Comparison of Contextual Importance and Utility with LIME and Shapley Values *

Kary Främling*[1,2][0000−0002−8078−5172], Marcus Westberg[1][0000−0001−5261−8898], Martin Jullum[3][0000−0003−3908−5155], Manik Madhikermi[1][0000−0002−0811−2256], and Avleen Malhi[2,4][0000−0002−9303−655X]

[1] Department of Computing Science, Umeå University, Sweden
{kary.framling,marcus.westberg,manik.madhikermi}@umu.se
[2] Department of Computer Science, Aalto University
{kary.framling,avleen.malhi@aalto.fi}
[3] Norwegian Computing Center, Gaustadalleen 23a, 0373 Oslo, Norway
jullum@nr.no
[4] Department of Computing and Informatics, Bournemouth University, UK
amalhi@bournemouth.ac.uk

Abstract. Different explainable AI (XAI) methods are based on different notions of 'ground truth'. In order to trust explanations of AI systems, the ground truth has to provide fidelity towards the actual behaviour of the AI system. An explanation that has poor fidelity towards the AI system's actual behaviour can not be trusted no matter how convincing the explanations appear to be for the users. The Contextual Importance and Utility (CIU) method differs from currently popular outcome explanation methods such as Local Interpretable Model-agnostic Explanations (LIME) and Shapley values in several ways. Notably, CIU does not build any intermediate interpretable model like LIME, and it does not make any assumption regarding linearity or additivity of the feature importance. CIU also introduces the value utility notion and a definition of feature importance that is different from LIME and Shapley values. We argue that LIME and Shapley values actually estimate 'influence' (rather than 'importance'), which combines importance and utility. The paper compares the three methods in terms of validity of their ground truth assumption and fidelity towards the underlying model through a series of benchmark tasks. The results confirm that LIME results tend not to be coherent nor stable. CIU and Shapley values give rather similar results when limiting explanations to 'influence'. However, by separating 'importance' and 'utility' elements, CIU can provide more expressive and flexible explanations than LIME and Shapley values.

Keywords: Explainable AI · Contextual Importance and Utility · Outcome Explanation · Post Hoc Explanation.

* Corresponding Author: Kary.Framling@cs.umu.se.
The work is partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

1 Introduction

The need for explainability in Artificial Intelligence (AI) has been understood since the very beginnings of AI, as seen for instance in MYCIN [23]. Even though the term Explainable AI (XAI) is quite recent, AI explainability was a very active domain during the 1990's when a list of five general desiderata for any explanation was identified in [24], which were *Fidelity*, *Understandability*, *Sufficiency*, *Low Construction Overhead*, and *Efficiency*. XAI research in the 1990's can be considered to have focused on so-called *intrinsic interpretability* or *interpretable model extraction* [7], i.e. extract rules or other interpretable forms of knowledge from a complex black box model and then use that representation as an explanation. One exception to that trend was the Contextual Importance and Utility (CIU) method for outcome explanation, first presented in [12] and explained in detail in [9]. However, CIU seems to have passed unnoticed by the XAI community because the first paper on CIU since 1996 wasn't published until 2019. Using modern terms of XAI, CIU can be classified as a model-agnostic outcome explanation method. The first objective of this paper is to provide a comparison of CIU with two of the most popular model-agnostic outcome explanation methods available, i.e. Shapley values [16, 22] and LIME [21]. The category of use cases and data sets considered in this paper is tabular data only.

CIU's mathematical foundation and underlying philosophy are different from those of Shapley values and LIME. Notably, CIU is not an additive feature attribution method. Furthermore, CIU estimates *Contextual Importance (CI)* and *Contextual Utility (CU)* instead of estimating feature 'influence' like most (or all) comparable methods. However, 'influence' can be calculated directly from CI and CU values, which simplifies the comparison with influence-based methods, such as LIME and Shapley values. The second objective of the paper is to study to what extent the explanations produced by the studied XAI methods provide fidelity towards the true behaviour of the model.

The next section provides a background and definitions used in the paper, as well as an overview of Shapley values and LIME methods. Section 3 describes CIU and its use in this paper. Section 4 shows experimental results and comparisons between the three methods, followed by Conclusion.

2 Background and Definitions

The outcome explanation concept may be divided into two separate settings based on the aim of the explanation task. The first setting seeks explanations of how (each of) the input features influences the outcome solely through the given prediction model. This setting is most relevant when trying to understand the behaviour of the prediction model in itself. The second setting also accounts for the dependence between the input features, and may therefore assign high importance to an input feature that has a minor direct impact on the output through the prediction formula, if the feature is highly correlated with one or more features that *do* have such a high direct impact. This is most relevant when

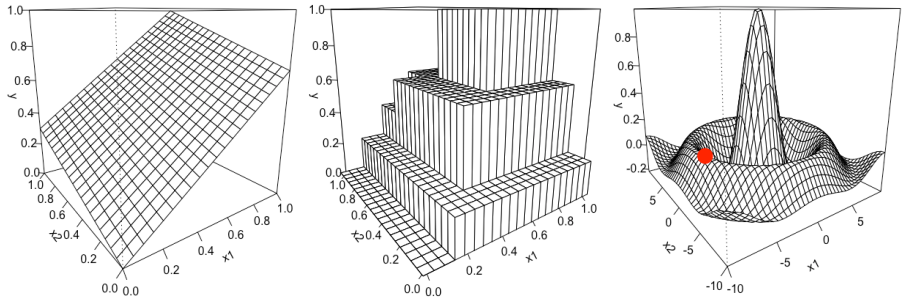
the actual real behaviour of the modelled output is of interest. Leaning on the fidelity criterion described below, we concentrate on the former setting here.

Going forward, it is important that we look at our definitions of the terms 'fidelity' and 'ground truth': When we refer to 'fidelity', what we mean is how accurately the explanation remains faithful/truthful to the underlying black-box model in its representation thereof. This follows similar definitions in [5, 19]. Following on that, the 'ground truth' of a model is the actual observed behaviour of that model. Concentrating solely on the model itself, it is generally admitted in the XAI domain that the actual input versus output behaviour of the underlying model is the so called ground truth against which the fidelity of an explanation should be assessed [4, 26]. The LIME (Local Interpretable Model-Agnostic Explanations) method [21], for instance, calculates to what extent the generated interpretable linear model gives similar results to the original black box model. That is called the Explanation Fit and is the R^2 error between the linear model and the actual model. Shapley values do not have a proper intermediate model where an Explanation Fit makes sense. However, one may interpret the additive Shapley value explanation as a model which is linear in the set of indicator variables defined as whether each of the input features are observed or not.

In human-to-human communication, an explanation lacking fidelity towards the real underlying model is usually considered to be a lie although it can appear convincing to the explainee. When developing and comparing XAI methods, the fidelity of the provided explanation in regard to the underlying model should be the first and foremost assessment criterion. An explanation lacking in fidelity might be considered easier to understand and accept than a true explanation, as depicted in some human surveys for assessing the goodness of different methods. However, a false explanation or lie that looks or sounds convincing should not lead to consideration that the underlying XAI method is better.

2.1 Core Definitions

The two fundamental concepts of CIU are 'importance' and 'utility' as explained in this section. Their origin is in Multi-Attribute Utility Theory [25], as explained also in [10]. An 'influence' concept can be calculated from 'importance' and 'utility' but it is not a core CIU concept and is here used mainly to simplify comparisons with other methods. In our usage of the terms 'influence', 'importance' and 'utility', the 'importance' of 'something' (such as an input feature of an AI model) denotes the significance of that 'something' but does NOT express adjoining positive or negative judgements. Something like 'good importance', 'bad importance', 'typical importance', etc., are not accurately represented by importance alone. Instead, adjectives such as 'good', 'bad', 'typical', 'favorable', etc., express judgments of the *utility* of feature *values* for the situation or context at hand, as provided by a *utility function* that expresses the *value utility* of both output and input values. LIME and Shapley values do not have a utility concept and typically use the term 'importance' for what we here call 'influence'. Even



(a) Weighted sum. (b) Rule-based. (c) Non-linear.

Fig. 1: Examples of linear, rule-based and non-linear models.

though [16] also uses the term 'importance', more recent Shapley values literature tends to also use the term feature *influence* [1]. [17] uses the term 'effect' in the same sense as we use 'influence' here.

The influence of a feature will depend on the feature's importance as well as on the utility of the current feature value. A feature with high importance and a good value utility will have great positive influence on the result. A feature with high importance and a bad value utility will have great negative influence on the result. A feature with zero importance for the output will have zero influence on the result, no matter what value utility it has. As such, our definitions for the core concepts of CIU look like this:

- **Importance**: The feature importance *in a particular context* of a factor impacting a particular decision.
- **Utility**: How well the values of the features in the same context match outcome expectations. This follows from the definition of *utility function* in decision theory, where it is a numerical representation of preference/desirability orderings [25].
- **Influence**: A combination value of utility and importance, representing the positive or negative impact of a factor on a particular decision, typically relative to some 'baseline' [6].

In the function $y = b(x) = 0.3x_1 + 0.7x_2$, represented by Figure 1a, the influence of the x_1 term is $0.3x_1$ and the influence of the x_2 term is $0.7x_2$ if we use a zero baseline. The function could also be expressed in the more generic form $y = w_1 \times x_1 + w_2 \times x_2$. The weights (importances) w_1 and w_2 are in this case 0.3 and 0.7. The utility function in this case is unity because the utility or 'goodness' of an input value is directly the input value itself. For instance, if x_1 and x_2 have different value ranges, then it becomes necessary to apply a utility function to them. If the input value range would be $[0, 5]$ rather than $[0, 1]$, then a utility function $u_i(x_i) = x_i/5$ would be appropriate. For generic black box models, the utility function can be an arbitrarily complex, non-linear function.

Interpretability becomes more challenging when dealing with step-wise functions such as the one illustrated in Figure 1b, which corresponds to the kind

of functions produced by rules and decision trees. Model-agnostic methods like the ones studied in this paper can also deal with such models. However, the significance of the concepts influence, importance and utility becomes more complicated than for the linear case. When dealing with non-linear models such as that in Figure 1c ($y = \sin(\sqrt{x_1^2 + x_2^2})/\sqrt{x_1^2 + x_2^2}$), the 'influence' concept alone becomes increasingly challenging to use. Thus, we argue for the contextual 'importance' and 'utility' concepts used in CIU.

2.2 LIME

Ribeiro et al. [21] in their research work proposed a method called Local Interpretable Model agnostic Explanations (LIME) for explanation of an individual prediction $b(x)$ made by a black box machine learning model. The approach used to explain the individual predictions can be detailed as: In sampling step (1), a set of normally distributed instances X_{sx} is drawn having same mean and standard deviation as the original feature space of X, which is done independently of the instance x to be explained. For the labels $Y_{sx} = b(X_{sx})$, LIME works with the prediction returned by the model b. In surrogate fitting step (2), the LIME surrogate is trained to locally approximate the decision boundary of the black-box model. The standard version (Linear LIME) uses linear regression with regularization to do this. The local surrogate model centered on x is fitted by having each instance of X_{sx} associated with a weight calculated using an RBF kernel by default, i.e. higher importance will be assigned to instances closer to x during the training [15]. In the last explanation step (3), the explanations for the prediction $b(x)$ are generated by using the trained surrogate s_x's linear regression coefficients. Choosing an adequate and representative sampling strategy for generating the instances to fit the surrogate model has a major impact on the quality of the local approximation of the black-box model and thus on the accuracy of the generated explanation [14]. In particular, the effect of locally important features can be hidden by globally important ones.

LIME's ground truth could be summarized as follows: Find a linear regression function that locally approximates the tangent plane of the underlying model as well as possible for the current instance.

LIME's fidelity towards the LIME ground truth is assessed based on how well the linear regression corresponds to the actual behaviour of the model, which LIME calls the 'explanation fit' and is an R^2 value calculated on the difference between the actual model output and the output given by LIME's linear regression function.

The LIME experiments of this paper have been executed using the R-package `lime`, version 0.5.1 [20].

2.3 Shapley values

Shapley value is a concept originating from cooperative game theory [22]. The concept was picked up by the XAI community and became popular for producing outcome explanations following [16], and the introduction of SHAP (SHapley

Additive exPlanations). The method distributes the difference between the prediction output and the global mean prediction, additively on the input features according to a formula which is consistent with a set of four theoretical properties. A key ingredient in the Shapley values methodology is the contribution function $v(S)$, measuring expected output $b(x)$ when only a subset S of the input features were available (x_S). Motivated by the fidelity criterion, we have used the so-called interventional conditional expectation, as in [16]. Other choices may be more appropriate in other explanations settings, as explained e.g. in [6]. In the case of interventional conditional expectation, the Shapley value for feature i is a weighted mean over $v(S+i)-v(S)$ for all subsets S, and therefore measures the influence that the act of observing feature i has on the predicted output, with or without each of the other features observed. This allows the Shapley value for a feature to be compared with other features within the individual/instance and also with the same feature for other individuals/instances. A significant drawback with Shapley values is that it is computationally costly when there are many input features. Explanation through Shapley values also requires the availability of the training set, which may not always be easily accessible.

Shapley values' ground truth could be summarized as follows: Distribute the difference between the current and expected (e.g. the global mean prediction) output value to the input features according to a 'fairness estimation' about how much each feature attributed to the output in a positive or negative way.

The fidelity of Shapley values towards the Shapley value ground truth can be guaranteed by a sufficiently great sampling of all value combinations. The main challenge is that the number of such combinations grows exponentially with the number of input features.

The experiments of this paper have been executed using the `iml` (Interpretable Machine Learning) R-package, version 0.10.1 [18].

3 Contextual Importance and Utility (CIU)

A formal presentation of CIU can be found in [11]. In this paper, we will explain the principles of CIU using the so-called 'sombrero' function in Figure 1c as an example. The studied instance or Context $\vec{C}$ is indicated by the red dot in Figure 1c and corresponds to the input values $(x_1, x_2) = (-7.5, -1.5)$. Figure 2 shows how the output value y changes as a function of x_1 and x_2 when keeping the other input at the $\vec{C}$ value. The range of possible input values is here $[-10, 10]$ for both x_1 and x_2. In Figure 2, five values are indicated:

- *absmin*, *absmax*: The minimal and maximal values that the output y can get. In classification tasks these values are typically zero and one for all outputs.
- *Cmin*, *Cmax*: The minimal and maximal values that the output y can take by changing the value of the studied input.
- *out*: The value of output y for the studied instance, i.e. with input values $\vec{C}$.

In Figure 2, $absmin = -0.217$, $absmax = 1$ and $out = 0.128$. For the x_1 input $Cmin = -0.217$ and $Cmax = 0.664$. The contextual importance CI expresses

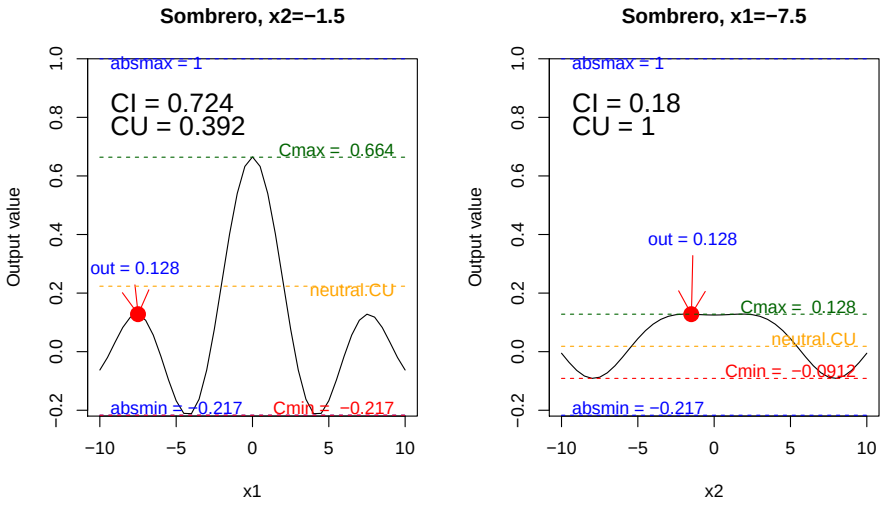


Fig. 2: CIU for sombrero function.

to what extent an input can modify the output value, which leads us to the following definition

$$CI_j(\vec{C}, \{i\}) = \frac{Cmax_j(\vec{C}, \{i\}) - Cmin_j(\vec{C}, \{i\})}{absmax_j - absmin_j}, \tag{1}$$

where the different variables have the same meaning as before but with appropriate indices as follows:

- $\{i\}$ defines the indices of inputs $\vec{x}$ for which CIU is calculated.
- j is the index of the studied output.

For input x_1 in Figure 2, this gives $CI_{x_1} = \frac{0.664-(-0.217)}{1-(-0.217)} = 0.724$ and $CI_{x_2} = \frac{0.128-(-0.0912)}{1-(-0.217)} = 0.18$ for input x_2. Therefore, x_1 is about four times as important as x_2 in the studied context $\vec{C}$.

CU expresses to what extent the current feature value(s) contribute to a high-utility output value, i.e what is the utility of the input value for achieving an output value that has a high utility. CU is expressed as

$$CU_j(\vec{C}, \{i\}) = \frac{y_j(\vec{C}) - Cmin_j(\vec{C}, \{i\})}{Cmax_j(\vec{C}, \{i\}) - Cmin_j(\vec{C}, \{i\})}, \tag{2}$$

where $y_j = b(\vec{C})$ corresponds to the *out* value in the example. For input x_1 in Figure 2, this gives $CU_{x_1} = \frac{0.128-(-0.217)}{0.664-(-0.217)} = 0.392$ and $CU_{x_2} = \frac{0.128-(-0.0912)}{0.128-(-0.0912)} = 1$ for input x_2. Therefore, x_1 has a less than average favorable value, whereas x_2 has the most favorable value possible in the context $\vec{C}$.

CI and CU are limited to the interval $[0, 1]$ by definition. In classification tasks, the transformation of output values into utility values is trivial because the output value can be considered to already be a probability/utility value in the range $[0, 1]$. For regression tasks, the output values need to be mapped into utility values through a utility function $u(y_j)$, where y_j is the value of output j. For instance, in the well-known Boston Housing data set, the output value is the median value of owner-occupied homes in \$1000's and is in the range $[5, 50]$. A straightforward way of transforming that value into a utility value is an affine transformation $[5, 50] \mapsto [0, 1]$, assuming that the preference is to have a higher value. However, from a buyer's point of view, the preference might be for lower prices and then the transformation would rather be $[50, 5] \mapsto [0, 1]$. It is important to point out that the definitions of CI and CU in Equations 1 and 2 assume that $u(y_j)$ is an affine transformation of the form $u(y_j) = Ay_j + b$ where A is positive. In principle, $u(y_j)$ could have any shape as long as it produces values in the range $[0, 1]$ but that case goes beyond the scope of the current paper.

In the original work by Främling [9], textual explanations were generated by quantifying CI and CU values according to intervals such as $very\_important = [0.9, 1]$ for CI and $very\_good = [0.9, 1]$ for CU. In this paper, CIU explanations are provided by bar plot explanations for simplifying comparisons with LIME and Shapley values. The only subjective parameter in that case is the choice of what CU value is considered 'neutral'. We call that parameter $neutral.CU$ here and it provides a 'baseline' for influence-based explanations using CIU. In Section 4, $CU = 0$ corresponds to red, $CU = 0.5$ is 'neutral' and corresponds to yellow, and $CU = 1$ corresponds to dark green, as illustrated by the colours of $Cmin$, $Cmax$ and $neutral.CU$ in Figure 2.

In order to make the difference between the concepts 'influence', 'importance' and 'utility' more explicit, we here provide a definition of *contextual influence*. Such a 'contextual influence' concept makes it possible to compare directly with the influence value ϕ of LIME and Shapley values, which is the reason why we use the symbol ϕ in Equation 4. However, using 'influence' makes explanations less expressive and less understandable than when using CI and CU, as illustrated in Section 4. We begin by defining contextual influence according to:

$$Cinfluence_j(\vec{C}, \{i\}) = CI_j(\vec{C}, \{i\}) \times CU_j(\vec{C}, \{i\}) \tag{3}$$

Since $Cinfluence$ is relative, it can be freely scaled into any desired range $[rmin, rmax]$. Such a 'scaled contextual influence' can be defined as follows:

$$\phi = (rmax - rmin) \times CI \times (CU - neutral.CU) \tag{4}$$

where '$_j(\vec{C}, \{i\})$' has been omitted from all three terms ϕ, CI, and CU for easier readability. For comparison with Shapley values and LIME, we use $[rmin, rmax] = [-1, 1]$ in Section 4. Setting $neutral.CU = 0.5$ also makes it possible to restrict ϕ values to only negative, zero or positive, as for Shapley values and LIME.

A formal study of the relationship between CU, CI and ϕ is out of the scope for the current paper. Other aspects of CIU that are not in the scope of this paper is how $Cmin$ and $Cmax$ are estimated. The sampling method used in this paper is described in [13]. Främling also introduced so-called *intermediate concepts* in [8, 9], which use the fact that CI and CU can be estimated for any joint combination of input features, i.e. the set $\{i\}$ in Equations 1 and 2 can contain any number of inputs, from one to all inputs. However, LIME and Shapley values do not have any intermediate concepts so it is not possible to perform a comparison with them, which is the main reason for not including intermediate concepts in this paper.

CIU's ground truth could be summarized as follows: Estimate how much the output can change when modifying the values of one or more input features, on a scale of 0-100% (Contextual Importance). Provide an estimate of how favorable the current value(s) are towards a high-utility output value, as compared to all possible values for the studied input features on a scale 0-100% (Contextual Utility). The fidelity of CIU towards its ground truth depends only on how accurately $Cmin$ and $Cmax$ values can be estimated.

The CIU experiments in this paper have been executed using the `ciu` R-package, version 0.1.0 [13] and using the latest version at https://github.com/KaryFramling/ciu for 'influence' plots.

4 Experiments

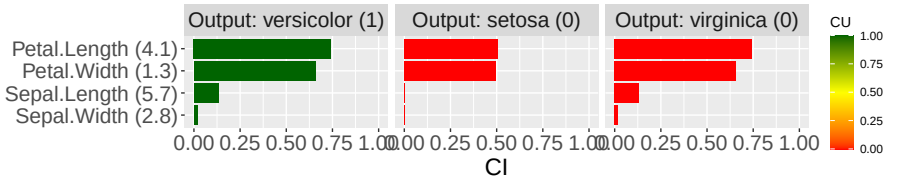
The data sets to be used for assessing the different methods have been selected so that discrete and continuous values are used as inputs and that both classification and regression tasks are taken into consideration. The results of the methods are evaluated mainly using two assessment criteria (AC):

- *AC1* Is the explanation rational and in line with the output value? For a high-utility output value, the total influence of features is expected to be highly positive, and vice versa for a low-utility output value.
- *AC2* Does the explanation correspond to the actual observed behaviour of the model?

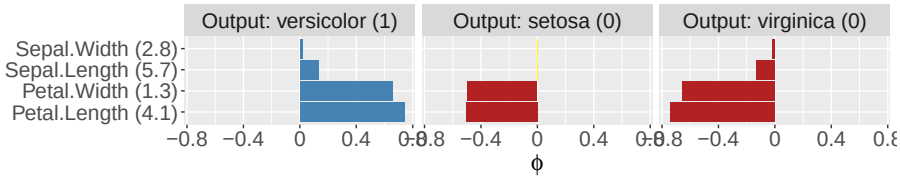
4.1 Classification with continuous inputs

The Iris data set has been chosen for this category mainly because the limits between the different classes require highly non-linear models for correctly estimating the probability of the three classes for each studied instance. Figure 3 shows CIU, Shapley values and LIME explanations generated for instance number 100 with a random forest model for the Iris data set. Any instance from the data set could be used but for Iris flowers the classes 'versicolor' and 'virginica' are usually the most interesting ones because they are more similar to each other than to 'setosa'. Instance number 100 is a 'versicolor'.

CIU with influence and Shapley values give almost identical results here. The output value is 'one' for the versicolor output and 'zero' for the two other classes,

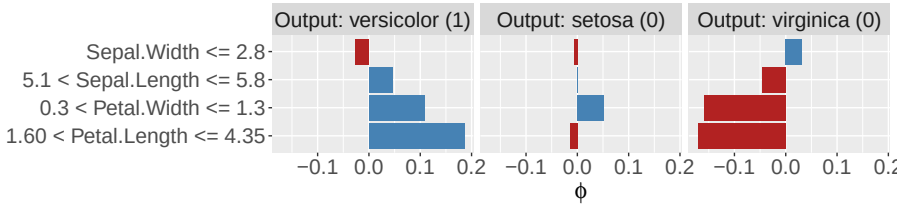


(a) CIU. Bar length shows CI, bar color shows CU according to palette on the right.



(b) CIU with influence. Positive influence is shown in blue, negative in red.

(c) Shapley values. Positive influence is shown in blue, negative in red.



(d) LIME. Positive influence is shown in blue, negative in red.

Fig. 3: Explanations with four methods for instance #100 of Iris data set. Bar length shows CI/ϕ value. CU value determines bar color in CIU plot. In influence plots, negative influence is shown in red and positive influence is shown in blue.

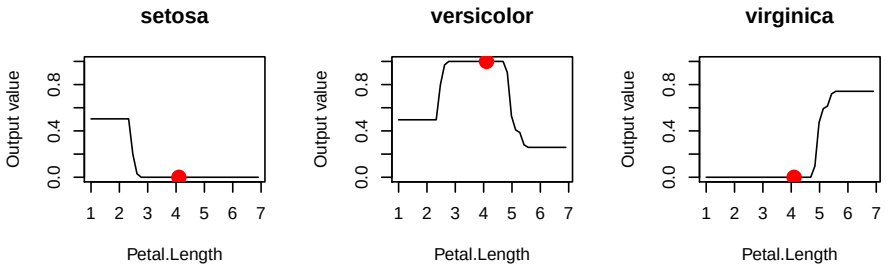


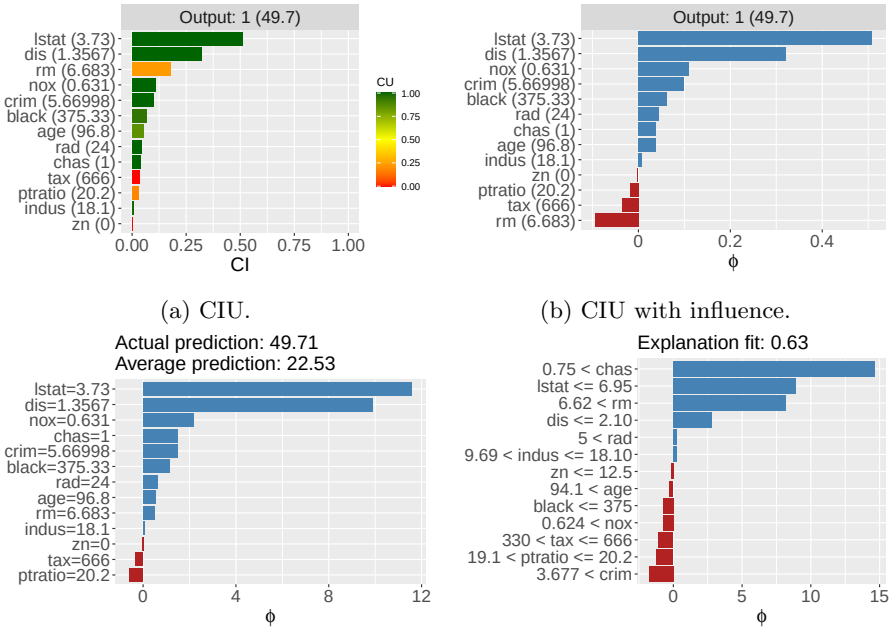
Fig. 4: Output values as a function of 'Petal Length' for the three Iris classes.

which is well represented also in the explanations so AC1 is fulfilled for CIU and Shapley values. LIME results differ significantly from CIU and Shapley values for the setosa class, where *Petal Width* has a significant positive influence that is not in line with the output value 'zero'. LIME results also tend to change from one run to the other. Therefore, LIME fails against AC1 for setosa explanation. It is also interesting to note that the 'Explanation fit' indicated by LIME is very low, i.e. < 0.1 for all three classes. Regarding AC2, Figure 4 shows the 'CIU ground truth' for the *Petal Length* feature. CIU values can be deduced directly from the figure and therefore fulfill AC2. Both Shapley values and LIME can also be considered to fulfill AC2 for *Petal Length*.

4.2 Regression with continuous inputs

The Boston Housing data has one continuous-valued output and only continuous-valued inputs. It is a regression task for which a Gradient Boosting Machine model is used here. Figure 5 shows CIU, Shapley values and LIME results for instance #370 of the data set. CIU and Shapley values again obtain quite similar results. Instance #370 has almost the highest possible value (49), which signifies that most input features should have a positive influence (but it could be any other instance too). The influence is here positive for most features with all methods, even though a little bit less so for LIME than for the others. Therefore, all methods satisfy AC1.

Regarding AC2, Figure 6 shows the 'CIU ground truth' for three input features. Again, CIU values can be deduced directly from these figures and therefore fulfill AC2, which is true also for Shapley values. For LIME, however, the dummy variable 'Charles River' (*chas*) is indicated as the most important one, which is a clear error. For the *lstat* feature, LIME only puts it third. For the *rm* feature, LIME gives a high positive influence (after *chas*), even though it is clear that 6.7 is only an average value for instance #370. Finally, LIME shows strong negative influence for the criminality rate (*crim*), even though the value 5.7 is actually good. Hence, LIME fails to satisfy AC2.



(a) CIU. (b) CIU with influence.

(c) Shapley values. (d) LIME.

Fig. 5: Explanations for instance #370 of Boston Housing data set. Bar length shows CI/ϕ value. CU value determines bar color in CIU plot. In influence plots, negative influence is shown in red and positive influence is shown in blue.

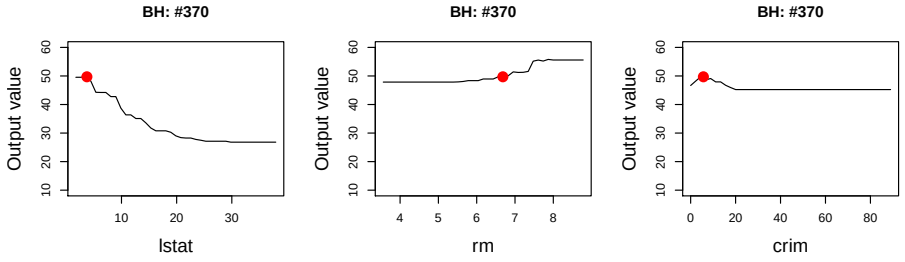
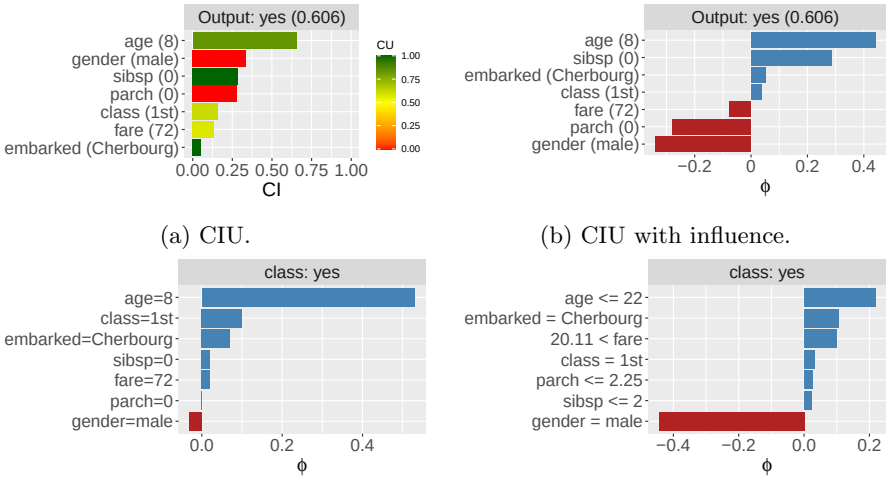


Fig. 6: Boston Housing output value as a function of input value for features 'lstat', 'rm' and 'crim'.

4.3 Classification with mixed discrete and continuous inputs



(a) CIU.

(b) CIU with influence.

(c) Shapley values

(d) LIME.

Fig. 7: Bar chart explanations for example person on Titanic. Only explanations for 'survives' have been included. Bar length shows CI/ϕ value. CU value determines bar color in CIU plot. In influence plots, negative influence is shown in red and positive influence is shown in blue.

The Titanic data set is a frequently used benchmark for machine learning methods. It has two output classes, i.e. survives or not. There are both discrete and continuous-valued input features, which makes it interesting also for this paper. A random forest model was used. The studied instance is an 8-year old boy. The corresponding feature values are shown by the red dots in Figure 8. With output probabilities of 0.61 for *survives* and 0.39 for *doesn't survive*, it could be expected that there's dominantly positive influence for *survives* and dominantly negative influence for *doesn't survive*. This is indeed the case for CIU, whereas Shapley values has almost only positive influence for *survives* and therefore almost only negative influence for *doesn't survive*. It can therefore be questioned whether Shapley values satisfies AC1 here. When studying the effect of input feature values on the probability for *survives*, it seems like the influence of *age* is by far over-estimated by Shapley values in this case, which signifies that the Shapley values explanation does not correspond to the true behaviour of the model. Therefore Shapley values does not satisfy AC2 in this case.

The LIME explanation again differs from the two others, where 'male' is indicated as the input feature that clearly has the greatest influence. LIME results

change slightly at every run and sometimes 'parch' gets a negative influence, which is in line with the results of the other methods.

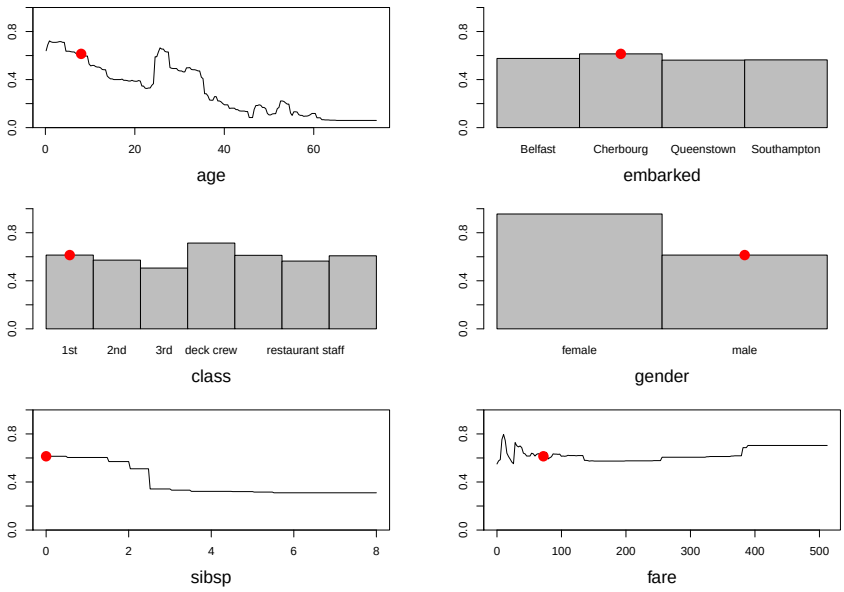


Fig. 8: Probability of survival for selected person in Titanic as a function of selected inputs.

5 Conclusion

As seen from the results in this paper, CIU provides a new alternative to LIME and Shapley values. The results confirm results of earlier research that LIME explanations tend to be less rational and provide a poor fidelity with the underlying model [2, 3]. CIU and Shapley values provide quite similar results for two of the studied use cases, which can be considered to be a comforting result for both methods. However, the 'ground truth' of Shapley values and CIU differ significantly and further empirical and theoretical studies regarding these differences and their effects would be important for the XAI community as a whole. The core conclusions of the paper are the following:

1. By considering 'importance' and 'utility' as different parts of an explanation, CIU can provide more versatile explanations than LIME and Shapley values.
2. Both 'importance' and 'utility' are absolute values in the range $[0, 1]$, whereas 'influence' is a relative value that only expresses how influent different input features are compared to each other.

3. CIU is not a black box itself because CI and CU values can be 'read out' by humans from input-output graphs at least for one input feature.
4. CIU does not need access to the training data. CIU can be applied to any model f, no matter if f has been produced by machine learning or not.

CIU is intuitively a more light-weight method than Shapley values because it only modifies the values of one input feature at a time, therefore requiring a smaller number of samples. However, the number of samples remains a compromise with the estimation accuracy, which makes it difficult to properly compare calculation overhead between the methods. Furthermore, calculation speed also depends on how the method has been implemented, not only on the method itself. Therefore, such a study is left as a topic of future work.

References

1. Aas, K., Jullum, M., Løland, A.: Explaining individual predictions when features are dependent: More accurate approximations to shapley values. arXiv preprint arXiv:1903.10464 (2019)
2. Adebayo, J., Gilmer, J., Goodfellow, I., Kim, B.: Local explanation methods for deep neural networks lack sensitivity to parameter values. arXiv preprint arXiv:1810.03307 (2018)
3. Alvarez-Melis, D., Jaakkola, T.S.: On the robustness of interpretability methods. arXiv preprint arXiv:1806.08049 (2018)
4. Amiri, S.S., Weber, R.O., Goel, P., Brooks, O., Gandley, A., Kitchell, B., Zehm, A.: Data representing ground-truth explanations to evaluate xai methods. arXiv preprint arXiv:2011.09892 (2020)
5. Barbado, A., Corcho, O.: Explanation generation for anomaly detection models applied to the fuel consumption of a vehicle fleet (2020)
6. Chen, H., Janizek, J.D., Lundberg, S., Lee, S.I.: True to the model or true to the data? arXiv preprint arXiv:2006.16234 (2020)
7. Du, M., Liu, N., Hu, X.: Techniques for interpretable machine learning. Communications of the ACM **63**(1), 68–77 (2020). https://doi.org/10.1145/3359786
8. Främling, K.: Explaining results of neural networks by contextual importance and utility. In: Proceedings of the AISB'96 conference. Brighton, UK (1-2 April 1996)
9. Främling, K.: Modélisation et apprentissage des préférences par réseaux de neurones pour l'aide à la décision multicritère. Phd thesis, INSA de Lyon (Mar 1996)
10. Främling, K.: Decision theory meets explainable ai. In: Calvaresi, D., Najjar, A., Winikoff, M., Främling, K. (eds.) Explainable, Transparent Autonomous Agents and Multi-Agent Systems. pp. 57–74. Springer International Publishing, Cham (2020)
11. Främling, K.: Explainable ai without interpretable model. arXiv preprint arXiv:2009.13996 (2020), https://arxiv.org/abs/2009.13996
12. Främling, K., Graillot, D.: Extracting Explanations from Neural Networks. In: ICANN'95 Conference. Paris, France (Oct 1995)
13. Främling, K.: Contextual importance and utility in R: the 'ciu' package. In: Proceedings of 1^{st} Workshop on Explainable Agency in Artificial Intelligence, at 35^{th} AAAI Conference on Artificial Intelligence. pp. 110–114 (2021)

14. Gruber, S., Kopper, P.: Introduction to local interpretable model-agnostic explanations (lime). https://compstat-lmu.github.io/iml_methods_limitations/lime.html (2020)
15. Laugel, T., Renard, X., Lesot, M.J., Marsala, C., Detyniecki, M.: Defining locality for surrogates in post-hoc interpretablity. arXiv preprint arXiv:1806.07498 (2018)
16. 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)
17. Molnar, C.: Interpretable Machine Learning. https://christophm.github.io/interpretable-ml-book/ (2019)
18. Molnar, C., Bischl, B., Casalicchio, G.: iml: An R package for interpretable machine learning. JOSS **3**(26), 786 (2018)
19. Papenmeier, A., Englebienne, G., Seifert, C.: How model accuracy and explanation fidelity influence user trust. CoRR **abs/1907.12652** (2019), http://arxiv.org/abs/1907.12652
20. Pedersen, T.L., Benesty, M.: lime: Local Interpretable Model-Agnostic Explanations (2019), https://CRAN.R-project.org/package=lime, r package version 0.5.1
21. Ribeiro, M.T., Singh, S., Guestrin, C.: " why should i trust you?" explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 1135–1144 (2016)
22. Shapley, L.S.: A value for n-person games. Contributions to the Theory of Games **2**(28), 307–317 (1953)
23. Shortliffe, E.H., Davis, R., Axline, S.G., Buchanan, B.G., Green, C., Cohen, S.N.: Computer-based consultations in clinical therapeutics: Explanation and rule acquisition capabilities of the mycin system. Computers and Biomedical Research **8**(4), 303 – 320 (1975). https://doi.org/https://doi.org/10.1016/0010-4809(75)90009-9
24. Swartout, W.R., Moore, J.D.: Explanation in Second Generation Expert Systems, p. 543–585. Springer-Verlag, Berlin, Heidelberg (1993)
25. Wallenius, J., Dyer, J.S., Fishburn, P.C., Steuer, R.E., Zionts, S., Deb, K.: Multiple criteria decision making, multiattribute utility theory: Recent accomplishments and what lies ahead. Manage. Sci. **54**(7), 1336–1349 (Jul 2008)
26. Yang, F., Du, M., Hu, X.: Evaluating explanation without ground truth in interpretable machine learning. arXiv preprint arXiv:1907.06831 (2019)