2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) | 978-1-6654-0915-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/WACV51458.2022.00393

Evaluating and Mitigating Bias in Image Classifiers: A Causal Perspective Using Counterfactuals

Saloni Dash
Microsoft Research India
Bangalore, Karnataka, India
t-sadash@microsoft.com

Vineeth N Balasubramanian,
Indian Institute of Technology
Hyderabad, Telangana, India
vineethnb@iith.ac.in

Amit Sharma
Microsoft Research India
Bangalore, Karnataka, India
amshar@microsoft.com

Abstract

Counterfactual examples for an input—perturbations that change specific features but not others—have been shown to be useful for evaluating bias of machine learning models, e.g., against specific demographic groups. However, generating counterfactual examples for images is non-trivial due to the underlying causal structure on the various features of an image. To be meaningful, generated perturbations need to satisfy constraints implied by the causal model. We present a method for generating counterfactuals by incorporating a structural causal model (SCM) in an improved variant of Adversarially Learned Inference (ALI), that generates counterfactuals in accordance with the causal relationships between attributes of an image. Based on the generated counterfactuals, we show how to explain a pre-trained machine learning classifier, evaluate its bias, and mitigate the bias using a counterfactual regularizer. On the Morpho-MNIST dataset, our method generates counterfactuals comparable in quality to prior work on SCM-based counterfactuals (DeepSCM), while on the more complex CelebA dataset our method outperforms DeepSCM in generating high-quality valid counterfactuals. Moreover, generated counterfactuals are indistinguishable from reconstructed images in a human evaluation experiment and we subsequently use them to evaluate the fairness of a standard classifier trained on CelebA data. We show that the classifier is biased w.r.t. skin and hair color, and how counterfactual regularization can remove those biases.

1. Introduction

A growing number of studies have uncovered biases in image classifiers, particularly against marginalized demographic groups [3, 15, 43, 2]. To avoid such biases, it is important to explain a classifier's predictions and evaluate its fairness. Given any pre-trained machine learning (ML) classifier, counterfactual reasoning is an important way to explain the classifier's decisions and to evaluate its fairness. Counterfactual reasoning involves simulating an alternative input with some specific changes compared to the original input. For example, to evaluate fairness of a classifier with respect to a sensitive attribute like *skin color*, we can ask how the classifier's output will change if a face that was originally *dark-skinned* is made *light-skinned* while keeping everything else constant. Since the only change to the input is the sensitive attribute, counterfactual reasoning is considered more robust than comparing available faces with different skin colors (association) or comparing simulated inputs with *light skin* or *dark skin* (intervention) since these comparisons may include changes in addition to *skin color*.

However, generating counterfactual (CF) examples for images is non-trivial. For instance, consider the simple task of changing a person's hair color in an image of their face. While Generative Adversarial Networks (GANs) can generate new realistic faces with any hair color [23], they are unable to generate the precise changes needed for a CF, i.e. changing hair color without changing hair style or other features of the face. Other explanation techniques based on perturbations such as occluding pixels [42] also do not support counterfactual reasoning based on high-level concepts.

There have been recent efforts on using GANs to generate counterfactuals using an added inference step (encoder). Given a pre-trained GAN model, Denton *et al*. [7] trained an encoder to match the input of a generated image. However, the latents thus encoded do not directly correspond to the given attributes of an image, and it is difficult to change a specific known attribute to generate a counterfactual. To change an attribute, Joo and Kärkkäinen [17] used the FaderNetwork architecture that inputs attributes of an image separately to the generator. However, their method does not account for causal relationships between attributes. Besides, while both these works use generated images to evaluate biases in a classifier, they do not provide any method to mitigate the found biases. We present a method for generating counterfactuals that is based on their formal causal definition, and present a novel counterfactual-based regularizer to mitigate biases in a given classifier.

Formally, a valid counterfactual example for an image is

978-1-6654-0915-5/21/$31.00 ©2021 IEEE
DOI 10.1109/WACV51458.2022.00393

defined with respect to a Structural Causal Model (SCM) over its attributes. An SCM encodes the domain knowledge about how attributes affect each other in the form of a graph with attributes as the nodes and accompanying functional equations connecting the nodes. Generating a counterfactual, therefore, requires modeling both the underlying SCM for the attributes as well as the generative process that uses the attributes to model the resulting image.

In this paper, we present *ImageCFGen*, a method that combines knowledge from a causal graph and uses an inference mechanism in a GAN-like framework to generate counterfactual images. We first evaluate *ImageCFGen* on the Morpho-MNIST dataset to show that it generates counterfactual images comparable to a prior SCM-based CF generation method (DeepSCM) [32]. Moreover, we show that our method is capable of generating high-quality valid counterfactuals for complex datasets like CelebA in comparison to DeepSCM. Specifically, on the CelebA dataset, *ImageCFGen* can generate CFs for facial attributes like *Black Hair*, *Pale Skin* etc. Based on these counterfactual images, we show that an image classifier for predicting attractiveness on the CelebA dataset exhibits bias with respect to *Pale Skin*, but not with respect to attributes like *Wearing Necklace*. We hence propose and demonstrate a bias mitigation algorithm which uses the counterfactuals to remove the classifier's bias with respect to sensitive attributes like *Pale Skin*. In summary, our contributions include:

- *ImageCFGen*, a method that uses inference in a GAN-like framework to generate counterfactuals based on attributes learned from a known causal graph.
- Theoretical justification that under certain assumptions, CFs generated by *ImageCFGen* satisfy the definition of a counterfactual as in Pearl [33].
- Detailed experiments on Morpho-MNIST and CelebA datasets that demonstrate the validity of CFs generated by *ImageCFGen* in comparison to DeepSCM [32].
- Evaluating fairness of an image classifier and explaining its decisions using counterfactual reasoning.
- A regularization technique using CFs to mitigate bias w.r.t. sensitive attributes in any image classifier.

2. Related Work

Our work bridges the gap between generating counterfactuals and evaluating fairness of image classifiers.

Counterfactual Generation. Given the objective to generate a data point X (e.g., an image) based on a set of attributes A, Pearl's ladder of causation [35] describes three types of distributions that can be used: *association* $P(X|A = a)$, *intervention* $P(X|\operatorname{do}(A = a))$ and *counterfactual* $P(X_{A=a}|A = a', X = x')$. In the associational distribution $P(X|A = a)$, X is conditioned on a specific attribute value a in the observed data. For e.g., conditional GAN-based methods [31] implement association between attributes and the image. In the interventional distribution $P(X|\operatorname{do}(A = a))$, A is changed to the value a irrespective of its observed associations with other attributes. Methods like CausalGAN [23] learn an interventional distribution of images after changing specific attributes, and then generate new images that are outside the observed distribution (e.g., women with moustaches on the CelebA faces dataset [23]).

The counterfactual distribution $P(X_{A=a}|A = a', X = x')$ denotes the distribution of images related to a specific image x' and attribute a', if the attribute of the same image is changed to a different value a. In general, generating counterfactuals is a harder problem than generating interventional data since we need to ensure that everything except the changed attribute remains the same between an original image and its counterfactual. Pawlowski *et al*. [32] recently proposed a method for generating image counterfactuals using a conditional Variational Autoencoder (VAE) architecture. While VAEs allow control over latent variables, GANs have been more successful over recent years in generating high-quality images. Thus, we posit that a GAN-based method is more ideally suited to the task of CF generation in complex image datasets, especially when the generated images need to be realistic to be used for downstream applications such as fairness evaluation. We test this hypothesis in Section 5.

Independent of our goal, there is a second interpretation of a "counterfactual" example w.r.t. a ML classifier [40], referring to the smallest change in an input that changes the prediction of the ML classifier. [27] use a standard GAN with a distance-based loss to generate CF images close to the original image. However, the generated CFs do not consider the underlying causal structure – terming such images as CFs can be arguable from a causal perspective. Besides, these CFs are not perceptually interpretable – ideally, a counterfactual image should be able to change only the desired attribute while keeping the other attributes constant, which we focus on in this work.

Fairness of Image Classifiers. Due to growing concerns on bias against specific demographics in image classifiers [3], methods have been proposed to inspect what features are considered important by a classifier [39], constrain classifiers to give importance to the correct or unbiased features [37], or enhance fairness by generating realistic images from under-represented groups [30]. Explanations, to study the fairness of a trained model, can also be constructed by perturbing parts of an image that change the classifier's output, e.g., by occluding an image region [42] or by changing the smallest parts of an image that change the classifier's output [13, 29, 14]. Such perturbation-based methods, however, are not designed to capture high-level concepts and do not enable study of fairness of classifiers w.r.t. human-understandable concepts (e.g. gender or race).

Existing work closest to our efforts include two adversarially-trained generative models to generate coun-

terfactuals for a given image. [7] changed attributes for a given image by linear interpolations of latent variables using a standard progressive GAN [19]. Similarly, [17] used a Fader network architecture [25] to change attributes. However, both these works ignore the causal structure associated with attributes of an image. In analyzing bias against an attribute, it is important to model the downstream changes *caused* by changing that attribute [24]. For instance, for a chest MRI classifier, age of a person may affect the relative size of their organs [32]; it will not be realistic to analyze the effect of age on the classifier's prediction without learning the causal relationship from age to organ size. Hence, in this work, we present a different architecture that can model causal relationships between attributes and provide valid counterfactuals w.r.t. an assumed structural causal model. In addition, using these counterfactuals, we present a simple regularization technique that can be used to decrease bias in any given classifier.

3. SCM-Based Counterfactuals

Let $\mathbf{X} = \mathbf{x} \in \mathcal{X}$ denote the image we want to generate the counterfactual for, and let $\mathbf{A} = \mathbf{a} = \{a_i\}_{i=1}^{n} \in \mathcal{A}$ be its corresponding attributes. In the case of human faces, attributes can be binary variables ($\in \{0, 1\}$) like *Smiling*, *Brown Hair*; or in the case of MNIST digits, continuous attributes ($\in \mathbb{R}$) like *thickness*, *intensity*, etc. A continuous attribute is scaled so that it lies in the range of $[0, 1]$. We have a training set $\mathcal{D}$ containing ($\mathbf{x}$, $\mathbf{a}$) (image, attribute) tuples. Given an image ($\mathbf{x}$, $\mathbf{a}$), the goal is to generate a counterfactual image with the attributes changed to $\mathbf{a}_c$.

3.1. SCM over Attributes

We assume an SCM for the true data-generating process that defines the relationship among the attributes $\mathbf{a}$, and from the attribute to the image $\mathbf{x}$. For instance, with two binary attributes (*Young*, *Gray Hair*) for an image $\mathbf{x}$ of a face, the true causal graph can be assumed to be *Young* $\rightarrow$ *Gray Hair* $\rightarrow \mathbf{x}$. We separate out the graph into two parts: relationships amongst the attributes ($\mathcal{M}_a$), and relationships from the attributes to the image ($\mathcal{M}_x$). We call $\mathcal{M}_a$ as the *Attribute-SCM* and model $\mathcal{M}_x$ as a generative model, given the attributes.

The Attribute-SCM ($\mathcal{M}_a$) consists of a causal graph structure and associated structural assignments. We assume that the causal graph structure is known. Given the graph structure, Attribute-SCM learns the structural assignments between attributes. E.g., given *Young* $\rightarrow$ *Gray Hair*, it learns the function g such that *Gray Hair* $= g($*Young*$, \epsilon)$, where ϵ denotes independent random noise. We use the well-known Maximum Likelihood Estimation procedure in Bayesian Networks [8] to estimate these functions for the attributes, but other methods such as Normalizing Flows [36, 32] can also be used.

Note that counterfactual estimation requires knowledge of both the *true* causal graph and the *true* structural equations; two SCMs may entail exactly the same observational and interventional distributions, but can differ in their counterfactual values [18]. In most applications, however, it is impractical to know the true structural equations for each edge of a causal graph. Therefore, here we make a simplifying empirical assumption: while we assume a known causal graph for attributes, we estimate the structural equations from observed data. That is, we assume that the data is generated from a subset of all possible SCMs (e.g., linear SCMs with Gaussian noise) such that the structural equations can be uniquely estimated from data. Details on Attribute-SCM are in Appendix A.

3.2. Image Generation from Attribute-SCM

For modeling the second part of the SCM from attributes to the image ($\mathcal{M}_x$), we build a generative model that contains an encoder-generator architecture. Given an Attribute-SCM (either provided by the user or learned partially, as in Sec 3.1), the proposed method *ImageCFGen* has three steps to generate a counterfactual for an image $(\mathbf{x}, \mathbf{a})$ such that the attributes are changed to $\mathbf{a}'$ (architecture in Figure 1).

- An encoder $E : (\mathcal{X}, \mathcal{A}) \rightarrow \mathbf{Z}$ infers the latent vector $\mathbf{z}$ from $\mathbf{x}$ and $\mathbf{a}$, i.e. $\mathbf{z} = E(\mathbf{x}, \mathbf{a})$ where $\mathbf{Z} = \mathbf{z} \in \mathbb{R}^m$.
- The Attribute-SCM intervenes on the desired subset of attributes that are changed from $\mathbf{a}$ to $\mathbf{a}'$, resulting in output $\mathbf{a}_c$. Specifically, let $\mathbf{a}_k \subseteq \mathbf{a}$ be the subset of attributes that changed between the inputs $\mathbf{a}$ and $\mathbf{a}'$. For every $a_i \in \mathbf{a}_k$, set its value to a'_i, then change the value of its descendants in the SCM graph by plugging in the updated values in the structural equations (see Lines 5-6 in Algorithm 1, Appendix A).
- Generator $G : (\mathbf{Z}, \mathcal{A}) \rightarrow \mathcal{X}$ takes as input $(\mathbf{z}, \mathbf{a}_c)$ and generates a counterfactual $\mathbf{x}_c$, where $\mathbf{z} \in \mathbf{Z} \subseteq \mathbb{R}^m$.

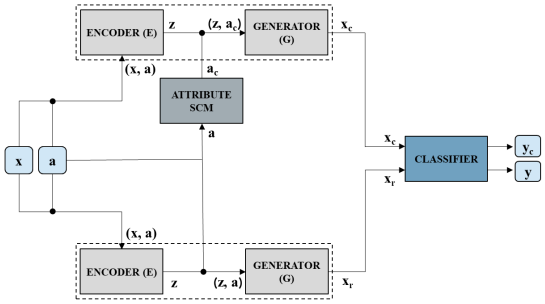


Figure 1: **Counterfactual Generation using *ImageCFGen***. The top half of the figure shows the CF generation procedure, and the bottom half of the figure shows the reconstruction procedure. Finally, the reconstructed image x_r and the counterfactual image x_c are used for a downstream task like fairness evaluation of a classifier.

The above method for CF generation can be written as:

$$\mathbf{x}_c = G(E(\mathbf{x}, \mathbf{a}), \mathbf{a}_c), \tag{1}$$

The complete algorithm is shown in Algorithm 1 in Appendix A. For our experiments, we use a novel improved variant of Adversarially Learned Inference [11] to train the encoder and generator. However, *ImageCFGen* can be extended to any encoder-generator (decoder) architecture.

3.3. Correspondence to Counterfactual Theory

The above architecture maps directly to the three steps for generating a valid counterfactual, for $(\mathbf{x}, \mathbf{a})$ as in [33]:

- **Abduction**: Infer latent $\mathbf{z}$ given the input $(\mathbf{x}, \mathbf{a})$ using the encoder.
- **Action**: Let $\mathbf{a}_k \subseteq \mathbf{a}$ be the set of k attributes that one wants to intervene on. Set attribute $a_i \rightarrow a_i' \; \forall a_i \in \mathbf{a}_k$, where $\mathbf{a}_k' = \{a_i'\}_{i=1}^k$.
- **Prediction**: Modify all the descendants of $\mathbf{a}_k$ according to the SCM equations learned by Attribute-SCM. This outputs $\mathbf{a}_c$, the intervened attributes. Use $\mathbf{z}$ from the encoder and $\mathbf{a}_c$ from the Attribute-SCM and input it to the generator to obtain the counterfactual x_c.

The proof that Equation 1 corresponds to generating a valid counterfactual is in Appendix B.

3.4. Implementing the Encoder and Generator

Many studies have reported generating high-quality, realistic images using GANs [41, 19, 20]. However, vanilla GANs lack an inference mechanism where the input $\mathbf{x}$ can be mapped to its latent space representation $\mathbf{z}$. We hence use Adversarially Learned Inference (ALI) [11], which integrates the inference mechanism of variational methods like VAEs in a GAN-like framework, thereby leveraging the generative capacity of GANs as well as providing an inference mechanism. We generate the images using a conditional variant of ALI where the model is conditioned on the attributes $\mathbf{a}$ while generating an image.

An ALI-based method, however, has two limitations: (1) generation capabilities are limited when compared to state-of-the-art [20]; and (2) reconstructions are not always faithful reproductions of the original image [11]. Image reconstructions are important in the counterfactual generation process because they indicate how good the inferred latent space variable z is, which is used in the abduction step of generating counterfactuals. We address both these issues by using a style-based generator for better generation and a cyclic cost minimization algorithm for improved reconstructions. We refer to our ALI model as Cyclic Style ALI (CSALI). We describe each of these components below.

Adversarially Learned Inference. ALI uses an encoder E and a generator G in an adversarial framework, where the encoder learns to approximate the latent space distribution from the input $\mathbf{x}$ and attributes $\mathbf{a}$, and the generator learns to generate realistic images from the latent space distribution and attributes $\mathbf{a}$. The discriminator D is optimized to differentiate between pairs of tuples containing {the real image, the corresponding approximated latent space variable, attributes} from joint samples of {the generated image, the input latent variable, attributes}. The Encoder and Generator are optimized to fool the discriminator. Unlike [11] which uses an embedding network, we directly pass the attributes to the Generator, Encoder and Discriminator since we found that it helped in conditioning the model on the attributes better. The conditional ALI hence optimizes:

$$\begin{aligned}\min_{G,E}\max_{D} V(D,G,E) = \mathbb{E}_{q(\mathbf{x})p(\mathbf{a})}[\log(D(\mathbf{x}, E(\mathbf{x},\mathbf{a}),\mathbf{a}))] \\ +\mathbb{E}_{p(\mathbf{z})p(\mathbf{a})}[\log(1 - D(G(\mathbf{z},\mathbf{a}),\mathbf{z},\mathbf{a}))]\end{aligned} \tag{2}$$

where G is the generator, E is the encoder and D is the discriminator. $q(\mathbf{x})$ is the distribution for the images, $p(z) \sim \mathcal{N}(0,1)$ and $p(a)$ is the distribution of the attributes. Image reconstructions are defined as:

$$\mathbf{x}_r = G(E(\mathbf{x},\mathbf{a}),\mathbf{a}) \tag{3}$$

Style-Based Generator. Style-based generator architectures (StyleGANs) implicitly learn separations of high-level attributes of images like hair color, pose, etc [20, 21], and generate images that are indistinguishable from real images [44]. To improve generation, we replace the ALI generator with the style-based generator architecture in [20]. Details of the architecture are provided in the Appendix D.

Cyclic Cost Minimization. To improve image reconstructions (which in turn indicate the goodness of the learned latents $\mathbf{z}$), we employ the cyclic cost minimization algorithm in [9] after training the style-based ALI model. The generator is fixed, and the encoder is fine-tuned to minimize a reconstruction loss computed using: (i) error in the image space $\mathcal{L}_{\mathbf{x}} = \mathbb{E}_{\mathbf{x}\sim q(\mathbf{x})} \|\mathbf{x} - G(E(\mathbf{x},\mathbf{a}))\|_2$; and (ii) error in the latent space $\mathcal{L}_{\mathbf{z}} = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \|\mathbf{z} - E(\mathbf{x},\mathbf{a})\|$, where $G(E(\mathbf{x},\mathbf{a}))$ is the reconstructed image $\mathbf{x}_r$ according to Eqn 3 and $E(\mathbf{x},\mathbf{a})$ is the encoder's output $\mathbf{z}_r$, which is expected to capture the image's latent space distribution. We fine-tune the encoder using the above reconstruction loss *post-hoc* after obtaining a good generator in order to explicitly improve image reconstructions.

4. Applications of Generated CFs

We now show how the counterfactuals generated using *ImageCFGen*can be used to evaluate fairness of, as well as explain a given image classifier. We will also present a method to mitigate any fairness biases in the classifier. Suppose we are given a pre-trained image classifier $\hat{f} : \mathcal{X} \rightarrow \mathcal{Y}$, such that $\hat{f}(\mathbf{x}) = \hat{y}$, where $\mathbf{x} \in \mathcal{X}$ refers to the images and $\hat{y} \in \mathcal{Y}$ refers to the classifier's discrete outcome. Let $\mathbf{a} \in \mathcal{A}$ be the corresponding image attributes, and let $\mathbf{a}_S \in \mathcal{A}_S \subseteq \mathcal{A}$ be the set of sensitive attributes we want to evaluate the classifier on.

4.1. Evaluating Fairness of a Classifier

We can use the generated CFs to estimate biases in a given classifier that predicts some discrete outcome $\hat{y} = y$ (like *Attractive*). In an ideal scenario, the latent variable $\mathbf{z}$

for the real image and its reconstructed image would match exactly. However, experiments using ALI demonstrate that the reconstructed images are not perfect reproductions of the real image [11, 10, 6]. Therefore, for objective comparison, we compare classification labels for reconstructed images ($\mathbf{x}_r$ from Eqn 3) and counterfactual images ($\mathbf{x}_c$ from Eqn 1), since the reconstructed images share the exact same latent $\mathbf{z}$ as the CF image (and hence the CF will be valid). We hence refer to the reconstructed images (which share the latent $\mathbf{z}$ with the CF) as *base* images for the rest of the paper.

We characterize a classifier as *biased* w.r.t. an attribute if: (a) it changes its classification label for the CF image (obtained by changing that attribute); and (b) if it changes the label to one class from another class more often than vice versa (for CFs across test images obtained by changing the considered attribute). (To illustrate the second condition, if setting hair color as blonde makes test images consistently be classified as attractive more often than otherwise, this indicates bias.) We capture these intuitions as a formula for the degree of bias in a binary classifier w.r.t. a considered attribute:

$$\begin{aligned}\text{bias} = p(y_r \neq y_c)(&p(y_r = 0, y_c = 1 | y_r \neq y_c) \\ &- p(y_r = 1, y_c = 0 | y_r \neq y_c))\end{aligned} \tag{4}$$

where y_r is the classification label for the reconstructed image, and y_c is the classification label for the CF image. Using Bayes Theorem, Eqn 4 reduces to:

$$\text{bias} = p(y_r = 0, y_c = 1) - p(y_r = 1, y_c = 0) \tag{5}$$

The bias defined above ranges from -1 to 1. It is 0 in the ideal scenario when the probability of CF label changing from 0 to 1 and vice-versa is the same (=0.5). The bias is 1 in the extreme case when the CF label always changes to 1, indicating that the classifier is biased *towards* the counterfactual change. Similarly if the CF label always changes to 0, the bias will be -1, indicating that the classifier is biased *against* the counterfactual change. In Appendix C, we show that a classifier with zero bias in the above metric is fair w.r.t. the formal definition of counterfactual fairness [24].

4.2. Explaining a Classifier

We can also use the CFs from *ImageCFGen* to generate explanations for a classifier. For any input $\mathbf{x}$, a local *counterfactual importance score* for an attribute $\mathbf{a}_i$ states how the classifier's prediction changes upon changing the value of $\mathbf{a}_i$. Assuming $\mathbf{a}_i$ can take binary values $a' = 1$ and $a = 0$, the local *CF importance score* of $\mathbf{a}_i$ is given by:

$$\begin{aligned}&\mathbb{E}_Y[Y_{\mathbf{a}_i \leftarrow a'} | \mathbf{x}, \mathbf{a}] - \mathbb{E}_Y[Y_{\mathbf{a}_i \leftarrow a} | \mathbf{x}, \mathbf{a}] \\ &= y_{\mathbf{a}_i \leftarrow a'} | \mathbf{x}, \mathbf{a} - y_{\mathbf{a}_i \leftarrow a} | \mathbf{x}, \mathbf{a}\end{aligned} \tag{6}$$

where Y is the random variable for the classifier's output, y is a value for the classifier's output, and the above equality is for a deterministic classifier. For a given $(\mathbf{x}, \mathbf{a})$, the score for each attribute (feature) can be ranked to understand the relative importance of features. To find global feature importance, we average the above score over all inputs.

4.3. Bias Mitigation for a Classifier

Finally, in addition to evaluating a classifier for bias, CFs generated using *ImageCFGen*can be used to remove bias from a classifier w.r.t. a sensitive attribute. Here we propose a *counterfactual* regularizer to ensure that an image and its counterfactual over the sensitive attribute obtain the same prediction from the image classifier. For an image $\mathbf{x}$, let $logits(\mathbf{x})$ be the output of the classifier $\hat{f}$ before the sigmoid activation layer. To enforce fairness, we can finetune the classifier by adding a regularizer that the logits of the image and its counterfactual should be the same, i.e.

$$\text{BCE}(y_{true}, \hat{f}(\mathbf{x})) + \lambda \text{MSE}(\text{logits}(\mathbf{x}_r), \text{logits}(\mathbf{x}_c)) \tag{7}$$

where BCE is the binary cross-entropy loss, y_{true} is the ground truth label for the real image $\mathbf{x}$, λ is a regularizing hyperparameter, MSE is the mean-squared error loss, and $\mathbf{x}_r$ and $\mathbf{x}_c$ are defined in Eqns 3 and 1 respectively.

5. Experiments and Results

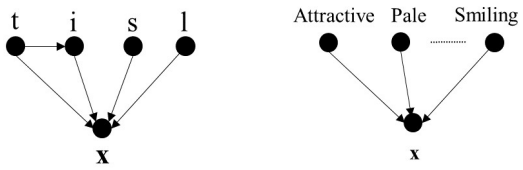
Considering the limited availability of datasets with known causal graphs, we study *ImageCFGen* on the Morpho-MNIST dataset (a simple dataset for validating our approach), and on the CelebA dataset (which provides an important context for studying bias and fairness in image classifiers). Specifically, we study the following:

- **Validity of *ImageCFGen* CFs.** We use the Morpho-MNIST dataset which adds causal structure on the MNIST images, to compare counterfactuals from *ImageCFGen* to the Deep-SCM method [32]. We show that CFs from *ImageCFGen* are comparable to those from DeepSCM, thus validating our approach.
- **Quality of *ImageCFGen* CFs.** On the more complex CelebA dataset, we evaluate the quality of *ImageCFGen* CFs by quantifying the generation and reconstruction, using established benchmark metrics. We find that using the proposed CSALI architecture offers significant advantages over the standard ALI model. We also contrast the quality and validity of the generated CFs with those of DeepSCM, and find that *ImageCFGen* outperforms DeepSCM.
- **Fairness Evalution and Explanation of a ML Classifier Using *ImageCFGen* CFs.** We show using *ImageCFGen* that a standard pre-trained classifier on CelebA that predicts whether a face is attractive or not, has bias w.r.t. attribute *Pale Skin* across all three hair colors (*Black Hair*, *Blond Hair*, *Brown Hair*). We also explain the classifier's predictions using CFs.
- **Bias Mitigation of a ML Classifier Using *ImageCFGen* CFs.** Finally, we show how our proposed method can be used to decrease detected bias in the classifier for the attributes mentioned above.

Baselines and Performance Metrics. We compare to DeepSCM's [32] results on Morpho-MNIST and CelebA.

We present both quantitative and qualitative performance of our method for these datasets. While we follow the metrics of [32] for Morpho-MNIST, for CelebA we report quantitative scores like Fréchet Inception Distance [16] (FID) and Mean Squared Error (MSE) to compare generation and reconstruction quality with the base ALI method. For measuring quality of generated counterfactuals, we report human evaluation scores, in addition to qualitative results. For bias evaluation, we compare *ImageCFGen* to affine image transformations like horizontal flip and brightness that are commonly used data augmentation techniques.

Datasets. *Morpho-MNIST* [4] is a publicly available dataset based on MNIST [26] with interpretable attributes like *thickness*, *intensity*, etc. It was extended by [32] to introduce morphological transformations with a known causal graph. The attributes are *thickness (t)* and *intensity (i)*, where *thickness* $\rightarrow$ *intensity* ($\rightarrow$ indicating causal effect). We extend this dataset by introducing an independent morphological attribute—*slant (s)* from the original Morpho-MNIST dataset and digit *label (l)* as an attribute. The causal graph for the dataset is given in Fig 2a.



(a) Morpho-MNIST (b) CelebA.

Figure 2: **Causal Graphs for Morpho-MNIST and CelebA.** Attributes for Morpho-MNIST are *thickness* t, *intensity* i, *slant* s and *label* l; for CelebA are *Pale*, *Black Hair*, etc. In both graphs, attributes cause the image x.

CelebA [28] is a dataset of 200k celebrity images annotated with 40 attributes like *Black Hair*, *Wearing Hat*, *Smiling* etc. We train an image classifier on the dataset that predicts the attribute *Attractive* as done in [38, 12]. While explaining the classifier's decisions, we generate CFs for all attributes excluding *Male*, *Young* and *Blurry*. For fairness evaluations, we focus on generating CFs for the attributes *Black Hair*, *Blond Hair*, *Brown Hair*, *Pale* and *Bangs*. Similar to [7], we do not generate CFs for *Male* because of inconsistent social perceptions surrounding gender, thereby making it difficult to define a causal graph not influenced by biases. Therefore, all attributes we consider have a clear causal structure (Fig 2b shows the causal graph). Additionally, our method can also be utilized in the setting where the attributes are connected in a complex causal graph structure, unlike [7, 17]. We show this by conducting a similar fairness and explanation analysis for a *Attractive* classifier in Appendix O, where *Young* affects other visible attributes like *Gray hair*.

Details of implementation, architecture (including ALI) and training are provided in Appendix D.

5.1. Validity of *ImageCFGen* CFs on Morpho-MNIST

We generate CFs using *ImageCFGen* on images from the Morpho-MNIST dataset by intervening on all four attributes - *thickness*, *intensity*, *slant* and *label* and observe how the image changes with these attributes. Fig 3 demonstrates CFs for a single image with label 0. Along the first column vertically, the label is changed from 0 to $\{1, 4, 6, 9\}$ while the thickness, intensity and slant are kept constant. Then, as we proceed to the right in each row, the attributes of thickness, intensity and slant are changed sequentially but the label is kept constant. Visually, the generated counterfactuals change appropriately according to the intervened attributes. For instance, according to the causal graph in Fig 2a, changing the digit label should not change the digit thickness intensity and slant. That is precisely observed in the first column of Fig 3. Whereas, changing the thickness should also change the intensity of the image which is observed in the third and fourth columns of Fig 3. Results of latent space interpolations and digit reconstructions are provided in Appendix E and F. We find that the encoder learns meaningful latent space representations, and the reconstructions are faithful reproductions of Morpho-MNIST digits.

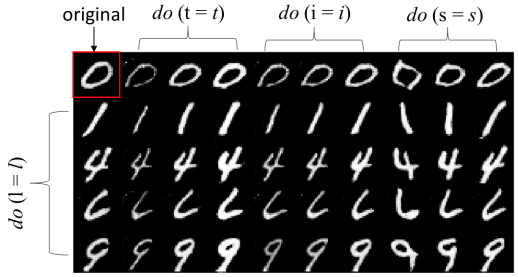


Figure 3: **Morpho-MNIST Counterfactuals.** Top-left cell shows a real image sampled from the test set. Vertically, rows correspond to interventions on the label, *do*(l = 1, 4, 6, 9). Moving horizontally, columns correspond to interventions on thickness: *do* (t=1, 3, 5), intensity: *do* (i = 68, 120, 224), and slant: *do* (s = -0.7, 0, 1) respectively.

To quantify these observations, we randomly sample hundred values for *thickness*, *intensity* and *slant* attributes and generate corresponding CFs for each test image. Fig 4 plots the target ground-truth attribute vs the measured attribute in the CF image (measured using a formula from Morpho-MNIST [32]), and shows that all attributes are on an average clustered along the reference line of y = x with some variability. We quantify this variability in Table 1 using median absolute error, by comparing the CFs generated using *ImageCFGen* vs those using the DeepSCM method [32] (we choose Median Absolute Error to avoid skewed measurements due to outliers). *ImageCFGen* and DeepSCM are comparable on attributes of *thickness* and *intensity*, showing the validity of our CFs. We could not compare *slant* since [32] do not use *slant* in their studies.

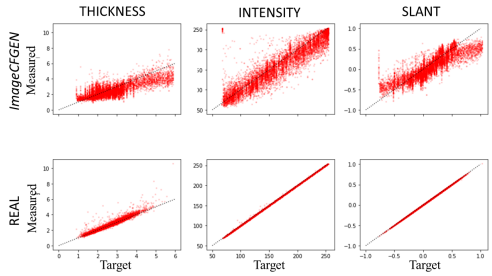


Figure 4: **Morpho-MNIST CFs.** For each attribute, Target (x axis) is the desired value for the CF and Measured (y axis) is the attribute value obtained from the generated counterfactual using a formula provided in the Morpho-MNIST dataset [32]. We compare between ground-truth CFs and CSALI-generated CFs. In an ideal scenario (real samples), points should lie along the $y = x$ line.

	do (thickness)	*do* (intensity)
ImageCFGen	0.408 ± 0.004	$\mathbf{10.740 \pm 0.161}$
DeepSCM	$\mathbf{0.253 \pm 0.002}$	12.519 ± 0.105

Table 1: **Median Abs. Error: *ImageCFGen* and DeepSCM.** Lower is better.

5.2. Quality of *ImageCFGen* CFs on CelebA

We now evaluate *ImageCFGen* on CelebA by comparing the quality of the generated CF images against the vanilla ALI model and the DeepSCM model [32].

Generation Quality. We show real images and their corresponding reconstructions using ALI and CSALI in rows (I), (II) and (IV) of Fig 13 in Appendix K. While the reconstructions of both ALI and CSALI are not perfect, those of CSALI are significantly better than ALI. Moreover, they capture majority of the high-level features of the real images like *Hair Color*, *Smiling*, *Pale*, etc. We quantify this in Table 2 using Fréchet Inception Distance [16] (FID) score for generated images, Mean Squared Error (MSE) between the real images $\mathbf{x}$ and reconstructed images $\mathbf{x}_r$ and Mean Absolute Error (MAE) between the real latent variable $\mathbf{z}$ and the encoder's approximation of the latent variable $\mathbf{z}_r$. We randomly sample 10k generated images to calculate FID scores, and randomly sample 10k real images for which we generate reconstructions and report the MSE and MAE metrics. We report these metrics on the baseline model of ALI as well. We observe a significant improvement in generation quality after using a style-based generator and significant improvements in reconstruction with the cyclic cost minimization algorithm (refer to Appendix K for ablation study). We report MSE on images and MAE on latent space variables since the cyclic cost minimization algorithm [9] uses these metrics to improve reconstructions.

Counterfactual Quality. We contrast the quality and validity of the CFs from *ImageCFGen* with the CFs from DeepSCM [32] (refer to Appendix H for our implementation of DeepSCM on CelebA). To generate the counterfactual images, we intervene on the attributes of *Black Hair*, *Blond Hair*, *Brown Hair*, *Pale* and *Bangs*. We observe in Figure 5 that the CFs from *ImageCFGen* are qualitatively better than those from DeepSCM. *ImageCFGen* CFs successfully change the hair color and skin color, in addition to adding bangs to the face (refer to Figure 12 in Appendix J for more *ImageCFGen* CFs and Appendix I for more comparisons with DeepSCM). In contrast, the CFs from DeepSCM only partially change the hair color and the skin color in columns (a) through (f) and fail to add bangs in column (g).

	FID	MSE (x,x_r)	MAE (z, z_r)
ALI	67.133	0.177	1.938
CSALI	**21.269**	**0.103**	**0.940**

Table 2: **FID, MSE and MAE scores for ALI and Cyclic Style ALI (CSALI).** Lower is better.

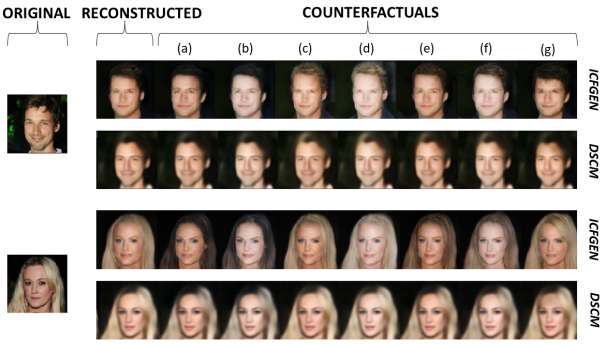


Figure 5: ***ImageCFGen* and DeepSCM Counterfactuals.** (a) denotes *do* (black hair = 1) and (b) denotes *do* (black hair = 1, pale =1). Similarly (c) denotes *do* (blond hair = 1); (d) denotes *do* (blond hair = 1, pale = 1); (e) denotes *do* (brown hair = 1); (hf denotes *do* (brown hair = 1, pale = 1); and (g) denotes *do* (bangs = 1).

We also perform a human evaluation experiment of the generated counterfactuals in Appendix K, which showed that the distribution of counterfactuals is identical to the distribution of their corresponding base images.

5.3. Bias Evaluation & Explaining a Classifier

We train a classifier on the CelebA dataset to predict attractiveness of an image w.r.t. an attribute (architecture and training details in Appendix L). We then use the generated CFs to identify biases in the classifier. We sample 10k points from the test set and generate seven CFs for each of them as shown in Fig 5 for different attributes. We only consider those images for which the corresponding attribute was absent in order to avoid redundancies. For instance, we filter out CF (c) of the second sample from Fig 5 since blond hair was already present in the base image. We then provide the generated CFs along with the base (reconstructed) image to the attractive classifier and compare their labels. As a baseline comparison, we also pass images with affine trans-

formations like horizontal flip and increasing brightness to the classifier. We treat the classifier's outputs as the probability of being labeled as attractive. Fig. 6 shows these probabilities for the base image, affine transformations, and the CFs. If the classifier is fair w.r.t. these attributes, all points should be clustered along the y=x line.

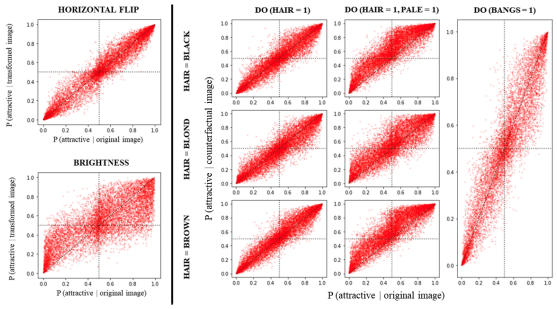


Figure 6: **Fairness Analysis.** Affine transformations *(left)*, CFs *(right)*. Each point in a scatter plot is a pair of a base image and its corresponding CF image. In the ideal case, all points should lie along the y = x line. To analyze the figures, divide the scatter plot into four quadrants formed by lines x = 0.5 and y = 0.5. Any point in the top left quadrant signifies that the attractive label was changed from 0 to 1 and vice-versa for the bottom right quadrant.

For the CF plots, for do (black hair =1, pale = 1) and do (brown = 1, pale = 1) almost all points are above the reference line, suggesting that the probability of being classified as attractive increases after applying these interventions. Not only does the probability increase, for do (black hair = 1, pale = 1), 18% of the CF labels are flipped from the base ones, and *94%* of these labels are changed from not attractive to attractive. In case of do (brown hair = 1, pale = 1), 19% of the CF labels are flipped and *94%* of the labels are flipped from not attractive to attractive. For the CF do (blond hair = 1, pale = 1), 16% of the labels are flipped and 74% of the labels are flipped from attractive to not attractive. In comparison, the affine transformations are unable to provide a clear picture of bias. For horizontal flip, the points are equally distributed on both sides of the reference line y = x. In the case of brightness, there is more variation.

In Table 3, we quantify these observations using Eqn 4. Our metric for bias measurement gives an overall estimate of the bias in classifier, and provides an interpretable and uniform scale to compare biases among different classifiers. The reported bias values reflect the observations from Fig 6. We observe that the CF of do (brown hair = 1, pale = 1) has the highest positive bias amongst all CFs and affine transformations, i.e. the classifier is biased towards labeling these CFs as more attractive in contrast to the base image. Using CFs, we are able to detect other significant biases towards setting skin color as $pale = 1$ for all hair colors (black, blond and brown). In contrast, using the baseline transformations, we are unable to detect skin color bias in the classifier since the calculated bias values are negligible.

	$p(a_r \neq a_c)$	$p(0 \rightarrow 1)$	bias
horizontal_flip	0.073	0.436	-0.009
brightness	0.192	0.498	-0.001
black_h	0.103	0.586	0.018
black_h, pale	0.180	0.937	**0.158**
blond_h	0.115	0.413	- 0.02
blond_h, pale	0.155	0.738	**0.073**
brown_h	0.099	0.704	0.041
brown_h, pale	0.186	0.942	**0.164**
bangs	0.106	0.526	0.005

Table 3: **Bias Estimation.** Bias values above a threshold of 5% are considered significant.

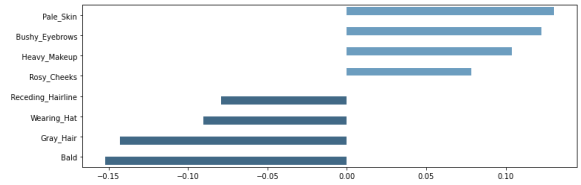


Figure 7: **Explaining a Classifier.** Attribute ranking of top 4 positive and top 4 negative influential attributes.

We also use CFs to explain the classifier's decisions for predicting attractiveness of a face. Fig 7 shows the top 4 positive and top 4 negative influences for classifying a face as attractive. We can see that *Pale* skin is the top attribute that contributes to the classifier predicting a face as attractive, while *Bald* is the top attribute that contributes to the classifier prediction of not attractive.

5.4. Bias Mitigation for a Classifier

Finally, using Eqn. 7, we employ the generated CFs to remove the identified biases in the *Attractive* classifier on do (black = 1, pale = 1), do (blond = 1, pale = 1) and do (brown = 1, pale = 1) . The CF-regularized classifier reports a bias score of 0.032 for black hair and pale (against 0.159 for the original classifier) and 0.012 for brown hair and pale (against 0.154 for the original classifier). Also, the reduced biases are no longer significant, without reducing the accuracy (82.3% versus 82.6%). Details are in Appendix M.

6. Conclusion

We propose a framework *ImageCFGen* for generating counterfactuals based on an underlying SCM, utilizing the generative capacity of GANs. We demonstrate how the counterfactuals can be used to evaluate and mitigate a classifier's biases and explain its decisions. That said, we acknowledge two limitations , 1) Our CF generation method relies on accurate knowledge of the causal graph; 2) It uses a statistical model that can have have unknown failure modes in generating meaningful counterfactuals. Therefore, this work should be considered as a prototype early work in generating counterfactuals, and is not suitable for deployment.

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

[2] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems*, pages 4349–4357, 2016.

[3] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91, 2018.

[4] Daniel C Castro, Jeremy Tan, Bernhard Kainz, Ender Konukoglu, and Ben Glocker. Morpho-mnist: Quantitative assessment and diagnostics for representation learning. *Journal of Machine Learning Research*, 20(178):1–29, 2019.

[5] François Chollet et al. Keras. `https://keras.io`, 2015.

[6] Yatin Dandi, Homanga Bharadhwaj, Abhishek Kumar, and Piyush Rai. Generalized adversarially learned inference. *arXiv preprint arXiv:2006.08089*, 2020.

[7] Emily Denton, Ben Hutchinson, Margaret Mitchell, and Timnit Gebru. Detecting bias with generative counterfactual face attribute augmentation. *arXiv preprint arXiv:1906.06439*, 2019.

[8] Jianguo Ding. Probabilistic inferences in bayesian networks. *InA. Rebai (ed.), Bayesian Network*, pages 39–53, 2010.

[9] Yahya Dogan and Hacer Yalim Keles. Semi-supervised image attribute editing using generative adversarial networks. *Neurocomputing*, 401:338–352, 2020.

[10] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. *arXiv preprint arXiv:1907.02544*, 2019.

[11] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016.

[12] Max Ehrlich, Timothy J Shields, Timur Almaev, and Mohamed R Amer. Facial attributes classification using multi-task representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 47–55, 2016.

[13] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3429–3437, 2017.

[14] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *ICML*, 2019.

[15] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *European Conference on Computer Vision*, pages 793–811. Springer, 2018.

[16] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017.

[17] Jungseock Joo and Kimmo Kärkkäinen. Gender slopes: Counterfactual fairness for computer vision models by attribute manipulation. *arXiv preprint arXiv:2005.10430*, 2020.

[18] Amir-Hossein Karimi, Julius von Kügelgen, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse under imperfect causal knowledge: a probabilistic approach. *Advances in Neural Information Processing Systems*, 33, 2020.

[19] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan, 2020.

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[23] Murat Kocaoglu, Christopher Snyder, Alexandros G Dimakis, and Sriram Vishwanath. Causalgan: Learning causal implicit generative models with adversarial training. In *International Conference on Learning Representations*, 2018.

[24] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in neural information processing systems*, pages 4066–4076, 2017.

[25] Guillaume Lample, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, and Marc'Aurelio Ranzato. Fader networks: Manipulating images by sliding attributes. In *Advances in neural information processing systems*, pages 5967–5976, 2017.

[26] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[27] Shusen Liu, Bhavya Kailkhura, Donald Loveland, and Yong Han. Generative counterfactual introspection for explainable deep learning. In *2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 1–5. IEEE, 2019.

[28] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[29] Ronny Luss, Pin-Yu Chen, Amit Dhurandhar, Prasanna Sattigeri, Yunfeng Zhang, Karthikeyan Shanmugam, and Chun-Chen Tu. Generating contrastive explanations with monotonic attribute functions. *arXiv preprint arXiv:1905.12698*, 2019.

[30] Daniel McDuff, Shuang Ma, Yale Song, and Ashish Kapoor. Characterizing bias in classifiers using generative models. In

Advances in Neural Information Processing Systems, pages 5403–5414, 2019.

[31] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. arxiv 2014. *arXiv preprint arXiv:1411.1784*, 2014.

[32] Nick Pawlowski, Daniel C Castro, and Ben Glocker. Deep structural causal models for tractable counterfactual inference. *arXiv preprint arXiv:2006.06485*, 2020.

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

[34] Judea Pearl. Bayesian networks. 2011.

[35] Judea Pearl. The seven tools of causal inference, with reflections on machine learning. *Communications of the ACM*, 62(3):54–60, 2019.

[36] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*, pages 1530–1538, 2015.

[37] Andrew Slavin Ross, Michael C Hughes, and Finale Doshi-Velez. Right for the right reasons: training differentiable models by constraining their explanations. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 2662–2670, 2017.

[38] Prasanna Sattigeri, Samuel C Hoffman, Vijil Chenthamarakshan, and Kush R Varshney. Fairness gan: Generating datasets with fairness properties using a generative adversarial network. *IBM Journal of Research and Development*, 63(4/5):3–1, 2019.

[39] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328, 2017.

[40] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

[41] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[42] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[43] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2989, 2017.

[44] Sharon Zhou, Mitchell L Gordon, Ranjay Krishna, Austin Narcomey, Li Fei-Fei, and Michael S Bernstein. Hype: A benchmark for human eye perceptual evaluation of generative models. *arXiv preprint arXiv:1904.01121*, 2019.