“Why did the Model Fail?”: Attributing Model Performance Changes to Distribution Shifts

Anonymous Author(s)
Affiliation
Address
email

Abstract

Performance of machine learning models may differ significantly in novel environments compared to during training due to shifts in the underlying data distribution. Attributing performance changes to specific data shifts is critical for identifying sources of model failures and designing stable models. In this work, we design a novel method for attributing performance differences between environments to shifts in the underlying causal mechanisms. We formulate the problem as a cooperative game and derive an importance weighting method for computing the value of a coalition of distributions. The contribution of each distribution to the total performance change is then quantified as its Shapley value. We demonstrate the correctness and utility of our method on two synthetic datasets and two real-world case studies, showing its effectiveness in attributing performance changes to a wide range of distribution shifts.

1 Introduction

Machine learning models are widely deployed in dynamic environments ranging from recommendation systems to personalized clinical care. Such environments are prone to distribution shifts, which may lead to serious degradations in model performance [12, 7, 17, 11, 23]. Importantly, such shifts are hard to anticipate and reduce the ability of model developers to design reliable systems. When the performance of a model *does* degrade during deployment, it is crucial for the model developer to know *how* the distribution has shifted to cause this change. Cognizant of this information, the model developer can then take mitigating actions such as additional data collection, data augmentation, and model retraining [3, 43, 32].

In this work, we present a method to attribute changes in model performance to shifts in a given set of distributions. Distribution shifts can occur in various marginal or conditional distributions that comprise variables involved in the model. Further, multiple distributions can change simultaneously. We handle this in our framework by defining the effect of changing any *set* of distributions on model performance, and use the concept of Shapley values [29] to attribute the change to individual distributions. The Shapley value is a co-operative game theoretic framework with the goal of distributing surplus generated by the players in the co-operative game according to their contribution. In our framework, the players correspond to individual distributions.

Most relevant to our contributions is the work of Budhathoki et al. [5], which attributes a shift between two joint distributions to a specific set of individual distributions (i.e. factorization of the joint distribution induced by causal structural assumptions). This line of work defines distribution shifts as interventions on causal mechanisms [25, 32, 33, 5, 36]. We build on their framework to justify the players in our cooperative game. We significantly differ from the end goal by attributing a change in *model performance* to individual distributions. Note that each shifted distribution may influence model performance differently and may result in different attributions than their contributions to the change in the joint distribution. We discuss additional related work in Appendix A.

Submitted to the Workshop on Distribution Shifts, 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

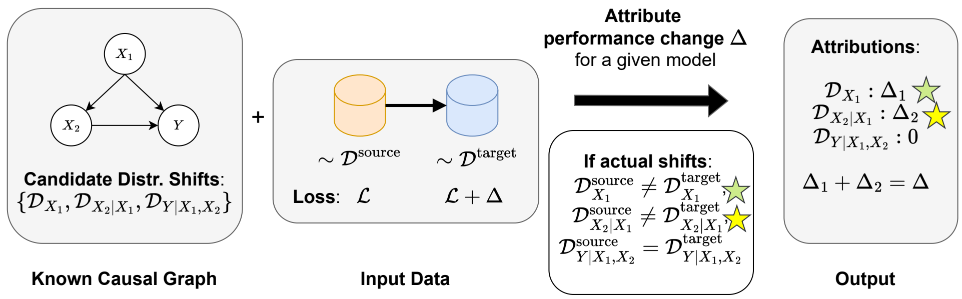


Figure 1: **Inputs and outputs for attribution.** Input: Causal graph, where all variables are observed providing the candidate distribution shifts we consider. The goal is to attribute the model's performance change Δ between source and target distributions to these candidate distributions. Here, out of the three candidate distributions, the marginal distribution of X_1 and the conditional distribution of X_2 given X_1 change. Our method attributes changes to each one such that the attributions sum to the total performance change Δ.

In this work, we focus on explaining the discrepancy in model performance as measured by some metric such as prediction accuracy. Explaining performance discrepancy requires us to develop specialized methods. We particularly focus on model-free importance sampling approaches and approximations of Shapley value estimation that allow us to expand the settings where our method is applicable.

2 Preliminaries

Consider a learning setup where we have some system variables denoted by V consisting of two types of variables $V = (X, Y)$, which comprises of features X and labels Y such that $V \sim \mathcal{D}$. Realizations of the variables are denoted in lower case. We assume access to samples from two environments. We use $\mathcal{D}^{\text{source}}$ to denote the source distribution and $\mathcal{D}^{\text{target}}$ for the target distribution. Subscripts on $\mathcal{D}$ refer to the distribution of specific variables. For example, $\mathcal{D}_{X_1}$ is the distribution of feature $X_1 \subset X$, and $\mathcal{D}_{Y|X}$ is the conditional distribution of labels given all features X.

Let $X_{\mathtt{M}} \subseteq X$ be the subset of features utilized by a given model f. We are given a loss function $\ell((x, y), f) \mapsto \mathbb{R}$ which assigns a real value to the model evaluated at a specific setting x of the variables. For example, in the case of supervised learning, the model f maps $X_{\mathtt{M}}$ into the label space, and a loss function such as the squared error $\ell((x, y), f) := (y - f(x_{\mathtt{M}}))^2$ can be used to evaluate model performance. We assume that the loss function can be computed separately for each data point. Then, performance of the model in some environment with distribution $\mathcal{D}$ is summarized by the average of the losses:

$$\text{Perf}(\mathcal{D}) := \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell((x, y), f)]$$

This implies that a shift in any variables V in the system may result in performance change across environments, including those that are not directly used by the model, but drive changes to the features $X_{\mathtt{M}}$ used by the model for learning.

3 Method

We now formalize our problem setup and motivate a game theoretic method for attributing performance changes to distributions over variable subsets. We show desirable properties of our method in Appendix C, and derive the analytical attributions for a synthetic setting in Appendix D.

3.1 Problem Setup

Suppose we are given a *candidate set* of (marginal and/or conditional) distributions $\mathtt{C}_{\mathcal{D}}$ over V that may account for the model performance change from $\mathcal{D}^{\text{source}}$ to $\mathcal{D}^{\text{target}}$: $\text{Perf}(\mathcal{D}^{\text{target}}) - \text{Perf}(\mathcal{D}^{\text{source}})$. **Our goal is to attribute this change to each candidate distribution in the candidate set $\mathtt{C}_{\mathcal{D}}$.** For our method, we assume access to the model f, and samples from $\mathcal{D}^{\text{source}}$ as well as $\mathcal{D}^{\text{target}}$ (see Figure 1). We make the following assumptions:

Assumption 3.1. The causal graph corresponding to the data-generating mechanism is known and all variables in the system are observed. Thus, the factorization of the joint distribution $\mathcal{D}_V$ is known.

Assumption 3.2. Distribution shifts of interest are due to (independent) shifts in one or more factors of $\mathcal{D}_V$.

3.2 Game Theoretic Distribution Shift Attribution

Consider the following attribution game where the set of *players* in this game are the candidate distributions. A *coalition* of any subset of players determines the distributions that are allowed to shift, keeping the rest fixed. The *value* for the coalition is the model performance change between the resulting distribution for the coalition and the training distribution.

Choice of Candidate Distribution Shifts. First, we clarify the choice of candidate distributions that will inform the coalition. In order to attribute performance changes to shifts in the distribution of input features or labels, our candidate distributions can constitute marginal and conditional distribution of the covariates and labels. For instance, it can be the set of marginal distributions on each system variable, $\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_{X_1}, \mathcal{D}_{X_2}, \cdots\}$, or distribution of each variable after conditioning on the rest, $\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_{X_1|V\setminus X_1}, \mathcal{D}_{X_2|V\setminus X_2}, \cdots\}$. Since we have combinatorially many shifts that can be defined on subsets of $V = (X, Y)$, the choice of candidate sets is challenging.

Here, we propose to use the knowledge of the causal graph [24] for the system as our candidate set. The causal graph specifies the factorization of the joint distribution into a set of distributions (or mechanisms). That is $\mathcal{D}_V = \prod_{X_i \in V} \mathcal{D}_{X_i|\text{parent}(X_i)}$ where parent(X_i) are the variables that have a directed edge to X_i in the causal graph. This factorization is known by Assumption 3.1. Then, we can form the candidate set constituting each distribution in this factorization. That is,

$$\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_{X_1|\text{parent}(X_1)}, \cdots, \mathcal{D}_{X_i|\text{parent}(X_i)}, \cdots\}_{i=1,\cdots,|V|}.$$

For a node without parents in the causal graph, the parent set can be empty, which reduces $\mathcal{D}_{X_i}$ to a marginal distribution.

Advantages of using causal mechanisms. This choice of candidate set has three main advantages. First, it is *interpretable* since the candidate shifts are specified by domain experts who constructed the causal graph. Second, it is *actionable* since identifying the causal mechanisms most responsible for performance change can inform mitigating methods for handling distribution shifts [32]. Third, it will lead to *succinct* attributions due to the independence property.

Value of a Coalition. Consider a coalition of distributions $\widetilde{\mathtt{C}} \subseteq \mathtt{C}_{\mathcal{D}}$. The resulting distribution over variables V in the system, corresponding to the coalition $\widetilde{\mathtt{C}}$ is

$$\widetilde{\mathcal{D}} = \left(\prod_{i:\mathcal{D}_{X_i|\text{parent}(X_i)} \in \widetilde{\mathtt{C}}} \mathcal{D}^{\text{target}}_{X_i|\text{parent}(X_i)} \right) \left(\prod_{i:\mathcal{D}_{X_i|\text{parent}(X_i)} \notin \widetilde{\mathtt{C}}} \mathcal{D}^{\text{source}}_{X_i|\text{parent}(X_i)} \right) \tag{1}$$

Note that the coalition only consists of distributions that are allowed to change across environments. All other relevant mechanisms are fixed to the source distribution. The value of the coalition $\widetilde{\mathtt{C}}$ with the full distribution $\widetilde{\mathcal{D}}$ is now given by

$$\text{Val}(\widetilde{\mathtt{C}}) := \text{Perf}(\widetilde{\mathcal{D}}) - \text{Perf}(\mathcal{D}^{\text{source}}) \tag{2}$$

Then, we obtain the attribution of each player $d \in \mathtt{C}_{\mathcal{D}}$ using the Shapley value framework [29]. Crucially, to compute our attributions, we need estimates of model performance under $\widetilde{\mathcal{D}}$. Note that we only have model performance estimates under $\mathcal{D}^{\text{source}}$ and $\mathcal{D}^{\text{target}}$, but not for any arbitrary coalition where only a subset of the distributions have shifted. To estimate the performance of any coalition, we propose to use importance sampling.

3.3 Estimating Performance using Importance Sampling

Assumption 3.3. $\mathtt{support}(\mathcal{D}^{\text{target}}_{X_i|\text{parent}(X_i)}) \subseteq \mathtt{support}(\mathcal{D}^{\text{source}}_{X_i|\text{parent}(X_i)})$ for all $\mathcal{D}^{\text{target}}_{X_i|\text{parent}(X_i)} \in \mathtt{C}_{\mathcal{D}}$.

Importance sampling allows us to re-weight the samples drawn from a given distribution, which can be $\mathcal{D}^{\text{source}}$ or $\mathcal{D}^{\text{target}}$, to simulate expectations for a desired distribution, which is the candidate $\widetilde{\mathcal{D}}$ in our case. Thus, we re-write the value as

$$\begin{aligned} \text{Val}(\widetilde{\mathtt{C}}) &= \text{Perf}(\widetilde{\mathcal{D}}) - \text{Perf}(\mathcal{D}^{\text{source}}) \\ &= \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{source}}}\left[\frac{\widetilde{\mathcal{D}}((x,y))}{\mathcal{D}^{\text{source}}((x,y))}\ell((x,y),f)\right] - \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{source}}}[\ell((x,y),f)] \end{aligned} \tag{3}$$

The importance weights are themselves a product of ratios of source and target distributions corresponding to the causal mechanisms in $\mathtt{C}_{\mathcal{D}}$ as follows:

$$w_{\widetilde{\mathtt{C}}}((x,y)) := \frac{\widetilde{\mathcal{D}}((x,y))}{\mathcal{D}^{\text{source}}((x,y))} = \prod_{d \in \widetilde{\mathtt{C}}} \frac{\mathcal{D}_d^{\text{target}}((x,y))}{\mathcal{D}_d^{\text{source}}((x,y))} =: \prod_{d \in \widetilde{\mathtt{C}}} w_d((x,y)) \tag{4}$$

By Assumption 3.3, we ensure that all importance weights are finite. Here, we use a simple approach for density ratio estimation via training probabilistic classifiers as described in Sugiyama et al. [34, Section 2.2].

Let D be a binary random variable, such that when $D = 1, Z \sim \mathcal{D}_d^{\text{target}}(Z)$, and when $D = 0, Z \sim \mathcal{D}_d^{\text{source}}(Z)$. Suppose $d = \mathcal{D}_{X_i|\text{parent}(X_i)}$, then

$$w_d = \frac{\mathbb{P}(D = 0|\text{parent}(X_i))}{\mathbb{P}(D = 1|\text{parent}(X_i))} \cdot \frac{\mathbb{P}(D = 1|X_i, \text{parent}(X_i))}{\mathbb{P}(D = 0|X_i, \text{parent}(X_i))},$$

where each term is computed using a probabilistic classifier trained to discriminate data points from $\mathcal{D}^{\text{source}}$ and $\mathcal{D}^{\text{target}}$ from the concatenated dataset. We show the derivation of this equation in Appendix B. In total, we need to learn $\mathcal{O}(|\mathtt{C}_{\mathcal{D}}|)$ models for computing all importance weights.

4 Empirical Evaluation

We first evaluate our method using a synthetic dataset where the ground-truth shifts are known (Section E.1). Then, we evaluate our method on a semi-synthetic dataset generated from CelebA using a CausalGAN [16] (Appendix Section E.2). Finally, we demonstrate the utility of our method on a real-world clinical mortality prediction task (shown here).

Setup. Clinical machine learning models are being increasingly deployed in the real-world in hospitals, laboratories, and Intensive Care Units (ICUs) [30]. However, prior work has shown that such machine learning models are not robust to distribution shifts, and frequently degrade in performance on distributions different than what is seen during training [31]. Here, we explore a simulated case study where a model which predicts mortality in the ICU is deployed in a different geographical region from where it is trained. We use data from the eICU Collaborative Research Database V2.0 [27]. Here, we simulate the deployment of a model trained on data from the Midwestern US (source) to the Southern US (target). We learn an `XGB` [6] model to predict mortality given vitals, labs, and demographics data. We assume the causal graph in Figure E.3b, informed by prior work utilizing causal discovery on this dataset [31]. As prior work has shown limited performance drops for models in this setting [44], we oversample younger population in the source environment to create an additional semi-synthetic distribution shift. We use our method to attribute the increase in Brier score from Midwest to South datasets.

Our method provides actionable attributions. First, we observe from our attributions (Figure E.8a) that shifts in the age distribution is responsible for 16.2% of the total shift. This confirms the validity of the attributions on a known semi-synthetic shift. Although there are more significant mechanism shifts (Figure E.8a), suppose that the practitioner decides to focus on mitigating the shift in age. To do so, they first plot the age distribution in the source and target environments (Figure E.8b), finding that the target domain has dramatically more older patients. Then, they choose to collect additional data from the older population in the source. Training a new model on this augmented dataset, they find that the drop in performance is reduced by 21.3%. The practitioner may next turn their attention to mitigating shifts in more impactful conditional mechanisms such as $\mathcal{D}_{\text{Labs|Age, Demo, Surgery}}$, using methods such as domain adversarial training [10] or GAN data augmentation [22], but we leave such explorations to future work.

5 Discussion

We propose a method to attribute changes in performance of a model deployed on a different distribution from the training distribution. Our work assumes knowledge of the causal graph to obtain interpretable and succinct attributions. While we can certainly obtain reasonable attributions from a misspecified graph, we argue that such attributions may not be minimal. Future work includes relaxing the assumption that all variables are observed, comparing strategies for mitigating conditional shifts, and extending the experiments to additional settings such as unsupervised learning and reinforcement learning.

References

[1] Alnur Ali, Maxime Cauchois, and John C. Duchi. The lifecycle of a statistical model: Model failure detection, identification, and refitting, 2022. URL `https://arxiv.org/abs/2202.04166`.

[2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[3] Rob Ashmore, Radu Calinescu, and Colin Paterson. Assuring the machine learning lifecycle: Desiderata, methods, and challenges. *ACM Computing Surveys (CSUR)*, 54(5):1–39, 2021.

[4] Benjamin Aubin, Agnieszka Słowik, Martin Arjovsky, Leon Bottou, and David Lopez-Paz. Linear unit-tests for invariance discovery. *arXiv preprint arXiv:2102.10867*, 2021.

[5] Kailash Budhathoki, Dominik Janzing, Patrick Bloebaum, and Hoiyi Ng. Why did the distribution change? In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1666–1674. PMLR, 13–15 Apr 2021. URL `https://proceedings.mlr.press/v130/budhathoki21a.html`.

[6] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[7] Prathyush Chirra, Patrick Leo, Michael Yim, B Nicolas Bloch, Ardeshir R Rastinehad, Andrei Purysko, Mark Rosen, Anant Madabhushi, and Satish Viswanath. Empirical evaluation of cross-site reproducibility in radiomic features for characterizing prostate mri. In *Medical Imaging 2018: Computer-Aided Diagnosis*, volume 10575, page 105750B. International Society for Optics and Photonics, 2018.

[8] Greg d'Eon, Jason d'Eon, James R. Wright, and Kevin Leyton-Brown. The spotlight: A general method for discovering systematic errors in deep learning models, 2021. URL `https://arxiv.org/abs/2107.00758`.

[9] Sabri Eyuboglu, Maya Varma, Khaled Kamal Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Re. Domino: Discovering systematic errors with cross-modal embeddings. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=FPCMqjI0jXN`.

[10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

[11] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[12] Lin Lawrence Guo, Stephen R Pfohl, Jason Fries, Alistair EW Johnson, Jose Posada, Catherine Aftandilian, Nigam Shah, and Lillian Sung. Evaluation of domain generalization and adaptation on improving model robustness to temporal dataset shift in clinical medicine. *Scientific reports*, 12(1):1–10, 2022.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[14] Dominik Janzing, Lenon Minorics, and Patrick Blöbaum. Feature relevance quantification in explainable ai: A causal problem. In *International Conference on artificial intelligence and statistics*, pages 2907–2916. PMLR, 2020.

[15] Alistair E. W. Johnson, Tom J. Pollard, and Tristan Naumann. Generalizability of predictive models for intensive care unit patients, 2018. URL `https://arxiv.org/abs/1812.02275`.

[16] Murat Kocaoglu, Christopher Snyder, Alexandros G Dimakis, and Sriram Vishwanath. Causalgan: Learning causal implicit generative models with adversarial training. *arXiv preprint arXiv:1709.02023*, 2017.

[17] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al.

Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021.

[18] Sean Kulinski, Saurabh Bagchi, and David I Inouye. Feature shift detection: Localizing which features have shifted via conditional distribution tests. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19523–19533. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper/2020/file/e2d52448d36918c575fa79d88647ba66-Paper.pdf`.

[19] Indra Kumar, Carlos Scheidegger, Suresh Venkatasubramanian, and Sorelle Friedler. Shapley residuals: Quantifying the limits of the shapley value for explanations. *Advances in Neural Information Processing Systems*, 34, 2021.

[20] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[21] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf`.

[22] Giovanni Mariani, Florian Scheidegger, Roxana Istrate, Costas Bekas, and Cristiano Malossi. Bagan: Data augmentation with balancing gan. *arXiv preprint arXiv:1803.09655*, 2018.

[23] Bret Nestor, Matthew BA McDermott, Willie Boag, Gabriela Berner, Tristan Naumann, Michael C Hughes, Anna Goldenberg, and Marzyeh Ghassemi. Feature robustness in non-stationary health records: caveats to deployable model performance in common clinical machine learning tasks. In *Machine Learning for Healthcare Conference*, pages 381–405. PMLR, 2019.

[24] Judea Pearl. *Causality*. Cambridge university press, 2009.

[25] Judea Pearl and Elias Bareinboim. Transportability of causal and statistical relations: A formal approach. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[26] Aleksandr Podkopaev and Aaditya Ramdas. Tracking the risk of a deployed model and detecting harmful distribution shifts. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=Ro_zAjZppv`.

[27] Tom J Pollard, Alistair EW Johnson, Jesse D Raffa, Leo A Celi, Roger G Mark, and Omar Badawi. The eicu collaborative research database, a freely available multi-center database for critical care research. *Scientific data*, 5(1):1–13, 2018.

[28] Stephan Rabanser, Stephan Günnemann, and Zachary Lipton. Failing loudly: An empirical study of methods for detecting dataset shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper/2019/file/846c260d715e5b854ffad5f70a516c88-Paper.pdf`.

[29] Alvin E Roth. *The Shapley value: essays in honor of Lloyd S. Shapley*. Cambridge University Press, 1988.

[30] Mark P Sendak, Joshua D'Arcy, Sehj Kashyap, Michael Gao, Marshall Nichols, Kristin Corey, William Ratliff, and Suresh Balu. A path for translation of machine learning products into healthcare delivery. *EMJ Innov*, 10:19–00172, 2020.

[31] Harvineet Singh, Vishwali Mhasawade, and Rumi Chunara. Generalizability challenges of mortality risk prediction models: A retrospective analysis on a multi-center database. *PLOS Digital Health*, 1(4):e0000023, 2022.

[32] Adarsh Subbaswamy, Peter Schulam, and Suchi Saria. Preventing failures due to dataset shift: Learning predictive models that transport. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3118–3127, 2019.

[33] Adarsh Subbaswamy, Roy Adams, and Suchi Saria. Evaluating model robustness and stability to dataset shift. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 2611–2619. PMLR, 13–15 Apr 2021. URL `http://proceedings.mlr.press/v130/subbaswamy21a.html`.

[34] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. Density-ratio matching under the bregman divergence: a unified framework of density-ratio estimation. *Annals of the Institute of Statistical Mathematics*, 64(5):1009–1044, 2012.

[35] Mukund Sundararajan and Amir Najmi. The many shapley values for model explanation. In *International conference on machine learning*, pages 9269–9278. PMLR, 2020.

[36] Nikolaj Thams, Michael Oberst, and David Sontag. Evaluating robustness to dataset shift via parametric robustness sets. *arXiv preprint arXiv:2205.15947*, 2022.

[37] Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.*, 41(3):647–665, dec 2014. ISSN 0219-1377. doi: 10.1007/s10115-013-0679-x. URL `https://doi.org/10.1007/s10115-013-0679-x`.

[38] Jiaxuan Wang, Jenna Wiens, and Scott Lundberg. Shapley flow: A graph-based approach to interpreting model predictions. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 721–729. PMLR, 13–15 Apr 2021. URL `https://proceedings.mlr.press/v130/wang21b.html`.

[39] Qing Wang, Sanjeev R Kulkarni, and Sergio Verdú. Divergence estimation of continuous distributions based on data-dependent partitions. *IEEE Transactions on Information Theory*, 51 (9):3064–3074, 2005.

[40] Qing Wang, Sanjeev R Kulkarni, and Sergio Verdú. A nearest-neighbor approach to estimating divergence between continuous random vectors. In *2006 IEEE International Symposium on Information Theory*, pages 242–246. IEEE, 2006.

[41] Qing Wang, Sanjeev R Kulkarni, and Sergio Verdú. Divergence estimation for multidimensional densities via k-nearest-neighbor distances. *IEEE Transactions on Information Theory*, 55(5): 2392–2405, 2009.

[42] Eric Wu, Kevin Wu, and James Zou. Explaining medical ai performance disparities across sites with confounder shapley value analysis, 2021. URL `https://arxiv.org/abs/2111.08168`.

[43] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017.

[44] Haoran Zhang, Natalie Dullerud, Laleh Seyyed-Kalantari, Quaid Morris, Shalmali Joshi, and Marzyeh Ghassemi. An empirical framework for domain generalization in clinical settings. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 279–290, 2021.

A Related Work

Identifying relevant distribution shifts. There has been extensive work that tests whether the data distribution has shifted (e.g. ones evaluated in Rabanser et al. [28]). Past work has proposed to identify sub-distributions (factors constituting the joint distribution as determined by a generative model for the data) that comprise the shift between two joint distributions and order them by their contribution to the shift [5]. However, as suggested before, the sub-distributions may have different influence on model performance. Even a small change in some (factors) may have a large effect on model performance (and vice-versa). Thus, a model developer has to filter distributions to identify ones that actually impact model performance (see Property 2.2 and Appendix D). Further, Budhathoki et al. [5] focuses on changes to the joint distribution as measured by the KL-divergence, which requires assumptions on the class of distributions to leverage closed-form expressions of KL-divergence (such as exponential families), or non-parametric KL estimation which is challenging in high dimensions [39, 40].

Other approaches which aim to localize shifts to individual variables (conditional on the rest of the variables) do not provide a way to identify the ones relevant to performance [18]. In contrast to testing for shifts, Podkopaev and Ramdas [26] tests for changes in model performance when distribution changes in deployment. Recent work by Wu et al. [42] decomposes performance change to changes in only marginal distributions using Shapley value framework [21]. However, the method as described is restricted to categorical variables.

Shapley values for attribution. Shapley value-based attribution has recently become popular for interpreting model predictions [37, 21, 38]. In most prior work, Shapley values have been leveraged for attributing a specific model prediction to the input features [35]. Challenges to appropriately interpreting such attributions and desirable properties thereof have been extensively discussed in [14, 19]. In this work, we advance the use of Shapley values for interpreting model performance changes to sub-distributions at the dataset level.

Detecting data partitions with low model performance. Recent work aims to find subsets of the dataset that have significantly worse (or better) performance [8, 9]. However, they do not study changes in the underlying data distribution. The work by Ali et al. [1] describes a method to identify and localize a change in model performance, and is applicable under distribution shifts. The main difference in our work is the data representations used for attribution. Instead of identifying subsets of *data* that are relevant to performance change, we find sub-*distributions* represented by causal mechanisms.

B Derivation of Importance Weights

Let D be a binary random variable, such that when $D = 1, X \sim \mathcal{D}^{\text{target}}(X)$, and when $D = 0, X \sim \mathcal{D}^{\text{source}}(X)$. Suppose $d = \mathcal{D}_{X_i|\text{parent}(X_i)}$, then, for a particular value (x, y):

$$\begin{aligned}
\mathcal{D}_d^{\text{target}}((x,y)) &:= \mathbb{P}(X_i = x|\text{parent}(X_i) = \text{parent}(x_i), D = 1) \\
&= \frac{\mathbb{P}(D = 1, \text{parent}(X_i) = x_i|X_i = x_i) \cdot \mathbb{P}(X_i = x_i)}{\mathbb{P}(D = 1, \text{parent}(X_i) = x_i)} \\
&= \frac{\mathbb{P}(D = 1|\text{parent}(X_i) = x_i, X_i = x_i) \cdot \mathbb{P}(X_i = x_i, \text{parent}(X_i) = X_i)}{\mathbb{P}(D = 1|\text{parent}(X_i) = x_i) \cdot \mathbb{P}(\text{parent}(X_i) = x_i)}
\end{aligned}$$

Then,

$$\begin{aligned}
w_d &= \frac{\mathcal{D}_d^{\text{target}}((x,y))}{\mathcal{D}_d^{\text{source}}((x,y))} \\
&= \frac{\mathbb{P}(D = 0|\text{parent}(X_i) = \text{parent}(x_i))}{\mathbb{P}(D = 1|\text{parent}(X_i) = \text{parent}(x_i))} \cdot \frac{\mathbb{P}(D = 1|X_i = x_i, \text{parent}(X_i) = \text{parent}(x_i))}{\mathbb{P}(D = 0|X_i = x_i, \text{parent}(X_i) = \text{parent}(x_i))} \\
&= \frac{1 - \mathbb{P}(D = 1|\text{parent}(X_i) = \text{parent}(x_i))}{\mathbb{P}(D = 1|\text{parent}(X_i) = \text{parent}(x_i))} \cdot \frac{\mathbb{P}(D = 1|X_i = x_i, \text{parent}(X_i) = \text{parent}(x_i))}{1 - \mathbb{P}(D = 1|X_i = x_i, \text{parent}(X_i) = \text{parent}(x_i))}
\end{aligned}$$

Thus, we learn a model to predict D from X_i, and a model to predict D from $[X_i; \text{parent}(X_i)]$, on the concatenated dataset. In practice, we learn these models on a 75% split of both the source and target data, and use the remaining 25% for Shapley value computation, which only requires inference on the trained models. Therefore, an upper limit on the number of weight models required is $2|\mathtt{C}_{\mathcal{D}}|$, though in practice, this number is often smaller as several nodes may have the same parents.

In the case where X_i is a root node, the expression becomes:

$$w_d = \frac{1 - \mathbb{P}(D = 1)}{\mathbb{P}(D = 1)} \cdot \frac{\mathbb{P}(D = 1|X_i = x_i)}{1 - \mathbb{P}(D = 1|X_i = x_i)}$$

Where we simply compute $P(D = 1)$ as the relative size of the provided source and target datasets.

C Properties of the Method

Under perfect computation of importance weights, the Shapley values resulting from the performance-change game have the following desirable properties.

Property 1. (Efficiency) $\sum_{d \in \mathtt{C}_{\mathcal{D}}} \text{Attr}(d) = \text{Val}(\mathtt{C}_{\mathcal{D}}) = \text{Perf}(\mathcal{D}^{\text{target}}) - \text{Perf}(\mathcal{D}^{\text{source}})$

By the efficiency property of Shapley values [29], we know that the sum of Shapley values equal the value of the all-player coalition. Thus, we distribute the total performance change due to the shift from source to target distribution to the shifts in causal mechanisms in the candidate set.

Property 2.1. (Null Player) $\mathcal{D}_d^{\text{source}} = \mathcal{D}_d^{\text{target}} \implies \text{Attr}(d) = 0.$

Property 2.2. (Relevance) Consider a mechanism d. If $\text{Perf}(\widetilde{\mathtt{C}} \cup \{d\}) = \text{Perf}(\widetilde{\mathtt{C}})$ for all $\widetilde{\mathtt{C}} \subseteq \mathtt{C}_{\mathcal{D}} \setminus d$, then $\text{Attr}(d) = 0$.

We can verify that our method gives zero attribution to distributions that do not shift between the source and target, and distribution shifts which do not impact model performance. First, we observe that in both cases, $\text{Val}(\widetilde{\mathcal{D}}) = \text{Val}(\widetilde{\mathcal{D}} \cup \{d\})$. For Property 2.1, this is because $\widetilde{\mathcal{D}} = \widetilde{\mathcal{D}} \cup \{d\}$ for any $\widetilde{\mathcal{D}} \subseteq \mathtt{C}_{\mathcal{D}}$ since the factor corresponding to d remains the same between source and target even when it is allowed to change as part of the coalition. For Property 2.2, this is clear from Eq. 3. By definition of Shapley value, $\text{Attr}(d) = 0$.

Property 3. (Attribution Symmetry) Let $\text{Attr}_{\mathcal{D}_1, \mathcal{D}_2}(d)$ denote the attribution to some mechanism d when $\mathcal{D}_1 = \mathcal{D}^{\text{source}}$ and $\mathcal{D}_2 = \mathcal{D}^{\text{target}}$. Then, $\text{Attr}_{\mathcal{D}_1, \mathcal{D}_2}(d) = -\text{Attr}_{\mathcal{D}_2, \mathcal{D}_1}(d)\ \forall d \in \mathtt{C}_{\mathcal{D}}$.

We overload $\text{Perf}_{src \to tar}(\widetilde{\mathtt{C}})$ for some coalition $\widetilde{\mathtt{C}}$ to denote $\text{Perf}(\widetilde{\mathcal{D}})$ where $\widetilde{\mathcal{D}}$ is given by Equation 1. Analogously, we denote $\text{Perf}_{tar \to src}(\widetilde{\mathtt{C}})$ to be $\text{Perf}(\widetilde{\mathcal{D}}')$ when $\widetilde{\mathcal{D}}'$ is given by

$$\widetilde{\mathcal{D}}' = \left(\prod_{i : \mathcal{D}_{X_i|\text{parent}(X_i)} \in \widetilde{\mathtt{C}}} \mathcal{D}^{\text{source}}_{X_i|\text{parent}(X_i)} \right) \left(\prod_{i : \mathcal{D}_{X_i|\text{parent}(X_i)} \notin \widetilde{\mathtt{C}}} \mathcal{D}^{\text{target}}_{X_i|\text{parent}(X_i)} \right)$$

Note that $\text{Perf}_{src \to tar}(\widetilde{\mathtt{C}})$ = $\text{Perf}_{tar \to src}(\mathtt{C}_{\mathcal{D}} \setminus \widetilde{\mathtt{C}})$ for all $\widetilde{\mathtt{C}} \subseteq \mathtt{C}_{\mathcal{D}}$.

We can use Equation 2 to rewrite the Shapley value equation as:

$$\begin{aligned}
\text{Attr}_{\mathcal{D}_1, \mathcal{D}_2}(d) &= \frac{1}{|\mathtt{C}_{\mathcal{D}}|} \sum_{\widetilde{\mathtt{C}} \subseteq \mathtt{C}_{\mathcal{D}} \setminus \{d\}} \binom{|\mathtt{C}_{\mathcal{D}}| - 1}{|\widetilde{\mathtt{C}}|}^{-1} \left(\text{Perf}_{src \to tar}(\widetilde{\mathtt{C}} \cup \{d\}) - \text{Perf}_{src \to tar}(\widetilde{\mathtt{C}}) \right) \\
&= \frac{-1}{|\mathtt{C}_{\mathcal{D}}|} \sum_{\widetilde{\mathtt{C}} \subseteq \mathtt{C}_{\mathcal{D}} \setminus \{d\}} \binom{|\mathtt{C}_{\mathcal{D}}| - 1}{|\widetilde{\mathtt{C}}|}^{-1} \left(\text{Perf}_{tar \to src}(\mathtt{C}_{\mathcal{D}} \setminus \widetilde{\mathtt{C}}) - \text{Perf}_{tar \to src}(\mathtt{C}_{\mathcal{D}} \setminus (\widetilde{\mathtt{C}} \cup \{d\})) \right) \\
&= \frac{-1}{|\mathtt{C}_{\mathcal{D}}|} \sum_{\widetilde{\mathtt{C}}' \subseteq \mathtt{C}_{\mathcal{D}} \setminus \{d\}} \binom{|\mathtt{C}_{\mathcal{D}}| - 1}{|\widetilde{\mathtt{C}}'|}^{-1} \left(\text{Perf}_{tar \to src}(\widetilde{\mathtt{C}}' \cup \{d\}) - \text{Perf}_{tar \to src}(\widetilde{\mathtt{C}}') \right) \\
&= -\text{Attr}_{\mathcal{D}_2, \mathcal{D}_1}(d)
\end{aligned}$$

Thus, the method attributes the overall performance change only to distributions that actually change in a way that affects the specified performance metric. The contribution of each distribution is computed by considering how much they impact the performance if they are made to change in different combinations alongside the other distributions.

D Shapley Values for A Synthetic Setting

D.1 Derivation

Suppose that we have the following data generating process for the source environment:

$$X \sim \mathcal{N}(\mu_1, \sigma_X^2)$$
$$Y \sim \theta_1 X + \mathcal{N}(0, \sigma_Y^2)$$

And for the target environment:

$$X \sim \mathcal{N}(\mu_2, \sigma_X^2)$$
$$Y \sim \theta_2 X + \mathcal{N}(0, \sigma_Y^2)$$

The model that we are investigating is $\hat{Y} = f(X) = \phi X$, and $l((x,y),f) = (y - f(x))^2$. Then,

$$\begin{aligned}
\text{Perf}(\mathcal{D}^{\text{source}}) &= \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{source}}}[l((x,y),f)] \\
&= \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{source}}}[\big(\theta_1 X + \mathcal{N}(0,\sigma_Y^2) - \phi X\big)^2] \\
&= \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{source}}}[\big(\mathcal{N}((\theta_1-\phi)\mu_1, (\theta_1-\phi)^2\sigma_X^2) + \mathcal{N}(0,\sigma_Y^2)\big)^2] \\
&= \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{source}}}[\big(\mathcal{N}((\theta_1-\phi)\mu_1, (\theta_1-\phi)^2\sigma_X^2 + \sigma_Y^2)\big)^2] \\
&= (\theta_1-\phi)^2\sigma_X^2 + \sigma_Y^2 + (\theta_1-\phi)^2\mu_1^2
\end{aligned}$$

$$\begin{aligned}
\text{Perf}(\mathcal{D}^{\text{target}}) &= \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{target}}}[l((x,y),f)] \\
&= (\theta_2-\phi)^2\sigma_X^2 + \sigma_Y^2 + (\theta_2-\phi)^2\mu_2^2
\end{aligned}$$

$$\begin{aligned}
\Delta &= \text{Perf}(\mathcal{D}^{\text{target}}) - \text{Perf}(\mathcal{D}^{\text{source}}) \\
&= \sigma_X^2((\theta_2-\phi)^2 - (\theta_1-\phi)^2) + (\theta_2-\phi)^2\mu_2^2 - (\theta_1-\phi)^2\mu_1^2 \\
&= \text{Val}(\mathtt{C}_{\mathcal{D}})
\end{aligned}$$

$$\begin{aligned}
\text{Val}(\{\mathcal{D}_X\}) &= (\theta_1-\phi)^2(\mu_2^2 - \mu_1^2) && (\theta_2 := \theta_1) \\
\text{Val}(\{\mathcal{D}_{Y|X}\}) &= (\sigma_X^2 + \mu_1^2)((\theta_2-\phi)^2 - (\theta_1-\phi)^2) && (\mu_2 := \mu_1)
\end{aligned}$$

$$\begin{aligned}
\text{Attr}(\mathcal{D}_X) &= \frac{1}{2}\big(\text{Val}(\mathtt{C}_{\mathcal{D}}) - \text{Val}(\{\mathcal{D}_{Y|X}\}) + \text{Val}(\{\mathcal{D}_X\}) - \text{Val}(\{\})\big) \\
&= \frac{1}{2}\big((\theta_2-\phi)^2(\mu_2^2-\mu_1^2) + (\theta_1-\phi)^2(\mu_2^2-\mu_1^2)\big) \\
&= (\frac{1}{2}\mu_2^2 - \frac{1}{2}\mu_1^2)((\theta_2-\phi)^2 + (\theta_1-\phi)^2)
\end{aligned}$$

$$\begin{aligned}
\text{Attr}(\mathcal{D}_{Y|X}) &= \frac{1}{2}\big(\text{Val}(\mathtt{C}_{\mathcal{D}}) - \text{Val}(\{\mathcal{D}_X\}) + \text{Val}(\{\mathcal{D}_{Y|X}\}) - \text{Val}(\{\})\big) \\
&= \frac{1}{2}\big((\sigma_X^2+\mu_2^2)((\theta_2-\phi)^2 - (\theta_1-\phi)^2) + (\sigma_X^2+\mu_1^2)((\theta_2-\phi)^2 - (\theta_1-\phi)^2)\big) \\
&= (\sigma_X^2 + \frac{1}{2}\mu_1^2 + \frac{1}{2}\mu_2^2)((\theta_2-\phi)^2 - (\theta_1-\phi)^2)
\end{aligned}$$

Note that $\text{Attr}(\mathcal{D}_X) + \text{Attr}(\mathcal{D}_{Y|X}) = \Delta$.

Using the method proposed by Budhathoki et al. [5], we get that:

$$\begin{aligned}
D(\tilde{P}_X || P_X) &= \frac{(\mu_2-\mu_1)^2}{2\sigma_X^2} \\
D(\tilde{P}_{Y|X} || P_{Y|X}) &= \mathbb{E}_{X\sim\tilde{P}_X}[D(\tilde{P}_{Y|X=x} || P_{Y|X=x})] \\
&= \mathbb{E}_{X\sim\tilde{P}_X}\left[\frac{((\theta_2-\theta_1)X)^2}{2\sigma_Y^2}\right] = \frac{(\theta_2-\theta_1)^2}{2\sigma_Y^2}(\sigma_X^2 + \mu_2^2)
\end{aligned}$$

Table D.1: Analytical expressions of the attributions for the simple synthetic case.

	Attr($\mathcal{D}_X$)	**Attr($\mathcal{D}_{Y\|X}$)**
Ours	$(\frac{1}{2}\mu_2^2 - \frac{1}{2}\mu_1^2)((\theta_2-\phi)^2 + (\theta_1-\phi)^2)$	$(\sigma_X^2 + \frac{1}{2}\mu_1^2 + \frac{1}{2}\mu_2^2)((\theta_2-\phi)^2 - (\theta_1-\phi)^2)$
Budhathoki et al. [5]	$\frac{(\mu_2-\mu_1)^2}{2\sigma_X^2}$	$\frac{(\theta_2-\theta_1)^2}{2\sigma_Y^2}(\sigma_X^2+\mu_2^2)$

We summarize the attribution of our method, along with the attribution using the joint method from Budhathoki et al. [5], in Table D.1. We highlight several advantages that our method has over the baseline.

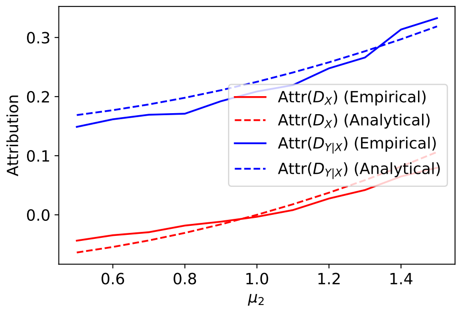
First, our attribution takes the model parameter ϕ into account in order to explain model performance changes, whereas Budhathoki et al. [5] do not, as they only explain shifts in (X, Y), or changes in simple functions such as $\mathbb{E}[X]$ of the variables. Second, we find that our Attr($\mathcal{D}_X$) is a function of θ_2. This is desirable, as covariate shift may compound with concept shift to increase loss non-linearly. This also ensures that both attributions always sum to the total shift. Third, we note that our attributions are *signed*, which is particularly important as some shifts may decrease loss. Finally, we note that our attributions are symmetric when the source and target data distributions are swapped by Property 3. This is not true of the baseline method in general, as the KL divergence is asymmetric. Since we assume knowledge of the true causal graph (which provides the factorization that determines the coalition), we also evaluate the attribution when the graph is misspecified. In this case, the coalition will consist of $\{\mathcal{D}_Y, \mathcal{D}_{X|Y}\}$. We include these attribution results in Figure D.2. In this case, as expected, both $\mathcal{D}_Y$ and $\mathcal{D}_{X|Y}$ are attributed the change in model performance (at varying levels depending on the magnitude of concept drift). While this is still a meaningful attribution, knowledge of the causal graph provides a more succinct interpretation of the behavior in the system.

D.2 Experiments

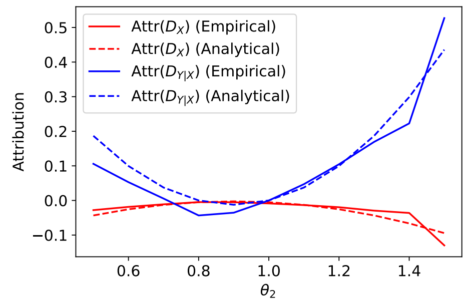
Now, we verify the correctness of our method by conducting a simulation of this setting, using $\mu_1 = 0$, $\theta_1 = 1$, $\sigma_X^2 = 0.5$, $\sigma_Y^2 = 0.25$, $\phi = 0.9$, and varying μ_2 (the level of covariate shift), and θ_2 (the level of concept drift). We generate $10,000$ samples from the source environment, and, for each setting of μ_2 and θ_2, we generate $10,000$ samples from the corresponding target environment. We then apply our method to attribute shifts to $\{\mathcal{D}_X, \mathcal{D}_{Y|X}\}$, using `XGB` to estimate importance weights. We also apply the joint method in Budhathoki et al. [5].

In Figure D.1, we compare our attributions with the baseline, when both covariate and concept drift are present. We find that for our method, the empirical results match with the previously derived analytical expressions, where any deviations can be attributed to variance in the importance weight computations. For Budhathoki et al. [5], we find that there appears to be very high variance in the attribution the attribution to $\mathcal{D}_{Y|X}$, which is likely a product of the nearest-neighbors KL estimator [41] used in their work.

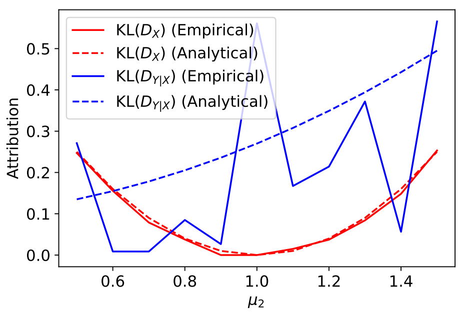
In Figure D.2, we explore the case where we have a misspecified causal graph. Specifically, we examine the case where only concept drift is present, for the actual graphical model ($\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_X, \mathcal{D}_{Y|X}\}$), and for a misspecified graphical model ($\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_Y, \mathcal{D}_{X|Y}\}$). We find that using the mechanisms from the true data generating process results in a *minimal* attribution (i.e. Attr($\mathcal{D}_X$) $= 0$), whereas the the misspecified causal graph gives non-zero attribution to both distributions.



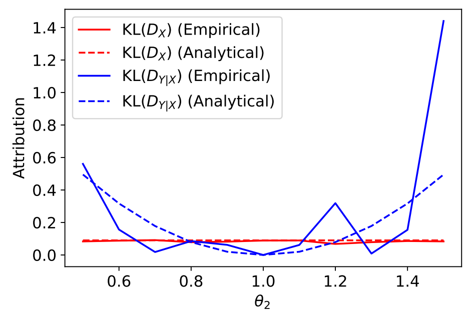
(a) Our method; Fix $\theta_2 = 1.3$ and vary μ_2.



(b) Our method; Fix $\mu_2 = 0.7$ and vary θ_2.

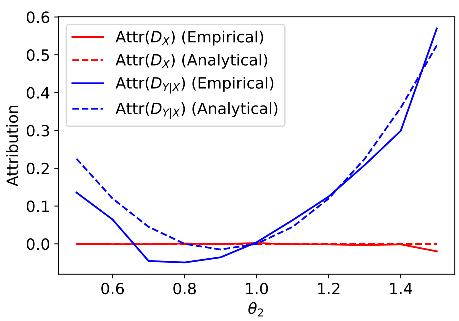


(c) Joint method from Budhathoki et al. [5]; Fix $\theta_2 = 1.3$ and vary μ_2.

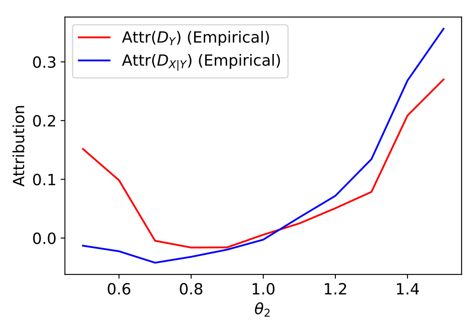


(d) Joint method from Budhathoki et al. [5]; Fix $\mu_2 = 0.7$ and vary θ_2.

Figure D.1: Mean squared error differences attributed by our model and Budhathoki et al. [5] in the synthetic setting described in Appendix D



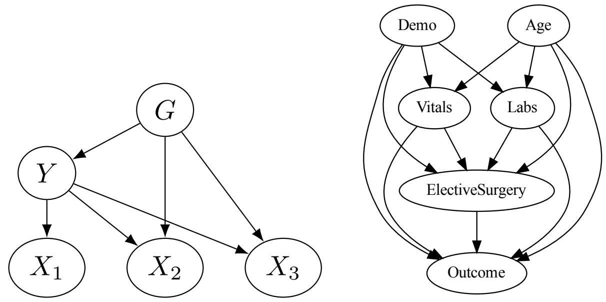
(a) Our method; Fix $\mu_2 = 1$ and vary θ_2, with $\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_X, \mathcal{D}_{Y|X}\}$, the actual causal graph



(b) Our method; Fix $\mu_2 = 1$ and vary θ_2, with $\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_Y, \mathcal{D}_{X|Y}\}$, a mis-specified causal graph

Figure D.2: Mean squared error differences attributed by our model when there is only concept drift, for the actual causal graph (a), and a mis-specified causal graph (b).

E Additional Experimental Results



(a) Synthetic data (b) eICU data

Figure E.3: Causal graphs for synthetic and eICU data

E.1 Synthetic Data

Setup. We generate a synthetic binary classification dataset with five variables according to the following data generating process, corresponding to the causal graph shown in Figure E.3a. Here, $\xi_p : \{0,1\} \to \{0,1\}$ is a function that randomly flips the input with probability p.

$$
\begin{aligned}
&G \sim Ber(0.5), && Y = \xi_q(G), && X_1 = \mathcal{N}(\omega \xi_{0.25}(Y), 1) \\
&X_2 = \mathcal{N}(\xi_{0.25}(Y) + G, 1) && X_3 = \mathcal{N}(\xi_{0.25}(Y) + \mu G, 1)
\end{aligned}
$$

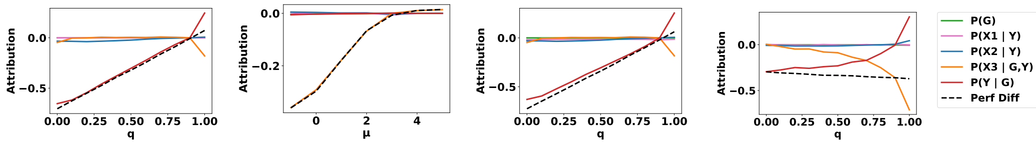
Where q, ω and μ are parameters of the data generating process. Here, G represents a spurious correlation [4, 2] that is highly correlated with Y, and is easily inferred from (X_2, X_3). By selecting a large value for q (the spurious correlation strength) on the source environment, we can create a dataset where models rely more heavily on using X_2 and X_3 to infer G and then Y, instead of infering $\xi_{0.25}(Y)$ across the three features to estimate Y directly.

In the source environment, we set $q = 0.9, \omega = 1$ and $\mu = 3$. We generate 20,000 samples using these parameters, and train logistic regression (`LR`) and XGBoost (`XGB`, [6]) models on (X_1, X_2, X_3) to predict Y, using 3-fold cross-validation to select the best model. We attribute performance changes for this model using the proposed method. We explore four data settings for the target environment:

(a) Label Shift: Vary $q \in [0, 1]$. Keep ω and μ at their source values. Only $P(Y|G)$ changes. This represents a label shift for the model across domains (which does not have access to G).
(b) Covariate Shift: Vary $\mu \in [0, 5]$. Keep q and ω at their source values. Only $P(X_3|G, Y)$ changes across domains.
(c) Combined Shift 1: Set $\omega = 0$ in the target environment and vary $q \in [0, 1]$. Keep μ at its source value. Both $P(X_1|Y)$ and $P(Y|G)$ change across domains, but the shift should be largely attributed to $P(Y|G)$ as the model relies on this correlation much more than X_1.
(d) Combined Shift 2: Set $\mu = -1$ in the target environment. Further, vary $q \in [0, 1]$. Keep ω at its source value. Both $P(X_3|Y)$ and $P(Y|G)$ change across domains, but their specific contribution to model performance degradation is not known exactly.

We use our method to explain performance changes in accuracy and Brier score for each model on target environments generated within each setting (with $n = 20,000$), computing density ratios using `XGB` models. Note that the causal graph shown in Figure E.3a implies five potential distribution in the candidate set: $\mathtt{C}_{\mathcal{D}} = \{\mathcal{D}_G, \mathcal{D}_{Y|G}, \mathcal{D}_{X_1|Y}, \mathcal{D}_{X_2|G,Y}, \mathcal{D}_{X_3|G,Y}\}$.

Our method correctly identifies distribution shifts. We focus on the output of our method with `LR` as the model of interest and accuracy as the metric in Figure E.1. We find that our method attributes all of the performance changes to the correct ground truth shifts, both when there is a single shift (Settings (a) and (b)) and when there are multiple shifts (Settings (c) and (d)). In the case of Setting (c), we find that our method attributes all of the performance drop to a shift in $P(Y|G)$. This is because the model relies largely on the spurious information (G inferred from X_2 and X_3) in the source environment. We verify this by examining the overall feature importance for both models (see Table E.2 in Appendix for details). Further, in the presence of multiple shifts which simultaneously impact model performance (Setting (d)), we find that our method is able to attribute a



(a) Label Shift (b) Covariate Shift (c) Combined Shift 1 (d) Combined Shift 2

Figure E.1: Attributions by our model for the change in accuracy to five potential distributional shifts on the synthetic dataset for the LR model. Further from 0 implies higher (signed) attribution We observe that the overall change (Perf Diff) is attributed to the true shift(s) in all cases. All attributions sum to the true performance change by Property 1.

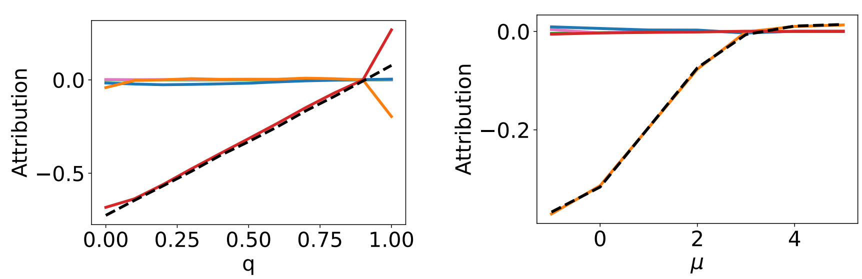
meaningful fraction of the performance shift to each distribution. We further demonstrate that our method correctly identifies distribution shifts (and attributions) for a CelebA gender classification task in Appendix E.2.

Table E.1: Performance of each model on the source environment for the synthetic dataset.

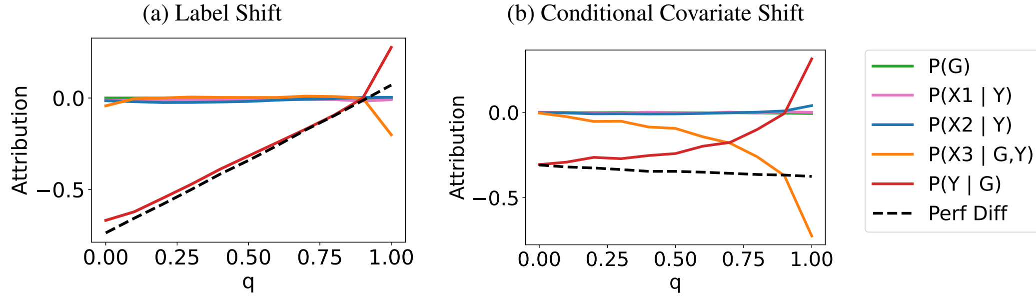
	Accuracy	Brier Score
`LR`	0.871	0.102
`XGB`	0.870	0.099

Table E.2: Feature importances of each model on the synthetic dataset. For LR, the model coefficient is shown, and for XGB, the total information gain from each feature.

	`LR` (Coefficient)	`XGB` (Gain)
X_1	0.400	31.1
X_2	0.381	29.2
X_3	1.994	358.2

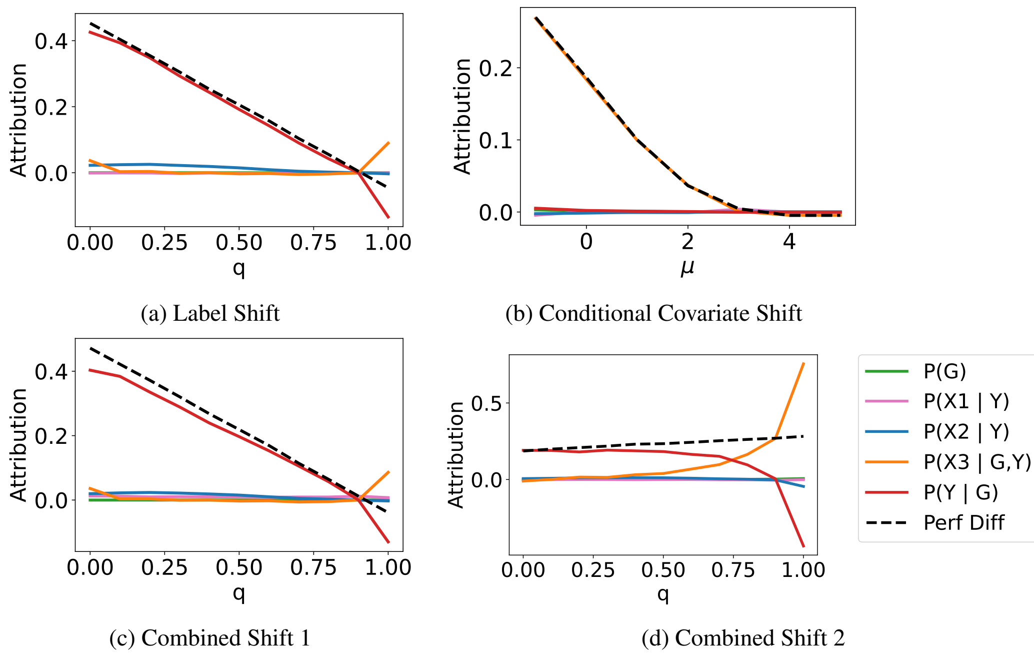


(a) Label Shift (b) Conditional Covariate Shift



(c) Combined Shift 1 (d) Combined Shift 2

Figure E.2: Accuracy differences attributed by our method to five potential distributional shifts on the synthetic dataset for the XGB model.



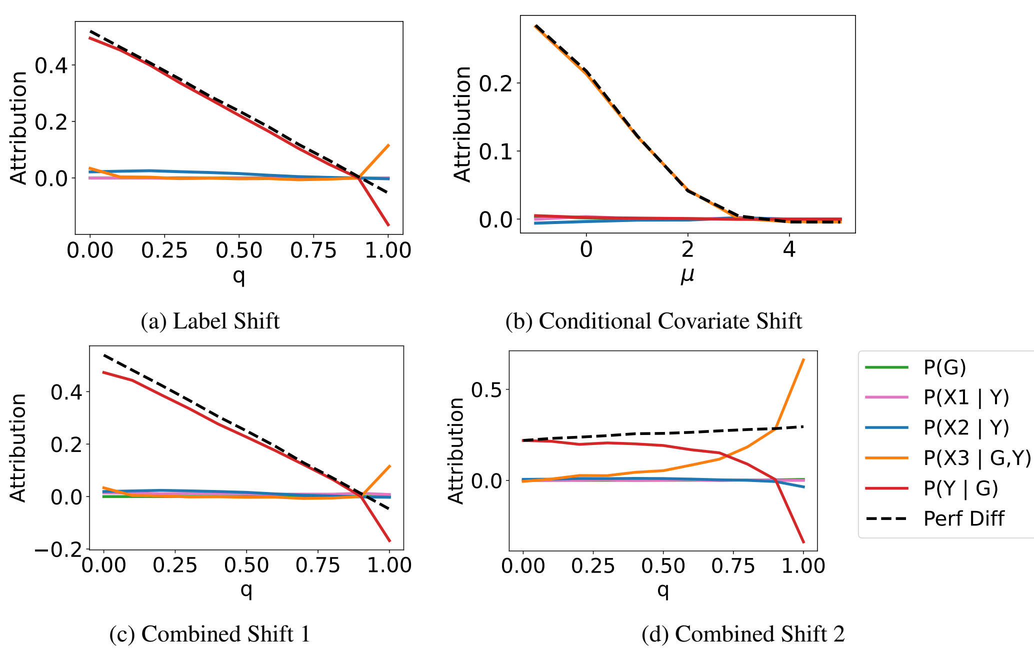
(a) Label Shift

(b) Conditional Covariate Shift

(c) Combined Shift 1

(d) Combined Shift 2

Figure E.3: Brier score differences attributed by our method to five potential distributional shifts on the synthetic dataset for the `LR` model.



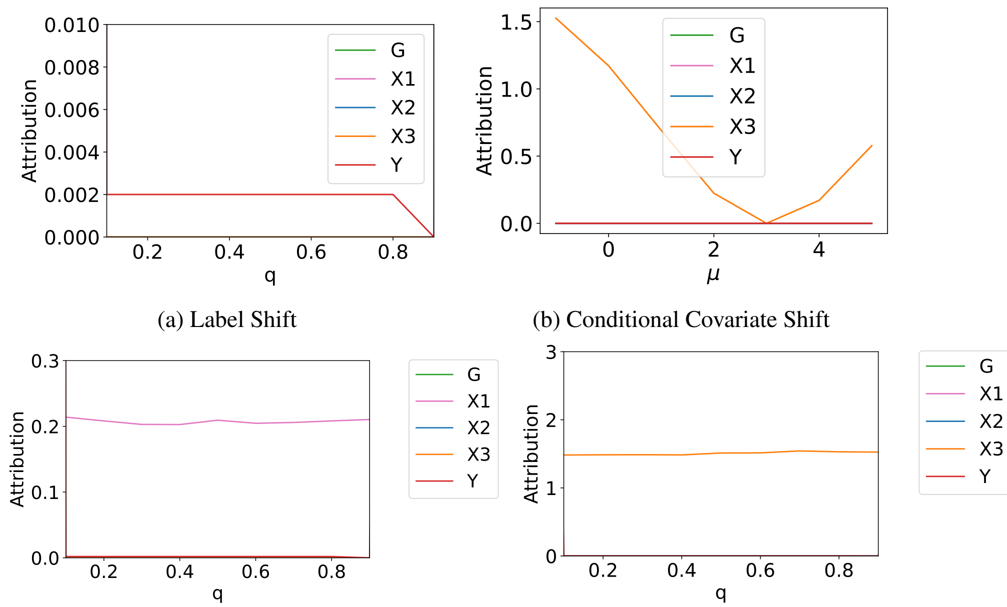
(a) Label Shift

(b) Conditional Covariate Shift

(c) Combined Shift 1

(d) Combined Shift 2

Figure E.4: Brier score differences attributed by our method to five potential distributional shifts on the synthetic dataset for the `XGB` model.



(a) Label Shift (b) Conditional Covariate Shift

(c) Combined Shift 1 (d) Combined Shift 2

Figure E.5: Attributions by the joint method in Budhathoki et al. [5] to five potential distributional shifts on the synthetic dataset. We note that the magnitude of the attribution is not informative in interpreting model performance changes, particularly when multiple shifts are present.

E.2 Gender Classification in CelebA

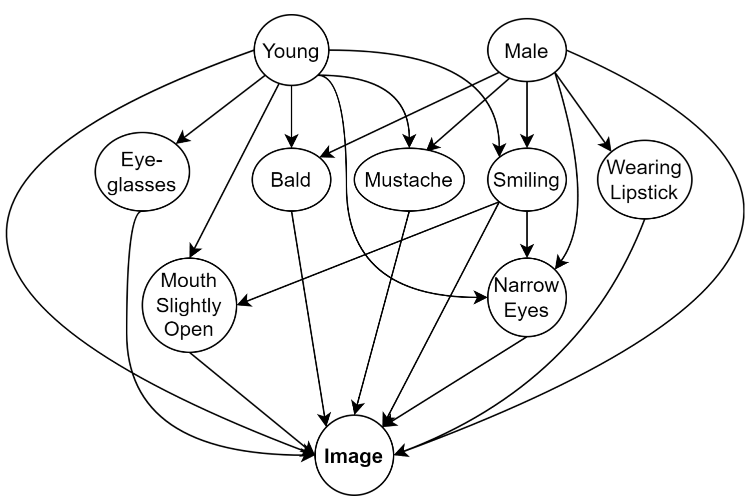


Figure E.6: Causal graph for the celebA dataset.

Setup. We use the CelebA dataset [20], where the goal is to predict gender from facial images. We adopt a setup similar to the one presented in Thams et al. [36]. We assume this data is generated from the causal graph shown in Figure E.6. We train a CausalGAN [16], a generative model that allows us to synthesize images faithful to the graph. CausalGAN allows to train attribute nodes (young, bald, etc) which are binary-valued, and then synthesize images conditioned on specific attributes. This allows us to simulate known distribution shifts (in attributes and hence images) across environments. We assume that the causal mechanisms in the source environment have log-odds equal to the ones shown in Table E.3. We omit $\mathcal{D}_{\text{Image}|\text{Pa(Image)}}$ from $\mathtt{C}_{\mathcal{D}}$, as 1) this distribution is parameterized by the CausalGAN and does not change, and 2) it is high-dimensional and difficult to work with. We investigate attribution to distribution shift of an ImageNet-pretrained ResNet-18 [13] finetuned to predict gender from the image using frozen representations. Note that the model is only given access to the image itself, but not any of the binary attributes in the causal graph. We conduct the following two experiments for evaluation.

Experiment 1. The purpose of this experiment is to demonstrate that our method provides the correct attributions for a wide range of random shifts. To create the target environment, we first select the number of mechanisms to perturb, $n_p \in \{1, 2, ..., 6\}$. We select n_p mechanisms from the causal graph, which we define as the ground truth shift. For each mechanism, we perturb one of the log odds by a quantity uniformly selected from $[-2.0, -1.0] \cup [1.0, 2.0]$. We then use the CausalGAN to simulate a dataset of $10,000$ images based on the modified mechanisms, and use our method to attribute the accuracy change between source and target. We select the n_p distributions from our method with the largest attribution magnitude, and compare this set with the set of ground truth shifts to calculate an accuracy score. We repeat this experiment 20 times for each value of $n_p \in \{1, 2, ..., 6\}$, and only select experiments with a non-trivial change in model performance (change in accuracy $\geq 1\%$).

Experiment 2. The purpose of this experiment is to investigate the magnitude of our model attributions in the presence of multiple shifts. We perturb the log odds for $P(\text{Wearing Lipstick}|\text{Male})$ and $P(\text{Mouth Slightly Open}|\text{Smiling})$ jointly by $[-3.0, 3.0]$. We compare the magnitude of the attributions for the two associated mechanisms, relative to the total shift in accuracy.

Table E.3: Data generating process for the causal graph shown in Figure E.6

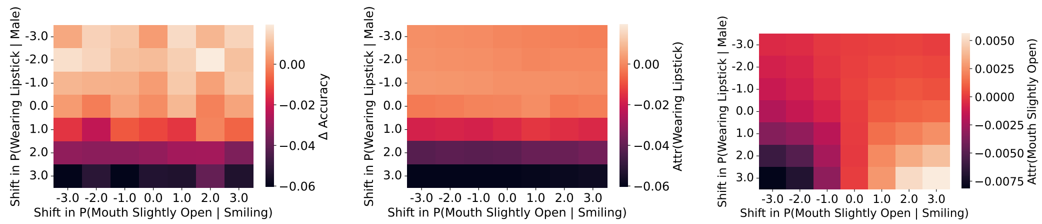
Variable	**Log Odds**
Young	Base: 0.0
Male	Base: 0.0
Eyeglasses	Base: 0.0, Young: -0.4
Bald	Base: -3.0, Male: 3.5, Young: -1.0
Mustache	Base: -2.5, Male: 2.5, Young: 0.5
Smiling	Base: 0.25, Male: -0.5, Young: 0.5
Wearing Lipstick	Base: 3.0, Male: -5.0
Mouth Slightly Open	Base: -1.0, Young: 0.5, Smiling: 1.0
Narrow Eyes	Base: -0.5, Male: 0.3, Young: 0.2, Smiling: 1.0

Table E.4: Average accuracy of our method in attributing shifts to the ground truth shift in CelebA for each number of perturbed mechanisms (n_p).

n_p	Avg Accuracy
1	1.00 ± 0.00
2	0.72 ± 0.36
3	0.90 ± 0.16
4	0.85 ± 0.13
5	0.93 ± 0.10
6	0.91 ± 0.09

Table E.5: Predictive performance of XGB models trained to predict attributes from the source environment in CelebA, and the correlation of each attribute the gender label, as measured by the Matthews Correlation Coefficient (MCC).

	Predictive Performance		Correlation
	AUROC	AUPRC	MCC
Wearing Lipstick	0.968	0.976	-0.837
Mouth Slightly Open	0.927	0.924	-0.036



(a) (b) (c)

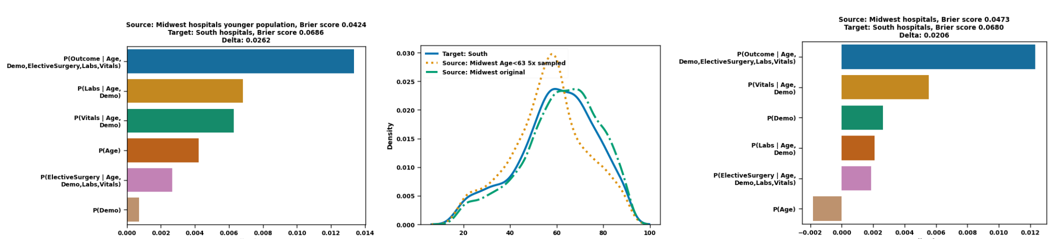
Figure E.7: We vary the perturbation in log odds in the target environment for the "wearing lipstick" and "mouth slightly open" attributes. We show (a) the total shift in accuracy, (b) our attribution to $P(\text{Wearing Lipstick}|\text{Male})$, (c) our attribution to $P(\text{Mouth Slightly Open}|\text{Young, Smiling})$.

Results. In Table E.4, we show the average accuracy of our method for each value of n_p. We find that our method achieves roughly 90% accuracy at this task. However, we note that this is not the ideal scenario to validate our method, as not all shifts in the ground truth set will result in a decrease in the model performance. As our method will not attribute a significant value to shifts which do not impact model performance, this explains the accuracy discrepancy observed.

In Figure E.7, we show the output of our method in Experiment 2. First, we find that shifting these two attributes causes a large decrease in the accuracy (up to 6%), and that $P(\text{Wearing Lipstick}|\text{Male})$ seem to be the stronger factor responsible for the decrease. Looking at our attributions, we find that we indeed attribute the large majority of the shift to $P(\text{Wearing Lipstick}|\text{Male})$. Here, the relative attribution to $P(\text{Wearing Lipstick}|\text{Male})$ is relatively unaffected by the shift in the other variable, as its effect on the total shift is so minuscule. However, looking at the attribution to $P(\text{Mouth Slightly Open}|\text{Young, Smiling})$, in addition to the small magnitude, we do observe an interesting effect, where the attributed accuracy drop is greater when the two shifts are combined.

To justify the magnitude of our attributions, we use an ad-hoc heuristic that attempts to approximate the model reliance on each attribute in making its prediction. First, we train XGBoost models on the ResNet-18 embeddings from the source environment to predict the two attributes. From Table E.5, we find that "Wearing Lipstick" is easier to infer from the representations than "Mouth Slightly Open". Next, we measure the correlation of each attribute to the label (gender), finding that the magnitude of the correlation is also much higher for "Wearing Lipstick". As "Wearing Lipstick" is both easier to detect from the image, and is also a stronger predictor of gender, it seems reasonable to conclude that the model trained on the source would utilize it more in its predictions, and thus our method should attribute more of the performance drop to the "Wearing Lipstick" distribution when it shifts.

E.3 eICU Experiment



(a) Attribution with resampled source (b) Shifted age distribution (c) Attribution with balanced age

Figure E.8: Attributing Brier score differences to candidate distributions on the eICU dataset for an `XGB` model trained on either (a) resampled or (c) balanced Midwest, and tested on South datasets.

Table E.6 lists the features that comprise the nodes in the causal graph. Please refer to [31, Supporting Information Table C] for descriptions. Code for preprocessing the eICU database for the mortality prediction task is made available at `https://github.com/alistairewj/icu-model-transfer` by Johnson et al. [15].

Total number of data points are 10,056 in Midwest and 7,836 in South datasets. Both of them have 20 features and a binary outcome. We randomly split both datasets into two halves for training the

Table E.6: Features comprising the nodes of the causal graph in Figure E.3b.

Variable	**Features**
Demo	is_female, race_black, race_hispanic, race_asian, race_other
Vitals	heartrate, sysbp, temp, bg_pao2fio2ratio, urineoutput
Labs	bun, sodium, potassium, bicarbonate, bilirubin, wbc, gcs
Age	age
ElectiveSurgery	electivesurgery
Outcome	death

XGBoost model (also, for estimating the Shapley values) and evaluation. To create the resampled Midwest dataset, we subsample 67% of the training set but selectively sample records with age less than 63 (which is the median age in Midwest dataset) with probability 5 times that of the probability of sampling the rest of the records.