Optimal Transport for Structure Learning Under Missing Data

Vy Vo [1 2] **He Zhao** [2] **Trung Le** [1] **Edwin V. Bonilla** [2] **Dinh Phung** [1 3]

Abstract

Causal discovery in the presence of missing data introduces a chicken-and-egg dilemma. While the goal is to recover the true causal structure, robust imputation requires considering the dependencies or preferably causal relations among variables. Merely filling in missing values with existing imputation methods and subsequently applying structure learning on the complete data is empirical shown to be sub-optimal. To this end, we propose in this paper a score-based algorithm, based on optimal transport, for learning causal structure from missing data. This optimal transport viewpoint diverges from existing score-based approaches that are dominantly based on EM. We project structure learning as a density fitting problem, where the goal is to find the causal model that induces a distribution of minimum Wasserstein distance with the distribution over the observed data. Through extensive simulations and real-data experiments, our framework is shown to recover the true causal graphs more effectively than the baselines in various simulations and real-data experiments. Empirical evidences also demonstrate the superior scalability of our approach, along with the flexibility to incorporate any off-the-shelf causal discovery methods for complete data.

1. Introduction

Discovering the causal structures among different variables holds great significance in many scientific disciplines (Sachs et al., 2005; Richens et al., 2020; Wang et al., 2020; Zhang et al., 2013). The gold standard approach to identifying causal relationships is through randomized controlled experiments, which are however often infeasible due to various ethical and practical constraints. There have thus been ongoing efforts towards causal discovery from observational data (Spirtes et al., 2000; Chickering, 2002; Heckerman et al., 2006; Zheng et al., 2018; Yu et al., 2019; Glymour et al., 2019; Zheng et al., 2020). Whereas existing structure learning algorithms mostly deal with complete data, practical real-world data are often messy, potentially with multiple missing values. Eliminating samples affected by 'missingness' and solely analyzing the observed cases are undesirable, since not only would it decrease the sample but also introduce bias to the estimations (Tu et al., 2019; Mohan & Pearl, 2021). Another strategy is to impute the missing values. The key challenge of imputation lies in effectively modeling the data distribution when dealing with a substantial number of missing values. Pioneered by Little & Rubin (2019), missing data mechanisms are commonly classified into three types: Missing Completely at Random (MCAR), Missing at Random (MAR), and Missing Not at Random (MNAR). Mohan & Pearl (2021) further propose modelling the missing process with causal graphs, and given such a graph, present conditions where the joint distribution is identifiable. Without considering the underlying causes of missingness, learning an imputation model from observed data is prone to bias wherein one may condition on some spurious variables that introduces extra relations that do not exist in the true graph (Kyono et al., 2021). It is indeed a radical challenge since the causal graph is mostly unknown in practice. Given that our goal here is to discover the causal structure from missing data, this indeed poses a chicken-and-egg problem.

A straightforward approach to dealing with our problem is to impute the missing values and subsequently apply any existing causal discovery method. This is however not an effective strategy. Figure 1 illustrates the quality of imputation (by Euclidean distance) in relation to causal discovery performance (by F1 score) across different missing rates and mechanisms. We study 3 popular imputation methods: Mean imputation, Iterative imputation and Optimal Transport (OT) imputation (using Sinkhorn divergence) (Muzellec et al., 2020). Iterative imputer consistently yields the best imputation, which however does not necessarily lead to better causal discovery, as shown more obviously in MAR and MNAR cases. It is possible that the filled-in data distribution encodes a different set of independence constraints, resulting a distribution compatible with a different causal graph from the true one. The sub-optimality of such a naive approach has also been reported in the prior research (Kyono

[1]Monash University, Australia [2]CSIRO's Data61, Australia [3]VinAI Research, Vienam. Correspondence to: Vy Vo <v.vo@monash.edu>.

et al., 2021; Gao et al., 2022), implying that "good" imputation in terms of reconstruction does not guarantee "good" imputation to causal discovery. This motivates the development of an end-to-end machinery dedicated to structure learning in the presence of missing data.

Depending on the base causal discovery algorithm (designed for complete data), existing methods can also be categorized as either constraint-based or score-based. Constraint-based methods are mainly built up on the PC algorithm, which exploits (conditional) independence tests (Strobl et al., 2018; Tu et al., 2019; Gain & Shpitser, 2018). Score-based causal discovery has recently taken off, owning to advances in exact characterization of acyclicity of the graph (Zheng et al., 2018; Yu et al., 2019; Bello et al., 2022). Extension of score-based approach in the context of missing data is nonetheless less studied[1]. A notable method in this line of research is MissDAG (Gao et al., 2022). MissDAG proposes an EM-style algorithm (Dempster et al., 1977) to point estimate the graph. In the E step, MissDAG imputes the missing entries by modelling a posterior distribution over the missing part of the data. This gives rise to the expected log-likelihood of the complete data, which MissDAG maximizes to estimate the model parameters in the M step. MissDAG has shown a significantly improved performance over constraint-based methods. However, the framework suffers from two key drawbacks. First, the time inefficiency issue of vanilla EM hinders scalability of MissDAG. Second, a large part of the work focuses on linear Gaussian models where the posterior exists in closed from. In the non-Gaussian and non-linear cases, MissDAG resorts to rejection sampling, which introduces extra computational overhead. Furthermore, approximate inference may compromise the estimation accuracy greatly, especially in real-world settings where the model tends to be mis-specified. Section 4.2 reveals the poor performance of MissDAG on real datasets, where the method fails to detect any true edges in several cases.

Contribution. In this work, we introduce **OTM** - an **O**ptimal **T**ransport framework for learning causal graphs under **M**issing data. From the viewpoint of optimal transport (OT, Villani et al., 2009), we fit structure learning into a general landscape of density fitting problems, to which OT has proved successful (Arjovsky et al., 2017; Tolstikhin et al., 2017; Vuong et al., 2023; Vo et al., 2023). The high-level idea is to minimize the Wasserstein distance between the model distribution (often in a parametric family) and the empirical data distribution. There are two key properties that makes OT an attractive solution to our problem: (1) Wasserstein distance is a metric, thus providing a more geometrically meaningful distance between two distributions than the standard $f-$divergences; (2) the Wasserstein estimator is empirically shown in past studies to be more robust to mis-specifications (Bernton et al., 2019; Vo et al., 2023) and several practical settings where the maximum likelihood estimate fails, for example when the model is a singular distribution or the two distributions have little or no overlapping support (Bassetti et al., 2006; Canas & Rosasco, 2012; Peyré et al., 2017; Arjovsky et al., 2017).

[1]A review of other related works can be found in Appendix B.

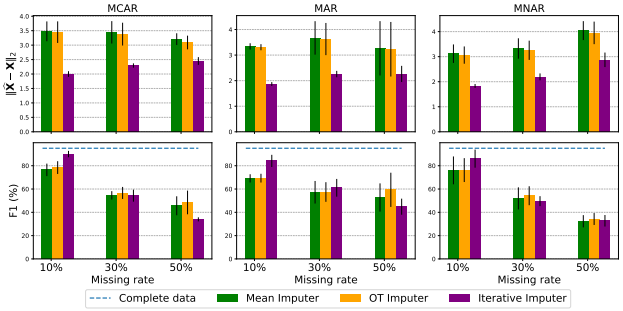

Figure 1. Visualization of the quality of imputation vs. causal discovery performance. Better imputation in terms of reconstruction quality does not always imply more accurate structure learning.

As a result, our proposal OTM is a flexible framework in the sense that (1) the method can accommodate any existing score-based causal discovery algorithm for complete data, and (2) we make no assumptions about the model forms nor the missing mechanism. In Section 3, we present our theoretical development that renders a tractable formulation of the OT cost to optimize the model distribution on the missing data. In the remainder of the paper, we provide a comprehensive study of the non-linear scenarios on both synthetic and real datasets. Across our experiments, OTM consistently exhibits the best/second-best performances in accurately recovering the true causal structures, while exhibiting superior scalability when the graph complexity increases.

2. Preliminaries

We first introduce the notations and basic concepts in the paper. We reserve bold capital letters (i.e., $\mathbf{G}$) for notations related to graphs. We use calligraphic letters (i.e., $\mathcal{X}$) for spaces and bold upper case (i.e. $\boldsymbol{X}$) for data matrix. We also use upper case letters (i.e., X) for random variables and lower case letters (i.e., x) for values. Finally, we use $[d]$ to denote a set of integers $\{1, 2, \cdots, d\}$.

2.1. Structural Causal Model

A directed graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ consists of a set of nodes $\mathbf{V}$ and an edge set $\mathrm{E} \subseteq \mathbf{V}^2$ of ordered pairs of nodes with $(v, v) \notin \mathbf{E}$ for any $v \in \mathbf{V}$ (one without self-loops). For a pair of nodes i, j with $(i, j) \in \mathrm{E}$, there is an arrow pointing from i to j and we write $i \to j$. Two nodes i and j are adjacent if either $(i, j) \in \mathbf{E}$ or $(j, i) \in \mathbf{E}$. If there is an arrow from i to j then i is a parent of j and j is a child of i.

Let PA_{X_i} denote the set of variables associated with parents of node i in $\mathbf{G}$.

The causal relations among d variables is characterized via a *structural causal model* (SCM, Pearl, 2009) over the tuple $\langle U, X, f\rangle$ that generally consists of a sets of assignments

$$X_i := f_i\big(\mathrm{PA}_{X_i}, U_i\big), \quad i \in \mathbf{V}, \tag{1}$$

where U_i is an exogenous variable assumed to be mutually independent with variables $\{U_1, \cdots, U_d\}\backslash U_i$. Given a joint distribution over exogenous variables $P(U_1, \cdots, U_d)$, the functions $f = [f_i]_{i=1}^d$ define a joint distribution $P(X)$ over the endogenous variables X. Each SCM induces a causal graph $\mathbf{G}$, which we assume in this work to be a *directed acylic graph* (DAG). We also make standard causal discovery assumptions: (1) the distribution $P(X)$ and the induced graph $\mathbf{G}$ satisfy the Markov property (Pearl, 2009) and (2) there are no latent confounders of the observed variables. In this work, we focus on the additive noise models (ANMs, Peters et al., 2014; Hoyer et al., 2008) where the graph $\mathbf{G}$ can be uniquely identified. In ANMs, (1) takes the form $X_i := f_i(\mathrm{PA}_{X_i}) + U_i, \quad i \in \mathbf{V}$.

2.2. DAG Learning and Acyclicity Characterization

The task of structure learning is to discover the DAG $\mathbf{G}$ underlying a given joint distribution $P(X)$. In this work, we follow a score-based approach where one defines a score or loss function $\mathcal{L}(f; \boldsymbol{X})$ that measures the "goodness of fit" of a candidate SCM to a data matrix $\boldsymbol{X}$. The goal is to learn a DAG that minimizes $\mathcal{L}(f; \boldsymbol{X})$, which can be any such as least squares, logistic loss or log-likelihood. The problem is generally NP-hard due to the combinatorial nature of the space of DAGs (Chickering, 1996; Chickering et al., 2004). Thanks to the existing proposals of smooth non-convex characterization of acyclicity (Zheng et al., 2018; Yu et al., 2019; Bello et al., 2022), score-based structure learning has successfully been cast as a continuous optimization problem that admits the use of gradient-based optimizers.

Here the graphical structure is represented by a weighted adjacency matrix $\mathbf{W} \in \mathbb{R}^{d\times d}$ where $i \to j$ if and only if $\mathbf{W}_{ij} \neq 0$. In the linear case, we can use $\mathbf{W} = [\mathbf{w}_1], \cdots, [\mathbf{w}_d]$ to define a SCM where (1) takes the form $X_i = \mathbf{w}_i^T X + U_i$. In the non-linear case, Zheng et al. (2018) propose to consider each f_j in the Sobolev space of square-integrable functions, denoted by $H^1(\mathbb{R}^d) \subset L^2(\mathbb{R}^d)$, whose derivatives are also square integrable. Let $\partial_i f_j$ denote the partial derivative of f_j w.r.t X_i. The dependency structure implied by the SCM $f = [f_i]_{i=1}^d$ can be encoded in a matrix $\mathbf{W} = \mathbf{W}(f) \in \mathbb{R}^{d\times d}$ with entries $[\mathbf{W}(f)]_{ij} := \|\partial_i f_j\|_2$, where $\|\cdot\|_2$ is the L^2 norm. Such construction is motivated from the property that f_j is independent of X_i if and only if $\|\partial_i f_j\|_2 = 0$ (Rosasco et al., 2013). In practice, f is approximated with a flexible family of parameterized function such as deep neural networks.

Learning the discrete graph $\mathbf{G}$ is now equivalent to optimizing for $\mathbf{W}$ in the continuous space of $d \times d$ real matrices. This allows for encoding the acyclicity constraint in $\mathbf{W}$ during optimization via a regularizer $R(\mathbf{W})$ such that $R(\mathbf{W}) = 0$ if and only $\mathbf{W} \in$ DAGs. We briefly review popular acyclicity characterizations in the following.

- **NOTEARS** (Zheng et al., 2018): $R(\mathbf{W}) = \mathrm{trace}\,[\exp(\mathbf{W} \circ \mathbf{W})] - d$.
- **Polynomial** (Yu et al., 2019): $R(\mathbf{W}) = \mathrm{trace}\,\big[(\mathbf{I} + \alpha \mathbf{W} \circ \mathbf{W})^d\big] - d$ for $\alpha > 0$.
- **DAGMA** (Bello et al., 2022): $R(\mathbf{W}) = -\log\det(s\mathbf{I} - \mathbf{W} \circ \mathbf{W}) + d\log s$ for $s > 0$.

where $\circ$ denotes the Hadamard product and $\mathbf{I}$ denotes the identity matrix.

2.3. Optimal Transport

Let $\alpha = \sum_{j=1}^n a_j \delta_{x_j}$ be a discrete measure with weights $\boldsymbol{a}$ and particles $x_1, \cdots, x^n \in \mathcal{X}$ where $\boldsymbol{a} \in \Delta^n$, a $(n-1)-$dimensional probability simplex. Let $\beta = \sum_{j=1}^n b_j \delta_{y_j}$ be another discrete measure defined similarly. The Kantorovich (Kantorovich, 2006) formulation of the OT distance between two discrete distributions α and β is

$$W_c\,(\alpha, \beta) := \inf_{\mathbf{P}\in\mathbb{U}(\boldsymbol{a},\boldsymbol{b})} \langle \mathbf{C}, \mathbf{P}\rangle, \tag{2}$$

where $\langle\cdot,\cdot\rangle$ denotes the Frobenius dot-product; $\mathbf{C} \in \mathbb{R}_+^{n\times n}$ is the cost matrix of the transport; $\mathbf{P} \in \mathbb{R}_+^{n\times n}$ is the transport matrix/plan; $\mathbb{U}(\boldsymbol{a}, \boldsymbol{b}) := \big\{\mathbf{P} \in \mathbb{R}_+^{n\times n} : \mathbf{P}\mathbf{1}_n = \boldsymbol{a}, \mathbf{P}\mathbf{1}_n = \boldsymbol{b}\big\}$ is the transport polytope of $\boldsymbol{a}$ and $\boldsymbol{b}$; $\mathbf{1}_n$ is the $n-$dimensional column of vector of ones. For arbitrary measures, Eq. (2) can be generalized as

$$W_c\,(\alpha; \beta) := \inf_{\Gamma\sim\mathcal{P}(X\sim\alpha, Y\sim\beta)} \mathbb{E}_{(X,Y)\sim\Gamma}\big[c(X, Y)\big], \tag{3}$$

where the infimum is taken over the set of all joint distributions $\mathcal{P}(X \sim \alpha, Y \sim \beta)$ with marginals α and β respectively. $c : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$ is any measurable cost function. If $c(x, y) = D^p(x, y)$ is a distance for $p \leq 1$, W_p, the p-root of W_c, is called the p-Wasserstein distance. Finally, for a measurable map $T : \mathcal{X} \mapsto \mathcal{Y}$, $T\#\alpha$ denotes the push-forward measure of α that, for any measurable set $B \subset \mathcal{Y}$, satisfies $T\#\alpha(B) = \alpha(T^{-1}(B))$. For discrete measures, the push-forward operation is $T\#\alpha = \sum_{j=1}^n a_j \delta_{T(x_j)}$.

2.4. Data with Missing Values

We consider a dataset of n data samples with d features stored in matrix $\boldsymbol{X} = \big[\boldsymbol{X}^1, \cdots, \boldsymbol{X}^n\big]^T \in \mathbb{R}^{n\times d}$. An indicator matrix $\boldsymbol{M} \in \{0, 1\}^{n\times d}$ is used to encode the missing

data so that $\boldsymbol{M}_i^j = 1$ indicates that the i^{th} feature of sample j is missing and $\boldsymbol{M}_i^j = 0$ otherwise. The missing values are initially assigned with Nan (Not a number). We follow Python/Numpy style matrix indexing $\boldsymbol{X}[\boldsymbol{M}] = \text{Nan}$ to denote the missing data.

The missing mechanism falls into one of the three common categories (Little & Rubin, 2019): missing completely at random (MCAR) where the cause of missingness is purely random and independent of the data; missing at random (MAR) where the probability of being missing depends only on observed values; missing not at random (MNAR) where the cause of missingness is unobserved. Dealing with MAR or MNAR is more challenging, as missing values may introduce significant biases into the analysis results (Mohan & Pearl, 2021; Kyono et al., 2021; Jarrett et al., 2022).

3. Methodology

3.1. Problem setup

We consider a causal DAG $\mathbf{G}(\mathbf{V}, \mathbf{E})$ on d random variables X induced by an underlying SCM over the tuple $\langle U, X, f \rangle$. Let θ denote the parameters of our model, **including** the weighted adjacency matrix $\mathbf{W}$ parametrizing the causal graph and the parameters of the functions approximating $f = [f_i]_{i=1}^d$. We will denote these functions as f_θ.

We focus on the finite-sample setting with missing data. Given a dataset $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{M})$, let $\mathrm{O} \subseteq [d]$ denote the set of entries/variables in each observation $\mathbf{X}^j$. Let $\boldsymbol{X}_\mathrm{O}^j = \left[\boldsymbol{X}_i^j : \boldsymbol{M}_i^j = 0, i \in [d]\right]$ denote the observed part of $\boldsymbol{X}^j$. Similarly, we have $\boldsymbol{X}_\mathrm{O} = \left[\boldsymbol{X}_i^j : \boldsymbol{M}_i^j = 0, i \in [d], j \in [n]\right]$ denote the entire observed part of the data.

Let $\mu_\mathcal{D}(\boldsymbol{X}_\mathrm{O}) = n^{-1} \sum_{j=1}^n \delta_{\boldsymbol{X}_\mathrm{O}^j}$ denote the (incomplete) empirical distribution over $\boldsymbol{X}_\mathrm{O}$. In the classic density-fitting problem where the data is complete, the task is to find θ that minimizes some distance/divergence between the model distribution and the empirical distribution over the observed data. This problem setup however does not translate smoothly to our setting of missing data since the definition of a model distribution over $\boldsymbol{X}_\mathrm{O}$, denoted as $\mu_\theta(\boldsymbol{X}_\mathrm{O})$, is not straightforward. Now we explain how to address this issue.

With a slight abuse of notation, we use $\boldsymbol{X}$ to denote the true (unknown) data matrix. Given a complete data matrix $\boldsymbol{X}$, we define h_m as a mechanism to produce missing data, or equivalently extracting the observed part, i.e., $h_m(\boldsymbol{X}) := \left[\boldsymbol{X}_i^j : \boldsymbol{M}_i^j = 0, i \in [d], j \in [n]\right] = \boldsymbol{X}_\mathrm{O}$ and h_m operates row-wisely according to $\boldsymbol{M}$. Given a subpart of data $\boldsymbol{X}_\mathrm{O}^j$, we construct a mechanism h_c to complete the data such that $h_c(\boldsymbol{X}_\mathrm{O}^j) = \widetilde{\boldsymbol{X}}^j \in \mathbb{R}^d$. For h_m and h_c defined above, we can define the model distribution over $\boldsymbol{X}_\mathrm{O}$ via its reconstruction from the model, that is $\mu_\theta(\boldsymbol{X}_\mathrm{O}) := n^{-1} \sum_{j=1}^n \delta_{h_m\{f_\theta[h_c(\boldsymbol{X}_\mathrm{O}^j)]\}}$.

We propose to learn the graph $\mathbf{G}$ using the OT cost as the score function. The goal is to find θ that minimizes the OT distance between $\mu_\mathcal{D}(\boldsymbol{X}_\mathrm{O})$ and $\mu_\theta(\boldsymbol{X}_\mathrm{O})$:

$$\hat{\theta} = \underset{\theta: f_\theta \in H^1(\mathbb{R}^d)}{\arg\min} W_c\left[\mu_\mathcal{D}(\boldsymbol{X}_\mathrm{O}), \mu_\theta(\boldsymbol{X}_\mathrm{O})\right] \quad \text{subject to } \mathbf{W}(f) \in \text{ DAGs}, \tag{4}$$

where the OT distance is given as

$$W_c\left[\mu_\mathcal{D}(\boldsymbol{X}_\mathrm{O}), \mu_\theta(\boldsymbol{X}_\mathrm{O})\right] = \min_{\mathbf{P} \in \mathbb{U}(\mathbf{1}_n, \mathbf{1}_n)} \sum_{j,k} c\left\{\boldsymbol{X}_\mathrm{O}^j, h_m\left[f_\theta\left(h_c(\boldsymbol{X}_\mathrm{O}^k)\right)\right]\right\} \mathbf{P}^{j,k}.$$

Note that $\mathbf{P} \in \mathbb{R}_+^{n \times n}$ is a coupling matrix and the minimum is taken over all possible couplings in the Birkhoff polytope $\mathbb{U}(\mathbf{1}_n/n, \mathbf{1}_n/n)$. The challenge now is how to define the cost $c(\cdot, \cdot)$ between two particles at locations (j, k), two arbitrary sub-vectors of possibly different dimensions. Notice that if the data matrix is complete i.e., each particle is a complete $d-$dimensional vector, the cost function can be defined naturally via a metric.

Definition 3.1. (Cost function for incomplete samples) The transport cost between a particle j from $\mu_\mathcal{D}(\boldsymbol{X}_\mathrm{O})$ and a particle k from $\mu_\theta(\boldsymbol{X}_\mathrm{O})$ is given by

$$c\left\{\boldsymbol{X}_\mathrm{O}^j, h_m\left[f_\theta\left(h_c(\boldsymbol{X}_\mathrm{O}^k)\right)\right]\right\} := d_c\left\{h_c\left(\boldsymbol{X}_\mathrm{O}^j\right), f_\theta\left[h_c(\boldsymbol{X}_\mathrm{O}^k)\right]\right\} \quad \forall j, k \in [n], \tag{5}$$

where d_c is a metric between two complete vectors in $\mathbb{R}^d$.

By Definition 3.1, we propose to define the cost $c(\cdot, \cdot)$ between two incomplete data points via a distance between their pseudo-completions from h_c. As an initializer, h_c can be any off-the-shelf imputer or a measurable function with learnable parameters to be optimized via an objective. The construction of an imputation mechanism h_c to create pseudo-complete values is a natural strategy to deal with missing data in this line of research. Muzellec et al. (2020) and Zhao et al. (2023) in particular propose a strategy for learning the imputations by minimizing the OT distance between two pseudo-complete batch of data points. In these works, the imputations for the missing entries are themselves parameters and h_c can be chosen to be such a mechanism.

Let $\mu_\theta(\boldsymbol{X}) := n^{-1} \sum_{j=1}^n \delta_{f_\theta(\boldsymbol{X}^j)}$ denote the model distribution over the true data.

Lemma 3.2. *For h_c, h_m defined as above, if h_c is optimal in the sense that h_c recovers the original data i.e., $h_c(\boldsymbol{X}_\mathrm{O}^j) = \boldsymbol{X}^j, \forall j \in [n]$, we have*

$$W_c\left[\mu_\mathcal{D}(\boldsymbol{X}_\mathrm{O}), \mu_\theta(\boldsymbol{X}_\mathrm{O})\right] = W_{d_c}\left[h_c \# \mu_\mathcal{D}(\boldsymbol{X}_\mathrm{O}), \mu_\theta(\boldsymbol{X})\right].$$

where $h_c \# \mu_{\mathcal{D}}(\boldsymbol{X}_{\mathrm{O}}) = n^{-1} \sum_{j=1}^{n} \delta_{h_c(\boldsymbol{X}_{\mathrm{O}}^j)}$*, which also represents empirical distribution over the true data.*

Proof. See Appendix A.

Lemma 3.2 follows directly from Definition 3.1, which translates the problem of minimizing the OT distance between two distributions over the *incomplete* data into minimizing the OT distance between two distributions over the *complete* data produced by h_c. Furthermore, if h_c optimally recovers the true data, the minimizer of the OP in (4) coincides with the solution θ "best" fits the true empirical data distribution.

For brevity, let us alternatively define an empirical distribution $\mu(\widetilde{\boldsymbol{X}}) = n^{-1} \sum_{j=1}^{n} \delta_{\widetilde{\boldsymbol{X}}^j}$ over the set of samples $\left\{ \widetilde{\boldsymbol{X}}^j = h_c(\boldsymbol{X}_{\mathrm{O}}^j) \right\}_{j=1}^{n}$. Lemma 3.2 motivates an OP for learning θ by minimizing $W_{d_c}\left[\mu(\widetilde{\boldsymbol{X}}); \mu_\theta(\boldsymbol{X})\right]$, where $\mu_\theta(\boldsymbol{X})$ is again the model distribution over the true data. On one hand, we cannot compute this OT distance directly because $\boldsymbol{X}$ is unknown. On the other hand, Eq. (5) suggests that one can minimize $W_{d_c}\left[\mu(\widetilde{\boldsymbol{X}}); f\#\mu(\widetilde{\boldsymbol{X}})\right]$ since fitting θ to $\widetilde{\boldsymbol{X}}$ is the same as fitting θ to true data $\boldsymbol{X}$ if h_c is optimal. However, this is rarely the case in practice because (1) from §1, we learn that the existing imputation methods are sub-optimal for learning the true causal model; (2) if h_c is learnable, that h_c is arbitrarily initialized provides no useful information about the true data or model distribution for θ to be updated effectively over training.

3.2. Proposed method

We now present our theoretical contribution that opens the door to solving the prevailing problem via a more sensible optimization objective.

Theorem 3.3. *Let* $\phi : \mathbb{R}^{n\times d} \mapsto \mathbb{R}^{n\times d}$ *be a stochastic map such that* $\phi\#\mu(\widetilde{\boldsymbol{X}}) = \mu_\theta(\boldsymbol{X})$ *and* $h_m\left[\phi(\widetilde{\boldsymbol{X}})\right] = \boldsymbol{X}_{\mathrm{O}}$*. For a fixed value of* θ*,*

$$W_{d_c}\left[\mu(\widetilde{\boldsymbol{X}}); \mu_\theta(\boldsymbol{X})\right] = \min_{\phi} \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}), \boldsymbol{Y}\sim\phi(\widetilde{\boldsymbol{X}})}\left[d_c\left(\widetilde{\boldsymbol{X}}, f_\theta(\boldsymbol{Y})\right)\right], \quad (6)$$

where d_c *is a metric between two complete vectors in* $\mathbb{R}^d$*.*

Proof. See Appendix A.

The push-forward operation $\phi\#\mu(\widetilde{\boldsymbol{X}}) = \mu_\theta(\boldsymbol{X})$ dictates the existence of a function ϕ that transports the pseudo-complete distribution $\mu(\widetilde{\boldsymbol{X}})$ to the model distribution such that the OT distance $W_{d_c}\left[\mu(\widetilde{\boldsymbol{X}}); \mu_\theta(\boldsymbol{X})\right]$ is tractable for any initializer h_c. The second condition $h_m\left[\phi(\widetilde{\boldsymbol{X}})\right] = \boldsymbol{X}_{\mathrm{O}}$ is to restrict ϕ to the class of functions that preserves the observed part of the data. To understand Theorem 3.3, let us recall the data generative mechanism according to an SCM, where a realization from $\mu_\theta(\boldsymbol{X})$ is produced by first obtaining values for the root nodes and then generating the data for the remaining nodes via ancestral sampling. Furthermore, if a sample $\boldsymbol{X}^j$ is indeed generated by the model, we should be able to reconstruct $\boldsymbol{X}^j$ via the mechanisms f_θ. At a certain θ, the optimal ϕ ensures the available data is actually generated from the SCM induced by θ. If ϕ is at optimality, optimizing (6) is equivalent to finding θ that can reconstruct the observed part of the data with minimal cost.

We first explain how to optimize for ϕ and elaborate on the intuition behind its construction in the next section. For fixed θ, the push-forward constraint ensures every sample $\boldsymbol{Y}^j \sim \phi(\widetilde{\boldsymbol{X}}^j)$ is to be generated according to the current SCM or more concretely, from the value of the root node in the current causal graph. Note that from one root data value, the model can generate many different samples sharing the same joint distribution. Optimizing such a mapping ϕ forces $\boldsymbol{Y}$ to have the same distribution with $f_\theta(\boldsymbol{Y})$ at the current θ. Let $\mu_\phi(\boldsymbol{Y}) := \mathbb{E}_{\widetilde{\boldsymbol{X}}}\left[\phi(\boldsymbol{Y}|\widetilde{\boldsymbol{X}})\right]$ be the marginal distribution induced by ϕ. By the above logic, we relax the constraint into minimizing some divergence between μ_ϕ and $f\#\mu_\phi$ where $f\#\mu_\phi(\boldsymbol{Y}) := n^{-1}\sum_{j=1}^{n}\delta_{f_\theta(\boldsymbol{Y}^j)}$ for $\boldsymbol{Y}^j \sim \mu_\theta(\boldsymbol{Y})$. We can then make our OP unconstrained by adding this penalty to (6).

Our **final optimization objective** is given by

$$\mathcal{L}(\phi;\theta) = \min_{\phi,\theta} \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}), \boldsymbol{Y}\sim\phi(\widetilde{\boldsymbol{X}})}\left[d_c\left(\widetilde{\boldsymbol{X}}, f_\theta(\boldsymbol{Y})\right)\right] + \lambda\, D\left(\mu_\phi; f\#\mu_\phi\right) + \gamma_1\, R(\mathbf{W}) + \gamma_2\, \|\mathbf{W}\|_1, \quad (7)$$

where $\lambda, \gamma_1, \gamma_2 > 0$ are trade-off hyper-parameters; $R(\mathbf{W})$ is any acyclicity regularizer of choice and $\|\mathbf{W}\|_1$ is to encourage sparsity; D is an arbitrary divergence measure which can be optimized with maximum mean discrepancy (MMD), the Wasserstein distance or adversarial training.

How OTM works. We now explain the intuition behind the objective in (7). Theoretically, the key role of ϕ is to render a tractable formulation for our desired OT distance in (6) when it is difficult to obtain samples from $\mu_\theta(\boldsymbol{X})$. To understand how ϕ supports optimization, we can view ϕ as a "correction" network that rectifies the random imputations made by h_c. Lemma 3.2 implies that the closer h_c is to optimality, meaning the imputed data well approximates the true data, the more effectively the model is to be optimized. Conversely, that θ optimally yields a distribution, from which the true data is actually generated, would also support the optimization of h_c. To facilitate this desired dynamic between ϕ and θ, what ϕ essentially does is push h_c closer to optimality by concretely forcing the imputed data to obey the dependencies given by the current SCM. This would

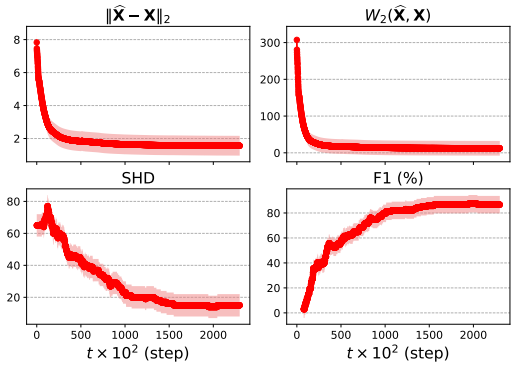

Figure 2. Visualization of the optimization process of OTM. $\mathbf{X}$ is the ground-true complete data. $\widehat{\mathbf{X}} = f_\theta\left[\phi(\mathbf{X}_\mathrm{O})\right]$ is estimated complete data from the model. **Top:** As training progresses, the model generates imputations that are closer to the original data both by Euclidean distance (value-wise) and Wasserstein distance (distribution-wise). **Bottom:** The quality of the estimated graph improves accordingly over training.

result from the effect of the second term in (7). Without this *distribution matching* term, intuitively, optimizing θ alone on the first term (for element-wise reconstruction) would be meaningless since the output imputation from h_c is often non-informative. Our empirical evidence suggests that such a dynamic takes place in our OTM framework where ϕ and θ supports one another to converge to a solution closest to the ground-truth. To further understand how OTM works, we visualize the training behavior in Figure 2. By minimizing the OT cost in (6), OTM seeks to find an SCM that generates data that are "most similar" to the original data both in terms of reconstruction (Euclidean distance) and distribution (Wasserstein distance). The figure shows that the causal structure resulting from such an SCM indeed approaches the ground-true one. Note that the true original data remains unknown during training. The objective (7) facilitates a joint optimization procedure over both ϕ and θ, which can be solved with gradient-based methods. It is worth noting that our formulation so far has not assumed any particular form for the functions f.

Practical implementation. The objective (7) applies to any choice of regularizer $R(\mathbf{W})$. This means that our algorithm can accommodate any existing score-based causal discovery algorithms with complete data. In our experiment, we choose DAGMA (Bello et al., 2022) as our base causal discovery method, which supports efficient optimization with stochastic gradient descent. For the distance d_c, we use MSE-based function, which are also the default score functions implemented in DAGMA. For the divergence measure, we use MMD[2] with RBF kernel which is found to work best in most of our experiments.

[2]See Eq. (11) for the formulation of MMD.

For the purpose of computational convenience, we approximate $\boldsymbol{X}_\mathrm{O} \approx \left[\boldsymbol{X}_i^j \text{ if } \boldsymbol{M}_i^j = 0 \text{ else } 0, i \in [d], j \in [n]\right]$: $\boldsymbol{X}_\mathrm{O} \in \mathbb{R}^{n \times d}$ by filling out zeros at the missing entries. Since both functions h_c and ϕ play the role of an missing value imputer, using amortized optimization technique (Amos et al., 2023), we can combine them into a "super" imputer and model it with a sufficiently expressive deep neural network that acts globally on the entire dataset. We therefore construct the function $\phi : \mathbb{R}^{n \times d} \mapsto \mathbb{R}^{n \times d}$ as $\phi(\boldsymbol{X}_\mathrm{O}) = \mathsf{NN}(\boldsymbol{X}_\mathrm{O}) \circ \boldsymbol{M} + \boldsymbol{X}_\mathrm{O}$, where $\mathsf{NN}(\cdot)$ is a neural network that returns a distribution over complete data values given $\boldsymbol{X}_\mathrm{O}$. Following the definition of ϕ, here only the missing entries in the data matrix are filled while the values at the observed entries are retained. The training procedure of OTM is summarized in Algorithm 1.

Algorithm 1 OTM Algorithm

Input: Incomplete data matrix $\boldsymbol{X} = \left[\boldsymbol{X}^1, \cdots, \boldsymbol{X}^n\right]^T \in \mathbb{R}^{n \times d}$; missing mask $\boldsymbol{M}$, regularization coefficients $\lambda, \gamma_1, \gamma_2 > 0$; distance function d_c; characteristic positive-definite kernel κ and $M-$matrix domain $s > 0$.
Output: Weighted adjacency matrix $\mathbf{W} \in \theta$.
Initialize the parameters θ, ϕ.
while (ϕ, θ) *not converged* **do**
  Set $\boldsymbol{X}_\mathrm{O} = \left[\boldsymbol{X}_i^j \text{ if } \boldsymbol{M}_i^j = 0 \text{ else } 0, i \in [d], j \in [n]\right]$;
  Sample $\widetilde{\boldsymbol{X}}$ from $\phi\left(\boldsymbol{X}_\mathrm{O}\right)$;
  Evaluate $\boldsymbol{Y} = f_\theta(\widetilde{\boldsymbol{X}})$;
  Update ϕ, θ by descending

$$\mathcal{L}(\phi, \theta) = \frac{1}{n}\sum_{j=1}^{n} d_c\left(\widetilde{\boldsymbol{X}}^j, \boldsymbol{Y}^j\right) + \lambda\, \mathsf{MMD}(\widetilde{\boldsymbol{X}}, \boldsymbol{Y}, \kappa)$$
$$+ \gamma_1\left[-\log\det\left(s\mathbf{I} - \mathbf{W} \circ \mathbf{W}\right) + d\log s\right] + \gamma_2 \left\|\mathbf{W}\right\|_1.$$

end while

4. Experiments

We evaluate OTM[3] on both synthetic and real-world datasets. The detailed implementations of all methods are presented in Appendix C. We focus on the non-linear setting in the main text since it is more challenging. Additional results on the linear models are reported in Appendix D.

Baselines. We consider imputation methods as baselines, including Mean imputation, Iterative imputation (Van Buuren & Oudshoorn, 2000; Pedregosa et al., 2011), and OT imputation (Batch Sinkhorn and Round-Robin Sinkhorn imputation) (Muzellec et al., 2020) to fill in the missing data and then apply DAGMA (Bello et al., 2022) for structure

[3]Our code is anonymously published at https://github.com/isVy08/causal-discovery-missing-data.

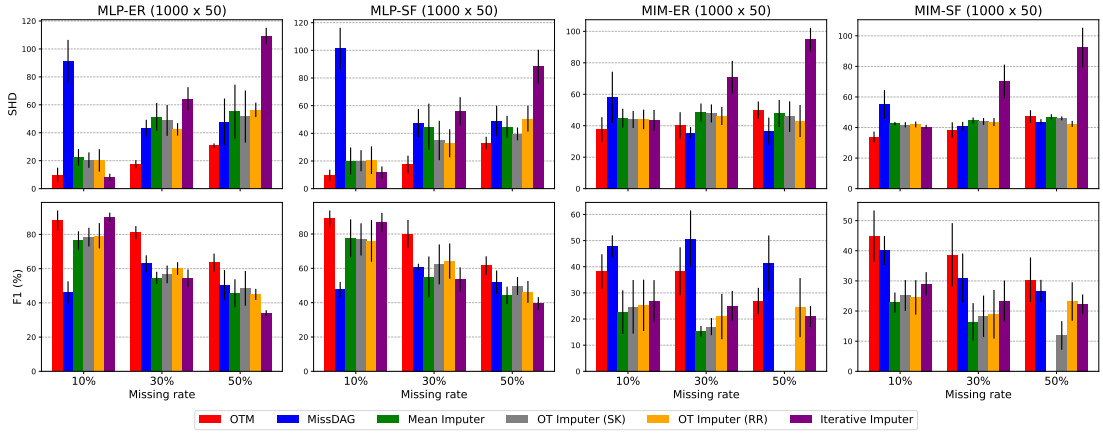

Figure 3. Nonlinear ANMs (MCAR). SHD ↓ and F1 ↑.

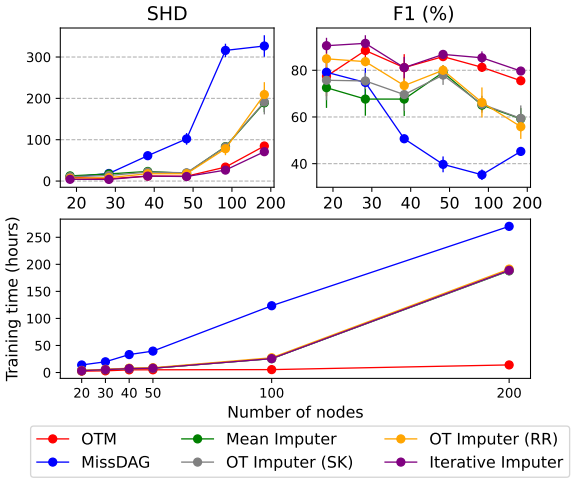

Figure 4. Scalability of methods in nonlinear ANMs (MCAR) at 10% missing rate. SHD ↓ and F1 ↑. The training time of the imputation baselines includes the time for learning imputations.

learning. We further compare OTM with MissDAG (Gao et al., 2022) as the key competing method, which has already been shown to outperform many constraint-based baselines. We do not report results for VISL (Morales-Alvarez et al., 2022) as they do not provide reproducible codes.

Thresholding & Metrics. Following the standard causal discovery practice (Zheng et al., 2018; Ng et al., 2020), a final thresholding of 0.3 is applied post-training to ensure the DAG output: it iteratively removes the edge with the minimum magnitude until the final graph is a DAG. All quantitative results are averaged over 5 random initializations. For comparing the estimated DAG with the ground-truth one, we report the commonly used metrics: F1-score and Structural Hamming Distance (SHD) with SHD referring to the smallest number of edge additions, deletions, and reversals required to transform the recovered DAG into the true one. Lower SHD is preferred (↓) while higher F1 is preferred (↑). Note that SHD is a standard metric used across causal discovery literature, but it can be biased. Because SHD quantifies the number of errors in absolute value, given a sparse graph, a method could achieve low SHD by predicting few edges, which obviously would compromise the accuracy score. Therefore, one must examine both metrics to assess the causal discovery performance thoroughly.

4.1. Simulations

We simulate synthetic datasets generating a ground-true DAG from one of the two graph models, Erdos-Rényi (ER) or Scale-Free (SF). Each function f_i is constructed from a multi-layer perceptron (MLP) and a multiple index model (MIM) with random coefficients, corresponding to an edge in the causal graph. We consider a general scenario of non-equal variances, sampling 1000 observations according to all missing mechanisms: MCAR, MAR and MNAR at $10\%, 30\%, 50\%$ missing rates. In the main text, we report the results in MCAR cases with the standard setting of 50 nodes and 100 directed edges, using Gaussian noise distribution. More results on MAR, MNAR cases as well as experiments with different number of nodes, degrees and noise distributions are reported in Appendix D.

Results. The effectiveness of OTM across various settings is demonstrated in Figures 3, 6 and 7. OTM achieves consistently low SHD scores with the highest/second-highest F1-scores. Sub-optimality is again observed in the simulations when a causal discovery method is applied on top of existing imputation baselines. Throughout our experiments, Iterative imputer still produces the best imputation quality and works best when the missing mechanism is MCAR and missing rate is as low as 10%. However, the method exhibits inconsistencies in the quality of the estimated DAGs in the other settings. This implies that one cannot rely on the imputation quality as a proxy for causal discovery perfor-

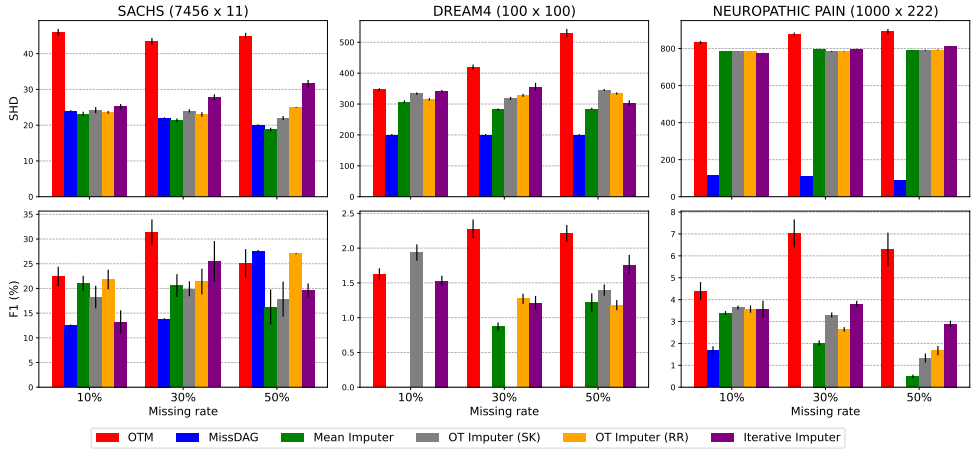

Figure 5. Real-world datasets (MCAR). SHD ↓ and F1 ↑.

mance under missing data. Despite the good performance on linear settings (See Figures 11), MissDAG is problematic in our non-linear settings. The key challenge arises from the intractability of the posterior distribution and non-trivial computation of likelihood expectation. MissDAG resorts to approximate posterior, concretely rejection sampling to fill in the missing values in the E steps. Our empirical evidence reveals that MissDAG still performs adequately in the simulated settings with possibly minor misspecifications. However, as we transition to real-world cases, the challenges become more pronounced, highlighting the limitations of the sampling process in practical scenarios.

4.2. Real-world datasets

We evaluate OTM on 3 well-known biological datasets with ground-true causal relations. The first one is the Neuropathic Pain dataset (Tu et al., 2019), containing diagnosis records of neuropathic pain patients. There are 1000 samples and 222 binary variables, indicating existence of the symptoms. The second causal graph named Sachs (Sachs et al., 2005) models a network of cellular signals, consisting of 11 continuous variables and 7466 samples. Dream4 (Greenfield et al., 2010; Marbach et al., 2010) is the last dataset which simulates gene expression measurements from five transcriptional regulator sub-networks of E. coli and S. cerevisiae. The provided graphs contain feedback loops and there are 100 continuous variables and 100 samples.

Results. It is worth noting that the performance of all methods is comparatively lower compared with one in the simulated settings. This is because in real-world scenarios, the model is prone to mis-specification, and assumptions like acyclicity or the absence of latent confounders can be easily violated. However, it is observed from Figures 5, 8 and 9 that OTM stands out from the baselines with superior accuracy. Meanwhile, MissDAG fails to detect the true edge directions in several settings on Neuropathic pain and Dream4 datasets. In these experiments, we find that its underlying causal discovery method, NOTEARS, does not converge in every EM iteration on the samples obtained from approximate posterior. An additional limitation of MissDAG pertains to the scalability of the method. As illustrated in Figure 4, our approach exhibits faster run-time compared to MissDAG. This enhanced efficiency can be attributed to two key differences between OTM and MissDAG. Firstly, OTM runs on DAGMA, the superior scalability of which over NOTEARS has been established in Bello et al. (2022). The main reason for this is that NOTEARS utilizes second-order optimization, while DAGMA leverages first-order optimization, which still yields good performance in practical settings. Second, the iterative nature of EM algorithm and the incorporation of rejection sampling in the E step contribute to the overall computational burden of MissDAG. Figure 4 demonstrates the undesirability of MissDAG when the number of nodes increases, wherein the performance of MissDAG degrades greatly while being extremely more timely expensive. While Figure 4 considers synthetic datasets, this sub-optimal behavior translates to the real-world settings where MissDAG fails on Neuropathic Pain and Dream4 datasets of high dimensionality.

5. Conclusion and Future Work

We propose OTM, a framework based on optimal transport for DAG learning under missing data. We reaffirm that subsequently performing causal discovery on the pre-imputed data is sub-optimal, whereas the existing end-to-end framework exhibits inefficacy on more complex graphs. OTM is shown to be more robust and scalable approach while being flexible enough to accommodate any existing score-based causal discovery algorithms. Our work contributes a promising avenue for further applications of OT in causal discovery. Our future research aims to address more challenging set-

tings wherein OTM remains limited, such as cyclic causal graphs or graphs with latent confounders.

Acknowledgements

Trung Le and Dinh Phung were supported by ARC DP23 grant DP230101176 and by the Air Force Office of Scientific Research under award number FA2386-23-1-4044.

References

Adel, T. and de Campos, C. Learning bayesian networks with incomplete data by augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Akbari, S. and Ganassali, L. Learning causal graphs via monotone triangular transport maps. *arXiv preprint arXiv:2305.18210*, 2023.

Amos, B. et al. Tutorial on amortized optimization. *Foundations and Trends® in Machine Learning*, 16(5):592–732, 2023.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.

Ashman, M., Ma, C., Hilmkil, A., Jennings, J., and Zhang, C. Causal reasoning in the presence of latent confounders via neural admg learning. In *The Eleventh International Conference on Learning Representations*, 2022.

Bassetti, F., Bodini, A., and Regazzini, E. On minimum kantorovich distance estimators. *Statistics & probability letters*, 76(12):1298–1302, 2006.

Bello, K., Aragam, B., and Ravikumar, P. Dagma: Learning dags via m-matrices and a log-determinant acyclicity characterization. *Advances in Neural Information Processing Systems*, 35:8226–8239, 2022.

Bernton, E., Jacob, P. E., Gerber, M., and Robert, C. P. On parameter estimation with the wasserstein distance. *Information and Inference: A Journal of the IMA*, 8(4): 657–676, 2019.

Bui, A. T., Le, T., Tran, Q. H., Zhao, H., and Phung, D. A unified wasserstein distributional robustness framework for adversarial training. In *International Conference on Learning Representations*, 2021.

Canas, G. and Rosasco, L. Learning probability measures with respect to optimal transport metrics. *Advances in Neural Information Processing Systems*, 25, 2012.

Charpentier, B., Kibler, S., and Günnemann, S. Differentiable dag sampling. *arXiv preprint arXiv:2203.08509*, 2022.

Chickering, D. M. Learning bayesian networks is np-complete. *Learning from data: Artificial intelligence and statistics V*, pp. 121–130, 1996.

Chickering, D. M. Optimal structure identification with greedy search. *Journal of machine learning research*, 3 (Nov):507–554, 2002.

Chickering, M., Heckerman, D., and Meek, C. Large-sample learning of bayesian networks is np-hard. *Journal of Machine Learning Research*, 5:1287–1330, 2004.

Cussens, J., Haws, D., and Studenỳ, M. Polyhedral aspects of score equivalence in bayesian network structure learning. *Mathematical Programming*, 164:285–324, 2017.

Dai, Z., Bu, Z., and Long, Q. Multiple imputation via generative adversarial network for high-dimensional blockwise missing value problems. In *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 791–798. IEEE, 2021.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum Likelihood from Incomplete Data Via the EM Algorithm . *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977. doi: 10.1111/j.2517-6161.1977.tb01600.x.

Deng, C., Bello, K., Aragam, B., and Ravikumar, P. K. Optimizing notears objectives via topological swaps. In *International Conference on Machine Learning*, pp. 7563–7595. PMLR, 2023.

Fang, F. and Bao, S. Fragmgan: generative adversarial nets for fragmentary data imputation and prediction. *Statistical Theory and Related Fields*, pp. 1–14, 2023.

Friedman, N. et al. Learning belief networks in the presence of missing values and hidden variables. In *Icml*, volume 97, pp. 125–133. Berkeley, CA, 1997.

Gain, A. and Shpitser, I. Structure learning under missing data. In *International conference on probabilistic graphical models*, pp. 121–132. PMLR, 2018.

Gao, E., Ng, I., Gong, M., Shen, L., Huang, W., Liu, T., Zhang, K., and Bondell, H. Missdag: Causal discovery in the presence of missing data with continuous additive noise models. *Advances in Neural Information Processing Systems*, 35:5024–5038, 2022.

Gao, Z., Niu, Y., Cheng, J., Tang, J., Li, L., Xu, T., Zhao, P., Tsung, F., and Li, J. Handling missing data via max-entropy regularized graph autoencoder. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 7651–7659, 2023.

Geffner, T., Antoran, J., Foster, A., Gong, W., Ma, C., Kiciman, E., Sharma, A., Lamb, A., Kukla, M., Pawlowski, N., et al. Deep end-to-end causal inference. *arXiv preprint arXiv:2202.02195*, 2022.

Glymour, C., Zhang, K., and Spirtes, P. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019.

Greenfield, A., Madar, A., Ostrer, H., and Bonneau, R. Dream4: Combining genetic and dynamic information to identify biological networks and dynamical models. *PloS one*, 5(10):e13397, 2010.

Heckerman, D., Meek, C., and Cooper, G. A bayesian approach to causal discovery. *Innovations in Machine Learning: Theory and Applications*, pp. 1–28, 2006.

Hoyer, P., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008.

Jarrett, D., Cebere, B. C., Liu, T., Curth, A., and van der Schaar, M. Hyperimpute: Generalized iterative imputation with automatic model selection. In *International Conference on Machine Learning*, pp. 9916–9937. PMLR, 2022.

Kantorovich, L. V. On a problem of monge. *Journal of Mathematical Sciences*, 133(4):1383–1383, 2006.

Kyono, T., Zhang, Y., Bellot, A., and van der Schaar, M. Miracle: Causally-aware imputation via learning missing data mechanisms. *Advances in Neural Information Processing Systems*, 34:23806–23817, 2021.

Lachapelle, S., Brouillard, P., Deleu, T., and Lacoste-Julien, S. Gradient-based neural dag learning. In *International Conference on Learning Representations*, 2019.

Little, R. J. and Rubin, D. B. *Statistical analysis with missing data*, volume 793. John Wiley & Sons, 2019.

Liu, J., Gelman, A., Hill, J., Su, Y.-S., and Kropko, J. On the stationary distribution of iterative imputations. *Biometrika*, 101(1):155–173, 2014.

Lorch, L., Rothfuss, J., Schölkopf, B., and Krause, A. Dibs: Differentiable bayesian structure learning. *Advances in Neural Information Processing Systems*, 34:24111–24123, 2021.

Lorch, L., Sussex, S., Rothfuss, J., Krause, A., and Schölkopf, B. Amortized inference for causal structure learning. *Advances in Neural Information Processing Systems*, 35:13104–13118, 2022.

Marbach, D., Prill, R. J., Schaffter, T., Mattiussi, C., Floreano, D., and Stolovitzky, G. Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of the national academy of sciences*, 107(14): 6286–6291, 2010.

Mattei, P.-A. and Frellsen, J. Miwae: Deep generative modelling and imputation of incomplete data sets. In *International conference on machine learning*, pp. 4413–4423. PMLR, 2019.

Mohan, K. and Pearl, J. Graphical models for processing missing data. *Journal of the American Statistical Association*, 116(534):1023–1037, 2021.

Morales-Alvarez, P., Gong, W., Lamb, A., Woodhead, S., Peyton Jones, S., Pawlowski, N., Allamanis, M., and Zhang, C. Simultaneous missing value imputation and structure learning with groups. *Advances in Neural Information Processing Systems*, 35:20011–20024, 2022.

Muzellec, B., Josse, J., Boyer, C., and Cuturi, M. Missing data imputation using optimal transport. In *International Conference on Machine Learning*, pp. 7130–7140. PMLR, 2020.

Nazabal, A., Olmos, P. M., Ghahramani, Z., and Valera, I. Handling incomplete heterogeneous data using vaes. *Pattern Recognition*, 107:107501, 2020.

Ng, I., Ghassami, A., and Zhang, K. On the role of sparsity and dag constraints for learning linear dags. *Advances in Neural Information Processing Systems*, 33:17943–17954, 2020.

Ng, I., Zhu, S., Fang, Z., Li, H., Chen, Z., and Wang, J. Masked gradient-based causal structure learning. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*, pp. 424–432. SIAM, 2022.

Nguyen, T., Nguyen, V., Le, T., Zhao, H., Tran, Q. H., and Phung, D. Cycle class consistency with distributional optimal transport and knowledge distillation for unsupervised domain adaptation. In *Uncertainty in Artificial Intelligence*, pp. 1519–1529. PMLR, 2022.

Ott, S. and Miyano, S. Finding optimal gene networks using biological constraints. *Genome Informatics*, 14:124–133, 2003.

Pamfil, R., Sriwattanaworachai, N., Desai, S., Pilgerstorfer, P., Georgatzis, K., Beaumont, P., and Aragam, B. Dynotears: Structure learning from time-series data. In *International Conference on Artificial Intelligence and Statistics*, pp. 1595–1605. PMLR, 2020.

Pearl, J. *Causality*. Cambridge university press, 2009.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Peis, I., Ma, C., and Hernández-Lobato, J. M. Missing data imputation and acquisition with deep hierarchical models and hamiltonian monte carlo. *Advances in Neural Information Processing Systems*, 35:35839–35851, 2022.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. Causal discovery with continuous additive noise models. 2014.

Peyré, G., Cuturi, M., et al. Computational optimal transport. *Center for Research in Economics and Statistics Working Papers*, (2017-86), 2017.

Richardson, T. W., Wu, W., Lin, L., Xu, B., and Bernal, E. A. Mcflow: Monte carlo flow models for data imputation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14205–14214, 2020.

Richens, J. G., Lee, C. M., and Johri, S. Improving the accuracy of medical diagnosis with causal machine learning. *Nature communications*, 11(1):3923, 2020.

Rosasco, L. A., Villa, S., Mosci, S., Santoro, M., and Verri, A. Nonparametric sparsity and regularization. 2013.

Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D. A., and Nolan, G. P. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721): 523–529, 2005.

Santambrogio, F. Optimal transport for applied mathematicians. *Birkäuser, NY*, 55(58-63):94, 2015.

Singh, M. Learning bayesian networks from incomplete data. *AAAI/IAAI*, 1001:534–539, 1997.

Spirtes, P. and Glymour, C. An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, 9(1):62–72, 1991.

Spirtes, P., Glymour, C. N., and Scheines, R. *Causation, prediction, and search*. MIT press, 2000.

Strobl, E. V., Visweswaran, S., and Spirtes, P. L. Fast causal inference with non-random missingness by test-wise deletion. *International journal of data science and analytics*, 6:47–62, 2018.

Teyssier, M. and Koller, D. Ordering-based search: A simple and effective algorithm for learning bayesian networks. *arXiv preprint arXiv:1207.1429*, 2012.

Tolstikhin, I., Bousquet, O., Gelly, S., and Schoelkopf, B. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, 2017.

Tu, R., Zhang, C., Ackermann, P., Mohan, K., Kjellström, H., and Zhang, K. Causal discovery in the presence of missing data. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1762–1770. PMLR, 2019.

Tu, R., Zhang, K., Kjellstrom, H., and Zhang, C. Optimal transport for causal discovery. In *International Conference on Learning Representations*, 2021.

Van Buuren, S. and Groothuis-Oudshoorn, K. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45:1–67, 2011.

Van Buuren, S. and Oudshoorn, C. G. Multivariate imputation by chained equations, 2000.

Van Buuren, S., Brand, J. P., Groothuis-Oudshoorn, C. G., and Rubin, D. B. Fully conditional specification in multivariate imputation. *Journal of statistical computation and simulation*, 76(12):1049–1064, 2006.

Villani, C. et al. *Optimal transport: old and new*, volume 338. Springer, 2009.

Vo, V., Le, T., Vuong, L.-T., Zhao, H., Bonilla, E., and Phung, D. Learning directed graphical models with optimal transport. *arXiv preprint arXiv:2305.15927*, 2023.

Vuong, T.-L., Le, T., Zhao, H., Zheng, C., Harandi, M., Cai, J., and Phung, D. Vector quantized wasserstein auto-encoder. *arXiv preprint arXiv:2302.05917*, 2023.

Wang, X., Du, Y., Zhu, S., Ke, L., Chen, Z., Hao, J., and Wang, J. Ordering-based causal discovery with reinforcement learning. *arXiv preprint arXiv:2105.06631*, 2021.

Wang, Y., Liang, D., Charlin, L., and Blei, D. M. Causal inference for recommender systems. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pp. 426–431, 2020.

Yoon, J., Jordon, J., and Schaar, M. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*, pp. 5689–5698. PMLR, 2018.

Yoon, S. and Sull, S. Gamin: Generative adversarial multiple imputation network for highly missing data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8456–8464, 2020.

You, J., Ma, X., Ding, Y., Kochenderfer, M. J., and Leskovec, J. Handling missing data with graph representation learning. *Advances in Neural Information Processing Systems*, 33:19075–19087, 2020.

Yu, Y., Chen, J., Gao, T., and Yu, M. Dag-gnn: Dag structure learning with graph neural networks. In *International Conference on Machine Learning*, pp. 7154–7163. PMLR, 2019.

Yuan, C., Malone, B., and Wu, X. Learning optimal bayesian networks using a* search. In *Twenty-second international joint conference on artificial intelligence*, 2011.

Zhang, B., Gaiteri, C., Bodea, L.-G., Wang, Z., McElwee, J., Podtelezhnikov, A. A., Zhang, C., Xie, T., Tran, L., Dobrin, R., et al. Integrated systems approach identifies genetic nodes and networks in late-onset alzheimer's disease. *Cell*, 153(3):707–720, 2013.

Zhang, C., Cai, Y., Lin, G., and Shen, C. Deepemd: Differentiable earth mover's distance for few-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5632–5648, 2022.

Zhao, H., Phung, D., Huynh, V., Le, T., and Buntine, W. Neural topic model via optimal transport. In *International Conference on Learning Representations*, 2020.

Zhao, H., Sun, K., Dezfouli, A., and Bonilla, E. V. Transformed distribution matching for missing value imputation. In *International Conference on Machine Learning*, pp. 42159–42186. PMLR, 2023.

Zheng, X., Aragam, B., Ravikumar, P. K., and Xing, E. P. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.

Zheng, X., Dan, C., Aragam, B., Ravikumar, P., and Xing, E. Learning sparse nonparametric dags. In *International Conference on Artificial Intelligence and Statistics*, pp. 3414–3425. PMLR, 2020.

Zhu, J. and Raghunathan, T. E. Convergence properties of a sequential regression multiple imputation algorithm. *Journal of the American Statistical Association*, 110(511): 1112–1124, 2015.

Zhu, S., Ng, I., and Chen, Z. Causal discovery with reinforcement learning. *arXiv preprint arXiv:1906.04477*, 2019.

A. Proofs

To make the proof self-contained, we here recap the problem setup in the main text.

Let $\boldsymbol{X}$ be the true (unknown) data matrix and $\boldsymbol{X}_{\mathrm{O}}$ denote the observed part of the data that is available. Given a complete matrix $\boldsymbol{X}$, we define h_m as a mechanism for producing the missing data, or equivalently extracting the observed part i.e., $h_m(\boldsymbol{X}) = \boldsymbol{X}_{\mathrm{O}}$ and h_m operates row-wise according to a given missing mask $\boldsymbol{M}$. Given an observational instance $\boldsymbol{X}_{\mathrm{O}}^j$, we construct a mechanism h_c to complete the data such that $h_c(\boldsymbol{X}_{\mathrm{O}}^j) = \widetilde{\boldsymbol{X}}^j \in \mathbb{R}^d$.

For h_m and h_c defined above, we can define the model distribution over $\boldsymbol{X}_{\mathrm{O}}$ via its reconstruction from the model, that is $\mu_\theta(\boldsymbol{X}_{\mathrm{O}}) := n^{-1}\sum_{j=1}^n \delta_{h_m\{f_\theta[h_c(\boldsymbol{X}_{\mathrm{O}}^j)]\}}$. Furthermore, let $\mu_\theta(\boldsymbol{X}) := n^{-1}\sum_{j=1}^n \delta_{f_\theta(\boldsymbol{X}^j)}$ denote the model distribution over the true data.

Definition 3.1. (Cost function for incomplete samples) The transport cost between a particle j from $\mu_{\mathcal{D}}(\boldsymbol{X}_{\mathrm{O}})$ and a particle k from $\mu_\theta(\boldsymbol{X}_{\mathrm{O}})$ is given by

$$c\left\{\boldsymbol{X}_{\mathrm{O}}^j, h_m\left[f_\theta\left(h_c(\boldsymbol{X}_{\mathrm{O}}^k)\right)\right]\right\} := d_c\left\{h_c\left(\boldsymbol{X}_{\mathrm{O}}^j\right), f_\theta\left[h_c(\boldsymbol{X}_{\mathrm{O}}^k)\right]\right\} \quad \forall j,k \in [n],$$

where d_c is a metric between two complete vectors in $\mathbb{R}^d$.

A.1. Proof for Lemma 3.2.

Lemma 3.2. For h_c, h_m defined as above, if h_c is optimal in the sense that h_c recovers the original data i.e., $h_c(\boldsymbol{X}_{\mathrm{O}}^j) = \boldsymbol{X}^j, \forall j \in [n]$, we have

$$W_c\left[\mu_{\mathcal{D}}(\boldsymbol{X}_{\mathrm{O}}), \mu_\theta(\boldsymbol{X}_{\mathrm{O}})\right] = W_{d_c}\left[h_c\#\mu_{\mathcal{D}}(\boldsymbol{X}_{\mathrm{O}}), \mu_\theta(\boldsymbol{X})\right].$$

where $h_c\#\mu_{\mathcal{D}}(\boldsymbol{X}_{\mathrm{O}}) = n^{-1}\sum_{j=1}^n \delta_{h_c(\boldsymbol{X}_{\mathrm{O}}^j)}$, which also represents empirical distribution over the true data.

Proof.

$$\begin{aligned}
&W_c\left[\mu_{\mathcal{D}}(\boldsymbol{X}_{\mathrm{O}}), \mu_\theta(\boldsymbol{X}_{\mathrm{O}})\right] \\
&= \min_{\mathbf{P}\in\mathbb{U}(\mathbf{1}_n,\mathbf{1}_n)} \sum_{j,k} c\left\{\boldsymbol{X}_{\mathrm{O}}^j, h_m\left[f_\theta\left(h_c(\boldsymbol{X}_{\mathrm{O}}^k)\right)\right]\right\}\mathbf{P}^{j,k} \\
&\overset{(1)}{=} \min_{\mathbf{P}\in\mathbb{U}(\mathbf{1}_n,\mathbf{1}_n)} \sum_{j,k} d_c\left\{h_c\left(\boldsymbol{X}_{\mathrm{O}}^j\right), f_\theta\left[h_c(\boldsymbol{X}_{\mathrm{O}}^k)\right]\right\}\mathbf{P}^{j,k} \\
&\overset{(2)}{=} \min_{\mathbf{P}\in\mathbb{U}(\mathbf{1}_n,\mathbf{1}_n)} \sum_{j,k} d_c\left\{h_c(\boldsymbol{X}_{\mathrm{O}}^j), f_\theta\left(\boldsymbol{X}^k\right)\right\}\mathbf{P}^{j,k} \\
&= W_{d_c}\left[h_c\#\mu_{\mathcal{D}}(\boldsymbol{X}_{\mathrm{O}}), \mu_\theta(\boldsymbol{X})\right],
\end{aligned}$$

where the minimum is taken over all possible couplings in the Birkhoff polytope $\mathbb{U}(\mathbf{1}_n/n, \mathbf{1}_n/n)$. Note that the equality $\overset{(1)}{=}$ follows from Definition 3.1. We further have $\overset{(2)}{=}$ since h_c is optimal. □

A.2. Proof for Theorem 3.3.

Given a set of observed samples $\boldsymbol{X}_{\mathrm{O}} = \{\boldsymbol{X}_{\mathrm{O}}^j\}_{j=1}^n$ and an imputer h_c, let $\mu(\widetilde{\boldsymbol{X}}) = n^{-1}\sum_{j=1}^n \delta_{\widetilde{\boldsymbol{X}}^j}$ define an empirical distribution over the set of samples $\left\{\widetilde{\boldsymbol{X}}^j = h_c(\boldsymbol{X}_{\mathrm{O}}^j)\right\}_{j=1}^n$.

Theorem 3.3. Let $\phi : \mathbb{R}^{n\times d} \mapsto \mathbb{R}^{n\times d}$ be a stochastic map such that $\phi\#\mu(\widetilde{\boldsymbol{X}}) = \mu_\theta(\boldsymbol{X})$. For a fixed value of θ,

$$W_{d_c}\left[\mu(\widetilde{\boldsymbol{X}}); \mu_\theta(\boldsymbol{X})\right] = \min_\phi \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}), \boldsymbol{Y}\sim\phi(\widetilde{\boldsymbol{X}})}\left[d_c\left(\widetilde{\boldsymbol{X}}, f_\theta(\boldsymbol{Y})\right)\right], \tag{8}$$

where d_c is a metric between two complete vectors in $\mathbb{R}^d$.

Proof. A sample $\boldsymbol{X}^j \sim \mu_\theta(\boldsymbol{X})$ is realized from an SCM by first sampling from the model the values for the root nodes and then generating the data for the remaining nodes via ancestral sampling. If a sample $\boldsymbol{X}^j$ is indeed generated by the model, we should be able to reconstruct $\boldsymbol{X}^j$ via the mechanisms f_θ, where every feature $\boldsymbol{X}_i^j$ is only determined by the features corresponding to the parent nodes of node i, while the effect of the non-parental features is zero-ed out by $\mathbf{W}(f)$. In the finite-sample setting, one thus can view $\boldsymbol{X}$ as its own parents and $\boldsymbol{X}$ reconstructs itself deterministically i.e., $\boldsymbol{X} = f_\theta(\boldsymbol{X})$.

We consider three distributions: $\mu(\widetilde{\boldsymbol{X}})$ over $A = \mathbb{R}^{n\times d}$, $\mu_\theta(\boldsymbol{X})$ over $B = \mathbb{R}^{n\times d}$ and $\mu_\theta(\boldsymbol{X})$ over $C = \mathbb{R}^{n\times d}$. Note that the establishment of B and C over the same distribution follows from the above mechanism where $\boldsymbol{X}$ reconstructs itself.

Let $\Gamma^* \in \mathcal{P}\left(\mu(\widetilde{\boldsymbol{X}}), \mu_\theta(\boldsymbol{X})\right)$ be the *optimal* joint distribution over the $\mu(\widetilde{\boldsymbol{X}})$ and $\mu_\theta(\boldsymbol{X})$ of the corresponding Wasserstein distance. Let $\alpha = (id, f_\theta)\#\mu_\theta(\boldsymbol{X})$ be a deterministic coupling or joint distribution over $\mu_\theta(\boldsymbol{X})$ and $\mu_\theta(\boldsymbol{X})$.

The Gluing lemma (see Lemma 5.5 in Santambrogio (2015) indicates the existence of a tensor coupling measure σ over $A \times B \times C$ such that $(\pi_{A,C})\#\sigma = \Gamma$ and $(\pi_{B,C})\#\sigma = \alpha$, where π are the projectors. Let $\beta = (\pi_{A,B})\#\sigma$ be a joint distribution over $\mu(\widetilde{\boldsymbol{X}})$ and $\mu_\theta(\boldsymbol{X})$.

Let $\phi(\widetilde{\boldsymbol{X}}) = \beta(\cdot|\widetilde{\boldsymbol{X}})$ further denote a stochastic map from A to B. Let $\sigma_{BC} = (\pi_{B,C})\#\sigma$. It follows that

$$
\begin{aligned}
W_c\left[\mu(\widetilde{\boldsymbol{X}}); \mu_\theta(\boldsymbol{X})\right] &= \mathbb{E}_{(\widetilde{\boldsymbol{X}},\boldsymbol{X})\sim\Gamma^*}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] = \mathbb{E}_{(\widetilde{\boldsymbol{X}},\boldsymbol{Y},\boldsymbol{X})\sim\sigma}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] \\
&= \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}\sim\beta(\cdot|\widetilde{\boldsymbol{X}}),\boldsymbol{X}\sim\sigma_{BC}(\cdot|\boldsymbol{Y})}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] \\
&\overset{(3)}{=} \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{X}\sim\phi(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}=f_\theta(\boldsymbol{X})}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] \\
&\overset{(4)}{=} \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}\sim\phi(\widetilde{\boldsymbol{X}})}\left[c\left(\widetilde{\boldsymbol{X}}, f_\theta(\boldsymbol{Y})\right)\right] \\
&\geq \min_{\phi} \quad \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}\sim\phi(\widetilde{\boldsymbol{X}})}\left[c\left(\widetilde{\boldsymbol{X}}, f_\theta(\boldsymbol{Y})\right)\right]. \qquad (9)
\end{aligned}
$$

Let ϕ^* be the *optimal* backward map satisfying $\phi\#\mu(\widetilde{\boldsymbol{X}}) = \mu_\theta(\boldsymbol{X})$. The joint distribution β over $\mu(\widetilde{\boldsymbol{X}})$ and $\mu_\theta(\boldsymbol{X})$ is now constructed by first sampling $\widetilde{\boldsymbol{X}}$ from $\mu(\widetilde{\boldsymbol{X}})$, then sampling $\boldsymbol{Y}$ from $\phi^*(\widetilde{\boldsymbol{X}})$ and finally collecting $(\widetilde{\boldsymbol{X}}, \boldsymbol{Y}) \sim \beta$.

Let us consider the joint distribution α over $\mu_\theta(\boldsymbol{X})$ and $\mu_\theta(\boldsymbol{X})$ as defined above. By the Gluing lemma, there exists a joint distribution σ over $A \times B \times C$ such that $(\pi_{A,B})\#\sigma = \beta$ and $(\pi_{B,C})\#\sigma = \alpha$. We denote $\Gamma = (\pi_{A,C})\#\sigma$ the induced joint distribution over $\mu(\widetilde{\boldsymbol{X}})$ and $\mu_\theta(\boldsymbol{X})$. Let $\sigma_{BC} = (\pi_{B,C})\#\sigma$.

$$
\begin{aligned}
&\min_{\phi} \quad \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}\sim\phi(\widetilde{\boldsymbol{X}})}\left[c\left(\widetilde{\boldsymbol{X}}, f_\theta(\boldsymbol{Y})\right)\right] \\
&= \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}\sim\phi^*(\widetilde{\boldsymbol{X}})}\left[c(\widetilde{\boldsymbol{X}}, f_\theta(\boldsymbol{Y}))\right] \\
&\overset{(5)}{=} \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}\sim\phi^*(\widetilde{\boldsymbol{X}}),\boldsymbol{X}=f_\theta(\boldsymbol{Y})}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] \\
&= \mathbb{E}_{\widetilde{\boldsymbol{X}}\sim\mu(\widetilde{\boldsymbol{X}}),\boldsymbol{Y}\sim\beta(.|\widetilde{\boldsymbol{X}}),\boldsymbol{X}\sim\sigma_{BC}(.|\boldsymbol{Y})}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] \\
&= \mathbb{E}_{(\widetilde{\boldsymbol{X}},\boldsymbol{Y},\boldsymbol{X})\sim\sigma}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] \\
&= \mathbb{E}_{(\widetilde{\boldsymbol{X}},\boldsymbol{X})\sim\Gamma}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] \\
&\geq \min_{\Gamma\sim\mathcal{P}\left(\mu(\widetilde{\boldsymbol{X}}),\mu_\theta(\boldsymbol{X})\right)} \mathbb{E}_{(\widetilde{\boldsymbol{X}},\boldsymbol{X})\sim\Gamma}\left[c(\widetilde{\boldsymbol{X}}, \boldsymbol{X})\right] = W_c\left[\mu(\widetilde{\boldsymbol{X}}); \mu_\theta(\boldsymbol{X})\right]. \qquad (10)
\end{aligned}
$$

□

Note that the equality's in $(3) - (5)$ are due to the fact that α is a deterministic coupling and the expectation is reserved through a deterministic push-forward map. From (9) and (10), we reach the desired equality.

B. Related Work

Missing data imputation. Besides basic imputation with mean/median/mode values, there are a plethora of advanced methods for filling in the missing values. In one class of methods, data features are handled one at a time. The key technique is to iteratively estimate the conditional distribution of the feature given the other features (Van Buuren et al., 2006; Van Buuren & Groothuis-Oudshoorn, 2011; Liu et al., 2014; Zhu & Raghunathan, 2015). Methods in the other class consider features altogether by learning their joint distribution explicitly or implicitly (Mattei & Frellsen, 2019; Yoon & Sull, 2020; Nazabal et al., 2020; Richardson et al., 2020; You et al., 2020; Dai et al., 2021; Peis et al., 2022; Fang & Bao, 2023; Gao et al., 2023). A separate line of works focuses on refining existing imputation methods, which show improvements over stand-alone methods (Yoon et al., 2018; Mohan & Pearl, 2021; Jarrett et al., 2022).

Causal discovery with complete data. Causal discovery algorithms primarily fall into two categories: constraint-based and score-based approaches. Constraint-based methods such as PC (Spirtes & Glymour, 1991) and FCI (Spirtes et al., 2000) extract conditional independencies from the data distribution to detect edge existence and direction. These approaches have been adapted to address the issue of missing data through test-wise deletion and adjustments (Strobl et al., 2018; Gain & Shpitser, 2018; Tu et al., 2019). On the other hand, score-based methods search for model parameters in the DAG space by optimizing a scoring function (Ott & Miyano, 2003; Chickering, 2002; Teyssier & Koller, 2012; Cussens et al., 2017). Historically, such methods come with significant computational burden due to combinatorial optimization complexities. Continuous optimization of structures, pioneered by NOTEARS (Zheng et al., 2018), later lays the foundation for the development of scalable causal discovery methods. NOTEARS introduces an algebraic characterization of DAGs via trace exponential, extending its applicability to nonlinear scenarios (Lachapelle et al., 2019; Zhu et al., 2019; Wang et al., 2020; 2021; Ng et al., 2022), time-series data (Pamfil et al., 2020), unmeasured confounders (Yuan et al., 2011), and topological ordering (Deng et al., 2023). Alternative DAG characterizations also exist such as based on the polynomial formulation (Yu et al., 2019) and log determinant (Bello et al., 2022). While early methods focus on the point estimation of graphs, modern works adopt Bayesian approach learning distributions over graphs. These techniques can incorporate DAG formulations seamlessly within their frameworks, capitalizing on the differentiability of DAG sampling and leveraging an amortized inference engine for enhanced efficiency (Lorch et al., 2021; Ashman et al., 2022; Geffner et al., 2022; Lorch et al., 2022; Charpentier et al., 2022).

Causal discovery with missing data. Extensions of the PC algorithm exist for learning causal graphs under missing data (Tu et al., 2019; Gain & Shpitser, 2018), which utilizes all the samples without missingness while eliminating the biases involved in the conditional independence test. A dominant family of methods is based on Expectation-Maximization (EM, Dempster et al., 1977) that iteratively infers missing values and performs structural learning. Adel & de Campos (2017) introduce a hill-climbing approximate algorithm for the completions of the missing values, which is followed by structure optimization step by any off-the-shelf algorithm for structure learning. Friedman et al. (1997) and Singh (1997) iteratively refine conditional distributions from which they sample missing values. These methods requires the posterior exists in closed form and can only discover the graph up to the Markov equivalence class. Leveraging continuous optimization from NOTEARS (Zheng et al., 2018), MissDAG (Gao et al., 2022) focuses on continuous identifiable ANMs and develop a method based on approximate posterior using Monte Carlo and rejection sampling where exact posterior is not available. VISL (Morales-Alvarez et al., 2022) is a divergent line of approach based on amortized variational inference. Different from OTM and MissDAG, VISL adopts Bayesian learning and assumes the existence of a latent, low-dimensional factor that effectively summarizes the data based solely on the observed part. The model learns the latent factors of the data, using which to discover the graph and reconstruct the full data.

Optimal transport. Optimal transport (OT) studies the optimal transportation of mass from one distribution to another (Villani et al., 2009). Through the notion of Wasserstein distance, OT offers a geometrically meaningful distance between probability distributions, proving effectiveness in various machine learning domains (Zhao et al., 2020; Zhang et al., 2022; Bui et al., 2021; Nguyen et al., 2022; Vuong et al., 2023; Vo et al., 2023). From the view of distribution matching, OT-based missing data imputation has been proposed (Muzellec et al., 2020; Zhao et al., 2023). The key idea is to learn the missing values by minimizing the distribution between two randomly sampled batches in the data. An emerging line of research is OT for causal discovery, initiated by (Tu et al., 2021) providing new identifiability result based on dynamic formulation of OT. This work considers a simple two-node setting solely for additive noise models. A modern extension is proposed in (Akbari & Ganassali, 2023), yet only applicable to complete data.

C. Experiment Details

This section provides the implementation details for OTM and the baseline methods, as well as the mechanism used to produce the missing data.

C.1. Training configuration

We apply the default structure learning setting of DAGMA to OTM and the imputation baselines: log MSE for the non-linear score function, number of iterations $T = 4$, initial central path coefficient $\mu = 1$, decay factor $\alpha = 0.1$, log-det parameter $s = \{1, 0.9, 0.8, 0.7\}$. DAGMA implements an adaptive gradient method using the ADAM optimizer with learning rate of 2×10^{-4} and $(\gamma_1, \gamma_2) = (0.99, 0.999)$, where γ_2 is the coefficient for l_1 regularizer included to promote sparsity.

OTM. As for OTM implementation, we note that OTM can be applied to various data types. As our experiments mostly deal with continuous variables, we model the imputation network $\mathsf{NN}(\cdot)$ with a local Gaussian distribution and design two multi-layer perceptrons (with ReLU activation) that output the means and diagonal covariance matrices given the missing data, where the missing entries are pre-imputed with zeros. The hidden units are equal to the number of nodes in the graph. For real-world datasets only, we add an extra Tanh+ReLU activation as the final layer since the input values are between 0 and 1. The imputation network is integrated into DAGMA, and all parameters are optimized for 8×10^3 iterations. The optimal values of λ are found to be 0.01 for simulations and 1.5 for real-world datasets. For the divergence measure D, we use MMD in our experiments.

The squared MMD between two set of samples $\boldsymbol{U} = \{\boldsymbol{U}^j\}_{j=1}^n$ and $\boldsymbol{V} = \{\boldsymbol{V}^j\}_{j=1}^n$, where $\boldsymbol{U}^j, \boldsymbol{V}^j \in \mathbb{R}^d$ is computed as

$$\mathsf{MMD}^2(\boldsymbol{U}, \boldsymbol{V}, \kappa) = \frac{1}{n(n-1)} \sum_{j \neq k} \kappa(\boldsymbol{U}^j, \boldsymbol{U}^k) + \frac{1}{n(n-1)} \sum_{j \neq k} \kappa(\boldsymbol{V}^j, \boldsymbol{V}^k) - \frac{2}{n^2} \sum_{j,k} \kappa(\boldsymbol{U}^j, \boldsymbol{V}^k), \tag{11}$$

where κ is the characteristic positive-definite kernel.

Baselines. For Mean imputation and Iterative imputation (the number of iterations is 50), we use `Scikit-learn` implementation. For OT imputer (Muzellec et al., 2020), we set the learning rate to 0.01 and the number of iterations to $10,000$. Following their code, we run DAGMA on the imputed data for 8×10^4 iterations. MissDAG is currently built on NOTEARS (Zheng et al., 2018). For MissDAG, we set the number of iterations for the EM procedure is 10 on synthetic datasets and 100 for real datasets, while leaving the other hyper-parameters of MissDAG same as reported in the paper.

C.2. Missingness

We follow the setup of MissDAG (Gao et al., 2022) to generate the missing indicator matrix $\boldsymbol{M}$ that is used to mask the original data. For completeness, we repeat the procedure here. Across the experiments, we set 30% of the variables to be fully-observed. Let r_m denote the missing rate.

- **MCAR:** This missing mechanism is independent from the variables. We sample a matrix $\boldsymbol{M}'$ from a Uniform$([0, 1])$ and set $\boldsymbol{M}[t, i] = 0$ if $\boldsymbol{M}'[t, i] \leq r_m$.
- **MAR:** The missing mechanism is systematically related to the observational variables but independent from the missing variables. The missingness of the remaining 70% variables are generated according to a logistic model with random weights that are related to the fully observed variables.
- **MNAR:** The missing mechanism is related to the missing variables. The self-masked missingness is taken as this mechanism, where the remaining 70% variables are masked according to a logistic model with random weights that are related to the corresponding variables.

D. Additional Experiments

In this section, we present additional results following the same setup described in Appendix C. Figures 6 to 9 illustrates the performance of methods on MAR and MNAR cases. Figure 10 further investigates the methods on different configurations of noise types and graph degrees in MCAR cases at 10% missing rate. The effectiveness of OTM remains stable across settings.

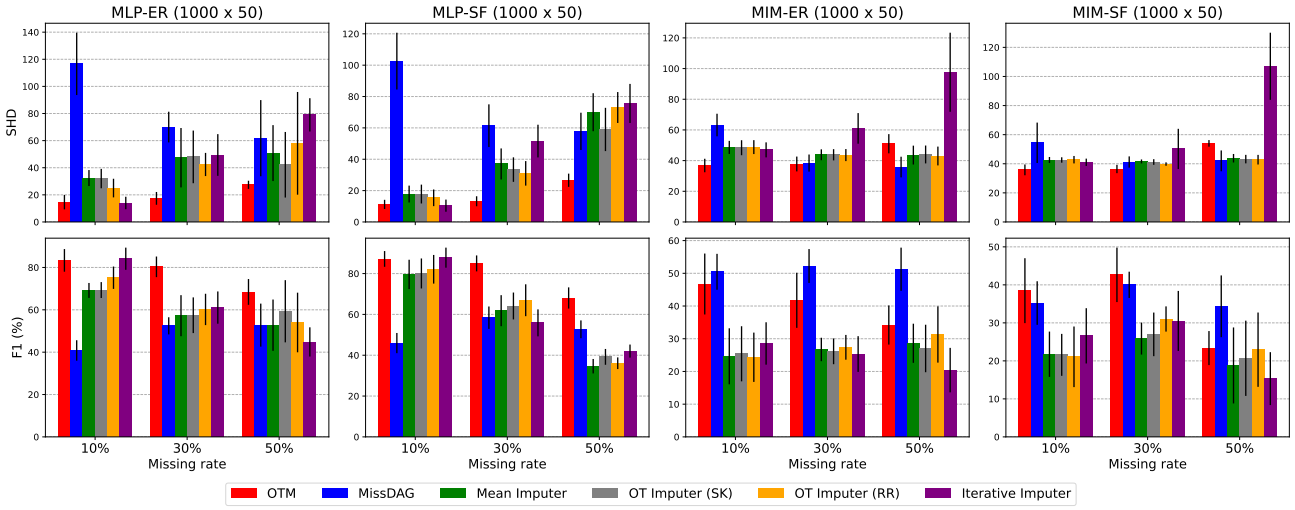

Figure 6. Nonlinear ANMs (MAR). SHD ↓ and F1 ↑.

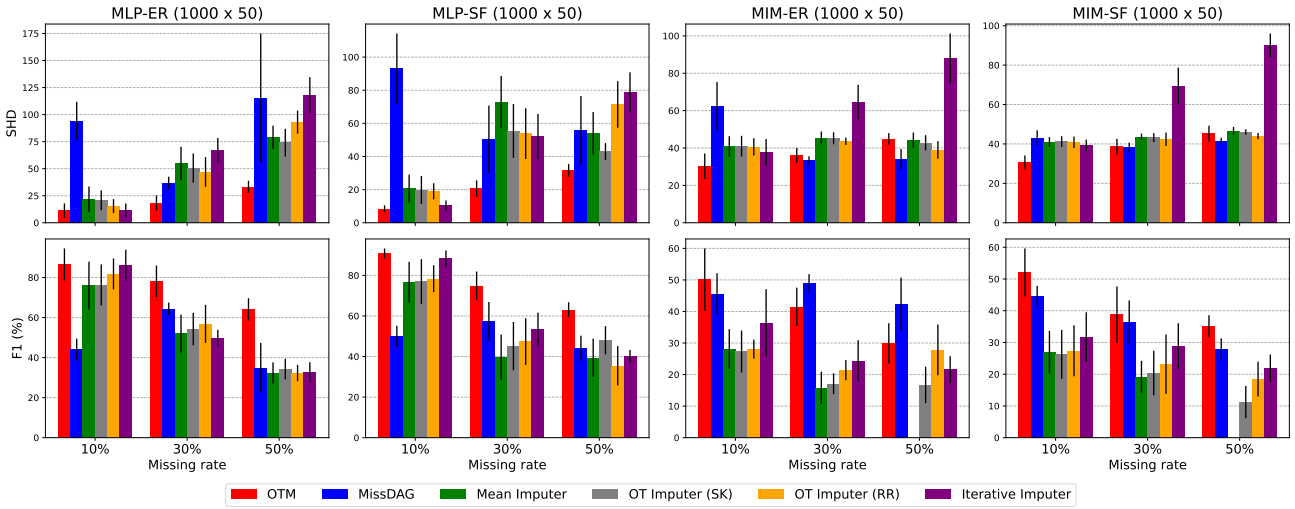

Figure 7. Nonlinear ANMs (MNAR). SHD ↓ and F1 ↑.

Linear ANMs This section further reports the performance of methods in linear cases. We specifically study linear Gaussian models with non-equal variances (LGM-NV). For linear models, we run OTM using NOTEARS (Zheng et al., 2018) as the underlying causal discovery method to show that OTM can be flexibly integrated into any existing DAG learning methods. We use the default setting of NOTEARS and model the imputation network with a simple deterministic linear layer. Figures 11 demonstrates the competitive performance of OTM against MissDAG. This is achieved without explicitly doing posterior inference.

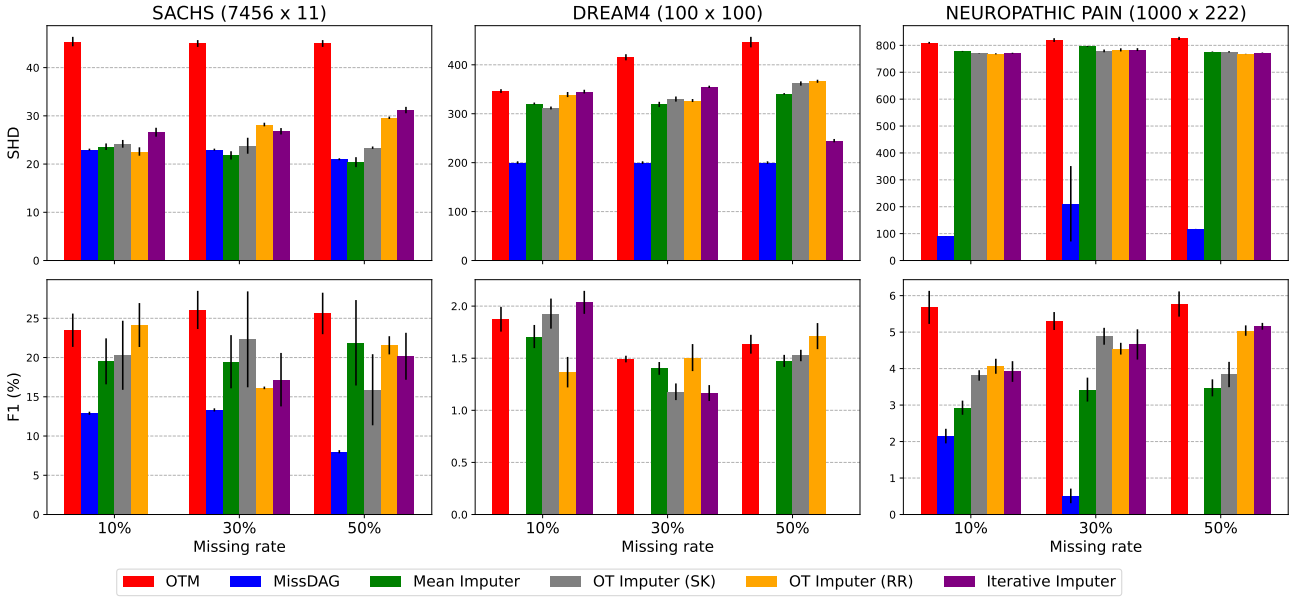

Figure 8. Real-world datasets (MAR). SHD ↓ and F1 ↑.

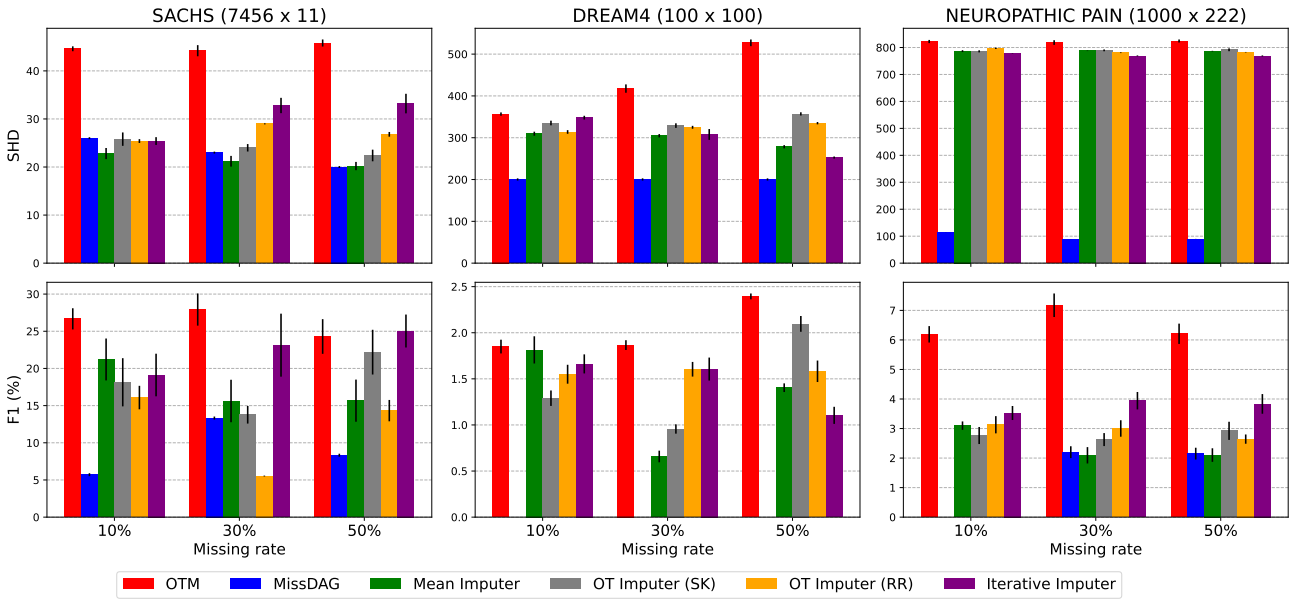

Figure 9. Real-world datasets (MNAR). SHD ↓ and F1 ↑.

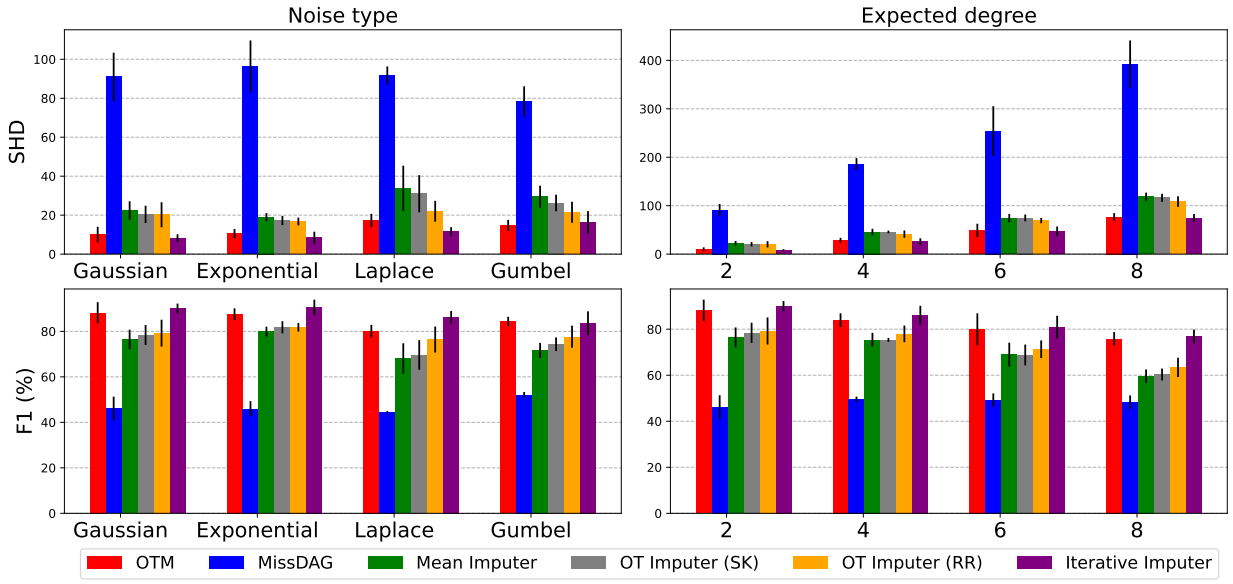

Figure 10. Results on different noise types and graph degrees in nonlinear ANMs (MCAR) at 10% missing rate. SHD ↓ and F1 ↑.

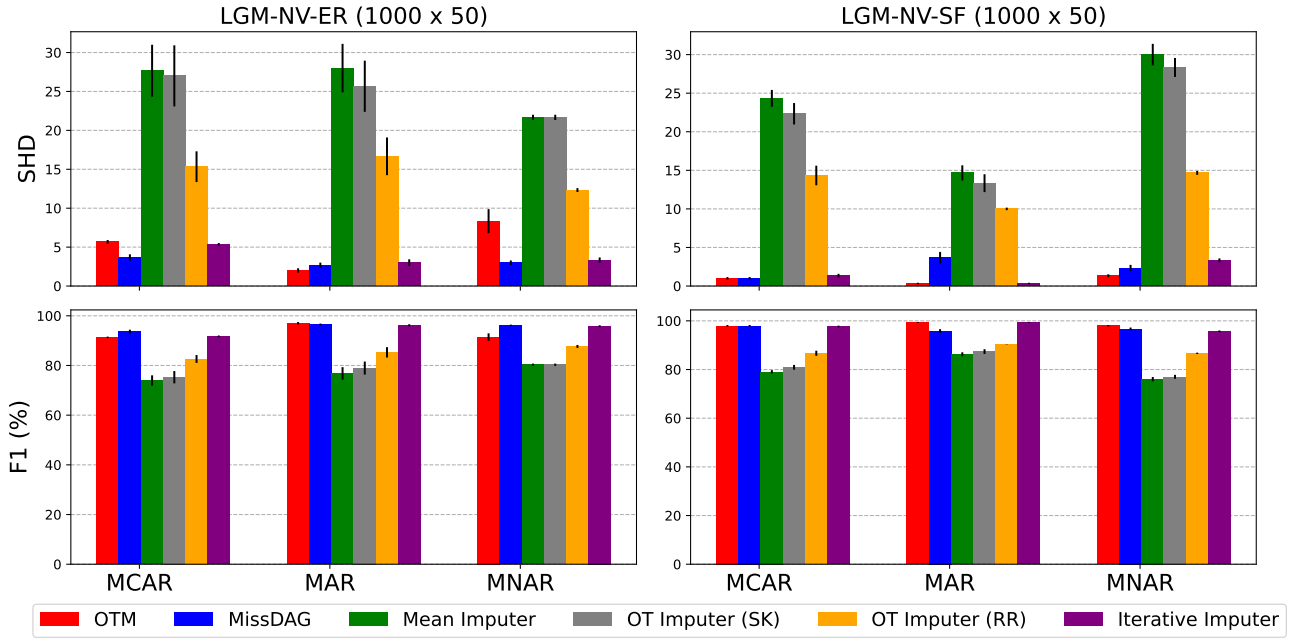

Figure 11. Linear ANMs at 10% missing rate. SHD ↓ and F1 ↑.