Unjustified Classification Regions and Counterfactual Explanations In Machine Learning

Thibault Laugel[1] [✉], Marie-Jeanne Lesot[1], Christophe Marsala[1],
Xavier Renard[2], and Marcin Detyniecki[1,2,3]

[1] Sorbonne Université, CNRS, LIP6, F-75005 Paris, France
[2] AXA, Paris, France
[3] Polish Academy of Science, IBS PAN, Warsaw, Poland
thibault.laugel@lip6.fr

Abstract. Post-hoc interpretability approaches, although powerful tools to generate explanations for predictions made by a trained black-box model, have been shown to be vulnerable to issues caused by lack of robustness of the classifier. In particular, this paper focuses on the notion of explanation justification, defined as connectedness to ground-truth data, in the context of counterfactuals. In this work, we explore the extent of the risk of generating unjustified explanations. We propose an empirical study to assess the vulnerability of classifiers and show that the chosen learning algorithm heavily impacts the vulnerability of the model. Additionally, we show that state-of-the-art post-hoc counterfactual approaches can minimize the impact of this risk by generating less local explanations[1].

Keywords: machine learning interpretability · counterfactual explanations.

1 Introduction

The soaring number of machine learning applications has been fueling the need for reliant interpretability tools to explain the predictions of models without sacrificing their predictive accuracy. Among these, post-hoc interpretability approaches [12] are popular as they can be used for any classifier regardless of its training data (i.e. *blackbox* assumption). However, this has also been a common criticism for such approaches, as it also implies that there is no guarantee that the built explanations are faithful to ground-truth data.

A specific type of interpretability approach generate counterfactual explanations (e.g. [25,30,11,28]), inspired from counterfactual reasoning (e.g. [5]) which aims at answering the question: *What will be the consequence of performing this action?* Adapted to the context of machine learning classification, counterfactual explanations try to identify how altering an input observation can influence its

[1] Source code available at: https://github.com/thibaultlaugel/truce

prediction, and in particular change its predicted class. These approaches have been recently under the spotlight as they provide a user with directly actionable explanations that can be easily understood [30].

However, used in a post-hoc context, counterfactual explanation approaches are vulnerable to aforementioned issues, leading to explanations that may not be linked to any ground-truth data and therefore not satisfying in the context of interpretability. In particular, [21] argues that an important criterion for interpretability is that counterfactual explanations should be *justified*, meaning continuously connected to ground-truth instances from the same class. However, they show that this connectedness criterion is not guaranteed in the post-hoc context, leading to unconnected classification regions and hence potential issues interpretability-wise. This paper is an extension of this previous work, proposing to further study the apparition of these unconnected regions and analyze how existing counterfactual approaches can avoid generating unjustified explanations.

This paper proposes the following contributions:

- We pursue the analysis of [21] about the unconnectedness of classification regions and analyze the vulnerability of various classifiers. We show that classifiers are not equally vulnerable.
- We study the link between unconnectedness and overfitting and show that controlling overfitting helps reducing unconnectedness.
- We show that state-of-the-art post-hoc counterfactual approaches may generate justified explanations but at the expense of counterfactual proximity.

Section 2 is devoted to presenting the state-of-the-art of post-hoc interpretability and counterfactual explanations, as well as highlighting studies that are similar to this work. Section 3 recalls the motivations and definition and for ground-truth backed counterfactual explanations, while Section 4 describes the algorithms used for this analysis. The study itself, as well as the obtained results, are presented in Section 5.

2 Background

2.1 Post-hoc Interpretability

In order to build explanations for predictions made by a trained black-box model, post-hoc interpretability approaches generally rely on sampling instances and labelling them with the model to gather information about its behavior, either locally around a prediction [26] or globally [6]. These instances are then used to approximate the decision function of the black-box and build understandable explanations, using for instance a surrogate model (e.g. a linear model in [26] or a decision tree in [13]), or by computing meaningful coefficients (e.g. Shapeley values [24] or decision rules [29]). Other methods rely on specific instances to build explanations using comparison to relevant neighbors, such as case-based reasoning approaches [17] and counterfactual explanation approaches.

Instead of simply identifying important features, the latter aim at finding the minimal perturbation required to alter a given prediction. A counterfactual explanation is thus a specific data instance, close to the observation whose prediction is being explained, but predicted to belong to a different class. This form of explanation provides a user with tangible explanations that are directly understandable and actionable; this can be opposed to other types of explanations using feature importance vectors, which are arguably harder to use and to understand for a non-expert user [30]. Several formalizations of the counterfactual problem can be found in the literature, depending on the formulation of the minimization problem and on the used distance metric. For instance, [20] looks uniformly for the L_2-closest instance of an other class, while [11] uses a local decision tree trained on instances sampled with a genetic algorithm to find the local L_0-closest instance. Another formulation of the counterfactual problem can be found in [19], which uses search algorithms to find the instance that has the highest probability of belonging to another class within a certain maximum distance. Finally, the problem in [30], formulated as a tradeoff between the L_1 closest instance and a specific classification score target, is another way to tackle the counterfactual problem.

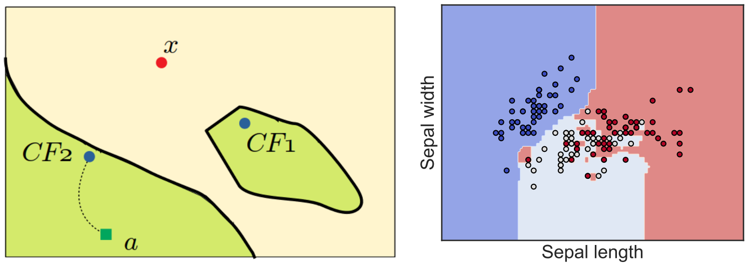
2.2 Studies of Post-hoc Interpretability Approaches

The post-hoc paradigm, and in particular the need for post-hoc approaches to use instances that were not used to train the model to build their explanations, raises questions about their relevance and usefulness. Troublesome issues have been identified: for instance, it has been noticed [2] that modeling the decision function of a black-box classifier with a surrogate model trained on generated instances can result in explanation vectors that point in the wrong directions in some areas of the feature space in trivial problems. The stability of post-hoc explainer systems has been criticized as well, showing that because of this sampling, some approaches are locally not stable enough [1] or on the contrary too stable and thus not locally accurate enough [22].

Recently, a few works [23,27] have started questioning the post-hoc paradigm itself and criticizing the risk of generating explanations that are disconnected from the ground truth, which is the main topic of this work. Identifying relevant ground-truth instances for a specific prediction has also been studied before (e.g. [16]), but generally require retraining the classifier, which is not possible in the context of a black-box. In a similar fashion, [15] try to identify which predictions can be trusted in regard to ground-truth instances based on their distance to training data in the context of new predictions.

2.3 Adversarial Examples

Although not sharing the same purpose, both counterfactual explanations and adversarial examples [3] are similar in the way they are generated, i.e. by trying to perturb an instance to alter its prediction. Studies in the context of image classification using deep neural networks [9] have shown that the connectedness



(a) Illustration of the idea behind connectedness

(b) Decision boundary of a RF classifier

Fig. 1. Left picture: illustration of the connectedness notion. The decision boundary learned by a classifier (illustrated by the yellow and green regions) has created two green regions. $CF1$ and $CF2$ are two candidate counterfactual explanations for x. $CF1$ can be connected to the training instance a by a continuous path that do not cross the decision boundary of f, while $CF1$ can not. Right picture: a random forest classifier with 200 trees has been trained on 80% of the dataset (a 2D version of the iris dataset) with 0.79 ± 0.01 accuracy over the test set. Because of its low robustness, unconnected classification regions can be observed (e.g. small red square in the light blue region).

notion studied in this paper was not enough for adversarial examples detection. However, since adversarial examples have been generally studied in the context of high-dimensional unstructured data, (i.e. images, text and sound for instance), as shown exhaustively in existing surveys (e.g. [4]), they remain out of the scope of this study.

The goal of this work is to study a desideratum for counterfactual explanations in the context of interpretability in structured data; it is not to detect a classification error nor an adversarial example.

3 Justification Using Ground-truth Data

This section recalls the definitions and intuitions behind the notion of *justification*, which is the main topic of this work.

3.1 Intuition and Definitions

The notion of ground-truth justification proposed in [21] aims at making a distinction between an explanation that has been generated because of some previous knowledge (such as training data) and one that would be a consequence of an artifact of the classifier. While generalization is a desirable property for a classifier for prediction, it is different in the context of interpretability, since an explanation that a user can not understand is useless.

For this purpose, an intuitive desideratum for counterfactual explanations can be formulated based on the distance between the explanation and ground-truth observations, defining a plausibility notion that is similar to the trust score proposed in [15]): to guarantee useful (plausible) explanations, a counterfactual example could be thus required to be close to existing instances from the same class. Detecting whether an explanation satisfies this criterion thus becomes similar to an outlier detection problem, where the goal is to have counterfactual explanations that do not lie out of the distribution of ground-truth instances.

However, discriminating counterfactual explanations based on their distance to ground-truth data does not seem to be good enough, as issues created by classification artifacts can arise in dense training regions as well. Such artifacts can be caused by a lack of robustness of the classifier (e.g. overfitting), or simply because it is forced to make a prediction in an area he does not have much information about. These issues may lead to classification regions that are close to but not supported by any ground-truth data, which is problematic in the context of interpretability.

In this context, a requirement for counterfactual explanations is proposed, where the relation expected between an explanation and ground-truth data is defined using the topological notion of path connectedness. In order to be more easily understood and employed by a user, it is argued that the counterfactual instance should be continuously connected to an observation from the training dataset. The idea of this *justification* is not to identify the instances from the training data that are *responsible* for the prediction of the counterfactual (such as in the aforementioned work of [16]), but the ones that are correctly being predicted to belong to the same class for similar reasons.

The following definition is thus introduced:

Definition 1 (Justification [21]). *Given a classifier $f : \mathcal{X} \rightarrow \mathcal{Y}$ trained on a dataset X, a counterfactual example $e \in \mathcal{X}$ is* justified *by an instance $a \in X$ correctly predicted if $f(e) = f(a)$ and if there exists a continuous path h between e and a such that no decision boundary of f is met.*

Formally, e is justified by $a \in X$ if: $\exists\ h : [0, 1] \rightarrow \mathcal{X}$ such that: (i) h is continuous, (ii) $h(0) = a$, (iii) $h(1) = e$ and (iv) $\forall t \in [0, 1]$, $f(h(t)) = f(e)$.

This notion can be adapted to datasets by approximating the function h with a high-density path, defining *ϵ-chainability*, with $\epsilon \in \mathbb{R}^+$: an *$\epsilon$-chain* between e and a is a finite sequence e_0, e_1, ... $e_N \in \mathcal{X}$ such that $e_0 = e$, $e_N = a$ and $\forall i < N$, $d(e_i, e_{i+1}) < \epsilon$, with d a distance metric.

Definition 2 (ϵ-justification [21]). *Given a classifier $f : \mathcal{X} \rightarrow \mathcal{Y}$ trained on a dataset X, a counterfactual example $e \in \mathcal{X}$ is* ϵ-justified *by an instance $a \in X$ accurately predicted if $f(e) = f(a)$ and if there exists an ϵ-chain $\{e_i\}_{i \leq N} \in \mathcal{X}^N$ between e and a such that $\forall i \leq N, f(e_i) = f(e)$.*

Consequently, a justified (resp. unjustified) counterfactual explanation (written JCF, resp. UCF) is a counterfactual example that does (resp. does not) satisfy Definition 2. Setting the value of ϵ, ideally as small as possible to guarantee a good approximation by the ϵ-chain of function h of Definition 1, depends on the considered problem (dataset and classifier) and can heavily influence the obtained results. A discussion about this problem is proposed in Section 5.

Since connectedness in unordered categorical data can not be properly defined, the notion of justification can not be directly applied to these domains. Hence, we restrict the analysis of this paper to numerical data.

Figure 1 illustrates both the idea behind the notion of connectedness and the issue it tries to tackle. The left picture illustrates an instance x whose prediction by a binary classifier is to be interpreted, as well as two potential counterfactual explanations, $CF1$ and $CF2$. $CF2$ can be connected to a ground-truth instance a without crossing the decision boundary of f and is therefore justified. On the contrary, $CF1$ is not since it lies in a classification region that does not contain any ground-truth instance from the same class (green ”pocket”). In the right picture, a classifier with low robustness creates classification regions that can not be connected to any instance from the training data.

3.2 Implementation

For an efficiency purpose, it is possible to draw a link between connectedness and density-based clustering. In particular, the well-known DBSCAN [8] approach uses the distance between observations to evaluate and detect variations in the density of the data. Two parameters ϵ and $minPts$ control the resulting clusters (resp. outliers), built from the core points, which are instances that have at least (resp. less than) $minPts$ neighbors in an ϵ-ball. Thus, having an instance being ϵ-justified by another is equivalent to having them both belong to the same DBSCAN cluster when setting the parameters $minPts = 2$ and same ϵ.

4 Procedures for Assessing the Risk of Unconnectedness

In this section, the two algorithms proposed by [21] and used in the experiments and discussion are described: LRA (*Local Risk Assessment*) is a diagnostic tool assessing the presence of unjustified regions in the neighborhood of an instance whose prediction is to be interpreted; VE (*Vulnerability Evaluation*) assesses whether or not a given counterfactual explanation is connected to ground-truth data.

4.1 LRA Procedure

This section recalls the LRA procedure, used to detect unjustified classification regions. Given a black-box classifier $f : \mathcal{X} \rightarrow \mathcal{Y}$ trained on the dataset X of instances of $\mathcal{X}$ and an instance $x \in \mathcal{X}$ whose prediction $f(x)$ is to be explained, the aim is to assess the risk of generating unjustified counterfactual examples in a local neighborhood of x. To do this, a generative approach is proposed that aims at finding which regions of this neighborhood are ϵ-connected to instances of X (i.e. verify Definition 2). The three main steps of the LRA procedure, described below, are written down in Algorithm 1.

Definition step. A local neighborhood of the instance x being examined is defined as the ball of center x and radius the distance between x and the closest ground-truth instance a_0 correctly predicted to belong to another class.

Initial assessment step. A high number n of instances are generated in the defined area and labelled using f. The instances predicted to belong to the class $f(a_0)$, as well as a_0, are clustered using DBSCAN (with parameters $minPts = 2$ and ϵ). The instances belonging to the same cluster as a_0 are identified as JCF.

Iteration step. If some instances do not belong to the same cluster as a_0, new instances are generated in the spherical layer between a_0 and a_1, the second closest ground-truth instance correctly predicted to belong to the same class. Along with the previously generated instances, a_0 and a_1, the subset of the new instances that are predicted to belong to the same class as a_0 and a_1 are clustered. Again, new JCF can be identified. If a cluster detected at a previous step remains the same (i.e. does not increase in size), it means that the procedure has detected an enclosed unconnected region (pocket) and the cluster is therefore labelled as UCF. This step is repeated when all instances of the initial assessment step are identified either as JCF or UCF.

In the end, the LRA procedure returns n_J (resp. n_U) the number of JCF (resp. UCF) originally generated in the local neighborhood defined in the Definition Step. If $n_U > 0$, there exists a risk of generating unjustified counterfactual examples in the studied area.

In particular, the following criteria are calculated:

$$S_x = \mathbb{1}_{n_U > 0} \quad \text{and} \quad R_x = \frac{n_U}{n_U + n_J} \, .$$

S_x labels the studied instance x as being vulnerable to the risk of having UCF in the area (i.e. having a non-null risk). The risk itself, measured by R_x, describes the likelihood of having an unjustified counterfactual example in the studied area when looking for counterfactual examples. As these scores are calculated for a specific instance x with a random generation component, their average values $\bar{S}$ and $\bar{R}$ for 10 runs of the procedure for each x and over multiple instances are calulated.

4.2 VE Procedure

A variation of the LRA procedure, *Vulnerability Evaluation* (VE), is proposed to analyze how troublesome unjustified regions are for counterfactual approaches.

The goal of the VE procedure is to assess the risk for state-of-the-art methods to generate UCF in regions where there is a risk. Given an instance $x \in X$, the LRA procedure is first used to calculate the risk R_x and focus on the instances where this risk is ”significant” (i.e. for instance where $R_x > 0.25$). Using a state-of-the-art method, a counterfactual explanation $E(x) \in \mathcal{X}$ is then generated. To check whether $E(x)$ is justified or not, a similar procedure as LRA is used: instances are generated uniformly in the local hyperball with center $E(x)$

Algorithm 1: LRA [21]

```
Require: $x$, $f$, $X$
Generate $n$ instances in the ball $\mathcal{B}(x, d(x, a_0))$
Label these instances using $f$, keep the subset $D$ of instances classified similarly
as $a_0$
Perform DBSCAN over $D \cup \{a_0\}$
Identify the instances of $D$ that belong to the same cluster as $a_0$ as JCF.
Calculate current $n_J$
while Some instances are neither JCF nor UCF do
    Generate new instances in the spherical layer defined by the next closest
    ground-truth instance
    Label these instances using $f$
    Apply DBSCAN to the subset of these instances classified as $f(a_0)$, along with
    previously generated instances and relevant ground-truth instances
    Update $n_J$ and $n_U$
end while
return $n_J$, $n_U$ ;
```

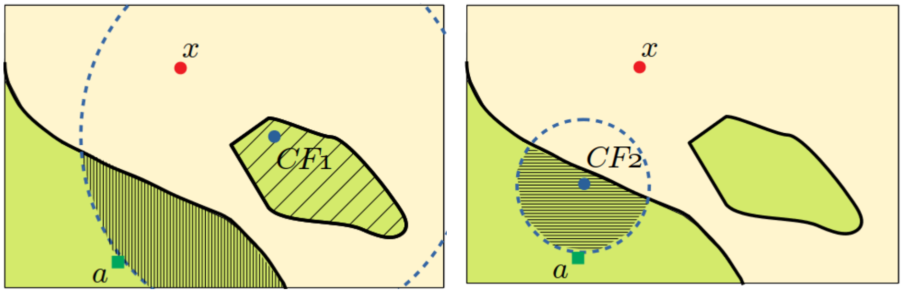


Fig. 2. Illustration of the VE procedure for two counterfactual explanation candidates. Left: CF1, which is not justified. Right: CF2, justified

and radius the distance between $E(x)$ and the closest ground-truth instance b_0 correctly predicted to belong to the same class as $E(x)$. the DBSCAN algorithm is then used with parameters $minPts = 2$ and same ϵ on the generated instances that are predicted to belong to $f(E(x))$. $E(x)$ is identified as a JCF if it belongs to the same DBCSCAN cluster as b_0 (cf. Definition 2).

If not, like with LRA, iteration steps are performed as many times as necessary by expanding the exploration area to b_1, the second closest ground-truth instance predicted to belong to $f(E(x))$.

An illustration of the procedure in a 2-dimensional binary classification setting is shown in Figure 2 for two counterfactual explanations $CF1$ and $CF2$ (blue dots), generated for the observation x (red dot). In the left picture, two clusters (hatched areas) are identified by DBSCAN in the explored area (blue dashed circle): $CF1$ and a, the closest training instance, do not belong to the same cluster, defining $CF1$ as unjustified. In the right picture, $CF2$ belongs to the same cluster as a and is therefore defined as justified.

In the end, the VE procedure returns a binary value indicating whether or not the analyzed counterfactual explanation $E(x)$ is justified, written $J_{E(x)}$ ($J_{E(x)} = 1$ if $E(x)$ is justified). Again, we also measure the average value $\bar{J}$ of $J_{E(x)}$ for multiple runs of the procedure for the instance x and over multiple instances.

5 Experimental Study: Assessing the Risk of Unjustified Regions

The two presented procedures LRA and VE can be used to analyze the unconnectedness of classification regions. As explained earlier, setting the values of n and ϵ is crucial for the experiments, as these parameters are central to the definition and procedures used. Therefore, a first experiment and discussion (cf Section 5.2) are conducted on this problem. Once adequate values are found, a second experiment (see Section 5.3) is performed, where we discuss how exposed different classifiers are to the threat unjustified classification regions. The link between this notion and overfitting is also studied. Finally, in a third experiment (cf. Section 5.4), the vulnerability of state-of-the-art post-hoc counterfactual approaches is analyzed, and we look into how they can minimize the problem of unjustified counterfactual explanations.

5.1 Experimental Protocol

In this section, we present the experimental protocol considered for our study.

Datasets. The datasets considered for these experiments include 2 low-dimensional datasets (half-moons and wine) as well as 4 real datasets (Boston Housing [14], German Credit [7], Online News Popularity [10] Propublica Recidivism [18]). These structured datasets present the advantage of naturally understandable features and are commonly used in the interpretability (and fairness) literature. All include less than 70 numerical attributes. As mentioned earlier, categorical features are excluded from the scope of the study.

Classifiers. For each considered dataset, the data is rescaled and a train-test split of the data is performed with 70%-30% proportion. Several binary classifiers are trained on each dataset: a random forest classifier, a support vector classifier with Gaussian kernel, an XGboost classifier, a Naive Bayes classifier and a k-nearest-neighbors classifier. Unless specified, the associated hyperparameters are chosen using a 5-fold cross validation to optimize accuracy. The AUC score values obtained on the test set with these classifiers are shown in Table 1. Several variations of the same classifier are also considered for one of the experiments (Section 5.3), with changes in the values of one of the associated

Dataset	RF	SVM	XGB	NB	KNN	1-NN
Half-moons	0.98 ± 0.01	0.99 ± 0.00	0.99 ± 0.00	0.95 ± 0.01	0.99 ± 0.00	0.95 ± 0.01
Wine	0.98 ± 0.01	0.99 ± 0.00	0.99 ± 0.00	0.95 ± 0.01	0.99 ± 0.00	0.95 ± 0.01
Boston	0.96 ± 0.02	0.97 ± 0.04	0.97 ± 0.03	0.87 ± 0.08	0.93 ± 0.06	0.85 ± 0.04
Credit	0.75 ± 0.05	0.64 ± 0.04	0.70 ± 0.08	0.66 ± 0.04	0.66 ± 0.02	0.55 ± 0.05
News	0.68 ± 0.02	0.68 ± 0.01	0.70 ± 0.02	0.65 ± 0.02	0.65 ± 0.02	0.55 ± 0.01
Recidivism	0.81 ± 0.01	0.82 ± 0.01	0.84 ± 0.01	0.78 ± 0.02	0.81 ± 0.02	0.68 ± 0.02

Table 1. AUC scores obtained on the test sets for a random forest (RF), support vector classifier (SVM), XGBoost (XGB), naive Bayes classifier (NB) k-nearest neighbors (KNN) and nearest neighbor (1-NN)

hyperparameters: the maximum depth allowed for each tree of the random forest algorithm, and the γ coefficient of the Gaussian kernel of the support vector classifier.

Protocol. The test set is also used to run the experiments, as described earlier. In Section 5.3, the LRA procedure is applied to each instance of the considered test sets, and the scores $\bar{S}$ and $\bar{R}$ are calculated and analyzed for each dataset and classifier. In Section 5.4, three post-hoc counterfactual approaches from the state-of-the-art (HCLS [19], GS [20] and LORE [11]) are used to generate explanations $E(x)$ for each instance of the considered test sets, and the VE procedure is applied to the obtained counterfactuals to calculate the scores $\bar{J}$, as well as the distance $d(x, E(x))$.

5.2 Defining the Problem Granularity: Choosing n and ϵ

As presented in [21] and mentioned above, the values of n and ϵ are crucial since they define the notion of ϵ-justification and impact the average distance between the generated instances. Choosing inadequate values for n and ϵ may lead to having some unconnected regions not being detected as such and vice versa.

These two values are obviously linked, as n defines, for a given $x \in X$, the density of the sampling in the initial assessment step of the LRA procedure, hence the average pairwise distance between the generated observations, and therefore the value ϵ should be taking. Identifying an adequate value for ϵ depends on the local topology of the decision boundary of the classifier, as well as the radius of the hyperball defined in LRA (cf. Section 4.1). In practice, because the instances B_x are generated in the initial assessment step before running DBSCAN, it is easier to set the value of ϵ to the maximum value of the distances of B to their closest neighbors: $\epsilon = \max\limits_{x_i \in B_x} \min\limits_{x_j \in B_x \setminus \{x_i\}} d(x_i, x_j)$. Using this value, the training instance a_0 is guaranteed to be in an actual cluster (i.e. not detected as an outlier), which is a desirable property of the approach: it is expected that since is a_0 correctly predicted, it should be possible to generate a close neighbor

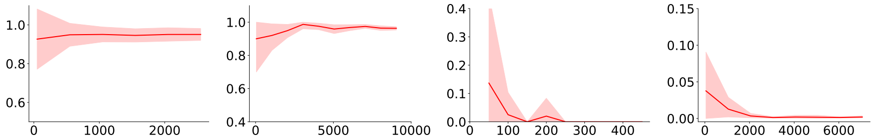


Fig. 3. Average R_x score for several instances of the half-moons dataset depending on the value of n. The first two instances are selected to verify $S_x = 1$, while the last two verify $S_x = 0$. After n reaches a certain value, R_x hardly changes anymore

classified similarly (in the same "pocket"). The problem thus becomes of setting the value of n alone.

In order to have the best performance, n should have the highest value as possible. However, this also increases dramatically the running time of the algorithm. Besides the complexity of the classifier's decision boundary, the value of n required to saturate the local space increases exponentially with the dimension of the problem. Furthermore, as the radius of generation increases during the iteration steps, the number of instances should also increase to guarantee constant space saturation across various steps. Instead, we choose to set a high initial value of n at the first step, leading to an even higher complexity.

In this context, we are interested in identifying a value for n that properly captures the complexity of the local decision boundary of the classifier without generating an unrequired amount of instances. We thus look at the value of R_x for several instances and several values of n to detect the threshold above which generating more instances does not change the output of the LRA procedure. Figure 3 illustrates this result for several instances of the half-moond dataset (two with $S_x = 1$ and two with $S_x = 0$: the R_x score reaches a plateau after a certain value of n. Using this assumption, the LRA procedure can be tested with various values of n to ensure a reasonable value is chosen for the results of the other experiments presented in the next section.

5.3 Detecting Unjustified Regions

While the existence and dangers of unjustified regions has been shown in [21], the extent to which classifiers are vulnerable to this issue remains unclear.

Comparing the Vulnerability of Classifiers In Table 2 are shown the proportion $\bar{S}$ of the studied instance that have unjustified classification regions in their neighborhood (LRA returning $S_x = 1$). Every classifier seems to be generating unjustified regions in the neighborhoods of test instances: in some cases, as much as 93% of the tested instances are concerned (XGB classifier trained on the German Credit dataset).

However, the extent to which each classifier is vulnerable greatly varies. For instance, among the considered classifiers, the random forest and XGBoost algorithms seem to be more exposed than other classifiers (average $\bar{S}$ value across

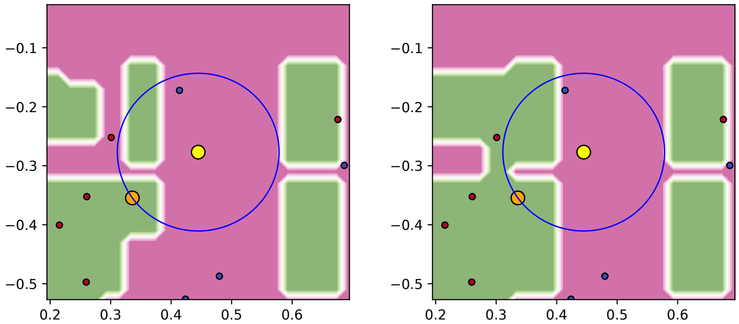


Fig. 4. Illustration of the LRA procedure applied to an instance of the half-moons dataset. Left: RF with no maximum depth precised. Right: maximum depth allowed is 10

dataset resp. 0.63 and 0.54, vs. 0.39 for the SVM for instance). The learning algorithm, and thus the associated complexity of the learned decision boundary, heavily influences the creation of classification regions. A link with predictive accuracy can thus be expected. This can be also observed in the results of the Naive Bayes classifier: while this classifier seems to be the more robust to the studied problem (average value of $\bar{S}$ across all datasets equals 0.29), it should be noted that it is also the classifier that performs the worst in terms of prediction (besides 1-NN, cf. Table 1).

These results are further confirmed by the values of $\bar{R}$ shown in Table 4, which also give an indication of the relative size of these unjustified classification regions: for instance, despite having similar values for $\bar{S}$ on the German Credit dataset, RF and XGboost have a higher $\bar{R}$ values than KNN, indicating that the formed unconnected regions are wider in average.

Differences in results can also be observed between datasets, since more complex datasets (e.g. less separable classes, higher dimension...) may also lead to classifiers learning more complex decision boundaries (when possible), and maybe favoring overfitting. This phenomenom is further studied in the next experiment.

Link Between Justification and Overfitting To further study the relation between the creation of unjustified regions and the learning algorithm of the classifier, we analyze the influence of overfitting over the considered metrics. For this purpose, we attempt to control overfitting by changing the values of the hyperparameters of two classifiers: the maximum depth allowed for a tree for RF, and the γ parameter of the Gaussian kernel for SVM. For illustration purposes, we apply the LRA procedure in a two-dimensional setting (half-moons) to a classifier deliberately chosen for its low robustness (a random forest with only 3 trees). Figure 4 shows a zoomed in area of the decision boundary of the

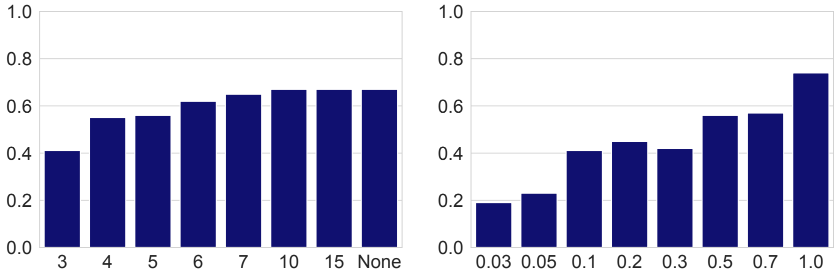


Fig. 5. $\bar{R}$ scores for RF (left) and SVM (right) classifiers on the Boston dataset for various values of hyperparameters (resp. maximum number of trees and γ parameter of the Gaussian kernel). "None" (left) means no maximum tree depth restriction is set

classifier (colored areas, the green and purple dots represent training instances), as well as the result of LRA for a specific instance x (yellow instance). In the left figure, the considered classifier has no limitation on the depth of the trees it can use, wereas in the right one this parameter is set to 10. As explained earlier, LRA explores the local neighborhood of x (blue circle), delimited by its closest neighbor from the training set correctly classified a_0 (orange instance). In the left figure, within this neighborhood, a green square region is detected as an unjustified region (top left from x): there is no green instance in this region, hence $S_x = 1$. However, in the right picture, this region is connected to green instances: $S_x = 0$.

Quantitative results of this phenomenom are shown in Figure 5, which illustrates the evolution of $\bar{S}$ and $\bar{R}$ scores for the two mentioned classifiers (left: RF; right: SVM). As expected, the more overfitting is allowed (i.e. when the maximum tree depth of RF and when the γ parameter of the RBF kernel of SVM increase), and the more prone to generate unjustified regions these two classifiers seem.

However, it should be noted that models such as logistic regression or 1-nearest neighbor (not appearing in Tables 2 and 3) have, by construction, no UCF ($\bar{S} = 0.0$): a logistic regression creates only two connected classification regions, and the predictions of a 1-NN classifier are by construction connected to their closest neighbor from the training data, despite this classifier being frequently referred to as an example of overfitting. Therefore, the notion of overfitting is not sufficient to describe the phenomenom of unconnectedness.

The tradeoff there seems to be between justification and accuracy (cf. Tables 1 and 2) seems to indicate that a lower complexity of the decision border favors better justification scores for at the cost of predictive performance (cf. Table 1 for the comparatively lower predictive performance of the 1-NN classifier).

Dataset	RF	SVM	XGB	NB	KNN
Half-moons	0.37	0.00	0.05	0.00	0.02
Wine	0.21	0.08	0.15	0.08	0.15
Boston	0.63	0.29	0.62	0.44	0.25
Credit	0.93	0.76	0.93	0.27	0.92
News	0.85	0.72	0.86	0.57	0.68
Recidivism	0.81	0.50	0.61	0.36	0.73

Table 2. Proportion of instances being at risk of generating a UCF ($\bar{S}$ score) over the test sets for 6 datasets

Dataset	RF	SVM	XGB	NB	KNN
Half-moons	0.07 (0.17)	0.00 (0.00)	0.01 (0.02)	0.0 (0.0)	0.00 (0.00)
Wine	0.01 (0.02)	0.02 (0.07)	0.00 (0.01)	0.01 (0.02)	0.01 (0.01)
Boston	0.16 (0.25)	0.06 (0.13)	0.14 (0.24)	0.07 (0.14)	0.03 (0.05)
Credit	0.44 (0.37)	0.10 (0.14)	0.45 (0.37)	0.06 (0.17)	0.31 (0.27)
News	0.35 (0.28)	0.18 (0.28)	0.33 (0.30)	0.12 (0.24)	0.37 (0.38)
Recidivism	0.26 (0.30)	0.14 (0.21)	0.21 (0.28)	0.08 (0.20)	0.20 (0.30)

Table 3. Average risk of generating an UCF ($\bar{R}$) and standard deviations for 6 datasets

5.4 Vulnerability of Post-hoc Counterfactual Approaches

In the post-hoc context, because no assumption is made about the classifier nor any training data, counterfactual approaches have been shown to be subject to generating unjustified explanations [21]. The state-of-the-art approaches mentioned in Section 5.1 are applied to the considered datasets in order to assess the extent of this risk.

The VE procedure is applied to the counterfactual explanations generated using state-of-the-art approaches for instances facing a significant justification risk (constraint arbitrarily set to $R_x \geq 0.25$) of the previously considered datasets, on which a random forest classifier was trained. The results ($\bar{J}$ score) are shown in Table 4. In addition to J_x, the distance between each tested instance and its generated counterfactual explanations are calculated, and represented in the table by their average value $\bar{d}$.

As expected, every considered counterfactual approach seems to be generating to some extent unjustified explanations: the Justification scores of the tested approaches can even fall as low as 30% (GS on the Online News Popularity dataset). However, some differences can be observed between the approaches: for instance, HCLS and LORE seem to achieve better performance than GS in terms of justification across all datsets (average $\bar{J}$ across datasets equals resp. 0.74 and 0.91 for HCLS and LORE, against only 0.62 for GS). However, we observe that the average distance $\bar{d}$ is also higher (resp. 1.38 and 1.46 for HCLS and LORE, against 0.90 for GS). This can be explained by the fact that GS directly minimizes a L_2 distance (the considered d distance), while LORE minimizes a

Dataset	**HCLS**		**GS**		**LORE**	
	$\bar{J}$	$\bar{d}$	$\bar{J}$	$\bar{d}$	$\bar{J}$	$\bar{d}$
Half-moons	0.83	0.45 (0.27)	0.67	0.48 (0.26)	0.83	1.19 (0.18)
Boston	0.86	1.99 (0.88)	0.84	0.84 (1.03)	1.0	1.58 (0.98)
Credit	0.65	1.78 (0.94)	0.59	0.82 (0.71)	1.0	1.57 (1.11)
News	0.46	1.81 (0.75)	0.30	1.68 (0.99)	0.77	1.74 (0.83)
Recidivism	0.91	0.89 (1.08)	0.70	0.70 (1.09)	0.98	1.23 (0.90)

Table 4. Proportion of generated counterfactuals that are justified ($\bar{J}$) for vulnerable instances ($R_x \geq 0.25$)

L_0 distance in a local neighborhood. By looking for counterfactuals in the direct proximity of x, GS thus tend to find unjustified regions more easily than the other approaches, whereas looking further away from the decision boundary probably enables LORE to favor explanations located closer to ground-truth instances, therefore more frequently justified.

Another observation is that despite achieving better performance than GS by trying to maximize the classification probability of the generated counterfactual, HCLS still comes short in terms of justification. This tend to illustrate that classification confidence, when available, is not a good way to detect unconnected classification regions and guarantee justified explanations, some unconnected regions probably having high classification confidence.

These results highlight that while classification confidence, when available, does not seem to help in generating justified explanations, there seems to be a tradeoff with the counterfactual distance, as LORE achieves in some cases perfect justification scores (e.g. Boston and Credit datasets).

6 Conclusion

The justification constraint that is studied in this work comes from an intuitive requirement explanations for machine learning predictions should satisfy, as well as from the assumption that post-hoc counterfactual explanations are not able to distinguish UCF from JCF. Results highlight that this vulnerability greatly depends on the nature of the classifier, and that all learning algorithms are not equally likely to form unconnected classification regions. In particular, controlling overfitting seems to be very important for some of the studied classifiers. In light of this study, generating justified counterfactual explanations in the post-hoc context seems complicated and using the training instances, when available, is necessary. To reduce the impact of these issues, state-of-the-art approaches may look for explanations located further away from the decision boundary. However, this raises the question of explanation locality, as explanations located far away from the decision boundary may be less tailored for each instance, and thus less useful in the case of counterfactuals.

Despite the importance of justification, the question of whether this requirement is sufficient to guarantee useful explanations remains. In particular, questions arise when a counterfactual explanation lies in a justified region where the associated ground-truth instances are far away (e.g. out of distribution). In this context, adding a distance constraint (as discussed in Section 3) to ensure plausible justified explanations may constitute an interesting lead for future works.

Extending these notions to high-dimensional data (e.g. images) however needs further research, as neither connectedness nor distance helps in guaranteeing useful explanations. A good example of this is adversarial examples, which are defined as being close to original observations and have been proven to be connected to ground-truth instances in the case of deep neural networks but do not constitute satisfying counterfactual explanations.

Finally, results about the link between justification and overfitting raise the question of the accuracy of the classifier in these unconnected regions. In this regard, future works include analyzing the connectedness of classification errors.

Aknowledgements

This work has been done as part of the Joint Research Initiative (JRI) project "Interpretability for human-friendly machine learning models" funded by the AXA Research Fund.

References

1. Alvarez Melis, D., Jaakkola, T.: Towards robust interpretability with self-explaining neural networks. Advances in Neural Information Processing Systems 31 pp. 7786–7795 (2018)
2. Baehrens, D., Schroeter, T., Harmeling, S., Hansen, K., Muller, K.R.: How to Explain Individual Classification Decisions Motoaki Kawanabe. Journal of Machine Learning Research **11**, 1803–1831 (2010)
3. Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., Roli, F.: Evasion attacks against machine learning at test time. In: Machine Learning and Knowledge Discovery in Databases. pp. 387–402 (2013)
4. Biggio, B., Roli, F.: Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition **84**, 317–331 (2018)
5. Bottou, L., Peters, J., Quiñonero Candela, J., Charles, D.X., Chickering, D.M., Portugaly, E., Ray, D., Simard, P., Snelson, E.: Counterfactual reasoning and learning systems: The example of computational advertising. Journal of Machine Learning Research **14**, 3207–3260 (2013)
6. Craven, M.W., Shavlik, J.W.: Extracting tree-structured representations of trained neural networks. Advances in Neural Information Processing Systems **8**, 24–30 (1996)
7. Dua, D., Graff, C.: UCI machine learning repository (2017)
8. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proc. of the 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD'96). pp. 226–231 (1996)
9. Fawzi, A., Moosavi-Dezfooli, S.M., Frossard, P., Soatto, S.: Empirical study of the topology and geometry of deep networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)

10. Fernandes, K., Vinagre, P., Cortez, P.: A proactive intelligent decision support system for predicting the popularity of online news. Proc. of the 17th EPIA 2015 - Portuguese Conference on Artificial Inteligence.
11. Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., Giannotti, F.: Local rule-based explanations of black box decision systems. arXiv preprint 1805.10820 (2018)
12. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D.: A survey of methods for explaining black box models. ACM Computing Surveys (CSUR) **51**(5), 93 (2018)
13. Hara, S., Hayashi, K.: Making tree ensembles interpretable. ICML Workshop on Human Interpretability in Machine Learning (WHI 2016) (2016)
14. Harrison, D., Rubinfeld, D.: Hedonic prices and the demand for clean air. Environment Economics and Management **5**, 81–102 (1978)
15. Jiang, H., Kim, B., Guan, M., Gupta, M.: To trust or not to trust a classifier. In: Advances in Neural Information Processing Systems 31. pp. 5541–5552 (2018)
16. Kabra, M., Robie, A., Branson, K.: Understanding classifier errors by examining influential neighbors. 2015 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) pp. 3917–3925 (2015)
17. Kim, B., Rudin, C., Shah, J.A.: The Bayesian Case Model: A generative approach for case-based reasoning and prototype classification. In: Advances in Neural Information Processing Systems. pp. 1952–1960 (2014)
18. Larson, J., Mattu, S., Kirchner, L., Angwin, J.: How we analyzed the compas recidivism algorithm. ProPublica (May 2016)
19. Lash, M., Lin, Q., Street, N., Robinson, J., Ohlmann, J.: Generalized inverse classification. In: Proc. of the 2017 SIAM Int. Conf. on Data Mining. pp. 162–170 (2017)
20. Laugel, T., Lesot, M.J., Marsala, C., Renard, X., Detyniecki, M.: Comparison-based inverse classification for interpretability in machine learning. In: Information Processing and Management of Uncertainty in Knowledge-Based Systems. pp. 100–111 (2018)
21. Laugel, T., Lesot, M.J., Marsala, C., Renard, X., Detyniecki, M.: The dangers of post-hoc interpretability: Unjustified counterfactual explanations. Proc of the 28th Int. Joint Conf. on Artificial Intelligence IJCAI-19 (to appear) (2019)
22. Laugel, T., Renard, X., Lesot, M.J., Marsala, C., Detyniecki, M.: Defining locality for surrogates in post-hoc interpretablity. ICML Workshop on Human Interpretability in Machine Learning (WHI 2018) (2018)
23. Lipton, Z.C.: The mythos of model interpretability. ICML Workshop on Human Interpretability in Machine Learning (WHI 2017) (2017)
24. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems 30. pp. 4765–4774 (2017)
25. Martens, D., Provost, F.: Explaining data-driven document classifications. MIS Q. **38**(1), 73–100 (2014)
26. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should I trust you?": Explaining the predictions of any classifier. In: Proc. of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. pp. 1135–1144. KDD '16 (2016)
27. Rudin, C.: Please stop explaining black box models for high stakes decisions. NeurIPS Workshop on Critiquing and Correcting Trends in Machine Learning (2018)
28. Russell, C.: Efficient search for diverse coherent explanations. In: Proc. of the Conf. on Fairness, Accountability, and Transparency (FAT* '19. pp. 20–28 (2019)

29. Turner, R.: A model explanation system. NIPS Workshop on Black Box Learning and Inference (2015)
30. Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box; automated decisions and the GDPR. Harvard Journal of Law & Technology **31(2)**, 841–887 (2018)