ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-6654-0540-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICASSP43922.2022.9747362

MISMATCHED SUPERVISED LEARNING

Xun Xian[⋆] *Mingyi Hong*[⋆] *Jie Ding*[†]

[⋆] Department of Electrical and Computer Engineering, University of Minnesota
[†] School of Statistics, University of Minnesota

ABSTRACT

Supervised learning scenarios, where labels and features are possibly mismatched, have been an emerging concern in machine learning applications. For example, researchers often need to align heterogeneous data from multiple resources to the same entities without a unique identifier in the socioeconomic study. Such a mismatch problem can significantly affect the learning performance if it is not appropriately addressed. Due to the combinatorial nature of the mismatch problem, existing methods are often designed for small datasets and simple linear models but are not scalable to large-scale datasets and complex models. In this paper, we first present a new formulation of the mismatch problem that supports continuous optimization problems and allows for gradient-based methods. Moreover, we develop a computation and memory efficient method to process complex data and models. Empirical studies on synthetic and real-world data show significantly better performance of the proposed algorithms than state-of-the-art methods.

Index Terms— Mismatched Data, Supervised Learning, Stochastic Gradient Descent (SGD), Permutation Matrix.

1. INTRODUCTION

In several emerging machine learning applications [1–3], the data labels and features variables are not always correctly aligned, known as the *mismatch supervised learning* problem. Such a problem can arise, for example, in Assisted Learning [3] where organizations holding heterogeneous features collaborate on a task, but their data may not be precisely aligned according to a data identifier. This is often the case when the privacy regulation [4] does not allow institutions such as a medical laboratory and a hospital to precisely link their data via a unique personal identifier, but only side information such as address and age can be utilized.

The mismatch between labels and features hinders the use of classical supervised learning techniques for predictive modeling. To address the mismatch issue, previous works have adopted a mismatch regression formulation. Specifically, suppose that the label vector Y is generated from $Y = \Pi^* X\beta^* + \epsilon$, where $Y = [y_1, ..., y_n]^\top \in \mathbb{R}^n$, $X = [x_1, ..., x_n]^\top \in \mathbb{R}^{n\times d}$, Π^* is an unknown permutation matrix (where each row and each column contain one entry of 1, and 0s elsewhere), and $\epsilon = [\epsilon_1, ..., \epsilon_n]^\top \in \mathbb{R}^n$ is the noise term. The mismatch regression problem [5–11] aims to simultaneously estimate the permutation matrix and linear regression coefficient from solving the following:

$$\min_{\beta\in\mathbb{R}^d, \Pi\in\mathcal{P}_n} \mathcal{L}(\beta, \Pi) = \|Y - \Pi X\beta\|^2, \tag{1}$$

where $\mathcal{P}_n$ is the set of all $n \times n$ permutation matrices. The above optimization problem is NP-hard for $d > 1$ [6], and the difficulty comes from the excessively large search space $\mathcal{P}_n$ of size $n!$. Some theoretical questions, including the recovery of Π^* in terms of the signal-to-noise (SNR) ratio (i.e., $\|\beta^*\|^2/Var(\epsilon_i)$), and the necessary condition on the SNR for approximately recovering β^* from Eq. (1) have been studied in [6, 8], respectively.

Developing efficient and effective algorithms for solving problem (1) is challenging due to its combinatorial nature (since $\mathcal{P}_n$ is discrete). There have been several research progresses in this line. For instance, a branch-and-bound-based method that can solve problem (1) with a small sample size e.g., $n = 20$, was proposed in [5]. Also, a concave minimization reformulation was proposed in [12] to solve problems with sample size n at the order of 100 with reasonable amount of time. Algebraic geometry-based algorithms that can solve problem (1) with low dimension e.g., $d = 5$, and sample size approximately 10^4 were demonstrated in [11], but the computational cost increases exponentially with d. The above methods are not scalable to large-scale datasets, and they can only be applied to simple linear models.

Recently, the work of [13] developed a new formulation that combines the bilevel optimization and the optimal transport [14] techniques to allow nonlinear modeling. The main idea is to cast the original problem (1) as an optimal transport problem [15] with a cost matrix $C \in \mathbb{R}^{n\times n}$ and a transport (permutation) plan Π. Then, the parameters are updated by treating β as an implicit function of Π. From the experimental study, the proposed bilevel approach significantly outperforms all other existing methods. However, a potential downside of the bilevel method is that, in each iteration, we need $O(n^2)$ memory and $O(dn^2)$ computational cost. Such costs

This paper is based upon work supported by the Cisco Research and the National Science Foundation under grant number DMS-2134148.

Authorized licensed use limited to: The University of Utah. Downloaded on July 15,2022 at 22:22:30 UTC from IEEE Xplore. Restrictions apply.

are prohibitive given real-world applications. We will refer to the bilevel approach as the state-of-the-art (SOTA) method for the rest of the paper.

In this paper, we develop computation and memory efficient methods. We first reformulate the original mixed-integer problem as a penalized optimization with continuous variables. The main idea is to first relax the discrete set of permutation matrices to its convex hull to obtain a continuous problem. Then, based on a critical observation that the permutation matrix is the sparsest among all the convex hulls, we enforce the optimization variable Π to be sparse by adding regularization terms. The reformulation enables the direct use of gradient-based optimization solvers. Moreover, we develop an algorithm to update the parameters using mini-batch samples, which can be easily integrated into popular programming frameworks.

2. PROBLEM AND FORMULATION

Notations. Let $\|\cdot\|$, $\|\cdot\|_1$ denote the ℓ_2, ℓ_1 norms of vectors, respectively. For a square matrix M, let $M_{i,\cdot}$ denote its i-th row and $M_{\cdot,j}$ denote its j-th column.

Suppose that the data $\{(y_i, z_i)\}_{i=1}^n$ and $\{x_j\}_{j=1}^n$ are collected from different sources. For instance, $\{(y_i, z_i)\}_{i=1}^n$ are the health records from a hospital and $\{x_j\}_{j=1}^n$ are the purchasing information from a company. Under the privacy regulations, $\{x_j\}_{j=1}^n$ may not be correctly aligned with $\{(y_i, z_i)\}_{i=1}^n$ in the absence of a unique identifier. To learn the underlying correspondence between $\{(y_i, z_i)\}_{i=1}^n$ and $\{x_j\}_{j=1}^n$, as well as the supervised relationship, we consider the following minimization problem of

$$\ell(\beta_1, \beta_2, \Pi) \triangleq \frac{1}{n}\|Y - g(Z;\beta_1) - \Pi f(X;\beta_2)\|^2,$$

over $(\beta_1, \beta_2, \Pi) \in \Theta$, where $\Theta = \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \times \mathcal{P}_n$ is the joint optimizing space, $\mathcal{P}_n$ is the set of $n \times n$ permutation matrices, $Y = [y_1, \ldots, y_n]^\top \in \mathbb{R}^n$ is the label vector, Π is a permutation matrix, $f : \mathbb{R}^{d_1} \to \mathbb{R}$ and $g : \mathbb{R}^{d_2} \to \mathbb{R}$ are unknown functions to be learned, and $g(Z;\beta_1) = [g(z_1;\beta_1), \ldots, g(z_n;\beta_1)]^\top \in \mathbb{R}^n$, $f(X;\beta_2) = [f(x_1;\beta_2), \ldots, f(x_n;\beta_2)]^\top \in \mathbb{R}^n$.

The above formulation for regression can be tailored to a classification problem by setting Y to discrete labels and changing the squared-ℓ_2 loss to other losses, e.g., the cross-entropy loss. For example, with discrete labels Y, in terms of logistic regression, the loss function is

$$\sum_{i=1}^n -(y_i\beta_2^\top \Pi_{i,\cdot} X + y_i\beta_1^\top z_i) + \ln(1 + e^{\beta_2^\top \Pi_{i,\cdot} X + \beta_1^\top z_i}).$$

In fact, directly optimizing Π over $\mathcal{P}_n$ could potentially lead to overfitting, since there are exponentially many choices for Π (i.e., $|\mathcal{P}_n| = n!$). To reduce the excessive freedom, we restrict Π to the set that the number of permuted entries is smaller than a pre-specified threshold K, namely to consider $\mathcal{P}_{n,K} \triangleq \{\Pi \in \mathcal{P}_n : d_H(\Pi, I_n) \le K\}$ with Hamming distance $d_H(A, B) \triangleq \sum_{1\le i,j\le n} \mathbf{1}(A_{i,j} \neq B_{i,j})$. Thus, the above mismatch regression problem becomes:

$$\min \ell(\beta_1, \beta_2, \Pi) \text{ s.t. } (\beta_1, \beta_2, \Pi) \in \Omega, \tag{2}$$

where $\Omega = \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \times \mathcal{P}_{n,K}$, and K is a pre-specified positive integer.

The new problem is challenging to solve since it involves a discrete variable Π, and the set $\mathcal{P}_{n,K}$ is generally infeasible to optimize [9, 16]. To address this challenge, we first transform the discrete problem into a continuous one through convexifying. Then, we use a special regularization term to push Π to be a permutation matrix with the number of permuted entries less than K.

To obtain a continuous optimization variable, we extend the discrete search set $\mathcal{P}_n$ to its convex, namely Brikhoff polytope (also known as the doubly stochastic set), $\mathcal{B}_n \triangleq \{\Pi : \Pi_{ij} \ge 0, \forall(i,j), \sum_{i=1}^n \Pi_{ij} = 1, \forall j, \sum_{j=1}^n \Pi_{ij} = 1, \forall i\}$. To further ensure that Π is a permutation matrix, we propose to utilize the following ℓ_{1-2} regularizer

$$P_{1-2}(\Pi) \triangleq \sum_{i=1}^n \|\Pi_{i,\cdot}\|_1 - \|\Pi_{i,\cdot}\|_2 + \sum_{j=1}^n \|\Pi_{\cdot,j}\|_1 - \|\Pi_{\cdot,j}\|_2.$$

The ℓ_{1-2} norm was historically studied in the context of compressed sensing [17] to produce sparse vectors under linear constraints [18, 19].

For the constraint $\mathcal{P}_{n,K}$, we propose to parameterize it with the following continuous trace penalty

$$P_K(\Pi) \triangleq [n - K - \text{Tr}(\Pi)]_+,$$

where $K \in \{1, \ldots, n\}$ is a pre-specified value, $\text{Tr}(\cdot)$ is the trace operator and $[x]_+ = \max(0, x)$. Combining the ℓ_{1-2} and the trace norms above, we propose to add the following regularization term to the original problem (2)

$$R_K(\Pi) \triangleq P_{1-2}(\Pi) + P_K(\Pi). \tag{3}$$

Next, we show in the following that the new penalty term is zero if and only if the Π is a permutation matrix whose number of permuted entries is fewer than or equal to K. Due to the page limit, we include the proof in the appendix.

Theorem 1. *A square matrix $\Pi \in \mathcal{B}_n$ is a permutation matrix with the number of permuted entries fewer or equal to K, if and only if $P(\Pi) = 0$.*

Based on the above result, we propose a new formulation of the mismatch regression problem (2) as the minimization of

$$\mathcal{L}(\beta_1, \beta_2, \Pi) = \ell(\beta_1, \beta_2, \Pi) + \lambda R_K(\Pi), \tag{4}$$

over $\beta_1 \in \mathbb{R}^{d_1}, \beta_2 \in \mathbb{R}^{d_2}, \Pi \in \mathcal{B}_n$, where $\lambda > 0$ is a tuning parameter, and $K \in \{1, 2, \ldots, n\}$ is a pre-specified value. From the computational aspect, the problem can be optimized with gradient-based methods, which will be elaborated in detail in the next section.

Authorized licensed use limited to: The University of Utah. Downloaded on July 15,2022 at 22:22:30 UTC from IEEE Xplore. Restrictions apply.

3. PROPOSED ALGORITHM

We present the pseudocode for solving the problem (4) (regression problems) in Algorithm 1. For solving classification problems, the algorithms will be similar to Algorithm 1 with slight modifications. We include the full details in Section C in the supplement.

The main idea for the proposed Algorithm 1 is to update Π and β_1, β_2 in a coordinate-wise fashion. At each iteration, we first perform mini-batch stochastic gradient descent (SGD) on Π (Lines 3 - 8). We briefly introduce the intuition of Line 6. In classical SGD, a key component is to find an unbiased estimate of the true gradient based on few samples. However, in our case, to calculate the derivative $\partial_{\Pi_{i,j}} \ell(\beta_1, \beta_2, \Pi) = -2(y_i - g(z_i; \beta_1) - \Pi_{i,\cdot} f(X; \beta_2)) f(x_j; \beta_2)$, we need to pass through the *full* batch of X instead of a mini-batch, which costs prohibitive memory. To address the challenge, we utilize the key fact that Π is a probability simplex. Consequently, $\Pi_{i,\cdot} f(X; \beta_2)$ is a weighted sum of $f(X; \beta_2)$ over probability simplex $\Pi_{i,\cdot}$. Then, a natural way to approximate $\partial_{\Pi_{i,j}} \ell(\beta_1, \beta_2, \Pi)$ is to sample a data index s from a multinominal distribution of parameter $\Pi_{i,\cdot}$, and replace $\Pi_{i,\cdot} f(X; \beta_2)$ with $f(x_s; \beta_2)$ in calculating the derivative.

The variable Π is no longer a doubly stochastic matrix after the above update (Lines 3-8). To maintain the doubly stochastic constraint, we apply the Sinkhorn procedure [20], which normalizes the matrix by its columns and rows sequentially (Lines 10-12). It was shown in [20] that such an iterative procedure can transform a nonnegative matrix into a doubly stochastic matrix. We only apply the Sinkhorn procedure once for computational efficiency. Although Π is not restricted to be doubly stochastic during the training process, from the experimental study, we observe that β_1 and β_2 can still be successfully estimated. Finally, we update β_1 and β_2 by using SGD based on the mini-batch samples (Line 13). It can be verified that the memory required for calculating gradients at each iteration is $O(m^2)$, which is much more efficient than the state-of-the-art method proposed in [13] when the mini-batch size m is much smaller than sample size n.

Algorithm 1 Mini-Batch SGD for Problem (4)

Input: Data $\{(y_i, x_i, z_i)\}_{i=1}^n$, Learning models f and g.
Initialization: $\Pi^1 = I_n$, $\beta_1^1 \sim \mathcal{N}(\mathbf{0}, I_{d_1})$, $\beta_2^1 \sim \mathcal{N}(\mathbf{0}, I_{d_2})$, learning rate sequence $\{\alpha_t\}_{t=1}^T$, tuning parameter $\lambda = 1$, mini-batch size m and the trace penalty parameter K.

1: **for** $t = 1$ to T **do**
2: Uniformly sample a mini-batch D^t with size m.
3: **for** $i \in$ index(D^t) **do**
4: **for** $j \in$ index(D^t) **do**
5: Sample an index s from Multinominal$(\Pi_{i,\cdot}^t)$
6: Calculate $d_{i,j} \triangleq 2(g(z_i; \beta_1^t) + f(x_s; \beta_2^t) - y_i) f(x_j; \beta_2^t)$
Calculate the derivative of $\partial_{i,j} R_K(\Pi^t)$ // $R_K(\cdot)$ is defined in Eq. (3)
7: **end for**
8: $\widetilde{\Pi}_{i,j}^t \leftarrow \Pi_{i,j}^t - \alpha_t(d_{i,j} + \lambda \partial_{i,j} R_K(\Pi^t))$
9: **end for**
10: $\widetilde{\Pi}^{t+1} \leftarrow max(0, \widetilde{\Pi}^t)$
11: $\widehat{\Pi}_{\cdot,j}^{t+1} \leftarrow \widetilde{\Pi}_{\cdot,j}^{t+1} / \left\| \widetilde{\Pi}_{\cdot,j}^{t+1} \right\|_1$ for $j = 1, \ldots, n$
12: $\Pi_{i,\cdot}^{t+1} \leftarrow \widehat{\Pi}_{i,\cdot}^{t+1} / \left\| \widehat{\Pi}_{i,\cdot}^{t+1} \right\|_1$ for $i = 1, \ldots, n$
13: Update β_1^t and β_2^t with Stochastic Gradient Descent based on the mini-batch D^t
14: **end for**

† index$(\cdot)$ outputs the indices of a subset of data, e.g., index$(\{x_1, x_5, x_9\}) = \{1, 5, 9\}$.

Output: Model parameters: $\beta_1^{T+1}, \beta_2^{T+1}$.

4. EXPERIMENTAL STUDY

We evaluates the proposed method in terms of both predictive and computational (in appendix) performances on various datasets. For all experiments, the 'oracle score' is the test error obtained by the model trained on the correctly matched data (in hindsight). We use the root-mean-square-error (RMSE) and Accuracy as evaluation metrics for regression and classification tasks, respectively. We set both f and g to be the same type of functions e.g., both f and g are linear functions. For the initial values of β_1 and β_2, we first fit models based on the original training data, and then initialize β_1 and β_2 to be the parameters of the fitted models. We set the initial Π^1 to be the identity matrix throughout this section.

4.1. Synthetic Data

Following the setting in [13], we generate 1000 training and 1000 test samples from $y = \beta^\top x + \varepsilon$, where $x \sim \mathcal{N}(0, I_{d_2})$, $\beta \sim \mathcal{N}(0, I_{d_2})$, and $\varepsilon \sim (0, \sigma^2)$. Here, we do not have the variable Z. We randomly permute different proportions of the training data and keep the test data untouched. We use the linear model and test on four combinations of different feature dimensions d_2, SNRs (i.e., $\|\beta\|^2/\sigma^2$), and permutation ratios.

Results are summarized in Table 1. With a permutation ratio of 10%, the out-sample prediction performance of our method approaches the oracle score, which indicates the wide adaptability of the new formulation and the proposed algorithms. When the permutation ratio increases to 50%, our method achieves far better performance than the state-of-the-art method. In addition to the linear case, we also test on a standard nonlinear regression dataset named Friedman 1 [21]. Due to the page limit, we include more experimental details e.g., non-linear data and partial mismatch scenarios in appendix.

4.2. Real-world Applications

MIMIC3 Medical Information Mart for Intensive Care III (MIMIC3 [22]) is a comprehensive clinical database containing de-identified information. We test on two benchmarks of the MIMIC3: (i) regression task: predicting the Length of Stay (LOS) and (ii) classification task: classifying Phenotypes. For the regression task, we use feature variables from

Authorized licensed use limited to: The University of Utah. Downloaded on July 15,2022 at 22:22:30 UTC from IEEE Xplore. Restrictions apply.

Table 1: The out-sample prediction performance of our proposed method and the SOTA approach [13] on synthetic linear data, MIMIC3, and SP. The second row lists the machine learning models used for regression and classification. LR is the linear regression, LR/C is the logistic model for classification, and NN is the two-layer neural network for regression. The third row indicates the SNR of the data generating process (if applicable), the feature dimension, and the permutation ratio. The fourth row describes the sample size and mini-batch size used for training. The rest rows summarize the average out-sample prediction performance as measured by RMSE for regression and Accuracy for classification, and standard errors over 20 replicates are included in the brackets.

Data	**Synthetic Linear**				**MIMIC3**		**SP**
Model Types	**LR**	**LR**	**LR**	**LR**	**LR/C**	**NN**	**NN**
SNR, Dim, Permu%	$50, 5, 10$	$50, 5, 50$	$10^2, 10, 10$	$10^2, 10, 50$	NA, $16, 50$	NA, $16, 75$	NA, $31, 30$
Sample/Batch size	$10^3/32$	$10^3/32$	$10^3/64$	$10^3/64$	$10^5/256$	$10^5/256$	$10^3/64$
Proposed Method	$0.28(.05)$	$0.58(.08)$	$0.30(.04)$	$0.88(.07)$	$0.60(.06)$	$114(2.2)$	$1.5(0.31)$
SOTA method [13]	$0.27(.04)$	$1.67(.62)$	$0.31(.03)$	$2.01(.81)$	/	/	/
Oracle Score	$0.26(.02)$	$0.26(.02)$	$0.29(.02)$	$0.29(.02)$	$0.71(.03)$	$108(1.1)$	$1.1(.18)$

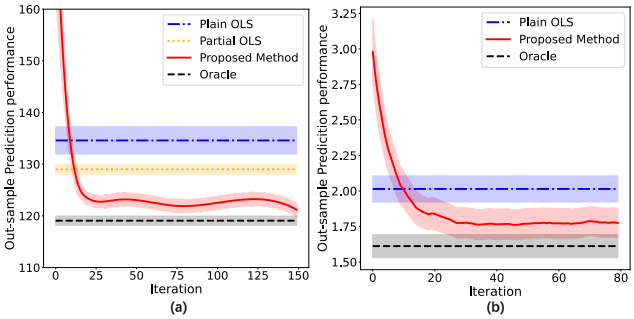

Fig. 1: The out-sample prediction performance of (a) linear model on MIMIC3 benchmark predicting the LOS against iterations; (b) linear model on SP predicting poverty level against iterations. The shaded regions describe the ± 1 standard errors. The Plain OLS is the test error of the model trained on the original training data. The partial OLS represents the test error of the model trained on the data where labels (Y) and partial features (Z) have right correspondence.

the laboratory table (X) and the ICU charted event table (Z) to predict the LOS (Y). Following the procedures in [23] with minor modifications, we randomly selected 10000 training and 10000 test cases, with 16 features (3 from the lab table and 13 from the ICU charted event table). A unique identifier can link the official data. Yet, in practice, the laboratory measurements (X) are often not precisely linked to the labels (Y) and the features (Z) under privacy regulations. To simulate such a situation, we randomly permute a proportion of the training data in the Lab table (X). As a result, the training data X loses correspondence with the training label Y and the remaining training data Z. Similar procedures are used to prepare the data for the classification task, and details are included in appendix. In addition, we use both linear models and two-layer neural networks. In Figure 1(a), with a permutation ratio of 50%, $\lambda = 100$, and linear model, the error terms quickly decrease and converge to the oracle score with negligible differences.

Socioeconomic Prediction (SP) In socioeconomic study, researchers often need to collect and combine data from multiple sources for further use. For instance, in a series of work [24], the authors integrate accurate satellite-based information (e.g., nightlights intensity) and survey data from some African countries to predict the poverty level. Specifically, a goal is to predict the *Consumption Level* in Malawi. Following the procedures in [24] with minor modifications, we randomly select 800 training and 400 test cases, and in total, 31 features. The nightlights data (Z) have exact correspondence concerning the label Y (*Consumption Level*) since they are collected according to the longitudes and latitudes. But the survey data (X) is often not precisely aligned with the real-time nightlights data Z and the label Y due to the periodic collecting process. We randomly permute a proportion of the training data (X) to simulate such a scenario. The training data X loses correspondence with the training label Y and the remaining training data Z. In Fig. 1(b), with a permutation ratio of 30%, $\lambda = 100$, and linear models, the error terms quickly decrease and stay close to the oracle score.

5. CONCLUSION

The mismatch data scenarios have posed challenges that classical supervised learning techniques cannot address well. We proposed a new problem formulation by casting the mismatch supervised learning problem from a discrete-space optimization into a continuous optimization problem with properly defined regularizations. Additionally, we developed a computation and memory efficient method that can scale to complex data and models. Finally, we demonstrated the effectiveness of our proposed approach through various experimental studies. An interesting future direction is to analyze the convergence properties of the developed method. We include the proofs and additional experimental results in the appendix.

Authorized licensed use limited to: The University of Utah. Downloaded on July 15,2022 at 22:22:30 UTC from IEEE Xplore. Restrictions apply.

6. REFERENCES

[1] Fritz Scheuren and William E Winkler, "Regression analysis of data files that are computer matched," *Surv. Methodol.*, vol. 19, no. 1, pp. 39–58, 1993.

[2] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong, "Federated machine learning: Concept and applications," *in Proc. TIST*, vol. 10, no. 2, pp. 12, 2019.

[3] Xun Xian, Xinran Wang, Jie Ding, and Reza Ghanadan, "Assisted learning: A framework for multi-organization learning," *in Proc. NEURIPS*, 2020.

[4] Paul Voigt and Axel Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, pp. 3152676, 2017.

[5] Valentin Emiya, Antoine Bonnefoy, Laurent Daudet, and Rémi Gribonval, "Compressed sensing with unknown sensor permutation," in *Proc. ICASSP*. IEEE, 2014, pp. 1040–1044.

[6] Ashwin Pananjady, Martin J Wainwright, and Thomas A Courtade, "Linear regression with an unknown permutation: Statistical and computational limits," in *Proc. AACCC*. IEEE, 2016, pp. 417–424.

[7] Abubakar Abid, Ada Poon, and James Zou, "Linear regression with shuffled labels," *arXiv preprint arXiv:1705.01342*, 2017.

[8] Daniel Hsu, Kevin Shi, and Xiaorui Sun, "Linear regression without correspondence," *arXiv preprint arXiv:1705.07048*, 2017.

[9] Martin Slawski, Emanuel Ben-David, et al., "Linear regression with sparsely permuted data," *Electron. J. Stat.*, vol. 13, no. 1, pp. 1–36, 2019.

[10] Jayakrishnan Unnikrishnan, Saeid Haghighatshoar, and Martin Vetterli, "Unlabeled sensing with random linear measurements," *IEEE Trans. Inf. Theory*, vol. 64, no. 5, pp. 3237–3253, 2018.

[11] Manolis C Tsakiris, Liangzu Peng, Aldo Conca, Laurent Kneip, Yuanming Shi, and Hayoung Choi, "An algebraic-geometric approach for linear regression without correspondences," *IEEE Trans. Inf. Theory*, vol. 66, no. 8, pp. 5130–5144, 2020.

[12] Liangzu Peng and Manolis C Tsakiris, "Linear regression without correspondences via concave minimization," *IEEE Signal Processing Letters*, vol. 27, pp. 1580–1584, 2020.

[13] Yujia Xie, Yixiu Mao, Simiao Zuo, Hongteng Xu, Xiaojing Ye, Tuo Zhao, and Hongyuan Zha, "A hypergradient approach to robust regression without correspondence," *arXiv preprint arXiv:2012.00123*, 2020.

[14] Marco Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *in Proc. NEURIPS*, vol. 26, pp. 2292–2300, 2013.

[15] Cédric Villani, *Optimal transport: old and new*, vol. 338, Springer, 2009.

[16] Martin Slawski, Emanuel Ben-David, and Ping Li, "Two-stage approach to multivariate linear regression with sparsely mismatched data," *J. Mach. Learn. Res.*, vol. 21, no. 204, pp. 1–42, 2020.

[17] David L Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[18] Ernie Esser, Yifei Lou, and Jack Xin, "A method for finding structured sparse solutions to nonnegative least squares problems with applications," *SIAM J. Imaging Sci.*, vol. 6, no. 4, pp. 2010–2046, 2013.

[19] Jiancheng Lyu, Shuai Zhang, Yingyong Qi, and Jack Xin, "Autoshufflenet: Learning permutation matrices via an exact lipschitz continuous penalty in deep convolutional neural networks," in *Proc. SIGKDD*, 2020, pp. 608–616.

[20] Richard Sinkhorn and Paul Knopp, "Concerning nonnegative matrices and doubly stochastic matrices," *Pac. J. Math.*, vol. 21, no. 2, pp. 343–348, 1967.

[21] Jerome H Friedman, "Multivariate adaptive regression splines," *Ann. Stat.*, vol. 19, no. 1, pp. 1–67, 1991.

[22] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark, "Mimic-iii, a freely accessible critical care database," *Sci. Data*, vol. 3, pp. 160035, 2016.

[23] Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan, "Multitask learning and benchmarking with clinical time series data," *Sci. Data*, vol. 6, no. 1, pp. 1–18, 2019.

[24] Neal Jean, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon, "Combining satellite imagery and machine learning to predict poverty," *Science*, vol. 353, no. 6301, pp. 790–794, 2016.

Authorized licensed use limited to: The University of Utah. Downloaded on July 15,2022 at 22:22:30 UTC from IEEE Xplore. Restrictions apply.