Proceedings of Machine Learning Research vol 140:1–22, 2022 1st Conference on Causal Learning and Reasoning

Causal Imputation via Synthetic Interventions

Author names withheld

Editor: Under Review for CLeaR 2022

Abstract

Consider the problem of determining the effect of a compound on a specific cell type. To answer this question, researchers traditionally need to run an experiment applying the drug of interest to that cell type. This approach is not scalable: given a large number of different actions (compounds) and a large number of different contexts (cell types), it is infeasible to run an experiment for every action-context pair. In such cases, one would ideally like to predict the outcome for every pair while only needing outcome data for a small *subset* of pairs. This task, which we label *causal imputation*, is a generalization of the causal transportability problem. To address this challenge, we extend the recently introduced *synthetic interventions* (SI) estimator to handle more general data sparsity patterns. We prove that, under a latent factor model, our estimator provides valid estimates for the causal imputation task. We motivate this model by establishing a connection to the linear structural causal model literature. Finally, we consider the prominent CMAP dataset in predicting the effects of compounds on gene expression across cell types. We find that our estimator outperforms standard baselines, thus confirming its utility in biological applications.

Keywords: Latent factor model, imputation, causal inference

1. Introduction

A central goal in science is to determine the outcome of an action (i.e., intervention). Traditionally, researchers rely on experimentation to answer such questions. However, this approach is rarely scalable and/or can be unethical in many critical applications, e.g., clinical trials and policy evaluation. As a result, there is a growing interest in utilizing already-existing data to help predict the effect of every candidate action, prior to deciding which action to execute. The key challenge underpinning this approach is that the available data regarding the effect of an action almost always comes from a *different context* than the one in which we are interested in making a prediction. To further motivate our problem setting, we provide an example from healthcare below.

A Motivating Example. Consider a prevalent medical scenario in which physicians aim to repurpose existing drugs to treat novel diseases such as COVID-19. Although there are over 20,000 small molecule compounds (actions) with known therapeutic capabilities, exposing COVID-19 infected cells (new contexts) to each compound would be far too time-consuming and costly. Fortunately, the publicly available CMAP dataset (Subramanian et al., 2017) has already catalogued the effects of these compounds on a variety of other cell types (different contexts). In such a setting, the primary intellectual challenge is to develop a methodology which uses the CMAP dataset to predict the outcome of applying each compound on COVID-19 infected cells. Such a method leads to efficient *hypothesis generation* for finding compounds with therapeutic potential for this disease.

Key Question. The focus of this work is to tackle problems such as the one described above. More generally, we aim to answer the following question:

"Given a dataset of outcomes from various action-context pairs, can we predict the outcome of a given action on a context in which it has not been applied?"

© 2022 .

Our Contributions. The main contributions of this work are three-fold: algorithmic, theoretical, and empirical. We summarize them below.

(i) Algorithmic. We extend the recently proposed synthetic interventions (SI) estimator (Agarwal et al., 2020a) to handle more general sparsity patterns. At its core, the SI estimator produces a counterfactual prediction for any "target" (novel) context-action pair by first constructing a "synthetic" version of the target context as a weighted combination of "donor" (different) contexts via (regularized) regression. Then, it uses the learnt model to re-scale the outcomes of those donor contexts under the target action to estimate the outcomes of the target context-action pair. Taking inspiration from biological applications, we adapt the SI estimator to instead construct a synthetic version of the target *action* as a weighted combination of donor actions. Given the more general observation patterns, we also equip the SI estimator to use *all* available data when learning the regression model. As a result, the SI estimator of Agarwal et al. (2020a) becomes a special case of our formulation. Moving forward, we refer to our estimator as `SI-A`. For details, please refer to Section 4.

(ii) Theoretical. We justify our algorithmic approach by presenting an "identification" result in Theorem 1. More formally, we establish that our estimator yields the correct causal estimates for the causal imputation task, under a factor model across contexts and actions. Further, we re-interpret the SI causal framework by connecting it to linear structural causal models (see Proposition 1).

(iii) Empirical. Using the prominent CMAP dataset, we extensively benchmark our approach against well-established baselines. The CMAP dataset contains gene expression signatures for over 20,000 different small molecule compounds (actions) across 70 different cell types (contexts). Using a small fraction of the observed signatures as a training set, we consider the task of imputing the signatures associated with a held-out test set that emulates an un-experimented collection of novel cell type-compound pairs. Empirically, we find that our estimator outperforms the baselines. In particular, the median normalized root mean-square error (NRMSE) of `SI-A` is 0.34, compared to 0.41 for the closest baseline. Similarly, `SI-A` improves the *alignment* of the predicted effect of the action with the true effect of the action, measured in terms of cosine similarity, improving from 0.44 in the best baseline to 0.68. Moreover, our experiments show that our variant of the SI estimator (which regresses along compounds) significantly outperforms the original SI estimator (which regresses along cell types), a result that may be of independent biological interest.

Organization of the paper. In Section 2, we review related work on causal imputation. Then, in Section 3, we formally state our problem setting. In Section 4, we describe our estimation strategy, which extends the original SI estimator. We state our formal results in Section 5, establishing an identification result under a factor model assumption, as well as providing an interpretation of this assumption in terms of a linear structural causal model. Finally, Section 6 showcases our empirical findings on the CMAP dataset.

2. Related Work

Predicting the (vector-valued) outcome of an action in a novel context, a task we call *causal imputation*, is a ubiquitous problem, and there are numerous lines of study tackling this challenge. We highlight some of the most relevant below. In practice, we require a method that (i) is *fast*, (ii) can model *heterogenous* effects across different cell types, (iii) comes with theoretical *guarantees*, and (iv) does *not* need a detailed model of the causal system as input. Existing methods, which we describe below, all lack one of these 4 desiderata, as summarized in Table 1.

	Fast	Heterogeneous Effects	Guarantees	Needs selection diagram
Fixed Effects (FE)	✓	✗	✓	✗
Autoencoding + FE	✓	✓	✗	✗
Causal Transportability	✓	✓	✓	✓
MICE/MissForest	✗	✓	✗	✗
Synthetic Interventions	✓	✓	✓	✗

Table 1: Summary of Causal Imputation Methods

Fixed effects. The simplest model for the effect of actions across different contexts is the *fixed effects* model, which associates each action with a vector representing the additive effect of that action, assumed to be *invariant* across contexts. We note that the fixed effects estimator is a special case of the regularized linear model proposed in Dixit et al. (2016). Their method makes predictions of the form $\widehat{\mathbf{x}}^{ca} = \beta^a + \mathbf{w}^c \beta^c$, where $\widehat{\mathbf{x}}^{ca}$ denotes the *outcome* vector associated with cell type c under action a and $\mathbf{w}^c$ is a set of covariates for the cell type c. When no covariates are available except a one-hot encoding of the cell type, their estimator reduces to the fixed-effects model $\widehat{\mathbf{x}}^{ca} = \beta^a + \beta^c$. Under a fixed effects model, the effect of an action in any new context can be predicted based solely on the effect of that action in a *single* other context, greatly reducing the number of required experiments. However, the fixed effects model is not realistic in our application of interest, as it does *not* allow for heterogeneity, even though the effect of a drug on the expression of a given gene can vary drastically between cell types (Kidd et al., 2016), e.g., increasing the expression of a gene in one cell type while decreasing it in a different cell type.

To overcome this limitation of the fixed effects model, Lotfollahi et al. (2019) introduced the *scGen* method. This method first trains a variational autoencoder (VAE) in order to find a low-dimensional representation (latent embedding) of gene expression data, then uses a fixed effects model in the *latent* space of the VAE. Similarly, one could instead apply our method, `SI-A`, in the *latent* space. The theoretical study of combining autoencoders, or other nonlinear embeddings, with "direct" causal imputation methods such as the one considered here, is an interesting direction for future work. To the best of our knowledge, no theoretical guarantees yet exist for such combinations, but promisingly, recent work using autoencoders to predict perturbation effects on SARS-CoV2 infected cells empirically demonstrated that autoencoders *align* representations in a way that induces linearity amongst the perturbations and cell types in the learned space (Belyaeva et al., 2020).

Causal transport. The goal in causal transport is to find a *transport formula* (Bareinboim and Pearl, 2014, 2016), i.e., a map from the outcomes of actions in a set of "source" contexts to the outcome in a "target" context under some action. Whereas causal transport focuses on prediction from data generated in *different* contexts than the target context, causal imputation generalizes this problem to *also* use data about *different* actions, including from the target context itself. Previous methods algorithmically derive transport formulas from a *selection diagram*, i.e., a causal model of the system that also specifies invariants between the source and target contexts. A major contrast between the present work on causal imputation and prior work on causal transport is that `SI-A` does *not* rely on specifying (or learning) a selection diagram. Indeed, as we show in Proposition 1 and Theorem 1, `SI-A` is consistent for *any* linear causal structural model within a large class. A further advantage of the factor model framework considered in this work is that the key assumption (As-

sumption 2) that enables SI-A to generalize from a small collection of experiments to predicting on untested context-action pairs can be tested via a simple data-driven hypothesis test (see Section 4).

Traditional imputation methods. As missing values are a common issue in many applications (Bell and Koren, 2007; Troyanskaya et al., 2001), a number of imputation methods have been developed to fill in missing entries, see e.g. Bertsimas et al. (2017) for a recent review. Two prominent methods which are widely used in imputing genomic and other biological data (Tan et al., 2017; Waljee et al., 2013) are MICE (Van Buuren and Oudshoorn, 1999) and MissForest (Stekhoven and Bühlmann, 2012). To the best of our knowledge, these methods have no formal identification guarantees. Moreover, they are several orders of magnitude slower than our SI-A estimator - see Appendix F.1 for a comparison to MICE - making them impractical for the problem considered here.

The linear factor model assumption (Definition 1) used by our model is similar to factor model assumptions made by methods for low-rank tensor decomposition, such as PARAFAC Harshman et al. (1970); Bro (1997). As in the matrix case (Chatterjee, 2015a; Agarwal et al., 2018; Sportisse et al., 2020), such decompositions can be useful tools for tensor estimation (Jain and Oh, 2014), especially in the missing-at-random (MAR) or missing-completely-at-random (MCAR) settings (Rubin, 1976). However, PARAFAC assumes a low-rank *CP decomposition*, which is more restrictive than the low-rank *mode-2 decomposition* in Definition 1. Furthermore, PARAFAC requires a costly alternating minimization scheme to optimize, so that it is again several orders of magnitude slower than our SI-A estimator.

Methods using other data modalities. The imputation methods we have thus far discussed all begin with genomic data from various compounds and cell types, and impute genomic data for *novel* pairs of compounds and cell types. Orthogonally, there is a vast literature on predicting perturbation response using *different* data modalities, e.g. images (Hofmarcher et al., 2019; Yang et al., 2018) or molecular structure (Stokes et al., 2020). There are many possible avenues for combining these approaches with our method, e.g., using coupled autoencoders (Yang et al., 2019) to represent each compound in terms of a combination of other compounds *and* an encoding of its molecular structure.

3. Problem Statement

We consider a collection of contexts (e.g., cell types), denoted by $\mathcal{C}$, and actions (e.g., compounds), denoted by $\mathcal{A}$. The outcomes of interest (e.g., gene expression signatures) associated with a given context $c \in \mathcal{C}$ under action $a \in \mathcal{A}$ are denoted by $\mathbf{x}^{ca} \in \mathbb{R}^p$. Collectively, this forms an order-three tensor $\mathcal{X} = [\mathbf{x}^{ca} : c \in \mathcal{C}, a \in \mathcal{A}] \in \mathbb{R}^{|\mathcal{C}| \times |\mathcal{A}| \times p}$.

Observations. To model real-world scenarios, we consider the setup where we not only have access to a sparse subset of the entries in $\mathcal{X}$ (e.g., corresponding to a limited number of historical experiments), but also noisy instantiations of those observations. More formally, let $\tilde{\mathbf{x}}^{ca} \in \mathbb{R}^p$ denote a corrupted version of $\mathbf{x}^{ca}$, e.g., $\tilde{\mathbf{x}}^{ca} = \mathbf{x}^{ca} + \mathbf{e}^{ca}$ or $\tilde{\mathbf{x}}^{ca} = \mathbf{x}^{ca} \circ \mathbf{e}^{ca}$, where $\mathbf{e}^{ca}$ denotes measurement noise and $\circ$ is the entry-wise product. Further, let $\Omega \subset \mathcal{C} \times \mathcal{A}$ denote the context-action pairs for which we observe their associated outcomes, i.e., for each $(c, a) \in \Omega$, we observe $\tilde{\mathbf{x}}^{ca}$. We denote our observation set as $\mathcal{O} = \{\tilde{\mathbf{x}}^{ca} : (c, a) \in \Omega\}$.

Goal. Our primary interest is in recovering $\mathcal{X}$ from $\mathcal{O}$.

Notations. For any $a \in \mathcal{A}$ and $c \in \mathcal{C}$, let $\mathcal{A}(c) = \{a \in \mathcal{A} : (c, a) \in \Omega\}$ and $\mathcal{C}(a) = \{c \in \mathcal{C} : (c, a) \in \Omega\}$ denote the set of actions which are measured for context c, and the set of contexts which are measured for action a, respectively. These notations are extended to sets of contexts and actions, i.e., $\mathcal{A}(\mathcal{C}) = \cap_{c \in \mathcal{C}} \mathcal{A}(c)$ and $\mathcal{C}(\mathcal{A}) = \cap_{a \in \mathcal{A}} \mathcal{C}(a)$.

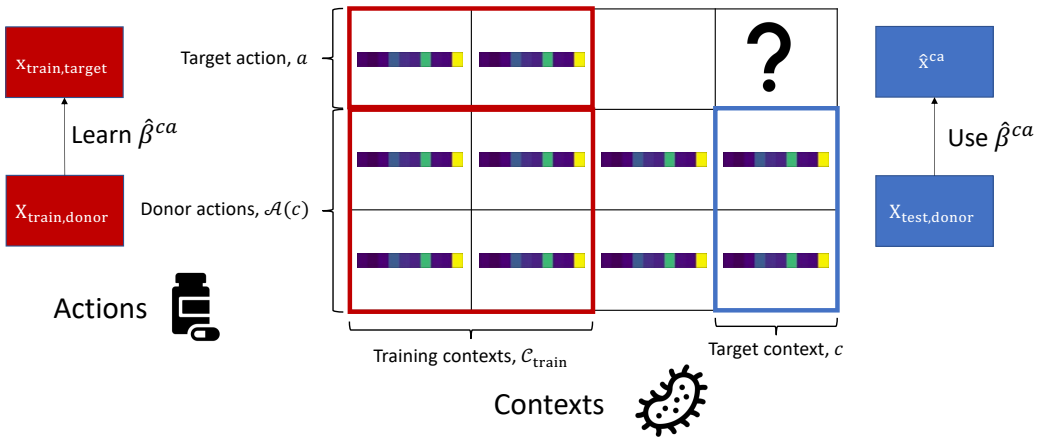


Figure 1: **The SI-Action Method.** To form an estimate $\hat{x}^{ca}$ of the outcome of target action a in target context c, we find a set of *donor actions* $\mathcal{A}(c)$ which are available in context c. Then, we find *training contexts* $\mathcal{C}_{\text{train}}$ for which both the donor actions and the target action are available, and learn weights $\hat{\beta}^{ca}$ via linear regression. Finally, we use these weights to form $\hat{x}^{ca}$.

4. Algorithm

We propose `SI-A`, which is an extension of the SI estimator introduced in Agarwal et al. (2020a), to handle more general sparsity patterns. Though our estimation strategy is applicable for any context-action pair of interest, we consider (without loss of generality) some $(c, a) \notin \Omega$. Below, we introduce additional notations to describe our estimator.

Additional Notations. We call $\mathcal{A}(c)$ the "donor actions" for context c. Let $\mathcal{C}_{\text{train}} = \mathcal{C}(\mathcal{A}(c) \cup \{a\})$ denote the set of contexts for which all donor actions and the target action are observed, we call these the "training contexts". Let $C = |\mathcal{C}_{\text{train}}|$ and $A = \mathcal{A}(c)$, and define $\tilde{\mathbf{X}}_{\text{train,donor}} \in \mathbb{R}^{p \cdot C}$, $\tilde{\mathbf{X}}_{\text{train,donor}} \in \mathbb{R}^{pC \times A}$ and $\tilde{\mathbf{X}}_{\text{test,donor}} \in \mathbb{R}^{p \times A}$ as

$$\tilde{\mathbf{x}}_{\text{train,target}} = [\tilde{\mathbf{x}}^{ia}]_{i \in \mathcal{C}_{\text{train}}} \qquad \tilde{\mathbf{X}}_{\text{train,donor}} = [\tilde{\mathbf{x}}^{ij}]_{i \in \mathcal{C}_{\text{train}}, j \in \mathcal{A}(c)} \qquad \tilde{\mathbf{X}}_{\text{test,donor}} = [\tilde{\mathbf{x}}^{cj}]_{j \in \mathcal{A}(c)}$$

Finally, let $\texttt{ME} : \mathbb{R}^{n \times p} \to \mathbb{R}^{n \times p}$ denote any matrix estimation method, used to recover a matrix from noisy observations, such as nuclear norm regularization or singular value thresholding.

Causal imputation for a novel context-action pair. We define the our estimation method as follows, where $\dagger$ denotes the pseudoinverse. The method is also presented visually in Fig. 1.

1. Model learning via regression
   (a) Define $\widehat{\mathbf{X}}_{\text{train,donor}} = \texttt{ME}(\tilde{\mathbf{X}}_{\text{train,donor}})$.
   (b) Define $\widehat{\beta}^{ca} = \widehat{\mathbf{X}}^{\dagger}_{\text{train,donor}} \tilde{\mathbf{x}}_{\text{train,target}} \in \mathbb{R}^{|\mathcal{A}(c)|}$
2. Causal imputation
   (a) Define $\widehat{\mathbf{X}}_{\text{test,donor}} = \texttt{ME}(\tilde{\mathbf{X}}_{\text{test,donor}})$.
   (b) Define $\widehat{\mathbf{x}}^{ca} = \widehat{\mathbf{X}}_{\text{test,donor}} \widehat{\beta}^{ca}$.

In Appendix A, we discuss how our estimator relates to the original SI estimator.

Extensions. We now discuss two natural modifications to the proposed method.

Reversing the axis of regression. The method we described learns weights $\widehat{\beta}^{ca}$ which express the action a as a linear combination of other actions $\mathcal{A}(c)$. The same method can be applied to instead express the context c as a linear combination of other contexts, by making the obvious replacements.

Using both estimators in tandem. The two estimation strategies described above—weighting actions and weighting contexts—can be used in a complementary fashion. For instance, say we apply the weighting actions estimator first. The imputed values increase the set of donor actions, which should lead to better outcomes when applying the weighting contexts estimator. An analogous argument can be made when the weighting contexts estimator is applied first. This suggests an algorithm that iteratively applies each approach until the imputed tensor converges to a desired threshold. However, one drawback of this approach is that it is computationally more demanding. Formally motivating and analyzing such an algorithm remains to be interesting future work.

Arbitrary sets of donor actions. For simplicity, we have written our estimator using *all* actions $\mathcal{A}(c)$ that are available for context c as "donor" actions. The corresponding set of training contexts are those for which each of these actions, along with the target action, is measured, i.e., $\mathcal{C}_{\text{train}} = \mathcal{C}(\mathcal{A}(c) \cup \{a\})$. Expressing $\widehat{\beta}$ as a linear combination of all actions $\mathcal{A}(c)$ may be suboptimal if it leads to a significantly smaller set $\mathcal{C}_{\text{train}}$. For example, say $|\mathcal{A}(c)| = 11$, and there is only one training context containing all of these actions, i.e., $|\mathcal{C}(\mathcal{A}(c) \cup \{a\})| = 1$. Suppose there is a subset $\mathcal{A}' \subset \mathcal{A}(c)$, with $|\mathcal{A}'| = 10$ with $|\mathcal{C}(\mathcal{A}' \cup \{a\})| = 100$, i.e., by excluding a single action from the linear combination, we can train on 100 times more contexts. Thus, we may consider the SI estimator induced by a specific *donor set* $\mathcal{A}_{\text{donor}} \subseteq \mathcal{A}(c)$, with the corresponding set of training contexts being $\mathcal{C}_{\text{train}} = \mathcal{C}(\mathcal{A}_{\text{donor}} \cup \{a\})$. The choice of donor set introduces a tradeoff: as the number of donor actions increases, the number of training contexts might decrease. This raises the question of how to pick an optimal set of donor actions for a given prediction. We provide a principled approach to this problem in Section 5.

5. Theoretical Results

In this section, we justify our algorithmic approach under a factor model and provide a justification for the factor model through the lens of causal structural models. We focus on the noiseless setting, i.e., for all $(c, a) \in \Omega$, we observe $\mathbf{y}^{ca} = \mathbf{x}^{ca}$. Hence, our results should be viewed as one of "identification" as we prove that $\mathcal{X}$ can be identified from noiseless data via our proposed algorithm. Indeed, identification arguments are considered a critical first step in causal inference (Shpitser and Pearl, 2008; Lee et al., 2020), as they are necessary for any analysis of consistency in the noisy setting. We also note that there is a rich body of literature that establishes formal guarantees for a suite of matrix estimation methods (see Chatterjee (2015b) and references therein). Formally building upon these results to provide rigorous sensitivity analyses remains future work.

5.1. Identification via `SI-A`

In what follows, we state our key modeling assumptions that will enable our estimator to impute $\mathcal{X}$. To reduce redundancy, the following discussion will be restricted to `SI-A`, i.e., where we learn a regression model across actions. Analogous statements apply to the case of learning across contexts.

First, we recall the definition of a linear factor model, also known as the interactive fixed effects model, which is prevalent within the causal inference literature (Agarwal et al., 2020a; Abadie et al., 2010; Bai, 2009; Athey et al., 2017) and critical to the SI framework of Agarwal et al. (2020a).

Definition 1 *We say that $\mathcal{X}$ satisfies a* linear factor model *if, for any $a \in \mathcal{A}$ and $c \in \mathcal{C}$, $\mathbf{x}^{ca} = U^c \mathbf{v}^a$, where $U^c \in \mathbb{R}^{p \times r}$ and $\mathbf{v}^a$ are latent factors associated with the context c and action a, respectively.*

Interpretation. Definition 1 posits that $\mathcal{X}$ satisfies a low-rank assumption (Kolda and Bader, 2009)[1]. While *exact* adherence to a linear factor model can be a strong assumption, it has been widely observed in practice that many big-data matrices are approximately low-rank. This phenomenon that has recently been motivated by generic latent-variable models (Udell and Townsend, 2019) and theoretically established in several works (Chatterjee, 2015b; Agarwal et al., 2020b). Since big data matrices with underlying structure are common in biological applications (due, for example, to similarities between cell types and drugs, and interdependence of gene expression values), these works suggest linear factor models as a natural starting point for developing principled causal imputation methods. Further, low-rank approximation can often be empirically verified by inspecting the spectrum of the observations, as we do in Section 6 (see Figure 5a).

Assumption 1 *Given a target context c and target action a, there exists $\beta \in \mathbb{R}^{|\mathcal{A}(c)|}$ such that $\mathbf{v}^a = \sum_{j \in \mathcal{A}(c)} \beta_j \mathbf{v}^j$.*

Interpretation. Under the linear factor model, this is a mild assumption, particularly in applications such as genomics, where outcomes are measured for many different actions and contexts. By definition, such a β exists if $\mathbf{v}^a$ and $[\mathbf{v}^j]_{j \in \mathcal{A}(c)}$ are linearly dependent. Recall $\mathbf{v}^a$ and $\mathbf{v}^j$ are in $\mathbb{R}^r$. If $r \ll |\mathcal{A}(c)|$, then by the definition of rank, it is easy to see that the 'pathological' case where $\mathbf{v}^a$ is linearly independent of $\{\mathbf{v}^j : j \in \mathcal{A}(c)\}$ is unlikely to hold; that is, in the worst case, this undesirable event occurs for at most r actions out of all possible $\mathcal{A}$, which is a small fraction if $r \ll |\mathcal{A}(c)| \leq |\mathcal{A}|$.

Assumption 2 *Given a target context c and target action a, let* $\text{rowspan}(\mathbf{x}_{\text{test}}) \subseteq \text{rowspan}(\mathbf{x}_{\text{train}})$,[2] *where $\mathbf{x}_{\text{train}}$ and $\mathbf{x}_{\text{test}}$ are defined in Section 4.*

Interpretation. This assumption states that the target context is no more "complex" than the training context in a linear algebraic sense. This is the key assumption that enables `SI-A` (and thus the original SI estimator) to generalize from a small set of experiments to novel context-actions pairs.

Theorem 1 *Suppose $\mathcal{X}$ satisfies a linear factor model. Further, for $(c, a) \notin \Omega$, suppose Assumptions 1 and 2 hold. Then $\widehat{\mathbf{x}}^{ca} = \mathbf{x}^{ca}$.*

Checking identifiability. Assumption 2 enables SI to recover $\mathcal{X}$ from a sparse set of observations Ω (Theorem 1). As such, Agarwal et al. (2020a) proposed a data-driven hypothesis test to check when this condition is satisfied in practice. Their test statistic $\widehat{\tau} = \|\mathbf{v}_{\text{test}} - \mathbf{v}_{\text{train}} \mathbf{v}_{\text{train}}^T \mathbf{v}_{\text{test}}\|_F^2$ measures the gap between the rowspaces of $\mathbf{X}_{\text{train}}$ and $\mathbf{X}_{\text{test}}$, where $\mathbf{v}_{\text{train}}$ and $\mathbf{v}_{\text{test}}$ are the right singular vectors of $\mathbf{X}_{\text{train}}$ and $\mathbf{X}_{\text{test}}$, respectively. Agarwal et al. (2020a) derived a critical value τ_α,

1. Formally, $\mathcal{X} = U \times_2 V$ for some $U \in \mathbb{R}^{|\mathcal{C}| \times r \times p}$, $V \in \mathbb{R}^{|\mathcal{A}| \times r}$, where $\times_2$ denotes the *mode-2* product.
2. The row span of a matrix $M \in \mathbb{R}^{m \times n}$ is the linear subspace of $\mathbb{R}^n$ spanned by the row vectors of M.

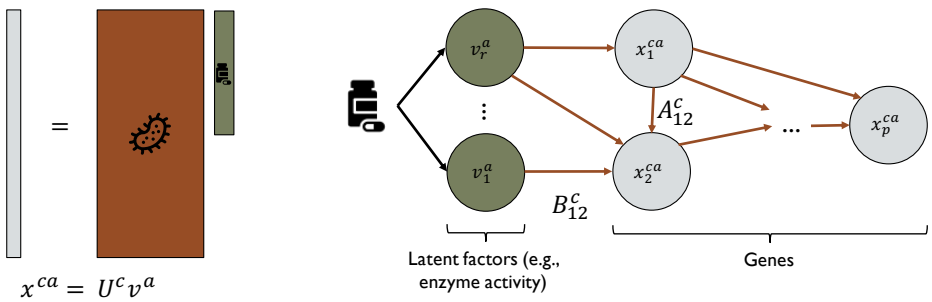


Figure 2: The assumed linear factor model and a linear structural equation model giving rise to it.

such that under the null hypothesis H_0 where Assumption 2 holds, $\mathbb{P}(\widehat{\tau} \geq \tau_\alpha \mid H_0) \leq \alpha$. However, their critical value depends on the underlying parameters of an underlying noise distribution. Instead of estimating these parameters, we follow a heuristic based on the interpretation of $\widehat{\tau}$ as the spectral energy of $\mathbf{v}_{\text{test}}$ that does *not* belong within $\text{span}(\mathbf{v}_{\text{train}})$. Specifically, we fix $\rho \in [0, 1]$ and reject H_0 if $\widehat{\tau} \geq \rho \cdot \text{rank}(\mathbf{v}_{\text{test}})$, i.e., if more than ρ fraction of the spectral energy in $\mathbf{v}_{\text{test}}$ lies outside of $\text{span}(\mathbf{v}_{\text{train}})$.

Rejection of the test implies that `SI-A` may not perform well on our prediction task, since Assumption 2 is unlikely to hold. In practice, this allows us to switch to a simpler baseline estimator, as we show in Section 6. This hypothesis test also suggests an elegant method for picking a set of donor actions, as discussed in Section 4. For a fixed significance level, we may aim to find a set passing the hypothesis test which *maximizes* the number of training contexts. Unfortunately, this induces a combinatorial optimization problem which may be difficult to solve when there are many actions. We consider two computationally efficient alternatives. First, we may greedily pick actions, according to whichever action *least* reduces the number of training contexts, until the set passes the hypothesis test. Second, we may *always* use $\mathcal{A}(c)$ as the donor set, then use the hypothesis test to decide on whether to use the `SI-A` estimate or a simpler baseline.

5.2. Connecting SI to Structural Causal Models

Here, we provide motivation for a linear factor model in terms of linear structural causal models, which have been frequently used as models of genomic networks (Friedman et al., 2000; Badsha et al., 2019). In particular, we consider the following set of models:

Definition 2 *A (noiseless)* linear structural equation model *(SEM) over the vector* $\mathbf{z} \in \mathbb{R}^p$ *is defined by the set of equations*

$$\mathbf{z}_i = \sum_{j \in \text{pa}_G(i)} A_{ij} \mathbf{z}_j,$$

where $\text{pa}_G(i)$ *denotes the* parents *of node* i *in the directed acyclic graph* G.

A particular subclass of linear structural equation models implies a factor model. In particular, we consider the following assumption on the SEM, illustrated in Fig. 2.

Assumption 3 *Given context* c*, assume* $\mathbf{z}^{ca} = (\mathbf{x}^{ca}, \mathbf{v}^a)$ *satisfies a linear structural equation model with coefficients* A^c, B^c*, and that* $\text{pa}_{\mathcal{G}}(\mathbf{v}_i) = \{a\}$ *for all* $i \in [r]$*. Equivalently,*

$$\mathbf{x}^{ca} = A^c \mathbf{x}^{ca} + B^c \mathbf{v}^a \tag{1}$$

Further assume that and $\mathbf{v}_i$ *is unobserved for all* $i \in [p]$.

Proposition 1 *Under Assumption 3,* $\mathcal{X}$ *satisfies a linear factor model.*

Proof Under acyclicity of G, A^c is lower-triangular up to a permutation, so that $(I - A^c)$ is invertible, and we may re-write $\mathbf{x}^{ca}$ as

$$\mathbf{x}^{ca} = \underbrace{(I - A^c)^{-1} B^c}_{U^c} \mathbf{v}^a$$

Defining U^c as $(I - A^c)^{-1} B^c$ completes the proof. ■

Interpretation. Although Proposition 1 is a simple observation, to the best of our knowledge, this connection has not been previously established. The SI framework of Agarwal et al. (2020a), and panel data literature more broadly (Abadie et al., 2010; Arkhangelsky et al., 2019; Athey et al., 2017; Bai, 2009), often imposes a linear factor model structure on $\mathbf{x}^{ca}$. Proposition 1 provides a novel motivation for these models through the lens of structural causal models. In our application, $\mathbf{v}^a$ might represent biological factors, e.g., molecule concentrations or enzyme activity levels, while U^c represents the "gene expression program" (Pope and Medzhitov, 2018) run by cell type c.

6. Empirical Results

In this section, we perform extensive experimentation on causal imputation for the CMAP dataset.

CMAP dataset. Subramanian et al. (2017) developed the L1000 assay, which allows for cost-effective measurement of the gene expression signatures. This assay measures the transcription levels of 978 "landmark" genes, picked via a data-driven approach based on their ability to recover information about the rest of the transcriptome. L1000 signatures have been measured from over 1,000,000 different samples, covering 71 different cell types and over 20,000 different chemical compounds. We randomly sample 100 of these compounds, along with the "control" compound, *DMSO*, to create a smaller, unbiased version of the dataset. For each cell-type, compound pair, we average all corresponding signatures. This gives a dataset of 519 gene expression signatures. A detailed evaluation of the dataset and a description of our preprocessing pipeline is described in Appendix C. The dataset can be accessed at https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE92742.

Baseline algorithms. Figure 3 shows an embedding of 11,185 gene expression vectors via UMAP (McInnes et al., 2018), colored by cell type. Clearly, most of the variation in the data is due to cell type rather than the compound applied to the cell. This is supported by Fig. 8 in Appendix D, where we see that most compound-induced expressions falls within the normal variation of the cell type; the additional variation due to compounds is minor. This suggests a natural *mean-over-actions* baseline

$$\widehat{\mathbf{x}}^{ca}_{\text{avg-a}} = \frac{1}{|\mathcal{A}(c)|} \sum_{a' \in \mathcal{A}(c)} \mathbf{x}^{ca'}. \tag{2}$$

Indeed, it is well-known that the effect of chemical compounds is much smaller than the effect of cell type, so that the simple mean-over-actions estimator already performs well, as we examine shortly. We consider three other natural baselines (see equations in Appendix E). The *mean-over-contexts* is

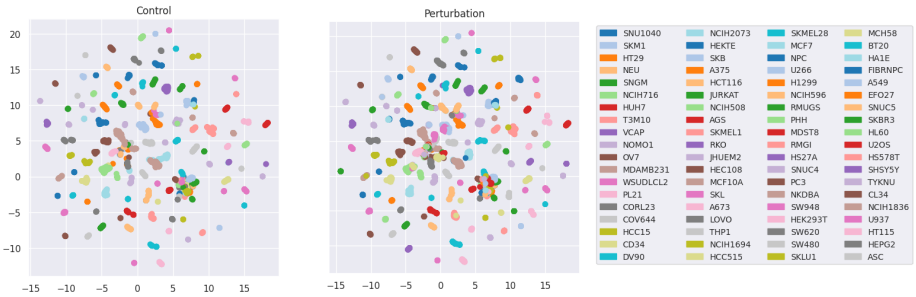


Figure 3: UMAP embedding of gene expression data, colored by cell type.

analogous to the mean-over-actions estimator, with the average taken over different contexts from the same action. We may combine these estimators into the parametric *two-way mean* estimator with parameter $\lambda_c \in [0, 1]$, by taking a convex combination of the *mean-over-actions* and *mean-over-contexts* estimators. We use $\lambda_c = 0.5$. The *fixed action effect estimator*, discussed in Section 2, is defined by computing the average shift induced by action a compared to the control, and adding that shift to the control for c. Finally, we compare against a variant of `SI-A` that re-scales contexts rather than actions, as discussed in Section 4, which we refer to as `SI-C`. Note that standard imputation methods are not scalable to the CMAP dataset: the MissForest implementation in `missingpy` takes 2.5 hours per prediction, and would take $2.5 \times 519/24 \approx 54$ days to run on the subsampled data. `IterativeImputer`, a version of MICE in `sklearn`, is somewhat more scalable, but still prohibitively slow on our subsampled data—see Appendix F.1 for a comparison on 120 signatures.

6.1. Prediction Error

For each algorithm, we measure performance using a leave-one-out (LOO) procedure. In particular, for each cell type and compound pair $(c, a) \in \Omega$ that is measured, we remove the true gene expression signature $\mathbf{x}^{ca}$ from the dataset, and use the remainder of the dataset to estimate $\widehat{\mathbf{x}}^{ca}$. We measure the accuracy of each estimate $\widehat{\mathbf{x}}^{ca}$ in two ways. Normalized root-mean-square error (NRMSE), $\|(\widehat{\mathbf{x}}^{ca} - \mathbf{x}^{ca})/p\|_2/\text{IQR}(\mathbf{x}^{ca})$, reported in Fig. 4, measures the absolute accuracy to predict the missing gene expression vector[3]. We show NRMSE since it is insensitive to scaling and shifting, but the relative ordering between methods is insensitive to the metric, as seen in Appendix F.2. Meanwhile, cosine similarity between the predicted shift $\widehat{\mathbf{x}}^{ca} - \mathbf{x}^{ca_0}$ and the true shift $\mathbf{x}^{ca} - \mathbf{x}^{ca_0}$ (where a_0 is the control action), reported in Fig. 4, measures the degree of *alignment* between the true effect of drug a in cell type c and the predicted effect. Appendix F.3 shows that `SI-A` and the baselines are roughly similar in terms of computation time.

The results of each estimator provide insight into the relationship between cell types and compounds. As expected from Figure 3, the mean-over-actions estimator is a strong baseline, with a median NRMSE of 0.41. In contrast, the mean-over-contexts estimator performs quite poorly. Seeing Figure 3, this is expected: the substantial variation *between* different cell types suggests that each cell type is *not* representable as a linear combination of the others. The fixed effect estimator, using $a' = \text{DMSO}$, performs similarly to the mean-over-actions estimator, indicating its prediction quality is dominated by the cell-type-specific baseline $\mathbf{x}^{ca'}$, i.e., the shift vector for the compound is small. Finally, `SI-A` performs best, with a median NRMSE of 0.34, a 17% reduction compared to the best

3. $\text{IQR}(\mathbf{x}^{ca})$ denotes the *interquartile range* for the vector $\mathbf{x}^{ca}$.

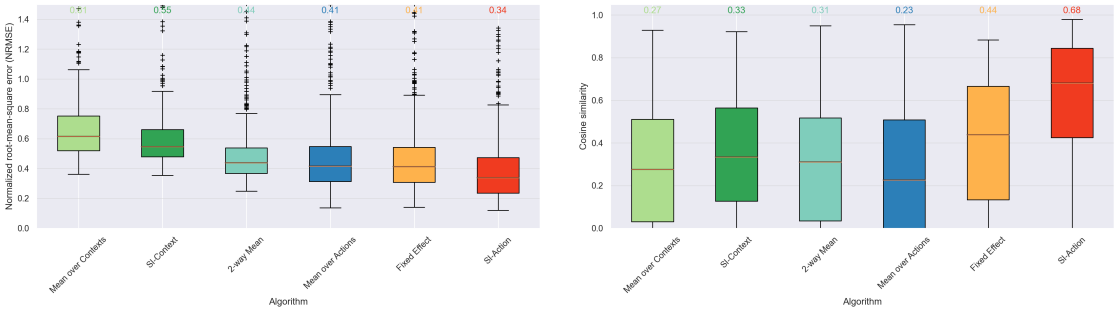
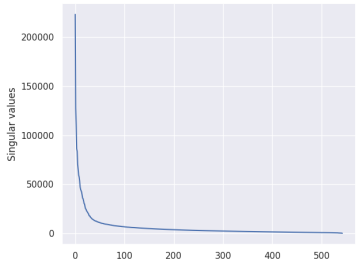
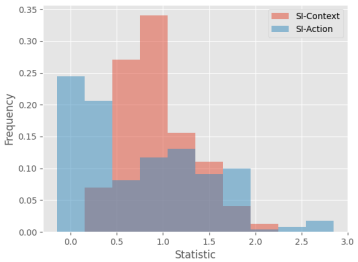


Figure 4: SI−A outperforms baselines on causal imputation in the CMAP dataset in terms of NRMSE (**left**) and cosine similarity (**right**) between the true and predicted effects of drugs in each cell type.



(a) Spectrum of subsampled dataset. 95% of spectral energy is captured in the top 53 singular values.



(b) Distribution of $\widehat{\tau}$ when regressing along cell type (context) and compound (action) dimensions.

(c) NRMSE as a function of donor size and training contexts. Error bars show 0.1 standard deviations.

Figure 5: Explaining efficacy of SI−A for CMAP dataset.

baseline. The findings on cosine similarity corroborate these results: the drug effects predicted by SI−A are significantly more aligned with the true drug effects compared to the baselines.

SI−A takes a weighted linear combination of signatures for a given cell type (across different compounds), in contrast to *mean-over-actions* baseline which simply takes equal weights across these signatures. It is noteworthy that such weights are learnt from signatures across compounds for *different* cell types, and yet it is still effective to *transfer* the learnt model to other cell types. This gives credence to the factor model laid out in Section 5.2 holding in our setting. In contrast, SI−C performs almost as poorly as the mean-over-contexts baseline, indicating again that the difference between cell types is too great to express one cell type as a linear combination of others.

Notably, we highlight that we do *not* de-noise the data in SI−A or SI−C to achieve their optimal empirical results, i.e., the results in Figure 4 do not use any ME algorithm to remove high amounts of noise that might be present. De-noising is not needed in our setting as the data is implicitly de-noised, since each measurement is an average of multiple observations for a given measured cell type and compound pair $(c, a) \in \Omega$ in the data. Indeed, as shown in Appendix G, applying ME does improve prediction if we restrict ourselves to use only a single sample per observed $(c, a) \in \Omega$.

SI−A is effective in biological settings. As discussed in Section 5, a visual inspection of the spectrum of our observations helps verify the validity of the linear factor model and (by extension) Assumption 1. Fig. 5a shows the singular values of the 519×978 matrix of gene expression signa-

tures, where over 95% of the spectrum is captured by the top 53 singular values, i.e., $r \approx 53$; hence, the CMAP dataset exhibits the desired low-rank factor structure. To verify Assumption 2, we plot frequencies of the test statistic $\widehat{\tau}$ (detailed in Section 4) in Figure 5b. A higher value of $\widehat{\tau}$ indicates that Assumption 2 is less likely to hold, and hence the results of our method are less meaningful. We see that the test statistic tends to be lower when regressing in the action dimension, explaining the superior performance of `SI-A` as compared to `SI-C`. This suggests that `SI-A` may be a useful tool in other biological settings, where similar invariance patterns and data availability are common.

`SI-A` takes advantage of data structure. Finally, to understand how the performance of `SI-A` depends on the number of donor actions (i.e., $|\mathcal{A}(c)|$) and training contexts (i.e., $|\mathcal{C}_{\text{train}}|$), we restrict our attention to context-action pairs with at least 10 donor actions and at least 5 training contexts, resulting in 75 pairs. For each context-action pair (c, a), and each $(i, j) \in \{1, \ldots, 10\} \times \{1, \ldots, 5\}$, we perform `SI-A` with i random donor actions and j random training contexts. In Fig. 5c, we show the average NRMSE of the predictions over all 75 pairs, for each tuple (i, j). As expected, increasing the number of donor actions leads to a large increase in performance, and increasing the number of training contexts also leads to a large increase in performance, especially when there are few donor actions. The original SI estimator always has $|\mathcal{C}_{\text{train}}| = 1$, but our results highlight the value of using *all* available data to learn the regression model, as is done by `SI-A`.

7. Discussion

In this paper, we introduced the `SI-A` estimator, an extension of the SI estimator of Agarwal et al. (2020a), for use on a task we call *causal imputation*: predicting the effect of an action across different contexts. We showed that the `SI-A` estimator provides valid estimates of unseen outcomes under a linear factor model, which we motivate via a connection to structural causal models. We demonstrated the superior performance of `SI-A` to other baselines on the task of causal imputation in the CMAP dataset, an important source of information for predicting the effect of various compounds on gene expression.

Several important directions are left open for future work, of which we cover only a few. First, the tradeoff between the number of donor actions and the number of training contexts raises the need for a principled method for picking "optimal" donor sets. One promising approach may be to frame this choice as a combinatorial optimization problem, where the objective function may be submodular under some assumptions on the problem structure. A related question is whether we can apply SI in a *sequential manner* to infer which samples are most informative to reduce sample complexity in an experimental design and/or active learning framework.

Another important direction for future work is on *nonlinear* methods to the causal imputation problem. Genomic data is known to exhibit highly nonlinear relationships, so that our model in Section 5.2 is only a coarse approximation. A straightforward nonlinear extension of our method would be to perform `SI-A` in a latent space learned by an autoencoder. Two concepts from this paper are likely to be useful in the development and analysis of nonlinear methods. First, we demonstrated that it is beneficial to develop representations for each action which are invariant to the context in which they occur, allowing for the effect of the action to be *transported* between contexts. Second, our mechanistic explanation in Section 5.2 for the success of `SI-A` may serve as a starting point for explaining the success of nonlinear methods.

References

Alberto Abadie, Alexis Diamond, and Jens Hainmueller. Synthetic control methods for comparative case studies: Estimating the effect of California's tobacco control program. *Journal of the American Statistical Association*, 2010.

Anish Agarwal, Muhammad Jehangir Amjad, Devavrat Shah, and Dennis Shen. Model agnostic time series analysis via matrix estimation. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2(3):1–39, 2018.

Anish Agarwal, Devavrat Shah, and Dennis Shen. Synthetic interventions, 2020a.

Anish Agarwal, Devavrat Shah, Dennis Shen, and Dogyoon Song. On robustness of principal component regression, 2020b.

Dmitry Arkhangelsky, Susan Athey, David A Hirshberg, Guido W Imbens, and Stefan Wager. Synthetic difference in differences. Technical report, National Bureau of Economic Research, 2019.

Susan Athey, Mohsen Bayati, Nikolay Doudchenko, and Guido Imbens. Matrix completion methods for causal panel data models. 2017.

Md Badsha, Audrey Qiuyan Fu, et al. Learning causal biological networks with the principle of Mendelian randomization. *Frontiers in Genetics*, 10:460, 2019.

Jushan Bai. Panel data models with interactive fixed effects. *Econometrica*, 77(4):1229–1279, 2009. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/40263859.

Elias Bareinboim and Judea Pearl. Transportability from multiple environments with limited experiments: Completeness results. In *Advances in Neural Information Processing Systems*, pages 280–288, 2014.

Elias Bareinboim and Judea Pearl. Causal inference and the data-fusion problem. *Proceedings of the National Academy of Sciences*, 113(27):7345–7352, 2016.

Robert M Bell and Yehuda Koren. Lessons from the netflix prize challenge. *Acm Sigkdd Explorations Newsletter*, 9(2):75–79, 2007.

Anastasiya Belyaeva, Louis Cammarata, Adityanarayanan Radhakrishnan, Chandler Squires, Karren D Yang, GV Shivashankar, and Caroline Uhler. Causal network models of SARS-CoV-2 expression and aging to identify candidates for drug repurposing. *arXiv preprint arXiv:2006.03735*, 2020.

Dimitris Bertsimas, Colin Pawlowski, and Ying Daisy Zhuo. From predictive methods to missing data imputation: an optimization approach. *The Journal of Machine Learning Research*, 18(1): 7133–7171, 2017.

Rasmus Bro. Parafac. tutorial and applications. *Chemometrics and intelligent laboratory systems*, 38(2):149–171, 1997.

Sourav Chatterjee. Matrix estimation by universal singular value thresholding. *The Annals of Statistics*, 43(1):177–214, 2015a.

Sourav Chatterjee. Matrix estimation by universal singular value thresholding. *Annals of Statistics*, 43:177–214, 2015b.

Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P Fulco, Livnat Jerby-Arnon, Nemanja D Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, et al. Perturb-seq: dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens. *Cell*, 167 (7):1853–1866, 2016.

Oana M Enache, David L Lahr, Ted E Natoli, Lev Litichevskiy, David Wadden, Corey Flynn, Joshua Gould, Jacob K Asiedu, Rajiv Narayan, and Aravind Subramanian. The gctx format and cmap {Py, R, M, J} packages: resources for optimized storage and integrated traversal of annotated dense matrices. *Bioinformatics*, 35(8):1427–1429, 2019.

Nir Friedman, Michal Linial, Iftach Nachman, and Dana Pe'er. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3-4):601–620, 2000.

Richard A Harshman et al. Foundations of the parafac procedure: Models and conditions for an" explanatory" multimodal factor analysis. 1970.

Markus Hofmarcher, Elisabeth Rumetshofer, Djork-Arne Clevert, Sepp Hochreiter, and Günter Klambauer. Accurate prediction of biological assays with high-throughput microscopy images and convolutional networks. *Journal of Chemical Information and Modeling*, 59(3):1163–1171, 2019.

Prateek Jain and Sewoong Oh. Provable tensor factorization with missing data. In *Advances in Neural Information Processing Systems*, pages 1431–1439, 2014.

Brian A Kidd, Aleksandra Wroblewska, Mary R Boland, Judith Agudo, Miriam Merad, Nicholas P Tatonetti, Brian D Brown, and Joel T Dudley. Mapping the effects of drugs on the immune system. *Nature biotechnology*, 34(1):47, 2016.

Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3): 455–500, 2009.

Sanghack Lee, Juan D Correa, and Elias Bareinboim. General identifiability with arbitrary surrogate experiments. In *Uncertainty in Artificial Intelligence*, pages 389–398. PMLR, 2020.

Mohammad Lotfollahi, Alexander F Wolf, and Fabian J Theis. scGen predicts single-cell perturbation responses. *Nature Methods*, 16(8):715, 2019.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Scott D Pope and Ruslan Medzhitov. Emerging principles of gene expression programs and their regulation. *Molecular cell*, 71(3):389–397, 2018.

Donald B Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

Ilya Shpitser and Judea Pearl. Complete identification methods for the causal hierarchy. *Journal of Machine Learning Research*, 9:1941–1979, 2008.

Aude Sportisse, Claire Boyer, and Julie Josses. Estimation and imputation in probabilistic principal component analysis with missing not at random data. *Advances in Neural Information Processing Systems*, 33, 2020.

Daniel J Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackerman, et al. A deep learning approach to antibiotic discovery. *Cell*, 180(4):688–702, 2020.

Aravind Subramanian, Rajiv Narayan, Steven M Corsello, David D Peck, Ted E Natoli, Xiaodong Lu, Joshua Gould, John F Davis, Andrew A Tubelli, Jacob K Asiedu, et al. A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. *Cell*, 171(6):1437–1452, 2017.

Meng How Tan, Qin Li, Raghuvaran Shanmugam, Robert Piskol, Jennefer Kohler, Amy N Young, Kaiwen Ivy Liu, Rui Zhang, Gokul Ramaswami, Kentaro Ariyoshi, et al. Dynamic landscape and regulation of rna editing in mammals. *Nature*, 550(7675):249–254, 2017.

Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001.

Madeleine Udell and Alex Townsend. Why are big data matrices approximately low rank? *SIAM Journal on Mathematics of Data Science*, 1(1):144–160, 2019.

Stef Van Buuren and Karin Oudshoorn. *Flexible multivariate imputation by MICE*. Leiden: TNO, 1999.

Akbar K Waljee, Ashin Mukherjee, Amit G Singal, Yiwei Zhang, Jeffrey Warren, Ulysses Balis, Jorge Marrero, Ji Zhu, and Peter DR Higgins. Comparison of imputation methods for missing laboratory data in medicine. *BMJ open*, 3(8), 2013.

Karren D Yang, Abigail Katcoff, and Caroline Uhler. Characterizing and learning equivalence classes of causal DAGs under interventions. In *International Conference on Machine Learning*, pages 5541–5550, 2018.

Karren D Yang, Anastasiya Belyaeva, Saradha Venkatachalapathy, Karthik Damodaran, Abigail Katcoff, Adityanarayanan Radhakrishnan, G. V. Shivashankar, and Caroline Uhler. Multi-domain translation between single-cell imaging and sequencing data using autoencoders. *Nature Communications*, 12:31, 2019.