Unlocking Insights and Trust: The Value of Explainable Clustering Algorithms for Cognitive Agents*

Federico Sabbatini[1],*, Roberta Calegari[2]

[1] *Department of Pure and Applied Sciences, University of Urbino Carlo Bo*
[2] *Department of Computer Science and Engineering (DISI), Alma Mater Studiorum–University of Bologna*

Abstract

In the realm of cognitive agents, including both human users and AI systems, explainable clustering algorithms have gained prominence. These algorithms offer enhanced transparency, making clustering results comprehensible to users and aiding AI systems in decision-making. They also facilitate knowledge discovery by revealing cluster characteristics, reducing cognitive load for users, and playing a vital role in ethical and bias mitigation. This paper introduces an innovative extension of the existing PSyKE framework, designed to support explainable clustering techniques and, thus, to augment cognitive agent capabilities. State-of-the-art review, experiment findings, and a synthesis of key insights are also provided.

Keywords

Explainable clustering, Explainable artificial intelligence, Symbolic knowledge extraction, PSyKE

1. Introduction

In the realm of cognitive agents, which encompass both human users and artificial intelligence (AI) systems, the advent of explainable clustering algorithms has gained significant attention [1]. These algorithms offer several advantages that amplify the efficacy and transparency of clustering processes across diverse domains [2]. This paper explores the benefits of explainable clustering algorithms to augment the capabilities of cognitive agents.

At the forefront of these advantages there is the enhanced transparency provided by explainable clustering algorithms [3]. They have the capability to yield clustering results in a comprehensible and interpretable manner, ensuring that both human users and AI systems can discern the rationale behind the grouping of data points. This transparency, we argue, is an indispensable element for fostering trust in AI systems and empowering human users to engage in the validation and comprehension of the clustering process. Moreover, explainable clustering algorithms offer improved decision support, which is invaluable in the realm of cognitive agents [4]. These agents often rely on clustering outcomes to make informed decisions or to

WOA 2023: 24th Workshop From Objects to Agents, November 6–8, Rome, Italy

*Original research paper.

*Corresponding author.

f.sabbatini1@campus.uniurb.it (F. Sabbatini); roberta.calegari@unibo.it (R. Calegari)
0000-0002-0532-6777 (F. Sabbatini); 0000-0003-3794-2942 (R. Calegari)

CEUR Workshop Proceedings (CEUR-WS.org)

provide recommendations. The inherent transparency in explainable clustering assists these agents in deciphering the intricate structures within data, thereby facilitating more informed and robust decision-making processes. Beyond their utility in decision-making, explainable clustering algorithms serve as an instrument for effective knowledge discovery [5]. Clustering serves as a foundational step in knowledge discovery, and explainable clustering algorithms take this a step further. Not only do they create clusters, but they also unravel the defining characteristics that distinguish each cluster. This capacity empowers cognitive agents to gain insights into complex data sets, enriching the pool of knowledge they can leverage. Further, explainable clustering algorithms also contribute to reduce cognitive load on human users, particularly in the face of complex tasks, such as clustering high-dimensional data sets [6]. By presenting clustering results in a more digestible and comprehensible manner, these algorithms alleviate the cognitive burden placed on human users, promoting efficiency and accuracy. Perhaps most significantly, explainable clustering plays a pivotal role in ethical and bias mitigation [3]. It empowers cognitive agents to identify and rectify potential biases or ethical concerns within the data or the clustering process itself. By enabling the explainability of clustering decisions, explainable clustering algorithms support the pursuit of fairness and equity in data-driven processes.

In light of these considerations, this paper introduces a groundbreaking extension of the PSyKE framework tailored to enhance the capabilities of cognitive agents via explainable clustering support. The paper is organised as follows. A state-of-the-art review is first provided (Section 2), followed by our proposal, the explainable clustering support integrated within the PSyKE Framework (Section 3). We then delve into the findings of experiments conducted (Section 4) and conclude with a synthesis of key insights derived from our exploration.

2. Related Works

2.1. Explainable Clustering

Several explainable clustering techniques have been developed in the last decades and it is possible to find in the literature their practical application in critical areas, also to tackle complex tasks involving image data sets and medical time series [7, 8, 9].

A subset of the proposed algorithms are based on tree-based clustering according to different strategies that may be classified as top-down or bottom-up [10, 11, 12, 13, 14, 15, 16]. Explainable clustering methods adhering to the top-down approach usually start by building the tree root node, associated with the whole training data set. Successively, the root node's data are partitioned into disjoint subsets associated with the child nodes and this is recursively repeated to grow the clustering tree. The tree expansion ends according to a stopping criterion that may consider the predictive performance of the clustering at a given depth and/or the availability of a fixed, minimum amount of training samples in deep nodes. Each internal node may correspond to a constraint on an individual input feature or a set of constraints involving all of them. A common characteristic of top-down strategies is the input feature space partitioning via cutting hyperplanes that are perpendicular to the data dimensions.

Different approaches are the explanation of clusters via rectangular input space partitioning [17], or the description of clusters in terms of centroids and distances [18]. The former may achieve a good human-interpretability extent, since it describes clusters based upon only 2 interval inclusion constraints. However, the algorithm may combine multiple input attributes and thus consider preconditions on new, composite features. This behaviour may constitute a hindering factor for human interpretability.

2.1.1. CLASSIX

The CLASSIX (contrived acronym defined by the authors as "CLustering by Aggregation with Sorting-based Indexing" and the letter "X" for "eXplainability") algorithm [18] has been recently proposed as a novel 2-phase explainable clustering procedure. It is presented as a technique denoted by small computational time requirements.

The first phase of CLASSIX is a greedy aggregation aimed at creating groups of training instances having "small" distances from each other. The distance may be tuned by users through a dedicated input parameter. It is worth noting that a preceding sorting step is required to perform the aggregation.

The second phase consists of merging the groups into definitive clusters. Two merging strategies are supported by CLASSIX, namely, density- or distance-based (see [18] for further details).

CLASSIX requires two used-defined parameters defining *(i)* a lower-bound for the accepted cluster size, intended as the number of samples, and *(ii)* an upper-bound for the distance between training instances assigned to the same group (with reference to the aggregation phase).

The CLASSIX technique may provide explanations locally or globally. Global explanations are built based on the coordinates of the initial points for each individual group created at the end of the first phase of the procedure. On the other hand, two kinds of local explanations are supported. It is possible to obtain the reason behind the cluster assignment corresponding to an individual instance or CLASSIX may be queried to explain why two instances are assigned to the same cluster or not. Local explanations are provided by listing the operations performed during CLASSIX's merging phase.

2.1.2. IMM

The IMM (Iterative Mistake Minimization) clustering procedure [13] is presented by the authors as an accurate, efficient, and interpretable method based on the induction of decision trees. The output decision trees are binary and their internal nodes are associated with training data partitions. Splits corresponding to internal nodes always involve individual input attributes.

The IMM algorithm requires growing a tree having k leaves to identify as many clusters. The tree induction considers a set of desiderata, e.g., keeping the tree size as small as possible and minimising the cluster's fragmentation while deepening the tree. Fragmentation is intended as spreading instances belonging to a single cluster over multiple subtrees.

Explanations for individual cluster assignments are provided by describing the complete paths starting from the tree root through the leaves associated with those assignments. It is also possible to obtain global explanations for the clustering by listing all the existing paths to the different leaves.

As for the tree growth complexity, it is worth noting that if IMM identifies k clusters the corresponding tree has a depth equal to $k - 1$ in the worst case (unbalanced tree). As a result, any clustering assignment is described in terms of the conjunction of at most $k - 1$ constraints on individual input features.

2.1.3. ExACT and CREAM

ExACT [15] and CREAM [14] are tree-based explainable clustering techniques achieving human-interpretability via hypercubic approximation of the identified clusters. Both algorithms induce a binary tree to partition the input feature space and each internal node of the tree corresponds to a hypercube-inclusion constraint.

Trees are built recursively, according to a top-down strategy, and each node of the tree corresponds to an input feature space subregion. The tree root is associated with the surrounding cube, i.e., the minimal cube enclosing all the training instances. During a single recursive iteration an internal node is marked with a hypercube-inclusion constraint and therefore its two child nodes represent the hypercubic partition of the input feature space denoted by the constraint on one side and the complementary subregion on the other side.

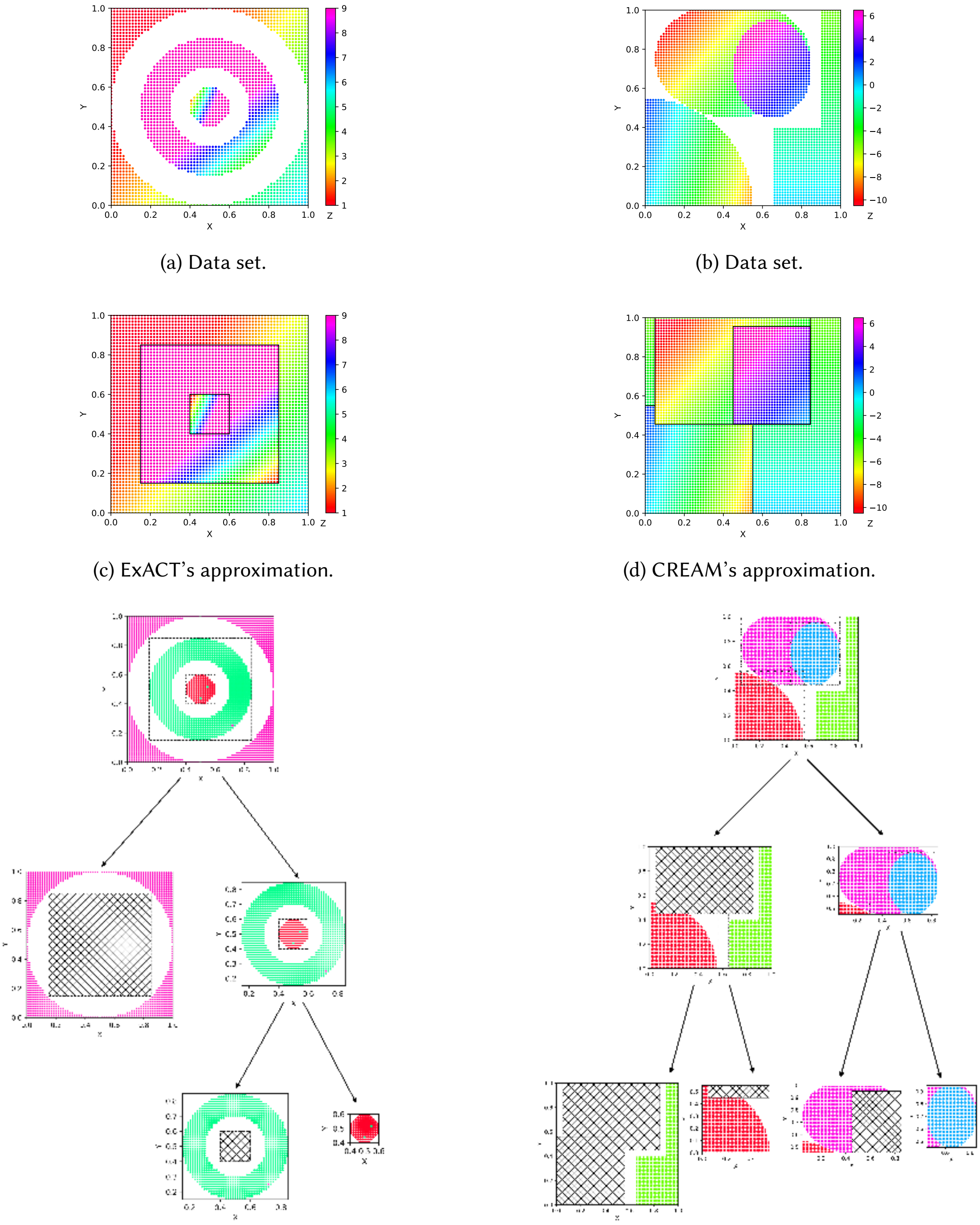
Both ExACT and CREAM exploit underlying instances of Gaussian mixture models [19] to identify relevant clusters of data and instances of DBSCAN [20] to remove outliers from the identified clusters. Clusters without outliers are then approximated via hypercubes. The selection of the best splits to be associated with internal nodes follows different approaches. ExACT tries to perform a greedy minimisation of the cluster fragmentation, whereas CREAM opts for a greedy maximisation of the estimated predictive performance corresponding to the available splits. The two approaches are thus based on the selection of best local alternatives, without any guarantees of absolute optimality. The differences between ExACT and CREAM are depicted in Figure 1 for artificial data sets having concentric [15] or overlapping [14] clusters.

Three user-defined parameters are required by ExACT and CREAM, namely: *(i)* a maximum tree depth; *(ii)* a predictive error threshold; and *(iii)* an upper-bound for the number of clusters identifiable via Gaussian mixture models. Depth and error threshold may be automatically tuned with the OrCHiD procedure [14].

It is worth noting that besides mere clustering tasks ExACT and CREAM may also be applied to perform explainable classification and regression, given that they are able to associate to each cluster one amongst the following outputs: cluster ids, class labels, constant values and linear combinations of the input features [21, 22].

2.2. The PSyKE Framework

PSyKE is a general-purpose Python software library mainly dedicated to symbolic knowledge extraction [23, 24], but also providing a suite of tools for data pre-processing, manipulation and visualisation as well as for machine learning tasks. It offers a unified interface for several extraction techniques belonging to the pedagogical paradigm and it supports interoperability with other widely adopted Python packages, as `numpy`, `pandas` and `sklearn` [25]. Interoperability with Semantic Web tools is also provided [26]. Knowledge-extraction techniques supported by PSyKE can be applied to any kind of supervised machine learning model without limitations



(a) Data set.

(b) Data set.

(c) ExACT's approximation.

(d) CREAM's approximation.

(e) Binary tree induced by ExACT.

(f) Binary tree induced by CREAM.

Figure 1: Comparison of the partitioning performed by ExACT and CREAM on artificial data sets having concentric clusters [15] or superposing clusters [14].

about the nature of the task at hand, i.e., classification as well as regression.

At the time of writing PSyKE includes implementations of the following knowledge-extraction procedures: Rule-extraction-as-learning (REAL) [27], Trepan [28], Cart [29], Iter [30], GridEx [31], GridREx [32] and CReEPy [33]. These techniques provide global explanations for the predictions obtained via opaque machine learning models in the form of a human-interpretable Prolog theory. Therefore, PSyKE may be exploited as a tool to achieve trustworthy artificial intelligence [34].

3. Explainable Clustering Support in the PSyKE Framework

In order to support explainable clustering within the PSyKE framework, the structure of the main project package (the `psyke` package) has been totally redesigned. The new structure of the software library, depicted in Figure 2, is effective from version 0.5[1]. Only the `psyke` package is shown in the figure, since it is the main subject of the presented framework extension.

3.1. The `EvaluableModel` Interface

The current design of PSyKE's main package is based on the notion of `EvaluableModel`, an interface representing any predictive model that may be evaluated via some scoring metric (e.g., a machine learning predictor, an interpretable model obtained via knowledge extraction, a clustering technique). Evaluable models in PSyKE come along with information about the pre-processing routines applied to the data sets, i.e., the parameters applied to perform normalisation and/or discretisation. Any evaluable model should be able to provide predictions and its predictive performance should be assessable through an adequate scoring function.

Accordingly, the interface exposes two methods. The `predict` method is abstract and accepts a dataframe (i.e., a `pandas` dataframe, but also numpy arrays are accepted) and returns the corresponding predictions. The definition of this method depends on the specific model. Therefore, it must be defined within other classes implementing the `EvaluableModel` interface. The `score` method accepts a dataframe and a scoring function and then returns the scoring function evaluated on the instances of that dataframe. The interface provides scoring functions for classification (e.g., classification accuracy, F_1 score and confusion matrices), regression (e.g., mean absolute/squared error and R^2 score), and clustering (e.g., adjusted Rand index, adjusted mutual information, V-measure and Fowlkes-Mallows score).

The `EvaluableModel` interface is extended by three other interfaces, namely:

`HyperCubePredictor` describing any evaluable model whose predictions are based on a hypercubic partitioning of the input feature space. The set of hypercubes is an attribute defined by the interface. It also defines the inherited `predict` method;

`Extractor` representing any evaluable model providing interpretable predictions by highlighting symbolic input/output relationships extracted from an opaque predictor. Relationships are learned via the `extract` method, requiring as input parameters a training dataframe and an opaque predictor (e.g., a machine learning model from the `sklearn` library or any

[1] Code available at https://github.com/psykei/psyke-python

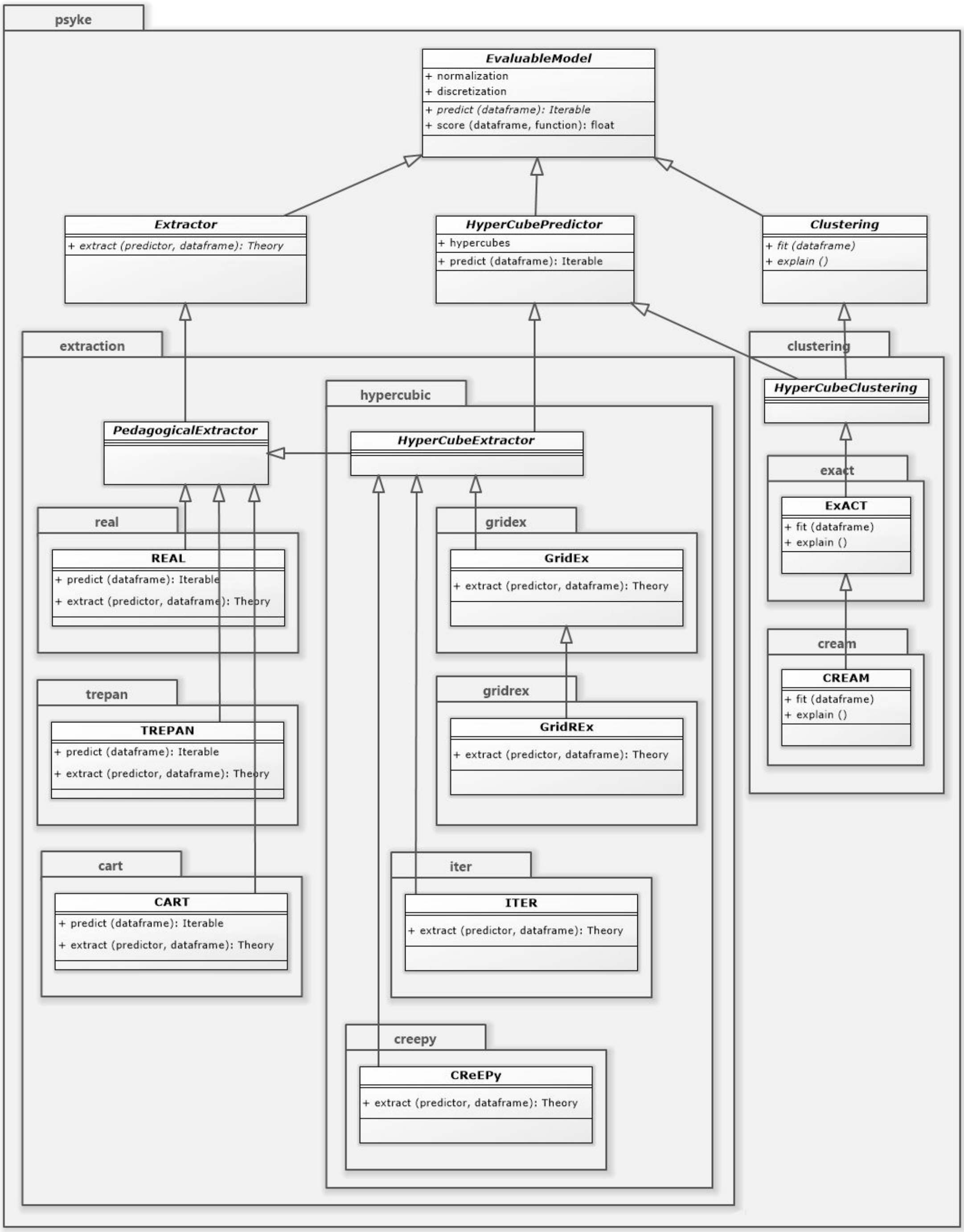

psyke
EvaluableModel
+ normalization
+ discretization
+ predict (dataframe): Iterable
+ score (dataframe, function): float
Extractor
+ extract (predictor, dataframe): Theory
HyperCubePredictor
+ hypercubes
+ predict (dataframe): Iterable
Clustering
+ fit (dataframe)
+ explain ()
extraction
hypercubic
clustering
PedagogicalExtractor
HyperCubeExtractor
HyperCubeClustering
real
REAL
+ predict (dataframe): Iterable
+ extract (predictor, dataframe): Theory
gridex
GridEx
+ extract (predictor, dataframe): Theory
exact
ExACT
+ fit (dataframe)
+ explain ()
gridrex
GridREx
+ extract (predictor, dataframe): Theory
cream
CREAM
+ fit (dataframe)
+ explain ()
trepan
TREPAN
+ predict (dataframe): Iterable
+ extract (predictor, dataframe): Theory
cart
CART
+ predict (dataframe): Iterable
+ extract (predictor, dataframe): Theory
iter
ITER
+ extract (predictor, dataframe): Theory
creepy
CReEPy
+ extract (predictor, dataframe): Theory

Figure 2: UML class diagram for the psyke package of PSyKE version 0.5.

other object having a `predict` method). The `extract` method is abstract since it differs based on the individual extraction techniques and thus it has to be defined by classes extending the `Extractor` interface;

`Clustering` resuming the properties of any explainable clustering technique that may be fitted on a dataframe and explained via human-interpretable descriptions of the identified clusters. Accordingly, it exposes two abstract methods for these purposes, to be defined by inheriting classes.

3.2. The `extraction` Package

The `psyke` package of the PSyKE library encloses the `extraction` sub-package, dedicated to symbolic knowledge extraction from opaque machine learning models. The sub-package contains an interface representing a generic pedagogical knowledge-extraction algorithm (the `PedagogicalExtractor` interface, extending the `Extractor` interface). Three pedagogical knowledge-extraction algorithms (namely, REAL, Trepan and Cart) implement the `PedagogicalExtractor` interface. Each algorithm is enclosed in a dedicated sub-package and the corresponding main class defines the abstract methods `predict` and `extract` inherited from `EvaluableModel` and `Extractor`, respectively.

The `extraction` sub-package contains an inner sub-package named `hypercubic`, dedicated to hypercube-based knowledge extractors. It defines the `HyperCubeExtractor` interface, representing a generic extractor of this kind and extending both the `HyperCubePredictor` and the `PedagogicalExtractor` interfaces. `HyperCubeExtractor` is realised by four different classes implementing as many knowledge extractors (i.e., GridEx, GridREx, Iter and CReEPy), each one encapsulated in an individual package. Only the `extract` method is defined by these classes, given that the `predict` method is common and already defined and inherited from `HyperCubePredictor`.

The features of knowledge-extraction algorithms implemented in PSyKE are listed in Table 1, with particular focus on the translucency of the extractors, the supported machine learning task, the kind of accepted input features and provided outputs, the shape of the extracted knowledge and the interpretability extent achieved by the algorithms.

3.3. The `clustering` Package

The explainable clustering techniques offered by PSyKE (ExACT and CREAM) are contained in the `clustering` package and realise the `HyperCubeClustering` interface, given that they are both clustering procedures based on hypercubic partitioning of the input feature space. The `HyperCubeClustering` interface, in turn, extends the aforementioned `HyperCubePredictor` and `Clustering` interfaces. As a result, only the `fit` and `explain` methods need to be defined within the classes implementing explainable clustering techniques. Since CREAM is an extension of the ExACT algorithm, the corresponding classes follow an adequate hierarchy. Nonetheless, each algorithm is encapsulated in an individual package.

The features of the explainable clustering techniques supported by PSyKE are resumed in Table 1.

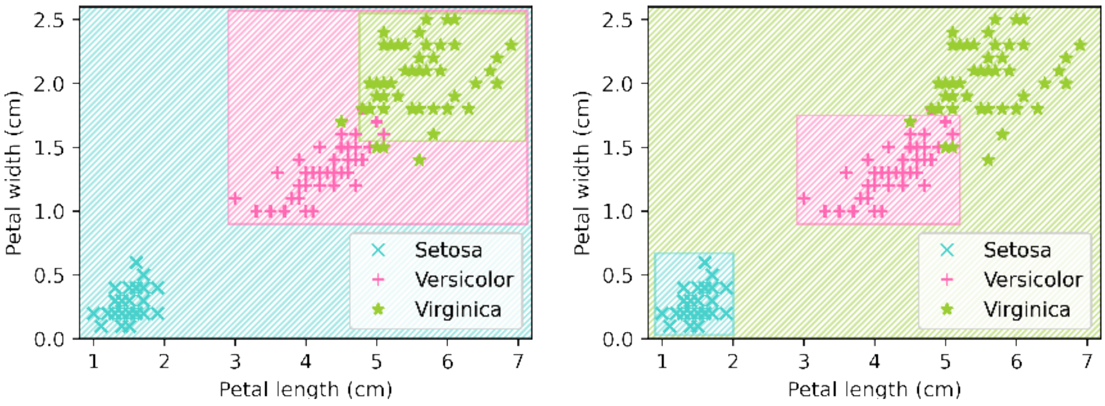
Table 1
Summary of the knowledge-extraction and explainable clustering algorithms supported by PSyKE version 0.5. Translucency is not a property of explainable clustering techniques; it is thus reported only for knowledge extractors.

	Knowledge extraction							Clustering	
	REAL	Trepan	Cart	Iter	GridEx	GridREx	CReEPy	ExACT	CREAM
Ref. paper	[27]	[28]	[29]	[30]	[31]	[32]	[33]	[15]	[14]
Trans.:									
Pedagogical	×	×	×	×	×	×	×		
Decomp.									
Task:									
Classification	×	×	×	×	×	×	×	×	×
Regression			×	×	×	×	×	×	×
Clustering								×	×
Input feat.:									
Binary	×	×	×	$\times^*$	$\times^*$	$\times^*$	$\times^*$	$\times^*$	$\times^*$
Discrete	$\times^\ddagger$	$\times^\ddagger$	×	$\times^\dagger$	$\times^\dagger$	$\times^\dagger$	$\times^\dagger$	$\times^\dagger$	$\times^\dagger$
Continuous			×	×	×	×	×	×	×
Output:									
String label	×	×	×	×	×	×	×	×	×
Constant			×	×	×	×	×	×	×
Linear eq.				×	×	×	×	×	×
Cluster id								×	×
Knowledge:									
Rule list	×			×	×	×			
Decision tree		$\times^\S$	$\times^\S$				$\times^\S$	$\times^\S$	$\times^\S$
Interpret.:									
Global	×	×	×	×	×	×	×	×	×
Local	$\times^\diamond$	$\times^\diamond$	$\times^\diamond$	$\times^\diamond$	$\times^\diamond$	$\times^\diamond$	$\times^\diamond$	$\times^\diamond$	$\times^\diamond$

* Only if binary values are encoded as numbers, e.g., 0 and 1
‡ Only if discrete values are binarised, e.g., via one-hot encoding
† Only if discrete values are numeric
§ Decision trees may be linearised into an ordered list of rules
$^\diamond$ Local interpretability may be achieved by considering individual items of the global explanation

4. Illustrative Experiment

To demonstrate the effectiveness and the human-interpretable degree of the explanations provided by PSyKE's clustering techniques we carried out a set of experiments on the well-known Iris data set [35]. The 150 data instances have been split into training and test sets (75% + 25%). The training set was then used to fit instances of ExACT and CREAM. Best values for the depth and error threshold hyper-parameters of the explainable clustering techniques have been estimated with OrCHiD. We used a value of 3 for the remaining parameter expressing the



(a) ExACT.

(b) CREAM.

Figure 3: Comparison of the partitioning performed by ExACT and CREAM on the iris data set.

Listing 1 Classification rules obtained from the ExACT clsutering on the Iris data set.

```
Class Virginica if PetalWidth in [1.6, 2.5] and PetalLength in [4.8, 6.9] and
                   SepalWidth in [2.5, 3.8] and SepalLength in [5.7, 7.9].
Class Versicolor if PetalWidth in [0.9, 2.5] and PetalLength in [3.0, 6.9] and
                    SepalWidth in [2.2, 3.8] and SepalLength in [4.9, 7.9].
Class Setosa otherwise.
```

Listing 2 Classification rules obtained from the CREAM clustering on the Iris data set.

```
Class Versicolor if PetalWidth in [0.9, 1.7] and PetalLength in [3.0, 5.2] and
                    SepalWidth in [2.0, 3.4] and SepalLength in [5.0, 7.0].
Class Setosa if PetalWidth in [0.0, 0.7] and PetalLength in [0.9, 2.0] and
                SepalWidth in [2.3, 4.4] and SepalLength in [4.3, 5.8].
Class Virginica otherwise.
```

maximum amount of identifiable clusters.

The decision boundaries for the three Iris classes are highlighted in Figure 3. The corresponding explanations are listed in Listings 1 and 2 for ExACT and CREAM, respectively. It is possible to notice that the two clustering algorithms provide very different decision boundaries and explanations, but they have comparable quality in terms of human-readability (1 rule per distinct class) and predictive performance (classification accuracy of about 93% on the test set).

Readability of explanations shown in Listings 1 and 2 could be improved by removing the least relevant input features and keeping only the most relevant ones, i.e., the petal length and width reported in Figure 3.

5. Conclusions

The paper delves into the pivotal role of explainable clustering algorithms within the domain of cognitive agents, encompassing both human users and AI systems. The advantages offered by these algorithms in the realm of cognitive agents are multifaceted, ranging from enhancing interpretability and fostering trust to facilitating more effective decision-making processes.

Motivated by the need to foster transparency and accountability in the operations of cognitive agents, we introduce an extension of the PSyKE framework's design to augment its capabilities through the incorporation of explainable clustering support. The discussion of this novel design, coupled with real world experiments, highlights the potential to significantly elevate the performance and ethical standing of cognitive agents.

The outcomes of this research pave the way for ongoing advancements in the field, emphasising the importance of continued development and integration of explainable clustering algorithms. By doing so, cognitive agents can be expected to evolve into more transparent, effective, and ethically responsible entities. As we move forward, future efforts will be directed towards the practical implementation of explainable clustering algorithms within cognitive agents, involving rigorous testing in simulated real-world scenarios.

Acknowledgments

This work has been supported partially by the European Union's Horizon Europe AEQUITAS research and innovation programme under grant number 101070363 and partially by the European Union ICT-48 2020 project TAILOR (No. 952215).

References

[1] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai, Information fusion 58 (2020) 82–115.

[2] A. Rahman, M. S. Hossain, G. Muhammad, D. Kundu, T. Debnath, M. Rahman, M. S. I. Khan, P. Tiwari, S. S. Band, Federated learning-based ai approaches in smart healthcare: concepts, taxonomies, challenges and open issues, Cluster computing 26 (2023) 2271–2311.

[3] S. Ali, T. Abuhmed, S. El-Sappagh, K. Muhammad, J. M. Alonso-Moral, R. Confalonieri, R. Guidotti, J. Del Ser, N. Díaz-Rodríguez, F. Herrera, Explainable artificial intelligence (xai): What we know and what is left to attain trustworthy artificial intelligence, Information Fusion 99 (2023) 101805.

[4] S. Dasgupta, N. Frost, M. Moshkovitz, C. Rashtchian, Explainable k-means and k-medians clustering, in: Proceedings of the 37th International Conference on Machine Learning, Vienna, Austria, 2020, pp. 12–18.

[5] S. Bobek, M. Kuk, J. Brzegowski, E. Brzychczy, G. J. Nalepa, Knac: an approach for enhancing cluster analysis with background knowledge and explanations, Applied Intelligence 53 (2023) 15537–15560.

[6] A. Endert, W. Ribarsky, C. Turkay, B. W. Wong, I. Nabney, I. D. Blanco, F. Rossi, The state of the art in integrating machine learning into visual analytics, in: Computer Graphics Forum, volume 36, Wiley Online Library, 2017, pp. 458–486.

[7] E. Horel, K. Giesecke, V. Storchan, N. Chittar, Explainable clustering and application to wealth management compliance, in: T. Balch (Ed.), ICAIF '20: The First ACM International Conference on AI in Finance, New York, NY, USA, October 15–16, 2020, ACM, 2020, pp. 47:1–47:6. URL: https://doi.org/10.1145/3383455.3422530. doi:10.1145/3383455.3422530.

[8] L. Manduchi, M. Hüser, M. Faltys, J. E. Vogt, G. Rätsch, V. Fortuin, T-DPSOM: an interpretable clustering method for unsupervised learning of patient health states, in: M. Ghassemi, T. Naumann, E. Pierson (Eds.), ACM CHIL '21: ACM Conference on Health, Inference, and Learning, Virtual Event, USA, April 8-9, 2021, ACM, 2021, pp. 236–245. URL: https://doi.org/10.1145/3450439.3451872. doi:10.1145/3450439.3451872.

[9] S. Deshmukh, B. K. Behera, P. Mulay, E. A. Ahmed, S. Al-Kuwari, P. Tiwari, A. Farouk, Explainable quantum clustering method to model medical data, Knowl. Based Syst. 267 (2023) 110413. URL: https://doi.org/10.1016/j.knosys.2023.110413. doi:10.1016/j.knosys.2023.110413.

[10] J. Basak, R. Krishnapuram, Interpretable hierarchical clustering by constructing an unsupervised decision tree, IEEE Trans. Knowl. Data Eng. 17 (2005) 121–132. URL: https://doi.org/10.1109/TKDE.2005.11. doi:10.1109/TKDE.2005.11.

[11] R. Fraiman, B. Ghattas, M. Svarc, Interpretable clustering using unsupervised binary trees, Adv. Data Anal. Classif. 7 (2013) 125–145. URL: https://doi.org/10.1007/s11634-013-0129-3. doi:10.1007/s11634-013-0129-3.

[12] D. Bertsimas, A. Orfanoudaki, H. M. Wiberg, Interpretable clustering via optimal trees, CoRR abs/1812.00539 (2018). URL: http://arxiv.org/abs/1812.00539. arXiv:1812.00539.

[13] S. Dasgupta, N. Frost, M. Moshkovitz, C. Rashtchian, Explainable k-means and k-medians clustering, CoRR abs/2002.12538 (2020). URL: https://arxiv.org/abs/2002.12538. arXiv:2002.12538.

[14] F. Sabbatini, R. Calegari, Explainable Clustering with CREAM, in: Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning, 2023, pp. 593–603. URL: https://doi.org/10.24963/kr.2023/58. doi:10.24963/kr.2023/58.

[15] F. Sabbatini, R. Calegari, ExACT explainable clustering: Unravelling the intricacies of cluster formation, in: Proceedings of the 2nd International Workshop on Knowledge Diversity, KoDis 2023, Rhodes, Greece, September 2–8, 2023 (to appear), 2023.

[16] F. Sabbatini, R. Calegari, Bottom-up and top-down workflows for hypercube- and clustering-based knowledge extractors, in: D. Calvaresi, A. Najjar, A. Omicini, R. Aydogan, R. Carli, G. Ciatto, K. Främling (Eds.), Explainable and Transparent AI and Multi-Agent Systems. Fifth International Workshop, EXTRAAMAS 2023, London, UK, May 29, 2023, Revised Selected Papers, volume 14127 of *LNCS*, Springer Cham, Basel, Switzerland, 2023, pp. 116–129. doi:10.1007/978-3-031-40878-6_7.

[17] J. Chen, Y. Chang, B. Hobbs, P. J. Castaldi, M. H. Cho, E. K. Silverman, J. G. Dy, Interpretable clustering via discriminative rectangle mixture model, in: F. Bonchi, J. Domingo-Ferrer, R. Baeza-Yates, Z. Zhou, X. Wu (Eds.), IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain, IEEE Computer Society, 2016, pp. 823–828. URL: https://doi.org/10.1109/ICDM.2016.0097. doi:10.1109/ICDM.2016.0097.

[18] X. Chen, S. Güttel, Fast and explainable clustering based on sorting, CoRR abs/2202.01456 (2022). URL: https://arxiv.org/abs/2202.01456. arXiv:2202.01456.

[19] K. P. Murphy, Machine learning – A probabilistic perspective, Adaptive computation and

machine learning series, MIT Press, 2012.

[20] R. F. Ling, On the theory and construction of k-clusters, The Computer Journal 15 (1972) 326–332. URL: https://doi.org/10.1093/comjnl/15.4.326. doi:10.1093/comjnl/15.4.326. arXiv:https://academic.oup.com/comjnl/article-pdf/15/4/326/1005965/150326.pdf.

[21] F. Sabbatini, G. Ciatto, R. Calegari, A. Omicini, Hypercube-based methods for symbolic knowledge extraction: Towards a unified model, in: A. Ferrando, V. Mascardi (Eds.), WOA 2022 – 23rd Workshop "From Objects to Agents", volume 3261 of *CEUR Workshop Proceedings*, Sun SITE Central Europe, RWTH Aachen University, 2022, pp. 48–60. URL: http://ceur-ws.org/Vol-3261/paper4.pdf.

[22] F. Sabbatini, G. Ciatto, R. Calegari, A. Omicini, Towards a unified model for symbolic knowledge extraction with hypercube-based methods, Intelligenza Artificiale 17 (2023) 63–75. URL: https://doi.org/10.3233/IA-230001. doi:10.3233/IA-230001.

[23] F. Sabbatini, G. Ciatto, R. Calegari, A. Omicini, On the design of PSyKE: A platform for symbolic knowledge extraction, in: R. Calegari, G. Ciatto, E. Denti, A. Omicini, G. Sartor (Eds.), WOA 2021 – 22nd Workshop "From Objects to Agents", volume 2963 of *CEUR Workshop Proceedings*, Sun SITE Central Europe, RWTH Aachen University, 2021, pp. 29–48. 22nd Workshop "From Objects to Agents" (WOA 2021), Bologna, Italy, 1–3 September 2021. Proceedings.

[24] F. Sabbatini, G. Ciatto, R. Calegari, A. Omicini, Symbolic knowledge extraction from opaque ML predictors in PSyKE: Platform design & experiments, Intelligenza Artificiale 16 (2022) 27–48. URL: https://doi.org/10.3233/IA-210120. doi:10.3233/IA-210120.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. VanderPlas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research (JMLR) 12 (2011) 2825–2830. URL: https://dl.acm.org/doi/10.5555/1953048.2078195.

[26] F. Sabbatini, G. Ciatto, A. Omicini, Semantic Web-based interoperability for intelligent agents with PSyKE, in: D. Calvaresi, A. Najjar, M. Winikoff, K. Främling (Eds.), Explainable and Transparent AI and Multi-Agent Systems, volume 13283 of *Lecture Notes in Computer Science*, Springer, 2022, pp. 124–142. URL: http://link.springer.com/10.1007/978-3-031-15565-9_8. doi:10.1007/978-3-031-15565-9_8.

[27] M. W. Craven, J. W. Shavlik, Using sampling and queries to extract rules from trained neural networks, in: Machine Learning Proceedings 1994, Elsevier, 1994, pp. 37–45. doi:10.1016/B978-1-55860-335-6.50013-1.

[28] M. W. Craven, J. W. Shavlik, Extracting tree-structured representations of trained networks, in: D. S. Touretzky, M. C. Mozer, M. E. Hasselmo (Eds.), Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference, The MIT Press, 1996, pp. 24–30. URL: http://papers.nips.cc/paper/1152-extracting-tree-structured-representations-of-trained-networks.pdf.

[29] L. Breiman, J. Friedman, C. J. Stone, R. A. Olshen, Classification and Regression Trees, CRC Press, 1984.

[30] J. Huysmans, B. Baesens, J. Vanthienen, ITER: An algorithm for predictive regression rule extraction, in: Data Warehousing and Knowledge Discovery (DaWaK 2006), Springer, 2006, pp. 270–279. doi:10.1007/11823728_26.

[31] F. Sabbatini, G. Ciatto, A. Omicini, GridEx: An algorithm for knowledge extraction from black-box regressors, in: D. Calvaresi, A. Najjar, M. Winikoff, K. Främling (Eds.), Explainable and Transparent AI and Multi-Agent Systems. Third International Workshop, EXTRAAMAS 2021, Virtual Event, May 3–7, 2021, Revised Selected Papers, volume 12688 of *LNCS*, Springer Nature, Basel, Switzerland, 2021, pp. 18–38. doi:10.1007/978-3-030-82017-6_2.

[32] F. Sabbatini, R. Calegari, Symbolic knowledge extraction from opaque machine learning predictors: GridREx & PEDRO, in: G. Kern-Isberner, G. Lakemeyer, T. Meyer (Eds.), Proceedings of the 19th International Conference on Principles of Knowledge Representation and Reasoning, KR 2022, Haifa, Israel. July 31 - August 5, 2022, 2022. URL: https://proceedings.kr.org/2022/57/. doi:10.24963/kr.2022/57.

[33] F. Sabbatini, R. Calegari, Unveiling opaque predictors via explainable clustering: The CReEPy algorithm, in: Proceedings of BEWARE-2023, Rome, Italy, November 6–9, 2023, (to appear), 2023.

[34] R. Calegari, F. Sabbatini, The PSyKE technology for trustworthy artificial intelligence 13796 (2023) 3–16. URL: https://doi.org/10.1007/978-3-031-27181-6_1. doi:10.1007/978-3-031-27181-6_1, xXI International Conference of the Italian Association for Artificial Intelligence, AIxIA 2022, Udine, Italy, November 28 – December 2, 2022, Proceedings.

[35] R. A. Fisher, The use of multiple measurements in taxonomic problems, Annals of Eugenics 7 (1936) 179–188. doi:10.1111/j.1469-1809.1936.tb02137.x.