Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions

Magliacane, S.; van Ommen, T.; Claassen, T.; Bongers, S.; Versteeg, P.; Mooij, J.M.

Published in:
32nd Conference on Neural Information Processing Systems 2018

Creative Commons License (see https://creativecommons.org/use-remix/cc-licenses):
Unspecified

Citation for published version (APA):
Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions

Sara Magliacane
IBM Research
sara.magliacane@gmail.com

Thijs van Ommen
University of Amsterdam
thijsvanommen@gmail.com

Tom Claassen
Radboud University Nijmegen
tomc@cs.ru.nl

Stephan Bongers
University of Amsterdam
srbongers@gmail.com

Philip Versteeg
University of Amsterdam
p.j.j.p.versteeg@uva.nl

Joris M. Mooij
University of Amsterdam
j.m.mooij@uva.nl

Abstract

An important goal common to domain adaptation and causal inference is to make accurate predictions when the distributions for the source (or training) domain(s) and target (or test) domain(s) differ. In many cases, these different distributions can be modeled as different contexts of a single underlying system, in which each distribution corresponds to a different perturbation of the system, or in causal terms, an intervention. We focus on a class of such causal domain adaptation problems, where data for one or more source domains are given, and the task is to predict the distribution of a certain target variable from measurements of other variables in one or more target domains. We propose an approach for solving these problems that exploits causal inference and does not rely on prior knowledge of the causal graph, the type of interventions or the intervention targets. We demonstrate our approach by evaluating a possible implementation on simulated and real world data.

1 Introduction

Predicting unknown values based on observed data is a problem central to many sciences, and well studied in statistics and machine learning. This problem becomes significantly harder if the training and test data do not have the same distribution, for example because they come from different domains. Such a distribution shift can happen whenever the circumstances under which the training data were gathered are different from those for which the predictions are to be made. A rich literature exists on this problem of domain adaptation, a particular task in the field of transfer learning; see e.g. Quinonero-Candela et al. (2009); Pan and Yang (2010) for overviews.

When the domain changes, so may the relations between the different variables under consideration. While for some sets of variables $A$, a function $f : A \rightarrow Y$ learned in one domain may continue to offer good predictions for $Y \in Y$ in a different domain, this may not be true of other sets $A'$ of variables. Causal graphs (e.g., Pearl [2009] Spirtes et al. [2000]) allow us to reason about this in a principled way when the domains correspond to different external interventions on the system, or more generally, to different contexts in which a system has been measured. Knowledge of the causal graph that describes the data generating mechanism, and of which parts of the model are invariant

*Most of the work was performed while at the University of Amsterdam.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.
across the different domains, allows one to transfer knowledge from one domain to the other in order to address the problem of domain adaptation (Spirtes et al., 2000; Storkey, 2009; Schölkopf et al., 2012; Bareinboim and Pearl, 2016).

Over the last years, various methods have been proposed to exploit the causal structure of the data generating process in order to address certain domain adaptation problems, each relying on different assumptions. For example, Bareinboim and Pearl (2016) provide theory for identifiability under transfer (“transportability”) assuming that the causal graph is known, that interventions are perfect, and that the intervention targets are known. Hyttinen et al. (2015) also assume perfect interventions with known targets but do not rely on complete knowledge of the causal graph, instead inferring the relevant aspects of it from the data. Rojas-Carulla et al. (2018) make the assumption that if the conditional distribution of the target given some subset of covariates is invariant across different source domains, then this conditional distribution must also be the same in the target domain. The methods proposed in (Schölkopf et al., 2012; Zhang et al., 2013, 2015; Gong et al., 2016) all address challenging settings in which conditional independences that follow from the usual Markov and faithfulness assumptions alone do not suffice to solve the problem, but additional assumptions on the data generating process have to be made.

In this work, we will make no such additional assumptions, and address the setting in which both the causal graph and the intervention types and targets may be (partially) unknown. Our contributions are the following. We consider a set of relatively weak assumptions that make the problem well-posed. We propose an approach to solve this class of causal domain adaptation problems that can deal with the presence of latent confounders. The main idea is to select the subset of features $A$ that leads to the best predictions of $Y$ in the source domains, while satisfying invariance (i.e., $P(Y | A)$ is the same in the source and target domains). To test whether the invariance condition is satisfied, we apply the recently proposed Joint Causal Inference (JCI) framework (Mooij et al., 2018) to exploit the information provided by multiple domains corresponding to different interventions. The basic idea is as follows. First, a standard feature selection method is applied to source domains data to find sets of features that are predictive of a target variable, trading off bias and variance, but unaware of changes in the distribution across domains. A causal inference method then draws conclusions from all given data about the possible causal graphs, avoiding sets of features for which the predictions would not transfer to the target domains. We propose a proof-of-concept implementation of our approach building on a causal discovery algorithm by Hyttinen et al. (2014). We evaluate the method on synthetic data and a real-world example.

2 Theory

Before giving a precise definition of the class of domain adaptation problems that we consider in this work, we begin with a motivating example.

Figure 1: In this scenario, an intervention $C_1$ leads to a shift of distribution between source domain and target domain (see also Example 1). Green crosses show source domain data ($C_1 = 0$), blue circles show target domain data ($C_1 = 1$). A standard feature selection method that does not take into account the causal structure, but would use $X_3$ to predict $Y := X_2$ (because $X_3$ is a good predictor of $Y$ in the source domain), would obtain extremely biased predictions in the target domain. Using $X_1$ instead yields less accurate predictions in the source domain, but much more accurate ones in the target domain.
Example 1. We are given three variables \(X_1, X_2, X_3\) describing different aspects of a system (for example, certain blood cell phenotypes in mice). We have observational measurements of these three variables (the source domain, designated with \(C_1 = 0\)), and in addition, measurements of \(X_1\) and \(X_3\) under an intervention (the target domain, designated with \(C_1 = 1\)), e.g., in which the mice have been exposed to a certain drug. The domain adaptation task is to predict the values of \(Y \equiv X_2\) in the interventional target domain (i.e., when \(C_1 = 1\)). Let us assume for this example that the causal graph in Figure 1a applies, i.e., we assume that \(X_2\) is affected by \(X_1\) and affects \(X_3\), while \(C_1\) affects both \(X_1\) and \(X_3\) (i.e., the intervention targets the variables \(X_1\) and \(X_3\)). This causal graph implies \(P(Y \mid X_1, C_1 = 0) = P(Y \mid X_1, C_1 = 1)\). Suppose further that the relation between \(X_1\) and \(X_2\) is about equally strong as the relation between \(X_2\) and \(X_3\), but considerably more noisy. Then a feature selection method using only available source domain data, and aiming to select the best subset of features to use for prediction of \(Y\) will prefer both \(\{X_3\}\) and \(\{X_1, X_3\}\) over \(\{X_1\}\) (because predicting \(Y\) from \(X_1\) leads to larger variance than predicting \(Y\) from \(X_3\), and to a larger bias than predicting \(Y\) from both \(X_1\) and \(X_3\)). However, under the intervention \((C_1 = 1)\), \(P(Y \mid X_3)\) and \(P(Y \mid X_1, X_3)\) both change, so that using those features to predict \(Y\) in the target domain could lead to extreme bias, as illustrated in Figure 1b. Because the conditional distribution of \(Y\) given \(X_1\) is invariant across domains, as illustrated in Figure 1b, predictions of \(Y\) based only on \(X_1\) can be safely transferred to the target domain.

This example provides an instance of a domain adaptation problem where feature selection methods that do not take into account the causal structure would pick a set of features that does not generalize to the target domain, and may lead to arbitrarily bad predictions (even asymptotically, as the number of data points tends to infinity). On the other hand, correctly taking into account the causal structure and the possible distribution shift from source to target domain allows to upper bound the prediction error in the target domain, as we will see in Section 2.3.

2.1 Problem Setting

We now formalize the domain adaptation problems that we address in this paper. We will make use of the terminology of the recently proposed Joint Causal Inference (JCI) framework [Mooij et al. 2018].

Let us consider a system of interest described by a set of system variables \(\{X_i\}_{i \in \mathcal{I}}\). In addition, we model the domain in which the system has been measured by context variables \(\{C_i\}_{i \in \mathcal{I}}\) (we will use “context” as a synonym for “domain”). We will denote the tuple of all system and context variables \(Y = (\{X_i\}_{i \in \mathcal{I}}, \{C_i\}_{i \in \mathcal{I}})\). System and context variables can be discrete or continuous. As a concrete example, the system interest could be a mouse. The system variables could be blood cell phenotypes such as the concentration of red blood cells, the concentration of white blood cells, and the mean red blood cell volume. The context variables could indicate for example whether a certain gene has been knocked out, the dosage of a certain drug administered to the mice, the age and gender of the mice, or the lab in which the measurements were done. The important underlying assumption is that context variables are exogenous to the system, whereas system variables are endogenous. The interventions are not limited to the perfect (“surgical”) interventions modeled by the do-operator of Pearl [2009], but can also be other types of interventions such as mechanism changes (Tian and Pearl [2001]), soft interventions (Markowitz et al. [2005]), fat-hand interventions (Eaton and Murphy [2007], activity interventions (Mooij and Heskes [2013]), and stochastic versions of all these. Knowledge of the intervention targets is not necessary (but is certainly helpful). For example, administering a drug to the mice may have a direct causal effect on an unknown subset of the system variables, but we can simply model it as a binary exogenous variable (indicating whether or not the drug was administered) or a continuous exogenous variable (describing the dosage of the administered drug) without specifying in advance on which variables it has a direct effect. We can now formally state the domain adaptation task that we address in this work:

Task 1 (Domain Adaptation Task). We are given data for a single or for multiple source domains, in each of which \(C_1 = 0\), and for a single or for multiple target domains, in each of which \(C_1 = 1\). Assume the source domains data is complete (i.e., no missing values), and the target domains data is complete with the exception of all values of a certain target variable \(Y = X_j\). The task is to predict these missing values of the target variable \(Y\) given the available source and target domains data.

More precisely, we should say that \(P(Y \mid X_3, C_1 = 1)\) may differ from \(P(Y \mid X_3, C_1 = 0)\), and similarly when conditioning on \(\{X_1, X_3\}\).
We will make the following assumptions on the causal structure (where henceforth we will simply
(JCI Assumptions)
Assumption 1
\[ \text{G}\]
write
\[ \text{G}\]
with respect to the ADMG
we assume a factorizing probability distribution on the exogenous variables. There could be cyclic

An example is provided in Figure 2. In the next subsection, we will formalize our assumptions to
(i.e., the labels of both system and context variables), directed edges
l
M
The SCM
M
In the acyclic case, this causal graph is an Acyclic Directed Mixed Graph (ADMG), and
l
2
Figure 2: Example of a causal domain adaptation problem. The causal graph is depicted on the right,
the corresponding data on the left. The task is to predict the missing values of \( Y = X_2 \) in the target
domains (\( C_1 = 1 \), based on the observed data from the source domains and the target domains,
without knowledge of the causal graph. See also Example 2.

2.2 Assumptions

Our first main assumption is that the data generating process (on both system and context variables)
can be represented as a Structural Causal Model (SCM) (see e.g., (Pearl, 2009)):
\[
\mathcal{M} : \begin{cases}
X_j &= f_j(X_{pa(j) \cup \mathcal{J}}, C_{pa(j) \cap \mathcal{I}}, E_{pa(j) \cap \mathcal{K}}), \quad j \in \mathcal{J} \\
p(E) &= \prod_{k \in \mathcal{K}} p(E_k) 
\end{cases}
\] (1)

Here, we introduced exogenous latent independent “noise” variables \( (E_k)_{k \in \mathcal{K}} \) that model latent
causes of the context and system variables. The parents of each variable are denoted by \( \text{PA}(\cdot) \). Each
context and system variable is related to its parent variables by a structural equation. In addition,
we assume a factorizing probability distribution on the exogenous variables. There could be cyclic
dependencies, for example due to feedback loops, but for simplicity of exposition we will discuss
only the acyclic case here, noting that the extension to the cyclic case is straightforward given recent
theoretical advances on cyclic SCMs (Bongers et al, 2018). This SCM provides a causal model for
the distributions of the various domains, and in particular, it induces a joint distribution \( P(V) \) on
the context and system variables. Note that we will assume that the data generating process can be
modeled by some model of this form, but we do not rely on knowing the precise model.

The SCM \( \mathcal{M} \) can be represented graphically by its causal graph \( \mathcal{G}(\mathcal{M}) \), a graph with nodes \( \mathcal{I} \cup \mathcal{J} \)
(i.e., the labels of both system and context variables), directed edges \( l_1 \rightarrow l_2 \) for \( l_1, l_2 \in \mathcal{I} \cup \mathcal{J} \) iff
\( l_1 \in \text{PA}(l_2) \), and bidirected edges \( l_1 \leftrightarrow l_2 \) for \( l_1, l_2 \in \mathcal{I} \cup \mathcal{J} \) iff there exists a \( k \in \text{PA}(l_1) \cap \text{PA}(l_2) \cap \mathcal{K} \).
In the acyclic case, this causal graph is an Acyclic Directed Mixed Graph (ADMG), and \( \mathcal{M} \) is also

We will make the following assumptions on the causal structure (where henceforth we will simply
write \( \mathcal{G} \) instead of \( \mathcal{G}(\mathcal{M}) \), which are discussed in detail by Mooij et al. (2018):
Assumption 1 (JCI Assumptions). Let \( \mathcal{G} \) be a causal graph with variables \( V \) (consisting of system
variables \( \{X_j\}_{j \in \mathcal{J}} \) and context variables \( \{C_i\}_{i \in \mathcal{I}} \).
(i) No system variable directly causes any context variable (“exogeneity”)
\( (\forall j \in \mathcal{J}, \forall i \in \mathcal{I} : X_j \rightarrow C_i \notin \mathcal{G}) \);
(ii) No system variable is confounded with a context variable (“randomization”)
\( (\forall j \in \mathcal{J}, \forall i \in \mathcal{I} : X_j \leftrightarrow C_i \notin \mathcal{G}) \);
(iii) Every pair of context variables is purely confounded ("genericity")
\((\forall i, i' \in I : C_i \leftrightarrow C_{i'} \in \mathcal{G} \land C_i \rightarrow C_{i'} \notin \mathcal{G})\).

The first assumption is the most crucial one that captures what we mean by “context”. The other two assumptions are less crucial and could be omitted, depending on the application. For a more in-depth discussion of these modeling assumptions and on how they compare with other possible causal modeling approaches, we refer the reader to [Mooij et al., 2018]. Any causal discovery method can in principle be used in the JCI setting, but identifiability greatly benefits from taking into account the background knowledge on the causal graph from Assumption 1.

In addition, in order to be able to address the causal domain adaptation task, we will assume:

Assumption 2. Let \(\mathcal{G}\) be a causal graph with variables \(V\) (consisting of system variables \(\{X_j\}_{j \in J}\) and context variables \(\{C_i\}_{i \in I}\)), and \(\mathbb{P}(V)\) be the corresponding distribution on \(V\). Let \(C_1\) be the source/target domains indicator and \(Y = X_j\) the target variable.

(i) The distribution \(\mathbb{P}(V)\) is Markov and faithful w.r.t. \(\mathcal{G}\);
(ii) Any conditional independence involving \(Y\) in the source domains also holds in the target domains, i.e., if \(A \cup B \cup S\) contains \(Y\) but not \(C_1\) then:
\[
A \perp B \mid S \ [C_1 = 0] \implies A \perp B \mid S \ [C_1 = 1];
\]
(iii) \(C_1\) has no direct effect on \(Y\) w.r.t. \(V\), i.e., \(C_1 \rightarrow Y \notin \mathcal{G}\).

The Markov and faithfulness assumptions are standard in constraint-based causal discovery on a single domain; we apply them here on the “meta-system” composed of system and context.

Assumption 2(i) may seem non-intuitive, but as we show in the Supplementary Material, it follows from more intuitive (but stronger) assumptions, for example if both the pooled source domains distribution \(\mathbb{P}(V \mid C_1 = 0)\) and the pooled target domains distribution \(\mathbb{P}(V \mid C_1 = 1)\) are Markov and faithful to the subgraph of \(\mathcal{G}\) which excludes \(C_1\). These stronger assumptions imply that the causal structure (i.e., presence or absence of causal relationships and confounders) of the other variables is invariant when going from source to target domains. Assumption 2(ii) is a weakened version of these more natural assumptions, allowing additional independences to hold in the target domains compared to the source domains, e.g., when \(C_1\) models a perfect surgical intervention.

Assumption 2(iii) is strong, yet some assumption of that type seems necessary to make the task well-defined. Without any information at all about the target(s) of \(C_1\), or the causal mechanism that determines the values of \(Y\) in the target domains, predicting the values of \(Y\) for the target domains seems generally impossible. Note that the assumption is more likely to be satisfied if the interventions are believed to be precisely targeted, and gets weaker the more relevant system variables are observed.

As one example of a real-world setting in which these assumptions are reasonable, consider a genomics experiment, in which gene expression levels of many different genes are measured in response to knockouts of single genes. Given our present-day understanding of the biology of gene expression, it is very reasonable to assume that the knockout of gene \(X_i\) only has a direct effect on the expression level of gene \(X_i\) itself. As long as we do not ask to predict the expression level of \(X_i\) under a knockout of \(X_i\), but only the expression level of other genes \(Y = X_j\) with \(j \neq i\), Assumption 2(iii) seems justified. It is also reasonable (based on present-day understanding of biology) to expect that a single gene knockout does not change the causal mechanisms in the rest of the system. This justifies Assumption 2(iii) in this setting if one is willing to assume faithfulness.

In the next subsections, we will discuss how these assumptions enable us to address the domain adaptation task.

---

3 Here, with \(A \perp B \mid S \ [C_1 = 0]\) we mean \(A \perp B \mid S \ [\mathbb{P}(V \mid C_1 = 0)]\), i.e., the conditional independence of \(A\) from \(B\) given \(S\) in the mixture of the source domains \(\mathbb{P}(V \mid C_1 = 0)\), and similarly for the target domains.

4 This assumption can be weakened further: in some circumstances one can infer from the data and the other assumptions that \(C_1\) cannot have a direct effect on \(Y\). For example: if there exists a descendant \(D \in \text{DE}(Y)\), and if there exists a set \(S \subseteq V \setminus \{C_1, Y\} \cup \text{DE}(Y)\), such that \(C_1 \perp D \mid S\), then \(C_1\) is not a direct cause of \(Y\) w.r.t. \(V\). For some proposals on alternative assumptions that can be made when this assumption is violated, see e.g., [Schölkopf et al., 2012; Zhang et al., 2013, 2015; Gong et al., 2016].
2.3 Separating Sets of Features

Our approach to addressing Task [1] is based on finding a separating set \( A \subseteq V \setminus \{C_1, Y\} \) of (context and system) variables that satisfies \( C_1 \perp Y \mid A [\mathcal{G}] \). If such a separating set \( A \) can be found, then the distribution of \( Y \) conditional on \( A \) is invariant under transferring from the source domains to the target domains, i.e., \( P(Y \mid A, C_1 = 0) = P(Y \mid A, C_1 = 1) \). As the former conditional distribution can be estimated from the source domains data, we directly obtain a prediction for the latter, which then enables us to predict the values of \( Y \) from the observed values of \( A \) in the target domains.\(^3\)

We will now discuss the effect of the choice of \( A \) on the quality of the predictions. For simplicity of the exposition, we make use of the squared loss function and look at the asymptotic case, ignoring finite-sample issues. When predicting \( Y \) from a subset of features \( A \subseteq V \setminus \{Y, C_1\} \) (that may or may not be separating), the optimal predictor is defined as the function \( \hat{Y} \) mapping from the range of possible values of \( A \) to the range of possible values of \( Y \) that minimizes the target domains risk \( \mathbb{E}((Y - \hat{Y}(A))^2 \mid C_1 = 1) \), and is given by the conditional expectation (regression function) \( \hat{Y}(A) := \mathbb{E}(Y \mid A = a, C_1 = 1) \). Since \( Y \) is not observed in the target domains, we cannot directly estimate this regression function from the data.

One approach that is often used in practice is to ignore the difference in distribution between source and target domains, and use instead the predictor \( \hat{Y}_A^0(a) := \mathbb{E}(Y \mid A = a, C_1 = 0) \), which minimizes the source domains risk \( \mathbb{E}((Y - \hat{Y}_A^0)^2 \mid C_1 = 0) \). This approximation introduces a bias \( \hat{Y}_A^1 - \hat{Y}_A^0 \) that we will refer to as the transfer bias (when predicting \( Y \) from \( A \)). When ignoring that source domains and target domains have different distributions, any standard machine learning method can be used to predict \( Y \) from \( A \). As the transfer bias can become arbitrarily large (as we have seen in Example [1]), the prediction accuracy of this solution strategy may be arbitrarily bad (even in the infinite-sample limit).

Instead, we propose to only predict \( Y \) from \( A \) when the set \( A \) of features satisfies the following separating set property:

\[
C_1 \perp Y \mid A [\mathcal{G}],
\]

i.e., it d-separates \( C_1 \) from \( Y \) in \( \mathcal{G} \). By the Markov assumption, this implies \( C_1 \perp Y \mid A [P(V)] \). In other words (as already mentioned above), for separating sets, the distribution of \( Y \) conditional on \( A \) is invariant under transferring from the source domains to the target domains, i.e., \( P(Y \mid A, C_1 = 0) = P(Y \mid A, Y, C_1 = 1) \). By virtue of this invariance, regression functions are identical for the source domains and target domains, i.e., \( \hat{Y}_A^0 = \hat{Y}_A^1 \), and hence also the source domain and target domain risks are identical when using the predictor \( \hat{Y}_A^0 \):

\[
C_1 \perp Y \mid A [\mathcal{G}] \implies \mathbb{E}((Y - \hat{Y}_A^0)^2 \mid C_1 = 1) = \mathbb{E}((Y - \hat{Y}_A^1)^2 \mid C_1 = 0). \tag{3}
\]

The r.h.s. can be estimated from the source domains data, and the l.h.s. equals the generalization error to the target domains when using the predictor \( \hat{Y}_A^0 \) trained on the source domains (which equals the predictor \( \hat{Y}_A^1 \) that one could obtain if all target domains data, including the values of \( Y \), were observed).\(^4\) Although this approach leads to zero transfer bias, it introduces another bias: by using only a subset of the features \( A \), rather than all available features \( V \setminus \{C_1, Y\} \), we may miss relevant information to predict \( Y \). We refer to this bias as the incomplete information bias, \( \hat{Y}_A^1 - \hat{Y}_A^0 \).

The total bias when using \( \hat{Y}_A^0 \) to predict \( Y \) is the sum of the transfer bias and the incomplete information bias:

\[
\hat{Y}_A^1 - \hat{Y}_A^0 = (\hat{Y}_A^1 - \hat{Y}_A^0) + (\hat{Y}_A^1 - \hat{Y}_A^0) - (\hat{Y}_A^1 - \hat{Y}_A^0) = \text{transfer bias} + \text{incomplete information bias}.
\]

For some problems, one may be better off by simply ignoring the transfer bias and minimizing the incomplete information bias, while for other problems, it is crucial to take the transfer into account to

\(^3\)This trivial observation is not novel; see e.g. (Ch. 7, p. 164, Spirtes et al. 2000). It also follows as a special case of (Theorem 2, Pearl and Bareinboim 2011). The main novelty of this work is the proposed strategy to identify such separating sets.

\(^4\)Note that this equation only holds asymptotically; for finite samples, in addition to the transfer from source domains to target domains, we have to deal with the generalization from empirical to population distributions and from the covariate shift if \( P(A \mid C_1 = 1) \neq P(A \mid C_1 = 0) \) (see e.g. Mansour et al. 2009).
obtain small generalization errors. In that situation, we could use any subset \( A \) for prediction that satisfies the separating set property (2), implying zero transfer bias; obviously, the best predictions are then obtained by selecting a separating subset that also minimizes the source domains risk (i.e., minimizes the incomplete information bias). We conclude that this strategy of selecting a subset \( A \) to predict \( Y \) may yield an asymptotic guarantee on the prediction error by (3), whereas simply ignoring the shift in distribution may lead to unbounded prediction error, since the transfer bias could be arbitrarily large in the worst case scenario.

### 2.4 Identifiability of Separating Feature Sets

For the strategy of selecting the best separating sets of features as discussed in Section 2.3, we need to find one or more sets \( A \subseteq V \setminus \{C_1, Y\} \) that satisfy (2). Of course, the problem is that we cannot directly test this in the data, because the values of \( Y \) are missing for \( C_1 = 1 \). Note that also Assumption (2)(i) cannot be directly used here, because it only applies when \( C_1 \) is not in \( A \cup B \). When the causal graph \( G \) is known, it is easy to verify whether (2) holds directly using d-separation. Here we address the more challenging setting in which the causal graph and the targets of the interventions are (partially) unknown.

Conceptually, one could estimate a set of possible causal graphs by using a causal discovery algorithm (for example, extending any standard method to deal with the missing conditional independence tests in \( C_1 = 1 \)), and then read off separating sets from these graphs. In practice, it is not necessary to estimate completely these causal graphs: we only need to know enough about them to verify or falsify whether a given set of features separates \( C_1 \) from \( Y \). The following example (with details in the Supplementary Material) illustrates a case where such reasoning allows us to identify a separating set.

**Example 2.** Assume that Assumptions (1) and (2) hold for two context variables \( C_1, C_2 \) and three system variables \( X_1, X_2, X_3 \) with \( Y := X_2 \). If the following conditional (in)dependencies all hold in the source domains:

\[
C_2 \perp X_2 | X_1 [C_1 = 0], \quad C_2 \not\perp X_2 | 0 [C_1 = 0], \quad C_2 \perp X_3 | X_2 [C_1 = 0], \quad (4)
\]

then \( C_1 \perp X_2 | X_1 [G] \), i.e., \( \{X_1\} \) is a separating set for \( C_1 \) and \( X_2 \). One possible causal graph leading to these (in)dependencies is provided in Figure 2 (the others are shown in Figure 1(a) in the Supplementary Material). For that ADMG, and given enough data, feature selection applied to the source domains data will generically select \( \{X_1, X_3\} \) as the optimal set of features for predicting \( Y := X_2 \), which can lead to an arbitrarily large prediction error. On the other hand, the set \( \{X_1\} \) is separating in any ADMG satisfying (4), so using it to predict \( Y \) leads to zero transfer bias, and therefore provides a guarantee on the target domains risk (i.e., it provides an upper bound on the optimal target domains risk, which can be estimated from the source domains data).

Rather than characterizing by hand all possible situations in which a separating set can be identified (like in Example 2), in this work we delegate the causal inference to an automatic theorem prover. Intuitively, the idea is to provide the automatic theorem prover with the (conditional) (in)dependencies that hold in the data, in combination with an encoding of Assumptions (1) and (2) into logical rules, and ask the theorem prover whether it can prove that \( C_1 \perp Y | A \) holds for a candidate set \( A \) from the assumptions and provided conditional (in)dependencies. There are three possibilities: either it can prove the query (and then we can proceed to predict \( Y \) from \( A \) and get an estimate of the target domains risk), or it can disprove the query (and then we know \( A \) will generically give predictions that suffer from an arbitrarily large transfer bias), or it can do neither (in which case hopefully another subset \( A \) can be found that does provably satisfy (2)).

### 2.5 Algorithm

A simple (brute-force) algorithm that finds the best separating set as described in Section 2.3 is the following. By using a standard feature selection method, produce a ranked list of subsets \( A \subseteq V \setminus \{Y, C_1\} \), ordered ascendingly with respect to the empirical source domains risks. Going through this list of subsets (starting with the one with the smallest empirical source domains risk),

---

7 Another option, proposed by Rojas-Carulla et al. (2018), is to assume that if \( p(Y \mid A) \) is invariant across all source domains (i.e., \( p(Y \mid A, C_1 = 0, C_1 = c) = p(Y \mid A, C_1 = 0) \) for all \( c \)), then the same holds across all source and target domains (i.e., \( p(Y \mid A, C_1 = 1) = p(Y \mid A, C_1 = 0, C_1 = c) \) for all \( c \)). This assumption can be violated in some simple cases, e.g. see Example 2.
test whether the separating set property can be inferred from the data by querying the automated theorem prover. If (2) can be shown to hold, use that subset $A$ for prediction of $Y$ and stop; if not, continue with the next candidate subset $A$ in the list. If no subset satisfies (2), abstain from making a prediction\textsuperscript{8}.

An important consequence of Assumption\textsuperscript{2(ii)} is that it enables us to transfer conditional independence involving the target variable from the source domains to the target domains (proof provided in the Supplementary Material):

**Proposition 1.** Under Assumption\textsuperscript{2}

$$A \perp B \mid S \ [C_1 = 0] \iff A \perp B \mid S \cup \{C_1\}$$

for subsets $A, B, S \subseteq V$ such that their union contains $Y$ but not $C_1$.

To test the separating set condition (2), we use the approach proposed by Hyttinen et al. (2014), where we simply add the JCI assumptions (Assumption\textsuperscript{1}) as constraints on the optimization problem, in addition to the domain-adaptation specific assumption that $C_1 \rightarrow Y \notin G$ (Assumption\textsuperscript{2(iii)}). As inputs we use all directly testable conditional independence test p-values $P_{A \perp B \mid S}$ in the pooled data (when $Y \notin A \cup B \cup S$) and all those resulting from Proposition\textsuperscript{1} from the source domains data only (if $Y \in A \cup B \cup S$). If background knowledge on intervention targets or the causal graph is available, it can easily be added as well. We use the method proposed by Magliacane et al. (2016) to query for the confidence of whether some statement (e.g., $Y \perp C_1 \mid A$) is true or false. The results of Magliacane et al. (2016) show that this approach is sound under oracle inputs, and asymptotically consistent whenever the statistical conditional independence tests used are asymptotically consistent. In other words, in this way the probability of wrongly deciding whether a subset $A$ is a separating set converges to zero as the sample size increases. We chose this approach because it is simple to implement on top of existing open source code\textsuperscript{9}.\footnote{Note that the computational cost quickly increases with the number of variables, limiting the number of variables that can be considered simultaneously.}

One remaining issue is how to predict $Y$ when an optimal separating set $A$ has been found. As the distribution of $A$ may shift when transferring from source domains to target domains, this means that there is a covariate shift to be taken into account when predicting $Y$. Any method (e.g., least-squares regression) could in principle be used to predict $Y$ from a given set of covariates, but it is advisable to use a prediction method that works well under covariate shift, e.g., Sugiyama et al. (2008).

### 3 Evaluation

We perform an evaluation on both synthetic data and a real-world dataset based on a causal inference challenge\textsuperscript{10}. The latter dataset consists of hematology-related measurements from the International Mouse Phenotyping Consortium (IMPC), which collects measurements of phenotypes of mice with different single-gene knockouts.

In both evaluations we compare a standard feature selection method (which uses Random Forests) with our method that builds on top of it and selects from its output the best separating set. First, we score all possible subsets of features by their out-of-bag score using the implementation of Random Forest Regressor from scikit-learn (Pedregosa et al., 2011) with default parameters. For the baseline we then select the best performing subset and predict $Y$. Instead, for our proposed method we try to find a subset of features $A$ that is also a separating set, starting from the subsets with the best scores. To test whether $A$ is a separating set, we use the method described in Section 2.5 using the ASP solver clingo 4.5.4 (Gebser et al., 2014). We provide as inputs the independence test results from a partial correlation test with significance level $\alpha = 0.05$ and combine it with the weighting scheme from Magliacane et al. (2016). We then use the first subset $A$ in the ranked list of predictive sets of features found by the Random Forest method for which the confidence that $C_1 \perp Y \mid A$ holds is positive. If there is no set $A$ that satisfies this criterion, then we abstain from making a prediction.

\textsuperscript{8}Abstaining from predictions can be advantageous when trading off recall and precision. If a prediction has to be made, we can fall back on some other method or simply accept the risk that the transfer bias may be large.

\textsuperscript{9}We build on the source code provided by Magliacane et al. (2016) which in turn extends the source code provided by Hyttinen et al. (2014). The full source code of our implementation and the experiments is available online at https://github.com/caus-am/dom_adapt.

\textsuperscript{10}Part of the CRM workshop on Statistical Causal Inference and Applications to Genetics, Montreal, Canada (2016). See also http://www.crm.umontreal.ca/2016/Genetics16/competition_e.php.
For the synthetic data, we generate randomly 200 linear acyclic models with latent variables and Gaussian noise, each with three system variables, and sample $N$ data points each for the observational and two experimental domains, where we simulate soft interventions on randomly selected targets, with different sizes of perturbations. We randomly select which of the two context variables will be $C_1$ and which of the three system variables will be $Y$. We disallow direct effects of $C_1$ on $Y$, and enforce that no intervention can directly affect all variables simultaneously. More details on how the data were simulated are provided in the Supplementary Material. Figure 3a shows a boxplot of the $L_2$ loss of the predicted $Y$ values with respect to the true values for both the baseline and our method, considering the 121 cases out of 200 in which our method does produce an answer. In particular, Figure 3a considers the case of $N = 1000$ samples per regime and interventions that all produce a large perturbation. In the Supplementary Material we show that results improve with more samples, both for the baseline, but even more so for our method, since the quality of the conditional independence tests improves. We also show that, according to expectations, if the target distribution is very similar to the source distributions, i.e., the transfer bias is small, our method does not provide any benefit and seems to perform worse than the baseline. Conversely, the larger the intervention effect, the bigger the advantage of using our method.

For the real-world dataset, we select a subset of the variables considered in the CRM Causal Inference Challenge. Specifically, for simplicity we focus on 16 phenotypes that are not deterministically related to each other. The dataset contains measurements for 441 “wild type” mice and for about 10 “mutant” mice for each of 13 different single gene knockouts. We then generate 1000 datasets by randomly selecting subsets of 3 variables and 2 gene knockout contexts, and always include also “wild type” mice. For each dataset we randomly choose $Y$ and $C_1$, and leave out the observed values of $Y$ for $C_1 = 1$. Figure 3b shows a boxplot of the $L_2$ loss of the predicted $Y$ values with respect to the real values for the baseline and our method. Given the small size of the datasets, this is a very challenging problem. In this case, our method abstains from making a prediction for 170 cases out of 1000 but performs similarly to the baseline on the remaining cases.

4 Discussion and Conclusion

We have defined a general class of causal domain adaptation problems and proposed a method that can identify sets of features that lead to transferable predictions. Our assumptions are quite general and in particular do not require the causal graph or the intervention targets to be known. The method gives promising results on simulated data. It is straightforward to extend our method to the cyclic case by making use of the results by Forré and Mooij (2018). More work remains to be done on the implementation side, for example, scaling up to more variables. Currently, our approach can handle about seven variables on a laptop computer, and with recent advances in exact causal discovery algorithms (e.g., Rantanen et al., 2018), a few more variables would be feasible. For scaling up to dozens of variables, we plan to adapt constraint-based causal discovery algorithms like FCI (Spirtes et al., 2000) to deal with the missing-data aspect of the domain adaptation task. We hope that this work will also inspire further research on the interplay between bias, variance and causality from a statistical learning theory perspective.
Acknowledgments

We thank Patrick Forré for proofreading a draft of this work. We thank Renée van Amerongen and Lucas van Eijk for sharing their domain knowledge about the hematology-related measurements from the International Mouse Phenotyping Consortium (IMPC). SM, TC, SB, and PV were supported by NWO, the Netherlands Organization for Scientific Research (VIDI grant 639.072.410). SM was also supported by the Dutch programme COMMIT/ under the Data2Semantics project. TC was also supported by NWO grant 612.001.202 (MoCoCaDi), and EU-FP7 grant agreement n.603016 (MATRICS). TvO and JMM were supported by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement 639466).

References


We prove that Assumption 2(ii) is a weakened version of two more standard assumptions, i.e., the causal Markov and faithfulness assumptions in both source and target domains separately. Note that assuming these two assumptions instead of Assumption 2(ii) implies we cannot have perfect interventions in the target domain, which is otherwise allowed.

**Proposition 2.** Assumption 2(ii) is implied by the following assumption:

(a) the pooled source domains distribution \( P(V | C_1 = 0) \) is Markov and faithful to \( G^{(C_1)} \), and

(b) the pooled target domains distribution \( P(V | C_1 = 1) \) is Markov and faithful to \( G^{(C_1)} \).

where \( G^{(C_1)} \) denotes the induced subgraph of the causal graph \( G \) on the nodes \( V \setminus \{C_1\} \) (i.e., it is obtained by removing \( C_1 \) and all edges involving \( C_1 \) from the causal graph \( G \)).

**Proof.** Let \( A, B, S \subseteq V \setminus \{C_1\} \). By assumption, we have that

\[
A \perp B | S [C_1 = c] \iff A \perp B | S [G^{(C_1)}]
\]

holds for both \( c = 0, 1 \), which directly gives Assumption 2(ii). \( \square \)

### A.2 Other proofs

**Proof of Example 2** In the JCI setting, we assume that in the full ADMG \( G \) over variables \( \{C_1, C_2, X_1, X_2, X_3\} \), \( C_1 \) and \( C_2 \) are confounded and not caused by system variables \( X_1, X_2, X_3 \). Furthermore, no pair of system variable and context variables is confounded.

In the context \( [C_1 = 0] \), if the conditional independences \( C_2 \perp X_2 | X_1 [C_1 = 0] \) and \( C_2 \not\perp X_2 | \emptyset [C_1 = 0] \) hold, then we can also derive that \( C_2 \not\perp X_1 | \emptyset [C_1 = 0] \), for example using Rule (9) from Magliacane et al. (2016). Moreover, we know that \( C_2 \) is not caused by \( X_1 \) and \( X_2 \), or in other words \( X_1 \not\rightarrow C_2 \) and \( X_2 \not\rightarrow C_2 \). Thus we conclude that \( \{C_2, X_1, X_2\} \) is an LCD triple (Cooper 1997) in the context \( C_1 = 0 \). Since in addition, in this case \( C_2 \) and \( X_1 \) are unconfounded, the marginal ADMG \( G'' \) on \( \{C_2, X_1, X_2\} \) (in the context \( C_1 = 0 \), and hence by Proposition 1 in all contexts) must be given by Figure 4c.

Therefore, the extended marginal ADMG \( G'' \) on variables \( \{C_1, C_2, X_1, X_2\} \) must also have a directed path from \( C_2 \) to \( X_1 \) and from \( X_1 \) to \( X_2 \). \( C_1 \) cannot be on these paths, as none of the variables causes \( C_1 \), and therefore \( G'' \) also contains the directed edges \( C_2 \rightarrow X_1 \) and \( X_1 \rightarrow X_2 \). Moreover, \( G'' \) cannot contain any edge between \( C_2 \) and \( X_2 \), nor a bidirected edge between \( X_1 \) and \( X_2 \), because that would...
We sample more information and experimental results for the synthetic data. We adapted the simulator of Hyttinen et al. (2014) to our setting. We generate randomly 200 acyclic models with three system variables, two context variables, and at most two latent variables (chosen randomly, so that the number of latent variables equals 1 or 2 each with probability 1/4, and 0 otherwise). Each latent variable has two system variables as children, while the other variables have a random number of system variables as children, where system variables must be consistent with a chosen topological ordering, and where we enforce that a context variable may not simultaneously affect all system variables. The system and latent variables are each described by a linear structural equation with independent noise terms distributed as $\mathcal{N}(0, 0.0064)$. In these equations, each variable is multiplied by a coefficient sampled from $\mathcal{N}(0, 0.64)$ or $\mathcal{N}(-0.2, 0.64)$ (each with probability 1/2 per variable). The context variables each correspond to an experimental domain; in their domain, that variable equals 1, otherwise it equals 0. This way, we simulate soft interventions. In order to scale the effect of these interventions, we multiply the coefficients of the context variables by the parameter $\gamma$, varying it from 0.1 to 100. We sample $N$ data points each for the observational and two experimental domains. Moreover, we randomly select $C_1$ and $Y$ from context and system variables respectively. We disallow direct effects of $C_1$ on $Y$.

As expected, our method performs well when the target distribution is significantly different from the source distributions. Figure 5 shows different settings with different scales of intervention effects. (In most graphs, the vertical axis has been adjusted to clearly show the boxplot, but leaving out the larger outliers.) In Figure 5 the intervention effects are all scaled by 0.1, resulting in very similar distributions in all domains. In this case, using our method does not offer any advantage with respect to the baseline and it actually performs worse. In the other cases, using our method starts to pay off in terms of prediction accuracy, and the difference increases with the scale of the interventions, as seen in Figure 5d.

A.3 Additional results on synthetic data

We provide more information and experimental results for the synthetic data. We adapted the simulator of Hyttinen et al. (2014) to our setting. We generate randomly 200 acyclic models with three system variables, two context variables, and at most two latent variables (chosen randomly, so that the number of latent variables equals 1 or 2 each with probability 1/4, and 0 otherwise). Each latent variable has two system variables as children, while the other variables have a random number of system variables as children, where system variables must be consistent with a chosen topological ordering, and where we enforce that a context variable may not simultaneously affect all system variables. The system and latent variables are each described by a linear structural equation with independent noise terms distributed as $\mathcal{N}(0, 0.0064)$. In these equations, each variable is multiplied by a coefficient sampled from $\mathcal{N}(0, 0.64)$ or $\mathcal{N}(-0.2, 0.64)$ (each with probability 1/2 per variable). The context variables each correspond to an experimental domain; in their domain, that variable equals 1, otherwise it equals 0. This way, we simulate soft interventions. In order to scale the effect of these interventions, we multiply the coefficients of the context variables by the parameter $\gamma$, varying it from 0.1 to 100. We sample $N$ data points each for the observational and two experimental domains. Moreover, we randomly select $C_1$ and $Y$ from context and system variables respectively. We disallow direct effects of $C_1$ on $Y$.

As expected, our method performs well when the target distribution is significantly different from the source distributions. Figure 5 shows different settings with different scales of intervention effects. (In most graphs, the vertical axis has been adjusted to clearly show the boxplot, but leaving out the larger outliers.) In Figure 5 the intervention effects are all scaled by 0.1, resulting in very similar distributions in all domains. In this case, using our method does not offer any advantage with respect to the baseline and it actually performs worse. In the other cases, using our method starts to pay off in terms of prediction accuracy, and the difference increases with the scale of the interventions, as seen in Figure 5d.
### Feature Selection

<table>
<thead>
<tr>
<th>Method</th>
<th>L2 loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature selection</td>
<td>Our method</td>
</tr>
</tbody>
</table>

#### Method

- **L2 loss**

#### Synthetic Data

1. **Synthetic data with a small perturbation** ($\gamma = 0.1$) and $N = 1000$ samples.
2. **Synthetic data with a medium perturbation** ($\gamma = 1$) and $N = 1000$ samples.
3. **Synthetic data with a large perturbation** ($\gamma = 10$) and $N = 1000$ samples.
4. **Synthetic data with a very large perturbation** ($\gamma = 100$) and $N = 1000$ samples.

#### Figure 5: Additional results when varying the causal effect of all interventions ($\gamma$).

#### Sample Size

1. **Synthetic data with** $N = 100$ samples per regime and a large perturbation ($\gamma = 10$).
2. **Synthetic data with** $N = 1000$ samples per regime and a large perturbation ($\gamma = 10$).
3. **Synthetic data with** $N = 5000$ samples per regime and a large perturbation ($\gamma = 10$).

#### Figure 6: Additional results when varying the sample size per regime ($N$).

In Figure 6, we vary the number of samples $N$ for each regime. The results improve with more samples, especially for our method, since the quality of the conditional independence test improves, but also for the baseline. In particular, as shown in Figure 6b, the accuracy is low for $N = 100$ samples, but it improves substantially with $N = 1000$ samples (Figure 5b).