Effective data exploration through clustering of local attributive explanations

Elodie Escriva, Tom Lefrere, Manon Martin, Julien Aligon, Alexandre Chanson, Jean-Baptiste Excoffier, Nicolas Labroche, Chantal Soulé-Dupuy, Paul Monsarrat

Effective data exploration through clustering of local attributive explanations

Elodie Escriva[a,b], Tom Lefrere[c], Manon Martin[c], Julien Aligon[a], Alexandre Chanson[c], Jean-Baptiste Excoffier[b], Nicolas Labroche[c], Chantal Soulé-Dupuy[a], Paul Monsarrat[d,e,f]

[a] *Université de Toulouse-Capitole, IRIT, (CNRS/UMR 5505), Toulouse (FR)*
[b] *Kaduceo, Toulouse (FR)*
[c] *Université de Tours, Laboratoire d'Informatique Fondamentale et Appliquée, Blois (FR)*
[d] *RESTORE Research Center, Toulouse (FR)*
[e] *Artificial and Natural Intelligence Toulouse Institute ANITI, Toulouse (FR)*
[f] *Oral Medicine Department, Toulouse (FR)*

Abstract

Machine Learning (ML) has become an essential tool for modeling complex phenomena, offering robust predictions and comprehensive data analysis. Nevertheless, the lack of interpretability in these predictions often results in a closed-box effect, which the field of eXplainable Machine Learning (XML) aims to address. Local attributive XML methods, in particular, provide explanations by quantifying the contribution of each attribute to individual predictions, referred to as influences. This type of explanation is the most acute as it focuses on each instance of the dataset and allows the detection of individual differences. Additionally, aggregating local explanations allows for a deeper analysis of the underlying data. In this context, influences can be considered as a new data space to reveal and understand complex data patterns. We hypothesize that these influences, derived from ML explanations, are more informative than the original raw data, especially for identifying homogeneous groups within the data. To identify such groups effectively, we utilize a clustering approach. We compare clusters formed using raw data

Email addresses: elodie.escriva@kaduceo.com (Elodie Escriva), julien.aligon@irit.fr (Julien Aligon), chanson@univ-tours.fr (Alexandre Chanson), jeanbaptiste.excoffier@kaduceo.com (Jean-Baptiste Excoffier), labroche@univ-tours.fr (Nicolas Labroche), chantal.soule-dupuy@irit.fr (Chantal Soulé-Dupuy), paul.monsarrat@univ-tlse3.fr (Paul Monsarrat)

against those formed using influences computed by various local attributive XML methods. Our findings reveal that clusters based on influences consistently outperform those based on raw data, even when using models with low accuracy.

Keywords: Explainable Machine Learning (XML), Prediction explanation, Explanations clustering, Instance clustering, Machine learning explanation, Explainable Artificial Intelligence (XAI).

---

1. Introduction

Data exploration is the crucial yet tedious task of analyzing possibly large and complex datasets to extract insights, i.e., interactively identifying findings that expose "the unanticipated" [1]. This activity is important in many domains such as finance, insurance, banking, chemistry and healthcare. This article proposes a new framework to efficiently explore datasets through eXplainable Machine Learning (XML, also termed eXplainable Artificial Intelligence - XAI). Although this framework is applicable to any domain, healthcare data will be used for illustration purposes.

Problem positioning. Data modeling is a very broad problem, based on a variety of techniques adapted to the desired purpose (e.g., business intelligence (BI), statistical, predictive). One of the possible goals of predictive analysis is to gain a better understanding of the relationships between the attributes of the dataset, especially the hidden relationships inferred by the machine learning model used. Based on this principle, a detailed analysis of predictions can be highly informative and multi-layered. By considering an analysis of predictions related to the presence of a systemic disease (patient level), it may be possible to gain insights from the predictive model to identify finer groups of patients who share the same biological, clinical or socio-demographic characteristics for a given health condition.

To create these groups, a trivial solution would be to cluster patients based on such medical characteristics. Still, then, there would be no advantage of considering the pre-existing knowledge about the presence of pathology in this clustering step. Adding pathology information directly to the description of instances can lead to misleading clustering, preventing the discovery of potential subgroups for that particular pathology.

A more appropriate solution to this problem is to consider semi-supervised clustering methods [2, 3], which rely on user constraints and preferences (such as pathology labels) as side information to improve the convergence or quality of the produced clustering. However, as advocated in [4, 5], the introduction of constraints can degrade the final clustering quality if not associated with some prior utility measure for a constraint [6]. More importantly, simple methods that directly implement user constraints during clustering may not be appropriate [7] and the development of adapted metrics, incorporating this side information, should instead be considered [8, 9].

Proposition. It is postulated that XML influences can be considered as a new data space to explore. XML methods provide either global or local insights about the behavior of a predictive ML model [10]. Among the most popular methods are local attribution XML methods that produce influences, especially LIME [11] and approximation of Shapley Values such as SHAP [12], the K-depth [13] and Coalitional approaches [14]. Their popularity is due to the instance-level accuracy of these explanations, which links the impact of each attribute to the prediction made for each instance. As a consequence, local explanations are increasingly used in AI-assisted tools to offer more information than solely the prediction [15]. Indeed, these influences may convey less noise or spurious indicators than the original space, as only the most significant information is preserved. Exploring the space of influences thus represents a methodology able to reveal the key attributes for the prediction, both in themselves and through their interactions, by improving the "signal-to-noise" ratio in the dataset.

Contribution. This paper builds upon our previous work in [16] by proposing a novel framework for data exploration. Rather than considering the raw data space, the present framework focuses on the benefits of using XML influence space for data exploration based on clustering algorithms and provides a real use case in the healthcare domain. This work can, therefore, be considered as a contribution to the domain of Actionable XAI, as outlined in [17], which considers actionable concepts, measures, and metrics for explainable learning and reasoning intending to improve data analysis or machine learning (ML) models based on explanations. The main advantage of the present methodology relies on reducing the disturbances in the description of instances through explanations. The predictive model allows for better

separation between instances, which should result in higher-quality clusters compared to the original raw data. This means one can identify more consistent subgroups of influences within the data, specifically related to the internal structure of the studied phenomenon. It is crucial to note that the quality and believability of the local explanations are directly linked to the quality of the predictive model used to generate them.

This work comprehensively investigates the benefits of using local influences as a new input for clustering, to identify more informative and homogeneous subgroups. Importantly, since the quality of the XML influence space may depend on the quality of the ML model that has been used, we also explore the robustness of this framework regarding low-accuracy models with misclassified instances.

The main contributions of the paper are the following:

1. We consider a novel data exploration approach that relies on the use of XML influence space instead of the original definition space of data when there exists an ML model attached to these data.

2. We illustrate this first contribution with a clustering framework for detecting subgroups based on local influences.

3. We extensively evaluate this clustering framework on a representative set of datasets with distinct challenges, along with several local attributive XML methods and clustering techniques, for a wide variety of cluster numbers.

4. We propose an *in-depth* study for the K-medoid clusters' quality to show the efficiency of considering influences space even for misclassified instances and ML models with low-performances.

5. We finally show that the exploration of clusters of explanations is an effective complement to traditional data analysis through a use case in healthcare.

The paper is organized as follows: Section 3 gives an overview of the current local attributive explanation methods and the clustering method families used in experiments. This section also explores how explanations are used to detect subgroups of instances in the literature. Then, section 4

details our clustering framework for detecting subgroups based on local influences. Section 5 describes the experimental protocol performed to evaluate our approach, involving 40 datasets, several clustering techniques and local attributive XML methods. Then, Section 7 discusses about the advantages of our approach in a broader context, linking results from clustering with knowledge from modeling and explanation methods. Section 6 introduces a real medical use case, example of how our approach can be used to support ML prediction and local explanations use. Finally, Section 8 concludes this paper and gives short and long-term perspectives.

2. Motivating example

Let us consider a data scientist whose objective is to explore the SA-Heart dataset[1] to binary predict a coronary heart disease (**CHD**). The dataset includes 462 individuals and 10 attributes defined as follows: **Age** (at the onset), **Adiposity** (estimation of the body fat percentage), **Obesity** (through the body mass index), **LDL** (low-density lipoprotein cholesterol), **Famhist** (family history of heart disease, present or absent), **Tobacco** (cumulative consumption tobacco), **SBP** (systolic blood pressure), **alcohol** (current alcohol consumption), and **type-A** (Type-A behavior scale).

The objective of the data scientist is to produce a clustering (and the appropriate metric) highlighting homogeneous subgroups in the space of explanations, thus revealing subgroups of subjects which for the same health status (a coronary heart disease or not) would have several explanation profiles. Following the methodology proposed in [18], the local influences for each instance were computed as follows: (i) a MLP classifier is trained (excluding the *type-A* attribute as performances were better without it), and (ii) a KernelSHAP XML model is used to compute the local explanations. The final model has a final accuracy of 0.77.

This problem can be illustrated by selecting pairs of instances, in Table 1. Table 1 identifies five pairs of patients who are consistently the closest instances in the original space, as evidenced by their proximity to one another (by Euclidean distance, below 3). In contrast, the mean distance between

[1] https://www.kaggle.com/datasets/emilianito/saheart

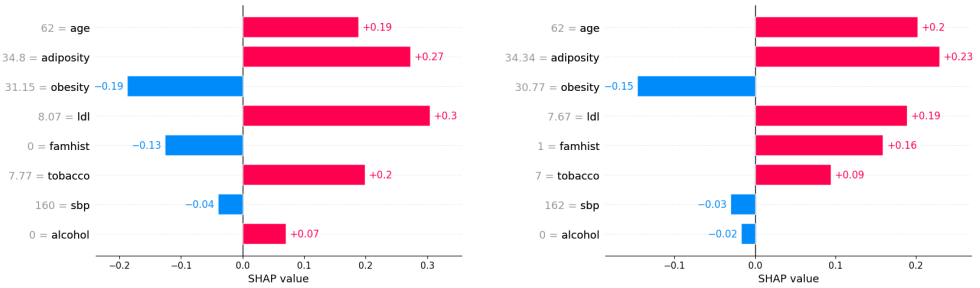

Figure 1: Illustration of similar instances (Pair 5) for which the XML attribute explanation space introduces significantly different representations.

instances in this dataset is approximately 45.5 ± 24.5. Whereas explanations are roughly similar within the first 4 pairs, pair 5 offers two different explanations (Figure 1). This highlights the fact that close instances in the data space can result in different explanations and that considering the explanation space provides a relevant angle of view on both the data and how that data has been processed by the ML algorithm. XML influences may consequently be a more discriminant space for clustering.

	Age	Adiposity	Obesity	LDL	Famhist	Tobacco	SBP	Alcohol	Distance
Pair 1	18	13.39	22.01	2.46	0	0	120	0.51	**1.48**
	18	13.35	23.37	2.77	0	0	120	1.03	
Pair 2	20	17.15	22.76	2.69	1	0.61	124	11.55	**2.03**
	20	16.64	22.26	3.74	1	1.8	124	10.49	
Pair 3	17	15.7	22.03	2.81	0	0	127	1.03	**2.15**
	17	15.11	22.17	3.95	0	0.21	126	2.42	
Pair 4	17	13.15	20.75	2.43	1	0	128	0	**2.22**
	17	12.51	20.28	1.88	1	0	130	0	
Pair 5	62	34.80	31.15	8.07	0	7.77	160	0	**2.47**
	62	34.34	30.77	7.67	1	7	162	0	

Table 1: Example of five pairs of instances from SA-Heart dataset. Each pair groups two patients who are very similar as attested by their descriptive attributes and class membership (i.e. pathology)

3. Related works

Our proposal relates to different fields of data analysis and exploration. One major axis of study concerns so-called explainable approaches [19], first

detailed in this section. A special emphasis is put on local attributive approaches such as SHAP [12] or LIME [11] that produce an influence score for each attribute describing an instance for a specific ML model. Then as our goal is to study clustering approaches in the influence space, we present a brief overview of the related methods and literature.

Explainable Machine Learning and local attributive methods. Explainable ML refers to the field focusing on the problems of understanding machine learning predictions and closed-box models, with multiple overlapping terminology: *eXplainable ML* (XML), *Interpretable ML* (IML), *eXplainable AI* (XAI), Interpretability, Explainability [20]. The main hypothesis is that more transparent, interpretable and explainable models lead users to understand and trust the intelligent system [21]. There are two main approaches for achieving explainable ML: intrinsically interpretable models and *post-hoc* explanation approaches. Intrinsically interpretable models refer to models understandable due to their inner structure, with components that can be analyzed individually and easily linked to understandable concepts. On the other hand, *post-hoc* explanation methods are methods that can be applied to already trained ML models, to grasp insights about how they work and their reasoning behind the predictions.

The field of intrinsically interpretable models existed before closed-box problems appeared in ML. The idea is to build ML models that are inherently interpretable and understandable, also called glass-box or open-box models. Linear regression, decision trees and rule-based machine learning models fall into this category. Intrinsically interpretable models are often defined as the easiest way to achieve explainability [19] and the use of *post-hoc* explanations over interpretable models in sensitive domains is particularly criticised [22]. Unfortunately, there are limitations to the intrinsic interpretation of these models and their use in modelling complex data. Especially, intrinsically interpretable models can also become *closed-box* models when they are proprietary -i.e. the one creating the model reserves the rights to use, modify or share- or when the complexity of the model increases to model complex data, hindering the interpretation of these models [23]. *Post-hoc* explanations, applicable to all models, are then the only solution currently available to provide explanations to users and attempt to explain the reasons for a prediction.

With *post-hoc* explanation methods, models remain closed-box and the prediction reflect both the attributes (data) and the trained ML model [24].

Models can be studied globally or locally, depending on the goal and the information requested. Global methods aim to describe the overall model behaviour (the role of each attribute over all the instances) while local methods explain the prediction for each instance of the dataset individually (the role of each attribute for each instance). In the field of local post-hoc explanations, one of the first methods was based on the Shapley values, a local attributive XML method [25], to explain ML predictions. The influence of each attribute over a prediction is computed as the difference in prediction from the model with and without the attribute, and represent the impact of each attribute over a prediction for each instance of the dataset. Local influences allow better appropriation without prior data science knowledge as they are easy to interpret and represent graphically. Secondary methods have emerged such as LIME [11] that uses linear surrogate models trained with sampled data to approximate the closed-box model locally. The Coalitional approaches [14] approximate the Shapley value by precomputing relevant groups of instances and reducing complexity. Finally, SHAP [12] mixes Shapley values with LIME and other methods to simulate the absence of attributes by sampling, find a linear model that explains the closed-box model locally and approximate the Shapley values. Nowadays, SHAP framework offers proven and easy-to-use methods, agnostically to the ML model or specific (e.g. KernelSHAP [12] or TreeSHAP [26]).

Analysis of clustering algorithms. According to [27], clustering consists of the unsupervised classification of patterns (being data items, attributes vectors, time series, graphs) into groups called clusters. There are no unique criteria to assess the quality of a grouping. For example, internal criteria such as Davies-Bouldin index [28] ensure that groups are compact and well-separated but impose to shape the clusters as hyper-spheres, similar to the Silhouette index [29]. External criteria such as (Adjusted) Rand Index [30] assess the quality of the grouping with a ground-truth knowledge that is to be known beforehand. Even if an evaluation criterion is known, clustering is an NP-hard problem since one would have to build all partitions for all possible numbers of clusters to determine the best clustering [31]. As such, there exists a large variety of clustering algorithms [32] depending if they produce a disjoint partition of the dataset such as k-means [33] or k-medoid [34], a fuzzy or soft partition [35] or a dendrogram that is a nested set of partitions such as in the hierarchical clustering [34]. In [32], the author identifies new trends for clustering algorithms such as the introduction of semi-supervision

to take into account expert knowledge when available [8, 6]. Other challenges involve dealing with large-scale datasets or streams [36] or proposing efficient co-clustering approaches that build a clustering of instances and attributes at the same time [37]. In our work, we focus first on simple use cases of data exploration, thus avoiding the impact of streams or external constraints on our experiments. Finally, another recent tendency in clustering is related to the use of deep architecture to build end-to-end clustering systems that go from data representation to clustering in a single algorithm. Such is the case of DEC (Deep Embedded Clustering) and its variants [38]. These latter will not be included in this study since they build their own embedding. A clustering method should be associated to a metric that is able to define the topology of the space, hence related to the geometry of the clusters. To preserve a variety of cluster shapes, clustering approaches relying on minimisation of variance in Euclidean space were considered in this study (k-means, k-medoids, hierarchical clustering with Ward's criterion [33, 39, 40]), Gaussian Mixture Models that leverage the constraint of uniform variance of k-means [41] and finally, a density-based algorithm (HDB-SCAN, based on DB-SCAN algorithm) that can find any type of cluster shape [42, 43].

Mixing clustering and local explanations for improving data exploration. Several papers in the last year have covered use cases combining machine learning explainability and clustering to find relationships between instances [18, 44]. Based on a COVID-19 dataset, [18] tries to better identify clusters based on KernelSHAP values. Rather than clustering on the original dataset (raw data), a classification model has been trained, explained by KernelSHAP, and the resulting influences were clustered with HDBSCAN. Graphical interpretation on UMAP and silhouette scores demonstrate the ability to better discriminate between sub-groups using explanations rather than data. Other papers also used clustering to determine groups and to recommend instances based on the influences on a single dataset. [45] explores healthcare risk stratification based on influences from TreeSHAP on a urinary disease dataset. Clustering patients by SHAP values allows the selection of representative patients and investigation of the risk factors for each cluster, where raw data only are insufficient to perform the same analysis. The same kind of analysis was performed on a COVID-19 dataset concerning the identification of subgroups of patients during the first lockdown in France [46]. With clinical and biological data from COVID-19 hospital patients, [46] uncovered the COVID-19 typology of patients to identify those most at risk of aggravation

during their hospital stay. ML combined with Explainability methods was used to highlight the most significant attributes and build an aggravation risk score. Then, clustering techniques on explanations aggregated patients and defined three clusters of patients that appear to be consistent with three distinct risk-score levels. Instance recommendations based on the medoïd of each cluster also allowed an *in-depth* study of each subgroup's characteristics. Although such works have explored the idea of using influences and clustering to extract more knowledge about the data on specific medical examples, none of them formally evaluates the contribution of explanation clustering as a whole. This article is therefore an original contribution to demonstrate the ability of using local attributive XML methods as a generic method to explore subgroups of data.

4. Influence-based clustering framework

Figure 2 shows the step-by-step framework to cluster instances based on their influences:

1. A machine learning model is trained with raw data and predicts classes of all the instances from the raw dataset.

2. A local attributive XML method explains the trained model. Users can choose the data used as input for the method. Influences are computed to explain the determinants of the predictions provided by the ML model.

3. A clustering algorithm is used on influences to create homogeneous groups of instances to detect their core attributes. The proportion of computed clusters can be adjusted by the user.

In this framework, various elements can be tuned according to user preferences. Any classification model can be used at Stage 1, as they are all designed to compute predictions, and Stage 3 allows any clustering method that produces a disjoint partition of the dataset. In Stage 2, the framework is designed to accept local attributive XML methods. These influences are represented as tabular data, where each instance has a value associated with each attribute. We directly use these influences data as input for the clustering stage. Influences provide additional information that the raw data does not: the link between the modeling predictions and the dataset attributes. Compared to raw data, explanations produced by local attributive

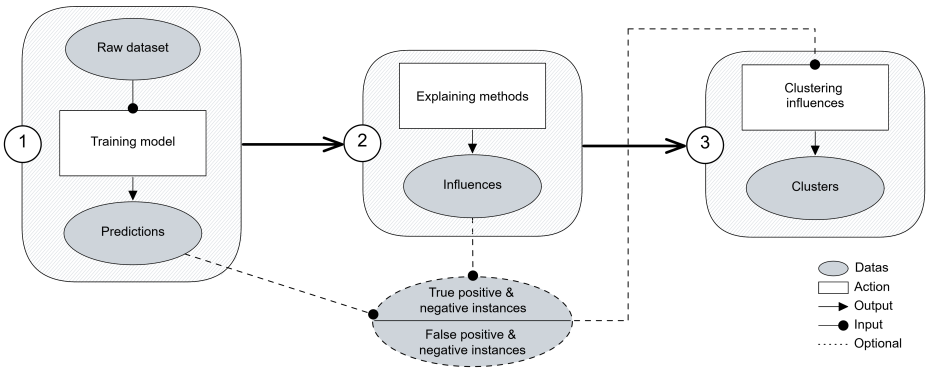

Figure 2: Our proposed Framework for explanation analysis.

XML methods have the same unit across all attributes, thus avoiding any problem of value ranges. Another advantage is that influence values are less noisy since the ML model mainly focuses on attributes relevant to the underlying predictive task and excludes information not explained by the complex attributes interaction, hence the relevance of carrying out clustering. For supervised tasks, local attributive XML methods usually generate a dataset for each class with identical dimensions as the raw data. For example, if the raw data consists of n instances and m attributes and the supervised task is a multi-class problem with c classes, the generated dataset (also called the influence dataset) is shaped as a tensor with $n \times m \times c$ dimensions. To have an influence dataset with the same dimension as raw data ($n \times m$) one can only select a single class and its associated influences. For example, regarding binary classification, the positive class is often chosen as the class of interest for influences.

An additional and optional step is to select a particular subset of the data for clustering. Indeed, it is possible to study the instances correctly and incorrectly classified by the model separately via instance clustering. This option has several advantages. Since the influences represent the model decisions, separating the instances can provide new knowledge. Studying the well-classified instances can help to identify their characteristic patterns by removing noise and outliers from the misclassified instances. This can give a more accurate idea of general patterns, for example, to check that there is no bias in the dataset. Regarding misclassified instances, they may cover different realities. They can represent real outliers, data whose variability may be intrinsic to the instance or indicative of an error in data acquisition. How-

ever, it may be a specific sub-population of the dataset whose frequency is too low to be significant in relation to the current number of instances of the dataset. This configuration is common in medicine, where inter-individual variability is often substantial and the size of data sets often modest. Separating the instances can, therefore, allow the exploration of new patterns that can be invisible if all the data were kept. This may be even more important for influences because of their direct link to the model. Indeed, when the model prediction is of low performance, the influences reflect this error and are directly impacted by the wrong prediction of the model.

The full implementation of our proposal is available here: https://github.com/kaduceo/XAI-based-instance-selection. The source code will evolve with future works. Additional materials are also available.

5. Experiments

This section details the experiments carried out to test our framework and, more generally, assess explanation clustering. We derive several research questions (RQ) to show whether clustering explanations produce better-quality clusters than raw data clustering (see Section 5.2.7):

- (**RQ1**): What is the sensitivity of our framework to the XML method? (see Section 5.2.1)
- (**RQ2**): What is the sensitivity of our framework to the clustering method? (see Section 5.2.2)
- (**RQ3**): Are there some best combinations of ML models and XML methods? (see Section 5.2.3)
- (**RQ4**): What is the sensitivity of our framework to the accuracy of the ML models? (see Sections 5.2.4 and 5.2.6)
- (**RQ5**): How size and dimensionality of datasets impact clustering? (see Section 5.2.5)

Based on these questions, we expect clustering in XML influence space will be valuable for all clustering techniques and local XML methods included in the experiment when compared to raw data clustering. We hypothesize that this approach is also relevant for low-performance models, and that clustering on misclassified instances can still provide insightful cluster information. Finally, we speculate that Shapley-based XML methods and Spearman

coalitional will exhibit better performance for clustering explanations than LIME as they should be able to capture more information from more complex ML models.

5.1. Experimental protocol

Clustering algorithms. Five clustering techniques have been selected for comparison to be representative of the main clustering families: two partitioning clustering (namely k-means and k-medoids), one hierarchical clustering (hierarchical agglomerative clustering), one density-based clustering (DB-SCAN) and one model-based clustering (Gaussian mixture model). As both raw data and influences data are tabular data with the same dimensions, we use the exact same implemented framework for our experimental comparisons, the only difference being the use of the initial *raw* space or the alternative XML representation space.

K-medoids algorithm [39] assigns data to k clusters iteratively based on their distance to a centroid point. This central point is always an instance from the dataset. Each iteration tries to maximize the distance between points from different clusters and minimize the intra-cluster distance. The number of clusters k is pre-defined. In the experiments, we chose the Euclidean distance. We also use *k-means* algorithm [33]. We can expect some differences as prototypes representative of clusters may not necessarily be part of the original instances with k-means. However, due to the continuous representation of prototypes, k-means can reach better compactness and separability between clusters when compared to k-medoids at the expense of the interpretability of cluster prototypes.

For these two clustering methods, to ensure the stability of the clustering, we use *k-means++* and *kmedoid++* for the initialization, which relies on a Monte-Carlo approach to decide on the selection of initial cluster centers. Based on these initialization methods, results were very consistent across runs. For this reason, only one result by clustering method was presented even though there might exist a small variability due to the non-deterministic nature of the choice of the initial clusters.

Agglomerative Nesting (Agnes) is a hierarchical clustering method [34]. Hierarchical clustering creates a hierarchy of nested clusters, and therefore, a pre-specified number of clusters is not required when the complete hierarchy is required. A number can be specified to extract the clusters from one level of the hierarchy. Agglomerative clustering is a bottom-up approach: at first, each instance is considered as a single-element cluster, and at each

iteration, the two most similar clusters are combined. The similarity between elements is based on the Euclidean distance, consistently to what is done for previous partitioning approaches, and cluster are merged with the Ward linkage criterion which aims at minimizing the variance of produced clusters.

HDBSCAN [43, 47] is a popular hierarchical density-based clustering technique. Based on a density measure for each instance defined as a number of instances in an ϵ-neighborhood centered around that instance, HDBSCAN groups together the points where the density is high (i.e., the points closely packed that have many neighbors). HDBSCAN performs multiple iterations of clustering for all possible density scales. This allows the detection of meaningful clusters in data of varying densities and the robustness to parameter selection, as opposed to the traditional DBSCAN [42]. HDBSCAN is stable over runs and resistant to noise and outliers. However, this technique detects automatically the number of clusters which makes it inappropriate for our test protocol where we evaluate iteratively different number of clusters. Moreover, HDBSCAN does not necessarily include all instances in clusters, some points being considered as noise. This raises questions about comparison with other approaches in our protocol. In conclusion, although we have run some preliminary tests with HDBSCAN, we do not consider this approach in this paper for the aforementioned reasons.

Gaussian- mixture model clustering algorithm (GMM) is an instance of Expectation Maximization clustering by [41] to cluster points based on statistical modeling and data distribution. EM clustering assigns data points to clusters iteratively to maximize the overall likelihood of the data. Unlike other clustering methods, EM is a soft clustering technique: each point has a probability of belonging to each cluster rather than a single assigned cluster. In our case, we assign instances to the cluster with the highest probability. We use the Gaussian mixture model in our test, which assumes that each attribute in each cluster is the observation of a Gaussian random variable. This version is a generalization of the k-means algorithm.

Datasets and classification task. We use 40 datasets from an Open ML collection[2] [48] that meet the following criteria: binary classification, more than 100 instances, more than four attributes and at most nine attributes due to the computational cost of producing influences. Table 2 provides descriptive statistics about the included datasets.

[2]Available in `https://www.openml.org/s/107/tasks`

Table 2: Descriptive statistics of the study datasets.

Number of attributes	5	6	7	8	9	10	All
Number of datasets	8	12	6	3	6	5	40
Mean number of instances	529	1229	1026	225	1128	818	917
Min number of instances	125	100	100	137	310	286	100
Max number of instances	1372	5404	3107	379	4177	1473	5404

Binary classification is chosen to facilitate the interpretation of influences. Indeed, for example, with SHAP values, influences for one class are the opposite of influences for the second class in play. For the sake of simplicity, we consider that all influences are based on class 1. In this case, influences represent the impact of each attribute on the probability of the instance being in class 1. We train a Random Forest model (RF) with a Grid Search Cross-Validation to optimise hyperparameters. This model was chosen to test tree-specific explanation methods while keeping a limited number of hyperparameters to avoid overfitting (compared to boosted trees). Only to evaluate the performance of the models, each dataset is divided into train and test sets according to the 75%/25% ratio. Table 3 shows the performances of all the models trained in our experiments. Models are trained adequately to capture most information of the dataset. The mean balanced accuracy is 0.88, meaning most models can accurately classify test instances. When we separate models based on an accuracy threshold set to 0.8, high-accuracy models have a median balanced accuracy of 0.98, whereas low-accuracy models have a median of 0.66.

Table 3: **Statistics of models trained**. Balanced accuracy and percentages of well-classified and misclassified instances are presented for the 40 datasets and separately based on the 0.8 accuracy threshold. For well-classified and misclassified instances, the median number of instances is presented along with the percentage.

	Balanced Accuracy			**% of Well-classified**	**% of Misclassified**
Models (#)	Median	Min	Max		
All (40)	0.88	0.50	1.0	85.6%	14.4%
Acc $\geq$ **0.8** (26)	0.98	0.83	1.0	94%	6%
Acc $<$ **0.8** (14)	0.66	0.50	0.79	72%	28%

We also study the number of instances well classified and misclassified

by the ML modeling in Table 3. We use three different data separations: all instances together, only well-classified instances, and only misclassified instances. For the experiments about (**RQ4**), as we separate well-classified and misclassified instances, we choose not to evaluate high-accuracy models on misclassified instances as there are not enough instances in most datasets to create clusters and properly evaluate them and compare the results. Then, when studying misclassified instances, we only work with models with low accuracy as the number of misclassified instances is higher and sufficient. Note that the number of well-classified instances is adequate for performing clustering for all models.

Explainability methods. We choose three different local attributive XML methods, which provide unique advantages and disadvantages to compute local influences [49]. SHAP, and more specifically its KernelSHAP implementation [12], is the reference method based on game theory, but suffers from a high time complexity due to the exploration of attributes power set and a potential bias with correlated attributes. Another SHAP implementation, TreeSHAP [26], provides the fastest computation for tree-based prediction models. In contrast, LIME [11] approximates influences following a linear local model that does not rely on the same game theory inspired kernel as SHAP. Finally, Spearman coalitional [14] improves over SHAP and LIME by taking into account potential correlations between attributes while reducing the size of the power set to explore with coalitions.

Setting the number of clusters. To define the number of clusters we use multiple proportions of the total number of instances in the dataset as the number of clusters. The following proportions were used: 0.01, 0.02, 0.03, 0.04, 0.05, 0.1. The number of clusters is then $n_{cluster} = p * n_{instances}$ with p the selection proportion between 0 and 1 with a minimum number of two clusters. As the size of the datasets greatly varies (see Table 2), a proportion rather than fixed numbers of instances was preferred to take into account the diversity of the datasets. As the aim is to study the comparative behavior of clustering on influences and raw data, multiple proportions per dataset can show how cluster quality evolves without looking for an "optimal" number of clusters (which may be different depending on the clustering method).

Comparison to ground-truth labels. Finally, to evaluate if clusters are well-defined and manage to group similar instances and separate dissimilar instances based on their *a-priori* labels, the *Entropy* metric was computed

[50]. There exists many external criterion to measure the agreement of a clustering relatively to a set of existing classes [51]. We use entropy as we expect XML representation space to produce clusters aligned with ideally only one original class, but we do not want to penalize an over-segmentation of the initial classes into more clusters. Indeed, we expect to highlight meaningful and more specialized patterns than what can be observed in the raw data as illustrated in Section 6. Entropy measures the distribution of labels in a cluster, i.e., the ability of the algorithm to differentiate between data that do not have the same "real" class. A perfect entropy means all instances from the same class are in the same clusters (lower entropy means better clustering).

$$Entropy = \sum_{k=1}^{K} \frac{n_k}{n} \Big(- \frac{1}{\log q} \sum_{i=1}^{q} \frac{n_k^i}{n_k} \log \frac{n_k^i}{n_k} \Big)$$

where C_k is a particular cluster of size n_k, q is the number of class in the dataset, K the number of clusters and n_k^i is the number of instances of the ith class assigned to the kth cluster.

5.2. Results

As expected, increasing the proportion of clusters leads to a slight decrease in entropy values (in this case lower values are better), reflecting more homogeneous clusters. In Table 4, most clustering approaches show a decrease in entropy when clustering percentage increases from from 0.01 to 0.1. Using explanations also leads to better clustering than using raw values, underlining the value of using explanations for less noisy data mining.

5.2.1. Spearman outperforms LIME and to an extent SHAP

Compared to Spearman and SHAP, LIME performs worse (see Table 4), offering clusters with higher entropy. This is expected as previous studies have already pointed out some limitations of LIME, such as the definition of a proper neighborhood to learn XML influences [19], the sensitivity to the non-linearity of the process to be learned or the instability of explanation from one instance to the closest one [52]. All of this causes explanations representation of instances from different classes to be close, or conversely instances from the same class to have different XML influence representations, which in turn, causes a more balanced representation of classes in each cluster.

		Clustering Method			
XML	Clustering Percentage	agnes *mean (std)*	gmm *mean (std)*	kmeans *mean (std)*	kmedoid *mean (std)*
Raw	0.01	0.59 (0.24)	0.47 (0.29)	0.52 (0.22)	0.54 (0.24)
	0.02	0.59 (0.25)	0.56 (0.27)	0.57 (0.24)	0.57 (0.24)
	0.03	0.57 (0.26)	0.55 (0.28)	0.57 (0.25)	0.57 (0.25)
	0.04	0.56 (0.27)	0.55 (0.28)	0.53 (0.25)	0.54 (0.26)
	0.05	0.54 (0.27)	0.52 (0.27)	0.52 (0.26)	0.54 (0.27)
	0.10	0.48 (0.28)	0.48 (0.27)	0.46 (0.26)	0.47 (0.26)
LIME	0.01	0.51 (0.25)	0.7 (0.28)*	0.46 (0.25)	0.46 (0.26)
	0.02	0.5 (0.25)	0.66 (0.29)	0.49 (0.25)	0.5 (0.25)
	0.03	0.49 (0.25)	0.68 (0.26)	0.47 (0.25)	0.48 (0.24)
	0.04	0.49 (0.26)	0.7 (0.25)*	0.47 (0.25)	0.48 (0.25)
	0.05	0.48 (0.25)	0.68 (0.23)*	0.47 (0.25)	0.48 (0.25)
	0.10	0.43 (0.26)	0.72 (0.23)*	0.43 (0.26)	0.43 (0.25)
SHAP	0.01	0.38 (0.27)*	0.58 (0.36)	0.31 (0.27)*	0.32 (0.28)*
	0.02	0.35 (0.28)*	0.7 (0.28)	0.31 (0.28)*	0.32 (0.28)*
	0.03	0.32 (0.27)*	0.74 (0.24)*	0.31 (0.27)*	0.32 (0.27)*
	0.04	0.31 (0.27)*	0.66 (0.28)	0.3 (0.27)*	0.31 (0.27)*
	0.05	0.3 (0.28)*	0.7 (0.25)*	0.29 (0.26)*	0.3 (0.27)*
	0.10	0.26 (0.26)*	0.72 (0.25)*	0.26 (0.26)*	0.26 (0.26)*
Spearman	0.01	0.33 (0.25)*	0.51 (0.3)	0.31 (0.26)*	0.32 (0.26)*
	0.02	0.34 (0.26)*	0.6 (0.3)	0.31 (0.25)*	0.32 (0.25)*
	0.03	0.32 (0.27)*	0.61 (0.34)	0.32 (0.28)*	0.33 (0.28)*
	0.04	0.31 (0.25)*	0.64 (0.32)	0.31 (0.26)*	0.32 (0.26)*
	0.05	0.3 (0.26)*	0.68 (0.28)*	0.3 (0.26)*	0.31 (0.26)*
	0.10	0.26 (0.24)*	0.66 (0.26)*	0.26 (0.25)*	0.27 (0.25)*

Table 4: Influence of clustering method, percentage, and XAI method on cluster entropy (fixed ML model to RF). Results are presented as mean (standard deviation). A * right of the standard deviation means the t-test rejected the null hypothesis (alpha = 0.05) that the XAI method performed as well as the baseline Raw on average (Welch correction was applied to account for unequal variance and Bonferroni correction to account for multiple testing).

5.2.2. Kmeans and kmedoid yield better results

Regarding the clustering method, the performance of GMM is the worst, followed by Agnes. The GMM clustering results may be limited by the

relatively low number of data instances relative to the larger number of parameters to learn. GMM may also be more sensitive to the difference in the range of attribute values between raw and the XML spaces as it learns their relative importance to properly estimate likelihood. Agnes exhibits performance close to k-means and k-medoid although a bit below for a low number of clusters (i.e. the strongest aggregation in the partition trees). This result is coherent since we use the Ward criterion which minimizes cluster variance similar to k-means and k-medoids. Finally, kmeans and kmedoid offer similar entropy values. Since k-medoid clustering can be represented by the dataset instances being the medoids of the different clusters (and not by the k-means calculated centroids), the following experiments will be conducted using k-medoid.

5.2.3. More complex ML models and XML methods yield better results

The proportion of clusters has little influence on cluster entropy depending on the type of ML model (Table 5). However, the quality of clustering is improved when a more complex XAI method or ML model is used (Spearman and SHAP versus LIME, RF versus LR, respectively).

As shown in Table 5, the results of linear models such as LR or SVM are less stable when compared to RF and do not provide as much improvement when compared to raw representation space. We hypothesize that this is expected since RF is a more expressive model that corresponds to multiple linear models when compared to single linear models in LR and SVM. As a consequence, RF captures more complex class structure from the initial representation space and XML influences attached to this model may identify more discriminant attributes whose information is otherwise more diluted with less expressive models such as LR and (linear) SVM.

In conclusion, SHAP and Spearman methods with an RF-type ML method, produce more homogeneous clusters, facilitating the exploration of instances sharing similar explanations.

5.2.4. Low accuracy models do impact clustering performance

Table 6 presents the entropy values for different clustering percentage for our three XML methods when compared to raw representation space in two distinct scenarios: Table 6-top details results for accurate model (whose

		ML Model		
XML	Clustering Percentage	LR	RF	SVM
Raw	0.01	0.54 (0.24)	0.54 (0.24)	0.54 (0.24)
	0.02	0.57 (0.24)	0.57 (0.24)	0.57 (0.24)
	0.03	0.57 (0.25)	0.57 (0.25)	0.57 (0.25)
	0.04	0.54 (0.26)	0.54 (0.26)	0.54 (0.26)
	0.05	0.54 (0.27)	0.54 (0.27)	0.54 (0.27)
	0.10	0.47 (0.26)	0.47 (0.26)	0.47 (0.26)
LIME	0.01	0.53 (0.26)	0.46 (0.26)	0.49 (0.24)
	0.02	0.56 (0.24)	0.5 (0.25)	0.53 (0.25)
	0.03	0.55 (0.26)	0.48 (0.24)	0.51 (0.25)
	0.04	0.56 (0.27)	0.48 (0.25)	0.5 (0.25)
	0.05	0.55 (0.27)	0.48 (0.25)	0.51 (0.25)
	0.10	0.51 (0.26)	0.43 (0.25)	0.47 (0.26)
SHAP	0.01	0.43 (0.29)	0.32 (0.28)*	0.34 (0.28)*
	0.02	0.47 (0.28)	0.32 (0.28)*	0.33 (0.28)*
	0.03	0.47 (0.29)	0.32 (0.27)*	0.35 (0.3)*
	0.04	0.46 (0.29)	0.31 (0.27)*	0.32 (0.29)*
	0.05	0.47 (0.3)	0.3 (0.27)*	0.32 (0.29)*
	0.10	0.41 (0.29)	0.26 (0.26)*	0.3 (0.29)*
Spearman	0.01	0.44 (0.28)	0.32 (0.26)*	0.43 (0.29)
	0.02	0.49 (0.27)	0.32 (0.25)*	0.46 (0.27)
	0.03	0.5 (0.29)	0.33 (0.28)*	0.45 (0.27)
	0.04	0.49 (0.29)	0.32 (0.26)*	0.44 (0.28)
	0.05	0.5 (0.29)	0.31 (0.26)*	0.44 (0.28)
	0.10	0.43 (0.28)	0.27 (0.25)*	0.38 (0.26)

Table 5: Influence of ML model, clustering percentage and XAI method on cluster entropy (fixed clustering algorithm to kmedoid). Results are presented as mean (standard deviation). A * right of the standard deviation means the t-test rejected the null hypothesis (alpha = 0.05) that the XAI method performed as well as the baseline Raw on average (Welch correction was applied to account for unequal variance and Bonferroni correction to account for multiple testing).

accuracy is above or equal 0.8) and Table 6-bottom details results for inaccurate models (whose accuracy is under 0.8). It can be seen that SHAP and Spearman still perform best when compared to LIME. Raw is the worst rep-

resentation space for clustering in both scenarios. Finally, it can be clearly observed that SHAP and Spearman do perform better on accurate models with a best score around 0.2 (clustering perc. equals to 0.1) while performances decrease with an entropy score above 0.3 for inaccurate models.

Model accuracy over 0.8

Clustering Perc.	**0.01**	**0.02**	**0.03**	**0.04**	**0.05**	**0.1**
RAW	0.5 (0.25)	0.55 (0.25)	0.53 (0.26)	0.5 (0.25)	0.5 (0.26)	0.42 (0.25)
LIME	0.43 (0.25)	0.46 (0.24)	0.43 (0.24)	0.42 (0.25)	0.42 (0.25)	0.36 (0.25)
SHAP	0.28 (0.26)	0.27 (0.27)	0.25 (0.24)	0.24 (0.24)	0.23 (0.24)	0.2 (0.23)
Spearman	0.27 (0.23)	0.28 (0.22)	0.28 (0.26)	0.26 (0.23)	0.25 (0.24)	0.22 (0.23)

Model accuracy under 0.8

Clustering Perc.	**0.01**	**0.02**	**0.03**	**0.04**	**0.05**	**0.1**
Raw	0.59 (0.23)	0.62 (0.24)	0.63 (0.25)	0.6 (0.27)	0.61 (0.28)	0.54 (0.27)
LIME	0.52 (0.27)	0.56 (0.25)	0.55 (0.25)	0.56 (0.25)	0.57 (0.24)	0.53 (0.23)
SHAP	0.39 (0.3)	0.39 (0.31)	0.42 (0.29)	0.41 (0.29)	0.4 (0.29)	0.35 (0.28)
Spearman	0.39 (0.31)	0.38 (0.29)	0.39 (0.3)	0.4 (0.3)	0.38 (0.3)	0.33 (0.28)

Table 6: Entropy for inaccurate and accurate ML models and the different XAI approaches using k-medoids clustering. Results are presented as mean (standard deviation).

5.2.5. Dataset size and dimensionality matter

Table 7 considers the typology of datasets according to their number of instances and attributes (considered as high or low). Again, increasing the proportion of clusters results in more homogeneous clusters with lower entropy. Interestingly, both the number of attributes and the number of instances impact the quality of clusters. In the case of datasets with a large number of attributes, including additional instances seems to be advantageous for a percentage of clusters above 0.03. Conversely, this approach is more beneficial in the context of datasets with a relatively limited number of attributes and a low number of instances. Noticeably, our best results are obtained in this context, which can be justified by the fact that this corresponds to the most straightforward case for our RF model, which can learn and generalize from very few training instances when dimensionality is reduced.

In conclusion, the quality of clustering using explanations is strongly influenced by informational adequacy when the number of attributes is consistent with the sample size.

Clustering Perc.	0.01	0.02	0.03	0.04	0.05	0.1
Var_high+Inst_high	0.38 (0.27)	0.37 (0.28)	0.36 (0.27)	0.34 (0.27)	0.33 (0.26)	0.29 (0.24)
Var_high+Inst_low	0.35 (0.30)	0.37 (0.33)	0.38 (0.31)	0.40 (0.32)	0.39 (0.31)	0.33 (0.29)
Var_low+Inst_high	0.34 (0.27)	0.31 (0.26)	0.29 (0.26)	0.27 (0.25)	0.26 (0.25)	0.22 (0.25)
Var_low+Inst_low	0.22 (0.24)	0.24 (0.23)	0.25 (0.20)	0.23 (0.19)	0.23 (0.20)	0.19 (0.22)

Table 7: Entropy for the different typology of datasets using k-medoids clustering, with RF model, SHAP, and trained on all instances versus the clustering percentage. The cut-off of high/low levels of attributes and instances was set at 7 and 500, respectively. Results are presented as mean (standard deviation).

5.2.6. Spearman is more resilient to variability in quality of the models

Table 8 considers the explanations provided by well or miss-classified instances. Compared to Table 4 (k-medoids), all methods have better entropy: it is easier to analyze instances according to whether or not the ML model has predicted their class correctly. However, using raw data or LIME explanations leads to higher entropy clustering. In a surprising way, Spearman provided clusters with low entropy both for well-classified and misclassified instances, while SHAP only provided clusters with low entropy for well-classified instances. This result concerning Spearman explanation method has to be confirmed by future research work. It is now hypothesized that a coalition is able to maintain superior discriminant information by addressing correlations, which assists in clustering even misclassified instances.

Well-classified instances

Clustering Perc.	0.01	0.02	0.03	0.04	0.05	0.1
Raw	0.38 (0.22)	0.43 (0.24)	0.43 (0.23)	0.4 (0.22)	0.38 (0.21)	0.31 (0.2)
LIME	0.33 (0.19)	0.32 (0.18)	0.31 (0.17)	0.31 (0.19)	0.3 (0.18)	0.25 (0.14)
SHAP	0.14 (0.16)	0.12 (0.14)	0.1 (0.11)	0.09 (0.09)	0.07 (0.07)	0.04 (0.06)
Spearman	0.16 (0.16)	0.16 (0.16)	0.14 (0.15)	0.12 (0.12)	0.12 (0.12)	0.08 (0.08)

Misclassified instances

Clustering Perc.	0.01	0.02	0.03	0.04	0.05	0.1
Raw	0.32 (0.22)	0.33 (0.22)	0.34 (0.22)	0.33 (0.21)	0.34 (0.2)	0.36 (0.21)
LIME	0.3 (0.23)	0.3 (0.23)	0.3 (0.22)	0.31 (0.22)	0.31 (0.22)	0.35 (0.22)
SHAP	0.26 (0.2)	0.25 (0.2)	0.26 (0.21)	0.25 (0.2)	0.26 (0.19)	0.22 (0.2)
Spearman	0.12 (0.14)	0.12 (0.14)	0.11 (0.11)	0.1 (0.11)	0.08 (0.1)	0.09 (0.1)

Table 8: Entropy for well-classified and misclassified instances and the different XAI approaches using k-medoids clustering. Results are presented as mean (standard deviation).

5.2.7. Statistical significance of findings

To sum up, Table 9 shows that the combined contribution of the chosen explanation method (including raw values) and the ML model has a strong influence on the quality of the clustering obtained. The best-performing explanation methods combined with more complex ML methods provide, by far, the best performance. On the other side of the spectrum, using LIME or raw values gives higher entropy clustering, whatever the ML model used (Figure 3). When now considering the combination of the explanation method and the type of clustering, using SHAP with kmeans or kmedoid generally provides better quality clusters.

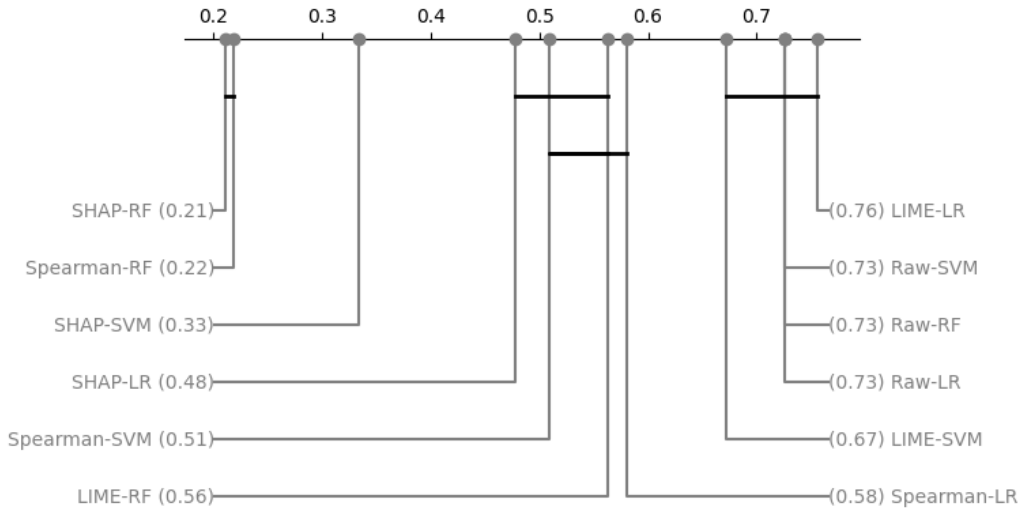

Figure 3: Critical rank diagram (see [53]) of Entropy comparing Raw and XAI approaches combine with various ML models (fixed clustering algorithm to kmedoid).

6. Medical Use Case: SA-Heart explanation clustering

This use case illustrates how this framework can be used on a medical dataset, to help classify patients, understand these categories and retrieve information from explanations. The example is based on the same dataset introduced in Section 2, using the same MLP model and KernelSHAP explanations.

Figure 4 displays the distribution of the obtained local explanations. Age, Tobacco and Family history are the three most important attributes for the

XAI Method	Clustering	Count
Raw	agnes	0
	kmeans	0
	kmedoid	3
LIME	agnes	5
	kmeans	9
	kmedoid	6
SHAP	agnes	72
	kmeans	91
	kmedoid	85
Spearman	agnes	51
	kmeans	58
	kmedoid	57

Table 9: Number of times each XML method performs with the lowest entropy (ties allowed) comparing the four clustering methods over the 40 datasets. In the event of a tie, each XML method gets one point.

model as per KernelSHAP. Based on the influences, a high value in these three attributes - i.e. old age, high consumption of tobacco or family history of heart disease - is associated with a higher prediction, a higher risk of coronary heart disease. For the other attributes, correlations between attribute and influence values are less obvious.

We applied hierarchical agglomerative clustering to the explanations. Hierarchical clustering allows us to think about nested groups, offering greater granularity when exploring populations. The used clustering algorithm is available in the *Scipy* library [3] and we select the *Ward criterion* [40] to aggregate clusters in the hierarchy. The optimal number of clusters was set to 5, based on the *L-method* [54], automation of the *Elbow method* for hierarchical clustering.

Figure 5 shows the mean attributes' importance for each cluster, based on the local influences of the instances in each cluster. Clusters 1 and 5 stand out clearly from the other clusters in terms of the mean importance of Age attributes for both clusters and tobacco for Cluster 1. However, this representation is insufficient to analyse the produced clusters.

[3] `https://docs.scipy.org/doc/scipy/index.html`

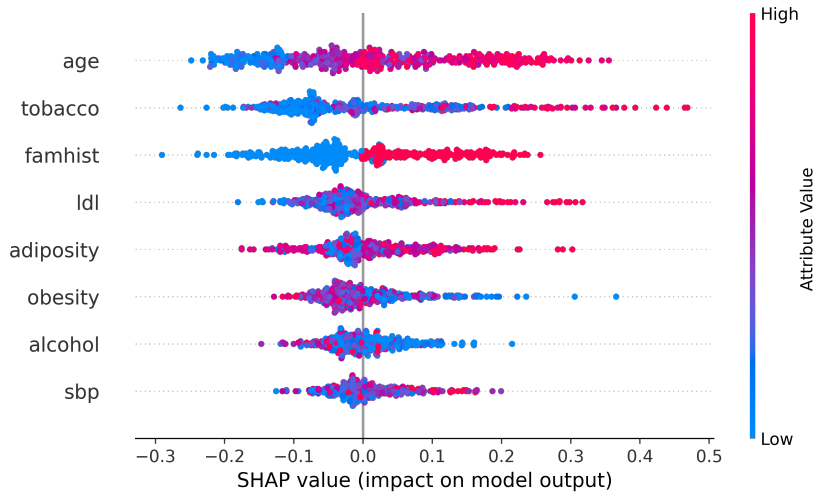

Figure 4: Distribution of KernelSHAP influences for each attribute over the dataset. The attributes are sorted in decreasing mean attribute importance from top to bottom and each dot represents an instance from the dataset, its colour representing the raw value of the attribute. The position on the x-axis represents the contribution of the attribute to the prediction of this individual, and overlapping dots are spread on the y-axis.

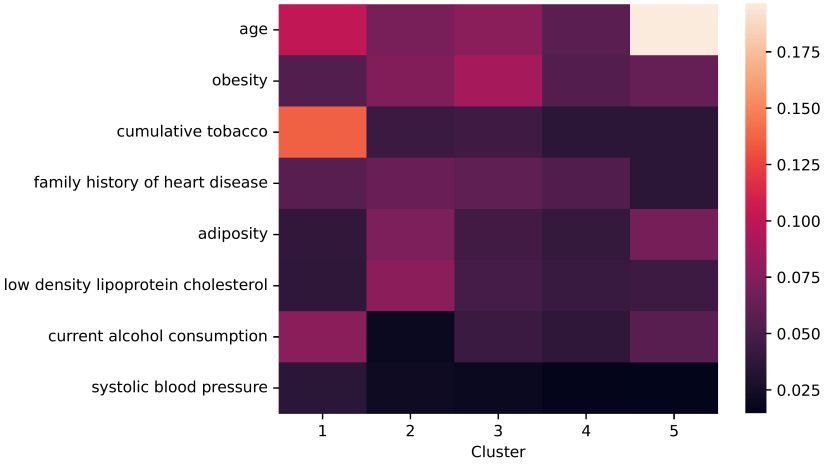

Figure 5: Heatmap of the mean importance of each attribute for each cluster

For a better understanding, the Skope-Rules [55] library was used to compute decision rules (such as a decision tree) for each cluster, to maximise the precision and recall of the rules. Perfect precision means that all instances

of the cluster respect the rule, and perfect recall that all instances respecting the rule belong to the cluster. The rules for each cluster (each cluster is defined based on the explanations) were defined based on the raw data, so that the rules can be read and understood with patient data.

Table 10 displays the rules obtained with Skope-rules on the 5 clusters. For each cluster, the correct classification rate of the model for this cluster and the mean prediction were presented. This information is valuable for both a Data Scientist and a medical expert as model performances reflects how easy it is for the ML model to understand the data for a given task. In the rules, Age seems to be an influential factor in dividing patients, followed by Cumulative tobacco and Obesity. This is consistent with the analysis from Figures 4 and 5 for Age and Tobacco, especially in Clusters 1 and 5. Based on the well-classification rate and the mean prediction, the model seems to better predict where the risk of coronary heart disease is low. Differences in performance between groups could indicate complex sub-groups in the data and a specific lack of data in these sub-groups. It could reveal the limitations of the model, like biases, and could raise fairness and ethical questions about the model [56]. In our case, the model performs better on patients under 23 than on other patients, even though their number is low (61 patients). This cluster highlights the obvious lower-risk subgroups, with patient characteristics that could explain the model's better performance: young patients with low BMI. For an expert in the field, these clusters can represent the major statistical trends in the population studied and simplified relationships relating to coronary heart disease.

Other examples of how explanation clusters can be used are available on the GitHub mentioned in Section 4.

7. Discussion

Clustering on XML influences showed better results than clustering on raw data, regardless of the percentage/number of clusters or the performance of the modeling, especially for Shapley-based XML methods. This behavior was seen for multiple clustering techniques and multiple ML models. This highlights the feasibility of exploring explanations through clustering. The influences seem to contain information allowing a better clustering, probably by highlighting the most significant attributes for each instance or removing noises from raw data. This finding seems consistent with the results of [18] while showing a more global approach, working with other XML meth-

Table 10: Decision rules for each cluster

Cluster	Rules	Well-classification rate	Mean prediction
1	cumulative tobacco >9.825 obesity >19.385 age >42.0	73.3%	66.7%
2	cumulative tobacco $\leq$ 10.85 low density lipoprotein cholesterol >5.065 age >37.5	67.5%	52.5%
3	low density lipoprotein cholesterol $\leq$ 4.76 obesity $\leq$ 24.33 age >43.5	71.4%	40.8%
4	current alcohol consumption $\leq$ 61.815 age $\leq$ 40.5 age >22.0	79%	20.3%
5	obesity $\leq$ 27.185 age $\leq$ 23.5	96.7%	3.28%

ods than KernelSHAP and 40 datasets of various number of attributes and instances.

Separating the instances correctly and incorrectly classified by the model also gives better results than keeping all the instances together. Since the information in the two subgroups is different, they each seem to create noise in the information of the other subgroup. Indeed, the misclassified instances are often outliers or critical instances in the dataset. Their behavior is different from the general behavior of the data, whereas correctly classified instances follow the behavior that the model detects. However, as some misclassification may result from bias in a subgroup of the data or from the atypical behavior of that subgroup compared to the whole dataset, it is of great interest to study them as a priority. When separating correctly and incorrectly classified instances, the differences in cluster quality seem to be more pronounced with the Spearman coalitional method than with KernelSHAP. The contribution seems to depend on the XML method used, probably because of the calculation of influences since KernelSHAP creates perturbations on the instances and Spearman coalitional keeps the input data as it is. A limit to these subgroups' separation is also the decrease of its relevance when the accuracy of the model increases. Indeed, the number of false instances logically decreases with increasing accuracy. Creating an XML model and clusters with a low instance count does not make sense and can only lead to data

misunderstanding. However, as the accuracy increases, the false instances become mostly outliers of the dataset or biased instances rather than sub-groups with their behaviors to analyze. Their small number can be analyzed manually without any particular clustering method.

Explanations clustering, being better than Raw clustering, also emerges when focusing on the optimal number of clusters for each dataset, clustering technique, and explanation type. SHAP and Spearman appear to be the best local explanation methods to perform explanation exploration. In this setup, LIME again produces worse results than the other explanation methods, suggesting that LIME influences are not suitable for clustering and data exploration through clustering. Shapley-based methods then seem more reliable for exploration, as SHAP also performs well with all clustering techniques.

Finally, the proposed approach also adds another use of influences. Clusters based on influences can be used to focus on sub-groups of data to be studied. Clustering can be combined with other approaches to understand the clusters created, like rule-based algorithms or instance selection. As mentioned before, the inner properties of each clustering technique can be used to explore clusters. Hierarchical clustering can be valuable for exploring similar influences and instances or how influences behave with different numbers of clusters/on different hierarchical levels. The data distribution and variance in each cluster from GMM clustering can explain what attributes are important for each cluster and may explain how clusters are built. Medoids from K-medoids can be used to select representative instances and summarise each cluster. Density-based approach may allow the discovery of prototypes from high-density clusters and outliers/critic instances from low-density clusters, enhancing the understanding of the dataset and summarising it as with medoids. Our results reinforce the idea that influences can be considered as new inputs for finer analysis on the ML modelling pipeline, to gain a more in-depth and concise understanding of the ML model and the underlying data. This is what we have illustrated in the medical use case: an analysis of clusters of explanations (summarised in the form of decision rules) shows hidden, explainable relationships between attributes for particular predictions (sub-populations).

8. Conclusion and perspectives

In this paper, we propose a novel approach based on the analysis of local explanations for data exploration. We use local attributive XML methods combined with clustering to explore the explantation space and discover new insights about explanations, predictions, modeling, and datasets. By providing clusters of instances based on their explanations, we aim to enhance data analysis. Our experiments demonstrate the effectiveness of influence-based clustering for various XML methods, clustering techniques, and different numbers of clusters. The clusters generated by our influence-based framework are more homogeneous and of higher quality, regardless of the XML methods and clustering techniques used. We show that the explanations-based clusters are of good quality and pertinent, even for low-performance models and misclassified instances. We show the advantages of splitting the well- and misclassified instances by the model when studying a dataset as a whole, as it highlights the most important subgroups of data and the behavior of outliers simultaneously. Finally, we provide a medical use case of how clusters of explanations can be used in real-world applications and support data analysis.

Perspectives. We will initially focus on extending our approach for other supervised tasks and pursuing our work on explanation clustering, particularly by analyzing and characterizing the clusters formed. Clusters can help select informative instances and provide a small number of instances to users. These instances can support understanding datasets and modeling using examples rather than statistical information. Based on the different advantages of each clustering technique (Agnes hierarchy, medoids from Kmedoids, and the variance of GMM clusters) we want to explore how to analyze and make the most of each cluster based on their characteristics, to better understand the explanations, the prediction, the models and the dataset. With users in the loop, the framework, augmented with additional clustering analysis, could be tested to evaluate the impact of the local explanations and their analysis and/or against other local explanation methods like example-based XML methods.

New information on the dataset and its subgroups may also provide feedback on the quality of the training data or the trained model to improve it. This idea of possible user feedback may be one way to improve data quality

and modeling. Clustering based on influences may help to understand *why* the model is wrong and not just *where* the model is wrong. These new insights could also detect biases in the model and the data. Subsequently, the feedback will be evaluated in order to ascertain its potential for implementation within the framework. Finally, our framework could be integrated into a complete system where users can interact with the modeling and define typical instances to profile new data patterns for user testing.

CRediT author statement

Elodie Escriva: Conceptualisation, Methodology, Software, Data Curation, Validation, Investigation, Writing - Original Draft, Visualisation.
Tom Lefrere: Methodology, Software, Investigation, Data Curation, Formal analysis, Visualisation.
Manon Martin: Methodology, Software, Investigation, Data Curation, Formal analysis, Visualisation.
Julien Aligon: Conceptualisation, Methodology, Writing - Review & Editing, Supervision.
Alexandre Chanson: Software, Writing - Review & Editing, Visualisation.
Jean-Baptiste Excoffier: Conceptualisation, Methodology, Writing - Review & Editing, Supervision.
Nicolas Labroche: Conceptualisation, Methodology, Software, Writing - Review & Editing, Supervision.
Chantal Soulé-Dupuy: Writing - Review & Editing, Supervision.
Paul Monsarrat: Formal analysis, Writing - Review & Editing.

Acknowledgements

We thank the French ANRT and Kaduceo company for providing us with PhD grants (no. 2020/0964). We thank Robin Cugny, Emmanuel Doumard, and Haomiao Wang for their help on the medical use case. This study has been partially supported through the grant EUR CARe N°ANR-18-EURE-0003 in the framework of the Programme des Investissements d'Avenir and the national infrastructure "ECELLFrance: Development of mesenchymal stem cell based therapies" (PIA-ANR-11-INBS-005).

References

[1] J. W. Tukey, Exploratory Data Analysis, Addison-Wesley, 1977.

[2] J. Cai, J. Hao, H. Yang, X. Zhao, Y. Yang, A review on semi-supervised clustering, Inf. Sci. 632 (2023) 164–200. doi:10.1016/J.INS.2023.02.088.
URL https://doi.org/10.1016/j.ins.2023.02.088

[3] V. Vu, N. Labroche, Active seed selection for constrained clustering, Intell. Data Anal. 21 (3) (2017) 537–552. doi:10.3233/IDA-150499.
URL https://doi.org/10.3233/IDA-150499

[4] K. Wagstaff, Value, cost, and sharing: Open issues in constrained clustering, in: Knowledge Discovery in Inductive Databases, 5th International Workshop, KDID 2006, Berlin, Germany, September 18, 2006, Revised Selected and Invited Papers, 2006, pp. 1–10. doi:10.1007/978-3-540-75549-4_1.
URL https://doi.org/10.1007/978-3-540-75549-4_1

[5] I. Davidson, K. Wagstaff, S. Basu, Measuring constraint-set utility for partitional clustering algorithms, in: Knowledge Discovery in Databases: PKDD 2006, 10th European Conference on Principles and Practice of Knowledge Discovery in Databases, Berlin, Germany, September 18-22, 2006, Proceedings, 2006, pp. 115–126. doi:10.1007/11871637_15.
URL https://doi.org/10.1007/11871637_15

[6] V. Vu, N. Labroche, B. Bouchon-Meunier, Improving constrained clustering with active query selection, Pattern Recognit. 45 (4) (2012) 1749–1758. doi:10.1016/J.PATCOG.2011.10.016.

[7] D. Klein, S. D. Kamvar, C. D. Manning, From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering, in: C. Sammut, A. G. Hoffmann (Eds.), Machine Learning, Proceedings of the Nineteenth International Conference (ICML 2002), University of New South Wales, Sydney, Australia, July 8-12, 2002, Morgan Kaufmann, 2002, pp. 307–314.

[8] M. Bilenko, S. Basu, R. J. Mooney, Integrating constraints and metric learning in semi-supervised clustering, in: C. E. Brodley (Ed.), Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004, Vol. 69 of ACM International Conference Proceeding Series, ACM, 2004. doi:

10.1145/1015330.1015360.
URL https://doi.org/10.1145/1015330.1015360

[9] B. M. Nogueira, Y. K. B. Tomas, R. M. Marcacini, Integrating distance metric learning and cluster-level constraints in semi-supervised clustering, in: 2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017, IEEE, 2017, pp. 4118–4125. doi:10.1109/IJCNN.2017.7966376.
URL https://doi.org/10.1109/IJCNN.2017.7966376

[10] H. Kaur, H. Nori, S. Jenkins, R. Caruana, H. Wallach, J. Wortman Vaughan, Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning, in: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, CHI '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 1–14. doi:10.1145/3313831.3376219.

[11] M. T. Ribeiro, S. Singh, C. Guestrin, "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, KDD '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 1135–1144. doi:10.1145/2939672.2939778.

[12] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, Vol. 30 of NIPS'17, Curran Associates, Inc., Red Hook, NY, USA, 2017, p. 4768–4777.
URL https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[13] G. Ferrettini, J. Aligon, C. Soulé-Dupuy, Explaining single predictions: A faster method, in: SOFSEM 2020: Theory and Practice of Computer Science, Springer International Publishing, Cham, 2020, pp. 313–324.

[14] G. Ferrettini, E. Escriva, J. Aligon, J.-B. Excoffier, C. Soulé-Dupuy, Coalitional Strategies for Efficient Individual Prediction Explanation, Information Systems Frontiers (2021). doi:10.1007/s10796-021-10141-9.
URL https://doi.org/10.1007/s10796-021-10141-9

[15] A. M. Antoniadi, Y. Du, Y. Guendouz, L. Wei, C. Mazo, B. A. Becker, C. Mooney, Current challenges and future opportunities for xai in machine learning-based clinical decision support systems: A systematic review, Applied Sciences 11 (11) (2021). doi:10.3390/app11115088.

[16] E. Escriva, J. Aligon, J.-B. Excoffier, P. Monsarrat, C. Soulé-Dupuy, How to Make the Most of Local Explanations: Effective Clustering Based on Influences, in: Advances in Databases and Information Systems, Vol. 13985 of Lecture Notes in Computer Science, Springer Nature Switzerland, Barcelone, 2023, pp. 146–160. doi:10.1007/978-3-031-42914-9_11.

[17] A. Holzinger, R. Goebel, R. Fong, T. Moon, K. Müller, W. Samek, xxai - beyond explainable artificial intelligence, in: xxAI - Beyond Explainable AI - International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers, 2020, pp. 3–10. doi:10.1007/978-3-031-04083-2_1.

[18] A. Cooper, O. Doyle, A. Bourke, Supervised Clustering for Subgroup Discovery: An Application to COVID-19 Symptomatology, in: Machine Learning and Principles and Practice of Knowledge Discovery in Databases. ECML PKDD 2021, Communications in Computer and Information Science, Springer International Publishing, Cham, 2021, pp. 408–422. doi:10.1007/978-3-030-93733-1_29.

[19] C. Molnar, Interpretable Machine Learning, 2nd Edition, 2022.
URL https://christophm.github.io/interpretable-ml-book

[20] Z. C. Lipton, The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery., Queue 16 (3) (2018) 31–57. doi:10.1145/3236386.3241340.

[21] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, Artificial Intelligence 267 (2019) 1–38. doi:10.1016/j.artint.2018.07.007.

[22] C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, Nat. Mach. Intell. 1 (5) (2019) 206–215. doi:10.1038/S42256-019-0048-X.

[23] E. Petersen, Y. Potdevin, E. Mohammadi, S. Zidowitz, S. Breyer, D. Nowotka, S. Henn, L. Pechmann, M. Leucker, P. Rostalski, C. Herzog, Responsible and Regulatory Conform Machine Learning for Medicine: A Survey of Challenges and Solutions, IEEE Access 10 (2022). doi:10.1109/ACCESS.2022.3178382.

[24] P. Linardatos, V. Papastefanopoulos, S. Kotsiantis, Explainable ai: A review of machine learning interpretability methods, Entropy 23 (1) (2021). doi:10.3390/e23010018.

[25] E. Štrumbelj, I. Kononenko, Explaining prediction models and individual predictions with feature contributions, Knowledge and Information Systems 41 (3) (2014) 647–665. doi:10.1007/s10115-013-0679-x.

[26] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, S.-I. Lee, From local explanations to global understanding with explainable ai for trees, Nature machine intelligence 2 (1) (2020) 56–67. doi:10.1038/s42256-019-0138-9.

[27] A. K. Jain, M. N. Murty, P. J. Flynn, Data clustering: A review, ACM Comput. Surv. 31 (3) (1999) 264–323. doi:10.1145/331499.331504.

[28] D. L. Davies, D. W. Bouldin, A cluster separation measure, IEEE Trans. Pattern Anal. Mach. Intell. 1 (2) (1979) 224–227. doi:10.1109/TPAMI.1979.4766909.

[29] P. J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20 (1987) 53–65. doi:https://doi.org/10.1016/0377-0427(87)90125-7.

[30] L. Hubert, P. Arabie, Comparing partitions, Journal of classification 2 (1985) 193–218.

[31] A. K. Jain, R. C. Dubes, Algorithms for Clustering Data, Prentice-Hall, 1988.

[32] A. K. Jain, Data clustering: 50 years beyond k-means, Pattern Recognit. Lett. 31 (8) (2010) 651–666. doi:10.1016/J.PATREC.2009.09.011.

[33] J. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proceedings of Berkeley Symposium on Mathematical Statistics & Probability, Vol. 5.1, University of California Press, 1965, pp. 281–297.

[34] L. Kaufman, P. J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, 1st Edition, Wiley Series in Probability and Statistics, Wiley, 1990. doi:10.1002/9780470316801.

[35] J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Kluwer Academic Publishers, USA, 1981.

[36] N. Labroche, Online fuzzy medoid based clustering algorithms, Neurocomputing 126 (2014) 141–150. doi:10.1016/J.NEUCOM.2012.07.057.

[37] L. Parsons, E. Haque, H. Liu, Subspace clustering for high dimensional data: a review, SIGKDD Explor. 6 (1) (2004) 90–105. doi:10.1145/1007730.1007731.

[38] J. Xie, R. B. Girshick, A. Farhadi, Unsupervised deep embedding for clustering analysis, in: Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016, Vol. 48 of JMLR Workshop and Conference Proceedings, JMLR.org, 2016, pp. 478–487.
URL http://proceedings.mlr.press/v48/xieb16.html

[39] L. Kaufman, P. Rousseeuw, Clustering by means of medoids, Data Analysis based on the L1-Norm and Related Methods (1987) 405–416.

[40] J. H. Ward Jr, M. E. Hook, Application of an hierarchical grouping procedure to a problem of grouping profiles, Educational and Psychological Measurement 23 (1) (1963) 69–81.

[41] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the Royal Statistical Society: Series B (Methodological) 39 (1) (1977) 1–22. doi:https://doi.org/10.1111/j.2517-6161.1977.tb01600.x.

[42] M. Ester, H. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Proceedings of the Second International Conference on Knowledge Discovery,and

Data Mining (KDD-96), Portland, Oregon, USA, AAAI Press, 1996, pp. 226–231.
URL http://www.aaai.org/Library/KDD/1996/kdd96-037.php

[43] L. McInnes, J. Healy, Accelerated hierarchical density based clustering, in: Data Mining Workshops (ICDMW), 2017 IEEE International Conference, IEEE, 2017, pp. 33–42. doi:10.1109/ICDMW.2017.12.

[44] K. Lee, M. V. Ayyasamy, Y. Ji, P. V. Balachandran, A comparison of explainable artificial intelligence methods in the phase classification of multi-principal element alloys, Scientific Reports 12 (1) (2022) 11591. doi:10.1038/s41598-022-15618-4.

[45] J.-B. Excoffier, E. Escriva, J. Aligon, M. Ortala, Local Explanation-Based Method for Healthcare Risk Stratification, in: Medical Informatics Europe 2022. Challenges of Trustable AI and Added-Value on Health, IOS Press, 2022, pp. 555–556. doi:10.3233/SHTI220520.

[46] J.-B. Excoffier, N. Salaün-Penquer, M. Ortala, M. Raphaël-Rousseau, C. Chouaid, C. Jung, Analysis of COVID-19 inpatients in France during first lockdown of 2020 using explainability methods, Medical & Biological Engineering & Computing 60 (6) (2022) 1647–1658. doi:10.1007/s11517-022-02540-0.

[47] L. McInnes, J. Healy, S. Astels, hdbscan: Hierarchical density based clustering, The Journal of Open Source Software 2 (11) (2017) 205. doi:10.21105/joss.00205.

[48] J. Vanschoren, J. N. van Rijn, B. Bischl, L. Torgo, Openml: Networked science in machine learning, ACM SIGKDD Exploration Newsletter 15 (2) (2014) 49–60. doi:10.1145/2641190.2641198.

[49] E. Doumard, J. Aligon, E. Escriva, J.-B. Excoffier, P. Monsarrat, C. Soulé-Dupuy, A quantitative approach for the comparison of additive local explanation methods, Information Systems 114 (2023) 102162. doi:https://doi.org/10.1016/j.is.2022.102162.

[50] J. G. Conrad, K. Al-Kofahi, Y. Zhao, G. Karypis, Effective document clustering for large heterogeneous law firm collections, in: Proceedings of the 10th International Conference on Artificial Intelligence and Law,

ICAIL '05, Association for Computing Machinery, New York, NY, USA, 2005, p. 177–187. doi:10.1145/1165485.1165513.

[51] M. J. Zaki, W. M. Jr, Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press, USA, 2014.

[52] D. Alvarez-Melis, T. S. Jaakkola, On the Robustness of Interpretability Methods, in: Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning, arXiv, Stockholm, Sweden, 2018. doi:10.48550/arXiv.1806.08049.

[53] J. Demšar, Statistical comparisons of classifiers over multiple data sets, The Journal of Machine learning research 7 (2006) 1–30.

[54] S. Salvador, P. Chan, Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms, in: 16th IEEE International Conference on Tools with Artificial Intelligence, IEEE Comput. Soc, 2004, pp. 576–584. doi:10.1109/ICTAI.2004.50.

[55] F. Gardin, R. Gautiern, N. Goix, B. Ndiaye, J.-M. Schertzer, Skope-rules (2019).
URL https://github.com/scikit-learn-contrib/skope-rules

[56] B. Giovanola, S. Tiribelli, Beyond bias and discrimination: redefining the ai ethics principle of fairness in healthcare machine-learning algorithms, AI & society 38 (2) (2023) 549–563. doi:10.1007/s00146-022-01455-6.