Matching the expert's knowledge via a counterfactual-based feature importance measure *

Antonio Luca Alfeo[1,2], Mario G.C.A. Cimino[1,2], and Guido Gagliardi[1,3,4]

[1] Dept. of Information Engineering, University of Pisa, Pisa, Italy
[2] Research Center E. Piaggio, University of Pisa, Pisa, Italy
[3] Dept. of Information Engineering, University of Florence, Florence, Italy
[4] Dept. of Electrical Engineering, KU Leuven, Leuven, Belgium
{luca.alfeo, mario.cimino}@unipi.it, guido.gagliardi@phd.unipi.it

Abstract. To be employed in real-world applications, explainable artificial intelligence (XAI) techniques need to provide explanations that are comprehensible to experts and decision-makers with no machine learning (ML) background, thus allowing for the validation of the ML model via their domain knowledge.
To this aim, XAI approaches based on feature importance and counterfactuals can be employed, although both have some limitations: the last provide only local explanations, whereas the first can be very computationally expensive. A less computationally-expensive global feature importance measure can be derived by considering the instances close to the model decision boundary and analyzing how often some minor changes in one feature's values do affect the classification outcome.
However, the validation of XAI techniques in the literature rarely employs the application domain knowledge due to the burden of formalizing it, e.g., providing some degree of expected importance for each feature. Still, given an ML model, it is difficult to determine whether an XAI technique may inject a bias in the explanation (e.g., overestimating or underestimating the importance of a feature) in the absence of such ground truth.
To address this issue, we test our feature importance approach both with the UCI benchmark datasets and real-world smart manufacturing data characterized by annotations provided by domain experts about the expected importance of each feature. If compared to the state-of-the-art, the employed approach results to be reliable and convenient in terms of computation time, as well as more concordant with the expected importance provided by the domain expert.

* Work partially supported by (i) the company Koerber Tissue in the project "Data-driven and Artificial Intelligence approaches for Industry 4.0"; and the Italian Ministry of University and Research (MUR) in the frameworks: (ii) PNRR - M4C2 - Investimento 1.3, Partenariato Esteso PE00000013 - "FAIR" - Spoke 1 "Human-centered AI"; (iii) National Center for Sustainable Mobility MOST/Spoke10; and (iv) of the FoReLab project (Departments of Excellence). The authors thank Michelangelo Martorana for his work on the subject during his master's thesis.

Keywords: eXplainable Artificial Intelligence · Expert-based validation · Feature Importance · Counterfactual explanation.

1 Introduction and Motivations

Recent artificial intelligence (AI) approaches can provide unprecedented recognition performances [8]. However, since AI approaches can work as a black box, domain experts cannot easily validate and trust their outcomes [7]. This is especially important in real-world scenarios, such as in smart manufacturing [6]. Indeed, the adoption of AI technology can provide improved productivity [4] if AI outcomes can be trusted enough to be integrated into decision-making processes [21]. Explainable Artificial Intelligence (XAI) approaches [22] can address this issue by providing some explanations for the AI outcomes [5]. Using XAI approaches for smart manufacturing applications can result in reduced production costs, classification error mitigation, and improved AI-based system debugging [3].

In this context, choosing the most suitable explanation form is an application-dependent design choice. Depending on the application and its end-users, the interpretability of the explanations may be prioritized over their faithfulness [29]. According to [29] an explanation can be considered *interpretable* if *(i)* it is not ambiguous, *(ii)* similar instances correspond to similar explanations, and *(iii)* it can be presented in a compact form. Also, an explanation can be considered *faithful* if it *(i)* describes the AI model comprehensively and correctly, and *(ii)* provides some degree of knowledge about the decision process embedded in the AI model. Typically, interpretable explanations are not faithful, and vice versa [29]. To be understandable by domain experts and decision-makers with no AI background, the explanation needs to be as interpretable as possible. To this aim, the explanation forms that can be employed are the attribution-based (e.g., feature importance) and instance-based (e.g., based on counterfactuals) [29]. *Counterfactual explanations* expose similarities and differences between an instance classified by the AI model and similar instances from a different class [16]. *Feature importance measures* evaluate the importance of parts of the input (e.g. features for tabular data) for a given classification [2]. However, both these approaches have their limitations. On one hand, counterfactual explanations can only explain a single result rather than the whole model. On the other hand, feature importance measures can be very computationally expensive. The shortcomings of feature importance e counterfactual-based XAI approaches can be addressed by combining those approaches to provide novel, model-agnostic, and robust XAI approaches [9].

A common challenge with newly proposed XAI methods is their validation. Indeed, it is well known that the assumptions underlying the correctness of explanations may not be verified in any real-world scenario. For example, in the presence of correlation and codependence between features some not measures of feature importance become unreliable [28]. In this case, it is not possible to know whether the XAI method is overestimating or underestimating the importance

of a feature, since it is typically very complex to have an *a priori* quantitative assessment of the informativeness of a specific feature for a classification problem [12], e.g., through the domain knowledge [10]. To address this issue, some approaches in the literature propose the synthetic generation of (i) datasets with known relative feature importance [13], [40], or (ii) ground truth explanation via transparent classifiers for which the reason for the decision taken is available by design [18]. The problem with these approaches is that any synthetic generation procedure could inject some bias in the generated data or explanations, thus preventing a reliable and effective evaluation of the XAI method.

In this paper, this issue is addressed by employing a real-world dataset in which the expected importance of each feature in the data is provided by domain experts and used to validate a measure of feature importance that is model-agnostic, global, and counterfactual-based. Moreover, this measure is compared against other feature importance measures from the state of the art with a number of benchmark datasets.

The paper is structured as follows. In Section 2 the background and related works are presented. The employed approach is detailed in Section 3. The case studies and the experimental setup are presented in Section 4. Section 5 discuss the obtained results, whereas Section 6 outlines the conclusions.

2 Related Works

Post-hoc feature importance measures such as Permutation Importance [1] and Shapley additive explanations (SHAP, [27]) are among the most widely used post-hoc explanation tools. Those approaches assess the relevance of the input features on the AI model's classification outcome. Permutation Importance measures the importance of each feature for a trained AI model by randomly permuting the rows of one feature and evaluating the effect on the final classification performance. This process breaks the relationship between a feature and the target class and the resulting decrease in performance indicates the extent to which the model relies on the permuted feature. Instead, the feature importance provided via SHAP evaluates the importance of a feature for the classification by measuring the average marginal contribution of that feature across all the possible subsets of features [32]. Due to its wide applicability and solid theoretical background, the SHAP framework can be considered a gold standard among the feature importance approaches [30]. At the same time, the extensive use of SHAP has exposed its main limitations, among which, there is its computational cost. Indeed, SHAP's time complexity grows exponentially with the number of features and linearly with the number of samples in the data [24]. This issue is not specific to SHAP only. Indeed, most feature importance measures tend to be computationally expensive [20].

In this regard, counterfactual explanations can result in reduced computational costs. Intuitively, given a data instance i and its predicted class, a counterfactual is an instance c 'similar' to i that has been recognized as a different class. A counterfactual explanation corresponds to finding that 'similar' instance

and understanding the minimal change needed to change the classification outcome. In the literature there is no agreement on the definition of such a 'minimal change', e.g. it can be the minimum number of features to change or the minimum distance between the original instance and the counterfactual instances [19]. A counterfactual explanation can be found by *(i)* adopting a heuristic search strategy, e.g., searching within a reference population of instances to be used as counterfactuals [36].; *(ii)* framing the search as an optimization problem where both loss and constraints are modified [31], [15]; *(iii)* or using a "brute force procedure", i.e. specifying the step size and the ranges of values for each feature to be explored around the instance being explained [35]. According to the results in [19], the heuristic search strategies based on K-Nearest Neighbour procedures can result in the smallest computational cost as compared to other ones [36]. Despite the chosen search strategy, the main limitation of counterfactual explanations is that, by being local, they do not provide any insights about the AI model reasoning as a whole [19], [34].

To overcome the shortcomings of both feature importance e counterfactual-based XAI approaches, an increasing number of research works are proposing novel strategies based on the combination of feature importance and counterfactual explanations. For instance, in [39] the authors attempt to generate counterfactuals by modifying the value of the most important features measured via SHAP. An approach based on probabilistic contrasting counterfactuals is proposed in [17] to generate global and local explanations. However, by being causality-based this approach requires structured knowledge, i.e. causal graph, and does not provide an actual feature importance measure. Authors in [38] locally approximate the model's decision boundary by using a variational autoencoder to generate counterfactuals in the neighborhood of an instance to be explained. Similarly, authors in [33] provide local decision rules that are consistent with the decision boundary, whereas in [25] the authors generate instances in a hypersphere build around the sample to explain in order to approximate the decision boundary, which is not feasible for datasets characterized by a great number of features. The feature importance measure employed in this study, i.e. BoCSoR, belongs to a novel thread in the XAI literature, in which different explanation strategies are combined to address their limitations [23].

3 Design

We employ the Boundary Crossing Solo Ratio (BoCSoR), a global feature importance measure obtained by aggregating local counterfactual explanations [9]. The main idea behind BoCSoR is assessing the importance of one feature by considering the frequency with which the samples close to the model's decision boundary result in a different classification outcome if the value of that specific feature is substituted with the one of the corresponding counterfactual sample.

To identify the samples close to the decision boundary, we consider the samples with the smaller inter-class distance, i.e. the Euclidean distance separating an instance from its closest instance of the counterfactual class. Indeed, the

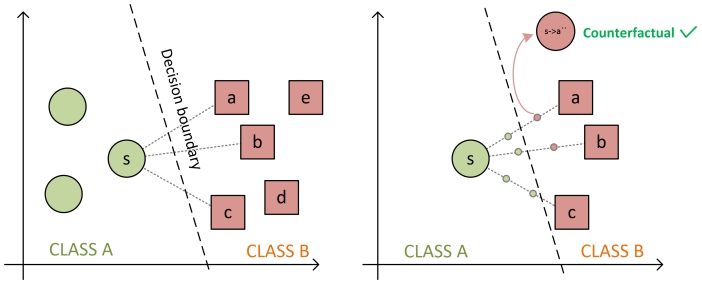

Fig. 1. The approach to finding the "closest" counterfactual (i.e., s$-> a''$) from class A to class B.

method "select" at line 6 of Algorithm 1 picks the instances whose inter-class distance falls below a specified percentile (*percentileTh* in Algorithm 1) among all inter-class distances.

Given a sample close to the decision boundary (s in Fig. 1), a counterfactual can be found by considering its K-Nearest-Neighbours of the counterfactual class (a, b and c in Fig. 1). The closest instance of a different class, however, could not correspond to the smallest change required to get a different classification. Midpoints are created between each possible counterfactual and the original instance in order to address this issue. Thus, via a step-wise exploration along the segments between the query sample and the neighbors, the closest midpoint corresponding to a different classification outcome can be considered as the closest counterfactual for the sample s (Fig. 1). The above-reported operations are provided via the method "findCF" in Algorithm 1 at line 8, which provides the closest counterfactual obtained via this step-wise exploration of the space spanning the decision boundary in the proximity of one sample.

"Closest" here is intended as relative to the step-wise exploration of the space between the two samples of different classes, rather than in an absolute sense as the minimum distance to change the classification outcome. Although this choice drops the guarantee of absolute minimal distance corresponding to the change in the classification outcome, it results in much lower computational costs and is therefore favored. This counterfactual search can be adjusted via two parameters: k and *steps* (Algorithm 1). k is the considered number of K-Nearest-Neighbours of the counterfactual class, *steps* is the number of midpoints (smaller circles in Fig. 1) along the segments between the query sample and its neighbors of the counterfactual class. We stress that the plausibility of the counterfactuals generated via linear interpolation is beyond the scope of the present study. In fact, the counterfactuals thus generated are not returned by our approach as the final explanation term. Instead, these are considered minimal perturbations that are relevant for the classification outcome, and thus useful for explaining the internal logic of the algorithm in terms of the sensitivity of the decision boundary to a change in a specific feature's value [38].

Given the closest counterfactual for one sample, it is possible to test which features alone can result in the crossing of the AI model's decision boundary, i.e. in a different classification outcome. Thus, starting from the counterfactual, we replace (one at a time) the feature values with those of the original sample (method "changeFeatureValue" in Algorithm 1, line 10). If this substitution corresponds to the crossing of the decision boundary, that feature is considered relevant (Algorithm 1 at line 11-13).

By taking into account all the samples close to the decision boundary, the times in which a feature is considered relevant (provided via the method "frequenceByFeature" in Algorithm 1, line 16) can be used as a proxy to evaluate the importance of each feature to distinguish the classes [38]. Finally, BoCSoR evaluates the importance of one feature by considering the frequency with which the changes of that feature alone do result in the crossing of the model's decision boundary, by considering the samples close to it. Algorithm 1 shows a high-level pseudo-code of the above-described procedure.

Algorithm 1 Procedure to measure the feature importance (i.e., $BoCSoR$).

Requires:
$M \Leftarrow$ trained machine learning model
$I \Leftarrow$ set of all the data instances
$F \Leftarrow$ set of all the features in the data
$class_o \Leftarrow$ original class
$class_c \Leftarrow$ counterfactual class
$percentileTh \Leftarrow$ threshold of the data
$k \Leftarrow$ # closest neighbours of s from $class_c$
$steps \Leftarrow$ # intermediate steps between s and its neighbours
Procedure:

```
1: relevantFeatures ⇐ emptyList()
2: set_O ⇐ select(I, label == class_o)
3: set_C ⇐ select(I, label == class_c)
4: pairwiseDist ⇐ computeDistance(set_O, set_C)
5: th ⇐ percentile(pairwiseDist, percentileTh)
6: instancesToQuery ⇐ select(set_O, pairwiseDist < th)
7: for each i ∈ instancesToQuery do
8:     closestCF ⇐ findCF(M, I, i, k, steps, class_o, class_c)
9:     for each f ∈ F do
10:        CF_tmp ⇐ changeFeatureValue(closestCF, i, f)
11:        if M.predict(CF_tmp) == class_o then
12:            relevantFeatures.append(f)
13:        end if
14:    end for
15: end for
16: featureImportance ⇐ frequenceByFeature(relevantFeatures)
17: return featureImportance
```

As shown in [9], BoCSoR results in a time complexity characterized by a linear growth with respect to the number of features (F) and a quadratic growth with respect to the number of samples (N), i.e. $O(N^2 + N \cdot F \cdot log(N))$. This is a smaller time complexity if compared to SHAP's one, which is characterized by linear growth with respect to the number of samples and exponential growth with respect to the number of features. If compared to other approaches able to combine feature importance and counterfactual explanations, BoCSoR (i) provides global feature importance (in contrast with [38], [25], and [33]), (ii) is characterized by a computational cost that scales linearly with the number of features (conversely to [25] and [27]), and (iii) does not require predetermined structured knowledge (as required by [17]).

4 Experimental datasets

In this section, the experimental datasets are described. A comparison with other feature importance approaches is obtained using five benchmark datasets collected from the well-known and publicly available UCI repository. Those datasets differ in terms of the number of features, classes, and instances. Specifically, the pen-based, satimage, segment, letter, and zoo datasets are employed [14]. The main characteristics of these data sets are summarized in Table 1.

Table 1. Characteristics of the benchmark datasets employed for this study.

Dataset	Penbased	Satimage	Segment	Letter	Zoo
#instances	10992	6435	2310	20000	101
#features	16	36	19	16	16
#classes	10	6	7	26	7

Moreover, a real-world dataset is employed in this study. Such data is provided by Koerber Tissue, a company that produces industrial machines to manufacture tissue paper. Each machine consists of two principal components: the embosser and the rewinder. The reels of raw paper layers are unwound and stacked by the rewinder and then passed to the embosser. Both rubber and steel rolls are used by the embosser to press and glue the tissue layers while imprinting a design on the paper. Each machine is tested with a variety of paper types and production settings, such as the rewinder speed or embossing pressure. For each production setting, some measurements are taken on the finished product. The final product's quality-related characteristics, like paper bulk and resistance, are included in these measurements. These characteristics can be described via levels (i.e., high, medium, low).

Like many real-world datasets, the company's data are characterized by a significant amount of missing values. The data is preprocessed to address this issue. First, all of the columns and rows with more than 50% missing values are removed. Then, the data instances are clustered considering the categorical features that do not present missing values. For each feature, the numerical missing

value of one data instance is replaced with the median (mode if categorical) of its cluster.

The resulting datasets consist of more than 650 instances. The dataset to recognize the paper resistance levels consists of 17 features, whereas the dataset for recognizing bulk levels consists of 15 features. Specifically, the data consists of the following features: (i) a unique identifier for each test measurement (ID), which is not considered an informative feature and thus it is removed from the analysis; (ii) the strength of the raw paper in the latitudinal (STRLA) and longitudinal direction (STRLO); (iii) the percentage of elongation of the raw paper in the latitudinal (ELOLA) and longitudinal direction (ELOLO); the weight of the raw paper (WEIGHT); (iv) the thickness of the raw paper (THICK); (v) the hardness of the rubber top (TRH) and bottom (BRH) roll used to imprint a motif on the paper, and measured in Shore A; (vi) a unique identifier for the process aimed at coupling different tissue layers (COUPL), which can be "molded" (M), "unmolded" (UM), or "glued embossing" (GE); (vii) a unique identifier for the embosser model; (viii) a unique identifier for the rewinder (REW) and embosser (EMB) model; (ix) a unique identifier of the motif characterizing the embosser top (ETR) and bottom (EBR) roll; (x) the type of product being manufactured (TYPRO); (xi) the number of tissue layers in the final product (LAYERS); (xii) the ratio of the raw paper resistance in the longitudinal and latitudinal directions, if dry (DRYRAT); and (xiii) a boolean indicating whether the raw paper is regular or structured (STRCT).

In order to obtain the ground truth of feature importance we gathered both the experts of the tissue production process and industrial machine data analysts and proposed to them a schema of expected importance consisting of 3 levels (LOW, MEDIUM, HIGH). Their task was to agree on how critical and informative each feature could be for recognizing the Bulk or Res levels of the final product according to their experience and domain knowledge. Bulk and Resistance are the targets of the analysis (can be low, medium, or high) and for this reason are not displayed in Table 2.

Each numerical feature is rescaled between 0 and 1 via a min-max procedure (Formulae 1). Moreover, each categorical feature is processed via a one-hot encoding procedure, i.e. replacing categorical labels with binary encodings of their enumerates. This also allows to employ the Euclidean distance measure for the counterfactuals' search.

$$MinMax(x, X) = \frac{x - min(X)}{max(X) - min(X)} \tag{1}$$

5 Results and discussion

The experimental results have been provided by considering five different benchmark datasets from the UC Irvine Machine Learning Repository (i.e. pen-based, satimage, segment, letter, and zoo) and a real-world industrial dataset. All the experiments are performed using a Monte Carlo 10 folds validation framework.

Table 2. Expected feature importance level according to the domain expert.

Attribute	Units	Imp. for Bulk	Imp. for Res
ID	Integer	-	-
STRLA	N/m	LOW	MEDIUM
STRLO	N/m	LOW	MEDIUM
ELOLA	%	LOW	LOW
ELOLO	%	LOW	LOW
WEIGHT	gr/m^2	MEDIUM	MEDIUM
THICK	mm	-	LOW
TRH	ShA	LOW	LOW
BRH	ShA	-	LOW
COUPL	Category	MEDIUM	LOW
EMB	Category	MEDIUM	LOW
REW	Category	MEDIUM	LOW
ETR	Category	MEDIUM	MEDIUM
EBR	Category	MEDIUM	MEDIUM
TYPRO	Category	HIGH	HIGH
LAYERS	Integer	HIGH	HIGH
DRYRAT	Real	LOW	MEDIUM
STRCT	Boolean	HIGH	HIGH

The performances obtained are presented via their mean. As the main performance measure for the classification performance, the accuracy (Formulae 2) is used. In Formulae 2, C_i is one if the classification of instance i is correct, zero otherwise.

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N} C_i \tag{2}$$

As the main computational complexity measure, we use the computational time (in seconds).

Firstly, different state-of-the-art feature importance measures have been employed and compared against our approach, considering the similarity between the features ranking provided. Then we validated and compared SHAP and BoCSoR with the ground truth explanation provided by the real-world industrial dataset. Both BoCSoR and SHAP are implemented in Python. To summarize, we are validating our XAI algorithm in its ability to *open-the-black-box* on several benchmark datasets against other state-of-the-art approaches; moreover, by comparing those explanations with the ground truth of the expected feature importance we are also investigating how much the explanations provided by our approach resemble the domain-knowledge.

5.1 UCI datasets

We trained and tested an MLP classifier in a 10-cross-fold validation setup over the different benchmark datasets. The MLP has been implemented via the well-

known *scikit-learn* python library and consists of 3 dense fully connected layers with 128, 64, and 32 neurons respectively, *ReLU* as the activation function, and a last classification layer with softmax activation function with a number of neurons equal to the number of classes to be recognized (Tab. 3).

Table 3. Average classification accuracy of the MLP classifier over different benchmark datasets.

	penbased	satimage	segment	letter	zoo
Train Acc. (%)	0.99	0.99	0.98	0.98	1.0
Test Acc. (%)	0.99	0.99	0.97	0.94	0.90

The trained MLP model is also used to compute the feature importance via our approach and other state-of-the-art methods. This enables the measurement of the similarity between the resulting feature's importance rankings by considering a couple of classes. To compute the similarity between the rankings obtained via BoCSoR and the ones obtained via other feature importance approaches we employ the coefficient of ranking similarity (WS). WS is a ranking similarity that weights the disagreement between two rankings according to their position in the ranks.

$$WS = 1 - \sum_{i=1}^{N} (2^{-R_{x_i}} \frac{|R_{x_i} - R_{y_i}|}{\max(|1 - R_{x_i}|, |N - R_{x_i}|)}) \tag{3}$$

In the formula WS is one value of the similarity coefficient, N is the length of the rank, and R_{x_i} is the index of the feature in position i in the ranking R for the feature ranking provided by the approach x.

Table 4 shows the WS obtained by measuring the ranking similarity between BoCSoR and four other features importance measures, i.e. mutual information [26], relief [37], permutation importance [11], and SHAP[27]. Given the high similarity of the ranks obtained from BoCSoR and other measures of feature importance, we can infer that such ranking has some degree of reliability in capturing importance as measured by other feature importance approaches already known.

Table 4. Feature importance ranking similarity between Bocsor and other state-of-the-art approaches over the benchmark datasets.

	penbased	satimage	segment	letter	zoo
Mutual info.	0.75	0.88	0.88	0.93	0.87
Relief	**0.82**	**0.94**	0.73	**0.99**	**0.97**
Permut. imp.	0.59	0.90	0.95	0.82	0.95
SHAP	**0.82**	0.91	**0.96**	0.92	0.90

Since one of the most discussed problems in the literature on features' importance approaches is their computational complexity, we consider the computational time required by the approaches employed in the study to measure the importance of the features. To provide a fair comparison, we used the same number of instances considered by BoCSoR, i.e., the number of instances close to the decision boundary. To this end, we employed the KernelExplainer provided by SHAP with a random subsampling strategy. All the experiments have been run on a machine with the following computational resources: GeForce GTX 1660 super as GPU, 16 GB 3200 MHz RAM, and AMD Ryzen 3600 as CPU. Table 5 summarizes the obtained results.

Table 5. Computational time in seconds

	penbased	satimage	segment	letter	zoo
Mutual info.	0.60	1.04	0.18	0.46	0.11
Relief	5.27	6.76	7.04	6.48	4.99
Permut. imp.	0.69	1.26	0.23	0.56	0.06
SHAP	73.42	75.51	7.40	49.88	0.06
BoCSoR	**0.51**	**0.87**	**0.16**	**0.40**	**0.03**

As the tables 4 and 5 shows, BoCSoR provides a feature importance ranking similar to the ones provided by SHAP and Relief, which are popular state-of-the-art approaches, with all the benchmark datasets. At the same time, BoCSoR results in way less computational time than the others, in particular, if compared to SHAP.

5.2 Real-world Industrial dataset

By using the ground-truth knowledge provided by the real-world industrial expert we validate the measured feature importance obtained via BoCSoR and SHAP. The MLP classifier has been trained and tested in a 10-fold cross-validation on both the tasks provided by the dataset, i.e. BULK and RES classification. The performances of the model are detailed in Tab 6.

Table 6. Average classification performances of the selected ML model in the two tasks, BULK and RES classification, on the provided real-world industrial datasets.

	BULK			RES		
class	precision	recall	f1-score	precision	recall	f1-score
1	0.95	0.79	0.86	0.71	0.62	0.67
2	0.64	0.90	0.75	0.55	0.75	0.63
3	0.94	0.77	0.85	1.00	0.71	0.83
macro avg.	0.85	0.82	0.82	0.75	0.70	0.71
micro avg.	0.86	0.82	0.82	0.74	0.70	0.71

To measure the feature importance for the MLP classifiers, two different configurations of BoCSoR have been tested. Specifically, we employed 10 as the configuration for *steps* parameter, a threshold percentile, $percentileTh \in \{10, 20\}$, and a number of nearest neighbors, $k \in \{10, 20\}$. To reduce the task to a binary classification for the computation of the features' importance, the MLP classifier is trained in a 1vsAll fashion. To measure the agreement between the ranking of the features obtained via BoCSoR and SHAP and the ground truth, we group the obtained ranks by levels so that an accuracy measure can be computed. Since only 3 levels of feature importance are known, the accuracy was computed by considering the number of correctly assigned levels of feature importance. For example, if out of 10 features 4 have importance HIGH, the first 4 most important features obtained by a measure (e.g. BoCSoR or SHAP) are assigned that level. This is repeated for each importance level. Once the ranks obtained from SHAP and BoCSoR are reduced to a 3-level rank, it is possible to calculate how many of them were correctly assigned (i.e., as indicated by ground truth) and measure this via an accuracy metric.

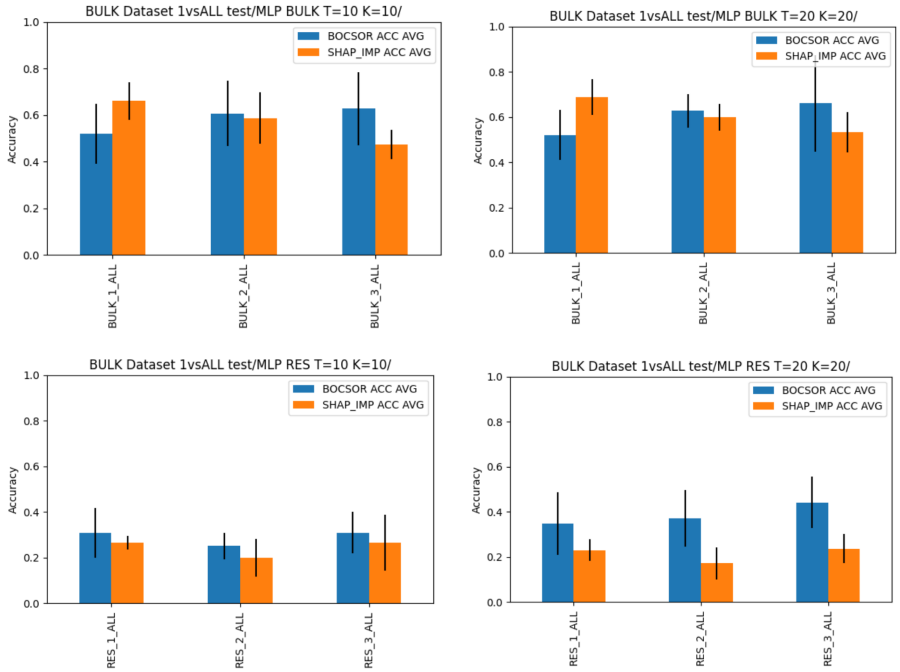

Fig. 2. Accuracy (mean and variance) of the rankings for the real world industrial dataset considering two different configurations of threshold percentile, T, and number of nearest neighbors selected, K.

As Fig. 2 shows, both SHAP and BoCSoR result in good performances when matching the ground truth provided by the domain expert for the classification of BULK levels. Specifically, BoCSoR has better performances than SHAP in both the configurations for the classes 2 and 3, while SHAP has better performances for the class 1. For the classification of RES levels instead, the accuracy of both approaches is slightly lower. Nevertheless, these performances are partially justified by the fact that the MLP results in lower RES levels' recognition performances if compared with the BULK levels' one. Considering the accuracy of the ranking of the features provided by SHAP and BoCSoR, Fig. 2 shows how BoCSoR has always better performances, i.e. if compared to SHAP, BoCSoR results in an improved agreement against the ground-truth explanations. This result is consistent despite the employed configurations of threshold percentile, T, and the number of nearest neighbors, K.

6 Conclusion

In this study, we proposed a knowledge-driven validation of a counterfactual-based features importance method, i.e. BoCSoR, and compared it against different state-of-the-art feature importance approaches. In our experiments, we considered five publicly available benchmark datasets and two real-world industrial datasets which also provide a ground truth explanation in the form of levels of features importance. The results obtained with the benchmark datasets show that BoCSoR provides features rankings comparable with the other state-of-the-art approaches with much less computation time, especially if we consider datasets with more number instances (e.g. penbased) and popular and successful state-of-the-art methods (e.g. SHAP). We also tested the ability of BoCSoR and SHAP to provide feature importance explanations aligned with the experts' domain knowledge. According to our results, BoCSoR provides more accurate feature importance in most of the configurations tested.

BoCSoR exploits a linear search starting from instances close to the decision boundary to keep the computational complexity low. However, this method cannot ensure the smallest distance between the instance and the obtained counterfactual, nor can it guarantee the best approximation of the decision boundary. The always-growing literature on counterfactual explanations can offer sophisticated approaches able to provide a better trade-off between decision boundary approximation and computational cost.

Also, BoCSoR treats categorical features via a one-hot encoding approach, but this may not be the optimal choice when dealing with a perturbation-based methodology to generate counterfactuals. Indeed, the midpoints between two categorical feature encodings might correspond to instances that are not meaningful for the problem under analysis.

Future research will explore these directions.

References

1. Adadi, A., Berrada, M.: Peeking inside the black-box: a survey on explainable artificial intelligence (xai). IEEE access **6**, 52138–52160 (2018)
2. Afchar, D., Guigue, V., Hennequin, R.: Towards rigorous interpretations: a formalisation of feature attribution. In: International inproceedings on Machine Learning. pp. 76–86. PMLR (2021)
3. Ahmed, I., Jeon, G., Piccialli, F.: From artificial intelligence to explainable artificial intelligence in industry 4.0: a survey on what, how, and where. IEEE Transactions on Industrial Informatics **18**(8), 5031–5042 (2022)
4. Alfeo, A.L., Cimino, M.G., Vaglini, G.: Technological troubleshooting based on sentence embedding with deep transformers. Journal of Intelligent Manufacturing **32**(6), 1699–1710 (2021)
5. Alfeo, A.L., Cimino, M.G., Vaglini, G.: Degradation stage classification via interpretable feature learning. Journal of Manufacturing Systems **62**, 972–983 (2022)
6. Alfeo, A.L., Cimino, M.G.C.A., Gagliardi, G.: Concept-wise granular computing for explainable artificial intelligence. Granular Computing pp. 1–12 (2022)
7. Alfeo, A.L., Cimino, M.G.C., , Gagliardi, G.: Automatic feature extraction for bearings' degradation assessment using minimally pre-processed time series and multi-modal feature learning. In: Proceedings of the 3rd International inproceedings on Innovative Intelligent Industrial Production and Logistics (IN4PL 2022) (2022)
8. Alfeo, A.L., Cimino, M.G.C., Egidi, S., Lepri, B., Pentland, A., Vaglini, G.: Stigmergy-based modeling to discover urban activity patterns from positioning data. In: Social, Cultural, and Behavioral Modeling: 10th International Conference, SBP-BRiMS 2017, Washington, DC, USA, July 5-8, 2017, Proceedings 10. pp. 292–301. Springer (2017)
9. Alfeo, A.L., Zippo, A.G., Catrambone, V., Cimino, M.G., Toschi, N., Valenza, G.: From local counterfactuals to global feature importance: efficient, robust, and model-agnostic explanations for brain connectivity networks. Computer Methods and Programs in Biomedicine p. 107550 (2023). https://doi.org/https://doi.org/10.1016/j.cmpb.2023.107550
10. Ali, S., Abuhmed, T., El-Sappagh, S., Muhammad, K., Alonso-Moral, J.M., Confalonieri, R., Guidotti, R., Del Ser, J., Díaz-Rodríguez, N., Herrera, F.: Explainable artificial intelligence (xai): What we know and what is left to attain trustworthy artificial intelligence. Information Fusion p. 101805 (2023)
11. Altmann, A., Toloşi, L., Sander, O., Lengauer, T.: Permutation importance: a corrected feature importance measure. Bioinformatics **26**(10), 1340–1347 (2010)
12. Arras, L., Osman, A., Samek, W.: Clevr-xai: a benchmark dataset for the ground truth evaluation of neural network explanations. Information Fusion **81**, 14–40 (2022)
13. Barr, B., Xu, K., Silva, C., Bertini, E., Reilly, R., Bruss, C.B., Wittenbach, J.D.: Towards ground truth explainability on tabular data. arXiv preprint arXiv:2007.10532 (2020)
14. Bay, S.D., Kibler, D., Pazzani, M.J., Smyth, P.: The uci kdd archive of large data sets for data mining research and experimentation. ACM SIGKDD explorations newsletter **2**(2), 81–85 (2000)
15. Crupi, R., Castelnovo, A., Regoli, D., San Miguel Gonzalez, B.: Counterfactual explanations as interventions in latent space. Data Mining and Knowledge Discovery pp. 1–37 (2022)

16. Delaney, E., Greene, D., Keane, M.T.: Instance-based counterfactual explanations for time series classification. In: International inproceedings on Case-Based Reasoning. pp. 32–47. Springer (2021)
17. Galhotra, S., Pradhan, R., Salimi, B.: Feature attribution and recourse via probabilistic contrastive counterfactuals. In: Proceedings of the ICML Workshop on Algorithmic Recourse. pp. 1–6 (2021)
18. Guidotti, R.: Evaluating local explanation methods on ground truth. Artificial Intelligence **291**, 103428 (2021)
19. Guidotti, R.: Counterfactual explanations and how to find them: literature review and benchmarking. Data Mining and Knowledge Discovery pp. 1–55 (2022)
20. Horel, E., Giesecke, K.: Computationally efficient feature significance and importance for predictive models. In: Proceedings of the Third ACM International Conference on AI in Finance. pp. 300–307 (2022)
21. İç, Y.T., Yurdakul, M.: Development of a new trapezoidal fuzzy ahp-topsis hybrid approach for manufacturing firm performance measurement. Granular Computing **6**(4), 915–929 (2021)
22. Jeyakumar, J.V., Noor, J., Cheng, Y.H., Garcia, L., Srivastava, M.: How can i explain this to you? an empirical study of deep neural network explanation methods. Advances in Neural Information Processing Systems **33**, 4211–4222 (2020)
23. Kommiya Mothilal, R., Mahajan, D., Tan, C., Sharma, A.: Towards unifying feature attribution and counterfactual explanations: Different means to the same end. In: Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society. pp. 652–663 (2021)
24. Kumar, I.E., Venkatasubramanian, S., Scheidegger, C., Friedler, S.: Problems with shapley-value-based explanations as feature importance measures. In: International in proceedings on Machine Learning. pp. 5491–5500. PMLR (2020)
25. Laugel, T., Renard, X., Lesot, M.J., Marsala, C., Detyniecki, M.: Defining locality for surrogates in post-hoc interpretablity. In: Workshop on Human Interpretability for Machine Learning (WHI)-International Conference on Machine Learning (ICML) (2018)
26. Liu, G., Yang, C., Liu, S., Xiao, C., Song, B.: Feature selection method based on mutual information and support vector machine. International Journal of Pattern Recognition and Artificial Intelligence **35**(06), 2150021 (2021)
27. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 4765–4774. Curran Associates, Inc. (2017), http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf
28. Marcílio, W.E., Eler, D.M.: From explanations to feature selection: assessing shap values as feature selection mechanism. In: 2020 33rd SIBGRAPI conference on Graphics, Patterns and Images (SIBGRAPI). pp. 340–347. Ieee (2020)
29. Markus, A.F., Kors, J.A., Rijnbeek, P.R.: The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies. Journal of Biomedical Informatics **113**, 103655 (2021)
30. Mosca, E., Szigeti, F., Tragianni, S., Gallagher, D., Groh, G.: Shap-based explanation methods: A review for nlp interpretability. In: Proceedings of the 29th International Conference on Computational Linguistics. pp. 4593–4603 (2022)
31. Mothilal, R.K., Sharma, A., Tan, C.: Explaining machine learning classifiers through diverse counterfactual explanations. In: Proceedings of the 2020 conference on fairness, accountability, and transparency. pp. 607–617 (2020)

32. Pat, N., Wang, Y., Bartonicek, A., Candia, J., Stringaris, A.: Explainable machine learning approach to predict and explain the relationship between task-based fmri and individual differences in cognition. bioRxiv pp. 2020–10 (2022)
33. Ribeiro, M.T., Singh, S., Guestrin, C.: Anchors: High-precision model-agnostic explanations. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
34. Setzu, M., Guidotti, R., Monreale, A., Turini, F., Pedreschi, D., Giannotti, F.: Glocalx-from local to global explanations of black box ai models. Artificial Intelligence **294**, 103457 (2021)
35. Sokol, K., Santos-Rodriguez, R., Flach, P.: Fat forensics: A python toolbox for algorithmic fairness, accountability and transparency. Software Impacts **14**, 100406 (2022)
36. Stepin, I., Alonso, J.M., Catala, A., Pereira-Fariña, M.: A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. IEEE Access **9**, 11974–12001 (2021)
37. Urbanowicz, R.J., Meeker, M., La Cava, W., Olson, R.S., Moore, J.H.: Relief-based feature selection: Introduction and review. Journal of biomedical informatics **85**, 189–203 (2018)
38. Vlassopoulos, G., van Erven, T., Brighton, H., Menkovski, V.: Explaining predictions by approximating the local decision boundary. arXiv preprint arXiv:2006.07985 (2020)
39. Wiratunga, N., Wijekoon, A., Nkisi-Orji, I., Martin, K., Palihawadana, C., Corsar, D.: Actionable feature discovery in counterfactuals using feature relevance explainers. CEUR Workshop Proceedings (2021)
40. Yang, M., Kim, B.: Benchmarking attribution methods with relative feature importance. arXiv preprint arXiv:1907.09701 (2019)