# BEYOND TRIVIAL COUNTERFACTUAL GENERATIONS WITH DIVERSE VALUABLE EXPLANATIONS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Explainability of black-box predictive models has gained considerable attention within our research community given the importance of deploying more reliable machine-learning systems. Explanability can also be helpful for model debugging. In computer vision applications, most methods explain models by displaying the regions in the input image that they focus on for their prediction, but it is difficult to improve models based on these explanations since they do not indicate why the model fail. Counterfactual methods, on the other hand, indicate how to perturb the input to change the model prediction, providing details about the model's decision-making. Unfortunately, current counterfactual methods make ambiguous interpretations as they combine multiple biases of the model and the data in a single counterfactual interpretation of the model's decision. Moreover, these methods tend to generate trivial counterfactuals about the model's decision, as they often suggest to exaggerate or remove the presence of the attribute being classified. Trivial counterfactuals are usually not valuable, since the information they provide is often already known to the system's designer. In this work, we propose a counterfactual method that learns a perturbation in a disentangled latent space that is constrained using a diversity-enforcing loss to uncover multiple valuable explanations about the model's prediction. Further, we introduce a mechanism to prevent the model from producing trivial explanations. Experiments on CelebA demonstrate that our model improves the success rate of producing high-quality valuable explanations when compared to previous state-of-the-art methods. We will make the code publicly available on github.

## 1 INTRODUCTION

Consider a face authentication system for unlocking a device. In case of non-authentications (possible false-negative predictions), this system could provide generic advices to its user such as "face the camera" or "remove any face occlusions". However, these may not explain the reason for the possible malfunction. To provide more insights regarding its decisions, the system could instead provide information specific to the captured image (its input data). It might list the input feature that most contributed to its decision (e.g., as a region of the input image), but this feature could be "face", which is trivial and does not suggest an alternative action to its user. Further, it provides little useful information about the model. Instead, valuable and diverse explanations may be key for better understanding and diagnosing the system— including the data it was trained on— and improving its reliability. Such explanations might improve systems across a wide variety of domains including in medical imaging [45], automated driving systems [36], and quality control in manufacturing [19].

The explainability literature aims to understand the decisions made by black-box models such as the aformentionned face authentication system. Counterfactual explanation methods [10, 12, 4] can help discover the limitations of black-box models by uncovering data and model biases. The counterfactual explanation methods provide perturbed versions of the input data that emphasize features that contributed most to the black-box model's output. For example, if an authentication system is not recognizing a user wearing sunglasses then the system could generate an alternative image of the user's face without sunglasses that would be correctly recognized. This is different from other types of explainability methods such as feature importance methods [38, 39, 4] and boundary approximation methods [35, 29]. The former highlight salient regions of the input but do not indicate how the black-box could achieve a different prediction. The second family of methods produce

explanations that are limited to linear approximations of the black-box model. Unfortunately, these linear approximations are often inaccurate. In contrast, counterfactual methods suggest changes in the input that would lead to a change in the corresponding output, providing information not only about where the change should be but also *what* the change should be.

Counterfactual explanations should be *actionable* and *proximal* [37, 30]. For example, an actionable explanation would suggest feasible changes like removing sunglasses instead of unrealistic ones like adding more eyes to the user's face. Most actionable explanations are *proximal* to the input, since it is easier to interpret a perturbed input that only changes a small number of attributes. The last desideratum is for explanations to be *diverse* [37, 30] thereby providing a set of actionable changes that each shed light on the model's prediction.

Current counterfactual generation methods like xGEM [23] generate a single explanation that is far from the input. Thus, they fail to be proximal, diverse, and actionable. Progressive Exaggeration (PE) [41] provides higher-quality explanations, making them more proximal than xGEM but still fails to provide a diverse set of non-trivial explanations. Like previous methods in the literature, both methods also tend to provide obvious (or not *valuable*) explanations about why the model is making a certain decision. Moreover, their image generator must be trained on the same data as the black-box model in order to detect biases thereby limiting their applicability.

We propose Diverse Valuable Explanations (DiVE), an explainability method that can interpret a black-box model by identifying sets of valuable attributes that have the most effect on the model's output. DiVE produces multiple counterfactual explanations which are actionable, diverse, valuable and more proximal than the previous literature. Our method first learns a generative model of the data using a $\beta$-TCVAE [5] to obtain a disentangled latent representation. Unlike PE, training DiVE does not require the original data used to train the black-box model nor the black-box itself. DiVE then learns a latent perturbation using constraints to ensure diversity and proximality. In addition, DiVE leverages the Fisher information matrix of its latent space to focus its search on the less influential factors of variation of the black-box. This mechanism enables the discovery of spurious correlations learned by the black-box.

We first test our method using two existing benchmarks. Using the first, we compare the quality of the explanations with xGEM [23] and Progressive exaggeration [41]. With the second, we evaluate our model's ability to detect biases in the black-box model and the data. We also introduce a new benchmark where we evaluate the model's ability to generate valuable explanations, *i.e.*, which are simultaneously proximal and actionable. Our method achieves state-of-the-art results in the first two setups, and establishes the first baseline for the setup of valuable explanations.

We summarize the contributions of this work as follows:

1. We propose DiVE, an explainability method that can interpret a black-box model by identifying the attributes that have the most effect on its output.
2. DiVE achieves state of the art in terms of the quality of the explanations, detecting biases on the datasets, and producing multiple explanations for an image.
3. We propose a new benchmark to evaluate how valuable the explanations are, and we set a first strong baseline with DiVE.

## 2 Related Work

Explainable artificial intelligence (XAI) is a suite of techniques developed to make either the construction or interpretation of model decisions more accessible and meaningful. Broadly speaking, there are two branches of work in XAI, ad-hoc and post-hoc.

Ad-hoc methods focus on making models interpretable, by imbuing model components or parameters with interpretations that are rooted in the data themselves [33, 31, 22]. Unfortunately, most successful machine learning methods, including deep learning ones, are uninterpretable [6, 26, 15, 21].

Post-hoc methods aim to explain the decisions of non interpretable model. Gradient based feature attribution methods [46, 47, 40, 42, 38, 1, 39] and reference based feature attribution methods [39, 10, 5, 7, 4] identify the input features responsible for the greatest change in the model outcome. However they do not explain how to modify these features to change the model outcome. Closest

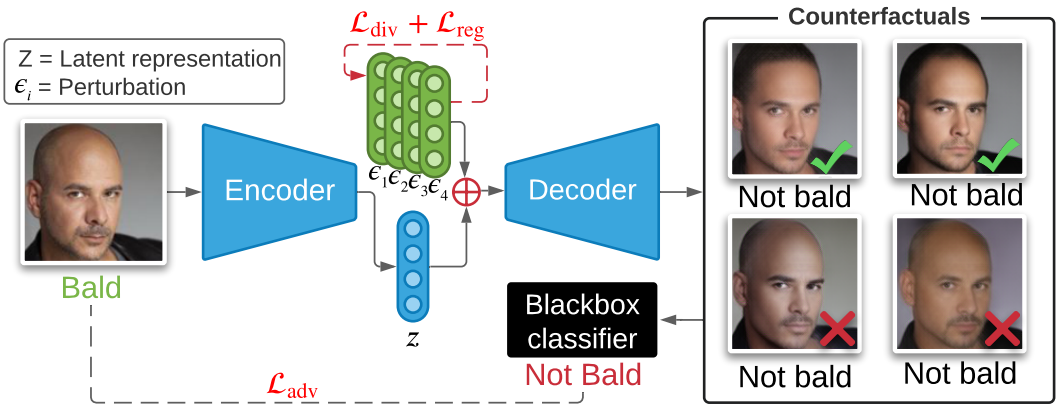



Figure 1: DiVE encodes the input image (left) to explain into a latent representation $z$. Then $z$ is perturbed by $\epsilon$ and decoded as counterfactual examples. During training, $\mathcal{L}_{\text{adv}}$ finds the set of $\epsilon$ that change the blackbox classifier outcome while $\mathcal{L}_{\text{div}}$ and $\mathcal{L}_{\text{reg}}$ enforce that the samples are diverse while staying similar to the input image. This example was taken from the experiment presented in Section 4.3. Specifically, all counterfactuals are classified as "Not bald" by the black-box classifier, as denoted by the black text. However, humans as well as our oracle classify the bottom ones as "Bald". We thus call these counterfactufals *non-trivial*, denoted here by the red crossmark. These counterfactuals identify a weakness in the blackbox model.

to our work are the counterfactual explanation methods [23, 9, 14, 30, 41] which generate perturbed versions of observed data that result in a corresponding change in the model prediction. While these methods are able to explain a model outcome, they fail to provide a diverse set of non-trivial explanations. In this work we propose DiVE, a counterfactual explanation method that generates actionable, proximal, diverse and non-trivial explanations. Appendix A provides a more exhaustive review of the related work.

## 3 Proposed Method

We propose DiVE, an explainability method that can interpret a black-box model by identifying the latent attributes that have the most effect on its output. Summarized in Figure 1, DiVE uses an encoder, a decoder, and fixed weight black-box model. The black-box model could be any function for which we have access to its gradients. In this work, we focus on a binary image classifier in order to produce visual explanations. DiVE consists of two main steps. First, the encoder and the decoder are trained in an unsupervised manner to approximate the data distribution on which the black-box model was trained. Unlike PE [41], our encoder-decoder model does not need to train on the same dataset that the black-box model was trained on. Second, we optimize a set of vectors $\boldsymbol{\epsilon}_i$ to perturb the latent representation $\mathbf{z}$ generated by the trained encoder. The details of the optimization procedure are provided in Algorithm 1 in the Appendix. We use the following 3 main losses for this optimization: an adversarial loss $\mathcal{L}_{\text{adv}}$ that attempts to fool the black-box model, an adversarial regularization loss $\mathcal{L}_{\text{reg}}$ that constrains the explanations with respect to the number of changing attributes, and a diversity loss $\mathcal{L}_{\text{div}}$ that enforces the model to produce multiple explanations with only one confounding factor for each of them. Next we explain the methodology in more detail.

### 3.1 Obtaining meaningful representations.

Given a data sample $\mathbf{x} \in \mathcal{X}$, its corresponding target $y \in \{0, 1\}$, and a potentially biased black-box model $f(\mathbf{x})$ that approximates $p(y|\mathbf{x})$, our method finds perturbed version of the same input $\tilde{\mathbf{x}}$ that produces a desired probabilistic outcome $\tilde{y} \in [0, 1]$, so that $f(\tilde{\mathbf{x}}) = \tilde{y}$. In order to produce semantically meaningful counterfactual explanations, perturbations are performed on a latent representation

$\mathbf{z} \in \mathcal{Z} \subseteq \mathbb{R}^d$ of the input $\mathbf{x}$. Ideally, each dimension in $\mathcal{Z}$ represents a different semantic concept of the data, *i.e.*, the different dimensions are *disentangled*.

For training the encoder-decoder architecture we use $\beta$-TCVAE [5] since it has been shown to obtain competitive disentanglement performance [28]. It follows the same encoder-decoder structure as the VAE [25], *i.e.*, the input data is first encoded by a neural network $q_\phi(z|\mathbf{x})$ parameterized by $\phi$. Then, the input data is recovered by a decoder neural network $p_\theta(\mathbf{x}|z)$, parameterized by $\theta$. Using a prior $p(z)$ and a uniform distribution over the indexes of the dataset $p(i)$, the original VAE loss is:

$$\mathcal{L}_{VAE} = \mathbb{E}_{p(i)}\mathbb{E}_{q(z|\mathbf{x}_i)}[\log p_\theta(\mathbf{x}_i|z)] - \mathbb{E}_{p(i)} D_{\mathrm{KL}}\left(q_\phi(z|\mathbf{x}_i)||p(z)\right), \tag{1}$$

where the first term is the reconstruction loss and the second is the average divergence from the prior. The core difference of $\beta$-TCVAE is the decomposition of this average divergence:

$$\begin{aligned}\mathbb{E}_{p(i)} D_{\mathrm{KL}}\big(q_\phi(z|\mathbf{x}_i)||p(z)\big) \rightarrow\ & D_{\mathrm{KL}}\left(q_\phi(z,\mathbf{x}_i)||q_\phi(z)p_\theta(\mathbf{x}_i)\right) + \textstyle\sum_j D_{\mathrm{KL}}\left(q_\phi(z_j)||p(z_j)\right) \\ & + \beta \cdot D_{\mathrm{KL}}\big(q_\phi(z)||\textstyle\prod_j q_\phi(z_j)\big),\end{aligned} \tag{2}$$

where the arrow represents a modification of the left terms and equality is obtained when $\beta = 1$. The third term on the right hand side is called total correlation and measures the shared information between all empirical marginals $q_\phi(z_j) = \mathbb{E}_{p(i)} q_\phi(z_j|\mathbf{x}_i)$. By using $\beta > 1$, this part is amplified and encourages further decorrelations between the latent variables and leads to better disentanglement.

In addition to $\beta$-TCVAE, we use the perceptual reconstruction loss from Hou et al. [17]. This replaces the pixel-wise reconstruction loss in Equation 1 by a perceptual reconstruction loss, using the hidden representation of a pre-trained neural network $R$. Specifically, we learn a decoder $D_\theta$ generating an image *i.e.*, $\tilde{\mathbf{x}} = D_\theta(\mathbf{z})$, and this image is re-encoded in a hidden representation: $\boldsymbol{h} = R(\tilde{\mathbf{x}})$, and compared to the original image in the same space using a normal distribution. The reconstruction loss of Equation 1 now becomes:

$$\mathbb{E}_{p(i)}\mathbb{E}_{q(\mathbf{z}|\mathbf{x}_i)}[\log \mathcal{N}(R(\mathbf{x}_i); R(D_\theta(\mathbf{z})), \boldsymbol{I})], \tag{3}$$

Once trained, the weights of the encoder and the decoder are kept fixed for the rest of the steps of our algorithm.

### 3.2 Interpreting the black-box model

In order to find weaknesses in the black-box model, the explainer searches for a collection of $n$ latent perturbation $\{\boldsymbol{\epsilon}_i\}_{i=1}^n$ such that the decoded output $\tilde{\mathbf{x}}_i = D_\theta(\mathbf{z} + \boldsymbol{\epsilon}_i)$ yields a specific response from the black-box model, *i.e.*, $f(\tilde{\mathbf{x}}) = \tilde{y}$ for any chosen $\tilde{y} \in [0, 1]$. We optimize $\boldsymbol{\epsilon}_i$'s by minimizing:

$$\mathcal{L}_{\mathrm{DiVE}}(\mathbf{x}, \tilde{y}, \{\boldsymbol{\epsilon}_i\}_{i=1}^n) = \textstyle\sum_i \mathcal{L}_{\mathrm{adv}}(\mathbf{x}, \tilde{y}, \boldsymbol{\epsilon_i}) + \lambda \cdot \sum_i \mathcal{L}_{\mathrm{reg}}(\mathbf{x}, \boldsymbol{\epsilon_i}) + \alpha \cdot \mathcal{L}_{\mathrm{div}}(\{\boldsymbol{\epsilon}_i\}_{i=1}^n), \tag{4}$$

where $\lambda$, and $\alpha$ determine the relative importance of the losses. We now describe these terms.

**Adversarial loss.** The goal of this loss function is to identify a change of latent attributes that will cause the black-box classifier $f$ to change it's prediction. For example, in face recognition, if the classifier detects that there is a smile present whenever the hair is brown, then this loss function is likely to change the hair color attribute. This is achieved by sampling from the decoder $\tilde{\mathbf{x}} = D_\theta(\mathbf{z} + \boldsymbol{\epsilon})$, and optimizing the binary cross-entropy between the target $\tilde{y}$ and the prediction $f(\tilde{\mathbf{x}})$:

$$\mathcal{L}_{\mathrm{adv}}(\mathbf{x}, \tilde{y}, \boldsymbol{\epsilon}) = \tilde{y} \cdot \log(f(\tilde{\mathbf{x}})) + (1 - \tilde{y}) \cdot \log(1 - f(\tilde{\mathbf{x}})). \tag{5}$$

Note that since the function $f$ and decoder $p_\theta$ have fixed parameters, there is no min-max game to be solved. Hence our algorithm does not suffer from traditional instabilities of adversarial algorithms.

**Regularization loss.** The goal of this loss function is to constrain the reconstruction produced by the decoder to be similar in appearance and attributes as the input . It consists of the following two terms,

$$\mathcal{L}_{\mathrm{reg}}(\mathbf{x}, \boldsymbol{\epsilon}) = ||\mathbf{x} - \tilde{\mathbf{x}}||_1 + \gamma \cdot ||\boldsymbol{\epsilon}||_1, \tag{6}$$

where $\gamma$ is a scalar weighting the relative importance of the two terms. The first term ensures that the explanations can be related to the input by constraining the input and the output to be similar. The second term aims to identify a sparse perturbation to the latent space $\mathcal{Z}$ that confounds the black-box model. This sparsity constrains the explainer to identify the least amount of attributes that affect the classifier's decision in order to produce proximal and actionable explanations.

**Diversity loss.** This loss prevents the multiple explanations of the model from being identical. For instance, if gender and hair color are spuriously correlated with smile, the model should provide images either with different gender or different hair color. To achieve this, we jointly optimize for a collection of $n$ perturbations $\{\boldsymbol{\epsilon}_i\}_{i=1}^n$ and minimize their pairwise similarity:

$$\mathcal{L}_{\text{div}}(\{\boldsymbol{\epsilon}_i\}_{i=1}^n) = \sqrt{\sum_{i \neq j} \left( \frac{\boldsymbol{\epsilon}_i^T}{\|\boldsymbol{\epsilon}_i\|_2} \frac{\boldsymbol{\epsilon}_j}{\|\boldsymbol{\epsilon}_j\|_2} \right)^2}. \tag{7}$$

**Beyond trivial counterfactual explanations.** Minimizing the losses above may produce trivial counterfactual explanations. For instance, in order to explain why a classifier incorrectly classified an image as containing a "smiling" face, the explainer could just exaggerate smile on that face, without considering other valuable biases in the black-box model such as hair color. While the diversity loss encourages the orthogonality of the explanations, there might still be several latent variables required to represent all variations of smile. To address this, we partition $\mathcal{Z}$ into subsets of latent factors that interact with each other when changing the predictions of the black-box. Such interaction can be estimated using the average Fisher information matrix:

$$\boldsymbol{F} = \mathbb{E}_{p(i)} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \mathbb{E}_{p(y|\mathbf{z})} \nabla_{\mathbf{z}} \ln p(y|\mathbf{z}) \; \nabla_{\mathbf{z}} \ln p(y|\mathbf{z})^T, \tag{8}$$

where $p(y=1|\mathbf{z}) = f(D_\theta(\mathbf{z}))$, and $p(y=0|\mathbf{z}) = 1 - f(D_\theta(\mathbf{z}))$.

Next, using $\boldsymbol{F}$ as an affinity measure, we use spectral clustering [43] to obtain a partition of $\mathcal{Z}$. This partition is represented as a collection of mask $\{\boldsymbol{m}_i\}_{i=1}^n$, where $\boldsymbol{m}_i \in \{0,1\}^d$ represents which dimensions of $\mathcal{Z}$ are part of cluster $i$. Finally, these masks are used in Equation 4 to bound each $\boldsymbol{\epsilon}_i$ to its subspace *i.e.*, $\boldsymbol{\epsilon}_i' = \boldsymbol{\epsilon}_i \circ \boldsymbol{m}_i$, where $\circ$ represents element wise multiplication. Since these mask are orthogonal, this effectively replaces $\mathcal{L}_{\text{div}}$. In Section 4, we highlight the benefits of this clustering approach by comparing to other baselines.

## 4 Experimental Results

In this section, we evaluate the described methods on 3 different aspects: (1) the quality of the generated explanations in Section 4.1; (2) the ability to discover biases within the black-box model and the data in Section 4.2; and (3) the ability to identify diverse valuable explanations for image misclassifications made by the black-box model in Section 4.3.

**Experimental Setup.** As common procedure [23, 9, 41], we perform experiments on the CelebA database [27]. CelebA is a large-scale dataset containing more than 200K celebrity facial images. Each image is annotated with 40 binary attributes such as "Smiling", "Male", and "Eyeglasses". These attributes allow us to evaluate counterfactual explanations by determining whether they could highlight spurious correlations between multiple attributes such as "lipstick" and "smile". In this setup, explainability methods do not have access to the labeled attributes during training. The labels can only be used during validation.

We compare four versions of our method to three existing methods. **(1) DiVE** is based on Section 3 but does not mask the gradient updates of $\epsilon$. **(2) DiVE−−** is the same as DiVE but uses the MAE reconstruction loss on the pixel space for training the autoencoder. **(3) DiVE$_{\text{Fisher}}$** extends DiVE by using the diagonal of the Fisher Information Matrix on the latent features and masks based on the partitions extracted from a sorted list of Fisher magnitudes. **(4) DiVE$_{\text{FisherSpectral}}$** is the same as DiVE$_{\text{Fisher}}$ but uses spectral clustering for obtaining the partitions as described at the end of Section 3. **(5) xGEM** as described in Joshi et al. [23]. **(6) xGEM+** is the same as xGem but uses the same auto-encoding architecture as DiVE. **(7) PE** as described by Singla et al. [41]. For our methods, we provide implementation details, architecture description, and algorithm in Appendix D.

### 4.1 Counterfactual Explanation Quality

We evaluate the quality of the counterfactual explanations using FID scores [16] as described by Singla et al. [41]. The scores are based on the target attributes "Smiling" and "Young", and are divided into 3 categories: *Present*, *Absent*, and *Overall*. *Present* considers explanations for which

Table 1: FID of DiVE compared to xGEM [23], Progressive Exaggeration (PE) [41], xGEM trained with our backbone (xGEM+), and DiVE trained without the perceptual loss (DiVE−−)

| Target Attribute | xGEM | PE | xGEM+ | DiVE−− | DiVE |
|---|---|---|---|---|---|
| | **Smiling** | | | | |
| Present | 111.0 | 46.9 | 67.2 | 54.9 | **30.6** |
| Absent | 112.9 | 56.3 | 77.8 | 62.3 | **33.6** |
| Overall | 106.3 | 35.8 | 66.9 | 55.9 | **29.4** |
| | **Young** | | | | |
| Present | 115.2 | 67.6 | 68.3 | 57.2 | **31.8** |
| Absent | 170.3 | 74.4 | 76.1 | 51.1 | **45.7** |
| Overall | 117.9 | 53.4 | 59.5 | 47.7 | **33.8** |

Table 2: **Bias detection experiment.** For the targets "Smiling" and "Non-Smiling" we generate explanations for a classifier biased on gender ($f_{\text{biased}}$) and an unbiased classifier ($f_{\text{un-biased}}$).

| black-box model | | Target label | | | | | |
|---|---|---|---|---|---|---|---|
| | | Smiling | | | Non-Smiling | | |
| | | PE | xGEM+ | DiVE | PE | xGEM+ | DiVE |
| $f_{\text{biased}}$ | Male | 0.52 | 0.06 | 0.11 | 0.18 | 0.77 | 0.84 |
| | Female | 0.48 | 0.94 | 0.89 | 0.82 | 0.24 | 0.16 |
| | Overall | 0.12 | **0.29** | 0.22 | 0.35 | 0.33 | **0.36** |
| | oracle | | 0.75 | | | 0.67 | |
| $f_{\text{un-biased}}$ | Male | 0.48 | 0.41 | 0.42 | 0.47 | 0.38 | 0.44 |
| | Female | 0.52 | 0.59 | 0.58 | 0.53 | 0.62 | 0.57 |
| | Overall | **0.07** | 0.13 | 0.10 | 0.08 | 0.15 | **0.07** |
| | oracle | | 0.04 | | | 0.00 | |

the black-box model outputs a probability greater than 0.9 for the target attribute. *Absent* refers to explanations for which the black-box model outputs a probability lower than 0.1 for the target attribute. *Overall* considers all the successful counterfactuals, which changed the original prediction of the black-box model.

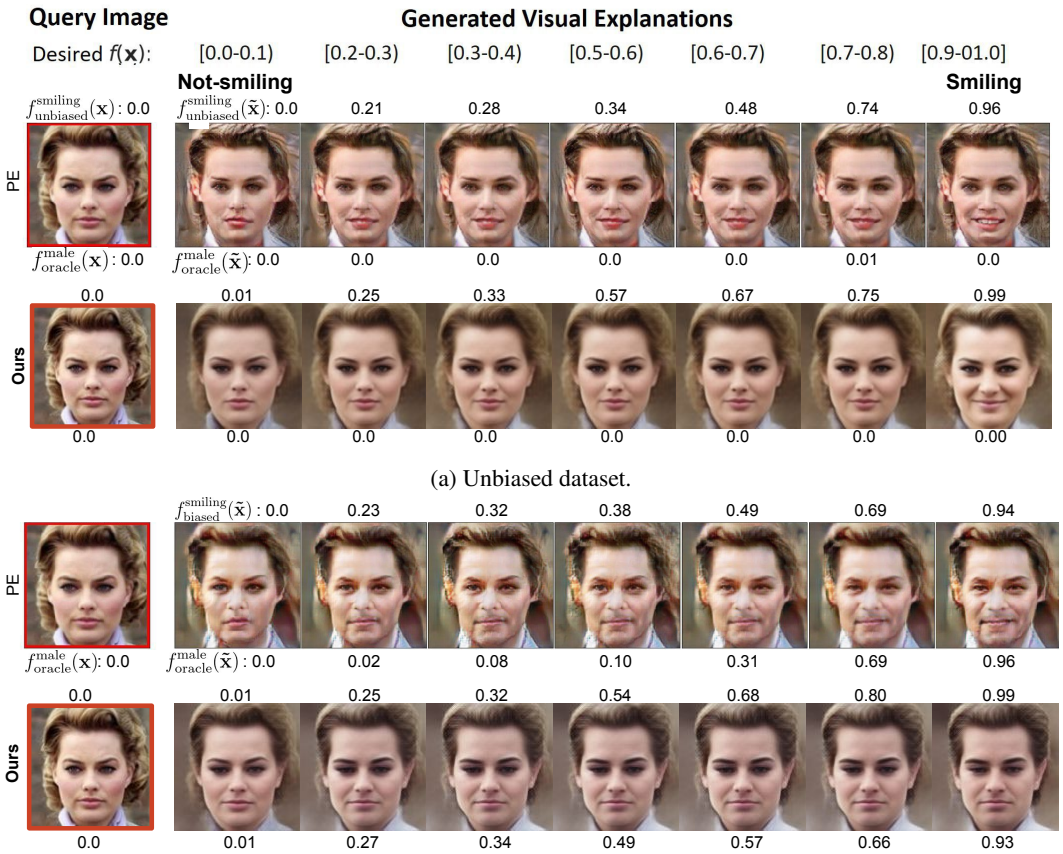
We report these scores in Table 1 for all 3 categories. DiVE produces the best quality counterfactuals, surpassing PE by 6.3 FID points for the "Smiling" target and 19.6 FID points for the "Young" target in the *Overall* category. DiVE obtains lower FID than xGEM+ which shows that the improvement not only comes from the superior architecture of our method. Further, there are two other factors that explain the improvement of DiVE's FID. First, the $\beta$-TCVAE decomposition of the KL divergence improves the disentanglement ability of the model while suffering less reconstruction degradation than the VAE. Second, the perceptual loss makes the image quality constructed by DiVE to be comparable with that of the GAN used in PE.

In Figure 2a we show qualitative results obtained by targeting different probability ranges for the output of the black-box model as described in PE. Note that PE directly optimizes the generative model to take an input variable $\delta \in \mathbb{R}$ that defines the desired output probability $\tilde{y} = f(\mathbf{x}) + \delta$. To obtain explanations at different probability targets, we train a second order spline on the trajectory of perturbations produced during the gradient descent steps of our method. As seen in Figure 2a, DiVE produces more natural-looking facial expressions than xGEM+ and PE. In Figure 2a it can be seen that, even though DiVE is not explicitly trained to produce exemplars at intermediate target probabilities, our explanations are more correlated with the target probabilities than PE. Additional results for "Smiling" and "Young" are provided in Figures 4 and 5 in the Appendix.

### 4.2 BIAS DETECTION

We evaluate DiVE's ability to detect biases in the data. We follow the same procedure as PE [41], and train two binary classifiers for the attribute "Smiling". The first one is trained on a biased version of CelebA where all the male celebrities are smiling and all the female are not smiling ($f_{biased}$). The second one is trained on the unbiased version of the data ($f_{unbiased}$). Both classifiers are evaluated on the same validation set. Also following Singla et al. [41], we train an oracle classifier $f_{\text{oracle}}$ based on VGGFace2 [3] which obtains perfect accuracy on the gender attribute. The hypothesis is that if "Smiling" and gender are confounded by the classifier, so should be the explanations. Therefore, we could identify biases when the generated examples not only change the target attribute but also the confounded one.

In Table 2, we follow the procedure in [41] and report the ratio of counterfactual explanations for "Smiling" that change the "Gender" attribute of the $f_{\text{biased}}$ and $f_{\text{unbiased}}$ classifiers. To generate the counterfactuals, DiVE produces perturbations until it changes the original prediction of the classifier from, say, "Smiling" to "Non-Smiling". We also see that DiVE is more successful than PE at detecting biases although the generative model of DiVE was not trained with the biased data. While, in some cases, xGEM+ has a higher success rate at detecting biases, it produces lower-quality images that are far from the input. In fact, Table 3 in the Appendix shows that DiVE is more successful at preserving the identity of the faces than PE and xGEM. These results suggest that the combination




(a) Unbiased dataset.

(b) Biased dataset.

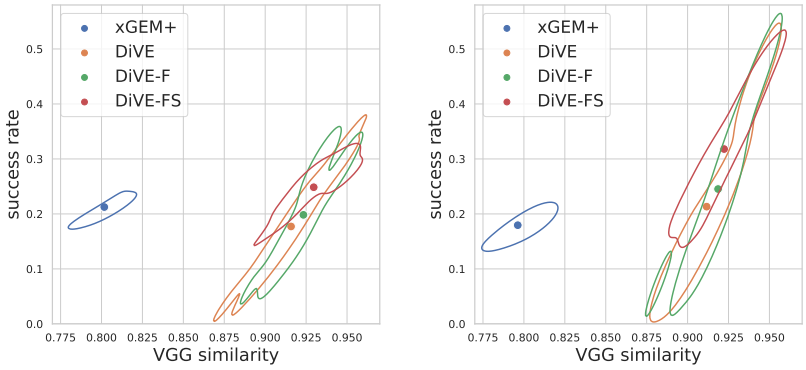
Figure 2: **Bias Detection experiment.** Each column presents an explanation for a target "Smiling" probability interval. Rows contain explanations produced by PE [41] and our DiVE. (a) of a gender-unbiased classifier, and (b) corresponds to explanations of a gender-biased "Smile" classifier. The classifier output probability is displayed on top of the images while the oracle prediction for gender is displayed at the bottom.

of a disentangled latent features and the regularization of the latent features help DiVE to produce the minimal perturbations of the input that produce a successful counterfactual.

In Figure 2b, we provide samples generated by our method with the two classifiers and compare them to PE, and extend it with xGEM+ in Figure 6 in the Appendix. As it can be seen, the gender changes with the "Smiling" attribute with $f_{\text{biased}}$ while for $f_{\text{unbiased}}$ it stays the same. In addition, we also observed that for $f_{\text{biased}}$ the correlation between "Smile" and "Gender" is higher than for PE. It can also be observed that xGEM+ fails to retain the identity of the person in $\mathbf{x}$ when compared to PE and our method.

### 4.3 Beyond trivial explanations

Previous works on counterfactual generations tend to produce trivial input perturbations to change the output of the black-box model. That is, they often tend to directly increase/decrease the presence of the attribute that the classifier is predicting. For instance, in Figure 2a all the explainers put a smile on the input face in order to increase the probability for "smile". While that is correct, this explanation does not provide much insight about the decisions of the black-box model. Instead, in this work we emphasize producing valuable or non-trivial explanations, that are different from the main attribute that the black-box model has been trained to identify. These kind of explanations provide more insight about the factors that affect the classifier and thus provide cues on how to improve the model or how to fix incorrect predictions.




(a) all explanations

(b) successful counterfactuals only

Figure 3: **Beyond trivial explanations experiment.** Success rate (y-axis) plotted against VGG similarity (x-axis) for all methods. For both metrics, higher is better. The dot denotes the mean of the performances and the curves are compute with KDE. All DiVE methods outperform xGEM+ on both metrics simultaneously when conditioning on successful counterfactuals.

To evaluate this, we propose a new benchmark that measures a method's ability to generate valuable explanations. We define a valuable explanation as one that is 1) misclassified by the black-box model (according to a human) and 2) has not diverged too much from the original sample. A misclassification provides insights into the weaknesses of the model. However, the counterfactual is even more insightful when it stays close to the original image as it singles-out spurious correlations learned by the black-box model. Because it is costly to provide human evaluation of an automatic benchmark, we approximate both the proximity and the real class with the VGGFace2-based oracle. For 1) we deem that an explanation is successful if the black-box and the oracle make different predictions about the counterfactual. E.g., the top counterfactuals in Figure 1 are not deemed successful explanations because both the black-box classifier and the oracle agree on its class, however the two bottoms ones are successful because only the oracle made the correct prediction. These explanations where generated by DiVE-FS. As for 2) we measure the proximity with the cosine distance between the sample and the counterfactual in the feature space of the oracle.

We test all methods introduced in Section 4 on a subset of the CelebA validation set described in Appendix E. We report the results of the hyperparameter search (see Appendix E) in Figure 3. We show results for all explanations in Figure 3a and only when the generated images are counterfactuals in Figure 3b. The dots denote the mean performances and the curves are computed with Kernel Density Estimation (KDE). On average, DiVE improves the similarity metric over xGEM+ highlighting the importance of disentangled representations for identity preservation. Moreover, using information from the diagonal of the Fisher Information Matrix as described in further improves performance. Finally, the proposed spectral clustering of the full Fisher Matrix attains the best performance. Also, discarding non-conterfactuals improves the success rate.

## 5 Conclusion

In this paper, we propose DiVE, a model for generating diverse and valuable explanations of a model's decision. During training, the model optimizes an auto-encoder that learns a disentangled representation of the data. At test time, the model optimizes a perturbation vector of the latent representation in order to generate explanations. This optimization involves an adversarial loss, a diversity-enforcing loss and an adversarial regularization loss. Further, our model uses the Fisher Information Matrix to mask the most influential dimensions of the latent features to enforce the model to produce more valuable explanations that are beyond trivial. Our experiments show that previous methods are limited to single explanations whereas ours can produce multiple, diverse explanations. The results also show that our model achieves state-of-the-art results in terms of proximity and actionability on the CelebA dataset.

## REFERENCES


[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, 2018.

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[3] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *International Conference on Automatic Face and Gesture Recognition*, 2018.

[4] Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and David Duvenaud. Explaining image classifiers by counterfactual generation. In *International Conference on Learning Representations*, 2019.

[5] Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, 2018.

[6] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3): 273–297, 1995.

[7] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. *arXiv preprint arXiv:1705.07857*, 2017.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, 2009.

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

[10] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *International Conference on Computer Vision*, 2017.

[11] Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical annealing schedule: A simple approach to mitigating kl vanishing. *arXiv preprint arXiv:1903.10145*, 2019.

[12] Yarin Gal, Jiri Hron, and Alex Kendall. Concrete dropout. In *Advances in neural information processing systems*, 2017.

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

[14] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. *arXiv preprint arXiv:1904.07451*, 2019.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, 2016.

[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. In *Advances in Neural Information Processing Systems*, 2017.

[17] Xianxu Hou, Linlin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. In *Winter Conference on Applications of Computer Vision*, 2017.

[18] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Computer Vision and Pattern recognition*, 2017.

[19] Farhad Imani, Ruimin Chen, Evan Diewald, Edward Reutzel, and Hui Yang. Deep learning of variant geometry in layerwise imaging profiles for additive manufacturing quality control. *Journal of Manufacturing Science and Engineering*, 141(11), 2019.

[20] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, 2015.

[21] Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Computer Vision and Pattern Recognition Workshops*, 2017.

[22] Finn V Jensen et al. *An introduction to Bayesian networks*, volume 210. UCL press London, 1996.

[23] Shalmali Joshi, Oluwasanmi Koyejo, Been Kim, and Joydeep Ghosh. xgems: Generating examplars to explain black-box models. *arXiv preprint arXiv:1806.08867*, 2018.

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

[25] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[26] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989.

[27] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *International Conference on Computer Vision*, 2015.

[28] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, 2019.

[29] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*, 2017.

[30] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *Conference on Fairness, Accountability, and Transparency*, 2020.

[31] John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384, 1972.

[32] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. *arXiv preprint arXiv:1709.07871*, 2017.

[33] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[34] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *International Conference on Learning Representations*, 2018.

[35] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?" explaining the predictions of any classifier. In *International Conference on Knowledge Discovery and Data Mining*, 2016.

[36] German Ros, Sebastian Ramos, Manuel Granados, Amir Bakhtiary, David Vazquez, and Antonio M Lopez. Vision-based offline-online perception paradigm for autonomous driving. In *Winter Conference on Applications of Computer Vision*, 2015.

[37] Chris Russell. Efficient search for diverse coherent explanations. In *Conference on Fairness, Accountability, and Transparency*, 2019.

[38] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *International Conference on Computer Vision*, 2017.

[39] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. *arXiv preprint arXiv:1704.02685*, 2017.

[40] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[41] Sumedha Singla, Brian Pollack, Junxiang Chen, and Kayhan Batmanghelich. Explanation by progressive exaggeration. In *International Conference on Learning Representations*, 2020.

[42] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[43] X Yu Stella and Jianbo Shi. Multiclass spectral clustering. In *null*, pp. 313. IEEE, 2003.

[44] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[45] David Vázquez, Jorge Bernal, F Javier Sánchez, Gloria Fernández-Esparrach, Antonio M López, Adriana Romero, Michal Drozdzal, and Aaron Courville. A benchmark for endoluminal scene segmentation of colonoscopy images. *Journal of healthcare engineering*, 2017.

[46] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014.

[47] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Computer Vision and Pattern Recognition*, 2016.

# APPENDIX

## A EXTENDED RELATED WORK

Counterfactual explanation lies inside a more broadly-connected body of work for explaining classifier decisions. Different lines of work share this goal but vary in the assumptions they make about what elements of the model and data to emphasize as way of explanation.

**Gradient based feature attribution.** These methods identify input features responsible for the greatest change in the loss function, as measured by the magnitude of the gradient with respect to the inputs. Early work in this area focused on how methodological improvements for object detection in images could be re-purposed for feature attribution [46, 47], followed by work summarized gradient information in different ways [40, 42, 38]. Closer inspection identified pitfalls of gradient-based methods, including induced bias due to gradient saturation or network structure [1], as well as discontinuity due to activation functions [39]. These methods typically produce dense feature maps, which are difficult to interpret. In our work we address this by constraining the generative process of our counterfactual explanations.

**Reference based feature attribution.** These methods focus instead on measuring the differences observed by substituting observed input values with ones drawn from some reference distribution, and accumulating the effects of these changes as they are back-propagated to the input features. Shrikumar et al. [39] use a modified back-propagation approach to gracefully handle zero gradients and negative contributions, but leave the reference to be specified by the user. Fong & Vedaldi [10] propose three different heuristics for reference values: replacement with a constant, addition of noise, and blurring. Other recent efforts have focused on more complex proposals of the reference distribution. Chen et al. [5] construct a probabilistic model that acts as a lower bound on the mutual information between inputs and the predicted class, and choose zero values for regions deemed uninformative. Building on desiderata proposed by Dabkowski & Gal [7], Chang et al. [4] use a generative model to marginalize over latent values of relevant regions, drawing plausible values for each. These methods typically either do not identify changes that would *alter* a classifier decision, or they do not consider the plausibility of those changes.

**Counterfactual explanations.** Rather than identify a set of features, counterfactual explanation methods instead generate perturbed versions of observed data that result in a corresponding change in model prediction. These methods use the original black-box model as well as a generative model of the data to find trajectories of variation that elucidate model behaviour for a given test instance. Joshi et al. [23] propose a gradient guided search in latent space (via a learned encoder model), where they progressively take gradient steps with respect to a regularized loss that combines a term for plausibility of the generated data, and the loss of the black box model. Denton et al. [9] use a Generative Adversarial Network (GAN) [13] for detecting bias present in multi-label datasets. They modify the generator to obtain latent codes for different data points and learn a linear decision boundary in the latent space for each class attribute. By sampling generated data points along the vector orthogonal to the decision boundary, they observe how crossing the boundary for one attribute causes undesired changes in others. Goyal et al. [14] eschew a generative model by instead solving a surrogate editing problem. Given an original image (with some predicted class), and an image with a desired class prediction value, they produce a counterfactual explanation through a series of edits to the original image by value substitutions in the learned representations of both images. Mothilal et al. [30] focus on tabular data, and generate sets of counterfactual explanations through iterative gradient based improvement.

## B ADDITIONAL QUALITATIVE RESULTS

Figure 4 and 5 present counterfactual explanations for additional persons and attributes. The results show that DiVE achieves higher quality reconstructions compared to other methods. Further, the reconstructions made by DiVE are more correlated with the desired target for the black-box model output $f(x)$. Figure 6 extends the bias detection qualitative results with counterfactuals generated from xGEM+. Finall, Figure 7 shows successful counterfactuals for different instantiations of DiVE.

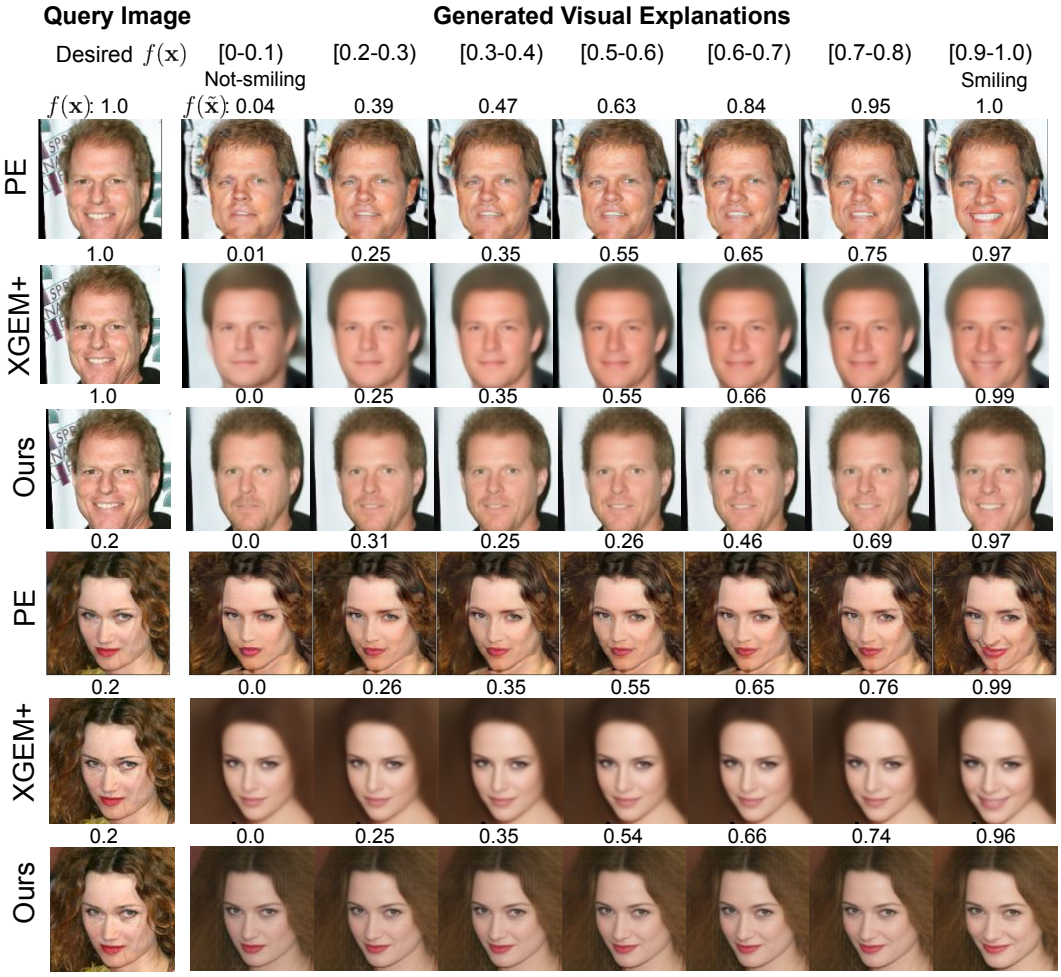



Figure 4: Qualitative results of DiVE, Progressive Exaggeration (PE) [41], and xGEM [23] for the "Smiling" attribute. Each column shows the explanations generated for a target probability output of the black-box model. The numbers on top of each row show the actual output of the black-box model.

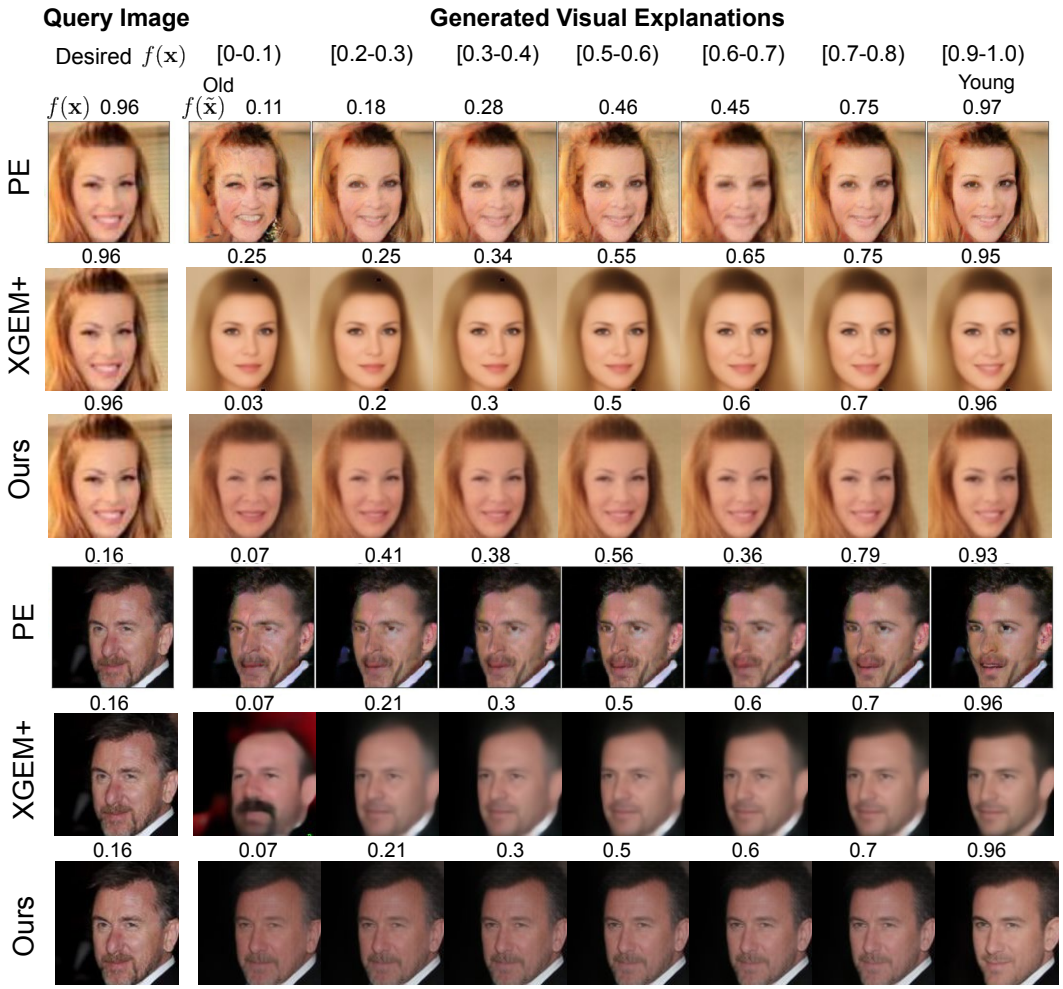



Figure 5: **Qualitative results** of DiVE, Progressive Exaggeration (PE) [41], and xGEM+ for the "Young" attribute. Each column shows the explanations generated for a target probability output of the black-box model. The numbers on top of each row show the actual output of the black-box model.

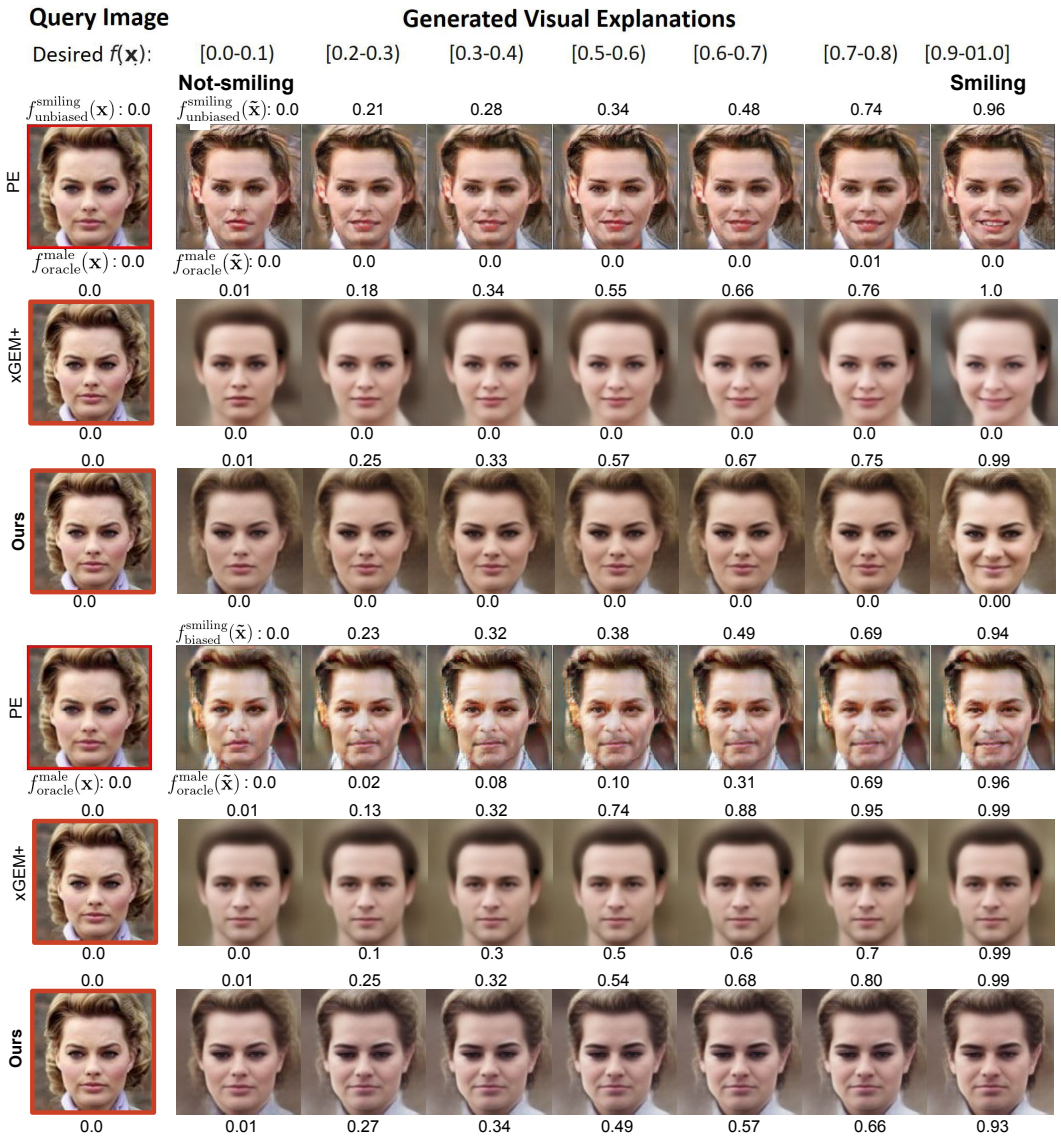



Figure 6: **Extended bias detection experiment.** Each column presents an explanation for a target "Smiling" probability interval. Rows contain explanations produced by PE [41], xGEM+ and our DiVE. (a) of a gender-unbiased classifier, and (b) corresponds to explanations of a gender-biased "Smile" classifier. The classifier output probability is displayed on top of the images while the oracle prediction for gender is displayed at the bottom.

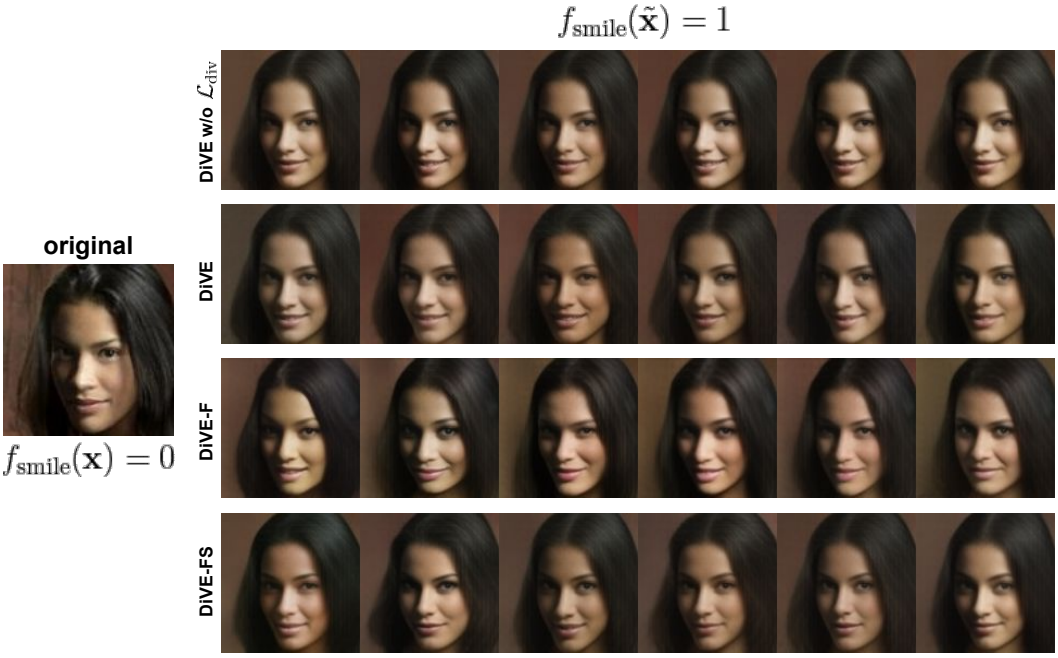



Figure 7: Successful counterfactual generations for different instantions of DiVE. Here, the original image was misclassified as non-smiling. All methodologies were able to correctly add a smile to the woman.

## C IDENTITY PRESERVATION

As argued, valuable explanations should remain proximal to the original image. Accordingly, performed the identity preservation experiment found in [41] to benchmark the methodologies against each other. Specifically, use the VGGFace2-based [3] oracle to extract latent codes for the original images as well as for the explanations and report **latent space closeness** as the fraction of time the explanations' latent codes are the closest to their respective original image latent codes' compared to the explanations on different original images. Further, we report **face verification** accuracy which consist of the fraction of time the cosine distance between the aforementioned latent codes is below 0.5.

Table 3 presents both metrics for DiVE and its baselines on the "Smilling" and "Young" classification tasks. We find that DiVE outperforms all other methods on the "Young" classification task and almost all on the "Smiling" task.

| | **CelebA:Smiling** | | | | **CelebA:Young** | | | |
|---|---|---|---|---|---|---|---|---|
| | xGEM | PE | xGEM+ | DiVE (ours) | xGEM | PE | xGEM+ | DiVE (ours) |
| **Latent Space Closeness** | 88.2 | 88.0 | **99.8** | 98.7 | 89.5 | 81.6 | 97.5 | **99.1** |
| **Face Verification Accuracy** | 0.0 | 85.3 | 91.2 | **97.3** | 0.0 | 72.2 | 97.4 | **98.2** |

Table 3: Identity preserving performance on two prediction tasks.

## D IMPLEMENTATION DETAILS

In this Section, we provide provide the details to ensure the that our method is reproducible.

**Architecture details.** DiVE's architecture is a variation BigGAN [2] as shown in Table 4. We chose this architecture because it achieved impressive FID results on the ImageNet [8]. The decoder (Table 4b) is a simplified version of the $128 \times 128$ BigGAN's residual generator, without non-local blocks nor feature concatenation. We use InstanceNorm [44] instead of BatchNorm [20] to obtain consistent outputs at inference time without the need of an additional mechanism such as recomputing statistics [2]. All the InstanceNorm operations of the decoder are conditioned on the input code $\mathbf{z}$ in the same way as FILM layers [32]. The encoder (Table 4a) follows the same structure as the BigGAN $128 \times 128$ discriminator with the same simplifications done to our generator. We use the Swish non-linearity [34] in all layers except for the output of the decoder, which uses a Tanh activation. For all experiments we use a latent feature space of 128 dimensions. To project the 2nd features produced by the encoder to a flat vector $(\mu, \log(\sigma^2))$, and to project the sampled codes $\mathbf{z}$ to a 2d space for the decoder, we use 3-layer MLPs. For the face attribute classifiers, we use the same DenseNet [18] architecture as described in Progressive Exaggeration [41].

**Optimization details.** All the models are optimized with Adam [24] with a batch size of 256. During the training step, the auto-encoders are optimized for 400 epochs with a learning rate of $4 \cdot 10^{-4}$. The classifiers are optimized for 100 epochs with a learning rate of $10^{-4}$. To prevent the auto-encoders from suffering KL vanishing, we adopt the cyclical annealing schedule proposed by Fu et al. [11] on the third term of Equation 2.

**Counterfactual inference details.** At inference time, the perturbations are optimized with Adam until the black-box output for the generated explanation $f(\tilde{\mathbf{x}})$ only differs from the target output $\tilde{y}$ by a margin $\delta$ or when a maximum number of iterations $\tau$ is reached. We set $\tau = 20$ for all the experiments since more than 90% of the counterfactuals are found after that many iterations. The different $\boldsymbol{\epsilon_i}$ are initialized by sampling from a normal distribution $\mathcal{N} \sim (0, 0.01)$. For the DiVE$_{Fisher}$ baseline, to identify the most valuable explanations, we sort $\boldsymbol{\epsilon}$ by the magnitude of $\mathbf{f} = \text{diag}(\boldsymbol{F})$. Then, we divide the dimensions of the sorted $\boldsymbol{\epsilon}$ into $N$ contiguous partitions of size $k = \frac{D}{N}$, where $D$ is the dimensionality of $\mathcal{Z}$. Formally, let $\boldsymbol{\epsilon}^{(\mathbf{f})}$ be $\boldsymbol{\epsilon}$ sorted by $\mathbf{f}$, then $\boldsymbol{\epsilon}^{(\mathbf{f})}$ is constrained as follows,

$$\epsilon_{i,j}^{(\mathbf{f})} = \begin{cases} 0, & \text{if } j \in [(i-1) \cdot k, i \cdot k] \\ \epsilon_{i,j}^{(\mathbf{f})}, & \text{otherwise} \end{cases}, \tag{9}$$

where $i \in 1..N$ indexes each of the multiple $\boldsymbol{\epsilon}$, and $j \in 1..D$ indexes the dimensions of $\boldsymbol{\epsilon}$. As a result we obtain partitions with different order of complexity. Masking the first partition results in explanations that are most implicit within the model and the data. On the other hand, masking the last partition results in explanations that are more explicit.

To compare with Singla et al. [41] in Figures 2-6 we produced counterfactuals at arbitrary target values $\tilde{y}$ of the output of the black-box classifier. To obtain explanations at different probability targets, we train a second order spline on the trajectory of perturbations produced during the gradient descent steps of our method. Thus, given the set of perturbations $\{\boldsymbol{\epsilon}_t\}, \ \forall t \in 1..\tau$, obtained during $\tau$ gradient steps, and the corresponding black-box outputs $\{f(y|\boldsymbol{\epsilon}_t)\}$, the spline obtains the $\boldsymbol{\epsilon}_{\tilde{y}}$ for a target output $\tilde{y}$ by quadratic interpolation.

# E BEYOND TRIVIAL EXPLANATIONS EXPERIMENTAL SETUP

The experimental benchmark proposed in Section 4.3 is performed on a subset of the validation set of CelebA. This subset is composed of 4 images for each CelebA attribute. From these 4 images, 2 were correctly classified by the black-box model, while the other 2 were misclassified. The two correctly classified images are chosen so that one was classified with a high confidence of $0.9$ and the other one with low confidence of $0.1$. The 2 misclassifications were chosen with the same criterion. The total size of the dataset is of 320 images. For each of these images we generate $k$ counterfactual explanations. From these counterfactuals, we report the ratio of successful explanations.

Here are the specific values we tried in our hyperparameter search: $\gamma \in [0.0, 0.001, 0.1, 1.0]$, $\alpha \in [0.0, 0.001, 0.1, 1.0]$, $\lambda \in [0.0001, 0.0005, 0.001]$, number of explanations 2 to 15 and learning rate $\in [0.05, 0.1]$. Be xGEM+ doesn't have a $\gamma$ nor $\alpha$ parameter, we increased its learning rate span to $[0.01, 0.05, 0.1]$ to reduce the gap in its search space compared with DiVE. We also changed the ranomd seeds and ran a total of 256 trials.

# F MODEL ARCHITECTURE

Table 4 presents the architecture of the encoder and decoder used in DiVE.

Table 4: DiCe architecture for $128 \times 128$ images. $ch$ represents the channel width multiplier in each network.

| |
|---|
| RGB image $x \in \mathbb{R}^{128 \times 128 \times 3}$ |
| ResBlock down $3ch \rightarrow 16ch$ |
| ResBlock $16ch \rightarrow 32ch$ |
| ResBlock down $32ch \rightarrow 32ch$ |
| ResBlock $32ch \rightarrow 64ch$ |
| ResBlock down $64ch \rightarrow 64ch$ |
| ResBlock $64ch \rightarrow 128ch$ |
| ResBlock down $128ch \rightarrow 128ch$ |
| ResBlock $128ch \rightarrow 128ch$ |
| ResBlock down $128ch \rightarrow 128ch$ |
| IN, Swish, Linear $128ch \times 4 \times 4 \rightarrow 128ch$ |
| IN, Swish, Linear $128ch \rightarrow 128ch$ |
| IN, Swish, Linear $128ch \rightarrow 128ch \times 2$ |
| $z \sim \mathcal{N}(\mu \in \mathbb{R}^{128}, \sigma \in \mathbb{R}^{128})$ |

(a) Encoder

| |
|---|
| $z \in \mathbb{R}^{128}$ |
| Linear $128ch \rightarrow 128ch$ |
| Linear $128ch \rightarrow 128ch$ |
| Linear $128ch \rightarrow 128ch \times 4 \times 4$ |
| ResBlock up $128ch \rightarrow 64ch$ |
| ResBlock up $64ch \rightarrow 32ch$ |
| ResBlock $32ch \rightarrow 16ch$ |
| ResBlock up $16ch \rightarrow 16ch$ |
| ResBlock $16ch \rightarrow 16ch$ |
| ResBlock up $16ch \rightarrow 16ch$ |
| ResBlock $16ch \rightarrow 16ch$ |
| IN, Swish, Conv $16ch \rightarrow 3$ tanh |

(b) Decoder

## G MODEL ALGORITHM

Algorithm 1 presents the steps needed for DiVE to generate explanations for a given black-box model using a sample input image.

**Algorithm 1:** Generating Explanations

**Input** : Sample image $x$, black-box model $f(\cdot)$
**Output** : Generated Conterfactuals $\tilde{\mathbf{x}}$

*Initialize a zero matrix parameter of size $n \times d$*
$\boldsymbol{\Sigma} \leftarrow zeros(n, d)$

*Get the original output from the black-box model*
$y \leftarrow f(x)$

*Extract the latent features of the original input*
$z \leftarrow q_\phi(x)$

*Obtain fisher information on $z$*
$f_z \leftarrow \boldsymbol{F}(z)$

*Obtain $k$ partitions using spectral clustering*
$\boldsymbol{P} \leftarrow SpectralClustering(f_z)$

*Initialize counter*
$i \leftarrow 0$

**while** $i < \tau$ **do**
  **for** *each* $\boldsymbol{\epsilon}, \boldsymbol{p} \in (\boldsymbol{\Sigma}, \boldsymbol{P})$ **do**
    *Perturb the latent features*
    $\tilde{\mathbf{x}} \leftarrow p_\theta(\mathbf{z} + \boldsymbol{\epsilon})$

    *Pass the perturbed image through the black-box*
    $\hat{y} \leftarrow f(\tilde{\mathbf{x}})$

    *Learn to reconstruct $\hat{Y}$ from $Y$*
    $\mathcal{L} \leftarrow$ compute Eq. 4

    Update $\boldsymbol{\epsilon}$ while masking a subset of the gradients
    $\boldsymbol{\epsilon} \leftarrow \boldsymbol{\epsilon} + \frac{\partial \mathcal{L}}{\partial \epsilon} \cdot p$
  **end**
  Update counter
  $i \leftarrow i + 1$
**end**