5

Connections to Machine Learning, I

As argued in Chapter 1, standard machine learning rests on the same basis as statistics: we use data sampled i.i.d. from some unknown underlying distribution, and seek to infer properties of that distribution. In contrast, causal inference assumes a stronger underlying structure, including directed dependences. This makes it harder to learn about the structure from data, but it also allows novel statements once this is done, including statements about the effect of distribution shifts and interventions. If we view machine learning as the process of inferring regularities (or "laws of nature") that go beyond pure statistical associations, then causality plays a crucial role. The present chapter presents some thoughts on this, focusing on the case of two variables only. Chapter 8 will revisit this topic and look at the multivariate case.

5.1 Semi-Supervised Learning

Let us consider a regression task, in which our goal is to predict a target variable Y from a d-dimensional predictor variable $\mathbf{X}$. For many loss functions, knowing the conditional distribution $P_{Y|\mathbf{X}}$ suffices to solve the problem. For instance, the regression function

$$f^0(\mathbf{x}) := \mathbb{E}[Y \,|\, \mathbf{X} = \mathbf{x}]$$

minimizes the L_2 loss,

$$f^0 \in \operatorname*{argmin}_{f:\mathbb{R}^d \to \mathbb{R}} \mathbb{E}\left[(Y - f(\mathbf{X}))^2\right].$$

In **supervised learning** , we receive n i.i.d. data points from the joint distribution: $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n) \overset{\text{iid}}{\sim} P_{\mathbf{X},Y}$. Regression estimation (with L_2 loss) thus amounts to estimating the conditional mean from n data points of the joint distribution. In (inductive) **semi-supervised learning** (SSL), however, we receive m additional unlabeled data points $\mathbf{X}_{n+1}, \ldots, \mathbf{X}_{n+m} \overset{\text{iid}}{\sim} P_{\mathbf{X}}$. The hope is that these additional data points provide information about $P_{\mathbf{X}}$, which itself tells us something about $\mathbb{E}[Y|\mathbf{X}]$ or more generally about $P_{Y|\mathbf{X}}$.[1] Many assumptions underlying SSL techniques [see Chapelle et al., 2006, for an overview] concern relations between $P_{\mathbf{X}}$ and $P_{Y|\mathbf{X}}$. The *cluster assumption*, for instance, stipulates that points lying in the same cluster of $P_{\mathbf{X}}$ have the same or a similar Y; this is similar to the *low-density separation* assumption that states that the decision boundary of a classifier (i.e., points $\mathbf{x}$ where $P(Y = 1|\mathbf{X} = \mathbf{x})$ crosses 0.5) should lie in a region where $P_{\mathbf{X}}$ is small. The *semi-supervised smoothness* assumption says that the conditional mean $\mathbf{x} \mapsto \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$ should be smooth in areas where $P_{\mathbf{X}}$ is large.

5.1.1 SSL and Causal Direction

In the simplest setting, where the causal graph has only two variables (cause and effect), a machine learning problem can either be **causal** (if we predict effect from cause) or **anticausal** (if we predict cause from effect). Practitioners usually do not care about the causal structure underlying a given learning problem (see Figure 5.1). However, as we argue herein, the structure has implications for machine learning.

In Section 2.1, we have hypothesized that causal conditionals are *independent* of each other (Principle 2.1 and subsequent discussion). Schölkopf et al. [2012] realize that this principle has a direct implication for SSL. Since the latter relies on the relation between $P_{\mathbf{X}}$ and $P_{Y|\mathbf{X}}$ and the principle claims that P_{cause} and $P_{\text{effect}|\text{cause}}$ do not contain information about one another, we can conclude that SSL will not work if $\mathbf{X}$ corresponds to the cause and Y corresponds to the effect (i.e., for a *causal* learning problem). In this case, additional $\mathbf{x}$-values only tell us more about $P_{\mathbf{X}}$ — which is irrelevant because the prediction requires information about the independent object $P_{Y|\mathbf{X}}$. On the other hand, if $\mathbf{X}$ is the effect and $\mathbf{Y}$ is the cause, information on $P_{\mathbf{X}}$ may tell us something about $P_{Y|\mathbf{X}}$.

A meta-study that analyzed results in SSL supports our hypothesis. All cases

[1] Again, we use the notation $P_{Y|\mathbf{X}}$ as a shorthand for the collection $(P_{Y|\mathbf{X}=\mathbf{x}})_{\mathbf{x}}$ of conditional distributions.

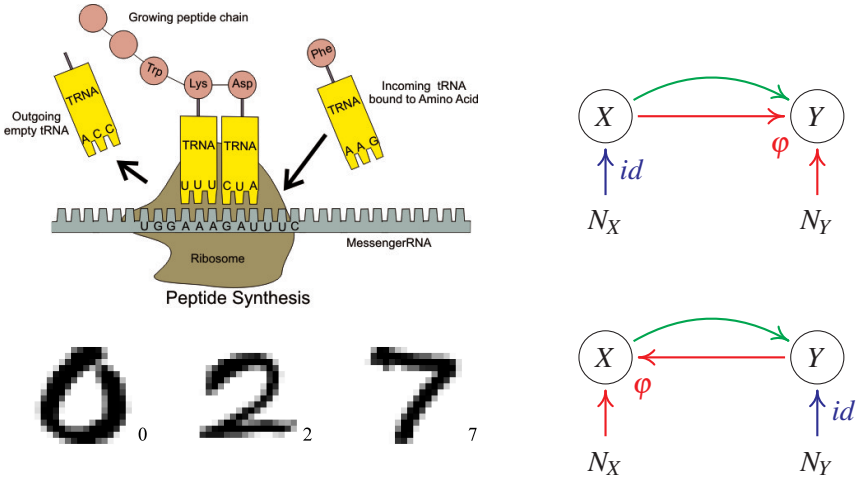

Figure 5.1: *Top:* a complicated mechanism φ called the ribosome translates mRNA information X into a protein chain Y.[2] Predicting the protein from the mRNA is an example of a *causal* learning problem, where the direction of prediction (green arrow) is aligned with the direction of causation (red). *Bottom:* In handwritten digit recognition, we try to infer the class label Y (i.e., the writer's intention) from an image X produced by a writer. This is an *anticausal* problem.

where SSL helped were anticausal, or confounded, or examples where the causal structure was unclear (see Figure 5.2).

Within the toy scenario of a bijective deterministic causal relation (see Section 4.1.7), Janzing and Schölkopf [2015] prove that whenever P_{cause} and $P_{\text{effect}|\text{cause}}$ are independent in the sense of (4.10), then SSL indeed outperforms supervised learning in the anticausal direction but not in the causal direction. The idea is that SSL employs the dependence (4.11) for an improved interpolation algorithm.

Sgouritsa et al. [2015] have developed a causal learning method that exploits the fact that SSL can only work in the anti-causal direction.

Finally, note that SSL contains some versions of unsupervised learning as a special case (with no labeled data). In clustering, for example, Y is often a discrete value indicating the cluster index. Similarly to the preceding reasoning, we can argue that if X is the cause and Y the effect, clustering should not work well. In

[2] By user "Boumphreyfr", `https://commons.wikimedia.org/wiki/File:Peptide_syn.png`, [CC BY-SA 3.0 (`http://creativecommons.org/licenses/by-sa/3.0`) or GFDL (`http://www.gnu.org/copyleft/fdl.html`)]

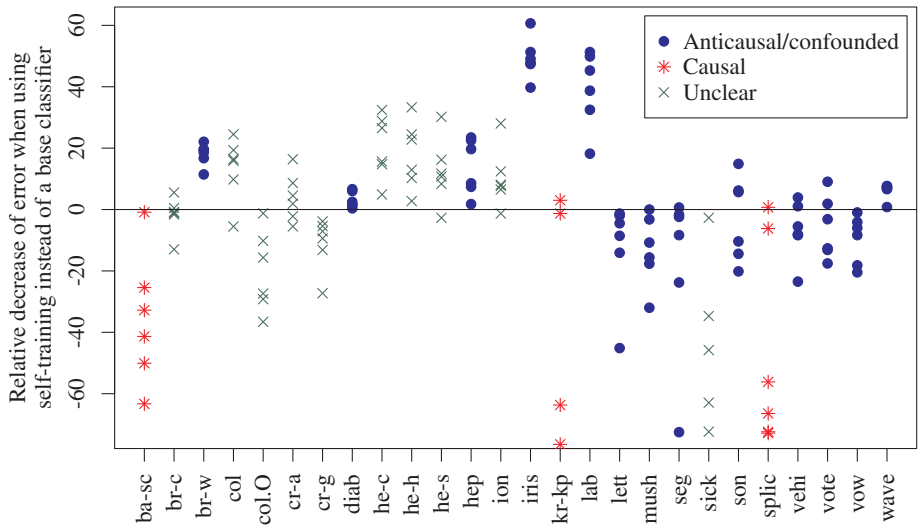

Figure 5.2: The benefit of SSL depends on the causal structure. Each column of points corresponds to a benchmark data set from the UCI repository and shows the performance of six different base classifiers augmented with self-training, a generic method for SSL. Performance is measured by percentage decrease of error relative to the base classifier, that is, $(\text{error(base)} - \text{error(self-train)})/\text{error(base)}$. Self-training overall does not help for the causal data sets, but it does help for some of the anticausal/confounded data sets [from Schölkopf et al., 2012].

many applications of clustering on real data, however, the cluster index is rather the cause than the effect of the features.

While the empirical results in Figure 5.2 are promising, the statement that SSL does not work in the causal direction (always assuming independence of cause and mechanism, cf. Principle 2.1) needs to be made more precise. This will be done in the following section; it may be of interest to readers interested in SSL and covariate shift, but could be skipped at first reading by others.

5.1.2 A Remark on SSL in the Causal Direction

A more precise form of our prediction regarding SSL reads as follows: if the task is to predict y for some specific x, knowledge of P_X does not help when $X \to Y$ is the causal direction. However, even if P_X does not tell us anything about $P_{Y|X}$ (due to $X \to Y$), knowing P_X can still help us for better estimating Y in the sense that we

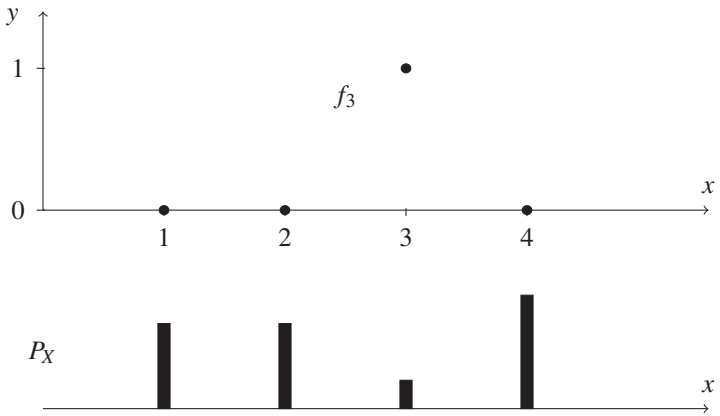

Figure 5.3: In this example, SSL reduces the loss even in the causal direction. Since for every x, the label zero is a priori more likely than the label one, the expected number of errors is minimized when a function is chosen that attains one at a point x where $p(x)$ is minimal (here: $x = 3$).

obtain lower risk in a learning scenario.

To see this, consider a toy example where the relation between X and Y is given by a deterministic function, that is, $Y = f(X)$, where f is known to be from some class $\mathcal{F}$ of functions. Let X take values in $\{1,\dots,m\}$ with $m \geq 3$ and let Y be a binary label attaining values in $\{0,1\}$. We define the function class $\mathcal{F} := \{f_1,\dots,f_m\}$ by $f_j(j) = 1$ and $f_j(k) = 0$ for $k \neq j$. In other words, $\mathcal{F}$ consists of the set of functions that attain the value one at exactly one point. Figure 5.3, top, shows the function f_3 for $m = 4$. Suppose that our learning algorithm infers f_j while the true function is f_i. For $i \neq j$, the risk, that is, the expected number of errors (see Equation (1.2)), equals

$$R_i(f_j) := \sum_{x=1}^{m} |f_j(x) - f_i(x)| p(x) = p(j) + p(i), \tag{5.1}$$

where p denotes the probability mass function for X. We now average $R_i(f_j)$ over the set $\mathcal{F}$ and assume that each f_i is equally likely. This yields the expected risk (where the expectation is taken with respect to a uniform prior over $\mathcal{F}$)

$$\mathbb{E}[R_i(f_j)] = \frac{1}{m} \sum_{i=1}^{m} \sum_{x=1}^{m} |f_j(x) - f_i(x)| p(x) \tag{5.2}$$

$$= \frac{1}{m} \sum_{i \neq j} (p(j) + p(i)) = \frac{m-2}{m} p(j) + \frac{1}{m}. \tag{5.3}$$

To minimize (5.3), we should thus choose f_k such that k minimizes the function p. This makes sense because for any point $x = 1, \ldots, m$, the label $y = 0$ is more likely than $y = 1$ (probability $(m-1)/m$ versus $1/m$). Therefore, we would actually like to infer zero everywhere, but since the zero function is not contained in $\mathcal{F}$, we are forced to select one x-value to which we assign the label zero. Hence, we choose one of the least likely x-values to obtain minimal expected loss (which is $x = 3$ for the distribution in Figure 5.3, bottom). Clearly, unlabeled observations help identify the least likely x-values, hence SSL can help. This example does not require any (x,y)-pairs (labeled instances); unlabeled data x suffices. It is thus actually an example of *un*supervised learning rather than being a typical SSL scenario. However, accounting for a small number of labeled instances in addition does not change the essential idea. Generically, these few instances will not contain any instance with $y = 1$ if m is large enough. Hence, the observed (x,y)-pairs only help because they slightly reduce $\mathcal{F}$ to a smaller class $\mathcal{F}'$ for which the analysis remains basically the same, and we still conclude that the unlabeled instances help.

Although we have not specified a supervised learning scenario as baseline (that is, one that does not employ knowledge of P_X), we know that it must be worse than the best semi-supervised scenario because the optimal estimation depends on P_X, as we have just argued.

Here, the independence of mechanisms is not violated (and thus, X can be considered as a cause for Y): f is assumed to be chosen uniformly among $\mathcal{F}$, and knowing P_X does not tell us anything about f. Knowing P_X is only helpful for minimizing the loss because $p(x)$ appears in (5.2) as a weighting factor.

The preceding example is close in spirit to a Bayesian analysis because it involved an average over functions in $\mathcal{F}$. It can be modified, however, to apply to a worst case analysis, in which the true function f is chosen by an adversarial to maximize (5.1) [see also Kääriäinen, 2005]. Given a function f_j, the adversarial chooses f_i with i an x-value different from j with maximal probability mass. The worst case risk thus reads $\max_{x \neq j}\{p(x)\} + p(j)$, which is, again, minimized when j is chosen to be an x-value that minimizes the probability mass function $p(x)$. Therefore, we conclude that optimal performance is attained only when P_X is taken into account.

Another example can be constructed on the basis of an argument that is given in a non-causality context by Urner et al. [2011, proof of Theorem 4]. They construct a case of model misspecification; where the true function f_0 is not contained in the class $\mathcal{F}$ that is optimized over. In their example, additional information about the marginal P_X helps for reducing the risk, even though the conditional $P_{Y|X}$ can be considered as being independent of the marginal. Our example above is not based

on the same kind of model misspecification. Each possible (unknown) ground truth f_i is indeed contained in the class of functions; however, we would like to minimize the *expectation* of the risk over a prior, and our function class does not contain a function that has zero expected risk. Therefore, for the expected risk, this is akin to a situation of model misspecification.

Finally, we try to give some further intuition about the example by Urner et al. [2011]. Since f_0 is not contained in the function class $\mathcal{F}$, we need to find a function $\hat{f} \in \mathcal{F}$ that minimizes the distance $d(f, f_0)$, defined as the risk of f, over $f \in \mathcal{F}$; we say f_0 is projected onto $\mathcal{F}$. Roughly speaking, additional information about P_X provides us with a better understanding of this projection.[3]

5.2 Covariate Shift

As explained in Section 2.1, the independence between P_{cause} and $P_{\text{effect|cause}}$ (Principle 2.1) can be interpreted in two different ways: in Section 5.1 above, we argued that given a fixed joint distribution, these two objects contain no information about each other (see the middle box in Figure 2.2). Alternatively, suppose the joint distribution $P_{\text{cause,effect}}$ changes across different data sets; then the change of P_{cause} does not tell us anything about the change of $P_{\text{effect|cause}}$ (this corresponds to the left box in Figure 2.2). Knowing that X is the cause and Y the effect thus has important consequences for a prediction scenario where Y is predicted from X. Assume we have learned the statistical relation between X and Y using examples from one data set and we are supposed to employ this knowledge for predicting Y from X for a second data set. Further, assume that we observe that the x-values in the second data set follow a distribution P'_X that differs from the distribution P_X of the first data set. How would we make use of this information? By the independence of mechanisms, the fact that P'_X differs from P_X does not tell us anything about whether $P_{Y|X}$ also changed across the data sets. Therefore, it might be the case that the conditional $P_{Y|X}$ still holds true for the second data set. Second, even if the conditional did change to $P'_{Y|X} \neq P_{Y|X}$, it is natural to still use $P_{Y|X}$ for our prediction. After all, the independence principle states that the new change of the marginal distribution from P_X to P'_X does not tell us anything about *how* the conditional has changed. Therefore, we use $P_{Y|X}$ in absence of any better candidate. Using the same conditional $P_{Y|X}$ although P_X has changed is usually referred to as

[3] We are grateful to several people who contributed to this discussion: Sebastian Nowozin, Ilya Tolstikhin, and Ruth Urner.

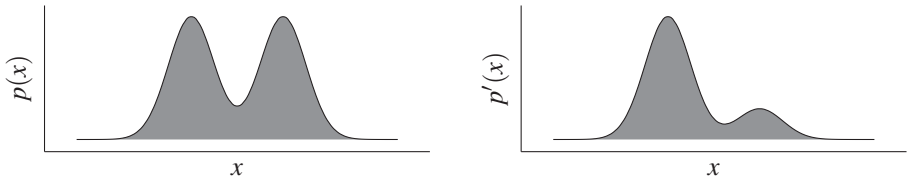

Figure 5.4: Example where P_X changes to P'_X in a way that suggests that P_Y has changed and $P_{X|Y}$ remained the same. When Y is binary and known to be the cause of X, observing that P_X is a mixture of two Gaussians makes it plausible that the two modes correspond to the two different labels $y = 0, 1$. Then, the influence of Y on X consists just in shifting the mean of the Gaussian (which amounts to an ANM — see Section 4.1.4), which is certainly a simple explanation for the joint distribution. Observing furthermore that the weights of the mixture changed from one data set to another one makes it likely that this change is due to the change of P_Y.

covariate shift. Meanwhile, this is a well-studied assumption in machine learning [Sugiyama and Kawanabe, 2012]. The argument that this is only justified in the *causal* scenario, in other words, if X is the cause and Y the effect, has been made by Schölkopf et al. [2012].

To further illustrate this point, consider the following toy example of an *anti-causal* scenario where X is the effect. Let Y be a binary variable influencing the real-valued variable X in an additive way:

$$X = Y + N_X, \tag{5.4}$$

where we assume N_X to be Gaussian noise, independent of Y. Figure 5.4, left, shows the corresponding probability density p_X.

If its width is sufficiently small, the distribution P_X is bimodal. Even if one does not know anything about the generating model, P_X can be recognized as a mixture of two Gaussian distributions with equal width. In this case, one can therefore guess the joint distribution $P_{X,Y}$ from P_X alone because it is natural to assume that the influence of Y consists only in shifting the mean of X. Under this assumption, we do not need any (x, y)-pairs to learn the relation between X and Y. Assume now that in a second data set we observe the same mixture of two Gaussian distributions but with different weights (see Figure 5.4, right). Then, the most natural conclusion reads that the weights have changed because the same equation (5.4) still holds but only P_Y has changed. Accordingly, we would no longer use the same $P_{Y|X}$ for our prediction and reconstruct $P'_{Y|X}$ from P'_X. The example illustrates that in the anticausal scenario the changes of P_X and $P_{Y|X}$ may be related and that this relation may be due to the fact that P_Y has changed and $P_{X|Y}$ remained the same. In other

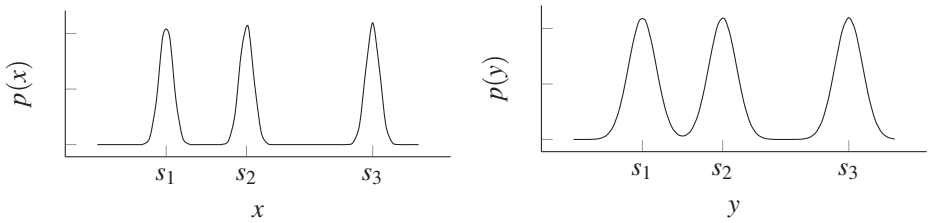

Figure 5.5: Example where X causes Y and, as a result, P_Y and $P_{X|Y}$ contain information about each other. Left: P_X is a mixture of sharp peaks at the positions s_1, s_2, s_3. Right: P_Y is obtained from P_X by convolution with Gaussian noise with zero mean and thus consists of less sharp peaks at the same positions s_1, s_2, s_3. Then $P_{X|Y}$ also contains information about s_1, s_2, s_3 (see Problem 5.1).

words, P_{effect} and $P_{\text{cause}|\text{effect}}$ often change in a dependent way because P_{cause} and $P_{\text{effect}|\text{cause}}$ change independently.

The previous example elicits a specific scenario. Conceiving of general methods exploiting the fact that P_{effect} and $P_{\text{cause}|\text{effect}}$ change in a dependent way is a hard problem. This may be an interesting avenue for further research, and we believe that causality could play a major role in domain adaptation and transfer problems; see also Bareinboim and Pearl [2016], Rojas-Carulla et al. [2016], Zhang et al. [2013], and Zhang et al. [2015].

5.3 Problems

Problem 5.1 (Independence of mechanisms) *Let P_X be the mixture of k sharp Gaussian peaks at positions $s_1, \ldots, s_k$ as shown in Figure 5.5, left. Let Y be obtained from X by adding some Gaussian noise N with zero mean and a width σ_N such that the separate peaks remain visible as in Figure 5.5, right.*

a) Argue intuitively why $P_{X|Y}$ also contains information about the positions $s_1, \ldots, s_k$ of the peaks and thus $P_{X|Y}$ and P_Y share this information.

b) The transition between P_X and P_Y can be described by convolution (from P_X to P_Y) and deconvolution (from P_Y to P_X). If $P_{Y|X}$ is considered as the linear map converting the input P_X to the output P_Y then $P_{Y|X}$ coincides with the convolution map. Argue why $P_{X|Y}$ does not coincide with the deconvolution map (as one may think at first glance).

8

Connections to Machine Learning, II

As argued in Chapter 5, the causal structure that underlies a statistical model can have strong implications for machine learning tasks such as semi-supervised learning or domain adaptation. We now revisit this general topic, focusing on the multivariate case. We begin with a method that uses machine learning to model systematic errors for a given causal structure, followed by some thoughts on reinforcement learning (with an application in computational advertising), and finally we comment on the topic of domain adaptation.

8.1 Half-Sibling Regression

This method exploits a given causal structure (see Figure 8.1) to reduce systematic noise in a prediction task. The goal is to reconstruct the unobserved signal Q. Schölkopf et al. [2015] suggest that we can denoise the signal Y by removing all information that can be explained by other measurements X that have been corrupted with the same source of noise. Here, X are measurements of some signals R that are independent of Q. Intuitively, everything in Y that can be explained by X must be due to the systematic noise N and should therefore be removed. More precisely, we consider

$$\hat{Q} := Y - \mathbb{E}[Y \,|\, X]$$

as an estimate for Q. Here, $\mathbb{E}[Y \,|\, X]$ is the *regression* of Y on its *half-siblings* X (note that X and Y share the parent N; see Figure 8.1).

One can show that for any random variables Q, X, Y that satisfy $Q \perp\!\!\!\perp X$, we have

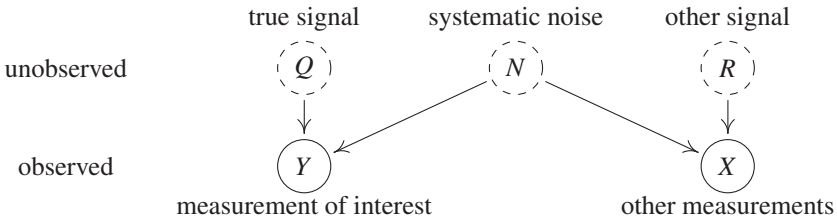

Figure 8.1: The causal structure that applies to the exoplanet search problem. The underlying signal of interest Q can only be measured as a noisy version Y. If the same noise source also corrupts measurements of other signals that are independent of Q, those measurements can be used for denoising. In our example, the telescope N constitutes systematic noise that affects measurements X and Y of independent light curves.

[Schölkopf et al., 2016, Proposition 1]:

$$\mathbb{E}\Big[\big(Q - E[Q] - \hat{Q}\big)^2\Big] \;\leq\; \mathbb{E}\Big[\big(Q - E[Q] - (Y - E[Y])\big)^2\Big],$$

that is, the method is never worse than taking the measurement Y. If, moreover, the systematic noise acts in an additive manner, that is, $Y = Q + f(N)$ for some (unknown) function f, we have [Schölkopf et al., 2016, Proposition 3]:

$$\mathbb{E}\Big[\big(Q - E[Q] - \hat{Q}\big)^2\Big] = \mathbb{E}[\mathrm{var}[f(N)|X]]. \tag{8.1}$$

If the additive noise is a function of X, that is, $f(N) = \psi(X)$ for some (unknown) function ψ, then the right-hand side of (8.1) vanishes and hence $\hat{Q}$ recovers Q up to an additive shift; see Schölkopf et al. [2016] for other sufficient conditions.

As an example, consider the search for exoplanets. The Kepler space observatory, launched in 2009, observed a small fraction of the Milky Way during its search for exoplanets, monitoring the brightness of approximately 150,000 stars.[1] Those stars that are surrounded by a planet with a suitable orbit to allow for partial occlusions of the star will exhibit light curves that show a periodic decrease of light intensity; see Figure 8.2. These measurements are corrupted with systematic noise that is due to the telescope and that makes the signal from possible planets hard to detect.

Fortunately, the telescope measures many stars at the same time. These stars can be assumed to be causally and therefore statistically independent since they are light-years apart from each other. Thus, the causal structure depicted in Figure 8.1 fits very well to this problem and we may apply the half-sibling regression. This simple method performs surprisingly well [Schölkopf et al., 2015].

[1] `https://en.wikipedia.org/wiki/Kepler_(spacecraft)`, accessed 13.07.2016.

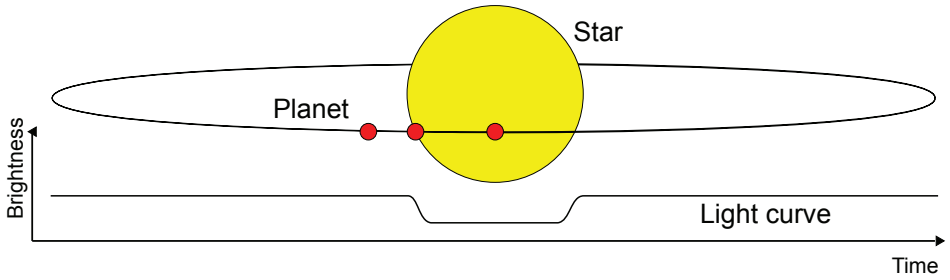

Figure 8.2: Every time a planet occludes a part of the star, the light intensity decreases. If the planet orbits the star, this phenomenon occurs periodically. (Image courtesy of Nikola Smolenski, `https://en.wikipedia.org/wiki/File:Planetary_transit.svg`, [CC BY-SA 3.0]. Image has been edited for clarity and style.)

Related approaches have been used in other application fields without reference to causal modeling [Gagnon-Bartsch and Speed, 2012, Jacob et al., 2016]. Considering the causal structure of the problem (Figure 8.1) immediately suggests the proposed methodology and leads to theoretical arguments justifying the approach.

8.2 Causal Inference and Episodic Reinforcement Learning

We now describe a class of problems in reinforcement learning from a causal perspective. Roughly speaking, in reinforcement learning, an agent is embedded in a world and chooses among a set of different actions. Depending on the current state of the world, these actions yield some reward and change the state of the world. The goal of the agent is to maximize the expected cumulated reward (see Section 8.2.2 for more details). We first introduce the concept of inverse probability weighting that has been applied in different contexts throughout machine learning and statistics and then relate it to episodic reinforcement learning. Drawing this connection is a first small step toward relating causality and reinforcement learning. The causal point of view enables us to exploit conditional independences that directly follow from the causal structure. We briefly mention two applications — blackjack and the placement of advertisement — and show how they benefit from causal knowledge. The causal formulation leads to these improvements of methodology very naturally but it is certainly possible to formulate these problems and corresponding algorithms without causal language. This section does not prove that reinforcement learning benefits from causality. Instead, we regard it as a step

toward establishing a formal link between these two fields that may lead to fruitful research in future [see also Bareinboim et al., 2015, for example]. More concretely, we believe that causality could play a role when transferring knowledge between different tasks in reinforcement learning (e.g., when progressing to the next level in a computer game or when changing the opponent in table tennis); however, we are not aware of any such result.

8.2.1 Inverse Probability Weighting

Inverse probability weighting is a well-known technique that is used to estimate properties of a distribution from a sample that follows a different distribution. It therefore naturally relates to causal inference. Consider the kidney stone example (Example 6.37). We defined the binary variables size S, treatment T, and recovery R, and after obtaining observational data, we were interested in the expected recovery rate $\tilde{\mathbb{E}}[R]$ in a hypothetical study in which everyone received treatment A, that is under a different distribution. Formally, consider an SCM $\mathfrak{C}$ entailing the distribution $P_{\mathbf{X}}^{\mathfrak{C}}$ over variables $\mathbf{X} = (X_1, \ldots, X_d)$. We have argued that one often observes a sample from the observational distribution $P_{\mathbf{X}}^{\mathfrak{C}}$, but one is interested in some intervention distribution $P_{\mathbf{X}}^{\tilde{\mathfrak{C}}}$. Here, the new SCM $\tilde{\mathfrak{C}}$ is constructed from the original $\mathfrak{C}$ by intervening on a node X_k, say,

$$do\left(X_k := \tilde{f}(X_{\widetilde{\mathbf{PA}}_k, \tilde{N}_k})\right);$$

see Section 6.3. In particular, we might want to estimate a certain property

$$\tilde{\mathbb{E}}\, \ell(\mathbf{X}) := \mathbb{E}_{P_{\mathbf{X}}^{\tilde{\mathfrak{C}}}}\, \ell(\mathbf{X})$$

of the new distribution $P_{\mathbf{X}}^{\tilde{\mathfrak{C}}}$ (in the kidney stone example, this is $\tilde{\mathbb{E}}[R]$). If densities exist, we have seen in Section 6.3 that the densities of $\mathfrak{C}$ and $\tilde{\mathfrak{C}}$ factorize in a similar way:

$$\begin{aligned} p(x_1, \ldots, x_d) &:= p^{\mathfrak{C}}(x_1, \ldots, x_d) = \prod_{j=1}^{d} p^{\mathfrak{C}}\left(x_j \,|\, x_{pa(j)}\right) \quad \text{and} \\ \tilde{p}(x_1, \ldots, x_d) &:= p^{\tilde{\mathfrak{C}}}(x_1, \ldots, x_d) = \prod_{j \neq k} p^{\mathfrak{C}}\left(x_j \,|\, x_{pa(j)}\right)\ \tilde{p}\left(x_k \,|\, x_{\widetilde{pa}(k)}\right). \end{aligned}$$

The factorizations agree except for the term of the intervened variable. We therefore have

$$\xi := \tilde{\mathbb{E}}\,\ell(\mathbf{X}) = \int \ell(\mathbf{x})\;\tilde{p}(\mathbf{x})\;d\mathbf{x} = \int \ell(\mathbf{x})\;\frac{\tilde{p}(\mathbf{x})}{p(\mathbf{x})}\,p(\mathbf{x})\;d\mathbf{x}$$
$$= \int \ell(\mathbf{x})\;\frac{\tilde{p}\left(x_k \,|\, x_{\widetilde{pa}(k)}\right)}{p\left(x_k \,|\, x_{pa(k)}\right)}\,p(\mathbf{x})\;d\mathbf{x}.$$

(For simplicity, we assume throughout the whole section that the densities are strictly positive.) Given a sample $\mathbf{X}^1,\ldots,\mathbf{X}^n$ drawn from the distribution $P_{\mathbf{X}}^{\mathfrak{C}}$, we can thus construct an estimator

$$\hat{\xi}_n := \frac{1}{n}\sum_{i=1}^{n} \ell(\mathbf{X}^i)\;\frac{\tilde{p}\left(X_k^i \,|\, \mathbf{X}^i_{\widetilde{pa}(k)}\right)}{p\left(X_k^i \,|\, \mathbf{X}^i_{pa(k)}\right)} = \frac{1}{n}\sum_{i=1}^{n} \ell(\mathbf{X}^i)\,w_i \tag{8.2}$$

for $\xi = \tilde{\mathbb{E}}\,\ell(\mathbf{X})$ by *reweighting* the observations; here, the weights w_i are defined as the ratio of the conditional densities. The data points, that have a high likelihood under $P_{\mathbf{X}}^{\tilde{\mathfrak{C}}}$ (they "could have been drawn" from the new distribution of interest) receive a large weight and contribute more to the estimate $\hat{\xi}_n$ than those with a small weight. This kind of estimator appears in the following three situations, for example.

(i) Suppose that $\mathbf{X} = (Y,Z)$ contains only a target variable Y and a *causal* covariate Z, that is, $Z \to Y$. Let us consider an intervention in Z and the function $\ell(\mathbf{X}) = \ell((Z,Y)) = Y$. Then, the estimator (8.2) reduces to

$$\hat{\xi}_n := \frac{1}{n}\sum_{i=1}^{n} Y^i\;\frac{\tilde{p}(Z^i)}{p(Z^i)}, \tag{8.3}$$

which is known as the **Horvitz-Thompson estimator** [Horvitz and Thompson, 1952]. This setting corresponds to the assumption of covariate shift [e.g., Shimodaira, 2000, Quionero-Candela et al., 2009, Ben-David et al., 2010]; see also Sections 5.2 and 8.3. The estimator (8.3) is an example of a weighted likelihood estimator.

(ii) For $\mathbf{X} = Z$, we may estimate the expectation $\tilde{\mathbb{E}}\,[\ell(Z)]$ under $\tilde{p}$ using data sampled from p. Thus, Equation (8.2) reduces to

$$\hat{\xi}_n := \frac{1}{n}\sum_{i=1}^{n} \ell(Z^i)\;\frac{\tilde{p}(Z^i)}{p(Z^i)},$$

a formula that is known as **importance sampling** [e.g., MacKay, 2002, Chapter 29.2]. The formula can be adapted if p and $\tilde{p}$ are known only up to constants.

(iii) We will make use of Equation (8.2) in the context of episodic reinforcement learning. We describe this application in a bit more detail next.

8.2.2 Episodic Reinforcement Learning

Reinforcement learning [e.g Sutton and Barto, 2015] models the behavior of agents taking actions in a world. Depending on the current state S_t of the world and the action A_t, the state of the world changes according to a **Markov decision process**, for example [e.g., Bellman, 1957]; that is, the probability $P(S_{t+1} = s)$ of entering a new state s depends only on the current state S_t and action A_t. Furthermore, the agent will receive some reward R_{t+1} that depends on S_t, A_t, and S_{t+1}; the sum over all rewards is sometimes called the return, which we write as $Y := \sum_t R_t$. The way the return Y depends on states and action is unknown to the agent who tries to improve his **strategy** $(a,s) \mapsto \pi(a\,|\,s) := P(A_t = a\,|\,S_t = s)$, that is, the conditional of the action he chooses depending on the observational part of the state of the world. In **episodic reinforcement learning**, the state is reset after a finite number of actions (see Figure 8.3). In Section 8.2.3, we consider the example of

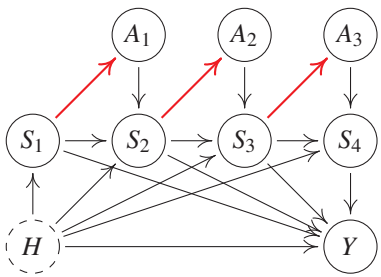

Figure 8.3: The graph describes an episodic reinforcement learning problem. The action variables A_i influence the system's next state S_{i+1}. The variable Y describes the output or return that we receive after one episode. This return Y may depend on the actions, too (edges omitted for clarity); it is often modelled as the (possibly weighted) sum of rewards that are received after each decision; see Section 8.2.3. The whole system can be confounded by an unobserved variable H. The bold, red edges indicate the conditionals that the player can influence, that is, the strategy. Equation (8.4) estimates the expected outcome $\tilde{\mathbb{E}}[Y]$ under a strategy $\tilde{\pi}$ from data obtained using strategy π. The equation still holds, when there are additional edges from the actions A to H and/or Y.

blackjack. In the example of Figure 8.3, the player makes $K = 3$ decisions, after which the cards are reshuffled. Then, a new episode starts.

Suppose that we play n games under a certain strategy $(a,s) \mapsto \pi(a\,|\,s)$, and each game is an episode. This function π does not depend on the number of "moves" we have played so far but just on the value of the state. As long as this strategy assigns a positive probability to any action, Equation (8.2) allows us to estimate the performance of a different strategy $(a,s) \mapsto \tilde{\pi}(a\,|\,s)$.

$$\hat{\xi}_{n,\mathrm{ERL}} := \frac{1}{n} \sum_{i=1}^{n} Y^i \, \frac{\prod_{j=1}^{K} \tilde{\pi}(A_j^i \,|\, S_j^i)}{\prod_{j=1}^{K} \pi(A_j^i \,|\, S_j^i)}. \tag{8.4}$$

This can be seen as a Monte Carlo method for off-policy evaluation [Sutton and Barto, 2015, Chapter 5.5]. In practice, the estimator (8.4) often has large variance; in continuous settings the variance may even be infinite. It has been suggested to reweight [Sutton and Barto, 2015] or to disregard the (five) largest weights [Bottou et al., 2013] to trade off variance for bias. Bottou et al. [2013] additionally compute confidence intervals and gradients in the case of parametrized densities. The latter are important if one wants to search for optimal strategies.

We now briefly discuss two examples, in which exploiting the causal structure leads to an improved *statistical* performance of the learning procedure. We regard them as interesting examples that shed some light on the relationship between reinforcement learning and causality.

8.2.3 State Simplification in Blackjack

The methodology proposed in Section 8.2.2 can be used to learn how to play blackjack (a card game). We pretend that a player enters a casino and starts playing blackjack knowing neither the objective of the game nor the optimal strategy; instead, he applies a random strategy. At each point in the game, the player is asked which of the legal actions he wants to take, and after the game has finished the dealer reveals how much money the player won or lost. After a while the player may update his strategy toward decisions that proved to be successful and continue playing. From a mathematical point of view, blackjack is solved. The optimal strategy (for infinitely many decks) was discovered by Baldwin et al. [1956] and leads to an expectation of $\mathbb{E}[Y] \approx -0.006$€ for a player betting 1€.

How does causality come into play? We have assumed that the player is unaware of the precise rules of blackjack; maybe he knows, however, that the win or loss is determined only by the values of the cards and not their suits; that is, the rules do not distinguish between a queen of clubs and a queen of hearts. The player can

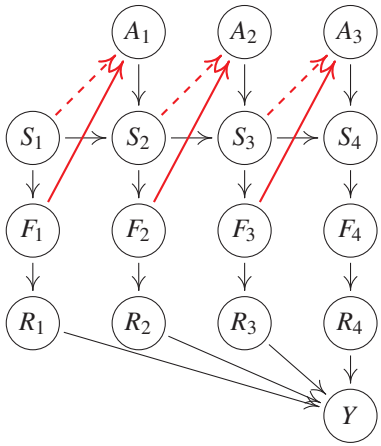

Figure 8.4: Here, there exist variables $F_1, \dots, F_4$ that contain all relevant information about the states $S_1, \dots, S_4$ in the sense that Equations (8.5) and (8.6) hold. Equation (8.6) is not represented in the graph. Then, it suffices if the actions A_j depend on F_{j-1} (red, solid lines) rather than S_{j-1} (red, dashed lines). In the blackjack example, the S_j's encode the dealer's hand and player's hand including suits, while the F_j encode the same information except for suits (suits do not have an influence on the outcome of blackjack). Since F_j take fewer values than S_j, the optimal strategy becomes easier to learn.

then immediately conclude that the optimal strategy does not depend on the suit. This comes with an obvious advantage when searching for the optimal strategy: the number of relevant state spaces and therefore the space of possible strategies reduces significantly. Figure 8.4 depicts this argument: the variables S_t contain all information, whereas the variables F_t do not contain suits. For example,

$$\begin{aligned} S_3 &= (\text{Player: } \heartsuit K, \spadesuit 5, \diamondsuit 4; \quad \text{Dealer: } \diamondsuit K) \\ F_3 &= (\text{Player: } \quad K, \quad 5, \quad 4; \quad \text{Dealer: } \quad K). \end{aligned}$$

Since the final result Y depends only on $(F_1, \dots, F_4)$ and not on the "full state" $(S_1, \dots, S_4)$, the actions may be chosen to depend on the F variables. Similarly, one may exploit that the order of the cards does not matter either. More formally, we have the following result:

Proposition 8.1 (State simplification) *Suppose that we are interested in the return $Y := \sum_j R_j$, and all variables are discrete. Assume that there is a function f such that for all j and for $F_j := f(S_j)$, we have*

$$R_j \perp\!\!\!\perp S_j \,|\, F_j, A_j, \tag{8.5}$$

and the full states do not matter for the change of states in the following sense: for all s_j and for all s_{j-1}, s°_{j-1} with $f(s_{j-1}) = f(s^\circ_{j-1})$

$$p(f(s_j)\,|\,s_{j-1}) = p(f(s_j)\,|\,s^\circ_{j-1}). \tag{8.6}$$

Then the optimal strategy $(a,s) \mapsto \pi_{opt}(a\,|\,s)$ depends only on F_j and not on S_j. There exists

$$\pi_{opt} \in \operatorname*{argmax}_{\pi} \mathbb{E}[Y],$$

such that

$$\pi_{opt}(a_j\,|\,s_{j-1}) = \pi_{opt}(a_j\,|\,s^\circ_{j-1}) \quad \forall s_{j-1}, s^\circ_{j-1} : f(s_{j-1}) = f(s^\circ_{j-1}).$$

This result is particularly helpful if F_j takes fewer values than S_j. The proof is provided in Appendix C.11. In the blackjack example, Equation (8.6) states that the probability of drawing another king depends only on the values of the cards drawn before (the number of kings in particular), not their suits.

8.2.4 Improved Weighting in Advertisement Placement

A related argument is used by Bottou et al. [2013] for the optimal placement of advertisements. Consider the following simplified description of the system. A company, which we will refer to as the publisher, runs a search engine and may want to display advertisements in the space above the search results, the mainline. Only if a user clicks on an ad does the publisher receive money from the corresponding company. Before displaying the ads, the publisher sets the mainline reserve A, a real-valued parameter that determines how many ads are shown in the mainline. In most systems, the number of mainline ads F varies between 0 and 4, that is, $F \in \{0,1,2,3,4\}$. The mainline reserve A usually depends on many variables (e.g., search query, date and time of the query, location), that we call the state S. If the search query indicates that the user intends to buy new shoes, for example, one may want to show more ads compared to when a user is looking for the time of the next service at church. We can model the system as episodic reinforcement learning with episodes of length 1.[2] The return Y equals the number of clicks per episode; its value is either 0 or 1. The question how to choose an optimal mainline reserve A then corresponds to finding the optimal strategy $(a,s) \mapsto \pi_{\text{opt}}(a|s)$. Figure 8.5 shows a picture of the simplified problem. The state S contains information

[2] In reality, the systems are usually more complicated. For example, in an auction-like procedure, the advertisers place bids on certain search queries, which then influence the price for a click.

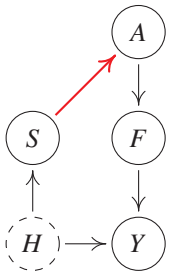

Figure 8.5: Example for the placement of advertisements. The target variable Y indicates whether a user has clicked on one of the shown ads. H (unknown) and S (known) are state variables and the action A corresponds to the mainline reserve, a real-valued parameter that determines how many ads are shown in the mainline. F is a discrete variable indicating the (known) number of ads placed in the mainline. Although the conditional $p(a\,|\,s)$ is randomized over, we may use $p(f\,|\,s)$ for the reweighting (see Proposition 8.2).

about the user that is available to the publisher. The hidden variable H contains unknown user information (e.g., his intention), the action A is the mainline reserve, and Y is the event whether or not a person clicks on one of the ads. Finally, F is the discrete variable that says, how many ads are shown. Evaluating new strategies $(a,s) \mapsto \tilde{p}(a\,|\,s)$, corresponds to applying Equation (8.4):

$$\hat{\xi}_{n,\mathrm{ERL}} := \frac{1}{n}\sum_{i=1}^{n} Y^i \,\frac{\tilde{p}(A^i\,|\,S^i)}{p(A^i\,|\,S^i)}.$$

(Here, we write $p(a\,|\,s)$ rather than $\pi(a\,|\,s)$ for notational convenience.) We can now benefit from the following key insight. Whether a person clicks on an ad depends on the mainline reserve A but only via the value of F. The user never sees the real-valued parameter A. This is a somewhat trivial observation, when we think about the causal structure of the system (see Figure 8.5). Exploiting this fact, however, we can use a different estimator

$$\frac{1}{n}\sum_{i=1}^{n} Y^i \,\frac{\tilde{p}(F^i\,|\,S^i)}{p(F^i\,|\,S^i)};$$

see Proposition 8.2. And since F is a discrete variable taking values between 0 and 4, say, this usually leads to weights that are much better behaved. In practice, the modification may reduce the size of confidence intervals considerably [Bottou et al., 2013, Section 5.1]. As in Section 8.1, we can exploit our knowledge of the causal structure to improve statistical performance. More formally, the procedure is justified by the following proposition:

Method	Training data from	Test domain
Domain generalization	$(\mathbf{X}^1, Y^1), \ldots, (\mathbf{X}^D, Y^D)$	$T := D+1$
Multi-task learning	$(\mathbf{X}^1, Y^1), \ldots, (\mathbf{X}^D, Y^D)$	$T \in \{1, \ldots, D\}$
Asymmetric multi-task learning	$(\mathbf{X}^1, Y^1), \ldots, (\mathbf{X}^D, Y^D)$	$T := D$

Table 8.1: In domain generalization, the test data come from an unseen domain, whereas in multi-task learning, some data in the test domain(s) are available.

Proposition 8.2 (Improved weighting) *Suppose there is a density p over $\mathbf{X} = (A, F, H, S, Y)$ that is entailed by an SCM $\mathfrak{C}$ with graph shown in Figure 8.5. Assume further that the density $\tilde{p}$ is entailed by an SCM $\tilde{\mathfrak{C}}$ that corresponds to an intervention in A of the form do $\big(A := \tilde{f}(S, \tilde{N}_A)\big)$ and satisfies $\tilde{p}(f \,|\, s) = 0$ if $p(f \,|\, s) = 0$ and $\tilde{p}(a \,|\, s) = 0$ if $p(a \,|\, s) = 0$. We then have*

$$\tilde{\mathbb{E}} Y \;=\; \int y \, \frac{\tilde{p}(a \,|\, s)}{p(a \,|\, s)} \, p(\mathbf{x}) \, d\mathbf{x} \;=\; \int y \, \frac{\tilde{p}(f \,|\, s)}{p(f \,|\, s)} \, p(\mathbf{x}) \, d\mathbf{x}.$$

The proof can be found in Appendix C.12. In general, the condition of the non-vanishing densities is indeed necessary: if there is a set of a and s values (with non-vanishing Lebesgue measure) that belong to the support of $\tilde{p}$ and contribute to the expectation of Y, there must be a non-vanishing probability under p to sample data in this area.

8.3 Domain Adaptation

Domain adaptation is another machine learning problem that is naturally related to causality [Schölkopf et al., 2012]. Here, we will relate domain adapation to what we called invariant prediction in "Different Environments" in Section 7.2.5. We do not claim that this connection, in its current form, yields major improvements, but we believe that it could prove to be useful for developing a novel methodology in domain adaptation.

Let us assume that we obtain data from a target variable Y^e and d possible predictors $\mathbf{X}^e = (X_1^e, \ldots, X_d^e)$ in different domains $e \in \mathcal{E} = \{1, \ldots, D\}$ and that we are interested in predicting Y. Adapting to widely used notation, we use the terms "domain" or "task." Table 8.1 describes a taxonomy of three problems in domain adaptation that we consider here.

Our main assumption is that there exists a set $S^* \subseteq \{1, \ldots, d\}$ such that the conditional $Y^e \,|\, \mathbf{X}^e_{S^*}$ is the same for all domains $e \in \mathcal{E}$, including the test domain, that is, for all $e, f \in \mathcal{E}$ and for all $\mathbf{x}_{S^*}$

$$Y^e \,|\, \mathbf{X}^e_{S^*} = \mathbf{x}_{S^*} \quad \text{and} \quad Y^f \,|\, \mathbf{X}^f_{S^*} = \mathbf{x}_{S^*} \quad \text{have the same distribution.} \tag{8.7}$$

In Sections 7.1.6 and 7.2.5 we have considered a similar setup, where we used the term "environments" rather than "domains" and called the property (8.7) "invariant prediction." We have argued that if there is an underlying SCM and if the environments correspond to interventions on nodes other than the target Y, property (8.7) is satisfied for $S^* = \mathbf{PA}_Y$ (cf. also our discussion of Simon's invariance criterion in Section 2.2). Property (8.7) may also hold, however, for sets other than the causal parents. Since our goal is prediction, we are most interested in sets S^* that satisfy (8.7) and additionally predict Y as accurately as possible. Let us for now assume, that we are given such a set S^* (we will return to this issue later) and point at how the assumption (8.7) relates to domain adaptation.

In settings of covariate shift [e.g., Shimodaira, 2000, Quionero-Candela et al., 2009, Ben-David et al., 2010], one usually assumes that the conditional $Y^e \,|\, \mathbf{X}^e = \mathbf{x}$ remains invariant over all tasks e. Assumption (8.7) means that covariate shift holds for some subset S^* of the variables and thus constitutes a generalization of the covariate shift assumption.

For domain generalization, and if the set S^* is known, we can then apply traditional methods for covariate shift for this subset S^*. For example, if the supports of the data in input space are overlapping (or the system is linear), we may use the estimator $f_{S^*}(\mathbf{X}^T_{S^*})$ with $f_{S^*}(\mathbf{x}) := \mathbb{E}\left[Y^1 \,|\, \mathbf{X}^1_{S^*} = \mathbf{x}\right]$ in test domain T. One can prove that this approach is optimal in an adversarial setting, where the distributions in the test domain may be arbitrarily different from the training domains, except for the conditional distribution (8.7) that we require to remain invariant [Rojas-Carulla et al., 2016, Theorem 1]. In multi-task learning, it is less obvious how to exploit the knowledge of such a set S^*. In practice, one needs to combine information gained from pooling the tasks and regressing Y on S^* with knowledge obtained from considering the test task separately [Rojas-Carulla et al., 2016].

If the set S^* is unknown, we again propose to search for sets S that satisfy (8.7) over available domains. When learning the causal predictors, one prefers to stay conservative, and the method of invariant causal prediction [Peters et al., 2016] therefore outputs the intersection of all sets S satisfying (8.7); see Equation (7.5). Here, we are interested in prediction instead. Among all sets that lead to invariant prediction, one may therefore choose the set S that leads to the best predictive performance, which is usually one of the larger of those sets. The same applies if there are different known sets S that all satisfy (8.7). If the data are generated by an SCM and the domains correspond to different interventions, the set S with the best predictive power that satisfies (8.7) can, in the limit of infinite data, be shown to be a subset of the Markov blanket of Y (see Problem 8.5).

8.4 Problems

Problem 8.3 (Half-sibling regression) *Consider the DAG in Figure 8.1. The fact that X provides additional information about Q on top of the one provided by Y follows from causal faithfulness. Why?*

Problem 8.4 (Inverse probability weighting) *Consider an SCM $\mathfrak{C}$ of the form*

$$\begin{aligned} Z &:= N_Z \\ Y &:= Z^2 + N_Y, \end{aligned}$$

with $N_Y, N_Z \overset{iid}{\sim} \mathcal{N}(0,1)$ and an intervened version $\tilde{\mathfrak{C}}$ with

$$do\left(Z := \tilde{N}_Z\right),$$

where $\tilde{N}_Z \sim \mathcal{N}(2,1)$.

a) (optional) Compute $\mathbb{E}[Y] := \mathbb{E}_{P^{\mathfrak{C}}}[Y]$ and $\tilde{\mathbb{E}}[Y] := \mathbb{E}_{P^{\tilde{\mathfrak{C}}}}[Y]$.

b) Draw $n = 200$ i.i.d. data points from the SCM $\mathfrak{C}$ and implement the estimator (8.3) for estimating $\tilde{\mathbb{E}}[Y]$.

c) Compute the estimate in b) and the empirical variance of the weights appearing in (8.3) for increasing sample size n between $n = 5$ and $n = 50,000$. What do you conclude?

Problem 8.5 (Invariant predictors) *We want to justify the last sentence in Section 8.3. Consider a DAG over variables Y, E, and $X_1, \ldots, X_d$, in which E (for "environment") is not a parent of Y and does not have any parents itself. Denote the Markov blanket of Y by M. Prove that for any set $S \subseteq \{X_1, \ldots, X_d\}$ with*

$$Y \perp\!\!\!\perp E \,|\, S$$

there is another set $S_{new} \subseteq M$ such that

$$Y \perp\!\!\!\perp E \,|\, S_{new} \quad \text{and} \quad Y \perp\!\!\!\perp (S \setminus S_{new}) \,|\, S_{new}.$$