Data Mining and Knowledge Discovery
https://doi.org/10.1007/s10618-024-01010-5

Interpretable representations in explainable AI: from theory to practice

Kacper Sokol[1,2] · **Peter Flach**[1]

Received: 31 March 2021 / Accepted: 22 January 2024

Abstract

Interpretable representations are the backbone of many explainers that target black-box predictive systems based on artificial intelligence and machine learning algorithms. They translate the low-level data representation necessary for good predictive performance into high-level human-intelligible concepts used to convey the explanatory insights. Notably, the explanation type and its cognitive complexity are directly controlled by the interpretable representation, tweaking which allows to target a particular audience and use case. However, many explainers built upon interpretable representations overlook their merit and fall back on default solutions that often carry implicit assumptions, thereby degrading the explanatory power and reliability of such techniques. To address this problem, we study properties of interpretable representations that encode presence and absence of human-comprehensible concepts. We demonstrate how they are operationalised for tabular, image and text data; discuss their assumptions, strengths and weaknesses; identify their core building blocks; and scrutinise their configuration and parameterisation. In particular, this in-depth analysis allows us to pinpoint their explanatory properties, desiderata and scope for (malicious) manipulation in the context of tabular data where a linear model is used to quantify the influence of interpretable concepts on a black-box prediction. Our findings lead

Responsible editor: Martin Atzmueller, Johannes Fürnkranz, Tomáš Kliegr and Ute Schmid.

This work was supported by the TAILOR Network—an ICT-48 European AI Research Excellence Centre funded by EU Horizon 2020 research and innovation programme (grant agreement number 952215). Additional funding was provided by the ARC Centre of Excellence for Automated Decision-Making and Society, sponsored by the Australian Government through the Australian Research Council (project number CE200100005); and the Hasler Foundation (grant number 23082).

✉ Kacper Sokol
k.sokol@bristol.ac.uk; kacper.sokol@inf.ethz.ch

Peter Flach
peter.flach@bristol.ac.uk

[1] Intelligent Systems Laboratory, University of Bristol, Bristol, UK
[2] Department of Computer Science, ETH Zurich, Zurich, Switzerland

to a range of recommendations for designing trustworthy interpretable representations; specifically, the benefits of class-aware (supervised) discretisation of tabular data, e.g., with decision trees, and sensitivity of image interpretable representations to segmentation granularity and occlusion colour.

Keywords Interpretability · Explainability · Surrogates · Post-hoc · Interpretable representations · Machine learning · Artificial intelligence

1 Introduction

Interpretable representations (IRs) are the foundation of many explainability methods that target black-box predictive models based on artificial intelligence (AI) and machine learning (ML) algorithms. They are the core component of surrogate explainers, which are techniques to approximate the functioning and behaviour of an unintelligible classifier or regressor with a simpler model (Friedman and Popescu 2008; Ribeiro et al. 2016; Lundberg and Lee 2017; Sokol et al. 2019). More broadly, IRs facilitate translating the "language" of ML models—low-level data representations required for good predictive performance, such as raw feature values and their complex embeddings—into high-level concepts that are understandable to humans. IRs, therefore, create an *interface* between a computer-readable encoding of a phenomenon (captured by the collected data) and cognitively digestible chunks of information, thus establishing a medium suitable for conveying explanations. Notably, interpretable representations directly control the (perceived) complexity of the ensuing explanations, define the question that these insights answer, and restrict the explanation types that can effectively communicate this information—e.g., influence or importance of interpretable concepts, counterfactuals and what-if statements—making IR-based explainers highly flexible, versatile and appealing. In essence, by customising the interpretable representation we can adjust the content and comprehensibility of the resulting explanations and tune them towards a particular *audience* and *application*.

The algorithmic process responsible for transforming data from their original domain into an interpretable representation is usually defined by a human. An IR of images, for example, can be created with a *super-pixel segmentation*, i.e., partitioning images into non-overlapping clusters of pixels, each one representing an object of interest or pieces thereof. Similarly, text can be split into *tokens* denoting individual words, their stems or collections of words that are not necessarily adjacent. Tabular data containing numerical features can be *discretised* to capture meaningful patterns, e.g., people belonging to different age groups. Such interpretable representations are often paired with a simple and inherently transparent model to form a *surrogate explainer*; for example, LIME—Local Interpretable Model-agnostic Explanations (Ribeiro et al. 2016)—uses a sparse linear model. IR-based explainers are thus data-universal, in addition to often being model-agnostic and post-hoc (Sokol and Flach 2020a), which makes them an attractive choice given that they can be used with pre-existing black-box ML models.

Given the high complexity of such end-to-end explainers, many of them are composed of generic, thus versatile, algorithmic building blocks and focus on maximising

their overall performance, hence forgoing selection and optimisation of their individual parts (Sokol et al. 2019). Moreover, these explainers seek to automate the entire process to enable their deployment and evaluation at scale, which understandably requires components that can be operationalised without human input. Given the vast array of algorithmic choices in this space—as well as their individual configurations—such explainers are in fact complex entities suffering from overparameterisation, which often manifests itself in multiple contributing sources of randomness and low fidelity of the resulting explanations (Zhang et al. 2019; Rudin 2019; Lakkaraju et al. 2019; Lakkaraju and Bastani 2020; Sokol et al. 2019; Sokol and Flach 2020b). This observation is particularly pertinent to interpretable representations, popular examples of which are: *quantile discretisation* for numerical features of tabular data; edge-based *super-pixel segmentation* for images, e.g., via quick shift (Vedaldi and Soatto 2008); and whitespace-based *tokenisation* for text (Ribeiro et al. 2016).

Many deficiencies plaguing such explainers can be attributed to misuse of the underlying interpretable representation, which can make or break an explainer (Sokol et al. 2019; Sokol and Flach 2020b; Sokol 2021). These problems can be magnified, possibly rendering the explainer unusable, through certain pairings of IRs and types of surrogate models, especially when the implicit assumptions behind both of these components are at odds. By understanding the characteristics and behaviour of each interpretable representation and its influence on the resulting explanations—both on its own and in conjunction with a particular surrogate model family—we can explicate the theoretical properties of such explainers and assess their applicability and usefulness for the problem at hand. This area of research is largely under-explored for IRs on their own and as a part of an explainer, potentially leading to suboptimal design choices and inadequate explanations. An especially impactful research direction, which we focus on in this paper, is *automatic* creation of IRs that are trustworthy, robust and faithful to enable their creation, optimisation and deployment with minimal human input.

The important tasks of choosing an appropriate interpretable representation and configuring it are not often considered in the literature. It is common to assume that an IR is given or to reuse one that was proposed in prior work without much afterthought or deliberation about its suitability, (implicit) assumptions, properties and caveats (Laugel et al. 2018; Zhang et al. 2019; Lakkaraju and Bastani 2020). As a result the interpretable representations introduced along the first explainers utilising them are still dominating the explainability landscape and are widely used despite possibly being a subpar choice. Specifically, the most popular use case of IRs—in which they are deployed to measure the positive or negative *influence* of each interpretable component (more precisely, information that it encodes) on a black-box prediction of a selected instance—comes with many unaddressed issues. For example, to carry out this *sensitivity analysis*, a random subset of IR elements needs to be "removed" a number of times and the resulting change in the model's prediction quantified, e.g., by the coefficients of a (surrogate) linear model. Most black-box models, however, cannot predict incomplete instances, especially for tabular and image data, in which case this procedure becomes ill-defined and replaced with a proxy operation—such as segment occlusion for images—potentially leading to biased and untrustworthy explanations.

In this paper, we investigate the capabilities and limitations of the most common type of interpretable representations, where presence and absence of interpretable con-

cepts is encoded with a binary on/off vector. We first overview relevant literature and introduce popular IRs for text, image and tabular data in Sect. 2, where we also show example explanations built upon them. This section additionally identifies the core elements, parameterisation and deficiencies of interpretable representations, which facilitates their analytical and experimental investigation. Next, in Sect. 3, we study the influence of suboptimal configuration of IRs and the implications of employing various algorithmic proxies necessary to make them computationally feasible and scalable whenever it is impossible or impractical to directly remove information from the underlying data. We also investigate implicit assumptions such as the *locality* of an explanation, which may prevent its completeness, and the *stochasticity* of the transformation between the original and interpretable domains (and vice versa), which may introduce unnecessary randomness, contribute to volatility and reduce fidelity and soundness of explanations, thereby harming their *veracity* (Sokol et al. 2019; Sokol and Flach 2020a). Our findings are supported by a range of experiments that analyse these factors for quartile-based discretisation and decision tree-based partition of numerical features for tabular data—where information removal is achieved through a random allocation of a feature value to one of these attribute ranges—as well as super-pixel segmentation of images with varying granularity—for which colour-based occlusion is used as an information removal strategy.

In Sect. 4 we examine the lineage and interpretation of influence-based explanations—determined by the coefficients of a linear model in a surrogate explainer setting—with respect to the properties of the underlying interpretable representation for tabular data with numerical features. In particular, we illustrate the limited explanatory capabilities of an IR built upon (unsupervised) discretisation of continuous attributes when paired with Ordinary Least Squares (OLS), i.e., a bare-bones version of LIME, for which an analytical (closed-form) solution is derived in Appendix A. Such explainers can lose the precise encoding of the black-box decision boundary and be (externally) manipulated by altering the distribution of the data sample used to fit the surrogate OLS, both of which undermine reliability of the ensuing explanations. As a solution we propose using *supervised* discretisation algorithms that produce up to *three bins* per numerical feature in addition to employing alternative types of surrogate models—a recommendation supported by a collection of theoretical and experimental results. Specifically, we investigate decision trees, which prove to be particularly suitable in this setting since they can both partition (i.e., discretise) the data space to crate meaningful interpretable concepts, and generate a wide array of appealing explanations such as exemplars, importance scores and counterfactuals (van der Waa et al. 2018; Sokol 2021).

All of these findings allow us to create guidelines for building trustworthy, faithful and algorithmically sound interpretable representations, which we outline in Sect. 5. This collection of insights is a stepping stone towards automatic generation of robust IRs with well-known properties and caveats. It also highlights how state-of-the-art research in fields such as natural language processing, image segmentation and discretisation of tabular data can inform better design of interpretable representations, in addition to discussing the beneficial role of humans in this process. Furthermore, our results demonstrate the importance of developing representative validation criteria and metrics for individual components of explainability algorithms, which is an

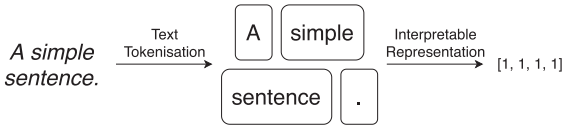
improvement over evaluating only the final, end-to-end explainer (Sokol et al. 2022a; Sokol and Vogt 2024; Small et al. 2023; Xuan et al. 2023. We conclude this paper with Sect. 6, which summarises our findings and discusses future research directions.

2 Interpretable representations

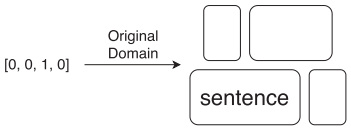
The choice of the interpretable representations and surrogate models used for our analysis is motivated by the popularity of these individual components in the literature. Specifically, LIME (Ribeiro et al. 2016) and RuleFit (Friedman and Popescu 2008) use a surrogate *linear* model to estimate influence of interpretable concepts. Additionally, LIME and SHAP—SHapley Additive exPlanations (Lundberg and Lee 2017)—employ an interpretable representation that encodes *presence and absence* of intelligible concepts to formulate their explanations. Similarly, Friedman and Popescu (2008) experimented with automatic learning of more complex IRs for tabular data by fitting a random forest and extracting rules from it. These logical statements, which capture various concepts, are then used as binary meta-features to train a linear model, thus offering a highly expressive interpretable representation. More recently, Garreau and Luxburg (2020) analysed theoretical properties and parameterisation of vanilla LIME for tabular data, including its interpretable representation and surrogate linear model; however, their work treated the explainer as an end-to-end algorithm and operated under quite restrictive assumptions, e.g., presupposing linearity of the underlying black box.

We focus on interpretable representations of tabular, image and text data given their dominant role in XAI. While the operationalisation of IRs varies across these data types, their machine representation is usually consistent: a binary vector indicating presence (*fact* denoted by 1) or absence (*foil* denoted by 0) of certain human-understandable concepts for a selected data point. The IRs of image and text data are relatively intuitive and share many properties. Images are partitioned into non-overlapping segments called super-pixels, which are then represented in the interpretable binary space as either preserved (i.e., original pixel values) or removed. Similarly, text is split into tokens that can encode individual words, their stems or collections of words, the presence or absence of which is captured by the IR. Tabular data, on the other hand, are more problematic since, first, numerical attributes need to be discretised to create a hyper-rectangle partition of the feature space, followed by a binarisation procedure that for each (now discrete) dimension records whether a data point is located within or outside of the hyper-rectangle selected to be explained.

The interpretable representations of text and images are reasonably easy to generate automatically and (when configured correctly) the meaning of the resulting explanations is relatively accessible to a lay audience—a characteristic that is not necessarily true of tabular data as we will see later. Additionally, the high dimensionality of raw text and image data does not impair their comprehensibility, but it does for tabular data as humans are generally confined to three dimensions given the inherent spatio-temporal limitations of our visual apparatus. Dimensionality reduction for images and text is thus unnecessary and may even be harmful; removal of super-pixels is an ill-defined procedure that results in blank spaces, whereas discarding stop words and



(a) Transformation from the original domain into the interpretable representation $\mathcal{X} \rightarrow \mathcal{X}^\star$.



(b) Transformation from the interpretable representation into the original domain $\mathcal{X}^\star \rightarrow \mathcal{X}$.

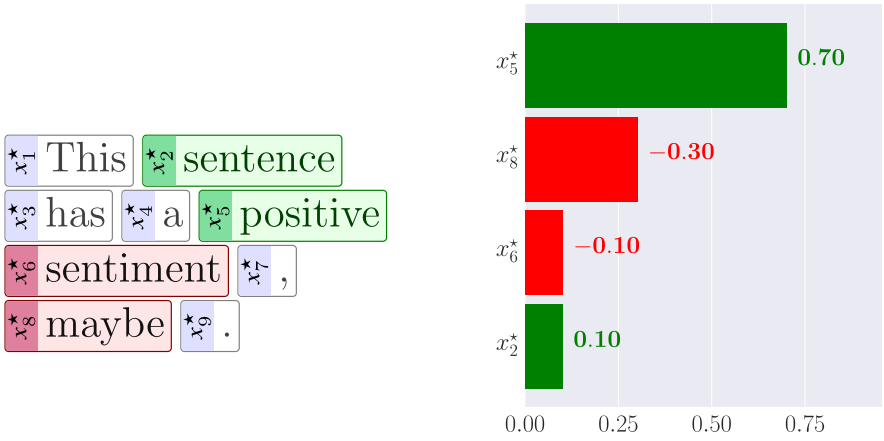
Fig. 1 Depiction of a forward and backward transformation between the original and interpretable representations of text data. Panel **a** shows steps required to represent a sentence as a binary on/off vector; Panel **b** illustrates this procedure in the opposite direction. Both transformations are *deterministic* given a fixed algorithm responsible for text pre-processing and tokenisation

punctuation marks as well as word transformations can be considered as pruning steps that should be incorporated directly into the interpretable representation composition function and executed prior to tokenisation. The process of transforming data from their original domain into an interpretable representation is in most cases defined by the user and built into the (surrogate) explainer. Uniquely for tabular data, however, it can be learnt as part of the *explanation generation* step depending on the surrogate model choice (Sokol et al. 2019; Sokol and Flach 2020b; Sokol 2021). Notably, specifying the foil—i.e., the operation linked to switching off a component of the IR by setting its binary value to 0—may not always be straightforward or even feasible in certain domains, requiring a problem-specific information removal proxy (Mittelstadt et al. 2019).

2.1 Text

The interpretable domain based on presence and absence of tokens in text feels natural and appealing to humans. Individual words and their groups encode understandable concepts and their absence may alter the meaning of a sentence, which arguably reflects how humans comprehend text. A naïve IR can represent text as a bag of words, where each word becomes a token, thereby forgoing the influence of word ordering and the information carried by their co-occurrence. We can easily improve upon that and capture the dependencies between words by including n-gram groupings. Applying other pre-processing steps, e.g., extracting word stems or lemmatisation, can also be beneficial for the human-comprehensibility of such interpretable representations. Machine processing of (natural language) text is a well-established research field (Manning and Schutze 1999) that can be a rich source of inspiration for designing appealing and informative IRs.

Once text is pre-processed and tokenised, it is *deterministically* transformed into the binary interpretable representation. To this end, a sentence is encoded as a Boolean



(a) Words and punctuation marks are the components of the interpretable representation.

(b) Explanations capture influence of tokens when classifying the sentiment of a text excerpt.

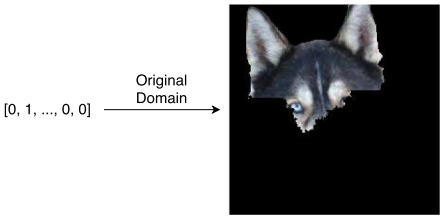
Fig. 2 Example of an influence-based explanation of text with a *bag-of-words* interpretable representation. Panel **a** illustrates a sentence whose (positive) *sentiment* is being decided by a black-box model. The colouring of each token in Panel **a** conveys its influence on the prediction, with Panel **b** depicting their respective magnitudes

vector of length equal to the number of tokens in the IR, where 1 indicates presence of a given token and 0 its absence—see Fig. 1 for a demonstration of this procedure. The original sentence is thus encoded with an all-1 vector. By flipping some components of this vector to 0, we effectively remove tokens from the underlying sentence and create its variations. Notably, the high dimensionality of this representation does not affect the readability of the resulting explanations since altered text cannot have more tokens than the original sentence. Explanations based on token influence can be overlaid on top of text by highlighting each token with a different shade of green (for positive influence) or red (for negative influence), thus expressing their respective influence on the explained class—see Fig. 2 for an example.

2.2 Images

Interpretable representations of image data rely on the same premise: images are algorithmically segmented into super-pixels, often using edge-based methods such as quick shift (Vedaldi and Soatto 2008; Ribeiro et al. 2016). The presence (1) or absence (0) of these segments is manipulated by the underlying binary representation, where an all-1 vector corresponds to the original picture—see Fig. 3 for a reference. However, since a segment of an image cannot be directly removed given that relevant classifiers are unable to handle missing data—in contrast to the equivalent procedure for text IRs—setting one of the interpretable components to 0 is an ill-defined operation. Instead, a *computationally-feasible proxy* is commonly used to hide or discard the information carried by super-pixels; specifically, segments are occluded with a solid

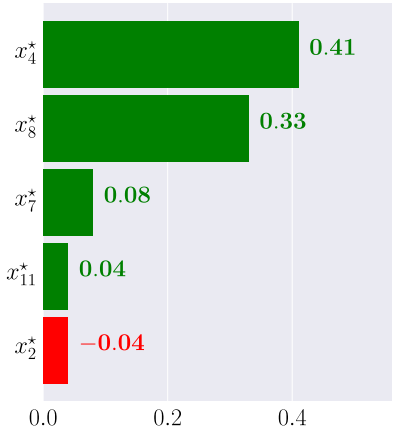
(a) Transformation from the original domain into the interpretable representation $\mathcal{X} \rightarrow \mathcal{X}^\star$.



(b) Transformation from the interpretable representation into the original domain $\mathcal{X}^\star \rightarrow \mathcal{X}$.

Fig. 3 Depiction of a forward and backward transformation between the original and interpretable representations of image data. Panel **a** shows steps required to represent a picture as a binary on/off vector; Panel **b** illustrates this procedure in the opposite direction. Both transformations are *deterministic* given fixed image segmentation and occlusion strategies

(a) Image segments are the components of the interpretable representation.



(b) Explanations show influence of the super-pixels on the selected class prediction (*Eskimo dog*).

Fig. 4 Example of an influence-based explanation of image data with the interpretable representation built upon *segmentation*. Panel **a** illustrates an image that is being classified by a black-box model. The colouring of each super-pixel in Panel **a** conveys its influence on the prediction of a user-selected class (*Eskimo dog* in this case), with Panel **b** depicting their respective magnitudes

colour. For example, LIME uses the mean colour of each super-pixel to mask its content (Ribeiro et al. 2016). Explanations based on such interpretable representations communicate the influence of each image segment on the black-box prediction of a user-selected class as shown in Fig. 4.

This approach comes with its own implicit assumptions and limitations. For one, an edge-based partition of an image may not capture concepts that are meaningful from a human perspective. *Semantic segmentation* or outsourcing this task to the user appears to yield better results (Sokol and Flach 2020b, c), possibly at the expense of automation difficulties. Furthermore, the information removal proxy could be improved by replacing colour-based occlusion of super-pixels with a more meaningful process that better reflects how humans perceive visual differences in images. For example, the content of a segment could be occluded with another object, akin to the procedure proposed by Benchmarking Attribution Methods (Yang and Kim 2019), or retouched in a context-aware manner, e.g., inpainted with what is anticipated in the background, thus preserving the integrity and colour continuity of the explained image. While both of these approaches are intuitive, they are difficult to automate and scale since the underlying operations are mostly limited to image partitions where each super-pixel represents a self-contained and semantically coherent object.

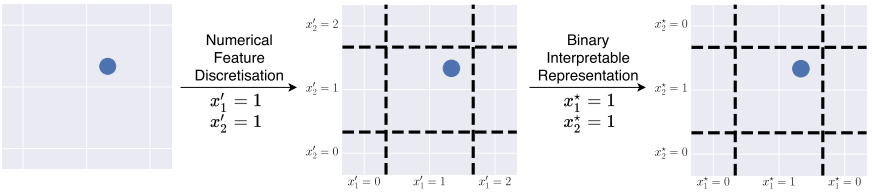
2.3 Tabular data

The raw features used by predictive models trained on images and text—e.g., pixel values and word embeddings—are often difficult to reason about, establishing the need for interpretable representations. In contrast, tabular data may not require an IR to become explainable if their attributes are human-comprehensible from the outset. On the other hand, if the explanation is to answer a specific question—as is the case for images and text—using an interpretable representation may be helpful. Continuing with the theme of investigating concept influence, for tabular data we are interested in how presence and absence of certain *binary* characteristics, which the explained data point exhibits, affect its prediction.

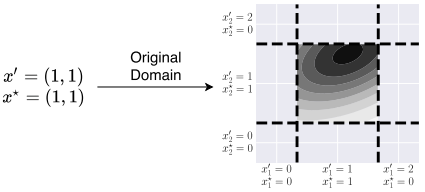
One approach is to treat the specific attribute values of the explained instance as concepts: if a feature value of a data point is identical to the value of the same attribute in the explained instance, the concept is *present* (1), otherwise it is *absent* (0), which procedure becomes the information removal proxy. For example, if the second feature x_2 of the explained instance $\mathring{x}$ is $\mathring{x}_2 = 70$, any instance x whose second attribute has the same value (70) is assigned $x^\star_2 = 1$ in the binary IR, and 0 otherwise, i.e.,

$$x^\star_i = \begin{cases} 1 & \text{if} \quad x_i = \mathring{x}_i, \\ 0 & \text{otherwise.} \end{cases}$$

While this may be appealing for categorical attributes, considering each and every unique value of a numerical feature is cumbersome. Moreover, doing so may not reflect the underlying human thought process: as an example, consider "high sugar content" in contrast to "70 g of sugar per 100 g of a product", with both 0 g and 100 g in the latter case encoded as an *absent* concept in the corresponding IR.

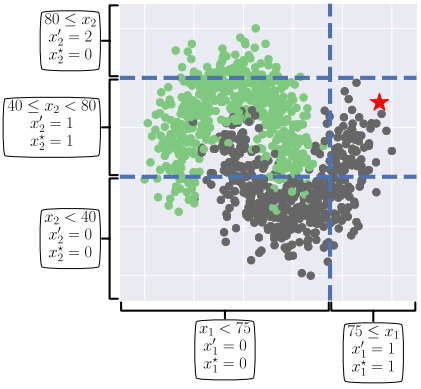


(a) Transformation from the original domain into the interpretable representation $\mathcal{X} \to \mathcal{X}^\star$.

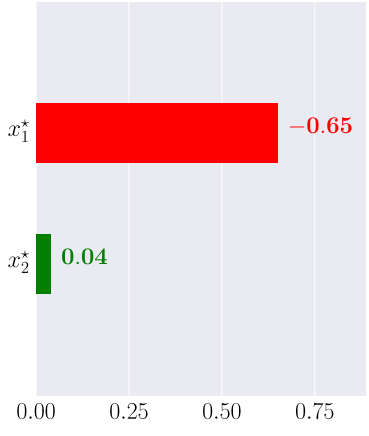


(b) Transformation from the interpretable representation into the original domain $\mathcal{X}^\star \to \mathcal{X}$.

Fig. 5 Depiction of a forward and backward transformation between the original and interpretable representations of tabular data. Panel **a** shows the discretisation and binarisation steps required to represent a data point as a binary on/off vector; Panel **b** illustrates this procedure in the opposite direction. The forward transformation is *deterministic* given a fixed discretisation algorithm (i.e., binning of numerical features); however, moving from the IR to the original domain is *stochastic* since it requires random sampling



(a) Discretised and binarised numerical features become the components of the tabular interpretable representation ($x^\star$).



(b) Explanations show influence of IR components on predicting the *grey* class for the red ⋆ instance and, more generally, the entire $x^\star = (1, 1)$ hyper-rectangle.

Fig. 6 Example of an influence-based explanation of tabular data with the interpretable representation built upon *discretisation* (x') of numerical features followed by *binarisation* ($x^\star$). Panel **a** illustrates a synthetically generated *two moons* toy data set with two numerical features (x_1, x_2) and two classes denoted by grey and green dots. The red ⋆ represents the explained instance x, the dashed blue lines mark the attribute binning (discretisation), x' is the (intermediate) discrete representation, and $x^\star$ encodes the binary IR created for the ⋆ data point. Panel **b** depicts the magnitude of the influence that $x^\star_1 : 75 \leq x_1$ and $x^\star_2 : 40 \leq x_2 < 80$ have on the underlying black-box model predicting the *grey* class for the ⋆ instance (and more broadly any other data point residing within the same hyper-rectangle)

Building on this observation, a natural extension of such a tabular interpretable representation is to discretise numerical features into (meaningful) categorical bins, e.g., $x_2 < 40$, $40 \leq x_2 < 80$ and $80 \leq x_2$, which procedure establishes high-level intelligible concepts (Kotsiantis and Kanellopoulos 2006; Garcia et al. 2012). (This step is not needed for categorical attributes, which already have discrete values.) Since a binary representation allows to encode only two events for each attribute, we need to define the corresponding information removal proxy, which determines the meaning of a discrete concept being *present* or *absent*. Given that the IR should be specific to the explained data point, the binary on/off vector is constructed to indicate whether a feature value of an arbitrary instance x is of the *same* (1 for a *present* concept) or *different* (0 for an *absent* concept) category as the corresponding attribute of the data point selected to be explained $\mathring{x}$. For example, if the second feature x_2 of the explained instance $\mathring{x}$ is $\mathring{x}_2 = 70$, based on the aforementioned bin boundaries, any instance x whose second attribute is within the $40 \leq x_2 < 80$ range is assigned $x^\star_2 = 1$ in the binary IR, and 0 otherwise, i.e.,

$$x^\star_i = \begin{cases} 1 & \text{if } x_i \text{and } \mathring{x}_i \text{ belong to the same numerical bin or} \\ & \text{have the same categorical value,} \\ 0 & \text{otherwise.} \end{cases}$$

This procedure is depicted in Fig. 5. When paired with a surrogate linear model, this interpretable representation allows us to investigate the influence of the designated concepts—i.e., each numerical feature being within the specified range and each categorical attribute being of the particular value—on the black-box prediction of the data point selected to be explained, or more precisely *any* instance located within the same hyper-rectangle—see Fig. 6 for an example.

Notably, such a binary IR of tabular data is specific to the explained instance (more generally, its hyper-rectangle), as was the case for images and text. Nonetheless, the discretisation underlying tabular data can be easily reused for explaining other instances from the same data set. While a common practice (Ribeiro et al. 2016), it should be added that such an approach might affect the faithfulness of the resulting insights since the goal is to produce a *local* explanation of the selected data point, hence the discretisation should explicitly model the characteristics of the explained neighbourhood. Reusing the same discretisation to generate instance-specific IRs for tabular data can be compared to creating a super-pixel partition of a particular image and then reapplying it to other, *unrelated* images, yielding a conceptually flawed interpretable representation. Additionally, selecting the bin boundaries when discretising tabular data, as well as grouping values of categorical attributes, is non-trivial and might bias the explanation similar to the influence of text pre-processing and tokenisation steps or image segmentation and occlusion strategies. Since neither globally nor locally faithful discretisation can capture uniqueness of a black-box decision boundary universally well for an arbitrary data subspace, each explained instance requires bespoke discretisation of the feature space (Sokol et al. 2019).

3 Configuring interpretable representations

Interpretable representations of text data that are based on token removal are appealing from an XAI perspective; they can be easily expanded with relevant natural language processing techniques without the need for any computational proxy, which makes these transformations largely consistent with human intuition in the explainability context. Image data, on the other hand, lack such a seamless operationalisation of IRs, forcing us to use an occlusion-based proxy to discard information from individual segments. This poses several challenges for the trustworthiness, robustness and computational soundness of the resulting explanations as well as their consistency with the explainees' intuition, especially since segmentation algorithms cannot rely on natural separation criteria such as whitespace characters and autonomy of words found in text data. In particular, we are faced with parameterising these IRs based on *segmentation granularity* and *occlusion strategy*, with certain choices possibly exhibiting undesired properties or being ineffective in "removing" super-pixels. The masking colour may impact the veracity of explanations, regardless of the underlying occlusion approach, since these insights rely on an implicit assumption that the black-box model is *neutral* with respect to the occlusion colour, i.e., none of the modelled classes is biased towards it (Mittelstadt et al. 2019). Adjusting the segmentation granularity can also play an important role given high correlation of adjacent super-pixels.

In contrast, tabular data require by far the most complicated interpretable representation whose explanatory meaning may be difficult to grasp due to the counterintuitive process of "switching off" interpretable concepts. Moreover, the underlying information removal proxy requires discretisation of continuous features followed by a binarisation step—a procedure that results in information loss and is sensitive to the selection of binning thresholds. Given its significance, the parameterisation of both image and tabular IRs should be explicitly optimised based on clearly defined objectives that appreciate the uniqueness of the problem at hand and recognise interpretable representations as independent entities and vital components of (surrogate) explainers (Sokol and Vogt 2024). Out of these three IRs, the one for text has the advantage of allowing the tokens to be *truly* removed from a sentence (although this is more of a property of the underlying predictive model rather than the interpretable representation itself). Specifically, text classifiers are more flexible and do not assume input of a fixed size, while vision models cannot handle missing pixels and tabular predictors usually require all features to be present. Investigating the effects of text pre-processing and tokenisation on the quality of the corresponding IRs is outside of the scope of this paper since it is a multifaceted endeavour, a narrow study of which may not provide comprehensive insights given the sheer quantity and diversity of available techniques.

The interpretable representations of image and text data discussed here are *implicitly local*—they are intended (and possibly valid) only for the data point (sentence or image) for which they were created—whereas the scope of the tabular IR is more ambiguous. Another property that the former two IRs have and the latter lacks is *determinism* of the underlying representation change (within the scope of a single instance) as demonstrated by Figs. 1, 3 and 5. Transforming images and text between the two domains only requires memorising the structure or skeleton of the explained data point: adjacency of segments and their original pixel values for images (assum-

ing that the segmentation and occlusion strategies are fixed), and order of tokens as well as their pre-processing for text. Tabular data IRs, however, lack this one-to-one correspondence between an instance and its interpretable representation—due to the aforementioned information loss caused by the discretisation and binarisation steps—which can only be restored by mapping each data point to its IR coordinates and storing this correspondence table for future retrieval (Sokol et al. 2019). While this workaround is possible when starting with an instance in the original representation, it cannot overcome stochasticity when we are only given the IR encoding of a data point. Notably, this property helps to guarantee uniqueness of explanations, which is important for their stability, hence preserving explainees' trust (Rudin 2019; Sokol and Flach 2020a, b).

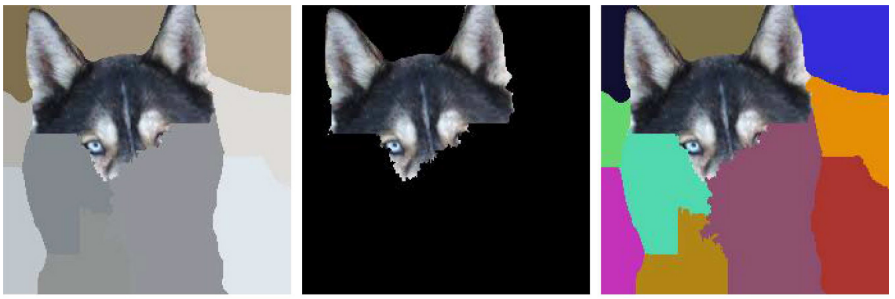
3.1 Occlusion-based interpretable representations of images

Occlusion-based interpretable representations of images are parameterised by *segmentation granularity* and *colouring strategy*. The former allows the explanation to capture a desired level of detail, while the latter is used as a proxy for removing the content of super-pixels to hide the information they carry from a black box. The exact influence of these two properties on the resulting explanations therefore needs to be uncovered to inform the design of robust explainability techniques with well-understood behaviour. For example, consider the **mean-colour occlusion** used by the popular LIME explainer (Ribeiro et al. 2016), which for some image partitions and super-pixel colour distributions may have undesired effects that undermine the utility of the occlusion procedure. With this approach, segments that have a relatively *uniform colour gamut* may, effectively, be impossible to remove; this is especially common for fragments that are in the background or out of focus, e.g., bokeh and depth-of-field effects. *Segmentation granularity* is also important: the smaller the segments are, the more likely it is that their colour composition is uniform given the "continuity" of images, i.e., high correlation of adjacent pixels. This *mosaic* or *blurring* effect is depicted in Fig. 7 for three different super-pixel granularity settings, showing how much of discriminative information is preserved in each case despite "removing" the content of *all* the segments.

Occluding each image fragment with a different colour manifests another issue, namely the *preservation of super-pixel contours*. This effect can be observed in Fig. 8a and c respectively for the mean and random-patch occlusion strategies. Notably, whenever the segmentation coincides with objects' edges or regions of an image where colour continuity is not preserved—which is common for edge-based segmenters—replacing super-pixels with their mean or a random colour causes (slight) colour variations between adjacent segments. These artefacts emphasise edges in a (partially) occluded image that may at times convey enough information for a black-box model to correctly recognise its class; for example, see Fig. 8a and c where despite replacing all the super-pixels but #4 with mean and random colours respectively, the model predicts the original class with 78% and 56% probability down from 84% for the unaltered photo. Since most of these issues are consequences of using the random-patch or mean-colour occlusion, it may seem that fixing a single masking colour for

(a) 17-segment partition. (b) 51-segment partition. (c) 71-segment partition.

Fig. 7 *Mosaic* or *blurring* effect observed for the mean-colour occlusion strategy when "removing" all of the super-pixels with segmentation granularity increasing in size. The image was split into **a** 17, **b** 51 and **c** 71 super-pixels with the SLIC algorithm (Achanta et al. 2012), which performs k-means clustering in the colour space



(a) Mean-colour occlusion. (b) Black occlusion. (c) Random-patch occlusion.

Fig. 8 Image occlusion strategy influences the resulting explanations. The picture shown in Fig. 4a is classified by a black box as *Eskimo dog* with 84% probability. Based on 11 super-pixels, the **a** mean-colour occlusion of all the segments but one results in 78%, **b** black occlusion in 15% and **c** random-patch occlusion in 56% probability of the same class. These results show that the mean occlusion strategy cannot effectively remove information from this particular image; the random-patch approach preserves segment edges, which are quite revealing in this case; and the black occlusion is relatively good at removing the content of super-pixels

all of the segments would eradicate some of these problems. Such an approach hides the edges between occluded super-pixels and removes their content instead of just "blurring" the image, which is the case for the mean-colouring strategy. However, the edges between occluded and preserved segments remain visible—see Fig. 8b, which depicts using black occlusion that yields only 15% probability of the original class—and choosing a neutral colour that does not bias the explanations—e.g., the relation between blue and objects such as the sky or bodies of water—remains an open question. Notably, the problem of selecting a *reference point* or *foil* for explanations is not unique to occlusion-based interpretable representations of images and it is particularly problematic for tabular data (Mittelstadt et al. 2019).

Experiment Setup To capture the influence of these characteristics—**colour uniformity**, **segmentation size** and **edge visibility**—on the effectiveness of the information

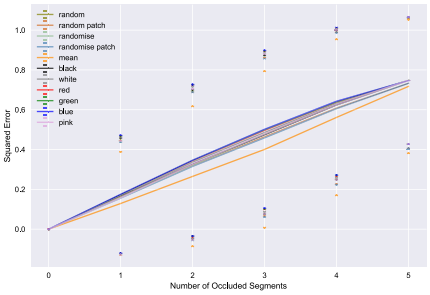
removal proxy, we design and execute ablation studies. Empirically quantifying the effect of the colouring strategy and segmentation granularity on the ability of a black box to consistently predict a (partially distorted) image allows us to better understand the significance of these choices. To this end, we use images from the ImageNet (Deng et al. 2009) validation set that are square and no smaller than 256×256 pixels. Next, we resize them to 256×256 pixels and segment them with the SLIC algorithm (Achanta et al. 2012)—which performs k-means clustering in the RGB (Red, Green, Blue) colour space—using the implementation provided by scikit-image (van der Walt et al. 2014). Since some of the images cannot be segmented into the desired number of super-pixels, only their subset (whose size is given in Fig. 9) is used for the study. For all of the experiments, our black box is the pre-trained *Inception v3* neural network distributed with PyTorch (Paszke et al. 2019). The study is implemented using the FAT Forensics Python package (Sokol et al. 2020, 2022b), with the experiment code published on GitHub[1].

Specifically, segment occlusion is done with the following selection of colouring strategies denoted in the RGB space:

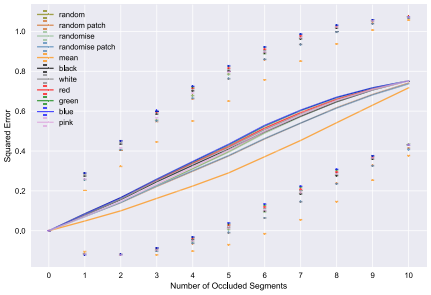
black	(0, 0, 0);
white	(255, 255, 255);
red	(255, 0, 0);
green	(0, 255, 0);
blue	(0, 0, 255);
pink	(255, 192, 203);
mean	each super-pixel is replaced with a solid patch of the mean colour computed for the pixels residing within this segment;
random	a *single* random colour, sampled uniformly from the RGB space, is used to occlude all super-pixels across all experiments for a single image and fixed segmentation size;
random patch	a *separate* random colour, sampled uniformly from the RGB space, is used to occlude each individual super-pixel across all experiments for a single image and fixed segmentation size;
randomise	a *single* random colour, sampled uniformly from the RGB space, is generated for each individual occlusion pattern; and
randomise patch	a *separate* random colour, sampled uniformly from the RGB space, is generated for each individual super-pixel.

The test images are partitioned into 5, 10, 15, 20, 30 and 40 regions to capture the influence of the segmentation granularity on the IR—these tiers are visualised in separate panels of Fig. 9. For a fixed number of segments, we iterate over the quantity of occluded super-pixels from 0 to all of the partitions (x-axes in Fig. 9), randomising the occlusion pattern 100 times at each step. We apply this procedure to all of our test images, separately for every colouring strategy. Finally, we measure the influence of each occlusion strategy and segmentation granularity by calculating the *squared error*—$SE = (y_i - \hat{y}_i)^2$—between the probability of the top class predicted for the unaltered image and the prediction of the same class when the image is (partially)

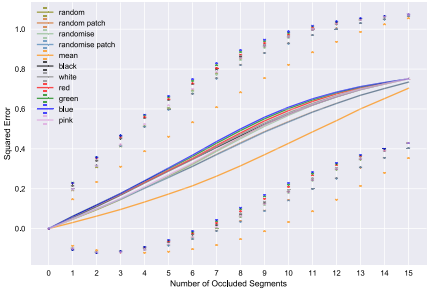
[1] https://github.com/So-Cool/bLIMEy/tree/master/DAMI_2024



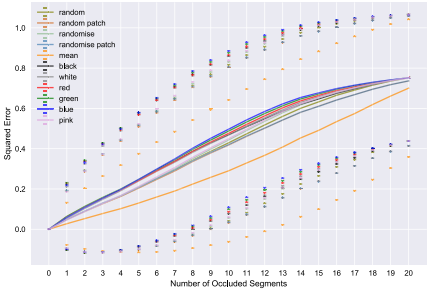
(a) 5-segment partition run on 749 images.



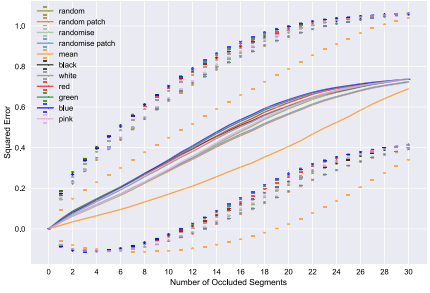
(b) 10-segment partition run on 873 images.



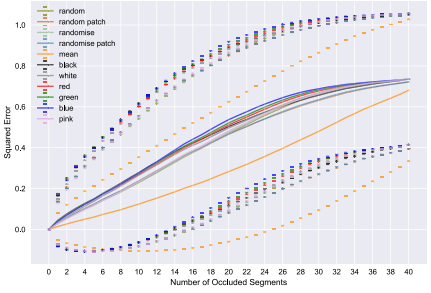
(c) 15-segment partition run on 801 images.



(d) 20-segment partition run on 884 images.



(e) 30-segment partition run on 710 images.



(f) 40-segment partition run on 605 images.

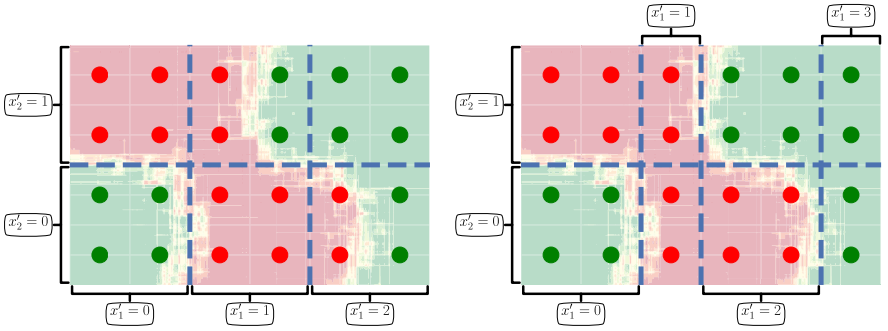
Fig. 9 Squared error (y-axis) calculated between the top prediction of an image (probability estimate) and predictions of the same class when incrementally occluding a higher number of random super-pixels (x-axis) with a given colouring strategy (legend). The segmentation is based on the SLIC algorithm (Achanta et al. 2012); the number of images used for each experiment is recorded in the captions above; a random sample of 100 occlusion patterns was generated for each step with a fixed number of super-pixel occlusions. The curves capture the mean of individual squared errors, with their standard deviation depicted by horizontal bars of the same colour—a lower value indicates that the black box is better able to predict the top class despite information removal. The panels show that the mean occlusion strategy is not as effective at hiding information from the black box as using a single, random or randomised colour to the same end. The plots also reveal that when an image is split into more segments, the ineffectiveness of the mean-colouring approach gets magnified due to the increased colour uniformity of individual super-pixels (see Fig. 7 for an example of this phenomenon)

occluded. We aggregate these scores by computing their mean and standard deviation (y-axes in Fig. 9), a low value of which indicates that the model can still predict the data point despite a distortion.

Occlusion Colour Figure 9 provides clear evidence that the *mean* occlusion strategy behaves unlike any other approach, including all of the methods based on *random* colour selection; additionally, there is no perceptible difference between the remaining colouring strategies—their squared error curves are bundled together. More precisely, the lower metric value for the *mean* technique indicates that it is not as effective at removing class-identifying information from images as any other occlusion strategy that we tested. Intuitively, the reason for this behaviour is the aforementioned *blurring* or *mosaic* effect depicted in Fig. 7. This phenomenon becomes especially pronounced when images are segmented into smaller super-pixels, as having more of them for a fixed image size makes each partition more uniform with respect to the colour of its individual pixels—the increasing separation of the squared error curve for the *mean* strategy when moving from 5 (Fig. 9a) to 40 (Fig. 9f) segments. Additionally, Fig. 9 illustrates the consequences of preserving contour lines between segments when occluding them with patches of different colour—an example of which is visualised in Fig. 8c. This behaviour is captured by the *random patch* and *randomise patch* strategies, both of which exhibit a lower squared error than any other technique based on a single, possibly random, occlusion colour; nonetheless, this effect appears negligible across our experiments.

Segmentation Granularity By inspecting each panel of Fig. 9, we can see that the granularity of segmentation directly affects the *mean*-colour occlusion strategy—the aforementioned separation between the squared error curve of the *mean* approach and every other curve. The behaviour of all the fixed-colour approaches, on the other hand, is very similar for any number of segments regardless of the exact occlusion colour (including its random selection)—these squared error curves are clustered together in Fig. 9. Notably, this observation also applies to the *random-patch* and *randomise-patch* strategies, which methods reveal segment boundaries and can be very volatile given their random assignment of the occlusion colour to each individual super-pixel. Both of these insights offer clear evidence that using the *mean* colouring should be avoided in occlusion-based interpretable representations of images. Figure 9 substantiates our observation that this occlusion strategy becomes less effective as the number of super-pixels increases since relatively small segments tend to have a uniform colour distribution because of the pixel *continuity*—i.e., high correlation of neighbouring pixels—making them visually similar to their respective *mean*-coloured patches. Additionally, this undesired phenomenon may affect images that have an out-of-focus background, e.g., portraits, since their blurry regions will be difficult to remove with the *mean*-colour occlusion strategy.

Snow in the Background Observing the influence of each algorithmic component on the effectiveness of occlusion-based interpretable representations for images, hence explainers built upon them such as LIME, has prompted us to re-examine some of the conclusions drawn by Ribeiro et al. (2016, § 6.4). In particular, the inability of the mean occlusion strategy to discard information—especially so for uniform colour



(a) *Distribution*-aware discretisation. **(b)** *Class*-aware (predictions) discretisation.

Fig. 10 Discretisation is the main building block of interpretable representations for tabular data with numerical attributes. It can either be learnt based on data features alone—an *unsupervised* approach shown in Panel **a**—or additionally consider their black-box predictions—a *supervised* approach shown in Panel **b**. These examples use a synthetic toy data set with two numerical features and 24 evenly-spaced instances whose class—red or green—is given by each point's colour; the background shading indicates the class prediction provided by the underlying (black-box) model; x'_1 and x'_2 encode the bin assignment for the first and second feature respectively

patches and high segmentation granularity—casts doubts on the veracity of explanations generated in the famous study of snow (visible in the background of a picture) biasing predictions of a model deciding between a wolf and an Eskimo dog. The mosaic effect resulting from this removal proxy—captured by Fig. 7—and the overall ineffectiveness of this approach—exemplified by Fig. 8—demonstrate acute problems of the downstream explanations in such a classification scenario. Specifically, consider segments of this image showing snow, which are replaced with their respective mean colours, thus producing off-white patches that still resembles snow; for example, compare the bottom-left and the bottom-right super-pixels in Figs. 4a and 8a. These almost visually indistinguishable alterations are likely to prevent the explainer from capturing the change in the probability predicted by the model under investigation, as shown by our experiments and exemplified in Fig. 8, affecting soundness of the resulting LIME explanations. While such techniques may generate insights into black-box classifiers and help us to uncover spurious correlations, if not tuned to the problem at hand, stock explainers may cause more harm than good (Rudin 2019).

3.2 Discretisation-based interpretable representations of tabular data

The two factors that influence the creation of a binary interpretable representation of tabular data are the *instance* selected to be explained—which determines the reference hyper-rectangle—and the ability of the *discretisation* algorithm to (locally) approximate the black-box decision boundary—which dictates its faithfulness (Sokol and Flach 2020a). Only the latter property, however, is controlled algorithmically and can either be explicitly *global*, i.e., learnt with respect to a whole data set, or *local*, thus focusing on a specific neighbourhood. Furthermore, each variant can either observe

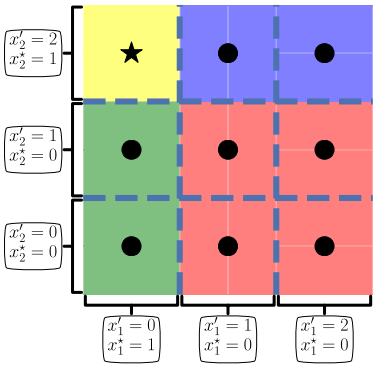


Fig. 11 Some hyper-rectangles $x' = (x_1', x_2')$ created through discretisation become indistinguishable in the binary interpretable representation $x^\star = (x_1^\star, x_2^\star)$ of tabular data. The $\star$ symbol indicates the explained instance and the background shading marks unique binary encodings $x^\star = (x_1^\star, x_2^\star) \in \{(0, 0), (0, 1), (1, 0), (1, 1)\}$. This example uses a synthetic toy data set with two numerical features and nine evenly-spaced instances, one per hyper-rectangle created by the discretisation step

just the *data distribution*, or additionally take into account their *black-box predictions*, presenting us with two distinct discretisation approaches:

distribution-aware (unsupervised) based on the density of data in the local or global region chosen to be explained, e.g., quantile discretisation (Fig. 10a); and

class-aware (supervised) partitioning data according to a black-box decision boundary confined within the local or global region chosen to be explained (Fig. 10b).

While the scope and supervision level of a discretisation are the two main properties that affect the quality of a tabular interpretable representation, other aspects of this process can be considered as well, a summary of which can be found in relevant surveys (Kotsiantis and Kanellopoulos 2006; Garcia et al. 2012).

Information Loss Discretisation and binarisation procedures tend to be many-to-one mappings. The intermediate *discrete* representation of a tabular IR uniquely encodes each created hyper-rectangle—see the (x_1', x_2') coordinates in Fig. 11—*explicitly* trading off precision for sparsity and intelligibility. However, the ensuing *binarisation* step *implicitly* loses information whenever a categorical feature has more than two unique values or a numerical attribute is partitioned into more than two intervals as shown by the background shading and the $(x_1^\star, x_2^\star)$ coordinates in Fig. 11. Recall that for each of these binary IR features, 1 is assigned to the partition that contains the explained data point and 0 to all the other intervals, effectively making the latter categories *indistinguishable*.

The impossibility to discern data points belonging to different hyper-rectangles in the binary interpretable representation is particularly detrimental to the IR's ability to capture the intricacy of the black-box decision boundary. While the underlying discretisation may have closely approximated its shape, these details can be lost when

transitioning into the binary space, especially if the decision boundary runs across hyper-rectangles that are merged in this process. For example, consider the discretisation shown in Fig. 10b, assuming that the explained instance resides in the $x' = (1, 1)$ hyper-rectangle—top row, second column from the left. In the binary representation, the remaining top-row hyper-rectangles $(0, 1)$, $(2, 1)$ and $(3, 1)$ would be bundled together—akin to the process depicted by the background shading in Fig. 11—thus forfeiting the information that the first one belongs to the red class and the latter two to the green class. A similar grouping will happen in the bottom row, where $(0, 0)$, $(2, 0)$ and $(3, 0)$ will be merged. Observing this redundancy, nonetheless, can help us in search of a better mechanism to build tabular interpretable representations.

Faithfulness Since the predominant role of *local surrogate explainers* is to approximate and simplify the behaviour of a black box near a selected instance, *local* and *class-aware* discretisation should be preferred. This procedure is a stepping stone towards representing interpretable concepts that are coherent with predictions of the underlying model, thus producing faithful and appealing insights. However, to the best of our knowledge, class-aware (supervised) discretisation approaches are absent in the explainability literature. Computationally, their objective can be expressed as *maximising the purity or uniformity* of each hyper-rectangle with respect to the black-box predictions of data that it encloses—this applies to regression as well as probabilistic and crisp classification models (Kotsiantis and Kanellopoulos 2006; Garcia et al. 2012). In particular, if the underlying task is crisp classification, we can use the *Gini impurity* ($\mathcal{L}_{\text{G}}$) defined in Eq. 1, where H_i is a set of data points and their labels (x, y) residing within the i^{th} hyper-rectangle and C is the set of all the unique labels c.

$$\begin{aligned} \mathcal{L}_{\text{G}}(H_i) &= \sum_{c \in C} p_{H_i}(c) \times \left(1 - p_{H_i}(c)\right) \\ p_{H_i}(c) &= \frac{1}{|H_i|} \sum_{(x,y) \in H_i} \mathbb{1}_{y=c} \end{aligned} \tag{1}$$

On the other hand, when the task is regression or probabilistic classification (the formula applies separately to each individual class in the latter case), we can use the *Mean Squared Error* ($\mathcal{L}_{\text{MSE}}$)—defined in Eq. 2—to quantify numerical uniformity of black-box predictions in each hyper-rectangle.

$$\begin{aligned} \mathcal{L}_{\text{MSE}}(H_i) &= \frac{1}{|H_i|} \sum_{(x,y) \in H_i} (y - \overline{y}_{H_i})^2 \\ \overline{y}_{H_i} &= \frac{1}{|H_i|} \sum_{(x,y) \in H_i} y \end{aligned} \tag{2}$$

When combining scores of multiple hyper-rectangles to assess the overall quality $\mathcal{Q}$ of an interpretable representation, we opt for a weighted average of individual scores $\mathcal{L}$ to account for the (possibly unbalanced) distribution of data points across these

segments—see Eq. 3.

$$\mathcal{Q} = \frac{1}{\sum_{H_i} |H_i|} \sum_{H_i} |H_i| \times \mathcal{L}(H_i) \tag{3}$$

We use this formulation to evaluate the faithfulness of a quartile-based tabular IR (distribution-aware discretisation) used by the popular LIME explainer (Ribeiro et al. 2016), and compare it with a simple tree-based IR (class-aware discretisation), testing both approaches in a global and local variant.

Quartile IR This interpretable representation is based on quartile discretisation of continuous features. The partition of the data space can either be global or local—i.e., with respect the entire data set or its subset—nonetheless each individual instance receives a distinct IR due to the binarisation step that follows. For each data point, global IRs are derived based on a shared discretisation computed for the entire data set. Local IRs, on the other hand, are composed separately for each instance in the data set based on samples located in its neighbourhood, which are used for the discretisation step. We rely on the formula given by Eq. 3 to evaluate the faithfulness of both steps: discretisation and binarisation. For global discretisation this validation is performed on the entire data set. All the other approaches are assessed on a subset of data that, centred around the explained data point, is within the radius of 30% of the maximum Euclidean distance computed between any two instances in the data set, which simulates locality of the explanation.

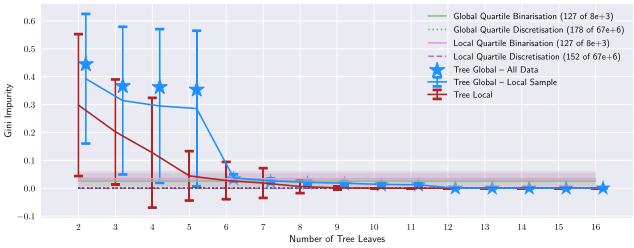
Tree-based IR This interpretable representation is based on a partition of the feature space learnt by a tree model. Its candidature stems from observing similarity between the tree learning objective and the proposed faithfulness evaluation metrics. Global analysis is performed by computing purity of the hyper-rectangles created by a tree fitted to the entire data set and validated on this training data as well as a local sample generated separately for each instance in the data set, which is a fair comparison given that the quartile-based IR can also access the whole data set. The local IR faithfulness, on the other hand, is calculated independently for each instance in the data set by learning a tree model on a subset of data that, centred around the explained instance, is within the radius of 30% of the maximum Euclidean distance computed between any two instances in the data set, with the same data subset used to evaluate the quality of the resulting hyper-rectangles.

Experiments We compare faithfulness of these two tabular interpretable representations on four real-life data sets, two of which are classification and the other two regression problems:

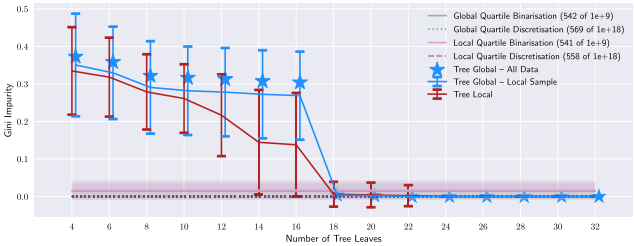
- wine recognition[2] (classification);
- breast cancer Wisconsin diagnostic[3] (classification);

[2] https://archive.ics.uci.edu/ml/datasets/wine

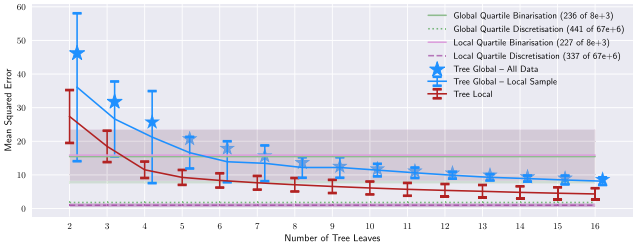
[3] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)



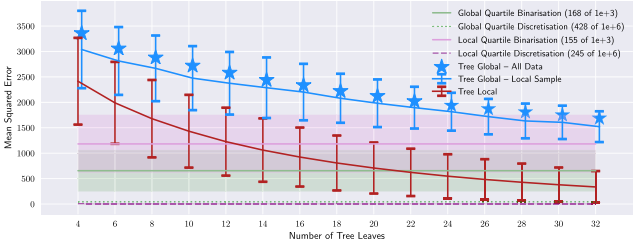
(a) Weighted average of the *Gini impurity* computed for the *wine* data IRs.



(b) Weighted average of the *Gini impurity* computed for the *cancer* data IRs.



(c) Weighted average of the *mean squared error* computed for the *housing* data IRs.



(d) Weighted average of the *mean squared error* computed for the *diabetes* data IRs.

Fig. 12 Interpretable representations based on decision trees achieve better faithfulness of hyper-rectangles (y-axes, lower is better) with fewer encodings (x-axes, small jitter added for readability), i.e., they are more flexible and expressive. This property is measured with a weighted average (over IR hyper-rectangles) of Gini impurity for classification tasks and mean squared error for regression or probabilistic classification tasks. The number of unique encodings generated by quartile-based IRs is constant for a data set and it is displayed in the legend, shown as the maximum number of encodings used, out of the theoretical limit supported by the IR; for tree-based IRs, on the other hand, it is equivalent to the number of leaves, which is recorded on the x-axes. Panels **c** and **d** do not capture the tree width at which this IR outperforms the global and local quartile discretisation steps alone, which is 80 or 64 (compared to 441 or 337) and 224 or 112 (compared to 428 or 245) respectively for the *housing* and *diabetes* data sets

– Boston house prices[4] (regression); and
– diabetes[5] (regression).

Using these data, we evaluate *quartile*- and *tree-based* IRs in two variants:

global where a single discretisation is generated for all the data points (and, in case of the quartile method, followed by creation of instance-specific binary IRs); and

local where a collection of distinct discretisations is composed separately for each individual data point (and then binarised based on the same instance for the quartile method).

In addition to evaluating the quartile-based binary interpretable representation, we compute faithfulness of the intermediate discretisation step to facilitate an in-detail comparison. The results of our experiments are depicted in Fig. 12, which reveals that tree-based IRs require a fraction of the expressiveness—i.e., unique encodings in the binary interpretable space—used by the quartile-based IRs to achieve a comparable level of hyper-rectangle faithfulness, especially in the *local* variant. This can be understood as tree-based interpretable representations being better able to capture the intricacies of the underlying (black-box) labelling mechanism with less complexity to the benefit of the ensuing explanations.

More precisely, Fig. 12 displays the impurity of interpretable representations achieved for a range of different tree widths, with the x-axes showing the limit imposed on the number of leaves. In each case, the leaves number can be compared to the number of unique hyper-rectangles generated by the discretisation and binarisation steps of the corresponding quartile-based IR. The y-axes, on the other hand, depict weighted Gini impurity or mean squared error, respectively for classification and regression tasks, computed for all the hyper-rectangles of each individual IR (Eq. 3). The dotted green and dashed pink lines labelled as "global quartile discretisation" and "local quartile discretisation" are the measure of impurity for the quartile discretisation step that underlies this type of an IR. The solid green and pink lines surrounded by shading—marked as "global quartile binarisation" and "local quartile binarisation"—correspond to the mean and standard deviation of the hyper-rectangle impurity scores computed for the global and local variants of quartile-based binary IRs (i.e., *discretisation* followed by *binarisation*) for each individual instance in a data set. Equivalent measurements are taken for the global and local tree-based IRs for a range of tree widths: "tree global" depicted in blue (the $\star$ symbol corresponds to evaluation on the entire data set whereas the curve captures the same measurement for the neighbourhood of each instance) and "tree local" plotted using the red curve, with the error bars denoting the standard deviation. In all of the plots, a lower score on the y-axes—capturing weighted faithfulness of an IR—is better.

The pair of numbers placed in brackets next to the quartile discretisation and binarisation labels in the legend of each plot communicates the maximum number of distinct hyper-rectangles for the former, and their binarisation-driven combinations for the latter, that are being used by the validation data, out of all the possible unique values that,

[4] https://archive.ics.uci.edu/ml/machine-learning-databases/housing

[5] https://www4.stat.ncsu.edu/~boos/var.select/diabetes.html

respectively, the quartile discretisation and its binarisation can theoretically encode. These quantities are directly comparable to the width of trees—recorded on the x-axes of the plots—used to partition the feature space to compose the tree-based IRs. Given a lack of a black-box model, whose predictions should be used to capture the distribution of the target variable within each hyper-rectangle, we instead utilise the ground truth provided with the aforementioned data sets since this proxy does not affect the validity of our experiments in any way. In summary, Fig. 12 illustrates that interpretable representations created with decision trees are more pure (i.e., uniform) than their quartile-based alternatives, therefore they are superior at capturing the complexity of the underlying labelling mechanism, whatever it may be. Furthermore, they achieve better performance with just a fraction of the encodings required by the other method, i.e., they are more expressive because of the elaborate (class-aware) mechanism used by decision trees to partition and merge a feature space.

4 Linking interpretable representations with surrogate models: analysis of tabular data explainability

Interpretable representations are paired with transparent predictive models to form surrogate explainers (Sokol et al. 2019). Linear models are a common choice that allows to capture the influence of human-comprehensible concepts on black-box predictions (Friedman and Popescu 2008; Ribeiro et al. 2016), in which case such explanations—determined by the coefficients of the underlying surrogate linear classifier—are subject to assumptions and limitations of these models. In particular, such explanatory insights can be deceiving when the target variable is *non-linear* with respect to data features, the attributes are *co-dependent* or *correlated*, and the feature values are *not normalised* to the same range (Sokol et al. 2019; Sokol and Flach 2020b). Intuitively, the first two properties may not hold for high-level interpretable representations since their components are highly inter-dependent—e.g., adjacent image segments, neighbouring words and bordering hyper-rectangles—therefore the resulting explanations can misrepresent the possible relations between these concepts and the behaviour of the explained black box. Friedman and Popescu (2008) addressed some of these concerns by using logical rules extracted from random forests as the binary interpretable concepts, which they then modelled with a linear predictor; however, the overlap between these rules still violates the feature independence assumption.

In addition to these limitations, linear models are inherently *incompatible* with the interpretable representation of tabular data introduced in Sect. 2.3. Recall that the information loss suffered when transitioning from the discrete into the binary representation partially forfeits the preceding effort of the discretisation step to faithfully capture the black-box decision boundary. The undesired side effect of this procedure adversely affects the weights of the linear model trained on top of such a binary IR. This can be observed by deriving an analytical solution to *ordinary least squares* in this specific setting, which is presented in Eq. 4 for a toy example with two numerical features similar to the scenario shown in Fig. 13. In this case, the coefficients $\Theta_{\mathbf{W}}$ of the OLS model depend on:

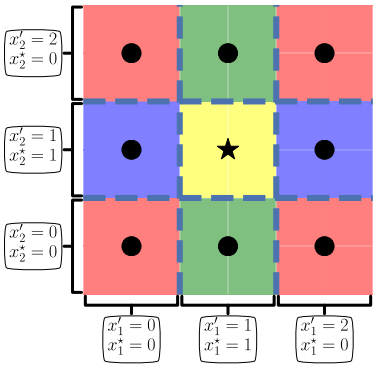


Fig. 13 Example of a discrete representation $x' = (x'_1, x'_2)$ and binary IR $x^\star = (x^\star_1, x^\star_2)$ of tabular data. The $\star$ symbol represents the explained instance. (Refer to Fig. 11 for a description of the toy data set used here)

1. the number of data points w_{ij} in the hyper-rectangles determined by the $x^\star = (i, j)$ coordinates of the binary interpretable representation; and
2. the average black-box prediction $\overline{y}$ in various IR partitions denoted by $\mathcal{W}_{ij}$, with the set of all the data points given by $\mathcal{W}$.

This formulation can be generalised to an arbitrary number of dimensions spanning numerical and categorical features, and it is applicable to regressors as well as crisp and probabilistic black-box classifiers. The derivation of this result is outlined in Appendix A.

$$\Theta_{\mathbf{W}} = \begin{bmatrix} 1 & \frac{w_{11}+w_{10}}{\sum w_{ij}} & \frac{w_{11}+w_{01}}{\sum w_{ij}} \\ 1 & 1 & \frac{w_{11}}{w_{11}+w_{10}} \\ 1 & \frac{w_{11}}{w_{11}+w_{01}} & 1 \end{bmatrix}^{-1} \times \begin{bmatrix} \overline{y}_{\mathcal{W}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \end{bmatrix} \tag{4}$$

This outcome allows us to draw conclusions about the meaning of the interpretable concept influence given by the coefficients of a linear surrogate when the intercept is modelled (red and blue shading in Eq. 4), and without it (blue shading). In particular, the influence of interpretable concepts is *solely* based on:

– **the proportion determined by the number of the data points** residing in the explained hyper-rectangle ($\mathcal{W}_{11}$) divided by the count of points located in the hyper-rectangles aligned with the explained hyper-rectangle along every axis, i.e., $\mathcal{W}_{11} \cup \mathcal{W}_{10}$ for the first feature and $\mathcal{W}_{11} \cup \mathcal{W}_{01}$ for the second attribute; and
– **the average value predicted by the explained black box** in the latter two subspaces—$\mathcal{W}_{11} \cup \mathcal{W}_{10}$ and $\mathcal{W}_{11} \cup \mathcal{W}_{01}$—scaled appropriately when the intercept is modelled.

For example, consider Fig. 13 where $x^\star_1$ denotes the first binary interpretable feature and $x^\star_2$ the second. In this case, $\mathcal{W}_{11}$ is the yellow hyper-rectangle; $\mathcal{W}_{11} \cup \mathcal{W}_{10}$ is the union of the yellow and green hyper-rectangles; and $\mathcal{W}_{11} \cup \mathcal{W}_{01}$ is the combination

of yellow and blue hyper-rectangles. Finally, $\overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}}$ is the average prediction in the vertical green and yellow column, and $\overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}}$ is the average prediction in the horizontal blue and yellow row.

When modelled, the intercept value is additionally determined by:

- *the proportion* given by the number of data points in the hyper-rectangles aligned with the explained hyper-rectangle along every axis, divided by the total number of data points; and
- *the average* value predicted by the black box for all the data points.

Intuitively, the instances not aligned with the explained hyper-rectangle—the red blocks in Fig. 13—are assigned the $x^\star = (0, 0)$ coordinates in the binary interpretable representation, therefore they cannot contribute to the feature coefficients of a linear model, just the intercept. This can be easily seen with the $g(x^\star; \Theta) = \sum_{i=0}^{n} \Theta_i x_i^\star$ formula, where $x_0^\star = 1$ is the phantom feature and the remaining data features $x_1^\star, \ldots, x_n^\star$ are all 0; therefore, these instances can only influence the intercept coefficient Θ_0.

An important insight uncovered by our results is **partial insignificance of the discretisation quality** given a fixed number of data points placed in the identified collections of relevant hyper-rectangles. Using this property we can *manipulate* the explanation by altering the number of data points in meaningful partitions, with the discretisation faithfulness having relatively minor influence. For example, consider the two discretisations depicted earlier in Fig. 10, assuming that the explained hyper-rectangle is $x' = (1, 1)$ for both panels, and that the $x' = (1, 0)$ and $x' = (1, 1)$ partitions in Fig. 10b have *three* additional data points each. In this scenario, when modelling the influence of interpretable components without the intercept, the only difference between these two cases are the black-box predictions of the instances placed in the expanded hyper-rectangles, i.e., $x' = (1, 0)$ and $x' = (1, 1)$, since:

Figure 10a $w_{11} = 4, w_{01} = 4{+}4 = 8$ and $w_{10} = 4$, leading to $\frac{w_{11}}{w_{11}+w_{10}} = \frac{4}{4+4} = \frac{1}{2}$ and $\frac{w_{11}}{w_{11}+w_{01}} = \frac{4}{4+8} = \frac{1}{3}$; and

Figure 10b $w_{11} = 2{+}3 = 5, w_{01} = 4{+}4{+}2 = 10$ and $w_{10} = 2{+}3 = 5$, leading to $\frac{w_{11}}{w_{11}+w_{10}} = \frac{5}{5+5} = \frac{1}{2}$ and $\frac{w_{11}}{w_{11}+w_{01}} = \frac{5}{5+10} = \frac{1}{3}$.

Depending on the gradient smoothness of the underlying probabilistic black box, these explanations may slightly differ. However, if the additional six data points are placed such that the average black-box predictions of $\mathcal{W}_{11} \cup \mathcal{W}_{10}$ and $\mathcal{W}_{11} \cup \mathcal{W}_{01}$ are identical across both discretisations, the resulting explanations will be the same. Alternatively, if they are crisp predictions instead of class probabilities, the two explanations will also be indistinguishable regardless of where the additional six instances are situated within their respective hyper-rectangles. Note that in general it is easier to manipulate the explanations when dealing with crisp predictions rather than probabilities as we only have to consider which side of the black-box decision surface—if one runs across a given hyper-rectangle—to place each data point. The added benefit of this observation is evidence that discretising each numerical feature into more than three bins is not necessarily beneficial, with the most important partition boundaries being the ones enclosing the explained data point. [6]

[6] To facilitate further exploration of the explanatory setting discussed in this section we implemented a no-code interactive widget within a Jupyter Notebook and published it on GitHub at https://github.com/

This complex relation between explanations, the discretisation underlying an interpretable representation and the distribution of data points transformed by the IR to fit a surrogate linear model must be well-understood to ensure the veracity of explanatory insights. Given how sensitive an explanation is to these factors, small variations to the parameterisation of the aforementioned building blocks may sometimes yield disparate or even opposing insights into the behaviour of a black box. Such counterintuitive explanations can, for example, be achieved by shifting the discretisation boundaries for a fixed data sample, or instead by moving around these instances with a fixed IR. The discretisation process should therefore be optimised to guarantee the most truthful explanation that cannot be easily swayed by altering the data sample either in its distribution or size.

To overcome some of these problems and facilitate explanations that are more diverse than influence of interpretable concepts, alternative surrogate models can be used (Sokol et al. 2019). Logical predictors, such as *decision trees*, are particularly appealing given that they provide a wide range of explanations and they do not introduce any restrictions on the behaviour or interrelation of features, albeit they do impose axis-parallel partition of the feature space (Sokol and Flach 2020b; Sokol 2021). They are particularly suited for explaining tabular data, for which they alleviate the need for a separate interpretable representation as noted in the previous section. In particular, they can automatically learn a locally faithful, class-aware discretisation, with the added benefit of modelling combinations of hyper-rectangles and not suffering from information loss or stochasticity when applying the IR transformation (Sokol et al. 2019). In the following section we explore more guidelines that can help us to design and build robust and trustworthy interpretable representations with well-understood properties.

5 Towards robust and trustworthy interpretable representations

Our investigation of interpretable representations has revealed that a one-size-fits-all approach is often suboptimal. As with many other steps of the machine learning workflow, IRs need to be crafted for the problem at hand to be useful, reliable, robust and trustworthy (Rudin 2019; Sokol and Flach 2021). Moreover, the way in which an interpretable representation is built and operationalised determines the meaning of the resulting explanations in addition to constraining their possible types and compatible communication media. IR properties and desiderata should therefore be well-understood, guiding their development and deployment in each unique context. By following best practice—aspects of which are outlined below—we can improve veracity and faithfulness of post-hoc and model-agnostic explainers that rely on IRs, thus address some criticism of these techniques (Rudin 2019). Notably, consulting the latest findings in each relevant discipline—natural language processing, computer vision and discretisation of numerical data—can offer a treasure trove of insights and contribute core concepts to the fundamental design of interpretable representations.

fat-forensics/resources/tree/master/tabular_surrogate_builder. It allows to investigate the influence of the numerical feature discretisation and the number of data points placed in each hyper-rectangle on the surrogate explanation extracted from a linear model by manually adjusting these parameters.

Popular data pre-processing and feature engineering techniques used across these domains can be adapted to build more informative IRs; for example, word clustering for text data, object detection and semantic segmentation for images, and subgroup discovery for tabular data.

Information Loss While the many-to-one mapping pertinent to tabular IRs may seem detrimental at first due to the resulting loss of information, this procedure creates sparsity and reduces the perceived complexity of the data, which are often a prerequisite of human intelligibility (Sokol and Vogt 2023). This situation is unique to tabular data since both images and text are inherently comprehensible. Given the necessity of representing numerical features as human-understandable categories, the mapping should focus on eliciting the most insightful concepts and only discard redundant information (refer to Sect. 3.2); for example, partitioning a range of numbers into relatively uniform bins with respect to the underlying labels while ensuring that they are also meaningful to explainees. Notably, non-adjacent numerical intervals can be combined into a single interpretable concept if such an aggregation improves human understanding. The discretisation process should be driven by precise optimisation and evaluation objectives that are defined based on the explanatory context, domain constraints and user expectations. Since different thresholds can yield distinct or even opposing explanations—thus adversely impacting their trustworthiness—it is important to set out well-defined goals and metrics that capture these properties in detail.

Optimisation and Evaluation Criteria Interpretable representation desiderata should be formalised to allow for straightforward IR optimisation, testing and comparison both with respect to domain-specific characteristics and technical properties, thus precisely guiding the development and evaluation of IRs. While the former objective is difficult to define in an application-agnostic way, the latter should take into consideration the structure of the entire data or the specific neighbourhood being explained (refer to Sect. 3.2). Since human comprehension of text is intrinsically consistent with how the corresponding IRs are operationalised—i.e., switching concepts on and off by removing relevant words from an excerpt of text—these criteria appear to be entirely encapsulated by the user's perception of individual tokens, which largely depends on the underlying pre-processing step (refer to Sect. 2.1). A similar line of reasoning applies to images, regardless of whether the IR is based on edge detection or semantic segmentation; the objective is to separate visual concepts that are distinct from a human perspective and relevant to the predictive task being explained (refer to Sects. 2.2 and 3.1). For example, an algorithmically generated image IR can be improved by merging, possibly non-adjacent, super-pixels representing the background of an object into a single interpretable concept.

Tabular data, on the other hand, in themselves provide a rich source of information that can be used to algorithmically navigate the discretisation process that underpins the corresponding interpretable representation. Characteristics such as data density, distribution, their black-box predictions and confidence thereof can be used to partition a feature space into geometrically consistent and homogeneous concepts. Generic metrics that determine the purity, uniformity and faithfulness of discretised data can be utilised to this end; for example, see the evaluation strategy proposed in Sect. 3.2.

While our analysis showed that discretising each numerical attribute into more than three bins is not necessarily beneficial (refer to Sect. 4), this is purely a consequence of the hyper-rectangle merging procedure employed by the binarisation step that follows. By designing a more complex transformation function from the discrete space into the binary interpretable representation, the IR can benefit from higher discretisation granularity, e.g., consider allowing multiple hyper-rectangles to contribute to the explained concept. Additional improvements can include only using intervals that are bounded from each side to narrow down the scope of the explanation and prevent it from being biased by out-of-distribution instances. The optimisation and evaluation of tabular IR discretisation can further be enhanced by domain-specific knowledge that assigns a human-comprehensible concept to each partition, akin to how image segmentation may be assessed based on the semantic consistency of visual object separation. Finally, in addition to an independent validation of IR quality, the robustness and stability of the resulting explanations can be measured while varying IR parameters (Sokol et al. 2022a).

Human-in-the-loop Design Explainee-driven interactive creation or personalisation of interpretable representations is an interesting avenue of research on the crossroads of explainable artificial intelligence and human–computer interaction (Sokol and Flach 2018, 2020c; Lage and Doshi-Velez 2020). It has the potential to formulate further recommendations for the composition and operationalisation of IRs for individual applications, but such a solution comes at the expense of a user-in-the-loop architecture that may be difficult to automate and scale. This design choice, however, can be easily justified since constructing an interpretable representation that is intelligible and useful is often user- and application-dependent or even unique to the explained data point. Moreover, the core premise of IRs is to encode concepts that are *meaningful* to the target audience and *relevant* to the question that prompted the need for explainability in the first place, thus relying upon computer-generated IRs without communicating their behaviour and properties to the explainees may be counterproductive (refer to Sect. 3.1). While promising, scaling up human-in-the-loop interpretable representations appears to be impractical without a concrete deployment use case.

Information Removal (Proxy) The interpretable representations th at we deal with in this paper capture human-intelligible concepts that can be switched on or off. This process defines the explanatory *fact* and *foil*, the distinction between which forms the basis of insights into the predictive behaviour of the model under investigation. Given the significance of this procedure, the difference between the two should be *semantically meaningful* as well as *computationally effective* in the chosen operational context (refer to Sect. 3). To this end, the strategy used to discard information from the explained instance must achieve its goal and avoid introducing unintended biases, both of which can be measured by observing the response of a black box when predicting data manipulated via an interpretable representation. This process does not affect text since its IRs support a direct removal of tokens and relevant predictive models can handle instances that have been altered in this way, however images and tabular data require an algorithmic proxy.

For images, this is achieved through *content replacement*, with retouching, object injection and region occlusion being the most popular choices (see Sect. 2.2 for more details). While the evidence presented in Sect. 3.1 clearly shows the disadvantage of using mean-colour occlusion and a relatively comparable properties of all the other tested colouring strategies, our results may not generalise to different image data sets and black-box models, therefore a similar analysis should be performed prior to deciding on the content replacement technique. Even if such an investigation shows that, overall, a selected information removal proxy behaves comparably to others for a specific setup, it may still be inappropriate for a particular image and explained class. For example, the *white* and *black* occlusion strategies are nearly indistinguishable in our experiments (refer to Fig. 9), but employing the former for the dog image used across this paper for illustration purposes (shown in Fig. 3) may be ineffective for explaining it, and *winter scenery* in general, given the photo's snowy background. Therefore, unless the data domain is highly homogeneous—e.g., all pictures follow a fixed object presentation pattern and colour scheme—the information removal proxy may need to be configured on a case-by-case basis.

An information removal proxy for tabular data is more complex given that it operates on an interpretable representation that is based on discretisation of numerical attributes followed by a binarisation step. Specifically, switching off an IR element is equivalent to placing the value of the corresponding numerical feature outside of the range encoded by this concept; or, if the attribute is categorical, choosing any other value not captured by this concept. Therefore, this operation should ensure that moving data between different hyper-rectangles (or their collections) determined by distinct binary interpretable spaces is *semantically meaningful* and corresponds to abstractions that can be captured *computationally* (e.g., by monitoring the change of black-box predictions for these instances).

When working with categories generated via discretisation, a step in this direction can be a more informed process of merging these hyper-rectangles into binary concepts—in contrast to doing so based on their geometrical alignment—in addition to narrowing down the scope of the foil by explicitly bounding the numerical ranges instead of comparing spaces that cannot be easily represented by finite data samples. A similar strategy may also be applied to images and text, where (non-adjacent) super-pixels and word-based tokens can be combined into a single IR component. Additionally, with a well-crafted tabular interpretable representation, the volatility of explanations may be reduced since such an IR becomes less reliant on the distribution of the (possibly random) data sample transformed into this domain and used to train a local surrogate model (refer to Sect. 4). Nonetheless, building an IR that expresses the fact and the foil as complementary events and allows to intuitively manipulate them by tweaking each IR component *independently* may be impractical or impossible to achieve, in which case self-contained regions that do not directly rely on manipulation of individual feature values—e.g., determined by decision tree leaves or data clustering—offer an attractive alternative.

Stochasticity Surrogate explainers—a big beneficiary of interpretable representations—sample the data required to train the (local) model directly from the binary IR (Ribeiro et al. 2016). While reasonable for images and text where generating

data by manipulating raw pixels, letters or words may easily introduce inconsistencies, following the same procedure for a tabular IR entails converting the binary sample back into the original domain (to be predicted by the explained black box), which requires *random sampling* because of the many-to-one forward transformations—see Fig. 5 for reference. [7] Specifically, to execute this process we first choose *at random* one of the merged hyper-rectangles if the binary component is 0; 1 uniquely identifies a hyper-rectangle in our case. Next, we draw a numerical value from the range defined by this hyper-rectangle, e.g., using a (truncated) Gaussian distribution fitted to the (training) data enclosed by this hyper-rectangle; categorical features are uniquely identified by a hyper-rectangle. However, tabular data can be sampled in their original representation, which removes the need for this stochastic operation, thus improving robustness and decreasing volatility of surrogate explanations (Sokol et al. 2019).

Data drawn from the original domain can be easily transformed into a discrete representation and then binarised. Moving in the opposite direction in a deterministic fashion requires memorising the correspondence between these points in different representations when executing the forward transformation. This matching offers an algorithmic workaround that can be compared to storing the pixel structure and segment adjacency for images or a sentence skeleton and any pre-processing steps for text. By sampling in the original domain and connecting different representations of each instance we avoid using the stochastic inverse IR transformation of tabular data, hence reduce randomness and improve stability of the resulting explanations. However, this strategy forfeits the implicit locality and diversity achieved by operating directly on the binary representation, therefore the substitute sampling algorithm should directly target a well-defined subspace to carefully capture the behaviour of the explained black box in this region (Sokol et al. 2019). More broadly, recognising the strengths and weaknesses of individual components from which explainability algorithms are built allows us to adapt their architecture accordingly, creating the best possible explainer for the problem at hand (Sokol et al. 2022a).

Alternative Surrogate Models Interpretable representations offer a sparse and human-comprehensible medium for communicating explanations of black-box models and their predictions. Section 4, nonetheless, demonstrated that certain pairings of IRs and surrogate models yield defective explainers whose insights can be misleading or outright incorrect. Given the data pre-processing that interpretable representations entail, *logical predictive models* appear to be a good (surrogate model) candidate since they are inherently transparent and intelligible. In particular, rule lists and decision trees should be considered to this end, with the latter choice being especially appealing for tabular data for which they can automatically compose an IR in addition to modelling it (Sokol et al. 2019). Moreover, they account for feature interactions, and their optimisation procedure is well-aligned with the IR faithfulness objectives discussed in Sect. 3.2. While tree training procedures tend to be greedy, alternatives that con-

[7] Transitioning from an original into a discrete representation is a many-to-one operation if the underlying data set contains numerical features. Transforming the discrete representation into a binary IR is also a many-to-one mapping if any discretised attribute has more than two unique values. Section 3.2 discusses the information loss pertinent to tabular interpretable representations in more detail.

sider multiple features at any given iteration could improve the quality of the resulting interpretable representations and surrogate explainers even further.

6 Conclusion and future work

Our findings show the importance of building robust, trustworthy and algorithmically sound interpretable representations as well as their role in defining the question answered by the resulting explanations and veracity thereof. Among others, we demonstrated that building IRs with generic algorithms may lead to subpar explainers, and that the intended application domain and audience should always be accounted for, in addition to considering interactive customisation and personalisation of interpretable representations. In particular, we discussed a popular operationalisation of IRs for image, text and tabular data in which they are used as binary indicators of presence and absence of human-intelligible concepts. This framework is then combined with surrogate models to quantify the influence of such concepts on individual black-box predictions.

In this setting, we identified challenges such as implicit assumptions, flawed information removal proxies, undesired parametrisation choices, insufficient faithfulness and transformation stochasticity—which are particularly prominent for tabular and image data—and showed how to overcome them. We also demonstrated the limitations of explaining binary interpretable representations of tabular data with linear models and suggested logical models as a viable alternative. Our findings reinforce the importance of considering the structure of the data, especially in the neighbourhood of the explained instance, when transforming them into an IR as well as having a well-defined objective and evaluation metric to aid in IR construction and optimisation. Many of these goals can be achieved by drawing inspiration from the fields of computer vision, natural language processing and data discretisation, and more broadly data wrangling and modelling in machine learning, which can inform better design of interpretable representations.

Our future work will investigate algorithmic information removal proxies, focusing on meaningful and effective occlusion approaches for images and feature value replacement techniques for tabular data; we will also look into user-in-the-loop design of interpretable representations. Furthermore, we will survey inherently transparent models that are best suited to various IRs, ensuring their technological compatibility and explanatory appeal, in particular focusing on different types of logical predictive models. To enable safe adoption of interpretable representations in real-life applications we will finally evaluate the most promising approaches with targeted user studies.

Funding This work was supported by the TAILOR Network—an ICT-48 European AI Research Excellence Centre funded by EU Horizon 2020 research and innovation programme (grant agreement number 952215). Additional funding was provided by the ARC Centre of Excellence for Automated Decision-Making and Society, sponsored by the Australian Government through the Australian Research Council (project number CE200100005); and the Hasler Foundation (grant number 23082).

Data availability Not applicable.

Code availability All of the experimental results presented in this paper can be reproduced with FAT Forensics (Sokol et al. 2020, 2022b)—a Python package that offers modular implementations of various explainability algorithms—with the experiment code available on GitHub (https://github.com/So-Cool/bLIMEy/tree/master/DAMI_2024). Additional resources—such as computational notebooks—that explore properties of tabular and image interpretable representations in the context of surrogate explainers are available as part of the FAT Forensics documentation (https://fat-forensics.org/how_to/index.html) and the "What and How of Machine Learning Transparency" hands-on tutorial (https://github.com/fat-forensics/Surrogates-Tutorial) organised at ECML-PKDD 2020 (Sokol et al. 2022a).

Declarations

Conflict of interest We declare that we have no conflicts of interest.

A Derivation of OLS explanations for binary IRs of tabular data

When analysing the behaviour of algorithmic black boxes with surrogate explainers, linear models can be used to quantify the positive or negative influence of interpretable concepts extracted from a data point of interest on its black-box prediction (Friedman and Popescu 2008; Ribeiro et al. 2016; Sokol et al. 2019). For some binary interpretable domains, however, such an approach is inherently flawed; it is particularly problematic for tabular data transformed into the binary interpretable representation introduced in Sect. 2.3, i.e., achieved through feature discretisation followed by a binarisation step. In this appendix, we derive a closed-form expression of this explanation type, which allows us to demonstrate how the influence of an interpretable concept measured by the coefficients of a linear model may be deceiving. The insights stemming from our analysis can be used to manipulate surrogate explanations, e.g., those produced by LIME (Ribeiro et al. 2016), through a specially crafted, yet perfectly valid, IR discretisation and data sample.

Our results are based on the analytical solution to unweighted (Θ) and weighted ($\Theta_{\mathbf{W}}$) Ordinary Least Squares outlined in Eqs. 5 and 6 respectively, where $\mathbf{W}$ is the weight matrix, $\mathbf{X}$ is the binary interpretable representation data matrix, and $\mathbf{y}$ is a vector holding the corresponding black-box predictions. In our analysis, we assume that the model under investigation is a probabilistic classifier, in which case $\mathbf{y}$ captures probabilities of the explained class; nonetheless, a similar line of reasoning applies to regressors and crisp classifiers. In the latter scenario, the elements of $\mathbf{y}$ are assumed to be 1 when the black-box predictions are the same as the explained class, and 0 for any other class. Modelling $\mathbf{y}$ in such a way generates one-vs-rest explanations—i.e., evidence for the black box predicting the explained rather than any other class—akin to the insights produced for probabilistic black boxes, for which the surrogate is trained

only on the probabilities of the *explained class*. Therefore, both approaches measure the influence of interpretable concepts—determined by the coefficients of the linear model—on a selected class when tasked with telling it apart from all the other classes.

$$\Theta = (\mathbf{X}^T\mathbf{X})^{-1}\,\mathbf{X}^T\mathbf{y} \tag{5}$$

$$\Theta_{\mathbf{W}} = (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\,\mathbf{X}^T\mathbf{W}\mathbf{y} \tag{6}$$

In the interest of brevity and readability, we analyse tabular data with two numerical features—similar to the examples shown in Figs. 11 and 13—nonetheless our findings generalise to an arbitrary number of attributes that are both categorical and numerical. In a generic setting, for n features there will be n binary concepts with 2^n unique encodings in the interpretable representation (the cardinality thereof). If additionally we choose to model the intercept of the linear regression, a phantom all-1 column vector is inserted at the front of the data matrix $\mathbf{X}$. Therefore, the $\mathbf{X}^T\mathbf{X}$ and $\mathbf{X}^T\mathbf{W}\mathbf{X}$ components of Θ and $\Theta_{\mathbf{W}}$ respectively are square matrices of $n \times n$ shape sans the intercept or $(n+1) \times (n+1)$ when the intercept is modelled.

Figure 11 depicts a simplistic view of data sampling for two numerical attributes with just one instance in each discrete hyper-rectangle. In reality, however, we should expect their large quantity since it allows to better approximate the behaviour of the underlying black box, especially when the number of features is high. In this particular case, the binary interpretable representation data matrix $\mathbf{X}$—with the first column (red) inserted to model the intercept and the remaining columns (blue) representing the binary data—is:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix},$$

which gives:

$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 9 & 3 & 3 \\ 3 & 3 & 1 \\ 3 & 1 & 3 \end{bmatrix}.$$

Since some of the hyper-rectangles are merged when transitioning from the discrete into the binary interpretable representation, $\mathbf{X}$ contains duplicated rows. The influence of this phenomenon is magnified even further when multiple data points are placed within a single hyper-rectangle. Without loss of generality, we can use the *weighted* variant of OLS with the data set $\mathbf{X}$ composed of only one copy of each unique binary data point and the weights corresponding to their counts. In this case:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{W} = \begin{bmatrix} w_{11} & 0 & 0 & 0 \\ 0 & w_{10} & 0 & 0 \\ 0 & 0 & w_{01} & 0 \\ 0 & 0 & 0 & w_{00} \end{bmatrix},$$

where w_{ij} is the count of data points residing in all of the hyper-rectangles that are assigned the (i, j) coordinates in the binary interpretable representation—see the $(x_1^\star, x_2^\star)$ coordinates in Fig. 11 for reference. Therefore, for an arbitrary number of instances in a data set with two numerical features when modelling the intercept:

$$\begin{aligned} \mathbf{X}^T\mathbf{W}\mathbf{X} &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} w_{11} & 0 & 0 & 0 \\ 0 & w_{10} & 0 & 0 \\ 0 & 0 & w_{01} & 0 \\ 0 & 0 & 0 & w_{00} \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} w_{11} & w_{10} & w_{01} & w_{00} \\ w_{11} & w_{10} & 0 & 0 \\ w_{11} & 0 & w_{01} & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \sum w_{ij} & w_{11} + w_{10} & w_{11} + w_{01} \\ w_{11} + w_{10} & w_{11} + w_{10} & w_{11} \\ w_{11} + w_{01} & w_{11} & w_{11} + w_{01} \end{bmatrix}. \end{aligned}$$

For the example in Fig. 11—where $w_{11} = 1$, $w_{10} = 2$, $w_{01} = 2$ and $w_{00} = 4$—a calculation for the weighted variant agrees with the previous result computed directly for $\mathbf{X}^T\mathbf{X}$.

Next, we analyse the second component of the $\Theta_{\mathbf{W}}$ formula:

$$\begin{aligned} \mathbf{X}^T\mathbf{W}\mathbf{y} &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} w_{11} & 0 & 0 & 0 \\ 0 & w_{10} & 0 & 0 \\ 0 & 0 & w_{01} & 0 \\ 0 & 0 & 0 & w_{00} \end{bmatrix} \times \begin{bmatrix} y_{11} \\ y_{10} \\ y_{01} \\ y_{00} \end{bmatrix} \\ &= \begin{bmatrix} w_{11} & w_{10} & w_{01} & w_{00} \\ w_{11} & w_{10} & 0 & 0 \\ w_{11} & 0 & w_{01} & 0 \end{bmatrix} \times \begin{bmatrix} y_{11} \\ y_{10} \\ y_{01} \\ y_{00} \end{bmatrix} \\ &= \begin{bmatrix} \sum w_{ij} y_{ij} \\ w_{11} y_{11} + w_{10} y_{10} \\ w_{11} y_{11} + w_{01} y_{01} \end{bmatrix}. \end{aligned}$$

This formulation, however, presupposes that all of the data points that share the same (i, j) coordinates in the binary interpretable representation have the same target value (i.e., black-box prediction) y_{ij}. To allow multiple copies of the same instance with different target values, we generalise this result by going back to Θ, which is the solution to the classic OLS. This approach is valid since weighted OLS for which the weights represent the count of each unique data point is equivalent to classic OLS for a data set whose instances are duplicated according to the counts given by the corresponding weights.

Let us denote $f : \mathcal{X} \rightarrow \mathcal{Y}$ as the black-box model and $IR : \mathcal{X} \rightarrow \mathcal{X}^\star$ as the transformation function from tabular data $\mathcal{X}$ into their binary interpretable representation $\mathcal{X}^\star$. Let us further define $\mathcal{W}_{ij} = \{x \in \mathbf{X} : IR(x) = (i, j)\}$ as the set of all the data points that are assigned to the same hyper-rectangle (i, j) in the binary interpretable representation, and $\mathcal{W} = \mathbf{X}$ as the set of all the data points. Now, recall that w_{ij} is the count of data points whose binary interpretable representation is (i, j), therefore $|\mathcal{W}_{ij}| = w_{ij}$ and $|\mathcal{W}| = \sum w_{ij}$. Without loss of generality, we can reformulate the $\mathbf{X}^T\mathbf{W}\mathbf{y}$ part of the $\Theta_\mathbf{W}$ equation as $\mathbf{X}^T\mathbf{y}$, which allows us to sum over all of the unique black-box predictions of instances assigned to particular hyper-rectangles:

$$\mathbf{X}^T\mathbf{y} = \begin{bmatrix} \sum w_{ij} y_{ij} \\ w_{11} y_{11} + w_{10} y_{10} \\ w_{11} y_{11} + w_{01} y_{01} \end{bmatrix} = \begin{bmatrix} \sum_{i \in \mathcal{W}} y_i \\ \sum_{i \in \mathcal{W}_{11} \cup \mathcal{W}_{10}} y_i \\ \sum_{i \in \mathcal{W}_{11} \cup \mathcal{W}_{01}} y_i \end{bmatrix}.$$

This step allows us to relax the assumption of duplicated target values y_{ij}, hence avoid imposing restrictions on the type of the model under investigation (probabilistic, crisp or regressor) and whether the binary representation has full fidelity with respect to the black box. [8]

Finally, to better understand the meaning of influence-based explanations, we reformulate the sum of black-box predictions as their average:

$$\begin{aligned} \mathbf{X}^T\mathbf{y} &= \begin{bmatrix} \sum_{i \in \mathcal{W}} y_i \\ \sum_{i \in \mathcal{W}_{11} \cup \mathcal{W}_{10}} y_i \\ \sum_{i \in \mathcal{W}_{11} \cup \mathcal{W}_{01}} y_i \end{bmatrix} = \begin{bmatrix} \sum_{i \in \mathcal{W}} y_i / \sum w_{ij} \times \sum w_{ij} \\ \sum_{i \in \mathcal{W}_{11} \cup \mathcal{W}_{10}} y_i / (w_{11} + w_{10}) \times (w_{11} + w_{10}) \\ \sum_{i \in \mathcal{W}_{11} \cup \mathcal{W}_{01}} y_i / (w_{11} + w_{01}) \times (w_{11} + w_{01}) \end{bmatrix} \\ &= \begin{bmatrix} \overline{y}_{\mathcal{W}} \times \sum w_{ij} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \times (w_{11} + w_{10}) \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \times (w_{11} + w_{01}) \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \overline{y}_{\mathcal{W}} \times \sum w_{ij} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \times (w_{11} + w_{10}) \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \times (w_{11} + w_{01}) \end{bmatrix} \end{aligned}$$

[8] Note that in the original formulation of $\mathbf{X}^T\mathbf{W}\mathbf{y}$ instances located within each hyper-rectangle determined by the underlying interpretable representation are assumed to share the same black-box prediction. This constraint makes the original weighted solution incompatible with regressors and probabilistic classifiers that are in need of explainability; for crisp classifiers, on the other hand, it implies that the binarised data space respects the black-box decision surface, i.e., it achieves full fidelity.

$$= \begin{bmatrix} \sum w_{ij} & 0 & 0 \\ 0 & w_{11}+w_{10} & 0 \\ 0 & 0 & w_{11}+w_{01} \end{bmatrix} \times \begin{bmatrix} \overline{y}_{\mathcal{W}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \end{bmatrix},$$

and combine this result with $\mathbf{X}^T\mathbf{W}\mathbf{X}$:

$$\begin{bmatrix} \sum w_{ij} & w_{11}+w_{10} & w_{11}+w_{01} \\ w_{11}+w_{10} & w_{11}+w_{10} & w_{11} \\ w_{11}+w_{01} & w_{11} & w_{11}+w_{01} \end{bmatrix}^{-1} \times \begin{bmatrix} \sum w_{ij} & 0 & 0 \\ 0 & w_{11}+w_{10} & 0 \\ 0 & 0 & w_{11}+w_{01} \end{bmatrix} \times \begin{bmatrix} \overline{y}_{\mathcal{W}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \end{bmatrix}$$

$$= \begin{bmatrix} \sum w_{ij} & w_{11}+w_{10} & w_{11}+w_{01} \\ w_{11}+w_{10} & w_{11}+w_{10} & w_{11} \\ w_{11}+w_{01} & w_{11} & w_{11}+w_{01} \end{bmatrix}^{-1} \times \begin{bmatrix} \frac{1}{\sum w_{ij}} & 0 & 0 \\ 0 & \frac{1}{w_{11}+w_{10}} & 0 \\ 0 & 0 & \frac{1}{w_{11}+w_{01}} \end{bmatrix}^{-1} \times \begin{bmatrix} \overline{y}_{\mathcal{W}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \end{bmatrix}$$

$$= \left(\begin{bmatrix} \frac{1}{\sum w_{ij}} & 0 & 0 \\ 0 & \frac{1}{w_{11}+w_{10}} & 0 \\ 0 & 0 & \frac{1}{w_{11}+w_{01}} \end{bmatrix} \times \begin{bmatrix} \sum w_{ij} & w_{11}+w_{10} & w_{11}+w_{01} \\ w_{11}+w_{10} & w_{11}+w_{10} & w_{11} \\ w_{11}+w_{01} & w_{11} & w_{11}+w_{01} \end{bmatrix} \right)^{-1} \times \begin{bmatrix} \overline{y}_{\mathcal{W}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & \frac{w_{11}+w_{10}}{\sum w_{ij}} & \frac{w_{11}+w_{01}}{\sum w_{ij}} \\ 1 & 1 & \frac{w_{11}}{w_{11}+w_{10}} \\ 1 & \frac{w_{11}}{w_{11}+w_{01}} & 1 \end{bmatrix}^{-1} \times \begin{bmatrix} \overline{y}_{\mathcal{W}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{10}} \\ \overline{y}_{\mathcal{W}_{11} \cup \mathcal{W}_{01}} \end{bmatrix}.$$

This formulation demonstrates an unexpected role of the:

1. number of data points sampled in each hyper-rectangle on the resulting explanations (magnitudes of concept influence); and
2. irrelevance of the feature partitions other than the ones determining the hyper-rectangles that directly enclose the explained instance.

A more detailed discussion of the interpretation and significance of this result can be found in Sect. 4.

References

Achanta R, Shaji A, Smith K, Lucchi A, Fua P, Süsstrunk S (2012) SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans Pattern Anal Mach Intell 34(11):2274–2282

Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) ImageNet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. IEEE, pp 248–255
Friedman JH, Popescu BE (2008) Predictive learning via rule ensembles. Ann Appl Stat 2(3):916–954
Garcia S, Luengo J, Sáez JA, Lopez V, Herrera F (2012) A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. IEEE Trans Knowl Data Eng 25(4):734–750
Garreau D, Luxburg U (2020) Explaining the explainer: a first theoretical analysis of LIME. In: Chiappa S, Calandra R (eds) Proceedings of the twenty third international conference on artificial intelligence and statistics, PMLR, Online, Proceedings of Machine Learning Research, vol 108, pp 1287–1296
Kotsiantis S, Kanellopoulos D (2006) Discretization techniques: a recent survey. GESTS Int Trans Comput Sci Eng 32(1):47–58
Lage I, Doshi-Velez F (2020) Human-in-the-loop learning of interpretable and intuitive representations. In: Proceedings of the ICML workshop on human interpretability in machine learning, Vienna, Austria, vol 17
Lakkaraju H, Bastani O (2020) "How do I fool you?" Manipulating user trust via misleading black box explanations. In: Proceedings of the 2020 AAAI/ACM conference on AI, ethics, and society
Lakkaraju H, Kamar E, Caruana R, Leskovec J (2019) Faithful and customizable explanations of black box models. In: Proceedings of the 2019 AAAI/ACM conference on AI, ethics, and society, ACM
Laugel T, Renard X, Lesot MJ, Marsala C, Detyniecki M (2018) Defining locality for surrogates in post-hoc interpretablity. In: 3rd Workshop on human interpretability in machine learning (WHI 2018) at the 35th international conference on machine learning (ICML 2018), Stockholm, Sweden
Lundberg SM, Lee SI (2017) A unified approach to interpreting model predictions. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds) Advances in neural information processing systems 30. Curran Associates Inc., pp 4765–4774
Manning C, Schutze H (1999) Foundations of statistical natural language processing. MIT Press, Cambridge
Mittelstadt B, Russell C, Wachter S (2019) Explaining explanations in AI. In: Proceedings of the 2019 conference on fairness, accountability, and transparency, pp 279–288
Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S (2019) PyTorch: an imperative style, high-performance deep learning library. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R (eds) Advances in neural information processing systems, vol 32. Curran Associates Inc., pp 8026–8037
Ribeiro MT, Singh S, Guestrin C (2016) "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, San Francisco, CA, USA, August 13–17, 2016, pp 1135–1144
Rudin C (2019) Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat Mach Intell 1(5):206–215
Small E, Xuan Y, Hettiachchi D, Sokol K (2023) Helpful, misleading or confusing: How humans perceive fundamental building blocks of artificial intelligence explanations. In: ACM CHI 2023 workshop on human-centered explainable AI (HCXAI)
Sokol K (2021) Towards intelligible and robust surrogate explainers: a decision tree perspective. PhD Thesis, University of Bristol
Sokol K, Flach P (2020a) Explainability fact sheets: a framework for systematic assessment of explainable approaches. In: Proceedings of the 2020 conference on fairness, accountability, and transparency, pp 56–67
Sokol K, Flach P (2020b) LIMEtree: consistent and faithful surrogate explanations of multiple classes. arXiv:2005.01427
Sokol K, Flach P (2020c) One explanation does not fit all. In: KI-Künstliche Intelligenz pp 1–16
Sokol K, Flach P (2021) Explainability is in the mind of the beholder: establishing the foundations of explainable artificial intelligence. arXiv:2112.14466
Sokol K, Flach PA (2018) Glass-box: explaining AI decisions with counterfactual statements through conversation with a voice-enabled virtual assistant. In: IJCAI, pp 5868–5870
Sokol K, Vogt JE (2023) (Un)reasonable allure of ante-hoc interpretability for high-stakes domains: transparency is necessary but insufficient for comprehensibility. In: ICML 3rd workshop on interpretable machine learning in healthcare (IMLH)
Sokol K, Vogt JE (2024) What does evaluation of explainable artificial intelligence actually tell us? A case for compositional and contextual validation of XAI building blocks. In: Extended abstracts of the 2024 CHI conference on human factors in computing systems

Sokol K, Hepburn A, Santos-Rodriguez R, Flach P (2019) bLIMEy: Surrogate prediction explanations beyond LIME. In: 2019 Workshop on human-centric machine learning (HCML 2019) at the 33rd conference on neural information processing systems (NeurIPS 2019), Vancouver, Canada

Sokol K, Hepburn A, Poyiadzi R, Clifford M, Santos-Rodriguez R, Flach P (2020) FAT Forensics: a Python toolbox for implementing and deploying fairness, accountability and transparency algorithms in predictive systems. J Open Source Softw 5(49):1904. https://doi.org/10.21105/joss.01904

Sokol K, Hepburn A, Santos-Rodriguez R, Flach P (2022) What and how of machine learning transparency: building bespoke explainability tools with interoperable algorithmic components. J Open Source Educ 5(58):175

Sokol K, Santos-Rodriguez R, Flach P (2022) Fat Forensics: a Python toolbox for algorithmic fairness, accountability and transparency. Softw Impacts 14:100406. https://doi.org/10.1016/j.simpa.2022.100406

Vedaldi A, Soatto S (2008) Quick shift and kernel methods for mode seeking. In: European conference on computer vision. Springer, pp 705–718

van der Waa J, Robeer M, van Diggelen J, Brinkhuis M, Neerincx M (2018) Contrastive explanations with local foil trees. In: Workshop on human interpretability in machine learning (WHI 2018) at the 35th international conference on machine learning (ICML 2018), Stockholm, Sweden

van der Walt S, Schönberger JL, Nunez-Iglesias J, Boulogne F, Warner JD, Yager N, Gouillart E, Yu T, The scikit-image contributors (2014) Scikit-image: Image processing in Python. PeerJ 2:e453. https://doi.org/10.7717/peerj.453

Xuan Y, Small E, Sokol K, Hettiachchi D, Sanderson M (2023) Can users correctly interpret machine learning explanations and simultaneously identify their limitations? arXiv:2309.08438

Yang M, Kim B (2019) Benchmark attribution methods with ground truth. In: 2019 Workshop on human-centric machine learning (HCML 2019) at the 33rd conference on neural information processing systems (NeurIPS 2019), Vancouver, Canada

Zhang Y, Song K, Sun Y, Tan S, Udell M (2019) "Why should you trust my explanation?" Understanding uncertainty in LIME explanations. In: AI for social good workshop at the 36th international conference on machine learning (ICML 2019), Long Beach, California