of simulated $\boldsymbol{W}^{(m)}$. For each TCGA $\boldsymbol{T}$ (Methods 4.4), the groundtruth factors were based on the $\boldsymbol{R}$ dataset which we had split to create $\boldsymbol{T}$. We factorized each $\boldsymbol{R}$ with MOFA, and treated the inferred $\boldsymbol{z}_{:k}$ and $\boldsymbol{w}_{k:}^{(m)}$ vectors as groundtruth factors for each $\boldsymbol{T}$ that had been created by splitting $\boldsymbol{R}$.

Differentially active groundtruth factors: For each simulated and TCGA $\boldsymbol{T}$, groundtruth factor k was differentially active if the group means for groundtruth $\boldsymbol{z}_{:k}$ differed between the target dataset groups. For each simulated $\boldsymbol{T}$, this was the group mean, $\mu_{g(n)k}$, used to simulate groundtruth $\boldsymbol{z}_{:k}$. For each TCGA $\boldsymbol{T}$, the factorization of corresponding $\boldsymbol{R}$ provided groundtruth $\boldsymbol{z}_{:k}$ and $\boldsymbol{w}_{k:}^{(m)}$ factor vectors. We performed either the Wilcoxian Rank Sum test (two cancer types), or the Kruskal-Wallis test (three cancer types), on each groundtruth $\boldsymbol{z}_{:k}$ to determine if there was a statistically significant difference between the cancer types. We classed a groundtruth factor as differentially active if its BH-adjusted p-value was below 0.05.

Post-processing: We post-processed inferred and groundtruth $\boldsymbol{W}^{(m)}$ matrices before evaluation. We scaled each feature vector, $\boldsymbol{w}_{:d}^{(m)}$, by its Frobenius norm. We then centered each factor vector, $\boldsymbol{w}_{k:}^{(m)}$, of scaled values separately for each m. We then concatenated $\boldsymbol{w}_{k:}^{(m)}$ vectors to produce a single vector, $\boldsymbol{w}_{k:}$, of centered and scaled feature weights for each factor k.

Best hits: For each factorization of each simulated and TCGA $\boldsymbol{T}$, we identified the best hits between the factor vectors inferred with the factorization of $\boldsymbol{T}$, and the groundtruth factor vectors. For two sets of vectors $\{\boldsymbol{v}_1, ..., \boldsymbol{v}_{K_v}\}$ and $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_{K_x}\}$ we define the best hit for vector $\boldsymbol{v}_{k_v}$ as

$$\text{BestHit}\left(\boldsymbol{v}_{k_v}\right) = \underset{\boldsymbol{x}_{k_x}}{\arg\max}\ \text{cor}\left(\boldsymbol{v}_{k_v}, \boldsymbol{x}_{k_x}\right) \tag{20}$$

where $\text{cor}\left(\boldsymbol{v}, \boldsymbol{x}\right)$ is the Pearson correlation coefficient between vectors $\boldsymbol{v}$ and $\boldsymbol{x}$. We define the best hit for vector $\boldsymbol{x}_{k_x}$ as

$$\text{BestHit}\left(\boldsymbol{x}_{k_x}\right) = \underset{\boldsymbol{v}_{k_v}}{\arg\max}\ \text{cor}\left(\boldsymbol{x}_{k_x}, \boldsymbol{v}_{k_v}\right) \tag{21}$$

For each simulated $\boldsymbol{T}$, we identified best hits between inferred and groundtruth $\boldsymbol{w}_{k:}$ vectors. For each TCGA $\boldsymbol{T}$, we identified best hits between inferred and groundtruth $\boldsymbol{w}_{k:}$ vectors, as well as between inferred and groundtruth $\boldsymbol{z}_{:k}$ vectors. We used shared features when calculating correlations for $\boldsymbol{w}_{k:}$ vectors, and we used shared samples for $\boldsymbol{z}_{:k}$ vectors. We calculated p-values for the correlations, and only considered correlations with a p-value < 0.05 (two-sided alternative hypothesis) when identifying best hits.

F-measure values: For each factorization of each TCGA $\boldsymbol{T}$, we calculated an F-measure value to assess the overall correlation between factor vectors inferred with the factorization of $\boldsymbol{T}$, and groundtruth factor vectors. We based this on the F-measure presented by Saelens et al., which we adapted in order to assess correlations. For a given set of inferred factor vectors, $\{\boldsymbol{v}_1, ..., \boldsymbol{v}_{K_v}\}$, and a set of groundtruth factor vectors, $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_{K_x}\}$, we calculated the F-measure as

$$FM = 2/((1/Relevance) + (1/Recovery)) \tag{22}$$

where

$$Relevance = \frac{1}{K_v}\sum_{k_v=1}^{K_v} \text{cor}\left(\boldsymbol{v}_{k_v}, \text{BestHit}\left(\boldsymbol{v}_{k_v}\right)\right) \tag{23}$$

and

$$Recovery = \frac{1}{K_x}\sum_{k_x=1}^{K_x} \text{cor}\left(\boldsymbol{x}_{K_x}, \text{BestHit}\left(\boldsymbol{x}_{K_x}\right)\right) \tag{24}$$

Here $\text{cor}\left(\boldsymbol{v}, \boldsymbol{x}\right)$ is the Pearson correlation coefficient between vectors $\boldsymbol{v}$ and $\boldsymbol{x}$. We calculated F-measure values for sets of inferred and groundtruth $\boldsymbol{w}_{k:}$ vectors, as well as for $\boldsymbol{z}_{:k}$ vectors.

F1 scores: We calculated F1 scores to evaluate the factorizations of simulated and TCGA $\boldsymbol{T}$ datasets:

$$\begin{aligned}
&F1 = (2 \times Precision \times Recall)\ /\ (Precision + Recall)\\
&Precision = True\ Positives\ /\ Predicted\ Positives\\
&Recall = True\ Positives\ /\ Actual\ Positives
\end{aligned} \tag{25}$$

Actual Positives were the groundtruth factors of $\boldsymbol{T}$, that were differentially active.

Predicted Positives were the groundtruth factors that were predicted as being differentially active, based on the factorization of $\boldsymbol{T}$. We firstly performed either the Wilcoxian Rank Sum test (two groups), or the Kruskal-Wallis test (three groups), on each $\boldsymbol{z}_{:k}$ vector inferred with the factorization of $\boldsymbol{T}$, and classed factors with a p-value < 0.05 as differentially active. For inferred factors classed as differentially active, we identified the best hits for their corresponding inferred $\boldsymbol{w}_{k:}$ vectors. We selected these best hits from the set of groundtruth $\boldsymbol{w}_{k:}$ vectors for $\boldsymbol{T}$. If groundtruth $\boldsymbol{w}_{k:}$ was selected as a best hit for a differentially active inferred factor, then groundtruth factor k was predicted as being differentially active.

$True\ Positives$ were the differentially active groundtruth factors that were predicted as being differentially active, based on the factorization of $\boldsymbol{T}$.

Statistical testing of differences between factorization methods: We calculated the differences in evaluation measures between factorization methods, and tested the statistical significance of these differences. To do this we fit generalized least squares regressions with the R package *nlme* (v.3.1.157) (Pinheiro and Bates, 2000). We fit a single regression to model the F1 scores for simulated data. For TCGA data we fit a separate regression for each evaluation measure. For each regression we modelled $y_i = \beta_0 + \boldsymbol{d}_i\boldsymbol{\beta}_d + \boldsymbol{f}_i\boldsymbol{\beta}_f + \epsilon_i$. The vector $\boldsymbol{d}_i = (d_{i1}, ..., d_{iT})$ indicates the simulation configuration, or cancer type, that y_i relates to, and vector $\boldsymbol{f}_i = (f_{i1}, ..., f_{iM})$ indicates the factorization method. Vectors $\boldsymbol{\beta}_d = (\beta_{d1}, ..., \beta_{dT})^\top$ and $\boldsymbol{\beta}_f = (\beta_{f1}, ..., \beta_{fM})^\top$ are estimated fixed effects and ϵ_i is the residual. We incorporated correlations between residuals from the same target dataset using the compound symmetry structure method. We calculated contrasts for the factorization method effects in $\boldsymbol{\beta}_f$ using the R package *emmeans* (v.1.8.7) (Searle et al., 1980), and used Tukey-adjusted p-values for assessing statistical significance.

Differentially active factors from glioblastoma target datasets: We identified differentially active factors from the MOTL factorization of each pd-GBSC target dataset, as well as from the direct MOFA factorization (without transfer learning), of each pd-GBSCs target dataset. We performed the Wilcoxian Rank Sum test on each $\boldsymbol{z}_{:k}$ vector inferred with the factorization of a pd-GBSC target dataset. We classed factors with a BH-adjusted p-value < 0.05 as differentially active between the normal samples and the cancer samples.

Gene set enrichment analysis: We used R package *fgsea* (v.1.24.0) (Sergushichev, 2016) to perform gene set enrichment analysis on differentially active groundtruth factors that were true positives for factorizations of TCGA $\boldsymbol{T}$ datasets. For each differentially active groundtruth TCGA factor k, we analysed vector $\boldsymbol{w}_k^{(m)}$ if the fraction of mRNA variance explained by k was > 0.01, and where m corresponded to the mRNA matrix. We tested KEGG, REACTOME, GO:BP, and GO:MF gene sets that have a size of between 15 and 500 genes, obtained using the R package *msigdbr* (v.7.5.1). We used an BH-adjusted p-value cutoff of 0.01 for selecting enriched gene sets. We also performed gene set enrichment analysis on differentially active factors from the pd-GBSC target datasets, and used the same criteria as outlined above for differentially active groundtruth TCGA factors.

4.9 Implementation

An open source R implementation of MOTL, as well as code for reproducing these analyses, is available at `https://github.com/david-hirst/MOTL`. The factorization fit of the full TCGA learning dataset, used for the application of MOTL to the pd-GBSC target dataset, is available at `https://zenodo.org/doi/10.5281/zenodo.10847986`.

We used a Dell computer with 20 cores at 3GHz, and 64 GB of RAM, to perform factorizations. To pre-process and factorize the $\boldsymbol{L}$ used in the TCGA evaluation protocol, it took 26,405 seconds (over seven hours). Hence, we have made the factorization of a large

TCGA $\boldsymbol{L}$ dataset publicly available for transfer learning. It took an average of 37 seconds to pre-process a $\boldsymbol{T}$ dataset, comprised of four omics, and factorize it directly with MOFA. The average time increased to 134 seconds for MOTL.

5 Supplementary Information

Supplementary Table 1 contains gene sets associated with differentially active groundtruth TCGA factors.

Supplementary Table 2 contains the percentage of variance explained, by groundtruth TCGA factors, for each omics (MOFA factorizations of TCGA reference datasets).

Supplementary Table 3 contains gene sets associated with factors differentially active between all pd-GBSC samples and normal samples.

Supplementary Table 4 contains gene sets associated with factors differentially active between all pd-GBSC samples and normal samples, as well as those associated with factors differentially active between pd-GBSC subtype samples (CL, MS and PN) and normal samples.

6 Acknowledgements

We would like to thank Carl Herrmann, Céline Chevalier, Kim-Anh Lê Cao, Lionel Spinelli and Olivia Angelin-Bonnet for helpful feedback and discussions.

7 Funding

This research was supported by l'Agence Nationale de la Recherche (ANR), project ANR-21-CE45-0001-01, by France 2030 PEPR Digital Health managed by ANR, project ANR-22-PESN-0013, and by AFM (Association Française contre les Myopathies).

References

Argelaguet, R., Arnol, D., Bredikhin, D., Deloro, Y., Velten, B., Marioni, J. C., and Stegle, O. (2020). MOFA+: a statistical framework for comprehensive integration of multi-modal single-cell data. *Genome Biology*, 21(1):111.

Argelaguet, R., Velten, B., Arnol, D., Dietrich, S., Zenz, T., Marioni, J. C., Buettner, F., Huber, W., and Stegle, O. (2018). Multi-Omics Factor Analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular Systems Biology*, 14(6):e8124. Publisher: John Wiley & Sons, Ltd.

Banerjee, J., Taroni, J. N., Allaway, R. J., Prasad, D. V., Guinney, J., and Greene, C. (2023). Machine learning in rare disease. *Nature Methods*, pages 1–12. Publisher: Nature Publishing Group.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518):859–877. arXiv: 1601.00670.

Cantini, L., Zakeri, P., Hernandez, C., Naldi, A., Thieffry, D., Remy, E., and Baudot, A. (2021). Benchmarking joint multi-omics dimensionality reduction approaches for the study of cancer. *Nature Communications*, 12(1):124. Number: 1 Publisher: Nature Publishing Group.

Chalise, P. and Fridley, B. L. (2017). Integrative clustering of multi-level 'omic data based on non-negative matrix factorization algorithm. *PLOS ONE*, 12(5):e0176278. Publisher: Public Library of Science.

Chauvel, C., Novoloaca, A., Veyre, P., Reynier, F., and Becker, J. (2020). Evaluation of integrative clustering methods for the analysis of multi-omics data. *Briefings in Bioinformatics*, 21(2):541–552. Publisher: Oxford University Press / USA.

Collado-Torres, L., Nellore, A., Kammers, K., Ellis, S. E., Taub, M. A., Hansen, K. D., Jaffe, A. E., Langmead, B., and Leek, J. T. (2017). Reproducible RNA-seq analysis using recount2. *Nature Biotechnology*, 35(4):319–321. Number: 4 Publisher: Nature Publishing Group.

Conesa, A. and Beck, S. (2019). Making multi-omics data accessible to researchers. *Scientific Data*, 6(1):251. Bandiera_abtest: a Cc_license_type: cc_by Cg_type: Nature Research Journals Number: 1 Primary_atype: Comments & Opinion Publisher: Nature Publishing Group Subject_term: Data integration;Genome Subject_term_id: data-integration;genome.

Davis-Marcisak, E. F., Fitzgerald, A. A., Kessler, M. D., Danilova, L., Jaffee, E. M., Zaidi, N., Weiner, L. M., and Fertig, E. J. (2021). Transfer learning between preclinical models and human tumors identifies a conserved NK cell activation signature in anti-CTLA-4 responsive tumors. *Genome Medicine*, 13(1).

Dermitzakis, E. T. (2008). From gene expression to disease risk. *Nature Genetics*, 40(5):492–493. Bandiera_abtest: a Cg_type: Nature Research Journals Number: 5 Primary_atype: News & Views Publisher: Nature Publishing Group.

Dong, A., Li, Z., and Zheng, Q. (2021). Transferred Subspace Learning Based on Non-negative Matrix Factorization for EEG Signal Classification. *Frontiers in Neuroscience*, 15.

Du, P., Zhang, X., Huang, C.-C., Jafari, N., Kibbe, W. A., Hou, L., and Lin, S. M. (2010). Comparison of Beta-value and M-value methods for quantifying methylation levels by microarray analysis. *BMC Bioinformatics*, 11(1):587.

Fertig, E. J., Ding, J., Favorov, A. V., Parmigiani, G., and Ochs, M. F. (2010). CoGAPS: an R/C++ package to identify patterns and biological process activity in transcriptomic data. *Bioinformatics*, 26(21):2792–2793.

Fox, C. W. and Roberts, S. J. (2012). A tutorial on variational Bayesian inference. *Artificial Intelligence Review*, 38(2):85–95.

Grimmer, J. (2011). An Introduction to Bayesian Inference via Variational Approximations. *Political Analysis*, 19(1):32–47.

Huang, S., Chaudhary, K., and Garmire, L. X. (2017). More Is Better: Recent Progress in Multi-Omics Data Integration Methods. *Frontiers in Genetics*, 8:84.

Hutter, C. and Zenklusen, J. C. (2018). The Cancer Genome Atlas: Creating Lasting Value beyond Its Data. *Cell*, 173(2):283–285.

Jaakkola, T. S. and Jordan, M. I. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, 10(1):25–37.

Lee, A. J., Park, Y., Doing, G., Hogan, D. A., and Greene, C. S. (2020). Correcting for experiment-specific variability in expression compendia can remove underlying signals. *GigaScience*, 9(11).

Lock, E. F., Hoadley, K. A., Marron, J., and Nobel, A. B. (2013). JOINT AND INDIVIDUAL VARIATION EXPLAINED (JIVE) FOR INTEGRATED ANALYSIS OF MULTIPLE DATA TYPES. *The annals of applied statistics*, 7(1):523–542.

Love, M. I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12):550.

Manzoni, C., Kia, D. A., Vandrovcova, J., Hardy, J., Wood, N. W., Lewis, P. A., and Ferrari, R. (2018). Genome, transcriptome and proteome: the rise of omics data and their integration in biomedical sciences. *Briefings in Bioinformatics*, 19(2):286–302.

Mao, W., Zaslavsky, E., Hartmann, B. M., Sealfon, S. C., and Chikina, M. (2019). Pathway-level information extractor (PLIER) for gene expression data. *Nature Methods*, 16(7):607–610.

McInnes, L., Healy, J., and Melville, J. (2020). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426 [cs, stat].

Mo, Q., Shen, R., Guo, C., Vannucci, M., Chan, K. S., and Hilsenbeck, S. G. (2018). A fully Bayesian latent variable model for integrative clustering analysis of multi-type omics data. *Biostatistics (Oxford, England)*, 19(1):71–86.

Mounir, M., Lucchetta, M., Silva, T. C., Olsen, C., Bontempi, G., Chen, X., Noushmehr, H., Colaprico, A., and Papaleo, E. (2019). New functionalities in the TCGAbiolinks package for the study and integration of cancer data from GDC and GTEx. *PLOS Computational Biology*, 15(3):e1006701. Publisher: Public Library of Science.

Peng, M., Li, Y., Wamsley, B., Wei, Y., and Roeder, K. (2021). Integration and transfer learning of single-cell transcriptomes via cFIT. *Proceedings of the National Academy of Sciences*, 118(10):e2024383118.

Pierre-Jean, M., Deleuze, J.-F., Le Floch, E., and Mauger, F. (2020). Clustering and variable selection evaluation of 13 unsupervised methods for multi-omics data integration. *Briefings in Bioinformatics*, 21(6):2011–2030.

Pinheiro, J. C. and Bates, D. M. (2000). *Mixed-Effects Models in S and S-PLUS*. Statistics and Computing. Springer, New York.

Rappoport, N. and Shamir, R. (2018). Multi-omic and multi-view clustering algorithms: review and cancer benchmark. *Nucleic Acids Research*, 46(20):10546–10562.

Rohart, F., Gautier, B., Singh, A., and Cao, K.-A. L. (2017). mixOmics: An R package for 'omics feature selection and multiple data integration. *PLOS Computational Biology*, 13(11):e1005752. Publisher: Public Library of Science.

Saelens, W., Cannoodt, R., and Saeys, Y. (2018). A comprehensive evaluation of module detection methods for gene expression data. *Nature Communications*, 9(1):1090.

Santamarina-Ojeda, P., Tejedor, J. R., Pérez, R. F., López, V., Roberti, A., Mangas, C., Fernández, A. F., and Fraga, M. F. (2023). Multi-omic integration of DNA methylation and gene expression data reveals molecular vulnerabilities in glioblastoma. *Molecular Oncology*, 17(9):1726–1743. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/1878-0261.13479.

Searle, S. R., Speed, F. M., and Milliken, G. A. (1980). Population Marginal Means in the Linear Model: An Alternative to Least Squares Means. *The American Statistician*, 34(4):216–221. Publisher: Taylor & Francis _eprint: https://www.tandfonline.com/doi/pdf/10.1080/00031305.1980.10483031.

Seeger, M. and Bouchard, G. (2012). Fast Variational Bayesian Inference for Non-Conjugate Matrix Factorization Models. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, pages 1012–1018. PMLR. ISSN: 1938-7228.

Sergushichev, A. A. (2016). An algorithm for fast preranked gene set enrichment analysis using cumulative statistic calculation. *bioRxiv*, page 060012. Publisher: Cold Spring Harbor Laboratory Section: New Results.

Sharma, G., Colantuoni, C., Goff, L. A., Fertig, E. J., and Stein-O'Brien, G. (2020). projectR: an R/Bioconductor package for transfer learning via PCA, NMF, correlation and clustering. *Bioinformatics*, 36(11):3592–3593.

Silva, T. C., Colaprico, A., Olsen, C., D'Angelo, F., Bontempi, G., Ceccarelli, M., and Noushmehr, H. (2016). TCGA Workflow: Analyze cancer genomics and epigenomics data using Bioconductor packages. *F1000Research*, 5:1542.

Stein-O'Brien, G. L., Arora, R., Culhane, A. C., Favorov, A. V., Garmire, L. X., Greene, C. S., Goff, L. A., Li, Y., Ngom, A., Ochs, M. F., Xu, Y., and Fertig, E. J. (2018). Enter the Matrix: Factorization Uncovers Knowledge from Omics. *Trends in Genetics*, 34(10):790–805.

Stein-O'Brien, G. L., Clark, B. S., Sherman, T., Zibetti, C., Hu, Q., Sealfon, R., Liu, S., Qian, J., Colantuoni, C., Blackshaw, S., Goff, L. A., and Fertig, E. J. (2019). Decomposing Cell Identity for Transfer Learning across Cellular Measurements, Platforms, Tissues, and Species. *Cell Systems*, 8(5):395–411.e8.

Subramanian, I., Verma, S., Kumar, S., Jere, A., and Anamika, K. (2020). Multi-omics Data Integration, Interpretation, and Its Application. *Bioinformatics and Biology Insights*, 14:1177932219899051. Publisher: SAGE Publications Ltd STM.

Taroni, J. N., Grayson, P. C., Hu, Q., Eddy, S., Kretzler, M., Merkel, P. A., and Greene, C. S. (2019). MultiPLIER: A Transfer Learning Framework for Transcriptomics Reveals Systemic Features of Rare Disease. *Cell Systems*, 8(5):380–394.e4.

Tini, G., Marchetti, L., Priami, C., and Scott-Boyer, M.-P. (2019). Multi-omics integration—a comparison of unsupervised clustering methodologies. *Briefings in Bioinformatics*, 20(4):1269–1279.

Veeramachaneni, S. D., Pujari, A. K., Padmanabhan, V., and Kumar, V. (2019). A Maximum Margin Matrix Factorization based Transfer Learning Approach for Cross-Domain Recommendation. *Applied Soft Computing*, 85:105751.

Weiss, K., Khoshgoftaar, T. M., and Wang, D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1):9.

Zhou, W., Triche, Jr, T. J., Laird, P. W., and Shen, H. (2018). SeSAMe: reducing artifactual detection of DNA methylation by Infinium BeadChips in genomic deletions. *Nucleic Acids Research*, 46(20):e123.