Black Box Explanation by Learning Image Exemplars in the Latent Feature Space

Riccardo Guidotti[1], Anna Monreale[2], Stan Matwin[3,4], and Dino Pedreschi[2]

[1] ISTI-CNR, Pisa, Italy, riccardo.guidotti@isti.cnr.it
[2] University of Pisa, Italy, {name.surname}@unipi.it
[3] Dalhousie University, stan@cs.dal.ca
[4] Institute of Computer Scicne, Polish Academy of Sciences

Abstract. We present an approach to explain the decisions of black box models for image classification. While using the black box to label images, our explanation method exploits the latent feature space learned through an adversarial autoencoder. The proposed method first generates exemplar images in the latent feature space and learns a decision tree classifier. Then, it selects and decodes exemplars respecting local decision rules. Finally, it visualizes them in a manner that shows to the user how the exemplars can be modified to either stay within their class, or to become counter-factuals by "morphing" into another class. Since we focus on black box decision systems for image classification, the explanation obtained from the exemplars also provides a saliency map highlighting the areas of the image that contribute to its classification, and areas of the image that push it into another class. We present the results of an experimental evaluation on three datasets and two black box models. Besides providing the most useful and interpretable explanations, we show that the proposed method outperforms existing explainers in terms of fidelity, relevance, coherence, and stability.

Keywords: Explainable AI, Adversarial Autoencoder, Image Exemplars.

1 Introduction

Automated decision systems based on machine learning techniques are widely used for classification, recognition and prediction tasks. These systems try to capture the relationships between the input instances and the target to be predicted. Input attributes can be of any type, as long as it is possible to find a convenient representation for them. For instance, we can represent images by matrices of pixels, or by a set of features that correspond to specific areas or patterns of the image. Many automated decision systems are based on very accurate classifiers such as deep neural networks. They are recognized to be "black box" models because of their opaque, hidden internal structure, whose complexity makes their comprehension for humans very difficult [5]. Thus, there is an increasing interest in the scientific community in deriving explanations able to

describe the behavior of a black box [5,22,13,6], or explainable by design approaches [19,18]. Moreover, the *General Data Protection Regulation*[5] has been approved in May 2018 by the European Parliament. This law gives to individuals the right to request "...meaningful information of the logic involved" when automated decision-making takes place with "legal or similarly relevant effects" on individuals. Without a technology able to explain, in a manner easily understandable to a human, how a black box takes its decision, this right will remain only an utopia, or it will result in prohibiting the use of opaque, but highly effective machine learning methods in socially sensitive domains.

In this paper, we investigate the problem of black box explanation for image classification (Section 3). Explaining the reasons for a certain decision can be particularly important. For example, when dealing with medical images for diagnosing, how we can validate that a very accurate image classifier built to recognize cancer actually focuses on the malign areas and not on the background for taking the decisions?

In the literature (Section 2), the problem is addressed by producing explanations through different approaches. On the one hand, gradient and perturbation-based attribution methods [27,25] reveal saliency maps highlighting the parts of the image that most contribute to its classification. However, these methods are *model specific* and can be employed only to explain specific deep neural networks. On the other hand, *model agnostic* approaches can explain, yet through a saliency map, the outcome of any black box [24,12]. Agnostic methods may generate a local neighborhood of the instance to explain and mime the behavior of the black box using an interpretable classifier. However, these methods exhibit drawbacks that may negatively impact the reliability of the explanations. First, they do not take into account existing relationships between features (or pixels) during the neighborhood generation. Second, the neighborhood generation does not produce "meaningful" images since, e.g., some areas of the image to explain in [24] are obscured, while in [12] they are replaced with pixels of other images. Finally, transparent-by-design approaches produce prototypes from which it should be clear to the user why a certain decision is taken by the model [18,19]. Nevertheless, these approaches cannot be used to explain a trained black box, but the transparent model has to be directly adopted as a classifier, possibly with limitations on the accuracy achieved.

We propose ABELE, an Adversarial Black box Explainer generating Latent Exemplars (Section 5). ABELE is a local, model-agnostic explanation method able to overcome the existing limitations of the local approaches by exploiting the latent feature space, learned through an adversarial autoencoder [20] (Section 4), for the neighborhood generation process. Given an image classified by a given black box model, ABELE provides an explanation for the reasons of the proposed classification. The explanation consists of two parts: *(i)* a set of *exemplars* and *counter-exemplars* images illustrating, respectively, instances classified with the same label and with a different label than the instance to explain, which may be visually analyzed to understand the reasons for the classification, and *(ii)* a

[5] `https://ec.europa.eu/justice/smedataprotect/`

saliency map highlighting the areas of the image to explain that contribute to its classification, and areas of the image that push it towards another label.

We present a deep experimentation (Section 6) on three datasets of images and two black box models. We empirically prove that ABELE overtakes state of the art methods based on saliency maps or on prototype selection by providing relevant, coherent, stable and faithful explanations. Finally, we summarize our contribution, its limitations, and future research directions (Section 7).

2 Related Work

Research on black box explanation methods has recently received much attention [5,22,13,6]. These methods can be characterized as model-specific *vs* model-agnostic, and local *vs* global. The proposed explanation method ABELE is the next step in the line of research on local, model-agnostic methods originated with [24] and extended in different directions by [9] and by [11,12,23].

In image classification, typical explanations are the *saliency maps*, i.e., images that show each pixel's positive (or negative) contribution to the black box outcome. Saliency maps are efficiently built by gradient [27,25,30,1] and perturbation-based [33,7] attribution methods by finding, through backpropagation and differences on the neuron activation, the pixels of the image that maximize an approximation of a linear model of the black box classification outcome. Unfortunately, these approaches are specifically designed for deep neural networks. They cannot be employed for explaining other image classifiers, like tree ensembles or hybrid image classification processes [13]. Model-agnostic explainers, such as LIME [24] and similar [12] can be employed to explain the classification of any image classifier. They are based on the generation of a local neighborhood around the image to explain, and on the training of an interpretable classifier on this neighborhood. Unlike the global distillation methods [17], they do not consider (often non-linear) relationships between features (e.g. pixel proximity), and thus, their neighborhoods do not contain "meaningful" images.

Our proposed method ABELE overcomes the limitations of both saliency-based and local model-agnostic explainers by using AAEs, local distillation, and exemplars. As ABELE includes and extends LORE [11], an innovation w.r.t. state of the art explainers for image classifiers is the usage of counter-factuals. Counter-factuals are generated from "positive" instances by a minimal perturbation that pushes them to be classified with a different label [31]. In line with this approach, ABELE generates counter-factual rules in the latent feature space and exploits them to derive counter-exemplars in the original feature space.

As the explanations returned by ABELE are based on exemplars, we need to clarify the relationship between *exemplars* and *prototypes*. Both are used as a foundation of representation of a category, or a concept [8]. In the prototype view, a concept is the representation of a specific instance of this concept. In the exemplar view, the concept is represented by means of a set of typical examples, or exemplars. ABELE uses exemplars to represent a concept. In recent works [19,4], image prototypes are used as the foundation of the concept for interpretabil-

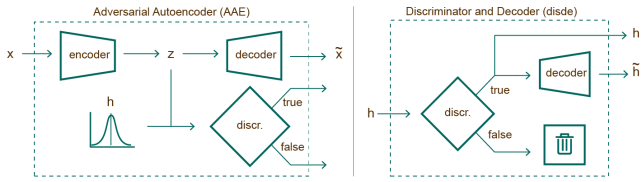

Fig. 1. *Left*: Adversarial Autoencoder architecture: the *encoder* turns the image x into its latent representation z, the *decoder* re-builds an approximation $\widetilde{x}$ of x from z, and the *discriminator* identifies if a randomly generated latent instance h can be considered valid or not. *Right*: Discriminator and Decoder (*disde*) module: input is a randomly generated latent instance h and, if it is considered valid by the *discriminator*, it returns it together with its decompressed version $\widetilde{h}$.

ity [2]. In [19], an explainable by design method, similarly to ABELE, generates prototypes in the latent feature space learned with an autoencoder. However, it is not aimed at explaining a trained black box model. In [4] a convolutional neural network is adopted to provide features from which the prototypes are selected. ABELE differs from these approaches because is model agnostic and the *adversarial* component ensures the similarity of feature and class distributions.

3 Problem Formulation

In this paper we address the *black box outcome explanation problem* [13]. Given a black box model b and an instance x classified by b, i.e., $b(x) = y$, our aim is to provide an explanation e for the decision $b(x) = y$. More formally:

Definition 1. Let b be a black box, and x an instance whose decision $b(x)$ has to be explained. The *black box outcome explanation problem* consists in finding an explanation $e \in E$ belonging to a human-interpretable domain E.

We focus on the black box outcome explanation problem for image classification, where the instance x is an image mapped by b to a class label y. In the following, we use the notation $b(X) = Y$ as a shorthand for $\{b(x) \mid x \in X\} = Y$. We denote by b a black box image classifier, whose internals are either unknown to the observer or they are known but uninterpretable by humans. Examples are neural networks and ensemble classifiers.We assume that a black box b is a function that can be queried at will.

We tackle the above problem by deriving an explanation from the understanding of the behavior of the black box in the local neighborhood of the instance to explain [13]. To overcome the state of the art limitations, we exploit adversarial autoencoders [20] for generating, encoding and decoding the local neighborhood.

4 Adversarial Autoencoders

An important issue arising in the use of synthetic instances generated when developing black box explanations is the question of maintaining the identity of the distribution of the examples that are generated with the prior distribution of the original examples. We approach this issue by using an Adversarial Autoencoder (AAE) [20], which combines a Generative Adversarial Network (GAN) [10] with the autoencoder representation learning algorithm. Another reason for the use of AAE is that, as demonstrated in [29], the use of autoencoders enhances the robustness of deep neural network classifiers more against malicious examples.

AAEs are probabilistic autoencoders that aim at generating new random items that are highly similar to the training data. They are regularized by matching the aggregated posterior distribution of the latent representation of the input data to an arbitrary prior distribution. The AAE architecture (Fig. 1-left) includes an *encoder* : $\mathbb{R}^n \to \mathbb{R}^k$, a *decoder* : $\mathbb{R}^k \to \mathbb{R}^n$ and a *discriminator* : $\mathbb{R}^k \to [0, 1]$ where n is the number of pixels in an image and k is the number of latent features. Let x be an instance of the training data, we name z the corresponding latent data representation obtained by the *encoder*. We can describe the AAE with the following distributions [20]: the prior distribution $p(z)$ to be imposed on z, the data distribution $p_d(x)$, the model distribution $p(x)$, and the encoding and decoding distributions $q(z|x)$ and $p(x|z)$, respectively. The encoding function $q(z|x)$ defines an aggregated posterior distribution of $q(z)$ on the latent feature space: $q(z) = \int_x q(z|x)p_d(x)dx$. The AAE guarantees that the aggregated posterior distribution $q(z)$ matches the prior distribution $p(z)$, through the latent instances and by minimizing the reconstruction error. The AAE generator corresponds to the encoder $q(z|x)$ and ensures that the aggregated posterior distribution can confuse the *discriminator* in deciding if the latent instance z comes from the true distribution $p(z)$.

The AAE learning involves two phases: the *reconstruction* aimed at training the *encoder* and *decoder* to minimize the reconstruction loss; the *regularization* aimed at training the *discriminator* using training data and encoded values. After the learning, the decoder defines a generative model mapping $p(z)$ to $p_d(x)$.

5 Adversarial Black Box Explainer

ABELE (Adversarial Black box Explainer generating Latent Exemplars) is a local model agnostic explainer for image classifiers solving the outcome explanation problem. Given an image x to explain and a black box b, the explanation provided by ABELE is composed of *(i)* a set of *exemplars* and *counter-exemplars*, *(ii)* a *saliency map*. Exemplars and counter-exemplars shows instances classified with the same and with a different outcome than x. They can be visually analyzed to understand the reasons for the decision. The saliency map highlights the areas of x that contribute to its classification and areas that push it into another class.

The explanation process involves the following steps. First, ABELE generates a neighborhood in the latent feature space exploiting the AAE (Sec. 4). Then, it

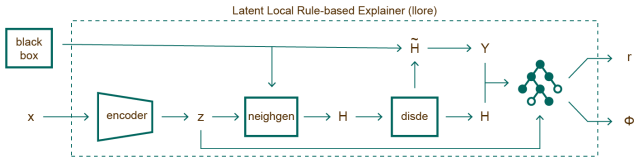

Fig. 2. Latent Local Rules Extractor. It takes as input the image x to explain and the black box b. With the *encoder* trained by the AAE, it turns x into its latent representation z. Then, the *neighgen* module uses z and b to generate the latent local neighborhood H. The valid instances are decoded in $\widetilde{H}$ by the *disde* module. Images in $\widetilde{H}$ are labeled with the black box $Y = b(\widetilde{H})$. H and Y are used to learn a decision tree classifier. At last, a decision rule r and the counter-factual rules Φ for z are returned.

learns a decision tree on that latent neighborhood providing local decision and counter-factual rules. Finally, ABELE selects and decodes exemplars and counter-exemplars satisfying these rules and extracts from them a saliency map.

Encoding. The image $x \in \mathbb{R}^n$ to be explained is passed as input to the AAE where the *encoder* returns the latent representation $z \in \mathbb{R}^k$ using k latent features with $k \ll n$. The number k is kept low by construction avoiding high dimensionality problems.

Neighborhood Generation. ABELE generates a set H of N instances in the latent feature space, with characteristics close to those of z. Since the goal is to learn a predictor on H able to simulate the local behavior of b, the neighborhood includes instances with both decisions, i.e., $H = H_{=} \cup H_{\neq}$ where instances $h \in H_{=}$ are such that $b(\widetilde{h}) = b(x)$, and $h \in H_{\neq}$ are such that $b(\widetilde{h}) \neq b(x)$. We name $\widetilde{h} \in \mathbb{R}^n$ the decoded version of an instance $h \in \mathbb{R}^k$ in the latent feature space. The neighborhood generation of H (*neighgen* module in Fig. 2) may be accomplished using different strategies ranging from pure random strategy using a given distribution to a genetic approach maximizing a fitness function [11]. In our experiments we adopt the last strategy. After the generation process, for any instance $h \in H$, ABELE exploits the *disde* module (Fig. 1-right) for both checking the validity of h by querying the *discriminator*[6] and decoding it into $\widetilde{h}$. Then, ABELE queries the black box b with $\widetilde{h}$ to get the class y, i.e., $b(\widetilde{h}) = y$.

Local Classifier Rule Extraction. Given the local neighborhood H, ABELE builds a decision tree classifier c trained on the instances H labeled with the black box decision $b(\widetilde{H})$. Such a predictor is intended to locally mimic the behavior of b in the neighborhood H. The decision tree extracts the decision rule r and counter-factual rules Φ enabling the generation of *exemplars* and *counter-exemplars*. ABELE considers decision tree classifiers because: *(i)* decision rules can naturally be derived from a root-leaf path in a decision tree; and, *(ii)* counter-factual rules can be extracted by symbolic reasoning over a decision tree. The

[6] In the experiments we use for the *discriminator* the default validity threshold 0.5 to distinguish between real and fake exemplars. This value can be increased to admit only more reliable exemplars, or decreased to speed-up the generation process.

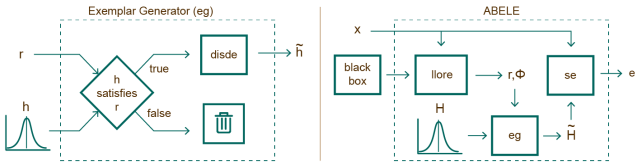

Fig. 3. *Left*: (Counter-)Exemplar Generator: it takes a decision rule r and a randomly generated latent instance h, checks if h satisfies r and applies the *disde* module (Fig.1-right) to decode it. *Right*: ABELE architecture. It takes as input the image x for which we require an explanation and the black box b. It extracts the decision rule r and the counter-factual rules Φ with the *llore* module. Then, it generates a set of latent instances H which are used as input with r and Φ for the *eg* module (Fig. 3-left) to generate exemplars and counter-exemplars $\widetilde{H}$. Finally, x and $\widetilde{H}$ are used by the *se* module for calculating the saliency maps and returning the final explanation e.

premise p of a decision rule $r{=}p{\rightarrow}y$ is the conjunction of the splitting conditions in the nodes of the path from the root to the leaf that is satisfied by the latent representation z of the instance to explain x, and setting $y{=}c(z)$. For the counter-factual rules Φ, ABELE selects the closest rules in terms of splitting conditions leading to a label $\hat{y}$ different from y, i.e., the rules $\{q{\rightarrow}\hat{y}\}$ such that q is the conjunction of splitting conditions for a path from the root to the leaf labeling an instance h_c with $c(h_c){=}\hat{y}$ and minimizing the number of splitting conditions falsified w.r.t. the premise p of the rule r. Fig. 2 shows the process that, starting from the image to be explained, leads to the decision tree learning, and to the extraction of the decision and counter-factual rules. We name this module *llore*, as a variant of LORE [11] operating in the latent feature space.

Explanation Extraction. Often, e.g. in medical or managerial decision making, people explain their decisions by pointing to exemplars with the same (or different) decision outcome [8,4]. We follow this approach and we model the explanation of an image x returned by ABELE as a triple $e = \langle \widetilde{H}_e, \widetilde{H}_c, s \rangle$ composed by *exemplars* $\widetilde{H}_e$, *counter-exemplars* $\widetilde{H}_c$ and a *saliency map* s. Exemplars and counter-exemplars are images representing instances similar to x, leading to an outcome equal to or different from $b(x)$. Exemplars and counter-exemplars are generated by ABELE exploiting the *eg* module (Fig. 3-left). It first generates a set of latent instances H satisfying the decision rule r (or a set of counter-factual rules Φ), as shown in Fig. 2. Then, it validates and decodes them into exemplars $\widetilde{H}_e$ (or counter-exemplars $\widetilde{H}_c$) using the *disde* module. The saliency map s highlights areas of x that contribute to its outcome and areas that push it into another class. The map is obtained by the saliency extractor *se* module (Fig. 3-right) that first computes the pixel-to-pixel-difference between x and each exemplar in the set $\widetilde{H}_e$, and then, it assigns to each pixel of the saliency map s the median value of all differences calculated for that pixel. Thus, formally for each pixel i of the saliency map s we have: $s[i] = median_{\forall \widetilde{h}_e \in \widetilde{H}_e}(x[i] - \widetilde{h}_e[i])$.

Table 1. Datasets resolution, type of color, train and test dimensions, and black box model accuracy.

dataset	resolution	rgb	train	test	RF	DNN
`mnist`	28×28	✗	$60k$	$10k$	.9692	.9922
`fashion`	28×28	✗	$60k$	$10k$	.8654	.9207
`cifar10`	32×32	✓	$50k$	$10k$	.4606	.9216

Table 2. AAEs reconstruction error in terms of RMSE.

dataset	train	test
`mnist`	39.80	43.64
`fashion`	27.41	30.15
`cifar10`	20.26	45.12

In summary, ABELE (Fig. 3-right), takes as input the instance to explain x and a black box b, and returns an explanation e according to the following steps. First, it adopts *llore* [11] to extract the decision rule r and the counterfactual rules Φ. These rules, together with a set of latent random instances H are the input of the *eg* module returning *exemplars* and *counter-exemplars*. Lastly, the *se* module extracts the *saliency map* starting from the image x and its exemplars.

6 Experiments

We experimented with the proposed approach on three open source datasets[7] (details in Table 1): the `mnist` dataset of handwritten digit grayscale images, the `fashion` mnist dataset is a collection of Zalando's article grayscale images (e.g. shirt, shoes, bag, etc.), and the `cifar10` dataset of colored images of airplanes, cars, birds, cats, etc. Each dataset has ten different labels.

We trained and explained away the following black box classifiers. Random Forest [3] (RF) as implemented by the *scikit-learn* Python library, and Deep Neural Networks (DNN) implemented with the `keras` library[8]. For `mnist` and `fashion` we used a three-layer CNN, while for `cifar10` we used the *ResNet20 v1* network described in [16]. Classification performance are reported in Table 1.

For `mnist` and `fashion` we trained AAEs with sequential three-layer encoder, decoder and discriminator. For `cifar10` we adopted a four-layer CNN for the encoder and the decoder, and a sequential discriminator. We used 80% of the test sets for training the adversarial autoencoders[9]. In Table 2 we report the reconstruction error of the AAE in terms of *Root Mean Square Error* (RMSE) between the original and reconstructed images. We employed the remaining 20% for evaluating the quality of the explanations.

We compare ABELE against LIME and a set of saliency-based explainers collected in the `DeepExplain` package[10]: Saliency (SAL) [27], GradInput (GRAD) [25], IntGrad (INTG) [30], ε-lrp (ELRP) [1], and Occlusion (OCC) [33]. We refer to the

[7] Dataset: `http://yann.lecun.com/exdb/mnist/`, `https://www.cs.toronto.edu/~kriz/cifar.html`, `https://www.kaggle.com/zalando-research/`.

[8] Black box: `https://scikit-learn.org/`, `https://keras.io/examples/`.

[9] The encoding distribution of AAE is defined as a Gaussian distribution whose mean and variance is predicted by the encoder itself [20]. We adopted the following number of latent features k for the various datasets: `mnist` k=4, `fashion` k=8, `cifar10` k=16.

[10] Github code links: `https://github.com/riccotti/ABELE`, `https://github.com/marcotcr/lime`, `https://github.com/marcoancona/DeepExplain` .

Fig. 4. Explain by saliency map `mnist`.

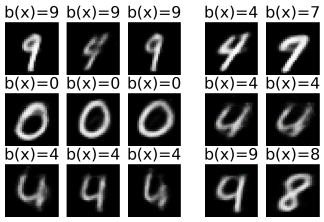

Fig. 5. Exemplars & counter-exemplars.

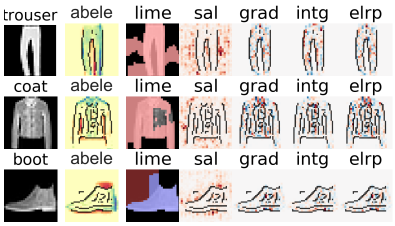

Fig. 6. Explain by saliency map `fashion`.

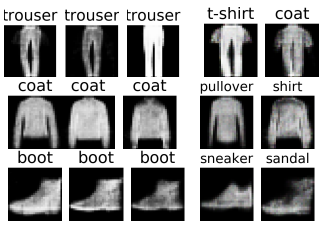

Fig. 7. Exemplars & counter-exemplars.

set of tested `DeepExplain` methods as DEX. We also compare the exemplars and counter-exemplars generated by ABELE against the prototypes and criticisms[11] selected by the MMD and K-MEDOIDS [18]. MMD exploits the maximum mean discrepancy and a kernel function for selecting the best prototypes and criticisms.

Saliency Map, Exemplars and Counter-Exemplars. Before assessing quantitatively the effectiveness of the compared methods, we visually analyze their outcomes. We report explanations of the DNNs for the `mnist` and `fashion` datasets in Fig. 4 and Fig. 6 respectively[12]. The first column contains the image to explain x together with the label provided by the black box b, while the second column contains the saliency maps provided by ABELE. Since they are derived from the difference between the image x and its exemplars, we indicate with yellow color the areas that are common between x and the exemplars $\widetilde{H}_e$, with red color the areas contained only in the exemplars and blue color the areas contained only in x. This means that yellow areas must remain unchanged to obtain the same label $b(x)$, while red and blue areas can change without impacting the black box decision. In particular, with respect to x, an image obtaining the same label can be darker in blue areas and lighter in red areas. In other words, blue and red areas express the boundaries that can be varied, and for which the class remains unchanged. For example, with this type of saliency map we can understand that a *nine* may have a more compact circle, a *zero* may be more inclined (Fig. 4), a *coat* may have no space between the sleeves and the

[11] Criticisms are images not well-explained by prototypes with a regularized kernel function [18].

[12] Best view in color. Black lines are not part of the explanation, they only highlight borders. We do not report explanations for `cifar10` and for RF for the sake of space.

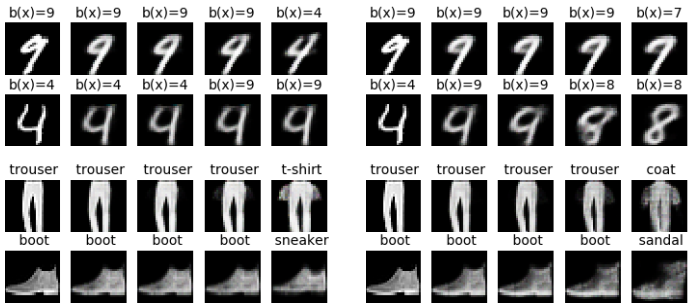

Fig. 8. Interpolation from the image to explain x to one of its counter-exemplars $\widetilde{h_c}$.

body, and that a *boot* may have a higher neck (Fig. 6). Moreover, we can notice how, besides the background, there are some "essential" yellow areas within the main figure that can not be different from x: e.g. the leg of the *nine*, the crossed lines of the *four*, the space between the two *trousers*.

The rest of the columns in Fig. 4 and 6 contain the explanations of the competitors: red areas contribute positively to the black box outcome, blue areas contribute negatively. For LIME's explanations, nearly all the content of the image is part of the saliency areas[13]. In addition, the areas have either completely positive or completely negative contributions. These aspects can be not very convincing for a LIME user. On the other hand, the DEX methods return scattered red and blue points which can also be very close to each other and are not clustered into areas. It is not clear how a user could understand the black box outcome decision process from this kind of explanation.

Since the ABELE's explanations also provide exemplars and counter-exemplars, they can also be visually analyzed by a user for understanding which are possible similar instances leading to the same outcome or to a different one. For each instance explained in Fig. 4 and 6, we show three exemplars and two counter-exemplars for the `mnist` and `fashion` datasets in Fig. 5 and 7, respectively. Observing these images we can notice how the label *nine* is assigned to images very close to a *four* (Fig. 5, 1^{st} row, 2^{nd} column) but until the upper part of the circle remains connected, it is still classified as a *nine*. On the other hand, looking at counter-exemplars, if the upper part of the circle has a hole or the lower part is not thick enough, then the black box labels them as a *four* and a *seven*, respectively. We highlight similar phenomena for other instances: e.g. a *boot* with a neck not well defined is labeled as a *sneaker* (Fig. 7).

To gain further insights on the counter-exemplars, inspired by [28], we exploit the latent representations to visually understand how the black box labeling changes w.r.t. real images. In Fig. 8 we show, for some instances previously analyzed, how they can be changed to move from the original label to the counter-factual label. We realize this change in the class through the latent representations z and h_c of the image to explain x and of the counter-exemplar

[13] This effect is probably due to the figure segmentation performed by LIME.

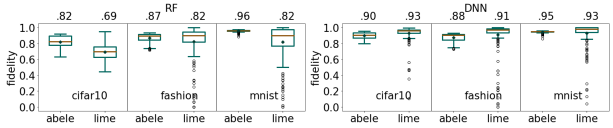

Fig. 9. Box plots of *fidelity*. Numbers on top: mean values (the higher the better).

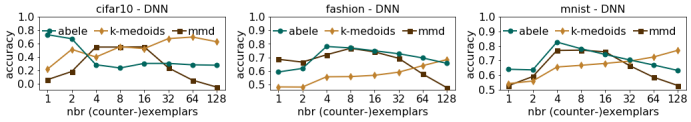

Fig. 10. 1-NN exemplar classifier accuracy varying the number of (counter-)exemplars.

$\widetilde{h}_c$, respectively. Given z and h_c, we generate through linear interpolation in the latent feature space intermediate latent representations $z<h_c^{(i)}<h_c$ respecting the latent decision or counter-factual rules. Finally, using the *decoder*, we obtain the intermediate images $\widetilde{h}_c^{(i)}$. This convincing and useful explanation analysis is achieved thanks to ABELE's ability to deal with both real and latent feature spaces, and to the application of latent rules to real images which are human understandable and also clear exemplar-based explanations.

Lastly, we observe that prototype selector methods, like MMD [18] and K-MEDOIDS cannot be used for the same type of analysis because they lack any link with either the black box or the latent space. In fact, they propose as prototypes (or criticism) existing images of a given dataset. On the other hand, ABELE generates and does not select (counter-)exemplars respecting rules.

Interpretable Classifier Fidelity. We compare ABELE and LIME in terms of *fidelity* [11,5], i.e., the ability of the local interpretable classifier c[14] of mimicking the behavior of a black box b in the local neighborhood H: $\mathit{fidelity}(H, \widetilde{H}) = \mathit{accuracy}(b(\widetilde{H}), c(H))$. We report the fidelity as box plots in Fig. 9. The results show that on all datasets ABELE outperforms LIME with respect to the RF black box classifier. For the DNN the interpretable classifier of LIME is slightly more faithful. However, for both RF and DNN, ABELE has a fidelity variance markedly lower than LIME, i.e., more compact box plots also without any outlier[15]. Since these fidelity results are statistically significant, we observe that the local interpretable classifier of ABELE is more faithful than the one of LIME.

Nearest Exemplar Classifier. The goal of ABELE is to provide useful exemplars and counter-exemplars as explanations. However, since we could not validate them with an experiment involving humans, inspired by [18], we tested their effectiveness by adopting memory-based machine learning techniques such as the k-nearest neighbor classifier [2] (k-NN). This kind of experiment provides an objective and indirect evaluation of the quality of exemplars and counter-exemplars.

[14] A decision tree for ABELE and a linear lasso model for LIME.
[15] These results confirm the experiments reported in [11].

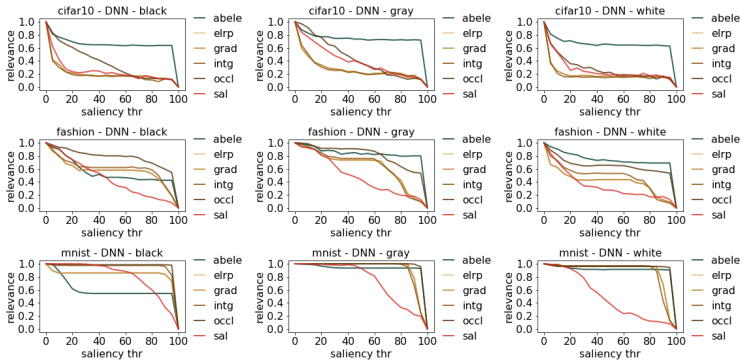

Fig. 11. Relevance analysis varying the percentile threshold τ (the higher the better).

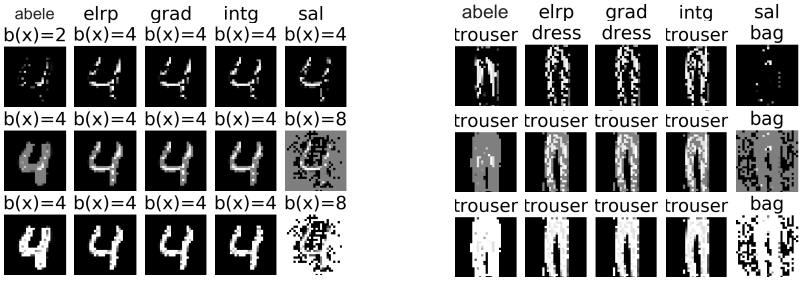

Fig. 12. Images masked with *black*, *gray* and *white* having pixels with saliency for DNN lower than $\tau = 70\%$ for the explanations of *four* and *trouser* in Fig. 4 and 6.

In the following experiment we generated n exemplars and counter-exemplars with ABELE, and we selected n prototypes and criticisms using MMD [18] and K-MEDOIDS [2]. Then, we employ a 1-NN model to classify unseen instances using these exemplars and prototypes. The classification accuracy of the 1-NN models trained with exemplars and counter-exemplars generated to explain the DNN reported in Fig. 10 is comparable among the various methods[16]. In particular, we observe that when the number of exemplars is low ($1 \leq n \leq 4$), ABELE outperforms MMD and K-MEDOIDS. This effect reveals that, on the one hand, just a few exemplars and counter-exemplars *generated* by ABELE are good for recognizing the real label, but if the number increases the 1-NN is getting confused. On the other hand, MMD is more effective when the number of prototypes and criticisms is higher: it *selects* a good set of images for the 1-NN classifier.

Relevance Evaluation. We evaluate the effectiveness of ABELE by partly masking the image to explain x. According to [15], although a part of x is masked, $b(x)$ should remain unchanged as long as relevant parts of x remain unmasked. To quantitatively measure this aspect, we define the *relevance* metric as the ratio of images in X for which the masking of relevant parts does not impact on the

[16] The ABELE method achieves similar results for RF not reported due to lack of space.

Table 3. Coherence analysis for DNN classifier (the lower the better).

dataset	ABELE	ELRP	GRAD	INTG	LIME	OCC	SAL
`cifar10`	.575 ± .10	.542 ± .08	.542 ± .08	.532 ± .11	1.919 ± .25	1.08 ± .23	.471 ± .05
`fashion`	.451 ± .06	.492 ± .10	.492 ± .10	.561 ± .17	1.618 ± .16	.904 ± .23	.413 ± .03
`mnist`	.380 ± .03	.740 ± .21	.740 ± .21	.789 ± .22	1.475 ± .14	.734 ± .21	.391 ± .03

Table 4. Stability analysis for DNN classifier (the lower the better).

dataset	ABELE	ELRP	GRAD	INTG	LIME	OCC	SAL
`cifar10`	.575 ± .10	.518 ± .08	.518 ± .08	.561 ± .10	1.898 ± .29	.957 ± .14	.468 ± .05
`fashion`	.455 ± .06	.490 ± .09	.490 ± .09	.554 ± .18	1.616 ± .17	.908 ± .23	.415 ± .03
`mnist`	.380 ± .04	.729 ± .21	.729 ± .21	.776 ± .22	1.485 ± .14	.726 ± .21	.393 ± .03

black box decision. Let $E=\{e_1, ..., e_n\}$ be the set of explanations for the instances $X=\{x_1, ..., x_n\}$. We identify with $x_m^{\{e,\tau\}}$ the masked version of x with respect to the explanation e and a threshold mask τ. Then, the explanation *relevance* is defined as: $relevance_\tau(X, E) = |\{x \mid b(x) = b(x_m^{\{e,\tau\}}) \; \forall \langle x, e\rangle \in \langle X, E\rangle\}| \; / \; |X|$. The masking $x_m^{\{e,\tau\}}$ is got by changing the pixels of x having a value in the saliency map $s \in e$ smaller than the τ percentile of the set of values in the saliency map itself. These pixels are substituted with the color *0*, *127* or *255*, i.e. *black*, *gray* or *white*. A low number of black box outcome changes means that the explainer successfully identifies *relevant* parts of the images, i.e., parts having a high relevance. Fig. 11 shows the *relevance* for the DNN[17] varying the percentile of the threshold from 0 to 100. The ABELE method is the most resistant to image masking in `cifar10` regardless of the color used. For the other datasets we observe a different behavior depending on the masking color used: ABELE is among the best performer if the masking color is *white* or *gray*, while when the mask color is *black*, ABELE's relevance is in line with those of the competitors for `fashion` and it is not good for `mnist`. This effect depends on the masking color but also on the different definitions of saliency map. Indeed, as previously discussed, depending on the explainer, a saliency map can provide different knowledge. However, we can state that ABELE successfully identifies relevant parts of the image contributing to the classification.

For each method and for each masking color, Fig. 12 shows the effect of the masking on a sample from `mnist` and another from `fashion`. It is interesting to notice how for the SAL approach a large part of the image is quite relevant, causing a different black box outcome (reported on the top of each image). As already observed previously, a peculiarity of ABELE is that the saliency areas are more connected and larger than those of the other methods. Therefore, given a percentile threshold τ, the masking operation tends to mask more contiguous and bigger areas of the image while maintaining the same black box labeling.

Robustness Assessment. For gaining the trust of the user, it is crucial to analyze the stability of interpretable classifiers and explainers [14] since the

[17] The ABELE method achieves similar results for RF not reported due to lack of space.

Table 5. Coherence (left) and stability (right) for RF classifier (the lower the better).

dataset	ABELE	LIME
`cifar10`	$.794 \pm .34$	$1.692 \pm .32$
`fashion`	$.821 \pm .37$	$2.534 \pm .70$
`mnist`	$.568 \pm .29$	2.593 ± 1.25

dataset	ABELE	LIME
`cifar10`	$.520 \pm .14$	$1.460 \pm .23$
`fashion`	$.453 \pm .06$	$1.464 \pm .18$
`mnist`	$.371 \pm .04$	$1.451 \pm .17$

stability of explanations is an important requirement for interpretability [21]. Let E=$\{e_1, ..., e_n\}$ be the set of explanations for X=$\{x_1, ..., x_n\}$, and $\{s_1, ..., s_n\}$ the corresponding saliency maps. We asses the *robustness* through the local Lipschitz estimation [21]: $robustness(x) = argmax_{x_i \in \mathcal{N}(x)}(\|s_i - s\|_2 / \|x_i - x\|_2)$ with $\mathcal{N}(x) = \{x_j \in X \mid \|x_j - x\|_2 \leq \epsilon\}$. Here x is the image to explain and s is the saliency map of its explanation e. We name *coherence* the explainer's ability to return similar explanations to instances labeled with the same black box outcome, i.e., similar instances. We name *stability*, often called also *sensitivity*, the capacity of an explainer of not varying an explanation in the presence of noise with respect to the explained instance. Therefore, for coherence the set X in the *robustness* formula is formed by real instances, while for stability X is formed by the instances to explain modified with random noise[18].

Tables 3 and 4 report mean and standard deviation of the local Lipschitz estimations of the explainers' *robustness* in terms of *coherence* and *stability*, respectively. As showed in [21], our results confirm that LIME does not provide robust explanations, GRAD and INTG are the best performers, and ABELE performance is comparable to them in terms of both *coherence* and *stability*. This high resilience of ABELE is due to the usage of AAE, which is also adopted for image denoising [32]. Table 5 shows the robustness in terms of coherence and stability for the model agnostic explainers ABELE and LIME with respect to the RF. Again, ABELE presents a more robust behavior than LIME. Fig. 13 and 14 compare the saliency maps of a selected image from `mnist` and `fashion` labeled with DNN. Numbers on the top represent the ratio in the robustness formula. Although there is no change in the black box outcome, we can see how for some of the other explainers like LIME, ELRP, and GRAD, the saliency maps vary considerably. On the other hand, ABELE's explanations remain coherent and stable. We observe how in both *nines* and *boots* the yellow fundamental area does not change especially within the image's edges. Also the red and blue parts, that can be varied without impacting on the classification, are almost identical, e.g. the *boots*' neck and the sole in Fig. 13, or the top left of the *zero* in Fig. 14.

7 Conclusion

We have presented ABELE, a local model-agnostic explainer using the latent feature space learned through an adversarial autoencoder for the neighborhood generation process. The explanation returned by ABELE consists of exemplar

[18] As in [21], in our experiments, we use ϵ=0.1 for $\mathcal{N}$ and we add salt and pepper noise.

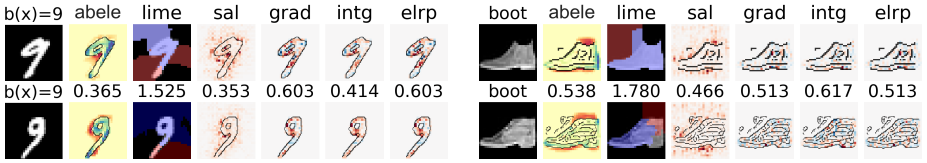

Fig. 13. Saliency maps for `mnist` (left) and `fashion` (right) comparing two images with the same DNN outcome; numbers on the top are the *coherence* (the lower the better).

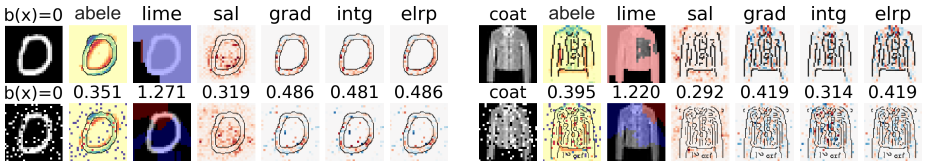

Fig. 14. Saliency maps for `mnist` (left) and `fashion` (right) comparing the original image in the first row and the modified version with salt and pepper noise but with the same DNN outcome; numbers on the top are the *stability* (the lower the better).

and counter-exemplar images, labeled with the class identical to, and different from, the class of the image to explain, and by a a saliency map, highlighting the importance of the areas of the image contributing to its classification. An extensive experimental comparison with state of the art methods shows that ABELE addresses their deficiencies, and outperforms them by returning coherent, stable and faithful explanations.

The method has some limitations: it is constrained to image data and does not enable casual or logical reasoning. Several extensions and future work are possible. First, we would like to investigate the effect on the explanations of changing some aspect of the AAE: *(i)* the latent dimensions k, *(ii)* the rigidity of the *discriminator* in admitting latent instances,*(iii)* the type of autoencoders (e.g. variational autoencoders [26]). Second, we would like to extend ABELE to make it work on tabular data and on text. Third, we would employ ABELE in a case study generating exemplars and counter-exemplars for explaining medical imaging tasks, e.g. radiography and fMRI images. Lastly, we would conduct extrinsic interpretability evaluation of ABELE. Human decision-making in a specific task (e.g. multiple-choice question answering) would be driven by ABELE explanations, and these decisions could be objectively and quantitatively evaluated.

Acknowledgements. This work is partially supported by the EC H2020 programme under the funding schemes: Research Infrastructures G.A. 654024 *SoBigData*, G.A. 78835 *Pro-Res*, G.A. 825619 *AI4EU* and G.A. 780754 *Track&Know*. The third author acknowledges the support of the Natural Sciences and Engineering Research Council of Canada and of the Ocean Frontiers Institute.

References

1. S. Bach, A. Binder, et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.
2. J. Bien et al. Prototype selection for interpretable classification. *AOAS*, 2011.
3. L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
4. C. Chen, O. Li, A. Barnett, J. Su, and C. Rudin. This looks like that: deep learning for interpretable image recognition. *arXiv:1806.10574*, 2018.
5. F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *arXiv:1702.08608*, 2017.
6. H. J. Escalante, S. Escalera, I. Guyon, et al. *Explainable and interpretable models in computer vision and machine learning*. Springer, 2018.
7. R. C. Fong and A. Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *ICCV*, pages 3429–3437, 2017.
8. M. Frixione et al. Prototypes vs exemplars in concept representation. KEOD, 2012.
9. N. Frosst et al. Distilling a neural network into a soft decision tree. *arXiv:1711.09784*, 2017.
10. I. Goodfellow et al. Generative adversarial nets. In *NIPS*, 2014.
11. R. Guidotti et al. Local rule-based explanations of black box decision systems. *arXiv:1805.10820*, 2018.
12. R. Guidotti, A. Monreale, and L. Cariaggi. Investigating neighborhood generation for explanations of image classifiers. In *PAKDD*, 2019.
13. R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, et al. A survey of methods for explaining black box models. *ACM CSUR*, 51(5):93:1–42, 2018.
14. R. Guidotti and S. Ruggieri. On the stability of interpretable models. IJCNN, 2019.
15. S. Hara et al. Maximally invariant data perturbation as explanation. *arXiv:1806.07004*, 2018.
16. K. He et al. Deep residual learning for image recognition. In *CVPR*, 2016.
17. G. Hinton et al. Distilling the knowledge in a neural network. *arXiv:1503.02531*, 2015.
18. B. Kim et al. Examples are not enough, learn to criticize! In *NIPS*, 2016.
19. O. Li, H. Liu, C. Chen, and C. Rudin. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In AAAI, 2018.
20. A. Makhzani, J. Shlens, et al. Adversarial autoencoders. *arXiv:1511.05644*, 2015.
21. D. A. Melis and T. Jaakkola. Towards robust interpretability with self-explaining neural networks. In *NIPS*, 2018.
22. C. Molnar. *Interpretable machine learning*. LeanPub, 2018.
23. C. Panigutti, R. Guidotti, A. Monreale, and D. Pedreschi. Explaining multi-label black-box classifiers for health applications. In *W3PHIAI*, 2019.
24. M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *KDD*, pages 1135–1144. ACM, 2016.
25. A. Shrikumar et al. Not just a black box: Learning important features through propagating activation differences. *arXiv:1605.01713*, 2016.
26. N. Siddharth, B. Paige, A. Desmaison, V. de Meent, et al. Inducing interpretable representations with variational autoencoders. *arXiv:1611.07492*, 2016.
27. K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv:1312.6034*, 2013.
28. T. Spinner et al. Towards an interpretable latent space: an intuitive comparison of autoencoders with variational autoencoders. In *IEEE VIS*, 2018.

29. K. Sun, Z. Zhu, and Z. Lin. Enhancing the robustness of deep neural networks by boundary conditional gan. *arXiv:1902.11029*, 2019.
30. M. Sundararajan et al. Axiomatic attribution for dnn. In *ICML*. JMLR, 2017.
31. J. van der Waa et al. Contrastive explanations with local foil trees. *arXiv:1806.07470*, 2018.
32. J. Xie et al. Image denoising with deep neural networks. In *NIPS*, 2012.
33. M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.