Can Causal Latent Variable Models Improve Robustness of Treatment Effect Estimation?

Stanford CS229 Final Project

Sean Zhang
Department of Computer Science
Stanford University
szhanggg@stanford.edu

1 Introduction

Causal inference, an emergent branch of statistics, involves estimating the effect of an intervention or treatment on an outcome. These estimates rely on the assumption of unconfoundedness, which states that the method of assigning treatments is conditionally independent of potential outcomes. However, this assumption is often violated due to latent (unobservable) confounding variables. More broadly, another common source of bias in machine learning (ML) is covariate shift, a phenomenon where the target covariate distribution differs from that of a model's training data. Both of these biases are pervasive in causal inference tasks yet have been minimally explored in conjunction. Thus, this project aims to compare various non-latent and latent methods for estimating treatment effects where hidden confounding and covariate shift occur independently. Namely, we will compare accuracy in treatment effect estimation of the noncausal machine learning methods of eXtreme Gradient Boosting (XGBoost), Support Vector Machine (SVM), and k-Nearest Neighbors (k-NN) with the causal inference methods of Inverse Probability Weighting (IPW), Double Machine Learning (DML), and X-Learner, as well as a prominent causal latent variable model known as the Causal Effect Variational Autoencoder (CEVAE).

2 Related Work

Early methods for addressing covariate shift were largely driven by Shimodaira (2000), who proved that under covariate shift and model misspecification, the asymptotically optimal loss function is given by weighting by the ratio of covariate densities in the target population to the densities of the observed data. Landeiro and Culotta (2016) address *confounding shift*, a separate case where only the degree of hidden confounding changes from train to test distributions, in the task of text classification. They apply Pearl's backdoor adjustment Pearl (1995) to condition on observed confounders during training and marginalize them out during prediction, improving performance across various datasets. Similarly, in the context of imitation and reinforcement learning, Tennenholtz et al. (2021) show that extreme cases of confounding shift between expert data and the online environment causes models to learn catastrophic policies that render imitation learning impossible. More recently, in a traditional predictive modeling task, Dharmakeerthi et al. (2024) address a combined setting of *concept shift* (where the conditional distribution of outcomes $P(Y|X)$ changes from train to test) with covariate shift, such that both shifts are caused by latent confounders. They assume the existence of latent exogenous variables which are invariant to environment shifts, enabling the learning of a lower-dimensional invariant subspace which can reduce bias in predictions. Altogether, these recent works underscore the need to address both distribution shifts and latent confounders. However, the settings they explored are separate from hidden confounding and covariate shift occurring independently, and they do not address the task of treatment effect estimation. Our closest reference for treatment effect estimation under hidden confounding is an augmentation of Variational Autoencoders (VAE) by Louizos et al. (2017), which introduces the Causal Effect Variational Autoencoder (CEVAE) for estimating treatment effects in the presence of hidden confounders. CEVAE uses neural networks to infer a latent variable Z, and outperforms non-latent causal methods in settings with proxy noise.

3 Datasets and Features

We evaluate our models on the Twins (Almond et al., 2004) dataset. This dataset consists of real-world data on twin births, with 46 measured covariates related to their parents, the pregnancy, and the birth. Each pair is treated as one sample, where the treatment is denoted as being born the heavier twin and the outcome is first-year mortality. Since the overall mortality rate is low, we restrict our evaluation to pairs where both twins which under two kilograms.

To introduce hidden confounding with proxies as in Louizos et al. (2017), the GESTAT10 variable (an ordinal feature with 10 categories ranging from 0 to 9) is encoded using one-hot encoding, resulting in a 10-dimensional binary vector. This vector is then replicated three times to produce 30 binary features, and to introduce noise, each of the 30 bits is randomly and independently flipped with a set probability ranging from 0.05 to 0.5. As an example, consider a GESTAT10 value of 4, which corresponds to the one-hot encoded vector:

$$[0, 0, 0, 0, 1, 0, 0, 0, 0, 0].$$

After replication, this becomes:

$$[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0].$$

If we apply a flipping probability of 0.1, each bit in the 30-dimensional vector has a 10% chance of being flipped (i.e., changing from 0 to 1 or 1 to 0). For example, the flipped vector might become:

$$[0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0].$$

At flipping probabilities near 0.5, the confounder becomes the most hidden, allowing us to induce hidden confounding. We will induce covariate shift separately by performing correlation analysis on the cleaned data and creating a test set with modified distributions of the top k correlated variables.

4 Methods

4.1 Models

For non-causal models, we choose to evaluate XGBoost, Support Vector Machine (SVM), and k-Nearest Neighbors (k-NN) because of their prevalence in modern regression problems and their range of model complexity (decreasing, in the order they were mentioned). XGBoost (Chen and Guestrin, 2016)is a gradient-boosting algorithm that builds an ensemble of decision trees by sequentially minimizing a custom loss function, where each tree corrects the residuals of the previous ones to improve accuracy. SVM determines the optimal hyperplane in a high-dimensional space to separate data points or predict outcomes by maximizing the margin between classes, and can be used to model non-linear relationships using kernel functions. k-NN predicts outcomes by analyzing the k closest points in the feature space, assigning weights based on distance or averaging neighbor values, with no explicit training phase but high computational demands for large datasets.

Then, regarding causal models, we evaluate Inverse Probability Weighting (IPW), Double Machine Learning (DML), and the X-Learner.

IPW (Rosenbaum and Rubin, 1983) adjusts for confounding by reweighting samples based on the inverse of their propensity scores, where the propensity score is defined as $e(X) = P(T = 1 \mid X)$, which is the probability of receiving treatment given covariates X. The weights for treated units are consequently given as $w_i = \frac{1}{e(X_i)}$ and for control units as $w_i = \frac{1}{1-e(X_i)}$, ensuring that the weighted distribution of covariates is balanced between treated and control groups. This technique allows the estimation of the Average Treatment Effect (ATE) by weighting the outcomes proportionally to their likelihood of being treated, effectively simulating a randomized controlled trial. However, it is sensitive to extreme propensity scores, where weights can become unstable, leading to high variance in the estimates.

Double Machine Learning (DML) (Chernozhukov et al., 2024) employs a two-stage approach to estimate treatment effects while accounting for confounding, leveraging the *orthogonalization property*, which ensures that the estimation of the treatment effect is orthogonal (uncorrelated) to errors in the nuisance parameter models (e.g., the propensity score or outcome model). This property reduces bias from potential overfitting or misspecification of these nuisance models. In the first stage, models for the treatment assignment $\hat{e}(X)$ (the propensity score) and the conditional outcome $\hat{Y}(X, T)$ are fitted using machine learning methods such as regression trees, random forests, or neural networks. In the second stage, residuals are computed for the treatment and outcome as $\tilde{T} = T - \hat{e}(X)$ and $\tilde{Y} = Y - \hat{Y}(X, T)$, effectively removing the influence of confounding variables. Finally, the treatment effect is estimated by regressing $\tilde{Y}$ on $\tilde{T}$, yielding an unbiased estimate of the Average Treatment Effect (ATE) under relatively weak assumptions about the correct specification of the nuisance models.

The X-Learner (Künzel et al., 2019) first fits separate outcome models for the treated $\hat{Y}^1(X)$ and control $\hat{Y}^0(X)$ groups, then computes individual-level treatment effect estimates as $D_1 = Y_1 - \hat{Y}^0(X)$ for treated units and $D_0 = \hat{Y}^1(X) - Y_0$ for control units. These estimates are then used to train separate treatment effect models for the treated and control groups, which are combined to calculate the overall ATE as a weighted average, leveraging information from both groups to improve precision. The X-Learner's structure allows it to handle heterogeneous treatment effects effectively, even in unbalanced datasets.

Finally, we introduce our candidate causal latent variable model. The Causal Effect Variational Autoencoder (CEVAE) leverages a latent variable model to estimate causal effects under the assumption that the joint distribution $p(Z, X, t, y)$, comprising latent confounders Z, observed features X, treatment t, and outcome y, can be approximated from observational data. The generative model assumes the factorization

$$p(Z, X, t, y) = p(Z)p(X|Z)p(t|Z)p(y|t, Z),$$

with $Z \sim \mathcal{N}(0, I)$, $X|Z \sim \prod_{j=1}^{D_x} p(x_j|Z)$, and $t|Z \sim \text{Bernoulli}(\sigma(f_1(Z)))$, where $f_1(\cdot)$ is a neural network, and D_x is the dimensionality of X.

The outcome model $p(y|t, Z)$ is parameterized as $\mathcal{N}(\mu, \tilde{\nu})$ for continuous y or $\text{Bernoulli}(\pi)$ for binary y, where $\tilde{\nu}$ represents a fixed variance for continuous outcomes, and

$$\mu = tf_2(Z) + (1-t)f_3(Z) \quad \text{and} \quad \pi = \sigma(tf_2(Z) + (1-t)f_3(Z)),$$

with f_2 and f_3 as neural networks. To approximate the true posterior $p(Z|X, t, y)$, the inference network $q(Z|X, t, y)$ employs a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$, with

$$\mu = t\mu_{t=1} + (1-t)\mu_{t=0} \quad \text{and} \quad \sigma^2 = t\sigma^2_{t=1} + (1-t)\sigma^2_{t=0}.$$

The means $\mu_{t=i}$ and variances $\sigma^2_{t=i}$ are computed using treatment-specific neural networks $g_2 \circ g_1$ and $g_3 \circ g_1$, where $g_1(\cdot)$ extracts a shared representation. The training objective maximizes the Evidence Lower Bound (ELBO):

$$\mathcal{L} = \sum_{i=1}^{N} \mathbb{E}_{q(Z|X_i,t_i,y_i)} \big[\log p(X_i, t_i|Z) + \log p(y_i|t_i, Z) + \log p(Z) - \log q(Z|X_i, t_i, y_i) \big].$$

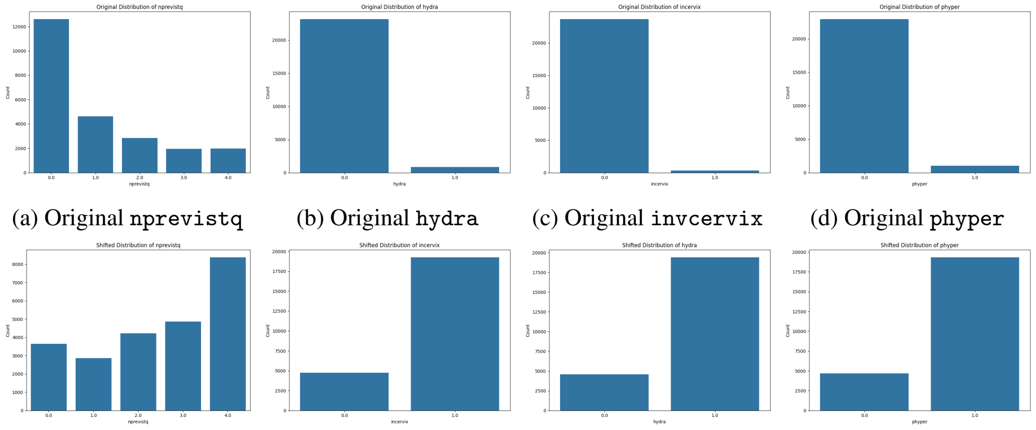
For out-of-sample predictions, i.e. new samples, CEVAE uses auxiliary distributions $q(t|X) = \text{Bernoulli}(\sigma(g_4(X)))$ and either $q(y|X, t) = \mathcal{N}(\mu, \tilde{\nu})$ for continuous y or $q(y|X, t) = \text{Bernoulli}(\pi)$ for binary y, where μ and π are parameterized by neural networks g_5, g_6, g_7. The final objective incorporates additional terms for the training data, where X_i^*, t_i^* and y_i^* are the observed training sample values for input, treatment, and outcome:

$$\mathcal{F}_{\text{CEVAE}} = \mathcal{L} + \sum_{i=1}^{N} \big(\log q(t_i = t_i^*|X_i^*) + \log q(y_i = y_i^*|X_i^*, t_i^*) \big).$$

Altogether, this architecture enables robust estimation of treatment effects under hidden confounding.

5 Experiments / Results / Discussion

Our final dataset consists of 11,984 pairs of twins. After performing correlation analysis, we obtained the four covariates most highly correlated with the outcome: the quintile ranking of prenatal visits (`nprevistq`), as well as three binary indicators of the conditions amniotic fluid deficiency, (`hydra`), incompetent cervix (`invcervix`), and high blood pressure (`phyper`). Respectively, these variables



(a) Original `nprevistq` (b) Original `hydra` (c) Original `invcervix` (d) Original `phyper`

(e) Shifted `nprevistq` (f) Shifted `incervix` (g) Shifted `hydra` (h) Shifted `phyper`

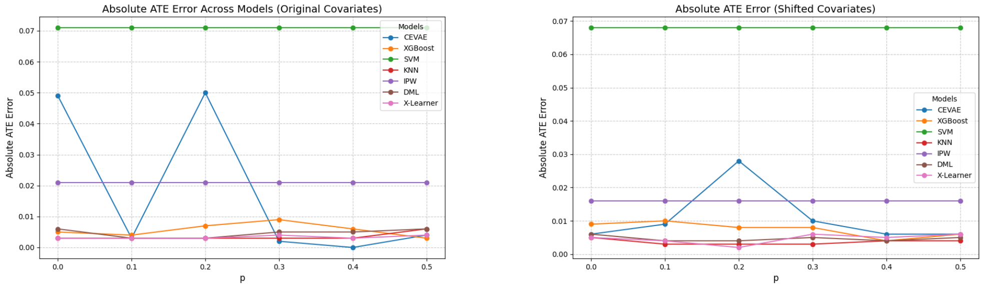
Figure 1: Original vs. Shifted Distributions

had correlations of -0.1647, 0.1367, 0.0824, and -0.0535 with mortality. We modify their distributions in a shifted version of the original covariate dataset, as displayed in Figure 1; the specific densities are given in Table 1.

Covariate	**Category**	**Original (%)**	**Shifted (%)**
`nprevistq`	0.0	52.55	15.00
	1.0	19.25	12.00
	2.0	11.83	18.00
	3.0	8.13	20.00
	4.0	8.24	35.00
`hydra`	0	96.53	80.00
	1	3.47	20.00
`incervix`	0	98.71	80.00
	1	1.29	20.00
`phyper`	0	95.60	80.00
	1	4.40	20.00

Table 1: Original and Shifted Distributions for Covariates

In our experiments, we use a CEVAE model with a specified latent dimension of 30 and two hidden layers each consisting of 300 nodes, mirroring the architecture of our Louizos et al. (2017). We train for 40 epochs with a learning rate of $1e^{-3}$ and decay rate of 0.95, weight decay rate of $1e^{-4}$, and batch size of 512. For all other models, we use default hyperparameter values, including a random forest with 100 estimators and unrestricted depth for DML and X-Learner. We measure our models' performance by the absolute error in Average Treatment Effect (ATE) estimation, formally defined as $\text{ATE} = E[Y|T=1] - E[Y|T=0]$. All metrics were obtained using 10-fold cross validation. The true ATE is estimated as -0.0284.



(a) With Original Covariates (b) With Shifted Covariates

Figure 2: Comparison of Model Performances

	ATE Absolute Error (No Shift)						ATE Absolute Error (Shifted)					
p	0.0	0.1	0.2	0.3	0.4	0.5	0.0	0.1	0.2	0.3	0.4	0.5
CEVAE	0.049	0.003	0.050	0.002	0.000	0.004	0.006	0.009	0.028	0.010	0.006	0.006
XGBoost	0.009	0.010	0.008	0.008	0.004	0.006	0.005	0.004	0.007	0.009	0.006	0.003
SVM	0.071	0.071	0.071	0.071	0.071	0.071	0.068	0.068	0.068	0.068	0.068	0.068
KNN	0.005	0.003	0.003	0.003	0.004	0.004	0.003	0.003	0.003	0.003	0.003	0.006
IPW	0.021	0.021	0.021	0.021	0.021	0.021	0.016	0.016	0.016	0.016	0.016	0.016
DML	0.006	0.004	0.004	0.005	0.004	0.005	0.006	0.003	0.003	0.005	0.005	0.006
X-Learner	0.005	0.004	0.002	0.006	0.005	0.006	0.003	0.003	0.003	0.004	0.003	0.004

Table 2: Comparison of ATE Absolute Errors

Our results are given in Figure 2 and Table 2, where p represents the bit flip probability in the hidden confounder generation process and varies evenly along the x-axis for six values from 0.0 to 0.5. Overall, the results are highly inconclusive. We do note that the CEVAE accuracy increased with increasing levels of p; for example, comparing the average error between the three lowest p values to the average from the three highest p values, we see that these average errors decreased from 0.0143 to 0.007 in the setting with shift and from 0.034 to 0.003 in the setting without shift. This implies that our latent variable generation works as intended; given that we initialized our CEVAE to account for a hidden confounder of dimension 30, it makes sense that it would perform worse when no such latent confounders are present, or if they are only weakly present, as in $p = 0.0, 0.1, 0.2$. However, as we increased p, we expected to see a degradation in performance across all models except the CEVAE. The results do not support this; across both the original and the shifted conditions, the estimated ATE errors remained remarkably stable, with even the non-causal estimators showing only minor fluctuations in absolute error. The fluctuations appear relatively nondeterministic, although X-Learner shows larger errors with increasing levels of confounding in both settings. One possible explanation is that the twins dataset, provides covariates that might not correlate heavily with the outcome, but contain unique, non-overlapping information such that altogether they hold significant predictive power. Similarly, another possibility is that the added noisy confounders were not sufficiently influential compared to the original covariates and treatment assignments, thereby failing to manifest in notably different ATE errors.

6 Conclusion / Future Work

In conclusion, while we expected greater divergence as p increased, the results suggest a surprising level of robustness in these models to the introduced hidden confounding. The combined setting of latent confounding and covariate shift co-occurring independently was not significantly addressed by the CEVAE, which we had expected to outperform other non-causal and non-latent models. Future analysis works could aim evaluate datasets that are more susceptible to covariate shift or have stronger influences from latent confounders. Future modeling works could seek to find a latent representation that captures both of these biases and can show consistent improvement in treatment estimation.

7 Contributions

As the only member I wrote the code for data processing and generation and model evaluation.

References

Douglas Almond, Kenneth Y Chay, and David S Lee. 2004. The costs of low birth weight. Working Paper 10552, National Bureau of Economic Research.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. 2024. Double/debiased machine learning for treatment and causal parameters.

Kulunu Dharmakeerthi, YoonHaeng Hur, and Tengyuan Liang. 2024. Learning when the concept shifts: Confounding, invariance, and dimension reduction.

Sören R. Künzel, Jasjeet S. Sekhon, Peter J. Bickel, and Bin Yu. 2019. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, 116(10):4156–4165.

Virgile Landeiro and Aron Culotta. 2016. Robust text classification in the presence of confounding bias. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

Christos Louizos, Uri Shalit, Joris Mooij, David Sontag, Richard Zemel, and Max Welling. 2017. Causal effect inference with deep latent-variable models. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6449–6459, Red Hook, NY, USA. Curran Associates Inc.

Judea Pearl. 1995. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688.

Paul R. Rosenbaum and Donald B. Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55.

Hidetoshi Shimodaira. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90:227–244.

Guy Tennenholtz, Assaf Hallak, Gal Dalal, Shie Mannor, Gal Chechik, and Uri Shalit. 2021. On covariate shift of latent confounders in imitation and reinforcement learning.