On the Relationship Between Explanation and Prediction: A Causal View

Amir-Hossein Karimi [1 2 3] **Krikamol Muandet** [1] **Simon Kornblith** [3] **Bernhard Schölkopf** [1] **Been Kim** [3]

Abstract

Explainability has become a central requirement for the development, deployment, and adoption for machine learning (ML) models and we are yet to understand what explanation methods can and cannot do. Several *factors* such as data, model prediction, hyperparameters used in training the model, and random initialization can all influence downstream explanations. While previous work empirically hinted that explanations (E) may have little relationship with prediction (Y), there is lack of conclusive study to quantify this relationship. Our work borrows tools from causal inference to systematically assay this relationship. More specifically, we measure the relationship between E and Y by measuring treatment effect when intervening on their causal ancestors (hyperparameters) (inputs to generate saliency-based Es or Ys). We discover that Y's relative direct influence on E follows an odd pattern; the influence is higher in the lowest performing models than mid-performing models, and it then decreases in the top performing of models. We believe our work is a promising first step towards providing better guidance for practitioners who can make more informed decisions in utilizing these explanations by knowing what factors are at play and how they relate to their end-task.

1. Introduction and related work

Being able to provide explanations has become one of the central topics in ML, not only to better understand a model's underlying rationale but also to comply with regulatory requirements (Parliament & of the European Union, 2016), control (Koh et al., 2020; Bau et al., 2020; Meng et al., 2022) or debug a model (Adebayo et al., 2022; Rieger et al., 2020). Although ML researchers have developed many interpretability tools, these tools have elicited pointed criticisms, often highlighting computational or qualitative user-study-based evidence that explanations generated from these tools must be used with care (Poursabzi-Sangdeh et al., 2018; Chu et al., 2020; Adebayo et al., 2018; Alqaraawi et al., 2020; Srinivas & Fleuret, 2021; Kindermans et al., 2019).

In particular, the relationship between explanations (E) and predictions (Y) has been of interest for many investigations. Some argue that since many explanation methods claim to reveal a model's rationale behind its *decision* (Y), the relationship between E and Y must be 'strong' (e.g., when Y changes significantly, E must do so as well) (Adebayo et al., 2018; Srinivas & Fleuret, 2021), while others argue that E should also reflect other factors such as data distribution in addition to Y (e.g., data points). Empirically, it has been shown that explanations from an untrained model and a trained model can be visually and statistically indistinguishable (Adebayo et al., 2018). Theoretically, it can be proven that E has no relation to Y in some cases (Nie et al., 2018; Srinivas & Fleuret, 2021). However, it remains an open question to quantitatively validate the relationship between E and Y when potential confounding factors change (e.g., hyperparameters, datasets).

In this work, we seek to formalize this relationship, inspired by the common cause principle of (Reichenbach, 1956) that states that, if two variables are *statistically* dependent, there must be a common *cause* influencing both of them, and this common cause can be chosen such that it explains all the dependence. We develop a measure of dependence via the Potential Outcomes framework (Rubin, 2005). Viewed through a lens of causality, we evaluate the treatment effect of hyperparameters, H (i.e., H taking on value h', the counterfactual antecedent) on E and Y conditioned on a particular instance x. In other words, by measuring the treatment effect of each hyperparameter, we are measuring its influence on E and Y, and in particular, how the influence is *different or similar* in E and Y (see Figure 1; left). Furthermore, under a careful evaluation, we tease apart the direct influence of H on E vs. its indirect influence mediated through Y to better understand the flow of causation.

Note that we may not be able to interpret the observed

This work was primarily conducted when the first author was interning at Google Brain. [1]MPI for Intelligent Systems [2]ETH Zurich [3]Google Research, Brain Team. Correspondence to: Amir-Hossein Karimi <amir@tue.mpg.de>.

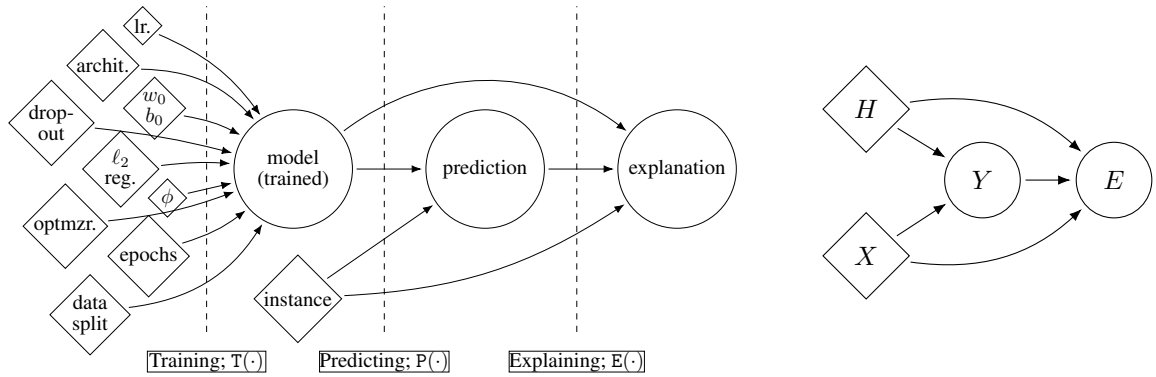

Figure 1: Explanation generating process involve three stages (training, predicting and explaining) (left). Intervening on factors (diamond nodes, e.g., H, X) allow for studying the treatment effect (i.e., causal influence) of factors on down-stream targets (i.e., Y, E) (right).

difference between predictions and explanations that arise from two different hyperparameters as a causal effect unless *ceteris paribus*, i.e., all else being equal, is fulfilled. Retraining almost identical neural networks with all possible values of hyperparameters is however computationally prohibitive and environmentally unsustainable. Instead, we perform an observational study on the model zoos, a large collection of pre-trained models (Unterthiner et al., 2020; Jiang et al., 2019), to study the relationship between E and Y.

Why hyperparameters as treatments? Under a fixed random seed, hyperparameters (e.g., choice of activation, initialization, training budget) are arguably the only reasonable causal ancestor of the model because they fully determine the weights of the resulting model and behavior thereof. They are also known to influence the inherent tendencies/performances of the model. Models trained on completely different hyperparameters could perform similarly under one metric (e.g., training loss), but have completely different task-specific performance (e.g., fairness) (D'Amour et al., 2020). One can also use the hyperparameters alone to predict the final performance of the models (Unterthiner et al., 2020) or even use model's weights to predict hyperparameters (Eilertsen et al., 2020).

Despite some methodological similarities, our work is fundamentally different from using causal inference to *generate* counterfactual explanations (e.g., Wachter et al. (2017)), where intervention is on the subset of features in an instance, rather than on a causal ancestor of E while keeping the dataset constant. Our goal is to study the relation between Y and E, and not to generate explanations.

Our study reveals surprising and unintuitive patterns of the relationship between E and Y (precisely, measured by how a causal ancestor of the two influences them). In particular, in top performing models, the influence on E from Y *decreases* compared to relatively lower performing models. For some methods, a causal ancestor of both Y and E directly influence E much more than Y, leaving Y's influence on E minimal, even though this ancestor, i.e, hyperparameter, should not inform the explanation of the model in any way. This finding was consistent across 30k pre-trained models with different hyperparameters across different datasets. We hope our work informs practitioners to decide what these explanations can and cannot be used for, and other precautions that must be in place before they use explanations for decision making.

2. Methodology

To understand the impact of various factors, we perform an exploratory analysis on a class of ML models and then analyze their causal effects on the downstream explanations.

Notation Let $X \in \mathcal{X} \subseteq \mathbb{R}^d$ be a random variable representing a data instance and $H \in \mathcal{H}$ a random variable representing a hyperparameter vector. For $x \in \mathcal{X}$ and $h \in \mathcal{H}$, let $Y_h^*(x)$ and $E_h^*(x)$ be random variables representing respectively prediction and explanation associated with the hyperparameter value h and data instance x. That is, $Y_h^*(x)$ and $E_h^*(x)$ correspond to the prediction and explanation when the model, trained with the hyperparameter vector $H = h$, is applied on the data point $X = x$. Put differently, the outcomes $Y_h^*(x)$ and $E_h^*(x)$ are realized by assigning the treatment (or intervention) $H = h$ (and the associated model) to the individual data $X = x$. The observed values of the prediction and explanation will be denoted by $\hat{y}_h(x)$ and $\hat{e}_h(x)$, respectively.

2.1. Explanation generating process

At a high level, the *explanation generating process* (EGP) shown in Figure 1 describes a mechanical system that is engineered to train an ML model given an initial set of hyperparameters, h, which then yields a local prediction[1] $\hat{y}_h(x)$ and an explanation $\hat{e}_h(x)$ given a test instance x. Formally,

an ML model is obtained through a *training procedure* $\mathtt{T} : \mathcal{H} \times \mathcal{D} \to \mathcal{F}$ given a set of training hyperparameters and a dataset $\mathcal{D} := (\mathcal{X}, \mathcal{Y})$. The training procedure is composed mainly of initialization, optimization, and regularization strategies. The trained model is then able to predict the target of a given test instance x via a *prediction procedure* $\mathtt{P} : \mathcal{F} \times \mathcal{X} \to \mathcal{Y}$. Finally, local explanations e are the result of an *explanation procedure* $\mathtt{E} : \mathcal{F} \times \mathcal{X} \times \mathcal{Y} \to \mathcal{E}$ applied to a tuple of a trained model, test instance, and predicted target, $\hat{y}_h(x)$. Note the absence of noise variables; under a fixed random seed, the procedures above yield outputs deterministically. Figure 1 depicts the overall procedure.

Although the procedures above may not be expressible in closed-form, e.g., one may not conclusively infer the trained weights of a neural network by only looking at the hyperparameters, each procedure is executable on a computer, e.g., the model weights can be obtained by training procedure under a training setting and given budget.

2.2. Potential Outcomes framework

To study the causal effects of hyperparameters, we adopt the Potential Outcomes (PO) framework (Rubin, 2005). Given the temporal precedence of hyperparameters over the trained model parameters and in turn over the prediction and explanation, one may alternatively view the mechanical system in Figure 1a as the causal system shown in Figure 1b (with graphical and structural components). In this framing, the *causal influence* of up-stream factors (e.g., H, X) on down-stream targets (e.g., Y, E) can be measured as the *treatment effect* of a factor (e.g., treatment $H = h$ vs. control $H = h'$), on the down-stream target.

In what follows, we will refer to $Y_h^*(x)$ and $E_h^*(x)$ as *potential* prediction and explanation on an instance x when the model is trained with the hyperparameter h. For any pair $h, h' \in \mathcal{H}$, the individual treatment effect (ITE), which quantifies the treatment effect of assigning two different parameters, can be defined as

$$\text{ITE}_Y(x) = Y_h^*(x) - Y_{h'}^*(x). \tag{1}$$

Similarly defined, the treatment effect for explanation is denoted as ITE_E. In principle, it is possible to realize $Y_h^*(x)$ and $E_h^*(x)$ for all $h \in \mathcal{H}$ given unlimited computational resources. As a result, one can evaluate $\text{ITE}(x)$ in practice by contrasting the predictions of models trained on hyperparameters h and h'. However, when this process becomes computationally prohibitive, we might face the so-called *fundamental problem of causal inference*, i.e., for each $x \in \mathcal{X}$, we can only observe $Y_h^*(x)$ and $E_h^*(x)$ for a small number of hyperparameters h, but not the other $h' \neq h$; see Section 2.3 for further discussion.

Since our research question seeks to investigate the impact of *multiple, potentially-non-binary* treatments (e.g., set of numerical and categorical `hparams`) on the target prediction/explanation (see Figure 1a), we amend the treatment definitions above as follows:

$$Y_{h=1}^*(x) - Y_{h=0}^*(x)$$
effect of $h = 1$ w.r.t $h = 0$ on $x \in X$
(single binary treatment) (2)

$$\mathbb{E}_{m \neq n}\left[Y_{h=n}^*(x) - Y_{h=m}^*(x)\right]$$
effect of $h = n$ w.r.t $h \neq n$ on $x \in X$
(single non-binary treatment) (3)

$$\mathbb{E}_{h_{\setminus i}}\left[\mathbb{E}_{m \neq n}\left[Y_{[h_i=n, h_{\setminus i}]}^*(x) - Y_{[h_i=m, h_{\setminus i}]}^*(x)\right]\right]$$
effect of $h_i = n$ w.r.t $h_i \neq n$ on $x \in X$
(multiple non-binary treatments) (4)

which allows for answering queries of the form "*what is the treatment effect of optimizer choice ν_1 as opposed to ν_2 on the local prediction of x*?". Were the optimizer choice, ν, to be the only hyperparameter in the system, this query would be answered by (3). In the setting of Figure 1a, however, (4) is employed to additionally marginalize out the effect of other hyperparameters. Although these expressions average over multiple `hparam` settings, they all refer to the prediction of the same individual (ITE); extensions to CATE and ATE, aggregated over $x \sim \mathcal{X}$, follow naturally.

To give (2), (3), and (4) a causal interpretation, the following assumption is required.

Assumption 2.1 (Full exchangeability)**.** $Y_h^* \perp\!\!\!\perp H$ and $E_h^* \perp\!\!\!\perp H$ for all $h \in \mathcal{H}$.

For example, random assignment of h within a given range of values h makes $Y_h^* \perp\!\!\!\perp H$ and $E_h^* \perp\!\!\!\perp H$. Although the treatment effects are identifable, evaluating them is computationally expensive. To understand why, it helps to illustrate a parallel with the setting of counterfactual explanations (Wachter et al., 2017). Whereas the treatment effects in our setting (see Equation (1)) contrasts $Y_h^*(x)$ and $Y_{h'}^*(x)$, the work of Wachter et al. (2017) contrasts $Y_h^*(x)$ and $Y_h^*(x')$. Unlike the latter which only requires the invocation of the *predicting procedure* given a new `instance` (e.g., a forward pass through a neural network), the former invokes the *training procedure* given a new `hparam` setting (i.e., a full re-training). In practice, computing power is limited and we may only have access to the predictions under a single model, say, $Y_h^*(x)$ and it can be prohibitively expensive to produce the prediction under a different model, $Y_{h'}^*(x)$, especially for large neural networks.

Note that the full exchangeability condition in Assumption 2.1 involves the "counterfactual" prediction Y_h^* and explanation E_h^* rather than the "observed" counterparts Y_h and E_h. The counterfactual variables Y_h^* and E_h^* describe

[1] We use the word "local" to refer to predictions and explanations for one `instance`, x, as opposed to "global" explanations at the population or model level.

the prediction and explanation one would observe had all instances in the entire population received the hyperparameters h as a treatment. Therefore, while in general, $Y_h(x) = Y_h^*(x)$ and $E_h(x) = E_h^*(x)$ can be random as well, e.g., if there is an exogenous noise, in our setting they are deterministic and randomness in the system arises only from the distribution of X (sampled from some dataset). As an analogy, imagine a treatment assigned to a patient: an individual outcome $Y_h^*(x)$ for each patient x and the population outcome Y_h^* can both be random, but the former (randomness in $Y_h^*(x)$) is missing in our setting.

Kernelized treatment effect (KTE) In addition to non-binary treatments, our work studies the effect of treatments on non-binary target variables ($Y_h^*(x)$ and $E_h^*(x)$) with dimensionality higher than that typically studied in the literature. For example, when x is an image of size $d_1 \times d_2$, $E_h^*(x) \in \mathbb{R}^{d_1 \times d_2}$. This means that (2) will yield a treatment effect *map* as opposed to a *scalar* treatment effect. In order to compare the relative effect of hyperparameters in various settings, we extend the standard definitions of treatment effects once again, by replacing the subtraction operator in (2) with an alternative notion of dissimilarity between counterfactuals, i.e.,

$$\begin{aligned}\|\phi(Y_h^*(x)) - \phi(Y_{h'}^*(x))\|_{\mathcal{G}}^2 &= k(Y_h^*(x), Y_h^*(x)) \\ &\quad - 2k(Y_h^*(x), Y_{h'}^*(x)) \\ &\quad + k(Y_{h'}^*(x), Y_{h'}^*(x))\end{aligned} \tag{5}$$

where $\phi : \mathcal{Y} \to \mathcal{G}$ is the canonical feature map associated with a positive definite kernel $k : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, i.e., $k(y, y') = \langle \phi(y), \phi(y') \rangle_{\mathcal{G}}$ for $y, y' \in \mathcal{Y}$, and $\mathcal{G}$ is a reproducing kernel Hilbert space (RKHS) associated with the kernel k; see, e.g., Schölkopf & Smola (2002) for detailed exposition on kernel methods. Similar extensions can be applied to explanations as well as to (3) and (4) for multiple non-binary treatments. In Section 3, we test various kernels k to test the sensitivity of our analysis to the choice of kernel. This enables us not only to work with the high-dimensional multivariate outcomes through positive definite kernels, but also to capture subtle effects of the hyperparameters on prediction and explanation that are beyond the mean effect. Finally, note that Zhao & Hastie (2021) also aims to provide a causal interpretation of the black-box model using partial dependence plot (PDP), and observe that the formulation of PDP is exactly the same as the back-door adjustment formula of Pearl (2009). Although this work might be applicable to analyze the causal effects of interest in this work, the authors emphasize that it should not replace a random experiment or a carefully designed observational study.

2.3. Observational study

In practice, we may not be able to compute $Y_h^*(x)$ and $E_h^*(x)$ for all $h \in \mathcal{H}$ because of the limit on computational resources. Hence, we face the fundamental problem of causal inference that prohibits us to evaluate the ITE in (1). To this end, we will denote the *observed* prediction and explanation by $Y_h(x) = Y_h^*(x) \,|\, H = h$ and $E_h(x) = E_h^*(x) \,|\, H = h$, respectively. Both (3) and (4) can be defined in terms of $Y_h(x)$ and $E_h(x)$, but the empirical estimates of these quantities may not correspond to the true treatment effects as Assumption 2.1 may not hold. We also state the common assumptions in the PO framework:

Assumption 2.2 (Unconfoundedness)**.** There exists no unobserved confounder between Y_h and H (and E_h and H).

Model zoos as data: In order to study the effect of hyperparameters on downstream Y and E, one must first obtain a large collection of models which are the result of combinations of the hyperparameters under study. Fortunately, such datasets already exist (*model zoo*s (Unterthiner et al., 2020; Jiang et al., 2019)). We use the dataset provided by Unterthiner et al. (2020), a large collection of existing models that have already been trained with pre-specified hyperparameters. We describe in greater detail in Section 3.1.

Direct vs. indirect influences: As we can see from Figure 1b, it is hypothesized that, given on the data instance x, there are two different paths from the hyperparameters H to explanation E. The former is a direct influence from H to E, whereas the latter is an indirect influence mediated through the prediction Y. A simple analysis can be performed to tell them apart. Let $(H_i(x), Y_i(x), E_i(x))_{i=1}^n$ be a collection of hyperparameters, corresponding predictions, and corresponding explanations, respectively. Then, we conduct the correlation analysis on this dataset, in particular, comparing the total causal effect of H on E vs. that of H on Y (Equation (4)). Next, we construct an artificial data set by randomly permuting the prediction $Y_i(x)$ in the original data, while keeping $H_i(x)$ and $E_i(x)$ fixed. This gives us a new data set $(H_i(x), Y_{[i]}(x), E_i(x))$ where $Y_{[i]}(x)$ is the permuted version of $Y_{[i]}(x)$. Finally, we repeat the same correlation analysis on the permuted data set. The rationale behind this is that random permutation of $Y_i(x)$ weakens the direct influence of $H_i(x)$ on $Y_i(x)$ as well as the direct influence of $Y_i(x)$ on $E_i(x)$. As a result, careful comparisons between both correlation analyses might reveal the extent to which the prediction Y mediates the total influence of F on E. However, since the underlying relationships can potentially be non-linear, one should avoid interpreting the difference in mediated (total) vs unmediated (direct) ITEs of H on E as the indirect effects (Pearl, 2022).

3. Analysis and results

This section provides details of our analysis and results of our observational study in both global setting (all models) and local setting (models in each performance buckets).

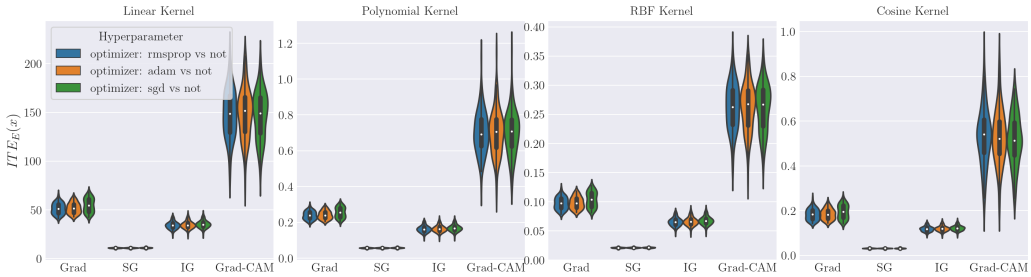

Figure 2: Comparison of the ITE_E values with kernelized version of (4) obtained for 100 `instances` from CIFAR10 for different choices of kernel (each column) shows that relative KTE values are not sensitive to the choice of kernels.

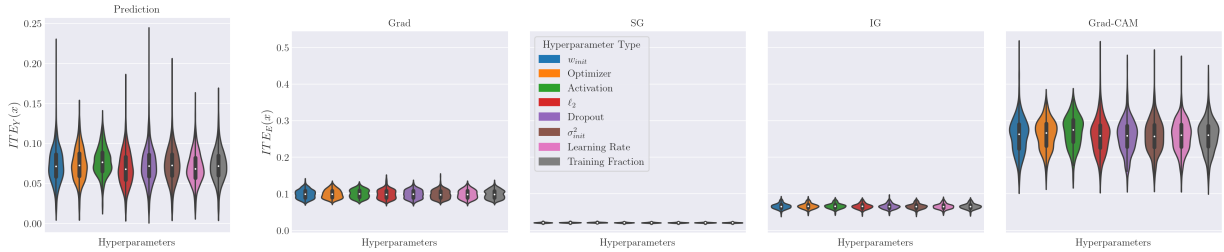

Figure 3: Comparison of ITE_Y and ITE_E for CIFAR10 shows that different types of H influence E and Y in a similar way.

3.1. Observational study details

Model zoo dataset and pre-processing explanations The dataset we leverage (Unterthiner et al. (2020)) contains 30,000 3-layer CNNs ($4,970$ parameters; weights and biases) that were trained until convergence (or a maximum of 86 epochs) for multiple datasets. The hyperparameters are drawn "independently at random" from pre-specified ranges (exact ranges in Appendix A.3). Both the ranges and the training procedure are natural and resemble standard practice in machine learning, and the models are trained on commonly used CIFAR10, SVHN, MNIST, and FASHION MNIST datasets. Note that the random seed (for mini-batch GD sampling and for weight initialization) and the architecture of the base models is fixed throughout, and the diversity of hyperparameters allows for a representative study of treatment effects.

We use Gradient (Simonyan et al., 2013; Erhan et al., 2009; Baehrens et al., 2009), SmoothGrad (Smilkov et al., 2017), Integrated Gradient (IG) (Sundararajan et al., 2017), and Grad-CAM (Selvaraju et al., 2016) in our work, methods are used due to their commonplace deployment (Adebayo et al., 2018). Note that many widely used methods are built based on these four methods (Xu et al., 2020; Wang et al., 2021; Simonyan et al., 2013). The generated explanation maps are pre-processed in conventional ways as done in (Adebayo et al., 2018) (details in Appendix A.3). Since some methods only produce positive attributions, we zero out any negative attributions for the methods that produce both positive and negative values. This is so that we can compare with all methods on an equal footing.

3.2. Results

KTE is not sensitive to the choice of kernel: As discussed in Section 2.2, the standard treatment effects in (4) were extended by replacing the $Y_h(x) - Y_{h'}(x)$ with $\|\phi(Y_h(x)) - \phi(Y_{h'}(x))\|^2_{\mathcal{G}}$ in order to go from a treatment effect *map* to a treatment effect *scalar*. Obtaining scalar values of treatment effect is important as it allows the comparison and ordering of effects between setups. To test the sensitivity of KTE with respect to choice of kernel, $k(\cdot,\cdot)$, empirically, we compare the distribution of ITEs obtained (as per (4)) for 4 choices of kernels: (i) linear: $k(a,b) = a^T b$; (ii) polynomial: $k(a,b) = (\gamma a^T b + 1)^3$ (with $\gamma = 1/\dim(a)$); (iii) RBF: $k(a,b) = \exp(-\gamma\|a-b\|^2)$; and (iv) cosine: $k(a,b) = a^T b/(\|a\|\,\|b\|)$. The results in Figure 2 show that the explanation ITE distributions are not sensitive to the choice of kernels that we tested. Note that similar trends hold for other hyperparameters in Figure 3. We use the RBF kernel for the remainder of the paper.

Most types of H influence E and Y in a similar way: Again, our goal is to measure the treatment effect of a causal ancestor (H) on E and Y. The H has different *types* (e.g., initialization, activation), and each type takes on multiple unique values (i.e., treatment values) whose treatment effect on Y or E can be evaluated via (4). As shown in Figure 3, this effect is similar across different types of H for both ITEs of Y and E. Stratifying the results per unique value of treatments also shows no apparent pattern, across all datasets considered (see Figure 12 - Figure 15).

While this phenomenon potentially hints at stable and uniform treatment effects on E and Y (good news), the treatment effect should only be stable when Y is not a random

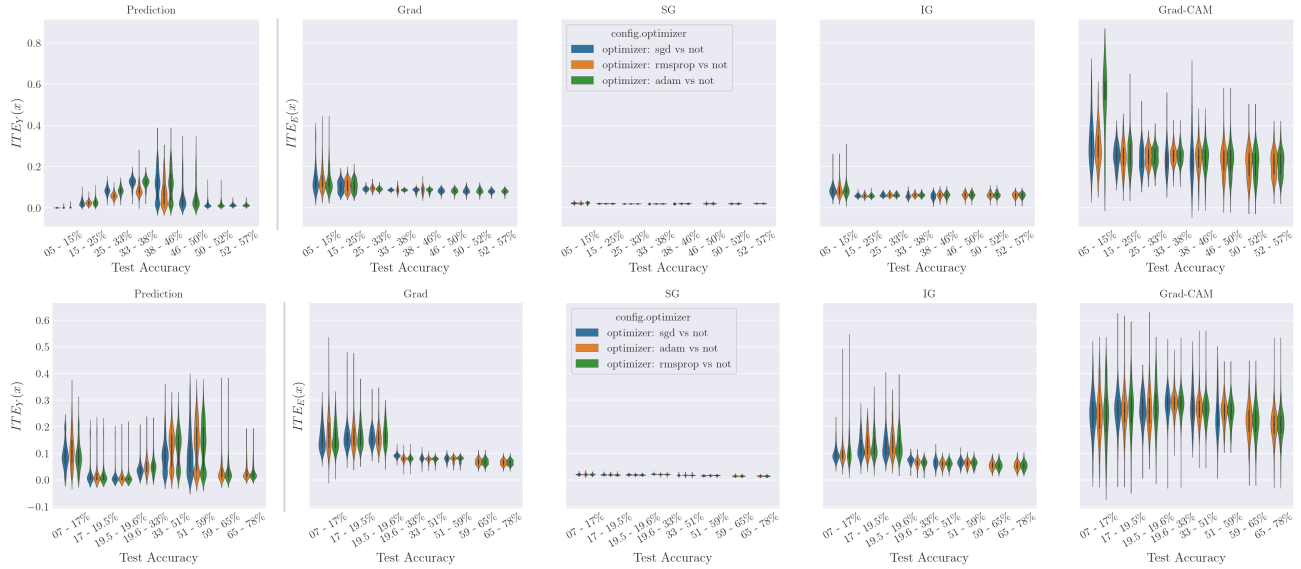

Figure 4: Comparison of ITE values of $h_{\text{optimizer}}$ on Y (left) and E (right) for models across different performance buckets, showing the discrepancy in the effect of H on Y vs. that on E (top: CIFAR10; bottom: SVHN). Interestingly, there is a difference of ITE_E across accuracy buckets, and more importantly, none of the explainability methods resemble ITE_Y.

prediction. In other words, in order to conclude that E reflects Y (measured by ITEs of E and Y), the observation must only happen when we expect E and Y to have a meaningful relationship (non-random prediction). Teasing out how much Y influences E is one of the long-standing questions in interpretability; some have argued that E is visually indistinguishable when Y is from trained or untrained models (Adebayo et al., 2018). How the relationship between E and Y changes as a function of performance of the model is important for practitioners in deciding when E can be used or not. Thus, we conduct the remaining analysis by stratifying models into different accuracy buckets.

Understanding ITE_Y and ITE_E separately across performance bucket: To understand the relation between Y and E as a function of model performance, we stratified 30k models into 8 buckets according to their accuracies to observe the treatment effect in each group (Figure 4). We use 0-20$^{\text{th}}$, 20-40$^{\text{th}}$, 40-60$^{\text{th}}$, 60-80$^{\text{th}}$ and 80-90$^{\text{th}}$, 90-95$^{\text{th}}$, 95-99$^{\text{th}}$ and 99-100$^{\text{th}}$ percentiles as groups (finer granularity for top models that are more likely to be deployed); see A.3 for more details.

The choice of the control group: Calculating ITE for each performance bucket requires a decision on control groups, i.e., the point of comparison. There are two natural choices 1) select a control group within each accuracy bucket or 2) use the same control group across all buckets. Each choice means we are answering slightly different questions; (1) answers "the effect of $h_i = n$ w.r.t. $h_i \neq n$ on $x \in X$ such that training on $h_i \neq n$ gives a similarly performing model" while (2) answers "the effect of $h_i = n$ w.r.t $h_i \neq n$ on $x \in X$ such that training on $h_i \neq n$ gives model with baseline performance". Although the latter enables comparison of performance buckets on similar footing, the simultaneous change of $h_i = n$ to $h_i \neq n$ and the change in performance bucket makes it difficult to attribute the ITE value to the hyperparameter. Therefore, we continue with within-accuracy-bucket control groups, and refrain from comparing absolute values of ITE (for Y or E) across buckets, but instead look to *relative* ITE values of H on Y and E across buckets (different baselines in Figure 10).

Unlike ITE_Ys, ITE_Es do not differ much across accuracy buckets, although the range of ITE_Es is slightly larger for the lowest accuracy bucket of some methods (Figure 4; right). Note that the range of ITE_Es are different in different methods, e.g., Grad-CAM shows a larger range of ITE_Es than all others, implying that the effect of H varies much more than all other methods. The different distribution of ITE_Es compared to ITE_Y in each accuracy bucket raises an important question: how does the relationship between Y and E (measured by treatment effect of H on both) change as models' performance changes?

Understanding the (odd) relationship between ITE_Y and ITE_E: Having investigated how ITEs of Y and E vary separately across performance buckets, we next turn to the relationship between ITE_Y and ITE_E (Figure 5). Interestingly, the pattern of this relationship reverses as it passes the mid-accuracy bucket. In low-accuracy buckets, hyperparameters seem to influence E in a greater variety of ways (bigger range) than Y, resulting in a mostly vertical scatter plot. This pattern reverses when the model accuracy passes 33% (note that accuracy by chance is 10%), and hyperparameters suddenly start influencing Y in a greater variety of

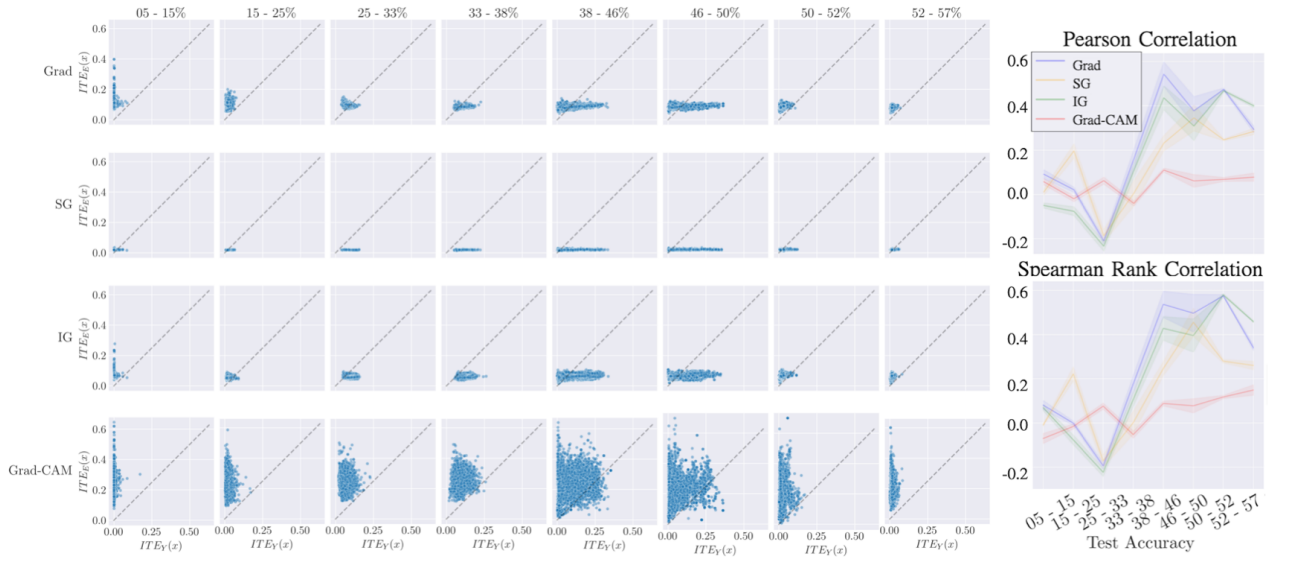

Figure 5: (left) Each column is a subset of models at each accuracy bucket, each row is different explanation methods. Whereas low-performing models (first column) show little change in predictions as their explanations differ, top-performing models show the reverse of this trend. (right) Correlation measures of the scatter plots on the left, showing decreased correlation in top 1% models.

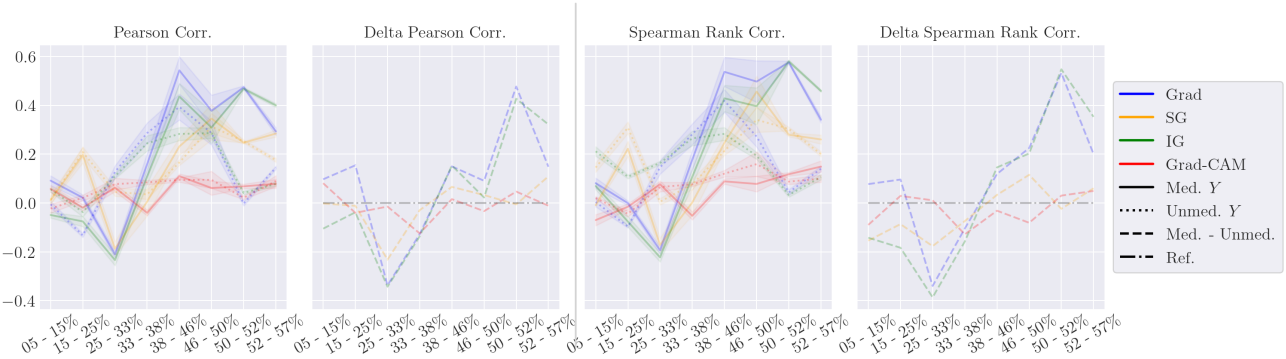

Figure 6: Pearson correlation and Spearman's Rank correlation for ITE_Y and ITE_E across different explanation methods and model performance buckets, for mediated (total) and unmediated (direct) Y. Higher absolute delta values mean lesser direct effect of hyperparameters on explanations, which is desired. Conversely, the reference line means the total effect is the direct effect (undesired).

ways than E. In other words, in high-performing models, E seems to be similar regardless of hyperparameters, while Y's vary more by the choice of hyperparameters. However, in the highest bucket, we observe better correlations between ITEs of E and Y.

One way to summarize the scatter plot is to compute correlation coefficients between Y and E (as shown in Figure 5; right). Since the absolute value of each ITE is not directly comparable (due to different domains for Y and E, an different baseline controls groups, as explained above), we measure Spearman and Pearson correlations between the raw ITE values. In summary, this result confirms that ITEs of E and Y vary across each accuracy, but not in an expected fashion. One may conjecture that ITEs of E and Y become increasingly more correlated as the accuracy increases, indicating that E becomes a better reflection of Y (at least) as model becomes more accurate. In addition, this correlation would be ideally close to 1 if the ITE of Y is strongly correlated with that of E. However, this is not what we observe. In most cases, the correlation *decreases* from mid-high performing bucket (80-99 percentile) to highest performing (99-100 percentile) bucket. Additionally, most of methods shows higher correlation in lowest performing bucket (0-20 percentile) than the mid-performing bucket (20-60 percentile).

This pattern described above is shared across all types of hyperparameters across four datasets (see Appendix A.3) with the exception of the activation function hyperparameter. For the activation function, this pattern seems to be even more exacerbated. Some methods achieve its lowest corre-

lation point when the model's accuracy is the highest (IG), while others achieve the best correlation when the model's accuracy is the lowest (SmoothGrad). Many others stay relatively low correlations throughout. This is interesting, as one can argue activation hyperparameter more actively shape the representational space of the model than other Hs. Further study is needed to tease out the effect of many other space-shaping hyperparameters. The next natural question is whether the amount of correlation between ITEs for Y and E is 'high enough'. We can partly answer this question by teasing apart the direct and indirect influence H on E via Y (see Figure 1b). In other words, how much of this correlation is because of Y v.s., a factor before model was even trained, H (a causal ancestor of E and Y)?

Direct vs. indirect influences of Y on E: The goal of teasing apart the direct and indirect influences of H on E (described in Section 2.3) is to observe the extend to which Y mediates the total influence of H on E. Intuitively, if explanations were sensitive only to predictions, one would observe high mediated influence of H on E (mediated via Y), and low unmediated influence. Conversely, a high unmediated affect of H on E suggests sensitivity of explanations to *factors* not related to the prediction, warranting further investigation.

Until now, the ITE_E values above measure the total effect of H on E (directly, and indirectly through Y). In order to measure indirect effect, one need only to "sever" the influence that H has on Y while retaining its effect on E. This is achieved by comparing H's treatment effects on E when Y is and is not randomized (Pearl, 2022). The difference between the total and direct effects (depicted in the second and fourth subfigures in Figure 6) roughly corresponds to the effect of H on E mediated through Y (which is ideally high). This follows roughly the same pattern as direct influence, not a monotonic increase and with an odd dip in mid-accuracy bucket. More importantly, while the influence of H on E mostly comes from Y between 80-99 percentile groups for IG and Grad (desirable), we observe a sudden decrease of relative direct influence of Y in the top-performing models (not desirable)–models that are more likely to be deployed. For the case of SG and Grad-CAM, the influence of H on E mostly comes from H, not from the trained model or the prediction from it Y. Putting it together, our comparison of direct and indirect influence reveals that the pattern of how Y mediates the total influence of H on E is surprising and undesirable at times. Results are robust with respect to the choice of bucketing strategy (Figure 19).

4. Discussion and conclusions

Our work investigates the relationship between E and Y using the potential outcomes framework. In analyzing the treatment effect of a causal ancestor (determined prior to model training) of E and Y on them, direct and indirect influence reveals that the pattern of how Y mediates the total influence of H on E is unintuitive, especially in top performing models where Y's direct influence on E decreases. In other words, in the top performing models, there are *other* factors that influences E more so than the prediction of the model, Y. We believe that the influence of most Hs on E should be mediated through Y (note that *which* H should not influence E is a decision by a user) and the goal of our work is to first show that such influence exists in current models and present methods to perform such analysis.

One can view our analysis as a more extensive, causal edition of Adebayo et al. (2018); we measure the treatment effect of H on E and Y using the Potential Outcome framework across 30k models, while they measure *visual* similarities of Es as varying the quality of Y in a single pair of models (trained and untrained). The relatively higher correlation in lowest accuracy bucket (which can be viewed as untrained network) from our analysis offers additional evidence for the phenomenon observed therein. Furthermore, our analysis reveals that Grad-CAM (which arguably 'passed' the sanity check in Adebayo et al. (2018)) shows worse correlation between the two ITEs across the buckets, meaning that the hyperparameters affect Y and E differently, hinting that no methods concretely outperform others. Our results should be taken as a strong encouragement for practitioners to review other evidence instead of taking explanations at their face value in their final decision making.

Limitations and Future Work

The problem framing in Figure 1, the formulations in Section 2, and the analytical framework presented over hyperparameter settings above naturally extend to any ML system (white-box or black-box) which have hyperparameters, $\mathcal{H}$, or more generally, up-stream *factors*, that affect a final model. The specific analyses presented in our paper, however, is bound by the choices made during the model zoo construction (in (Unterthiner et al., 2020)), e.g., choice and range/values of hyperparameters, and thus, the interpretation must be limited to the domain of $\mathcal{H}$ that we tested. For instance, while the model zoo offers an extensive number of models, their architecture is kept constant in all models (3 CNN layers, $\mathcal{O}(1e3)$). Further studies on larger and complex models (e.g., (Frankle & Carbin, 2018; Jiang et al., 2019)) or similar analysis when training dataset is (adversarially) changed (e.g., (Wang et al., 2021)) across different stages of training could reveal interesting insights. Finally, extending our work to uncover the effect of hyperparameters on other types of explanations would be interesting, e.g., influential samples (Koh & Liang, 2017), Shapley values (Lundberg & Lee, 2017), concept-based methods (Kim et al., 2018) surrogate-based methods, and recourse-based explanations and recommendations (Karimi et al., 2020).

References

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

Adebayo, J., Muelly, M., Abelson, H., and Kim, B. Post hoc explanations may be ineffective for detecting unknown spurious correlation. *ICLR*, 2022.

Alqaraawi, A., Schuessler, M., Weiß, P., Costanza, E., and Berthouze, N. Evaluating saliency map explanations for convolutional neural networks: a user study. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pp. 275–285, 2020.

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., and Müller, K.-R. How to explain individual classification decisions. *arXiv preprint arXiv:0912.1128*, 2009.

Bau, D., Liu, S., Wang, T., Zhu, J., and Torralba, A. Rewriting a deep generative model. *CoRR*, abs/2007.15646, 2020. URL https://arxiv.org/abs/2007.15646.

Chu, E., Roy, D., and Andreas, J. Are visual explanations useful? a case study in model-in-the-loop prediction. *arXiv preprint arXiv:2007.12248*, 2020.

D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., et al. Underspecification presents challenges for credibility in modern machine learning. *arXiv preprint arXiv:2011.03395*, 2020.

Eilertsen, G., Jönsson, D., Ropinski, T., Unger, J., and Ynnerman, A. Classifying the classifier: dissecting the weight space of neural networks. *CoRR*, abs/2002.05688, 2020. URL https://arxiv.org/abs/2002.05688.

Erhan, D., Bengio, Y., Courville, A., and Vincent, P. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

Frankle, J. and Carbin, M. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Jiang, Y., Krishnan, D., Mobahi, H., and Bengio, S. Predicting the generalization gap in deep networks with margin distributions. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJlQfnCqKX

Kapishnikov, A., Venugopalan, S., Avci, B., Wedin, B., Terry, M., and Bolukbasi, T. Guided integrated gradients: An adaptive path method for removing noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5050–5058, 2021.

Karimi, A.-H., Barthe, G., Schölkopf, B., and Valera, I. A survey of algorithmic recourse: contrastive explanations and consequential recommendations. *ACM Computing Surveys (CSUR)*, 2020.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

Kindermans, P.-J., Hooker, S., Adebayo, J., Alber, M., Schütt, K. T., Dähne, S., Erhan, D., and Kim, B. The (un) reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pp. 267–280. Springer, 2019.

Koh, P. W. and Liang, P. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, pp. 1885–1894. PMLR, 2017.

Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., and Liang, P. Concept bottleneck models. In *International Conference on Machine Learning*, pp. 5338–5348. PMLR, 2020.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pp. 4768–4777, 2017.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual knowledge in gpt. *arXiv preprint arXiv:2202.05262*, 2022.

Nie, W., Zhang, Y., and Patel, A. A theoretical explanation for perplexing behaviors of backpropagation-based visualizations. In *ICML*, 2018.

Parliament and of the European Union, C. General data protection regulation. 2016.

Pearl, J. *Causality*. Cambridge university press, 2009.

Pearl, J. Direct and indirect effects. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pp. 373–392. 2022.

Poursabzi-Sangdeh, F., Goldstein, D. G., Hofman, J. M., Vaughan, J. W., and Wallach, H. Manipulating and measuring model interpretability. *arXiv preprint arXiv:1802.07810*, 2018.

Reichenbach, H. *The Direction of Time*. University of California Press, Berkeley, CA, 1956.

Rieger, L., Singh, C., Murdoch, W. J., and Yu, B. Interpretations are useful: penalizing explanations to align neural networks with prior knowledge. In *ICML*, 2020.

Rubin, D. B. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

Schölkopf, B. and Smola, A. J. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

Selvaraju, R. R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., and Batra, D. Grad-cam: Why did you say that? *arXiv preprint arXiv:1611.07450*, 2016.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Smilkov, D., Thorat, N., Kim, B., Viégas, F., and Wattenberg, M. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Srinivas, S. and Fleuret, F. Rethinking the role of gradient-based attribution methods for model interpretability. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=dYeAHXnpWJ4.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR, 2017.

Unterthiner, T., Keysers, D., Gelly, S., Bousquet, O., and Tolstikhin, I. Predicting neural network accuracy from weights, 2020. https://github.com/google-research/google-research/tree/master/dnn_predict_accuracy.

Wachter, S., Mittelstadt, B., and Russell, C. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

Wang, Z., Fredrikson, M., and Datta, A. Robust models are more interpretable because attributions look normal. *arXiv preprint arXiv:2103.11257*, 2021.

Xu, S., Venugopalan, S., and Sundararajan, M. Attribution in scale and space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9680–9689, 2020.

Zhao, Q. and Hastie, T. Causal interpretations of black-box models. *Journal of Business & Economic Statistics*, 39 (1):272–281, 2021.

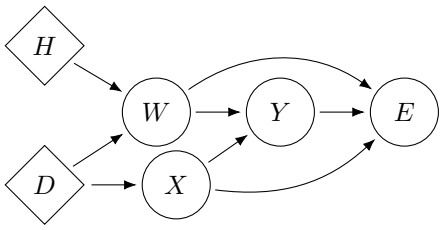

Figure 7: Extended version of explanation generating process from Figure 1b, now with weights W and dataset D made explicit.

A. Additional background material

A.1. The explanation generation process

To ease understandability, we refer to Figure 7 as the extended graph of Figure 1b which makes the weights W and data D explicit variables. Similar to Figure 1, diamond nodes are considered factors whose effect we study, and circle nodes are random variables. In this extended graph, we clarify that H is *not* the model or trained weights. In other words, what we call hyperparameters (H) are sets like "method of optimization: SGD, AdaGrad" or "regularizer coefficients: 0.1 or 0.01 etc". All Hs take a constant value before we train any model and before observing any data. We can train many models with the same hyperparameter values. Note that we don't have weights (let's call it W) in Figure 1, as they are not the focus of our study. We are only interested in whether and how decisions made prior to training a model (Hs) influence explanations

Furthermore, considering the manner in which the model zoo was constructed whereby hyperparameters are sampled independently from some domain, there are no edges (no backdoors) from X (or D) to H. On the other hand, W may be affected by the data distribution D, directly and/or through the training samples, but W is not the focus of our work. Since we focus on the causal effect of hyperparameters H on Y and E (not the weights W on Y and E), the formulations in Section 2.2 remain unchanged.

A.2. On the identifiability and computability of treatment effects

An astute reader may notice that evaluating the treatment effects above as the difference between counterfactual contrasts bears a resemblance to another common explainability method, namely *counterfactual explanations* (Wachter et al., 2017). This parallel is evident when thinking of Figure 1 in a coarser manner, i.e., $\mathcal{H}, \mathcal{X} \rightarrow \mathcal{Y}$, whereby the hyperparameters and dataset instance enter a *potentially blackbox but queriable procedure* and yield a prediction. Whereas the counterfactual explanations of Wachter et al. (2017) aim to identify minimal feature perturbations of the dataset instance under a fixed model (i.e., the hyperparameters do not change; procedure: *model prediction*), evaluating treatment effects as in Equation (1) is done by iterating over values of hyperparameters to contrast resulting predictions given a fixed dataset instance (procedure: *model training*).

Due to our mechanical setup, a number of interesting observations arise. Though the training, predicting, and explaining procedures may not be expressible in closed-form, the prediction Y_h in Equation (1) is exactly computable on a computer through *forward simulation*. In other words, upon selecting a set of hyperparameters, h, and under a fixed seed, all sources of randomness are controlled for and the procedures `T`, `P`, `E` deterministically yield a trained model, a prediction for a given instance, and the explanation for the said instance and model. This is significant as it allows for the *exact computation* of both $Y_{\text{TREATMENT}}$ and Y_{CONTROL} which in turn yields the value of the ITE exactly. In other words, we can view both $Y_{\text{TREATMENT}}$ and Y_{CONTROL} as *factual* outcomes. Therefore, unlike real world settings (e.g., taking a headache medication) where the analysis of ITE is not possible due to the impossibility of observing both *factual* and *counterfactual* outcomes simultaneously,[2] the effect of all treatments, on individual or aggregate levels, are identifiable.

Although the treatment effects are identifable, evaluating them is computationally expensive. To understand why, it helps to illustrate a parallel with the setting of counterfactual explanations (Wachter et al., 2017). Whereas the treatment effects in our setting (see Equation (1)) contrasts $Y^*_h(x)$ and $Y^*_{h'}(x)$, the work of Wachter et al. (2017) contrasts $Y^*_h(x)$ and $Y^*_h(x')$. Unlike the latter which only requires the invocation of the *predicting procedure* given a new `instance` (e.g., a forward pass through a neural network), the former invokes the *training procedure* given a new hyperparameter setting (i.e., a full

[2] In such cases, the ITE is either approximated, or the ATE is used instead.

Table 1: Comparison of the classical and mechanical (our) setting for computing ITE values.

(a) In the classical setting for computing treatment effects, only one of the potential outcomes for each individual, i, is observable. The average treatment effect is defined as the average difference between individual treatment effects $ATE = \mathbb{E}[Y_1^{(i)}] - \mathbb{E}[Y_0^{(i)}]$.

i	Y_0	Y_1	Y_2
1	a	-	-
2	-	f	-
3	-	-	k
4	-	h	-
⋮	⋮	⋮	⋮

(b) In our mechanical setting, given a model, $\hat{f}_h$, the potential outcome for any and all instances is computable (i.e., $\exists\, Y_h(X_i), i \in \mathcal{I} \implies \exists\, Y_h(X_k)\ \forall\, k \in \mathcal{I}$). Instead, one asks how to compute the treatment effect for h' when no data is available for this hyperparameter.

i	Y_0	Y_1	Y_2
1	a	e	-
2	b	f	-
3	c	g	-
4	c	h	-
⋮	⋮	⋮	⋮

re-training). In practice, computing power is limited and we may only have access to the predictions under a single model, say, $Y_h^*(x)$ and it can be prohibitively expensive to produce the prediction under a different model, $Y_{h'}^*(x)$, especially for large neural networks.

In order to reason about $Y_{h'}^*(x)$, one is compelled to instead ask a *counterfactual* question: "*What would the prediction have been, had the optimizer been* ν'?" which can be answered through causal modeling without conducting real-world experiments, i.e., retraining with optimizer ν'. Metaphorically, there would have been no need for counterfactuals had one been able to simulate the entire universe (limited by either identification or computation). It is the physical constraints that call for these counterfactuals. Unfortunately, the procedures in Figure 1 (left) are not available in closed form. We clarify that unlike the classical randomized control trial (RCT) setting of evaluating ATE by contrasting average ITE values (where instances are randomly assigned to control or treatment), the mechanical nature of our setting allows for the target evaluation of all instances under control (h) or any treatment regime (h'); the challenge lies in the fact that applying a treatment to any one individual is as expensive as applying it to all individuals (see Table 1a and Table 1b for comparison). In this case, future research may explore the question of whether one can learn approximate procedures (i.e., approximate structural equations) to *predict the predictions of an untrained classifier, given only its hyperparameters*.

An implicit assumption made in (4) was that of mutual independence between hyperparameters, i.e., $h_i \perp\!\!\!\perp h_j\ \forall\ j \neq i \implies h_{\setminus i} \sim \prod_{j \neq i} \mathbb{P}(h_j)$. This assumption yields an *unconditional* treatment effect, whereby the causal effect of $h_i = \text{TREATMENT}$ vs $h_i = \text{CONTROL}$ is averaged over all possible combinations of other hyperparameters, even if the combination rarely occurs in high performing models. In practice, however, it is conceivable that the hyperparameters are selected carefully by the system designer and may be interpreted as being sampled from a distribution over hyperparameters, $\mathcal{H}$, internalized by the designer through *prior* experience in training desirable models (e.g., accuracy, fairness). Such downstream criteria may act as a common child of the hyperparameters, inducing complex inter-dependencies (cf. Berkson's paradox, (Pearl, 2009)). In this case (i.e., $h_{\setminus i} \not\sim \prod_{j \neq i} \mathbb{P}(h_j)$), the treatment effect answers such a query as "among the set of hyperparameters that yield models with at least γ performance, what is the treatment effect of optimizer choice ν_1 as opposed to ν_2 on the local prediction of x?" Therefore, whether or not we assume hyperparameters to be mutually independent depends on the query being asked and assumptions made of the prediction/explanation generative process. Finally, one could consider straightforward extensions of (3) and (4) to support distributions over baseline control groups by adding an outer expectation that weights over the probability control group occurrence.

A.3. Model zoo details

For each of the 4 datasets (CIFAR10, SVHN, MNIST, FASHION) we consider $30,000$ pre-trained models, with diverse test accuracies resulting from the combinations of hyperparameters considered in the zoo (Unterthiner et al., 2020, Fig. 6). We optionally analyze models stratified by their test performance, over 8 performance buckets; Table 2 shows the boundaries of these buckets.

As a demonstration, Figure 8 shows the diversity in predictions of $30,000$ base models for a subset of CIFAR10 images for 1 randomly sampled datapoint from each class. It is noteworthy that the non-kernelized ITE values of (4) can be read directly from the figure, by contrasting the mean (shown in diamond) of each pair of nested bar plots (via application of linearity of expectations to (4)).

Table 2: Test accuracy boundaries for each performance bucket for each dataset in the model zoo (Unterthiner et al., 2020).

percentile	0-20	20-40	40-60	60-80	80-90	90-95	95-99	99-100
CIFAR10	5-15	15-25	25-33	33-38	38-46	46-50	50-52	50-57
SVHN	7-17	17-19.5	19.5-19.6	19.6-33	33-51	51-59	59-65	65-78
MNIST	4-11	11-35	35-73	73-89	89-95	95-96	96-97	97-98
FASHION	1-11	11-47	47-68	68-76	76-82	82-84	84-85	85-88

Pre-processing explanations and other details To study the effect of hyperparameters on explanations, we generate explanations, $E_h(x)$, via saliency-based methods. In particular, the Gradient (Simonyan et al., 2013; Erhan et al., 2009; Baehrens et al., 2009) and its smooth counterpart, SmoothGrad (Smilkov et al., 2017), Integrated Gradient (IG) (Sundararajan et al., 2017), and Grad-CAM (Selvaraju et al., 2016) methods are used due to their commonplace deployment[3] (Adebayo et al., 2018). Note that many other widely used methods are based on these four methods (Kapishnikov et al., 2021; Xu et al., 2020; Wang et al., 2021; Simonyan et al., 2013). The generated explanation maps $E_h(x)$ are then processed to first remove outliers (via percentile clipping the values above 99th percentile), following by normalizing all attributions to fall in $[0, 1]$. For Grad-CAM which only generates positive attributes, this is straightforward; for other methods that give positive and negative attributes (as each carry different semantics; contributing towards/against the prediciton), we first normalize to $[-1, 1]$ and then clip any value below 0.

The set of hyperparameters considered include the choice of optimizer, w_0 type, w_0 std., b_0 type, choice of activation function, learning rate, ℓ_2 regularization, dropout strength, and dataset split (see Unterthiner et al., 2020, Appendix A.2). To evaluate treatment effects as per (4), continuous features are discretized by (log-)rounding to the nearest predetermined marker from within the range of the feature.[4]

B. Additional experimental results

In this section, we present additional experimental results to complement those in the main body across different data dimensions or on new datasets.

As a demonstration, Figure 8 shows the diversity in predictions of $30,000$ base models for a subset of CIFAR10 (top) and SVHN (bottom) images for 1 randomly sampled datapoint from each class. It is noteworthy that the non-kernelized ITE values of (4) can be read directly from the figure, by contrasting the mean (shown in diamond) of each pair of nested bar plots (via application of linearity of expectations to (4)).

Alternative bucketing strategies

We recognize that any arbitrary bucketing (of which there are infinite ways) may create artifacts such as Simpson's paradox; however, this paradox is important to consider when the within-bucket analysis may be used to infer across-bucket trends. In our study, both before (Figure 2 and Figure 3) and after (Figure 4 and later figures) bucketing, the analysis always focuses on higher-level trends across buckets. We use bucketing as a mere convenience, to perform a finer grained analysis than bulking all models together; for this, we required some bucketing scheme that would show a trend but also retain a sufficient number of models in each bucket for the analysis to be statistically-sound.

Importantly, we do not use an equal-sized bucket strategy as this would necessarily miss/average out important signals, especially in the set of likely-deployed models (highly performant candidate models). For example, in a 3-equal-sized-bucket strategy, the last suggested bucket includes models with accuracies in the 66-100 percentile, arguably a very large band. One can argue that practitioners would want to deploy a model, among a set of high accuracy models, that also yields good explanations. Yet, the coarser bucketing (such as 3 bucketing) will average out 66th-percentile accuracy models and 100th-percentile accuracy models. Put differently, our conclusion would have been less relevant to practitioners had we only shown those 3 buckets – they may argue that the poor quality of explanation is merely the result of poor model prediction! For this reason, our bucketing reports finer granularity as accuracy increases.

Having said this, we do acknowledge that our bucketing choice is one of many possible choices (we will more clearly

[3] All methods are openly accessible here: https://github.com/PAIR-code/saliency.

[4] The following markers are used for (log-)rounding continuous features: ℓ_2 reg.: $[1e^{-8}, 1e^{-6}, 1e^{-4}, 1e^{-2}]$, dropout: $[0, 0.2, 0.45, 0.7]$, w_0 std.: $[1e^{-3}, 1e^{-2}, 1e^{-1}, 0.5]$, learning rate: $[5e^{-4}, 5e^{-3}, 5e^{-2}]$.

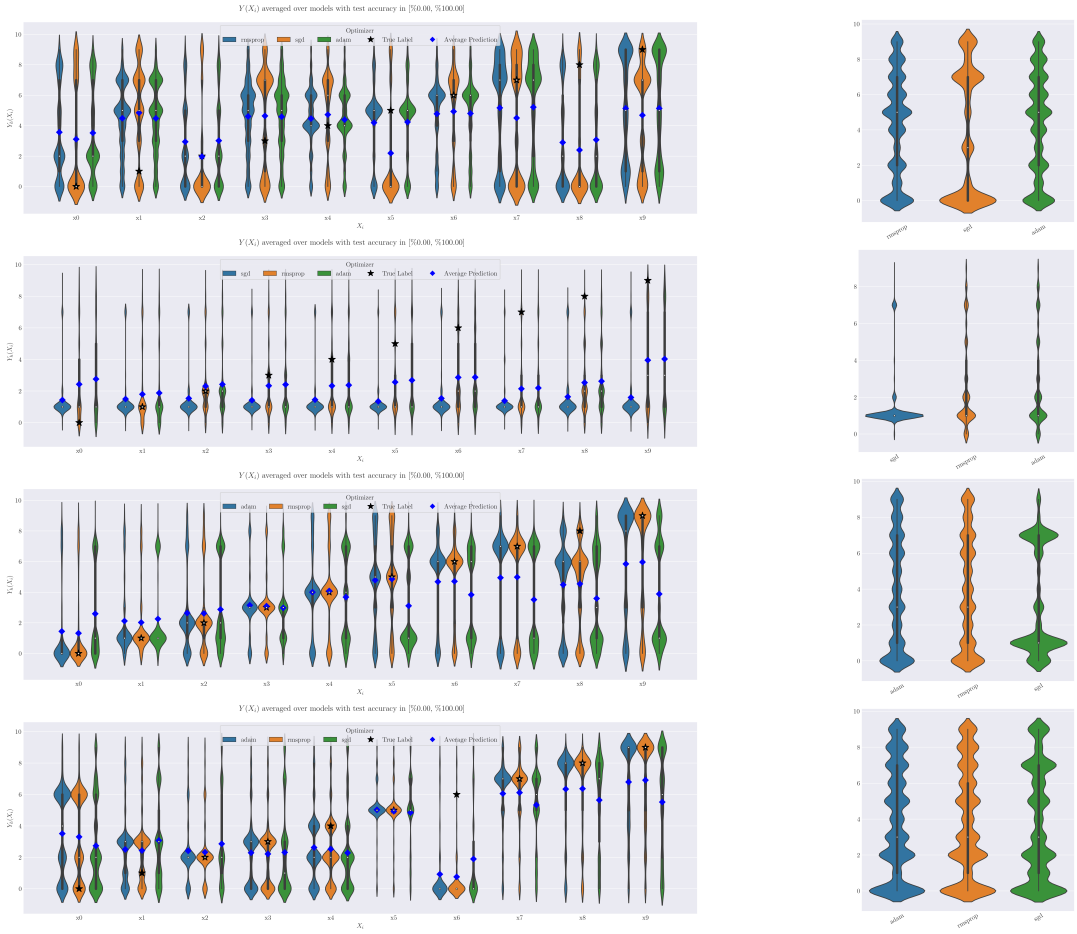

Figure 8: The distribution of $Y_h(x_i)$ for a subset of 10 random `instances` (1 per class) on $30,000$ base models (row 1: CIFAR10; row 2: SVHN; row 3: MNIST; row 4: FASHION). For each instance, each column holds the value of $h_{\text{optimizer}}$ fixed at one of m unique values pertaining to this hyperparameter, while unconditionally iterating over other hyperparameters. In this manner, the difference in predictions across values of the hyperparameter, both at an individual (left) and aggregate level (right) can be attribute to, and only to, changes in this hyperparameter.

acknowledge this in the final draft). Our strategy was to ensure that we have enough samples in each bucket to ensure credible conclusion, while it is not too large (to prevent averaging over too broad of an accuracy range which could cancel out individual model effects). In other words, had we had infinite models, we argue that finer grain (e.g., 10k models per each accuracy decimals) would have been optimal, but we were limited by the availability of the dataset.

For instance, an alternative 3-bucket scheme (not equally split, but with finer granularity for higher performant models) would consider, e.g., the 0-80 pctl, 80-99 pctl, and 99-100 percentile buckets. This plot can be drawn over the existing results in Fig 18 , by averaging the correlation values across buckets (NB this is approximate because the value averages the within-bucket ITE of multiple existing buckets, but does not consider cross-bucket ITEs). The resulting plot, shown in Figure 19, yields the same conclusion: the highest-performing models (in 99-100 pctl bucket) generate explanations that are more directly dependent on H, compared to their lower- performant counterparts (80-99 pctl).

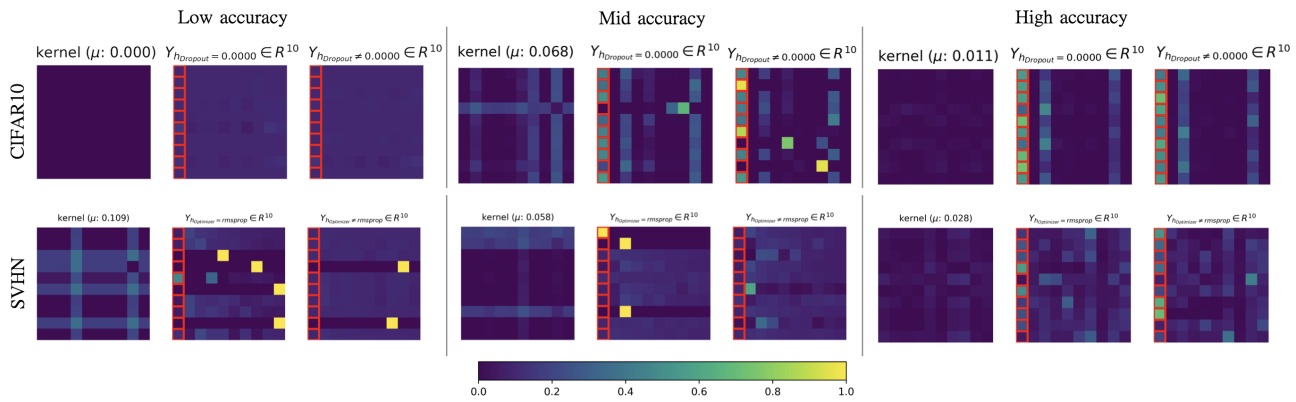

Figure 9: Examples of class predictions ($Y_{h=n}(x)$ and $Y_{h\neq n}(x)$) and their dissimilarities ($\|\phi(Y_{h=n}(x)) - \phi(Y_{h\neq n}(x))\|^2_{\mathcal{G}}$) for different accuracy buckets for CIFAR10 (top) and SVHN (bottom). Each row shows 10 random predictions from 3 models in the low- (left), mid- (center), and top- (right) performance buckets, under two different treatment groups for the dropout value ($= 0$ and $\neq 0$). In each performance bucket, there are three subplots. Each subplot is showing 10 randomly selected samples (each row) and their post-softmax values for one of the 10 classes (hence a 10×10 grid). The first plot in each trio shows the RBF kernel evaluation of the center and right predictions. The center and right plots show these treatment/control groups. This figure is intended to complement Figure 4 to explain why ITE for Y is large for mid-accuracy buckets and small for high-accuracy buckets. For CIFAR10, the values are small for low-performing models (most models in this bucket predicting similarly) but for SVHN the values are large due to different diverse predictions.

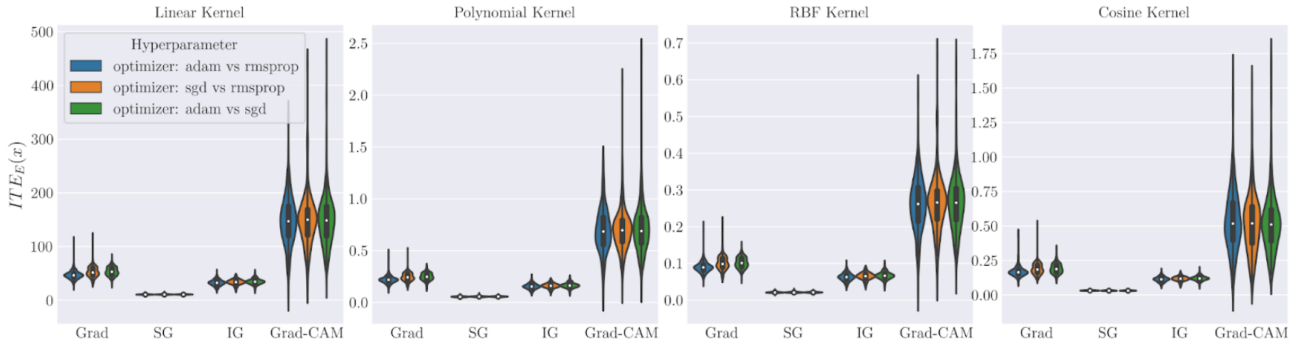

Figure 10: Comparison of the ITE values with kernelized version of (4) for $E_h(x)$ obtained for 100 `instances` from CIFAR10 for different choices of kernel (each column) shows that KTE is not sensitive to the choice of kernels. Contrast this figure with Figure 2; we conclude that the choice of baseline also does not affect the overall trend and should be chosen according to the question in mind: to compare the effect of a hyperparameter value against all other possible values, or against a particular value.

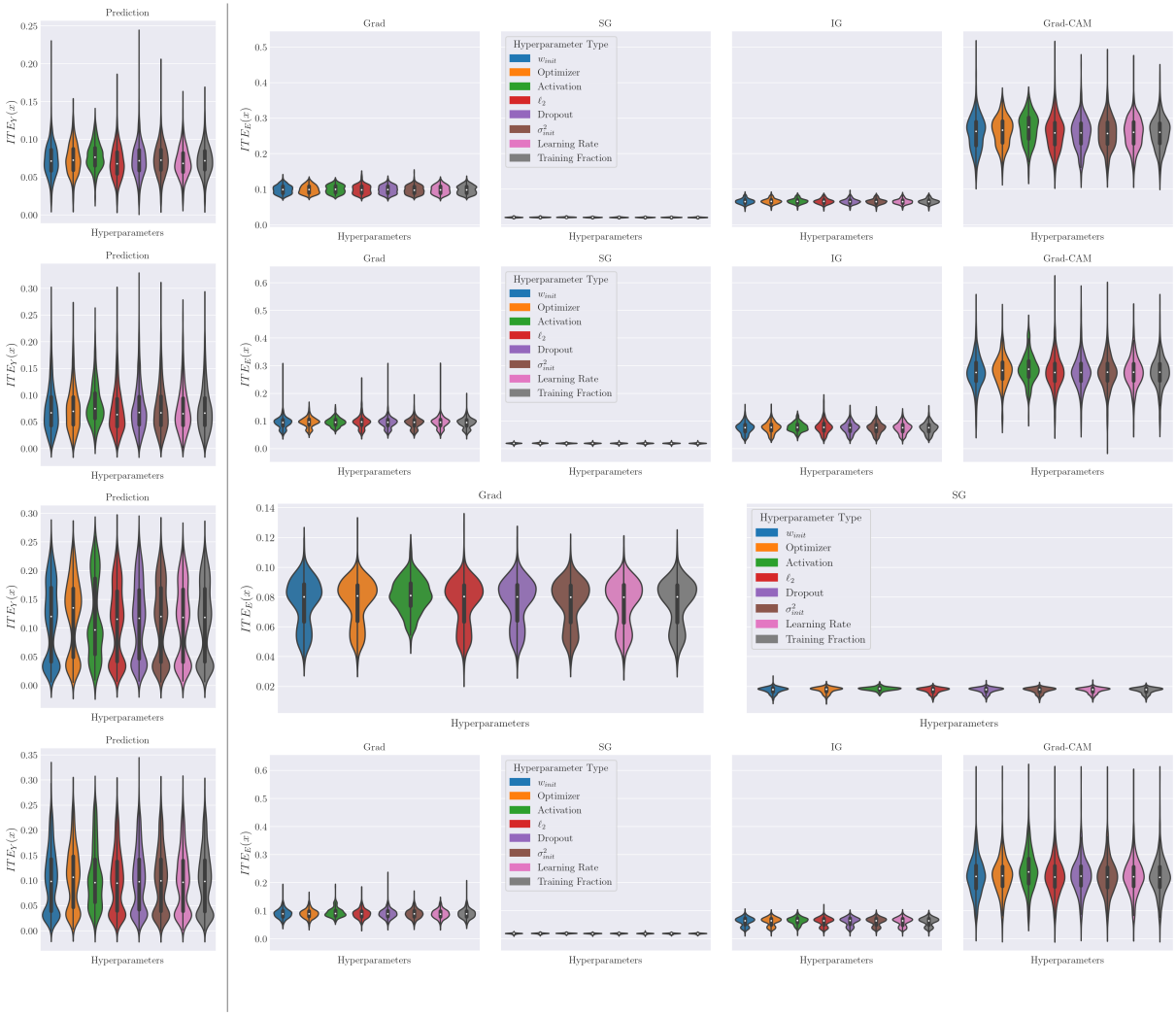

Figure 11: ITE values for Y (left) and E (right) show similar effect for different *types* of H across CIFAR10 (row 1), SVHN (row 2), MNIST (row 3), FASHION (row 4).

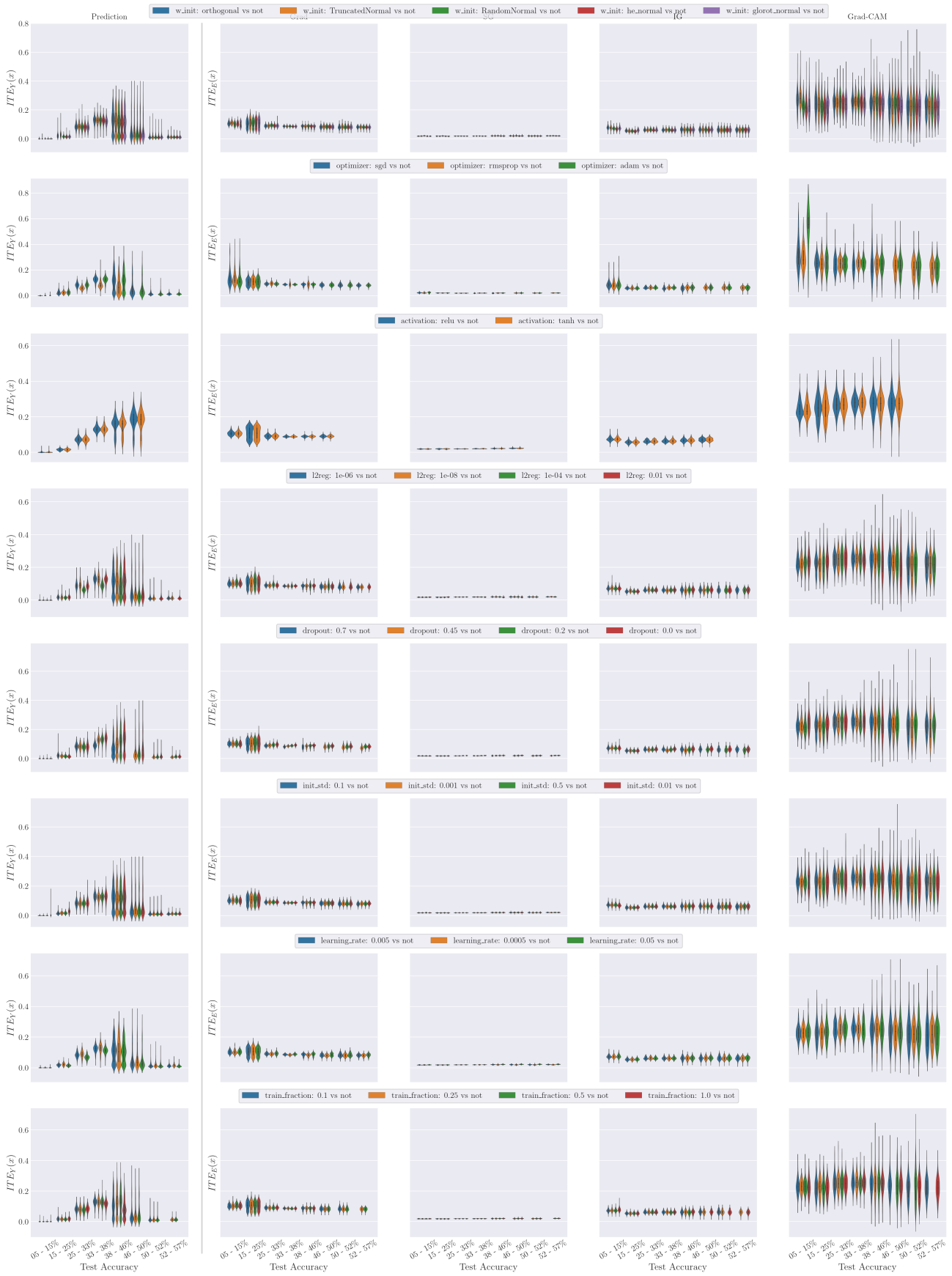

Figure 12: Comparison of ITE values of $h_{\text{optimizer}}$ on Y (left) and E (right) for models trained on CIFAR10 across different performance buckets, showing the discrepancy in the effect of H on Y vs. that on E.

Figure 13: Comparison of ITE values of $h_{\text{optimizer}}$ on Y (left) and E (right) for models trained on SVHN across different performance buckets, showing the discrepancy in the effect of H on Y vs. that on E.

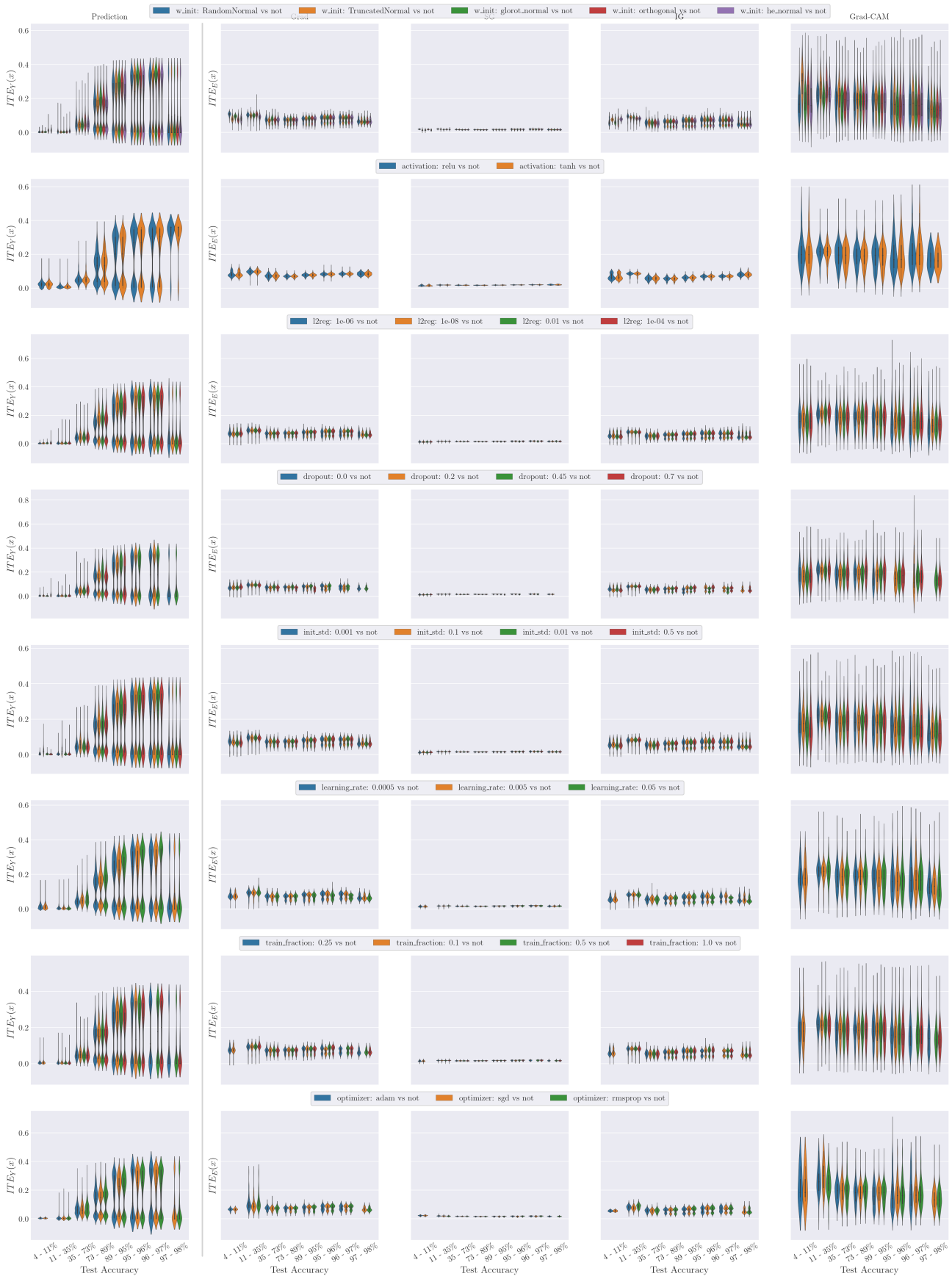

Figure 14: Comparison of ITE values of $h_{\text{optimizer}}$ on Y (left) and E (right) for models trained on MNIST across different performance buckets, showing the discrepancy in the effect of H on Y vs. that on E.

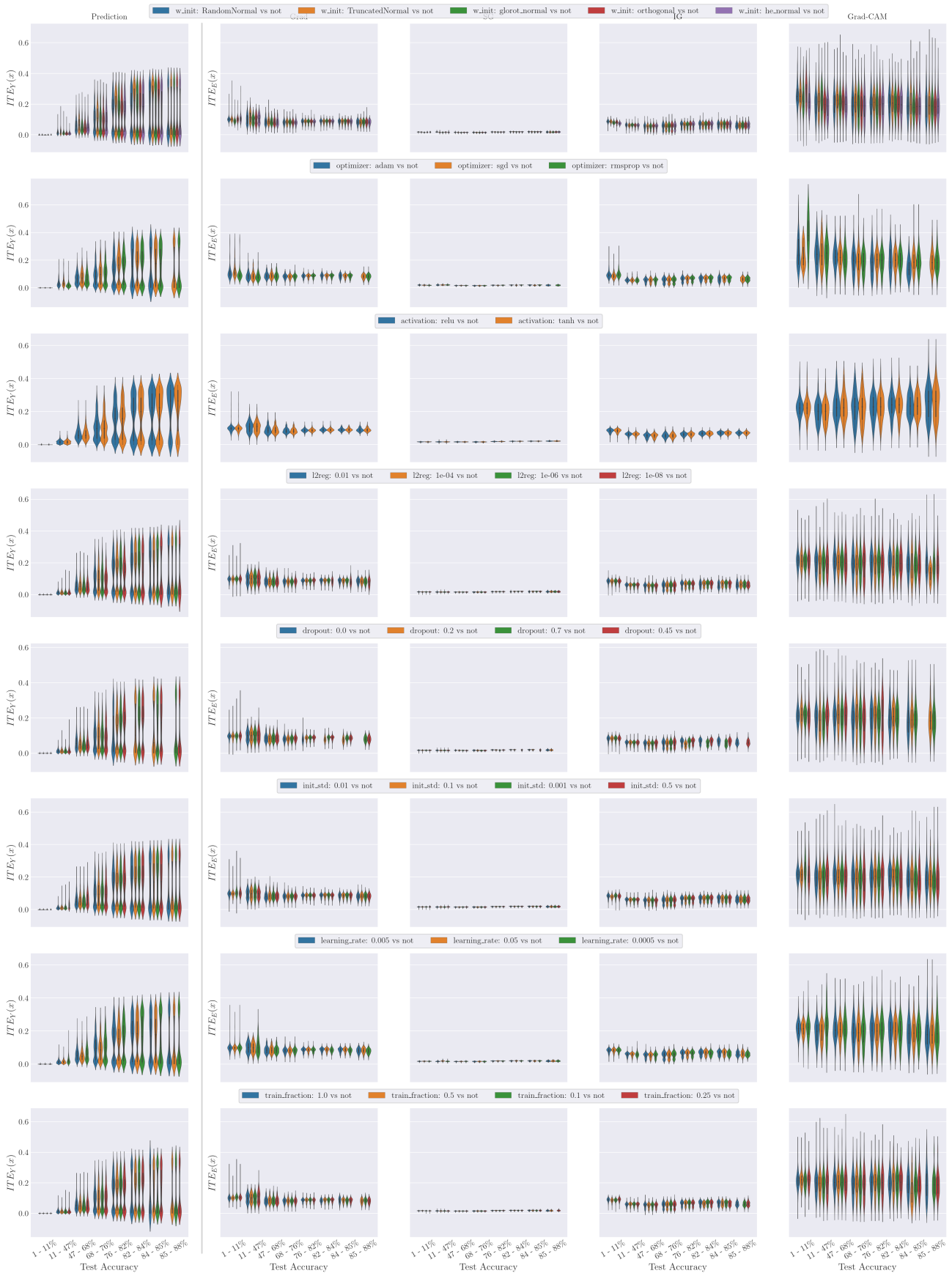

Figure 15: Comparison of ITE values of $h_{\text{optimizer}}$ on Y (left) and E (right) for models trained on FASHION across different performance buckets, showing the discrepancy in the effect of H on Y vs. that on E.

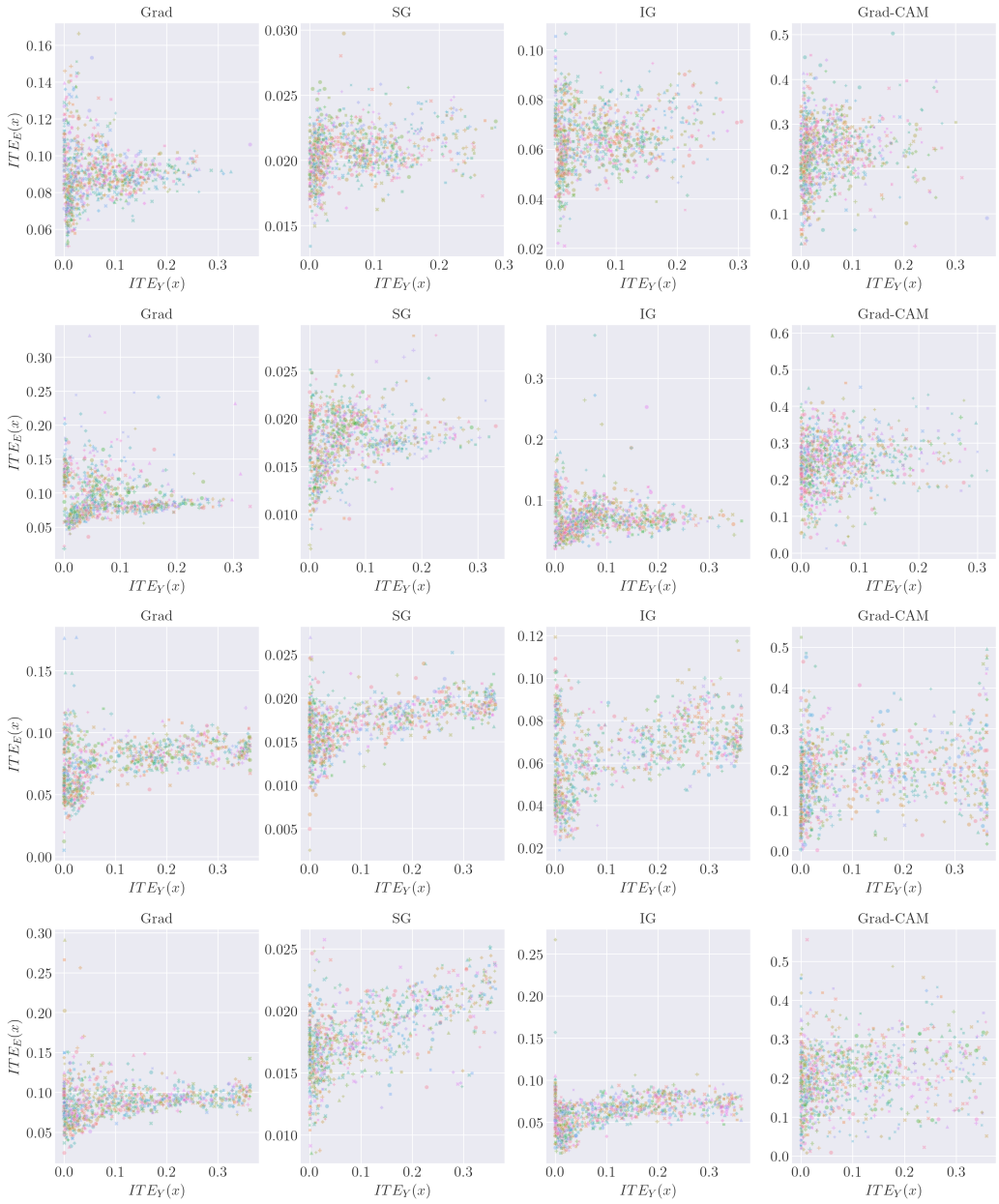

Figure 16: Scatter plot of ITE values for Y and E (row 1: CIFAR10; row 2: SVHN; row 3: MNIST; row 4: FASHION) across explanation methods reveals no apparent patterns.

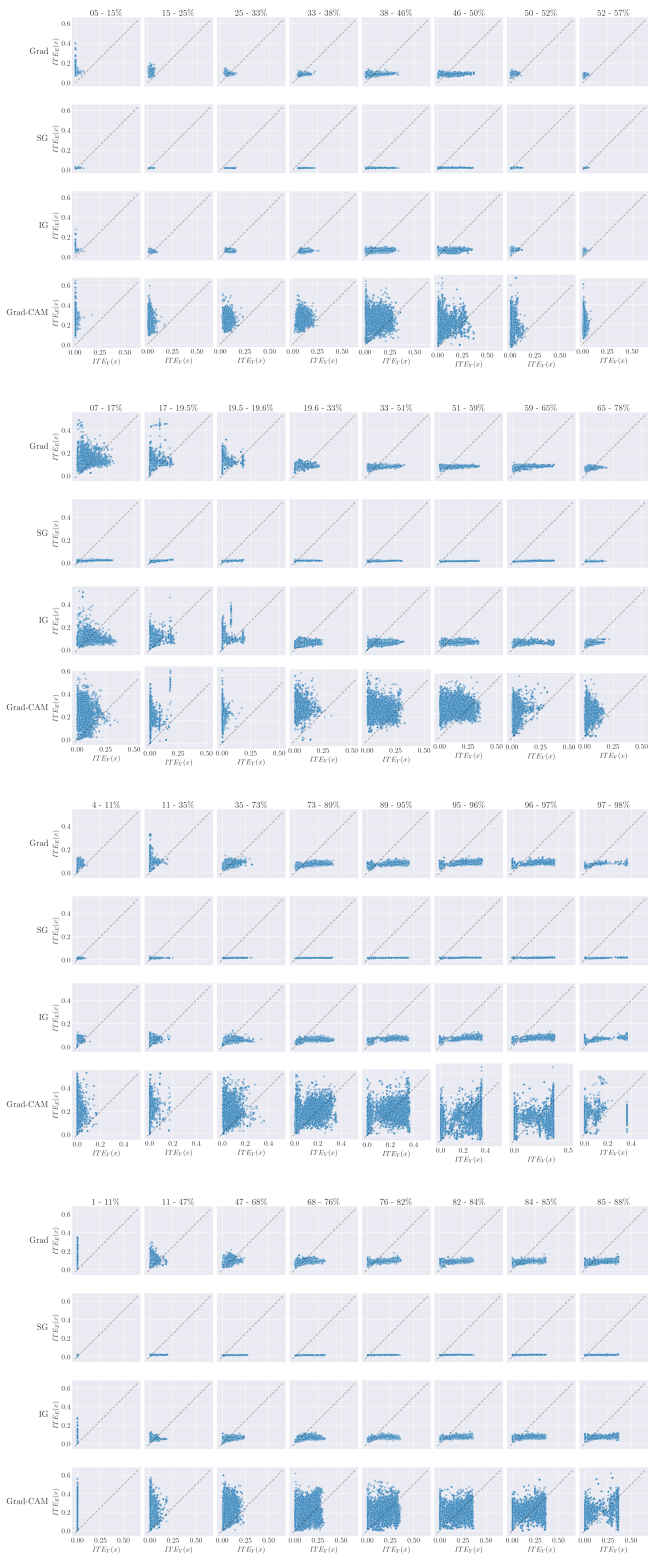

Figure 17: Each column is a subset of models at each accuracy bucket, each row is different explanation methods (row 1: CIFAR10; row 2: SVHN; row 3: MNIST; row 4: FASHION). Whereas low-performing models (first column) show little change in predictions as their explanations differ, top-performing models show the reverse of this trend.

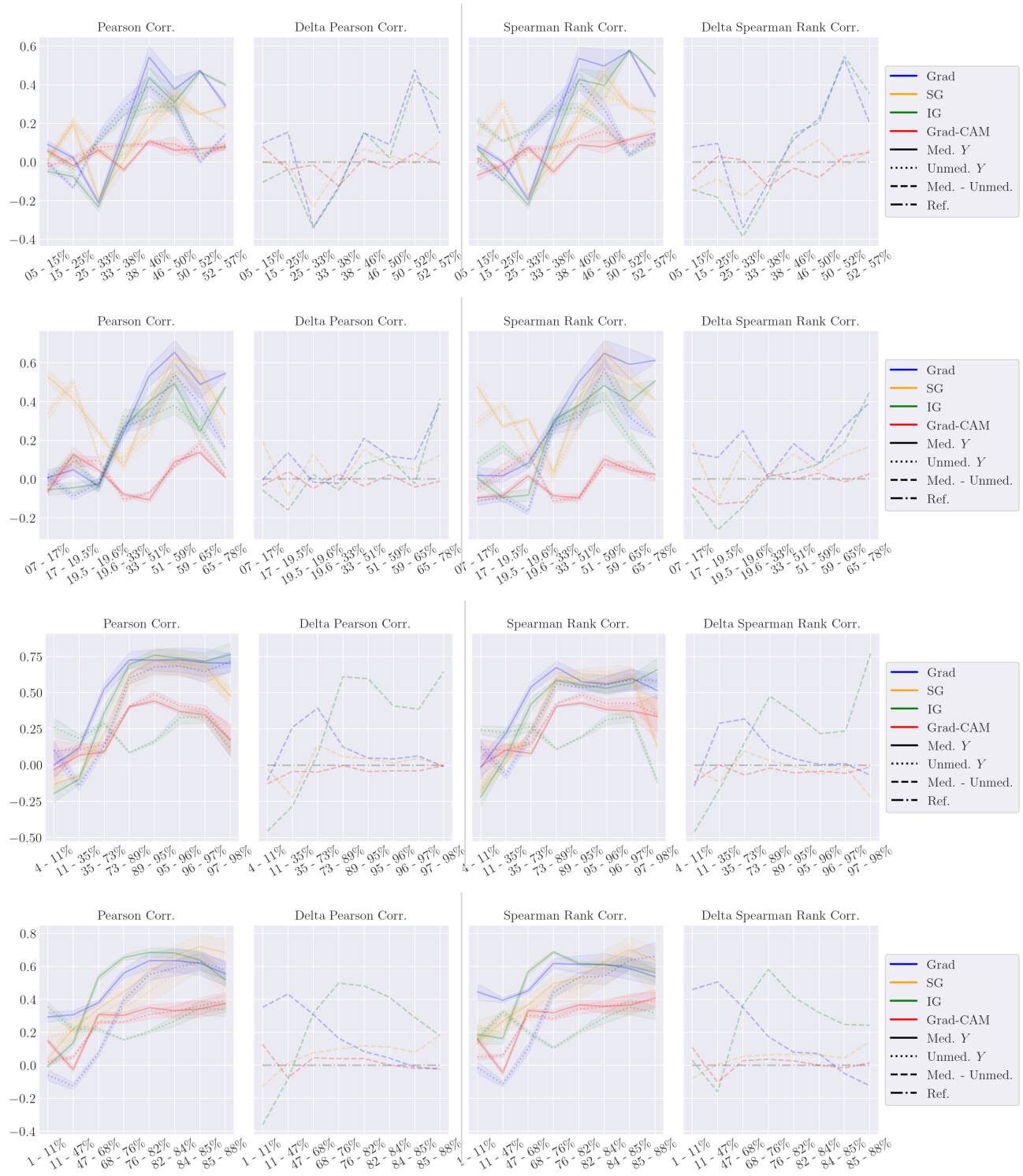

Figure 18: Pearson correlation and Spearman's Rank correlation for ITE of Y and ITE of E across different explanation methods and model performance buckets, for mediated and unmediated Y (row 1: CIFAR10; row 2: SVHN; row 3: MNIST; row 4: FASHION). Absolute values of correlation values are smaller across both datasets (max around 0.5), suggesting that E takes influence from H that does not necessarily pass through Y. The final absolute correlation is going down for top-performing models in both datasets. The increase in delta correlation between mediated and unmediated Y suggests that the direct impact of Y on E is becoming even more important in top-performing models, even more so for SVHN than for CIFAR10.

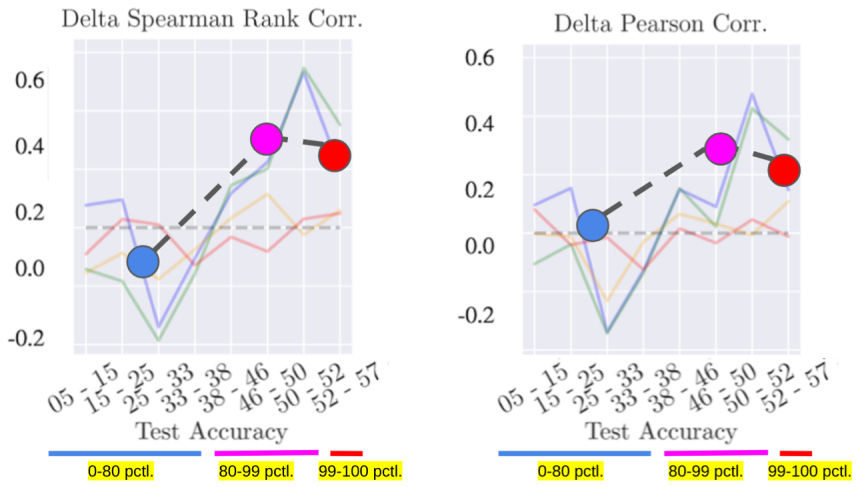

Figure 19: Pearson correlation and Spearman's Rank correlation for ITE of Y and ITE of E across different explanation methods and three model performance buckets, for mediated and unmediated Y.