

Polish Academy of Sciences
Systems Research Institute

Ph.D. Dissertation

mgr Kamil Kmita

**Explainable Impact of Partial Supervision in
Semi-Supervised Fuzzy Clustering**

Supervisor
dr hab. Katarzyna Kaczmarek-Majer

Warsaw 2026

Polska Akademia Nauk
Instytut Badań Systemowych

Rozprawa Doktorska

mgr Kamil Kmita

**Wyjaśnialny wpływ częściowego nadzoru
w modelach rozmytej analizy skupień**

Promotor
dr hab. Katarzyna Kaczmarek-Majer

Warszawa 2026

Abstract

This dissertation is devoted to the explainable impact of partial supervision in semi-supervised fuzzy clustering. It is shown that, despite decades of advances in the field, the critical mechanism of regulating this impact with the use of the hyperparameter α , called the scaling factor, has not been explained, in the sense that it has not been fully studied in the context of the fuzzy clustering model as a whole, including the objective function, the optimization problem, the solution algorithm, and the model outcomes: degrees of membership and clusters' prototypes. According to the principles of Explainable Artificial Intelligence, the lack of explainability of such a mechanism results in reduced trust in modeling results, which is particularly problematic in the analysis of sensitive data, such as medical data. To fill the identified research gap, a novel model-agnostic explainability framework for semi-supervised fuzzy clustering is proposed. The framework is further used to explain the impact of α on the outcomes of two classical models: Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means. The proposed theoretical advances led to an extended empirical analysis of the impact of partial supervision, which is enabled by an original open-source implementation of the respective models in the R programming language. The empirical analysis of real-world and simulated data led to identification of novel phenomena: underimpact of partial supervision, related to using too small values of α , and overimpact of partial supervision, related to using too big values of α . As demonstrated in this thesis, the underimpact of partial supervision, in particular, can compromise the validity of inferences drawn from affected models. As a remedy, two model-agnostic diagnostic plots are proposed: the U-plot to detect underimpact and the O-plot to detect overimpact; these diagnostic tools serve as the basis for the proposed guidance on constructing grids of scaling factor α values in practical scenarios.

Keywords: fuzzy clustering, semi-supervised fuzzy clustering, scaling factor α , impact of partial supervision, Explainable Artificial Intelligence, Semi-Supervised Fuzzy c-Means, Semi-Supervised Possibilistic c-Means.

Streszczenie

Niniejsza rozprawa poświęcona jest wyjaśnieniu wpływu częściowego nadzoru w modelach rozmytej analizy skupień. Wykazano, że pomimo dekad postępu w tej dziedzinie, do tej pory nie został wyjaśniony kluczowy mechanizm regulowania tego wpływu za pomocą hiperparametru α nazywanego współczynnikiem skalowania - nie zbadano dotąd jego roli w kontekście całości modelu obejmującej funkcję celu, problem optymalizacyjny, algorytm rozwiązujący ten problem, oraz właściwy rezultat modelu: macierze stopni przynależności oraz prototypów skupień. Zgodnie z nurtem Wyjaśnialnej Sztucznej Inteligencji, brak wyjaśnialności takiego mechanizmu skutkuje zmniejszonym zaufaniem do wyników modelowania, które jest kluczowe zwłaszcza w analizie danych wrażliwych, takich jak dane medyczne. Wobec zidentyfikowania powyższej luki badawczej, zaproponowano nowatorski, wspólny dla wszystkich modeli rozmytej analizy skupień aparat poznawczo-badawczy dedykowany zagadnieniu wyjaśnienia wpływu częściowego nadzoru w badanej dziedzinie. Aparat ten został użyty do pełnego wyjaśnienia wpływu współczynnika α na rezultaty dwóch klasycznych modeli, Częściowo Nadzorowanych Rozmytych c-Średnich oraz Częściowo Nadzorowanych Posybilistycznych c-Średnich. Przedstawione rozważania teoretyczne stanowią podstawę do szczegółowych badań empirycznych, umożliwionych przez autorską implementację wybranych modeli w języku programowania R. W wyniku analizy rzeczywistych oraz symulowanych zbiorów danych zidentyfikowano dwa nowe zjawiska: niedostateczny wpływ częściowego nadzoru, polegający na użyciu zbyt małej wartości α , oraz nadmierny wpływ częściowego nadzoru, polegający na użyciu zbyt dużej wartości współczynnika α . Jak wykazano, szczególnie niedostateczny wpływ współczynnika α może mieć znamienne konsekwencje dla poprawności wniosków wyciąganych w oparciu o dotknięte nim modele. Wobec powyższego, zaproponowano dwa wykresy diagnostyczne do detekcji niedostatecznego wpływu (U-wykres) oraz do detekcji nadmiernego wpływu (O-wykres); w oparciu o powyższe narzędzia diagnostyczne wypracowano procedurę tworzenia siatek wartości α w praktycznych zastosowaniach.

Słowa kluczowe: rozmyta analiza skupień, częściowy nadzór, współczynnik skalowania α , wpływ częściowego nadzoru, Wyjaśnialna Sztuczna Inteligencja, Częściowo Nadzorowane Rozmyte c-Średnie, Częściowo Nadzorowane Posybilistyczne c-Średnie.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to Dr hab. Katarzyna Kaczmarek-Majer, the supervisor of my doctoral dissertation. I thank you for the years of collaboration, your scholarly guidance, kindness, and continuous care for my academic development, which together led to the completion of this dissertation.

I would also like to thank Prof. dr hab. inż. Olgierd Hryniewicz, who headed the Department of Stochastic Methods for almost the entire period during which this dissertation was prepared. Our conversations have always been a source of inspiration for me, both academically and personally.

I wish to express my gratitude to Prof. Przemysław Grzegorzewski. My studies at the Warsaw University of Technology under your supervision allowed me to develop the skills and research mindset without which I would not have dared to undertake the challenge of conducting my own research.

I also thank my colleagues for their encouragement, contributions and discussions, which supported both my development and the work presented in this thesis: Prof. Gabriella Casalino, Prof. Giovanna Castellano, Prof. Maria Brigida Ferraro, Prof. Paolo Giordani, Dr Olga Kamińska, Jakub Małecki.

On a personal note, I would like to thank my parents for their dedication to my educational development, their support and faith in my ability to rise to every challenge. I am also grateful to my parents-in-law for their support, care and precious advice. Above all, I thank my beloved wife, Paulina, for her unwavering patience, reassuring presence and encouragement throughout this ambitious journey.

Kamil Kmita

Contents

Abbreviations	1
1 Introduction	2
1.1 Background	2
1.2 Problem statement	5
1.3 Research gap	7
1.4 Illustratory example	11
1.5 Research objectives	14
1.6 Major contributions	17
1.7 Structure of the thesis	19
2 Fuzzy clustering	21
2.1 Unsupervised fuzzy clustering	21
2.1.1 Fuzzy approach to the task of clustering	21
2.1.2 Fuzzy c-Means	26
2.1.3 Possibilistic c-Means	30
2.1.4 Related work in unsupervised fuzzy clustering	36
2.2 Semi-supervised fuzzy clustering	39
2.2.1 Fuzzy clustering and partial supervision	40
2.2.2 Additive combination technique	42
2.2.3 Related work in semi-supervised fuzzy clustering	45
2.2.4 Related work on the scaling factor α	47
3 Explanations of the impact of partial supervision	51
3.1 Related work in Explainable Artificial Intelligence	52
3.1.1 Narrow understanding of explanation.	53
3.1.2 Broad understanding of explanation.	54
3.2 The proposed explainability framework for semi-supervised fuzzy clustering	58
3.2.1 Generic fuzzy clustering objective function	58

3.2.2	Outcome of fuzzy clustering model	60
3.2.3	Impact of partial supervision as a function	63
3.2.4	Supervised and non-supervised clusters	67
3.2.5	Order of clusters and prototypes	69
3.3	Explanations in Semi-Supervised Fuzzy c-Means	70
3.3.1	Semi-Supervised Fuzzy c-Means model	70
3.3.2	Impact on memberships in Semi-Supervised Fuzzy c-Means	73
3.3.3	Impact on prototypes in Semi-Supervised Fuzzy c-Means	77
3.4	Explanations in Semi-Supervised Possibilistic c-Means	81
3.4.1	Semi-Supervised Possibilistic c-Means model	81
3.4.2	Impact on typicalities in Semi-Supervised Possibilistic c-Means	82
3.4.3	Impact on prototypes in Semi-Supervised Possibilistic c-Means	84
4	Software implementation	86
4.1	Related work on replicability and reproducibility	87
4.2	Review of reproducibility in semi-supervised fuzzy clustering	91
4.3	Selected programming language	95
4.4	Proposed vectorised implementations	97
4.4.1	Notation	98
4.4.2	Fuzzy c-Means	100
4.4.3	Semi-Supervised Fuzzy c-Means	104
4.4.4	Possibilistic c-Means	107
4.4.5	Semi-Supervised Possibilistic c-Means	108
4.5	Comparison of implementations	110
5	Empirical analysis of impact of partial supervision and experiments	116
5.1	Underimpact of partial supervision	117
5.1.1	Explanation of differences in accuracy	117
5.1.2	Coincident prototypes in Semi-Supervised Possibilistic c-Means	121
5.1.3	Conditions of underimpact	123
5.1.4	The proposed visualization of underimpact	130
5.1.5	Empirical analysis of datasets with U-plots	133
5.2	Overimpact and empirical impact of partial supervision function	142
5.3	Guidance on creating grids of scaling factor α	149
5.4	Simulations - frequency of underimpact	150
5.4.1	Automated detection of underimpact	151
5.4.2	Analyses of datasets	153

CONTENTS

6 Conclusions	158
Bibliography	164
Internet Sources	181
List of Figures	182
List of Tables	184
List of Pseudocodes	185
List of Author's Publications	186

Abbreviations

AI	Artificial Intelligence
AO algorithm	Alternating Optimization algorithm
FCM, FCMeans	Fuzzy c-Means
O-plot	Overimpact-plot
PCM, PCMeans	Possibilistic c-Means
SSFC	semi-supervised fuzzy clustering
SSFCM, SSFCMeans	Semi-Supervised Fuzzy c-Means
SSL	Semi-Supervised Learning
SSPCM, SSPCMMeans	Semi-Supervised Possibilistic c-Means
U-plot	Underimpact-plot
XAI	Explainable Artificial Intelligence

Chapter 1

Introduction

1.1 Background

The term “fuzzy clustering” typically refers to a fuzzy approach to clustering, the classical unsupervised learning problem of partitioning N objects into c groups (clusters), $c < N$, such that similar objects are grouped together in the same cluster. The semi-supervised fuzzy clustering (SSFC) approach considered in this thesis extends this problem by introducing *a priori* knowledge about the class belonging of a subset of M observations, $M < N$, referred to as *partial supervision*. Such *partial* availability of labels denoting the classes of observations frequently occurs in practice due to limitations of various natures, such as the high cost of fully annotating data or methodological constraints. An example of the latter is the problem of remote monitoring of patients diagnosed with bipolar disorder: observations collected from sensors are annotated with the state of the disease (either `depression`, `mixed`, `euthymia`, or `dysfunction`) only for observations recorded on the day of stationary psychiatric assessment or near this day. Observations collected far from the day of stationary assessment are assumed to be unsupervised (Kmita et al., 2022).

This semi-supervised extension opens up many research directions. In particular, one can further focus on the partitioning problem and study how the outcomes of semi-supervised fuzzy clustering compare to those of purely unsupervised fuzzy clustering; alternatively, one can treat an SSFC model as a classifier, focusing on the classification error and comparing SSFC with other classifiers, which falls within the domain of supervised learning (Chapelle et al., 2006). Regardless of these possibilities, the semi-supervised fuzzy clustering problem remains, at its core, the same as unsupervised fuzzy clustering; thus, one should treat “fuzzy clustering” as a union of “unsupervised fuzzy clustering” and “semi-supervised fuzzy clustering.” This can be clearly seen by investigating the approach

used to estimate partitions from the data.

Let $j = 1, \dots, N$ index the objects, and let $k = 1, \dots, c$ index the clusters. Further, let $U = [u_{jk}]$ be a c -partition matrix containing the soft assignments of each j th observation to each k th cluster, denoted by $u_{jk} \in [0, 1]$. In the fuzzy approach, a given object is allowed to belong to multiple clusters to varying degrees of membership; the greater the membership u_{jk} , the more the j th object is assumed to belong to the k th cluster. This soft assignment mechanism is closely linked to the theory of fuzzy sets, which is the direct inspiration for the fuzzy approach to clustering (Ruspini et al., 2019).

Further, let $X = [\mathbf{x}_j]$, where $\mathbf{x}_j \in \mathcal{R}^p$, be the observation matrix numerically representing the objects, and let $V = [\mathbf{v}_k]$ denote the cluster prototype matrix, where a prototype $\mathbf{v}_k \in \mathcal{R}^p$ (also called a barycenter or a centroid) conceptually formalizes an abstract notion of a cluster. The similarity between objects is assessed by measuring distances between the numerical representations of objects and the cluster prototypes in a given metric space. Let d denote the distance function. A classical choice, which is also assumed in this thesis, is the Euclidean distance function.

$$d(\mathbf{x}_j, \mathbf{v}_k) = \|\mathbf{x}_j - \mathbf{v}_k\|. \quad (1.1)$$

Finally, let Θ be a set of hyperparameters, understood as model parameters that are not estimated from the data but must be configured in advance. A concrete c -partition is obtained in unsupervised fuzzy clustering by formulating the optimization problem

$$\min_{U, V} Q(U, V; X, \Theta), \quad (1.2)$$

where Q is an objective function defining a specific unsupervised fuzzy clustering model (J. C. Bezdek, 1981).

Partial supervision considered in this thesis takes the form of labels - short descriptions representing the classes of objects. Let $\mathcal{Y} = (y^1, \dots, y^c)$ be a c -tuple, an ordered collection of c labels. Semi-supervised fuzzy clustering models extend the unsupervised problem by encoding partial supervision in a specially designed matrix F of the same dimensions as the memberships matrix U (Pedrycz & Waletzky, 1997). It is constructed such that each k th column of F encodes the information about the known class belonging of observations for the class represented by $y^k \in \mathcal{Y}$. If j th observation is annotated with the class corresponding to the k th column of F , then $f_{jk} = 1$; otherwise, $f_{jk} = 0$. The rows of the matrix F corresponding to the unsupervised objects are filled with zeros.

The optimization problem for semi-supervised fuzzy clustering is formulated as follows:

$$\min_{U,V} J(U, V; X, F, \Theta), \quad (1.3)$$

where J is an objective function defining a specific SSFC model.

Although semi-supervised fuzzy clustering models can be used and studied as classifiers (in the sense that one can derive a decision function $\rho : \mathbf{x} \rightarrow \mathcal{Y}$ based on the estimated SSFC model), the above equation shows that the direct goal of SSFC is not the minimization of classification error but the optimization of partitioning, which places it in the same class of models as unsupervised fuzzy clustering. Thus, the focus and presentation of this thesis will draw heavily on the achievements of unsupervised fuzzy clustering, which are somewhat underrepresented in many SSFC works that focus only on the “supervised” component enabled by incorporating partial supervision into the equation.

With respect to how partial supervision is handled, in this thesis we investigate in detail the additive combination approach of Kmita et al. (2024), aiming to formulate the SSFC objective function as a weighted sum of the unsupervised objective function Q and a novel supervised component function Q^S . While each model proposes a different form of the supervised component Q^S , they are all based on the same idea of introducing a penalty term $\mathcal{L}(U, F)$ to guide the values of the matrix U toward the binary matrix F (for the rows corresponding to supervised objects). The semi-supervised objective function is formulated as follows:

$$J = Q(U, V; X, \Theta) + \alpha \cdot Q^S(U, V; X, F, \Theta), \quad (1.4)$$

where the scalar $\alpha \geq 0$ (called a scaling factor) is a novel hyperparameter introduced in this semi-supervised extension of the core unsupervised fuzzy clustering approach. When $\alpha = 0$, the semi-supervised problem reduces to the unsupervised one. In this sense, the introduction of partial supervision impacts the modeling outcomes compared with the case of no partial supervision, and the mechanism for regulating this impact is to configure the value of the hyperparameter α ; the greater the α , the greater the impact of partial supervision.

This thesis is devoted to studying the impact of partial supervision in semi-supervised fuzzy clustering understood as the impact of the scaling factor α . The above clarification is required because there are additional factors impacting the outcomes of modeling that are related to partial supervision; these include the ratio of the supervised observations to all observations $\frac{M}{N}$, differences in characteristics between the unsupervised and the supervised observations, and even the functional form of the penalty term $\mathcal{L}(U, F)$, **but the special meaning of the scaling factor α is that it**

is the only endogenous factor (in the sense that the user of the model can freely adjust its value) whereas all the other factors mentioned above are exogenous (in the sense that they are given and the user of the specific model cannot alter them). Therefore, in this thesis we will further on interchangeably use the terms *impact of partial supervision* and *impact of the scaling factor α* .

Finally, the English term “impact” denotes the influence of one thing on another. In the case of partial supervision in SSFC, having clarified the special role of α , we can represent this relation as follows:

$$\pi(\alpha, \mathcal{T}) : \alpha \rightarrow \mathcal{T}, \quad (1.5)$$

where $\mathcal{T}$ denotes a thing (object, element, concept) being impacted by α , which must be further specified. In contrast to the unified interest in the role of α as an impacting factor, the choice of $\mathcal{T}$ differs in the literature¹. As we show in this thesis, the consequences of choosing a specific $\mathcal{T}$ are far-reaching, as they establish a context within which the impact of partial supervision is considered. The identified lack of an appropriate context enabling an exhaustive presentation of the consequences of choosing various values of the scaling factor α has been a direct motivation for undertaking the work in this thesis, as it can greatly influence the validity of inferences drawn from the results of a specific semi-supervised fuzzy clustering model. Trust in the validity of conclusions drawn from a specific model is of critical importance for their use in highly sensitive areas, such as the above-mentioned case of remote monitoring of patients diagnosed with bipolar disorder.

1.2 Problem statement

Despite its significant role in regulating the impact of partial supervision, the scaling factor α has not been fully studied. This matter is typically raised as a side topic in the literature on the semi-supervised fuzzy clustering. Interpretations of the role of α vary in terminology, level of analytical detail, and – most importantly, as already mentioned – in the choice of $\mathcal{T}$ in Eq. (1.5), as well as in practical guidance regarding the specific values of α to use when fitting a semi-supervised fuzzy clustering model to data. To address this gap, this thesis is devoted to the following problem.

Problem statement. The key problem in the class of semi-supervised fuzzy clustering models adapting the additive combination technique (as defined in Eq. 1.4) is to explain the role of the scaling factor α in regulating the impact of partial supervision.

¹The detailed literature review is presented in Section 2.2.4.

It is the precise meaning of the term *to explain* used in the problem statement that defines the research gap this thesis is aiming to fill in. In non-scientific, common language, *to explain* means “to make something plain or understandable”² and is a synonym with the verb *to interpret*, which means “to explain or tell the meaning of”³. In this common-language sense, many explanations/interpretations of the scaling factor α already exist, viewing it as a factor regulating the impact of partial supervision.

However, on the scientific ground, the term *explanation* has a special meaning. In this thesis, we propose to build on achievements in model explainability, undertaken within the framework of Explainable Artificial Intelligence (XAI), which has become a popular umbrella term widely used by researchers (Ali et al., 2023; Saeed & Omlin, 2023). Although XAI is typically associated with research aimed at enabling comprehension of the inner logic of highly complex AI models with millions of parameters estimated from data (Ali et al., 2023), it also includes other topics related to explainability. In particular, a topic of special interest for the matter of explainability of the impact of partial supervision in semi-supervised fuzzy clustering is **the precise definition of what constitutes an explanation of any model (whether it is a highly complex deep learning model or a fuzzy clustering model) and the distinction between an interpretation and an explanation of the model or its elements** (Broniatowski, 2021; Phillips et al., 2021).

Although various approaches exist for defining an *explanation* and distinguishing it from an *interpretation*, they are all underpinned by the same motivation, which Confalonieri et al. (2021) define as “the need of conveying safety & trust in the «how» and «why» of automated decision-making”. An explanation of the model or its elements should be of such a nature and quality as to ensure the required level of safety and trust. Consequently, not every description that explains the impact in the common-language sense – i.e., *tells the meaning of the scaling factor α in regulating the impact of partial supervision* or *makes the role of the scaling factor α in regulating the impact of partial supervision plain or understandable* – can be considered a proper scientific explanation of the mechanism by which partial supervision impacts semi-supervised fuzzy clustering models, conveying the appropriate safety and trust in the “how” and “why” of the inferences drawn from these models.

In this thesis, we define an explanation of the impact of partial supervision based on a specific approach to model explainability, as presented by Broniatowski (2021) and ISO/IEC TS 6254:2025 (2025). Broniatowski (2021) provided a simple but concrete requirement: “an explanation of a model result is a description of how a model’s outcome

²<https://www.merriam-webster.com/dictionary/explain> (accessed on 31 January 2026)

³<https://www.merriam-webster.com/dictionary/interpret> (accessed on 31 January 2026)

came to be”. Applying this to the problem of explaining the impact of partial supervision, the above requirement postulates that $\mathcal{T}$ in the relation $\pi(\alpha, \mathcal{T})$ defined in Eq. (1.5) corresponds to the *outcome of the model*. In our approach, the precise definition of this *outcome* is based on the definition of a “decision” of any Artificial Intelligence model⁴ presented in ISO/IEC TS 6254:2025 (2025) as a “content or item produced by the AI system [understood as SSFC model in the context of this thesis] as a fulfilment of its task, based on a given input”. Throughout this thesis, we will use the terms *outcome* and *decision* interchangeably to refer to the above definition.

The decision/outcome of any fuzzy clustering model – unsupervised or semi-supervised – is a pair of numerical matrices that minimize the respective objective function: let $\tilde{U}$ denote the estimated memberships matrix, which includes the estimated degrees of membership of objects to the clusters, and let $\tilde{V}$ denote the estimated clusters’ prototypes matrix.

Therefore, we conclude that an explanation requires specifically that $\mathcal{T} \equiv (\tilde{U}, \tilde{V})$, and that the impact of partial supervision in SSFC is a relation

$$\pi(\alpha, (\tilde{U}, \tilde{V})) : \alpha \rightarrow (\tilde{U}, \tilde{V}). \quad (1.6)$$

Any other description of the impact of partial supervision, considering a different context $\mathcal{T}$, cannot be regarded as an explanation, but only as an interpretation. We formalize the above postulate in the form of the following definitions.

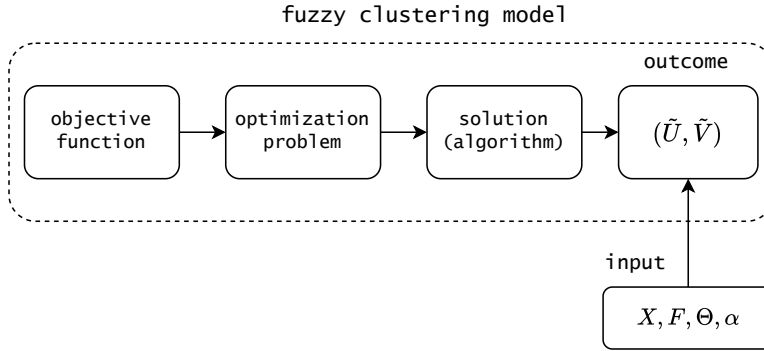
Definition 1 (Interpretation). *The interpretation of the role of the scaling factor α is a human-understandable, meaningful description of the role of α that puts it in a certain context.*

Definition 2 (Explanation). *The explanation of the role of the scaling factor α is a human-understandable, meaningful description of the role of α that describes its impact on the outcome of semi-supervised fuzzy clustering model $(\tilde{U}, \tilde{V})$.*

1.3 Research gap

Based on the definition in ISO/IEC TS 6254:2025 (2025), the outcome $(\tilde{U}, \tilde{V})$ of any SSFC model represents the fulfilment of its task and should be understood as a solution to the corresponding optimization problem.

⁴ The definition of an AI system proposed in ISO/IEC TS 6254:2025 (2025) is inclusive, treating both fuzzy clustering and highly complex deep neural networks equally as AI systems, since it defines an AI system as “engineered system that generate outputs such as content, forecasts, recommendations or decisions for a given set of human-defined objectives”.

Figure 1.1: The entire context of the fuzzy clustering model.

Source: own work.

To explain how the scaling factor α impacts this decision, one must consider the entire fuzzy clustering model, which consists of the following elements, depicted in Fig. 1.1:

- the objective function $J(U, V)$, where U and V denote the variables to be optimized: the memberships matrix and the prototypes matrix, accordingly. The objective function is critical in formulating the optimization problem, but represents only the starting point of the overall analytical procedure,
- the optimization problem, together with the assumed constraints,
- the solution to the optimization problem, which in case of fuzzy clustering is an approximate solution,
- the outcome $(\tilde{U}, \tilde{V})$,
- the input X, F, Θ, α , where X denotes the observations (the data), F denotes partial supervision, and Θ is the set of all respective hyperparameters except of α .

Regarding the state of the art, descriptions and analyses of the role of the scaling factor α are motivated by a key need inherent in the use of any hyperparameter in any AI model: the need to configure its value when fitting the model to the data. Two approaches can be distinguished in this regard, which we propose to call the *balancing approach* and the *grid approach*.

Balancing approach. The name of the first approach, the balancing approach, is meant to emphasize its connection with the specific interpretation of the scaling factor α provided in the publication that introduced the first classical SSFC model. Pedrycz and Waletzky (1997) proposed their Semi-Supervised Fuzzy c-Means model by introducing

the idea of using the scaling factor $\alpha \geq 0$ and described that its role is “to maintain a balance between the supervised and unsupervised component within the optimization mechanism” (Pedrycz & Waletzky, 1997). Despite decades of advances in semi-supervised fuzzy clustering and the many novel, often complex models proposed during this time, the original description by Pedrycz and Waletzky (1997) has remained the prevalent approach for interpreting the role of the scaling factor α . Other researchers have sometimes used different wording. For example, Antoine et al. (2022) described the scaling factor as “a trade-off coefficient between the inherent structure retrieved without any supervision and the consideration of the labeled patterns,” and Gan et al. (2018) stated that it is “a trade-off parameter that controls the balance between the unsupervised component and the given class labels”. Nevertheless, none of these (or other) interpretations change the core meaning of the scaling factor α , which is presented in the context of the additive combination technique.

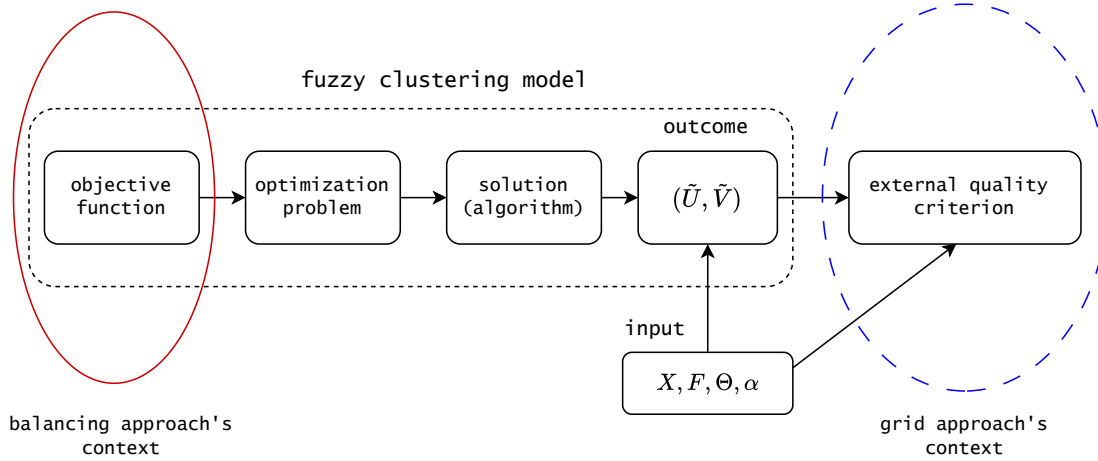
In the balancing approach, a single value of α is considered to achieve the balance. Two main concepts are proposed in this regard: either to tie the value of α to the ratio of the unsupervised to supervised observations, $\frac{N}{M}$, or to use a data-independent value, $\alpha = 1$. The motivation for the first concept is that frequently, $M \ll N$, and a data-dependent value of α would result in fair weighting of the components of the objective function J (Pedrycz & Waletzky, 1997). The motivation for the data-independent concept is that setting $\alpha = 1$ results in equal weights, reducing the weighted sum in Eq. (1.4) to a simple sum.

The problem with the balancing approach is that it presents the role of the scaling factor α in regulating the impact of partial supervision only within the context of a specific method for constructing the objective function, namely, the additive combination mechanism. This context is depicted by the solid red ellipsis in Fig. 1.2. Clearly, none of such descriptions can be considered an explanation, since they all associate $\mathcal{T}$ with J and discuss the relation $\pi(\alpha, J) : \alpha \rightarrow J$.

Grid approach. In practice, the majority of the publications does not consider a single α value, but a finite set of candidate values $\mathcal{A} \subset \mathcal{R}_+$, from which the optimal value of the scaling factor α is selected $\mathcal{A}^5$. Formally, we introduce the order and define $\mathcal{A}$ as a q -tuple $\mathcal{A} = (\alpha_1, \dots, \alpha_q)$; it is also referred to as a grid. For convenience, let M_α denote the decision of a given SSFC model parameterized by the scaling factor α . The hyperparameter tuning problem can be formalized as:

$$\text{optimize Crit}(M_\alpha) \quad \text{s.t. } \alpha \in \mathcal{A}, \quad (1.7)$$

⁵The detailed literature review is presented in Section 2.2.4.

Figure 1.2: The contexts of balancing and grid approaches.


Source: own work.

where $\text{Crit}(\cdot)$ denotes an external quality criterion used to evaluate the decision of a given model M_α , for example, an accuracy metric measuring the model's performance in classification, or a partition coefficient measuring the model's quality in clustering. These two criteria are discussed in more detail in Section 1.4.

The presentation in Eq. (1.7) is intended to emphasize that the value of the optimal argument, α^* , is not the primary focus of this approach. Analyzing the impact of various values $\alpha \in \mathcal{A}$, researchers are focused on a relation

$$\pi(\alpha, \text{Crit}(M_\alpha)) : \alpha \rightarrow \text{Crit}(M_\alpha) \quad (1.8)$$

rather than on the direct impact of the scaling factor α on the model's decision $(\tilde{U}, \tilde{V})$. Consequently, none of these descriptions can be considered a proper explanation of the scaling factor α , as they assume $\mathcal{T} \equiv \text{Crit}(\tilde{U}, \tilde{V})$. This context of the grid approach is depicted by the dashed blue ellipsis in Fig. 1.2 as clearly external to the fuzzy clustering model's context.

Despite the major issue of the indirect nature of analyses in the grid approach, the mere act of constructing a grid could represent an initial step toward providing an explanation, by considering the functional form of the impact of partial supervision as a function of α , i.e., $\pi(\alpha)$.

This is because creating the grid involves decisions regarding: (i) the grid boundaries, α_1 and α_q , which determine the range of values to be examined; (ii) the number of elements in the grid, equal to q , which is associated solely with the computational cost acceptable within a given experiment; and (iii) the grid scale, defined as the spacing

between consecutive elements.

However, the above aspects of grid construction are rarely discussed explicitly in works addressing the topic of semi-supervised fuzzy clustering. Typically, one can only implicitly infer the understanding of the scale by examining the specific $\mathcal{A}$ grid used. Moreover, the approaches to determining the scale of the α hyperparameter differ. For example, Golzari Oskouei et al. (2024) use a linear scale, $\mathcal{A}_1 = \{1, 2, 4\}$, whereas Gan et al. (2024) use a non-linear scale, $\mathcal{A}_2 = \{0.001, 0.01, 0.1, 1, 10, 100\}$; however, neither work justifies the selection of the scale.

A similar problem appears with the analysis of the boundaries of the grid: in the examples above, the motivation for selecting the specific boundaries, α_1 and α_q , is not disclosed. Such a motivation is provided for example by Antoine et al. (2022), who use a non-linear grid $\mathcal{A}_3 = \{0.01, 0.05, 0.1, 0.5, 1\}$ and justify that “too low values [of the scaling factor α] make the algorithm neglect labels, and inversely, too high values force the algorithm to respect labels but do not ensure an overall coherent solution with compact classes”, treating the value of $\alpha = 1$ as *high*. This contradicts the notion of a high value of the scaling factor α implicitly suggested e.g. by Gan et al. (2024), where $\max(\mathcal{A}_2) = 100$. It must be emphasized that this interpretation is not based on any formal analysis, which is symptomatic of the debate on the role of the scaling factor α in the field of semi-supervised fuzzy clustering.

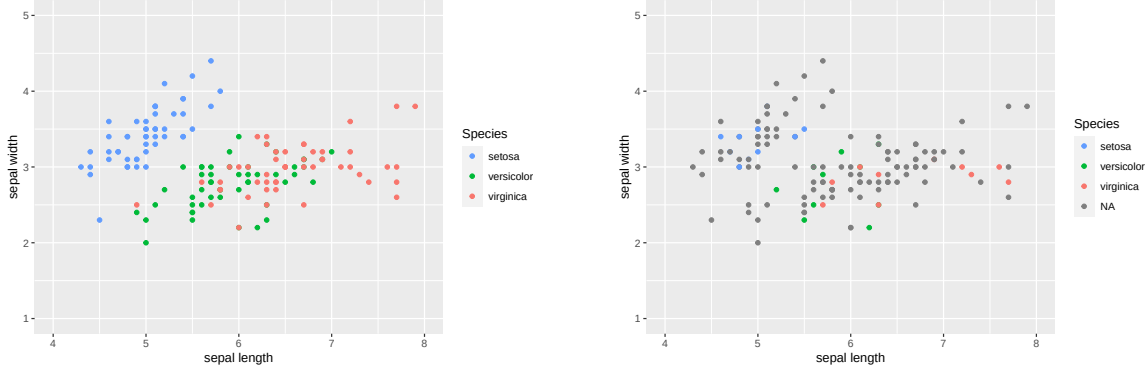
Summary. There exists a research gap concerning the problem addressed in this thesis, since none of the existing descriptions of the scaling factor α examines its impact on the decision $(\tilde{U}, \tilde{V})$ of semi-supervised fuzzy clustering models, which is required for a proper explanation of this impact. Moreover, the proposed grids of α values in the literature vary significantly, without any scientific framework to enable a thorough comparison of these competing interpretations of the impact of partial supervision, understood as a function of the scaling factor α .

1.4 Illustratory example

To illustrate the practical consequences of the lack of explainability, and how it can lead to reduced trust in the results of SSFC models, let us consider the well-known `iris` dataset (R. A. Fisher, 1936), which will be used throughout this thesis. In particular, to enable clear visualization in the $\mathcal{R}^2$ space, we analyze two features of the iris dataset: sepal length and sepal width. Fig. 1.3a shows the data, with colours indicating the true class of each observation: `setosa`, `versicolor`, or `virginica`. Clearly, the `setosa` irises are well separated from the other species in this space, whereas `versicolor` and

virginica significantly overlap, making this dataset a suitable case for fuzzy clustering, which handles such ambiguous phenomena through its soft membership mechanism.

Figure 1.3: Dataset *iris* (R. A. Fisher, 1936).



(a) Fully supervised data.

(b) Simulated partial supervision ($\frac{M}{N} = 0.2$).

Source: own work.

We simulate partial supervision by retaining the class belonging information for only 20% of randomly sampled observations from each class, as shown in Fig. 1.3b. This approach – simulating partial supervision by removing labels from a fully supervised dataset – enables a thorough investigation of the clustering models’ results against a reference partition. Introducing partial supervision, we set an arbitrary order of classes:

$$\mathcal{Y} = (\text{setosa}, \text{versicolor}, \text{virginica})$$

and construct the supervision matrix F accordingly. In our experiment, we fit four models to the data, varying only the value of the scaling factor α :

1. M_0 : a classical unsupervised Fuzzy c-Means,
2. $M_{0.1}$: a classical Semi-Supervised Fuzzy c-Means with $\alpha = 0.1$,
3. M_1 : SSFCM with $\alpha = 1$ that adheres to the $\alpha = 1$ heuristic from the balancing approach,
4. M_5 : SSFCM with $\alpha = 5$ that adheres to the $\alpha = \frac{N}{M}$ rule from the balancing approach.

The experiment relies on calculating four classical external quality criteria $\text{Crit}(\cdot)$: one related to classification quality (accuracy) and three related to clustering quality.

To enable measuring accuracy as an external quality criterion, we further split the observation matrix into two matrices: X_{train} , used to estimate the decision of the model, and X_{test} , used to measure accuracy on independent data set. The test matrix X_{test}

is created by balanced random sampling of 10% of observations from each of the three classes.

Treating any SSFC model as a classifier involves creating a decision rule $\rho : \mathcal{R}^p \rightarrow \mathcal{Y}$ that predicts the label associated with the prototype closest to a given observation:

$$\rho(\mathbf{x}) = y^{k^*}, \quad \text{where } k^* = \arg \min_k d(\mathbf{x}, \tilde{\mathbf{v}}_k). \quad (1.9)$$

In addition to accuracy, we consider three other criteria: the partition coefficient (PC) proposed by J. C. Bezdek (1974), the clustering validity function proposed by Xie and Beni (1991) (abbreviated as XB, after the authors' surnames), and the Silhouette index (Rousseeuw, 1987). PC is a function of the decision memberships and takes values in the interval $[\frac{1}{c}, 1]$. Lower PC values indicate fuzzier memberships and correspond to low separation between clusters. Therefore, higher PC values are desired. XB is a function of the triplet $(\tilde{U}, \tilde{V}, X_{\text{train}})$ and has only a lower bound of 0. It is a relative measure, with lower values indicating better clustering results. Unlike the fuzzy criteria PC and XB, the Silhouette index is a non-fuzzy measure that evaluates how similar an object is to its own cluster relative to other clusters. The index takes values in the interval $[-1, 1]$, with higher Silhouette values being desired, as they indicate better-defined clusters.

The results of applying the above external quality criteria $\text{Crit}(\cdot)$ functions are presented in Table 1.1.

Table 1.1: Values of the considered external quality criteria.

Crit	M_0	$M_{0.1}$	M_1	M_5	range
Partition Coefficient	0.709	0.684	0.726	0.757	$[0.33, 1]$
Xie-Bieni	0.155	0.2	0.168	0.194	> 0
Silhouette index	0.619	0.622	0.561	0.559	$[-1, 1]$
accuracy	NA	0.083	0.833	0.833	$[0, 1]$

Source: own work.

We can observe that for the clustering quality criteria, the results of all four models are stable, with only minor fluctuations, whereas for the classification quality criterion (i.e., accuracy), the results differ substantially between $\text{acc}(M_{0.1}) = 0.083$ and $\text{acc}(M_1) = \text{acc}(M_5) = 0.833$. The accuracy of $\text{acc}(M_{0.1}) = 0.083$ is lower than the expected accuracy from random class prediction on the test set, which is surprising given the simplicity of this scenario.

The motivation for this illustrative example is to show that even in simple scenarios, when focusing on external quality criteria, one may encounter contradictory and surprising conclusions that cannot be easily explained. Intuitively, the low value of the scaling factor,

$\alpha = 0.1$, appears to cause a problem, as higher values, $\alpha \in \{1, 5\}$, result in much higher accuracy. However, without the explainability framework proposed in this thesis, further insight into this matter is not possible.

We call the underlying phenomenon *the underimpact of partial supervision* and study this untoward phenomenon, associated with using *too low values of the scaling factor*, in detail in Section 5.1. As we shall show, this phenomenon is stochastic in nature, and the illustrative example presented above was specifically chosen to demonstrate its occurrence. The frequency of occurrence of underimpact is further studied in Section 5.4 using an appropriate simulation protocol.

A related phenomenon, not discussed in this Section, concerns the opposite case, where *too big values of the scaling factor* are used. We call this *the overimpact of partial supervision* and study it in Section 5.2. Finally, the explainability framework enables additional insights that may not be as immediately visible as the underimpact of partial supervision. In particular, explanations of the impact of partial supervision provide new understanding of the semantics of membership degrees in semi-supervised fuzzy clustering.

1.5 Research objectives

The lack of explainability regarding the impact of partial supervision in semi-supervised fuzzy clustering – understood as insufficient analysis of the role of the scaling factor α within the full context of a fuzzy clustering model – has led to the underdevelopment of both the theoretical foundations and practical considerations, which this thesis aims to address.

Primary Objective 1. To introduce explainability framework for impact of partial supervision in semi-supervised fuzzy clustering.

As the famous philosopher Ludwig Wittgenstein stated, “The limits of my language mean the limits of my world” (Wittgenstein, 1981). In light of this proposition, to expand the limits of the world, one must first expand their language. The proposed *explainability framework* refers to the metaphor of the language - it is an arbitrary collection of formalism, notation, definitions, and conceptualization of certain phenomena that has not been previously proposed in the literature. It is devoted to explaining the impact of partial supervision (understood as in Eq. 1.6) in any semi-supervised fuzzy clustering model that adapts the additive combination technique (understood as in Eq. 1.4).

Part of this Primary Objective 1 is also to justify the definition of the term *explanation*, which requires to review and build on the achievements in the general study of model

explainability, now mostly conducted on the grounds of so-called Explainable Artificial Intelligence (XAI).

Primary Objective 2. To explain the impact of partial supervision in two classical SSFC models: Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means.

Having established the explainability framework, we apply it to provide proper explanations of the impact of partial supervision on the outcomes of two classical models, Semi-Supervised Fuzzy c-Means (SSFCMeans) and Semi-Supervised Possibilistic c-Means (SSPCMeans). These models represent two competing approaches to fuzzy clustering and serve as the building blocks for numerous derivative models. Explaining the impact of partial supervision for these models provides a theoretical framework that can be readily applied to any other semi-supervised fuzzy clustering model.

Primary Objective 2 also includes the study of semantic issues related to membership degrees, which are important for understanding the connection between fuzzy clustering and the theory of fuzzy sets, as well as the practical derivation of functions $\pi_U(\alpha)$ and $\pi_V(\alpha)$ that quantify the impact of partial supervision on the elements of any fuzzy clustering model's outcome, $(\tilde{U}, \tilde{V})$, as explicit functions of the scaling factor α .

Secondary Objective 1. To create novel modular open-source software implementation of selected semi-supervised fuzzy clustering models.

Due to the identified lack of reproducible open-source software implementations for classical semi-supervised fuzzy clustering models, this Secondary Objective 1 is devoted to providing novel open-source software implementations of Fuzzy c-Means, Possibilistic c-Means, and their semi-supervised counterparts: Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means, using the R programming language.

In addition to the lack of available software, there is also a lack of an alternative algorithmic presentation of the models' estimation procedures. Existing literature typically provides pseudo-code that relies on direct use of `for` loops, which is generally inefficient in modern high-level programming languages such as R. The part of Secondary Objective 1 is to thus present a novel vectorized, formalized description of the classical baseline fuzzy clustering models, providing a convenient foundation for implementing not only the proposed models but also any derivative models.

Secondary Objective 2. To identify and provide empirical analysis of underimpact of partial supervision.

The theoretical work carried out under the Primary Objectives, together with the software delivered under Secondary Objective 1, enables the identification and empirical analysis of the underimpact of partial supervision. This negative, stochastic phenomenon occurs when *too small* values of the scaling factor α are used and can compromise the validity of inferences drawn from affected models.

The goal of Secondary Objective 2 is to precisely define the underimpact of partial supervision, explain the conditions that lead to its occurrence, and provide a diagnostic tool in the form of a dedicated underimpact plot, as well as a concrete procedure describing how to avoid underimpact of partial supervision, labeled as Rule 1. In addition, simulation protocols are provided, and the frequency of underimpact occurrences is analyzed under Secondary Objective 2.

Secondary Objective 3. To identify and provide empirical analysis of overimpact of partial supervision.

The theoretical work carried out under the Primary Objectives, together with the software delivered under Secondary Objective 1, enables the identification and empirical analysis of the overimpact of partial supervision. This potential negative phenomenon occurs when *too big* values of the scaling factor α are used, since, in theory, there is no upper bound on the values of α .

The goal of Secondary Objective 3 is to precisely define the overimpact of partial supervision, explain its potential manifestations, and provide a diagnostic tool in the form of a dedicated overimpact plot that enables empirical investigation of whether overimpact occurs in a given scenario. The overimpact plot is shown to be an effective tool for estimating the empirical impact of partial supervision - the functional form of the impact of the scaling factor α under concrete exogenous experimental factors, including the dataset, its characteristics, the applied partial supervision, and the configuration of the remaining hyperparameters of the model.

Secondary Objective 4. To provide guidance on how to create grids of the scaling factor α values.

Since the scaling factor α is formally a hyperparameter of the semi-supervised fuzzy clustering model, researchers are often interested in selecting its optimal value with respect to an external criterion - for example, fitting a grid of models with different α values and

selecting the model that achieves the highest accuracy on the test dataset for further analyses.

This approach, known as hyperparameter tuning, depends on the quality of the constructed grids of α values. Secondary Objective 4 builds on the work carried out under Secondary Objectives 2 and 3 and provides guidance on each aspect of creating grids of the scaling factor α values: (i) the grid boundaries, α_1 and α_q , which determine the range of values to be examined; (ii) the number of elements in the grid, equal to q , which relates directly to the computational cost acceptable within a given experiment; and (iii) the grid scale, defined as the spacing between consecutive elements

1.6 Major contributions

1. Framework for explaining the impact of partial supervision in semi-supervised fuzzy clustering.

Section 3.2, composed of five subsections, presents a coherent collection of formalism, notation, definitions, considerations, and conceptualizations that have either not been discussed in the literature or have been addressed only to a very limited extent. We collectively refer to the contents of Section 3.2 as the *explainability framework*. The framework constitutes the primary contribution, as all other contributions are developed using it.

The special role of the scaling factor α as the only endogenous factor regulating the impact of partial supervision has been emphasized; therefore, the phrases *impact of partial supervision* and *impact of the scaling factor α* are used interchangeably throughout the thesis.

2. Explanations of impact of the scaling factor α in Semi-Supervised Fuzzy c-Means.

Section 3.3.2 explains the impact of the scaling factor α on the decision memberships $\tilde{U}$ of the model and concludes with formal Explanation 1. Section 3.3.3 explains the impact of the scaling factor α on the decision prototypes $\tilde{V}$ of the model and concludes with formal Explanation 2.

Moreover, the issue of the semantics of the degrees of membership in the semi-supervised case is discussed, and the conclusion, presented as Interpretation 2, addresses the importance of focusing on the impact of α on prototypes, **which constitutes a novel proposition.**

3. Explanations of impact of the scaling factor α in Semi-Supervised Possibilistic c-Means.

Section 3.4.2 explains the impact of the scaling factor α on the decision memberships $\tilde{U}$ of the model and concludes with formal Explanation 4. Section 3.4.3 explains the impact of the scaling factor α on the decision prototypes $\tilde{V}$ of the model and concludes with formal Explanation 5.

4. Formal description of novel software implementation of four classical fuzzy clustering models.

Section 4.4 presents a formal, mathematical description of novel vectorized formulae for the optimal memberships $\hat{U}$ and $\hat{V}$ for four models: unsupervised Fuzzy c-Means, unsupervised Possibilistic c-Means, Semi-Supervised Fuzzy c-Means, and Semi-Supervised Possibilistic c-Means. The novelty lies in the vectorized form of matrix operations. Until now, optimal memberships have only been formulated at the level of individual elements $\hat{u}_{jk}$ (scalars), and optimal prototypes at the level of individual vectors $\hat{\mathbf{v}}_k$.

5. R package `fussclust` implementing four classical fuzzy clustering model.

A dedicated R package called `fussclust` (*Fuzzy Unsupervised and Semi-Supervised Clustering*) was created and is available at <https://github.com/kkmita/fussclust> under MIT license.

6. Identification and analysis of underimpact of partial supervision.

In Section 5.1, the phenomenon of underimpact of partial supervision is formulated for the first time in the literature and analyzed. It pertains to the reordering of rows in the prototypes matrix when random initialization of the optimization algorithm is untoward partial supervision, and the value of the scaling factor α is too small to counteract this effect.

7. Identification and analysis of overimpact of partial supervision.

In Section 5.2, the potential phenomenon of overimpact is formulated for the first time in the literature and analyzed. It pertains to excluding the impact of any unsupervised observation as α diverges towards infinity. Moreover, the associated diagnostic overimpact-plot is shown to be an effective tool for estimating the empirical impact of partial supervision on the decision prototypes $\tilde{V}$.

8. Guidance on creating grids of the scaling factor α values.

Section 5.3 provides concise guidance on constructing grids of α values, $\mathcal{A} = (\alpha_r)_{r=1,\dots,q}$, including the selection of boundary values a_1 and a_q , the number of elements q , and the spacing between consecutive elements of the grid. This guidance is based on the analysis of the underimpact and overimpact phenomena.

1.7 Structure of the thesis

This thesis is composed of six numbered Chapters devoted to the problem of explainable impact of partial supervision in semi-supervised fuzzy clustering, and seven unnumbered auxiliary Chapters, providing the list of abbreviations, the bibliography and internet sources used, the lists of figures, tables and pseudocodes, and the list of author's publications.

With respect to the numbered chapters, Chapter 1 *Introduction* presents the problem addressed in this thesis. It begins with the background Section 1.1, which provides the necessary preliminaries, followed by sections devoted to the problem statement, research gap, and an illustrative example motivating the work undertaken in this thesis. Chapter 1 concludes with formulation of research objectives, major contributions of this thesis, and the description of the structure of the thesis.

Chapter 2 *Fuzzy clustering* is composed of two sections: Section 2.1 *Unsupervised fuzzy clustering*, and Section 2.2 *Semi-supervised fuzzy clustering*, which introduce the two subclasses of the general class of fuzzy clustering models. This Chapter does not include any novel propositions; it provides a summary of the state of the art and the necessary preliminaries.

Chapter 3 *Explanations of the impact of partial supervision* is related to Primary Objectives 1 and 2; it is composed of four sections. Section 3.1 discusses related work on explainability in the field of Explainable Artificial Intelligence (XAI). Section 3.2 presents a proposed explainability framework for semi-supervised fuzzy clustering. Sections 3.3 and 3.4 provide explanations of the impact of the scaling factor α in the Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means models, respectively.

Chapter 4, *Software Implementation*, addresses the novel implementation of four classical fuzzy clustering models in the R programming language: Fuzzy c-Means, Possibilistic c-Means, Semi-Supervised Fuzzy c-Means, and Semi-Supervised Possibilistic c-Means. It begins with a general discussion of reproducibility and replicability (Section 4.1), followed by a review of reproducibility in the field of semi-supervised fuzzy clustering (Section 4.2). Section 4.4 provides a formal description of the novel algorithm, and Section 4.5 concludes

the chapter with a comparison of the efficiency of the proposed algorithm against state-of-the-art R packages.

Chapter 5, *Empirical Analysis of the Impact of Partial Supervision and Experiments*, is devoted to the empirical study of the impact of the scaling factor α . Section 5.1 introduces and analyzes in detail the phenomenon of underimpact of partial supervision. Section 5.2 addresses the phenomenon of overimpact of partial supervision. Section 5.3 provides guidance on creating grids of α values based on the conclusions of the two preceding sections. Finally, Section 5.4 provides estimates of the frequency of underimpact (which is a stochastic phenomenon) based on the established simulation protocols.

The thesis is concluded by Chapter 6 *Conclusions*.

Chapter 2

Fuzzy clustering

In the introductory Section 1.1 of this thesis we stated that both unsupervised and semi-supervised fuzzy clustering models should be treated as special cases of a general class of fuzzy clustering models, as they share the same problem formulation, and the solution of it. In light of this statement, in this Chapter we present the necessary theoretical preliminaries related to fuzzy clustering, starting with the unsupervised problem, and further moving on to the semi-supervised problem. We recall basic definitions, set up the common notation, and present the related work on the topic of fuzzy clustering and the impact of partial supervision.

2.1 Unsupervised fuzzy clustering

Since semi-supervised fuzzy clustering emerged as an extension of unsupervised fuzzy clustering, it is important to describe the foundations and developments in this baseline field of unsupervised fuzzy clustering which itself emerged as an extension to crisp clustering (Jain, 2010).

2.1.1 Fuzzy approach to the task of clustering

Hastie et al. (2009) define that “the goal of cluster analysis is to partition the observations into groups («clusters») so that the pairwise dissimilarities between those assigned to the same cluster tend to be smaller than those in different clusters”. Frequently, as studied in this thesis, the similarity is formalized as a distance between the observations in a given metric space - in such a way that small distances correspond to high similarities between the objects.

A multitude of clustering models has been proposed in the literature, with novel methods still emerging. Aggarwal and Reddy (2018) categorized the most popular models

into the following classes:

1. probabilistic and generative, in which it is assumed that data is generated by a specific process, and the parameters of the underlying distribution are to be estimated,
2. density and grid-based, in which a density is understood as the number of points in a given neighbourhood of each data point,
3. distance-based, which are close to *generative*, but do not assume any underlying probabilistic distribution.

The class of *distance-based* models is further split into hierarchical models, that consider a sequence of nested partitions, and partitional (or flat) models, that assume a number of clusters is prespecified in advance. **Fuzzy clustering studied in this thesis belongs to the distance-based partitional class of clustering models.** Therefore, in this thesis, the scope is limited to the fuzzy, distance-based, partitional clustering models and these models are subject to the literature review. Other models are not discussed due to the lack of space.

One of the first partitional algorithms, popular up until now, is the k -means model, proposed first by Lloyd (1982)¹. It defines the clustering task as the optimization problem to find a partition minimizing the squared error between points in the cluster and the empirical average of the points in the cluster.

Let $\Omega = \{\omega_j\}, j = 1, \dots, N$, be a set of N objects, and $X = \{\mathbf{x}_j\}, \mathbf{x}_j \in \mathcal{R}^p$, a set of observations - corresponding numerical representations of the objects from Ω . Further, let $\Psi = \{\psi_1, \dots, \psi_c\}$ be a set of c clusters. The empirical average of observations belonging to the k th cluster is denoted by $\bar{\psi}_k$. The optimization problem, according to the modern description by Jain (2010), presents as

$$\min_{\Psi} \sum_{k=1}^c \sum_{\mathbf{x}_j \in \psi_k} \|\mathbf{x}_j - \bar{\psi}_k\|^2. \quad (2.1)$$

This problem is known to be NP-hard (Drineas et al., 2004), and hence many approximate solutions have been proposed over the years, including genetic algorithms (Hall et al., 1999; Pacheco & Valencia, 2003), or particle swarm optimization (Dey & Mukhopadhyay, 2011). However, J. C. Bezdek (2025) mentions that the algorithm that has been most extensively described in the literature and most commonly applied for approximate solution of Problem (2.1) is a technique frequently referred to as Alternating Optimization (AO)

¹J. C. Bezdek (2025) points to this work, with a caveat that original Lloyd's report remained unpublished for many years.

Algorithm 1 Alternating Optimization Algorithm for k -means (Jain, 2010)

- 1: Initialize the algorithm with $\bar{\psi}_k$ for each k th cluster
- 2: **repeat**
- 3: assign each ω_j to the l th cluster by finding

$$\arg \min_{k \in \{1, \dots, c\}} \|\mathbf{x}_j - \bar{\psi}_k\|^2$$

- 4: update

$$\bar{\psi}_k = \frac{1}{|\psi_k|} \sum_{j \in \psi_k} \mathbf{x}_j \quad \forall k,$$

- 5: **until** no change in clusters' assignments Ψ
 - 6: **Output:** clusters' assignment Ψ , empirical averages $\bar{\Psi}$
-

that is studied in this thesis. The AO algorithm to approximately solve Problem (2.1) is summarized in the steps of Algorithm 1.

Problem (2.1) and its solutions are referred to as to *crisp* (or *hard*) clustering, as each object ω_j is assigned unequivocally to a single cluster only. The idea of fuzzy clustering, considered in this thesis, was to relax this constraint. Ruspini et al. (2019) stress that fuzzy clustering is inherently related to the idea of fuzzy sets introduced by Zadeh (1965) as an extension of classical, *crisp* sets. Following a modern notation proposed by Zimmermann (2001), let a fuzzy set A be defined as follows:

$$A = \{(\omega, \mu_A(\omega)) \mid \omega \in \Omega\}, \quad (2.2)$$

where $\mu_A : \Omega \mapsto [0, 1]$ is called a membership function. The value $\mu_A(\omega)$ is called a membership degree, or degree of belongingness, and “measures the amount to which ω is a member of A ” (Ruspini et al., 2019). Based on the idea of fuzzy sets, fuzzy clustering then is a task of finding a partition matrix, called a membership matrix, $U : X \mapsto [0, 1]^c$, where every object represented by $\mathbf{x}_j$ can belong to multiple clusters to a certain membership degree, i.e., $U : \mathbf{x}_j \mapsto (u_1(\mathbf{x}_j), \dots, u_c(\mathbf{x}_j))$, where $u_k(\mathbf{x}_j) \in [0, 1]$.

In the classical work, J. C. Bezdek (1981) provides a following definition of a fuzzy c -partition space:

Definition 3 (Fuzzy c -partition space (J. C. Bezdek, 1981)). *Let $\mathcal{R}_{Nc}$ denote a set of real $N \times c$ matrices. Then, the fuzzy c -partition space for a finite set composed of N observations is defined as*

$$M_{fc} = \{U \in \mathcal{R}_{Nc} \mid u_{jk} \in [0, 1]; \sum_{k=1}^c u_{jk} = 1; 0 < \sum_{j=1}^N u_{jk} < N\}. \quad (2.3)$$

Using this nomenclature, one can also define a hard c -partition space, which translates the suite of possible solutions for the hard k -means model into a unified language of partitions:

Definition 4 (Hard c -partition space (J. C. Bezdek, 1981)). *Let $\mathcal{R}_{Nc}$ denote a set of real $N \times c$ matrices. Then, the hard c -partition space for a finite set composed of N observations is defined as*

$$M_c = \{U \in \mathcal{R}_{Nc} | u_{jk} \in \{0, 1\}; \sum_{k=1}^c u_{jk} = 1; 0 < \sum_{j=1}^N u_{jk} < N\}, \quad (2.4)$$

and we have $M_c \subset M_{fc}$.

The obvious advantage of the fuzzy approach is that it enables such partitions that reflect the intuitive properties of mixed physical processes, or phenomena. J. C. Bezdek (1981) provides an example of partitioning three fruits: ω_1 being a peach, ω_2 being a nectarine, and ω_3 being a plum, where a nectarine is known to be a hybrid of a peach and a plum. In this case, there are three hard 2-partitions possible (adjusted for the order of rows in the matrix)

$$\begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad (2.5)$$

and neither of them reflects the characteristics of a nectarine being similar to the peach and the plum at the same time.

On the contrary, there exists infinitely many fuzzy 2-partitions – including the three hard 2-partitions presented above – and some of the fuzzy 2-partitions better reflect this scenario. J. C. Bezdek (1981) provides an example of such a concrete fuzzy 2-partition

$$\begin{array}{cc} & k = 1 \quad k = 2 \\ \omega_1 = \text{peach} & \begin{pmatrix} 0.91 & 0.09 \end{pmatrix} \\ \omega_2 = \text{nectarine} & \begin{pmatrix} 0.58 & 0.42 \end{pmatrix} \\ \omega_3 = \text{plum} & \begin{pmatrix} 0.13 & 0.87 \end{pmatrix} \end{array} \quad (2.6)$$

where clearly, the first cluster becomes associated with a concept of a peach, the second cluster with the concept of a plum, and the balanced degrees of belonging of a nectarine to each of the clusters – $u_{21} = 0.58$, and $u_{22} = 0.42$, respectively – reflect its properties as being a hybrid of these two fruits.

Fuzzy clustering can also outperform hard clustering from a purely computational perspective (even in unambiguous scenarios), because – as Klawonn and Hoffmann (2023)

state – it tends to be less prone to ending up in the local minima of the objective function. This statement has been studied from theoretical (Jayaram & Klawonn, 2013) and empirical perspective (Klawonn et al., 2015), providing arguments in favour of this statement. However, as Klawonn and Hoffmann (2023) note, a general proof has not been yet proposed. Nevertheless, this is clearly an advantage of fuzzy clustering over hard k -means, where special adjustments to the mechanism of initialization of the algorithm are needed, for example in a form of a popular k -means++ initialization technique proposed by Arthur and Vassilvitskii (2007).

Finally, the fuzzy approach provides additional value in the wider range of cluster validation techniques. The above discussion on the technical advantage of fuzzy clustering falls into the category of studies that, in order to compare various models, investigate the partitions obtained from these models and compare them against reference groupings (provided by experts) that serve as ground-truth labels (Gagolewski, 2022). However, in real-life scenarios, one typically does not possess the reference knowledge about the objects being partitioned, and needs to choose a *valid* partition of the data from a multitude of partitions obtained from various models (J. C. Bezdek et al., 1981). This includes not only a decision on how many clusters there are in the data – the value of the c hyperparameter – but also, which of the fuzzy c -partitions for a given c presents as the *valid* one. The difficult nature of this task is also stressed by Hastie et al. (2009), who note that it is a challenging task to determine whether conclusions drawn from the results of most unsupervised learning methods are valid.

In this context, fuzzy clustering enables a rich family of cluster validity measures, that takes into account the fuzzy character of the degrees of membership. In a recent extensive literature review, H.-Y. Wang et al. (2022) used a name *fuzzy clustering validity indices*, and grouped the methods into two categories - those based purely on memberships, and those based on both memberships and a geometric structure of the dataset. Regarding classical propositions, both the *partition coefficient* (J. C. Bezdek, 1974), and *partition entropy* (Bezdek†, 1973), fall into the first category, while a classical index from the second category is the *clustering validity function* proposed by Xie and Beni (1991). The developments in the field include novel propositions in both categories, including (Silva et al., 2015; D. Zhang et al., 2014) that are based on memberships only, and (Y. Liu et al., 2021; Ouchicha et al., 2020; Zhu et al., 2019) that take into account the geometrical structure of the dataset.

2.1.2 Fuzzy c-Means

Having stated the above, the classical unsupervised fuzzy clustering model, a correspondent of k -means algorithm, is a Fuzzy c-Means (FCM) model, with the objective function proposed in (J. C. Bezdek, 1981):

$$Q_{\text{FCM}}(U, V) = \sum_{j=1}^N \sum_{k=1}^c u_k^m(\mathbf{x}_j) \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2, \quad (2.7)$$

where $m \in [1, \infty)$, called originally by (J. C. Bezdek, 1981) as “weighting exponent”, is also referred to as a *fuzzifier* (refer to (Klawonn & Höppner, 2003) for in-depth study on the role of m in fuzzy clustering), and $\mathbf{v}_k$ is a cluster’s prototype (with various observations contributing to it do a different degree, as opposed to k -means algorithm). It is worth noting that FCM as formulated in (J. C. Bezdek, 1981) is a generalization of earlier work of (Dunn, 1973), who focused on a special case $m = 2$ only.

It shall be stressed that the notation used in this thesis differs from other propositions. Two most important aspects of notation include the presentation of the memberships matrix U , and the indices running through the observations, and the clusters, that appear very frequently in various equations. Firstly, J. C. Bezdek (1981) considered U to be of $c \times N$ dimensions, that is - each k th row corresponded to the fuzzy set denoting k th cluster. In this thesis, U is presented in the transposed form with respect to the Bezdek’s presentation. These are the columns of matrix $U_{N \times c}$ that define the clusters. Secondly, J. C. Bezdek (1981) used

$$k = 1, \dots, n \quad \text{to index observations,} \quad (2.8)$$

$$i = 1, \dots, c \quad \text{to index clusters,} \quad (2.9)$$

whereas in this thesis, we use

$$j = 1, \dots, N \quad \text{to index observations,} \quad (2.10)$$

$$k = 1, \dots, c \quad \text{to index clusters.} \quad (2.11)$$

There also exist other propositions of nomenclature, for example Ferraro and Giordani (2020) use the name *Fuzzy k-Means*, rather than *Fuzzy c-Means*, and use the $i = 1, \dots, n$ to index observations, and $g = 1, \dots, k$ to index clusters.

For the reasons of coherent presentation, all the nomenclature from the references has been adapted to a unified presentation. We will also further uses interchangeably

$$u_k(\mathbf{x}_j) = u_{jk} \quad (2.12)$$

and denote in general the distance between observation $\mathbf{x}_j$ and prototype $\mathbf{v}_k$ as

$$d_{jk} = d(\mathbf{x}_j, \mathbf{v}_k) \quad (2.13)$$

for simpler notation.

The minimization problem – corresponding to the objective function Q_{FCM} – to solve is

$$\min_{U, V} Q_{\text{FCM}}(U, V; X) \quad (2.14a)$$

$$\text{s.t.} \quad \sum_{k=1}^c u_{jk} = 1 \quad \forall j = 1, \dots, N, \quad (2.14b)$$

$$0 < \sum_{j=1}^N u_{jk} < N \quad \forall k = 1, \dots, c, \quad (2.14c)$$

$$u_{jk} \in [0, 1], \quad (2.14d)$$

where Criteria Eq. (2.14b), Eq. (2.14c), Eq. (2.14d) correspond to the criteria provided in the definition of a fuzzy c -partition space in Eq. (2.3).

Problem (2.14) has a global minimum (U^*, V^*) , because each of the summands of (2.7) is greater than or equal to zero, but analytically finding it is intractable, since Q_{FCM} is a non-convex function of its arguments (U, V) treated jointly. Different approaches to approximately solving this problem were proposed in the literature, just as for the hard k -means Problem (2.1), including ideas to apply genetic algorithms (Jebari et al., 2019; Jiang & Sun, 2014; Salcedo-Sanz et al., 2014) and particle swarm optimization (S. Yu et al., 2012; J. Zhang & Shen, 2014), but it is the Alternating Optimization algorithm that has been remaining vastly popular until nowadays (J. C. Bezdek, 2025). **In this thesis, the Fuzzy c-Means model will refer to the specific combination of the objective function, and the Alternating Optimization algorithm used to solve it.**

The AO algorithm, as described in (J. C. Bezdek, 1981), relies on considering first-order necessary conditions for the global minimizer (U^*, V^*) of the Problem (2.14). The respective optimization of partial functions, i.e. $Q_{\text{FCM}}(U; V, X)$, and $Q_{\text{FCM}}(V; U, X)$, can be performed using the standard convex optimization techniques, since both functions are convex functions of their arguments. As a result, one obtains minimizers of these convex

subproblems, i.e., the optimal memberships $\hat{U}$ minimizing $Q_{\text{FCM}}(U; V, X)$, and the optimal prototypes $\hat{V}$ minimizing $Q_{\text{FCM}}(V; U, X)$. The algorithm requires an initialization - by either an arbitrary memberships matrix, or arbitrary prototypes matrix. In this thesis, an initialization by memberships matrix is assumed, although different approaches exist - as Pal and Bezdek (1995) state, “the literature contains both specifications [either initialize by arbitrary memberships, or arbitrary prototypes]; the convergence theory is the same in either case”.

Solving the subproblems is achieved as follows. For the prototypes, finding $\hat{V}$ is in essence independently finding optimal k th cluster’s prototype, as the columns of V are independent:

$$\min_{\mathbf{v}_k \in \mathcal{R}^p} \sum_{j=1}^N u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2, \quad (2.15)$$

and the respective solution is

$$\hat{\mathbf{v}}_k = \frac{\sum_{j=1}^N (u_{jk}^m \cdot \mathbf{x}_j)}{\sum_{j=1}^N u_{jk}^m}. \quad (2.16)$$

The second subproblem is posed as

$$\min_{U \in M_{fc}} \sum_{j=1}^N \sum_{k=1}^c u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2, \quad (2.17)$$

The precise description of the solution of this problem can be found in (J. C. Bezdek, 1981; J. C. Bezdek et al., 1984). The first step is to relax constraint Eq. (2.14c), and instead consider

$$0 \leq \sum_{j=1}^N u_{jk} \leq N \quad \forall k = 1, \dots, c. \quad (2.18)$$

This relaxation leads to a so-called *degenerate* fuzzy c -partition space, denoted by M_{fco} . Operating in M_{fco} allows to treat rows of U as independent², which changes the optimization Problem (2.17) into

$$\min_{U \in M_{fco}} \sum_{j=1}^N \sum_{k=1}^c u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2, \quad (2.19)$$

²In (J. C. Bezdek, 1981), columns of U instead of rows of U are mentioned due to a different formulation of the memberships matrix. J. C. Bezdek (1981) considers U^T , a transposition of the matrix $U_{N \times c}$ form considered in this thesis.

Algorithm 2 Alternating Optimization Algorithm for Fuzzy c-Means (J. C. Bezdek, 1981)

Define function $W : M_{fc} \rightarrow \mathcal{R}^{c \times p}$ s.t. $W(U) = [\hat{\mathbf{v}}_k(U)]$, where $\hat{\mathbf{v}}_k$ is calculated according to Eq. (2.16)

Define function $P : \mathcal{R}^{c \times p} \rightarrow M_{fc}$ s.t. $P(V) = [\hat{u}_{jk}(V)]$, where $\hat{u}_{jk}$ is calculated according to Eq. (2.22)

Set convergence threshold $\epsilon > 0$

1: Initialize the algorithm with $U \in M_{fc}$

2: **repeat**

3: $U_{\text{prev}} \leftarrow U$

4: $V \leftarrow W(U_{\text{prev}})$

5: $U \leftarrow P(V)$

6: **until** $\|U - U_{\text{prev}}\| < \epsilon$

7: **Output:** U, V

that due to the independence of the rows of U , can be transformed to

$$\sum_{j=1}^N \min_{\mathbf{u}_k \in \mathcal{R}^c} \sum_{k=1}^c u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2 \quad (2.20a)$$

$$\text{s.t.} \quad \sum_{k=1}^c u_{jk} = 1 \quad \forall j = 1, \dots, N, \quad (2.20b)$$

$$u_{jk} \geq 0. \quad (2.20c)$$

The next step is to apply the Lagrange multipliers technique to directly account for the constraint Eq. (2.14b). As shown by J. C. Bezdek (1981), this will lead to arriving at the formula of the optimal membership degree of j th observsation to k th cluster

$$\hat{u}_{jk} = \frac{1}{\sum_{g=1}^c \left(\frac{d_{jk}}{d_{jg}} \right)^{\frac{2}{m-1}}}, \quad (2.21)$$

where for simpler notation we use the notation d_{jk} introduced in Eq. (2.13) to refer to the Euclidean distance. In case of $m = 2$, assumed further in this thesis, the formula for the optimal membership simplifies to

$$\hat{u}_{jk} = \frac{1}{\sum_{g=1}^c d_{jk}^2 / d_{jg}^2}. \quad (2.22)$$

Having established optimal formulae for $\hat{U} = [\hat{u}_{jk}]$ and $\hat{V} = [\hat{\mathbf{v}}_k]$, a similar AO algorithm is applied to iterate between them as in the case of k -means. The pseudocode for relevant Alternating Optimization mechanism is presented in Algorithm 2. The properties

of this algorithm and its convergence properties were subject of classical studies. Firstly, J. C. Bezdek (1981) himself provided convergence analysis that turned out to be partially incorrect. The ultimate results from R. J. Hathaway and Bezdek (1988), R. J. Hathaway et al. (1987), and R. Hathaway and Bezdek (1995) established that the AO algorithm as described above converges either to minimum (local or global), or a saddle point, of the original Problem (2.14).

The key aspects of the optimization procedure described in detail by J. C. Bezdek (1981) are as follows. Firstly, the “trick” to relax the respective constraint defining M_{fc} and consider Eq. (2.18) instead, allowed to optimize on the level of individual membership degrees, but lead to a solution that lies in M_{fc} anyway - since the image of $\hat{u}_{jk}$ is $(0, 1)$, and $\hat{U} = [\hat{u}_{jk}]$. Secondly, one must also note that out of the three constraints defining M_{fc} , the constraint Eq. (2.14d) requiring that $u_{jk} \in [0, 1] \forall_{j,k}$ has not been directly imposed in any step of the optimization procedure. Rather, it is the final formula for optimal memberships that assures $\hat{u}_{jk} \in (0, 1)$, and therefore, assures that the decision memberships $\tilde{U} \in M_{fc}$. This remark will be important in further discussion on the semantics of membership degrees in semi-supervised fuzzy clustering undertaken in Section 3.3.2.

2.1.3 Possibilistic c-Means

Fuzzy c-Means model described above, understood as a framework that resulted in posing the optimisation problem, and a specific algorithm to solve it, is a building block of all of the advancements in the field of fuzzy clustering. One of the first developments considered the constraint from (2.14b), that we will further refer to as to *sum-to-one* constraint. This constraint has been frequently referred to as to the *probabilistic constraint*, although it has nothing to do with the additive law of probabilities (Ruspini et al., 2019) that it might resemble - it is a generalization of the corresponding proposition from the fuzzy set theory that requires that for any given fuzzy set A , and its complement $\neg A$, one has

$$\mu_A(\omega) + \mu_{\neg A}(\omega) = 1. \quad (2.23)$$

A classical model, the second that we study in this thesis, is Possibilistic c-Means model proposed by Krishnapuram and Keller (1993), who modified Fuzzy c-Means by dropping the *sum-to-one* constraint, and accounted for it by proposing a following objective function

$$Q_{\text{PCM}}(U, V) = \sum_{j=1}^N \sum_{k=1}^c u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2 + \sum_{j=1}^N \sum_{k=1}^c \gamma_k (1 - u_{jk})^m, \quad (2.24)$$

and the associated problem

$$\min_{U,V} Q_{\text{PCM}}(U, V; X) \quad (2.25a)$$

$$\text{s.t.} \quad 0 < \sum_{j=1}^N u_{jk} \leq N \quad \forall k = 1, \dots, c, \quad (2.25b)$$

$$u_{jk} \in [0, 1], \quad (2.25c)$$

Krishnapuram and Keller (1993) interpret the addition of the term $(1 - u_{jk})^m$ in Q_{PCM} as follows: “the first term demands that the distances from the feature vectors to the prototypes be as low as possible, whereas the second term forces the u_{jk} to be as large as possible, thus avoiding the trivial solution”.

The solution of the optimization problem mirrors the solution of Fuzzy c-Means presented in the preceding Section. The optimal memberships in PCM present as follows:

$$\hat{u}_{jk} = \frac{1}{1 + \left(\frac{d_{jk}^2}{\gamma_k} \right)^{\frac{1}{m-1}}}, \quad (2.26)$$

which in case of $m = 2$ simplifies to

$$\hat{u}_{jk} = \frac{1}{1 + \frac{d_{jk}^2}{\gamma_k}} = \frac{\gamma_k}{\gamma_k + d_{jk}^2}, \quad (2.27)$$

and the optimal prototypes are

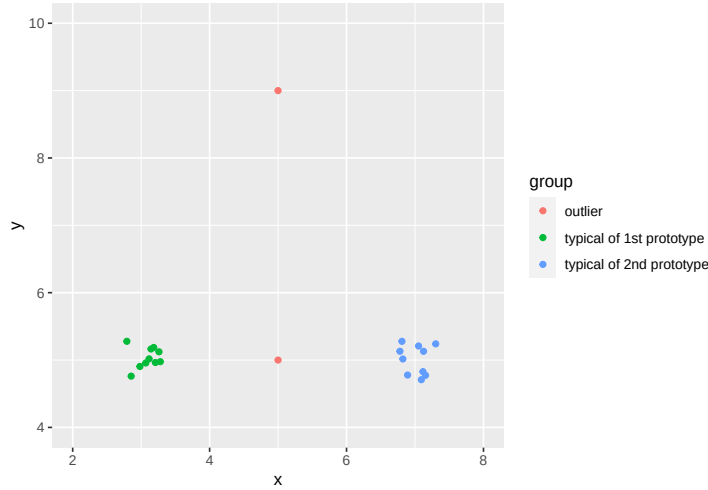
$$\hat{\mathbf{v}}_k = \frac{\sum_{j=1}^N (u_{jk}^m \cdot \mathbf{x}_j)}{\sum_{j=1}^N u_{jk}^m}. \quad (2.28)$$

The role of the cluster-specific hyperparameter γ_k is that it determines the distance, at which the membership of a given observation to this clusters becomes 0.5 (Krishnapuram & Keller, 1993), since

$$\hat{u}_{jk} = \frac{\gamma_k}{\gamma_k + d_{jk}^2} \stackrel{d_{jk}^2 := \gamma_k}{=} \frac{\gamma_k}{\gamma_k + \gamma_k} = 0.5. \quad (2.29)$$

Krishnapuram and Keller (1993) stress that each γ_k should correspond to the size and shape of the respective cluster ψ_k , and to achieve that, they propose to calculate it according to the following formula

Figure 2.1: An example of an unsupervised dataset with atypical observations.



Source: own work based on Fig. 1c from (Krishnapuram & Keller, 1993). The colours denoting the *groups*, provided by Krishnapuram and Keller (1993) as experts' reference partition, were added to ease the interpretation; the reference partition is provided to interpret the different results of unsupervised fuzzy clustering models and does not constitute partial supervision.

$$\gamma_k \sim \frac{\sum_{j=1}^N u_{jk}^m d_{jk}^2}{\sum_{j=1}^N u_{jk}^m}, \quad (2.30)$$

where respective data can be obtained for example by initial run of other appropriate model, such as Fuzzy c-Means.

The motivation to drop the *sum-to-one* constraint was of both theoretical, and practical nature. It was pointed by Krishnapuram and Keller (1993) that the optimal memberships in the Fuzzy c-Means problem are *relative*, in a sense that a membership of j th observation to k th cluster depends on the entire memberships' distribution of this specific observation, and that “it may not always correspond to the intuitive concept of degree of belonging or compatibility” (Krishnapuram & Keller, 1993). This constitutes the theoretical aspect of the Possibilistic c-Means model, as the discussion on the semantics of fuzzy logic, and the notion of the membership degree, is a complex debate. Dubois and Prade (1997) in their work on the semantics of fuzzy sets state that the majority of critical remarks made by fuzzy sets oponents found in the literature concern how membership grades should be interpreted. The other classical works on the semantics of fuzzy sets include (Ruspini, 1991, 1990).

The practical motivation to design a different model than Fuzzy c-Means is depicted in Fig. 2.1, which presents a simple example of two clusters, centered around $\mathbf{v}_1 = [3, 5]^T$, and $\mathbf{v}_2 = [7, 5]^T$, with 10 observations typical of the first cluster (coloured in green), and 10 observations typical of the second cluster (coloured in blue). The points $\mathbf{x}_{21} = [5, 5]^T$

and $\mathbf{x}_{22} = [5, 9]^T$ are considered as noise, each point equidistant from each of the clusters' prototypes - they are coloured in orange. This example is reproduced based on the discussion in Krishnapuram and Keller (1993), who originally provided a similar example in Fig. 1c of their paper; the colours are added to ease the interpretation of the experts' reference partition provided by Krishnapuram and Keller (1993) and do not constitute any type of partial supervision.

The decision memberships for these two atypical points obtained by running a Fuzzy c-Means model present as follows

$$\begin{array}{c} k = 1 \quad k = 2 \\ (j = 21) \left[\begin{array}{cc} 0.48 & 0.52 \end{array} \right], \\ (j = 22) \left[\begin{array}{cc} 0.48 & 0.52 \end{array} \right], \end{array} \quad (2.31)$$

and these estimated by the Possibilistic c-Means present as follows

$$\begin{array}{c} k = 1 \quad k = 2 \\ (j = 21) \left[\begin{array}{cc} 0.21 & 0.20 \end{array} \right], \\ (j = 22) \left[\begin{array}{cc} 0.05 & 0.05 \end{array} \right]. \end{array} \quad (2.32)$$

These decision memberships from the two models provide arguments for Krishnapuram and Keller (1993) who argue that, as opposed to FCM, Possibilistic c-Means is capable of handling such situations with outliers, atypical observations. Even more - PCM is capable of distinguishing between “moderately atypical member [observation $\mathbf{x}_{22}$] and an extremely atypical member [observation $\mathbf{x}_{21}$]”, whereas Fuzzy c-Means is not. Indeed, regarding the estimates obtained from Possibilistic c-Means in our example, the distribution of estimated membership degrees for the more atypical outlier $\mathbf{x}_{22} = [5, 9]^T$ is equal to $[0.05, 0.05]^T$, whereas the distribution for the less atypical outlier $\mathbf{x}_{21} = [5, 5]^T$ is equal to $[0.21, 0.20]^T$. On the contrary, the estimates obtained from Fuzzy c-Means are highly similar for both observations, approximately equal to 0.5 for each membership of each of the outliers to both clusters.

In defense of the results obtained by Fuzzy c-Means, one must stress that this model has strong roots in the fuzzy logic. To enable interpretation of the above results, let us define the core of a fuzzy set in context of fuzzy clustering based on two definitions from (Ruspini et al., 2019; Zadeh, 1965).

Definition 5 (Core of a fuzzy set). *Let the core of a fuzzy set A be*

$$C(A) = \{\omega \mid \omega \in \Omega \text{ and } \mu_A(\omega) = 1\}. \quad (2.33)$$

Then, treating columns of the membership matrix U as fuzzy sets, we can refer to the interpretation from Ruspini et al. (2019), who discuss the *sum-to-one* criterion and its consequences: “representation of a point $\mathbf{x}$ belonging to a fuzzy set A to the degree 0.5 and to the fuzzy set B to the same degree is intended to indicate that the point lies midway between the cores of A and B , rather than to make a statistical assumption about a hidden aleatory process having equal chances of assigning $\mathbf{x}$ to one of two distinct classes A and B ”. Indeed, the distribution of the decision memberships for both $\mathbf{x}_{21}$ and $\mathbf{x}_{22}$ observations in our example aligns with the interpretation that they lie midway between the cores of $U_{:1}$, the first fuzzy set, and $U_{:2}$, the second fuzzy set, where $U_{:k}$ denotes the k th column of the memberships matrix U .

In this sense, Possibilistic c-Means deviates from the fuzzy sets theory to enable an understanding of memberships as *typicalities*. For this reason, in modern presentations – refer e.g. to Antoine et al. (2022) – a symbol t_{jk} is used instead of u_{jk} to stress this different understanding of the degree of belonging. We will further adhere to this convention, rewriting Eq. (2.27) as

$$\hat{t}_{jk} = \frac{1}{1 + \frac{d_{jk}^2}{\gamma_k}} = \frac{\gamma_k}{\gamma_k + d_{jk}^2}. \quad (2.34)$$

The term *possibilistic* in the name of the PCM model proposed by Krishnapuram and Keller (1993) is meant to stress this understanding of degrees of belonging as typicalities, since a conception that one should be able to distinguish between “«equal evidence» and «ignorance»” when discussing degrees of membership is discussed in the field of possibility theory, presented e.g. in Dubois and Prade (1988). However, the connections of PCM to the possibility theory are much more loose than the corresponding connections of FCM to the fuzzy set theory, which is stressed by Ruspini et al. (2019), who state that “nothing about PCM is possibilistic in the true sense of possibility theory”, as presented in Dubois and Prade (2001) and Zadeh (1999). Further, they point that Krishnapuram and Keller (1993) chose the name *possibilistic* merely to differentiate their approach from the *sum-to-one* partitions produced by FCM, and that the similarity between the optimization problems posed by the two models remains very high, since the proof of Possibilistic c-Means theorem closely follows the structure of the corresponding Fuzzy c-Means theorem’s proof (Krishnapuram & Keller, 1993).

In this thesis, in light of the approach presented by Ruspini et al. (2019), we will thus treat **Possibilistic c-Means and Fuzzy c-Means as two classical foundational models falling into the same class of *fuzzy clustering* models**, which is a subclass of general distance-based, partitional class of clustering models. We stress that since

an opposing view can also be found in the literature, e.g. Ferraro (2024) distinguishes between fuzzy, possibilistic, and rough clustering methods.

Coincident prototypes in Possibilistic c-Means

With respect to the core PCM model, and its use in this thesis, a problem called the coincidence of the prototypes must be presented. To show its occurrence, let us fit unsupervised Fuzzy c-Means and Possibilistic c-Means models, denoted correspondingly by M_{FCM} and M_{PCM} , to the X_{train} matrix created out of `iris` dataset that was described in Section 1.4. When estimating the M_{PCM} model, we will calculate the values of the $(\gamma_1, \gamma_2, \gamma_3)$ 3-tuple according to the formula Eq. (2.30), and in addition, we set $U^{(0)} = \tilde{U}_{\text{FCM}}$, i.e., initiate the Possibilistic c-Means model with the decision memberships of the Fuzzy c-Means model.

The decision of the Fuzzy c-Means model presents as follows:

$$\tilde{V}_{\text{FCM}} = \begin{pmatrix} 6.856 & 3.077 \\ 5.822 & 2.750 \\ 4.978 & 3.340 \end{pmatrix}, \quad (2.35)$$

and the decision of the Possibilistic c-Means model presents as follows:

$$\tilde{V}_{\text{PCM}} = \begin{pmatrix} 5.933 & 2.909 \\ 5.939 & 2.904 \\ 5.939 & 2.905 \end{pmatrix}. \quad (2.36)$$

The issue with $\tilde{V}_{\text{PCM}}$ is called the problem of coincident clusters (Y. Gao et al., 2024). This problem is due to the very nature of the classical Possibilistic c-Means model. Timm et al. (2004) state that examining the objective function presented in Eq. (2.24), one must note that it is ultimately minimized when all prototypes are identical, and hence, PCM tends to estimate identical clusters in certain circumstances. The usual remedy is to initialize the Possibilistic c-Means with appropriate values of (γ_k) , and if that does not help - with the appropriate values of $U^{(0)}$ (e.g., obtained by initial run of Fuzzy c-Means), but in the case of `iris` data, these steps did not help in avoiding the problem of coincident clusters, as can be seen above.

Over the years, propositions to improve the core Possibilistic c-Means model to counteract this common issue have been proposed. An idea to apply mutual repulsion of the clusters was proposed and studied in (Timm & Kruse, 2002; Timm et al., 2004, 2001); a different approach was recently proposed by Y. Gao et al. (2024) who resolve the coincident cluster's issue by appropriate adaptation of the Adaptive Nearest Neighbours method into the Possibilistic c-Means clustering framework. Nevertheless, in this thesis

we limit to the classical models, aiming to explain the impact of partial supervision in their semi-supervised counterparts, and do not analyze in detail any derivative models, such as the above propositions due to the lack of space. For this reason, Semi-Supervised Possibilistic c-Means – the semi-supervised counterpart of PCM – will be analyzed and explained theoretically in Chapter 3, but will not be analyzed empirically in Chapter 5; refer to the respective Section 5.1.2 for more detailed justification.

2.1.4 Related work in unsupervised fuzzy clustering

This section is devoted to the presentation of the selected developments in the field of fuzzy clustering, aimed at showing how critical a sound understanding of the classical Fuzzy c-Means and Possibilistic c-Means models is for majority of the related work. There does not exist a single, comprehensive survey on the topic of fuzzy clustering, due to the reason that was highlighted by three major contributors to the field of fuzzy clustering: E.H. Ruspini, J. C. Bezdek, and J. M. Keller, who stated in their recent overview of the historical development of fuzzy clustering that such a comprehensive survey “would require an article the size of an encyclopedia and an army of well-informed authors” (Ruspini et al., 2019). Nevertheless, it is worth to distinguish high-level overview publications. Firstly, Ruspini et al. (2019) themselves describe in the detail the key ideas, developments, and the differences between various major approaches in the field of fuzzy clustering proposed over the years. A comprehensive, more recent overview of the history and applications of Fuzzy c-Means was provided by Ferraro (2024), but it excludes the Possibilistic c-Means from the scope of models considered. Finally, an extensive survey was provided by Nayak et al. (2015), but it is focused only on the developments in a specific time range varying from 2000 to 2014.

For clarity, it shall be noted that the idea of *soft* clustering, as opposed to *hard* clustering (of which k -means is a typical representant), goes beyond the class of fuzzy clustering models, which is the topic of this thesis. In particular, the branches of probabilistic clustering, where the classical approach involves Gaussian Mixture Models (McLachlan & Peel, 2000), or evidential clustering based on the use of mass functions coming from the Dempster-Shafer theory (Denceux & Kanjanatarakul, 2016) are also considered as *soft* in a sense they result in non-binary assignments of observations to clusters. Some of the propositions in the literature on evidential clustering are even direct attempts to represent the core fuzzy models within the light of the new theory, refer to (Albert et al., 2025; Masson & Denceux, 2008). In addition to fuzzy, probabilistic and evidential models, Ferraro and Giordani (2020) list also rough-set inspired clustering algorithms as soft clustering models. These include models proposed by Lingras and West (2004) and Peters

(2006) that are based on the rough set theory introduced and studied by Pawlak (1982) and Pawlak (1991).

Regarding the field of fuzzy clustering itself, Klawonn and Hoffmann (2023) point that one can clearly show two major directions of research by rewriting the objective function from (2.7) using a generic notation as

$$Q_{\text{FCM}}(U, V) = \sum_{j=1}^N \sum_{k=1}^c u_{jk}^m \cdot d_{jk}^2, \quad (2.37)$$

where – let us recall – $u_{jk} = u_k(\mathbf{x}_j)$ is a membership degree, and $d_{jk} = d(\mathbf{x}_j, \mathbf{v}_k)$ denotes a distance (measured as the Euclidean norm in the case of core Fuzzy c-Means model). From the above equation, it becomes clearly visible that the modifications and research pertain mostly to two aspects:

1. “how the memberships u_{jk} enter the objective function” (Klawonn & Hoffmann, 2023),
2. how the distance d_{jk} between objects and prototypes is measured.

Regarding the developments in the formulation of the objective function with respect to the membership value, it must be stressed that Possibilistic c-Means itself is an example of such a modification, as it added a new penalty term $\gamma_k \cdot (1 - u_{jk})^m$ in the respective objective function. The study on the advantages and disadvantages of both FCM and PCM lead to the formulation of hybrid models, starting with Fuzzy-Possibilistic c-Means (FPCM) model proposed in (Pal et al., 1997), which was designed to “possess the good features of both models [FCM and PCM], and while eliminating some of their bad features” (Pal et al., 1997). The idea of FPCM was to estimate both memberships matrix and typicalities matrix simultaneously. However, a specific design choice in the Fuzzy-Possibilistic c-Means, namely the requirement that both memberships and typicalities of all objects to the given cluster must sum up to 1, turned out to be problematic, and led Pal et al. (2005) to the proposition of the adjusted hybrid proposition, called Possibilistic-Fuzzy c-Means (PFCM) model to distinguish it from FPCM, that alleviated this constraint. All of the four models: FCM, PCM, FPCM, and PFCM, constitute fundamental fuzzy clustering models, representing different approaches to the problem of estimating memberships and/or typicalities.

A different idea was to remove the fuzzifier m from equation. Instead, a regularization term based on the concept of entropy is considered.

Definition 6 (Gray (2011)). *Let Y be a discrete random variable with probability mass*

function $p(Y)$. The entropy of Y is

$$H(Y) = - \sum_y p(y) \cdot \log p(y). \quad (2.38)$$

Based on the concept of *Maximum Entropy Inference* (MEI), where maximization of entropy is proposed, Rui-Ping Li and Mukaidono (1995) proposed a following objective function, adapting MEI to the realm of fuzzy clustering

$$Q_{\text{ENT}}(U, V) = \sum_{j=1}^N \sum_{k=1}^c u_{jk} d_{jk}^2 + \lambda \sum_{j=1}^N \sum_{k=1}^c (u_{jk} \cdot \log(u_{jk})), \quad (2.39)$$

where the second, entropic component of the objective function aimed to prevent memberships to converge close to extreme values of 0 or 1. Further research on this topic includes (Kamman et al., 2012; Liangqun et al., 2006; Sahbi & Boujemaa, 2005; Tran & Wagner, 2000; Yang et al., 2006; Zarinbal et al., 2014).

Regarding the approach to measure the distance between the observations and prototypes, first propositions in the literature aimed at guiding the shapes of the clusters towards being able to reflect specific shapes, or patterns, present in the data. Especially popular and practical solution was provided by Gustafson and Kessel (1978), who introduced the idea of adaptive distance through the use of Mahalanobis distance and cluster-specific covariance matrices. A different idea was introduced by J. C. Bezdek et al. (1981) as the Fuzzy c-Varieties approach, which replaced the notion of a cluster's prototype with the notion of linear varieties, including lines, or planes. Yet another example of a focus on specific desired shape of clusters is a branch of finding "shell-like" clusters initiated by Krishnapuram et al. (1995).

A different stream is to weigh features' contribution, where an assumption is made that each element of the features vector $\mathcal{R}^p \ni \mathbf{x}_j = [x_{j1}, \dots, x_{jp}]^T$ can have a different significance in the process of clustering structure formation for the input space X . A thorough review of this research direction that focuses on finding a relevant set of features' weights to be used in clustering process is provided by (Golzari Oskouei et al., 2024), who distinguished as many as six groups of related algorithms: feature reduction-based, entropy-based, variance-based, membership-based, optimization-based, and meta-heuristic-based.

Yet another idea related to calculating distances between observations and clusters' prototypes is to use the so-called kernel trick that refers to the use of a kernel function $k(\mathbf{x}_j, \mathbf{v}_k)$ that computes the inner product of its arguments in a high-dimensional feature space, without requiring the explicit mapping (Hofmann et al., 2008). Formally, given a

nonlinear feature map $\phi : \mathcal{R}^p \rightarrow \mathcal{H}$ the kernel is defined as

$$k(\mathbf{x}_j, \mathbf{v}_k) = \langle \phi(\mathbf{x}_j), \phi(\mathbf{v}_k) \rangle_{\mathcal{H}}. \quad (2.40)$$

This formulation allows machine learning algorithms such as support vector machines to exploit richer, possibly infinite-dimensional representations while performing computations solely through $k(\cdot, \cdot)$, thus avoiding the explicit construction of $\phi(\mathbf{x})$. One of the first ideas to use the kernel function as the distance function d in the field of fuzzy clustering was proposed in (Zhong-dong Wu et al., 2003), and further research includes, inter alia, (Hsin-Chien Huang et al., 2012; Simões & T. De Carvalho, 2023; Tang et al., 2023).

To conclude, unsupervised fuzzy clustering is a rich field of research, where the classical models: Fuzzy c-Means and Possibilistic c-Means are still used directly or indirectly, as a building block on which novel propositions are built. A thorough understanding of these core models is thus necessary, and the same will apply to semi-supervised fuzzy clustering that extend the underlying, unsupervised models.

2.2 Semi-supervised fuzzy clustering

Semi-supervised fuzzy clustering (SSFC) belongs to the same class of fuzzy clustering models as unsupervised fuzzy clustering, i.e., to the distance-based, partitional class of clustering models that use the soft membership assignment mechanism inspired by the theory of fuzzy sets. The only difference, but with far-reaching consequences, is that SSFC incorporates partial supervision to alter the outcomes of modelling – the decision memberships $\tilde{U}$ and $\tilde{V}$ – as compared with the unsupervised approach.

The most important consequence of introducing partial supervision is the ability to treat the SSFC model as a classifier, and derive a decision function ρ based on its outcomes; refer to Eq. (1.9). Nevertheless, as postulated in this thesis, it is not the optimization of the classification error that is the direct goal of SSFC, but the optimization of partitioning. However, it is the use of SSFC model as classifier that is frequently of ultimate interest of the researchers and practitioners, and semi-supervised fuzzy clustering shares concerns and research goals related to the quality of classification. We therefore provide a more detailed discussion on the connections of SSFC with the field called Semi-Supervised Learning (SSL).

Since semi-supervised fuzzy clustering shares the same problem formulation, its solution, notation, and conceptualization with unsupervised fuzzy clustering introduced in the preceding Section, we will not repeat the already presented theoretical preliminaries.

Instead, in this Section we will focus on presenting the key aspect of SSFC that lies in a specific mechanism of introducing partial supervision, and regulating its impact, within the fuzzy clustering paradigm. A first, canonical proposition of such mechanism was introduced by Pedrycz and Waletzky (1997) and became widespread. In the author of this thesis’ paper (Kmita et al., 2024), a generalization of this canonical idea was derived and called *the additive combination technique*, which enabled to present the entire range of SSFC models as a cohesive class building on the initial proposition by Pedrycz and Waletzky (1997). We introduce it in this Section, and significantly extend the range of the literature review on this topic.

Although the additive combination technique became widely reused, it lacks proper explainability, which is the problem this thesis is stating and aiming to resolve. The introduction of the respective explainability framework and actual explanations is discussed in Chapter 3 of this thesis; in this Section, we provide a detailed literature review focused on the specific mechanism of impact of partial supervision in SSFC, which supports the research gap identified and stated in Section 1.3.

2.2.1 Fuzzy clustering and partial supervision

The consequence of sole interest in semi-supervised fuzzy clustering models as classifiers leads to ignoring the strong connections between SSFC and unsupervised fuzzy clustering. It is particularly visible when considering the approach to the comparison of various semi-supervised fuzzy clustering models’ performance on the benchmark datasets - for example, in their meta-review of the classical SSFC models, Lai and Garibaldi (2011) consciously decide to use “classification rate[s]” in place of “cluster validity indices” as external quality criterion $\text{Crit}(\cdot)$, justifying that it is a prevalent approach in the literature.

For the above reasons, semi-supervised fuzzy clustering is also presented in the context of the Semi-Supervised Learning (SSL), a broad research field working with problems of partially supervised data. In the introduction of the classical reference book, Chapelle et al. (2006) state that “Semi-supervised learning is halfway between supervised and unsupervised learning”. Such associations of SSFC with the field of SSL arise naturally due to the very name of this class of models, which includes the phrase “semi-supervised”.

Nevertheless, they are misleading, since the above view of Chapelle et al. (2006) is presented in the context of the characteristics of the partially supervised data, not as a consideration regarding a taxonomy of the models. Chapelle et al. (2006) clarifies that within SSL as understood in their book, an assumption is made that a set of objects Ω can be split into the union of two subsets $\Omega^U \sqcup \Omega^S$, where Ω^U contains unsupervised objects, and Ω^S contains objects for which certain prior knowledge – the supervision –

is available³. Correspondingly, numerical representations of the unsupervised objects are available in X^U , and numerical representations of the supervised objects are available in X^S . The cardinalities of the sets are denoted as $|\Omega| = N, |\Omega^U| = H, |\Omega^S| = M$. Even though the data is of mixed nature in SSL, the optimization problem is not. It is either the supervised problem, or the unsupervised one.

In the supervised problem, one starts from the supervised data X^S (and the corresponding supervision), and tries to incorporate the additional knowledge available in the unsupervised part X^U to improve the performance of the estimated model. In the unsupervised problem, one uses all of the available data anyway – i.e., the entire features matrix X – but tries to incorporate the supervision available for the objects in the corresponding set Ω^S to alter the results of the given unsupervised task – in the case of this thesis: clustering. It is the former – supervised – problem that is typically referred to as the Semi-Supervised Learning problem. Seeger (2006) states that “the semi-supervised learning problem belongs to the supervised category, since the goal is to minimize the classification error”, and suggests that a term *semi-supervised learning* would be more appropriate for the problems where it is the use of supervision about objects from Ω^S to elevate the results of the unsupervised problems. At the same time, Seeger (2006) claims that his statement is subject to debate, but his view is prevalent in the literature. The confirmation of this view is present in the recent, broad survey by Van Engelen and Hoos (2020), who state that in general, the vast majority of studies in semi-supervised learning concentrate on the classification task.

Nevertheless, the taxonomy of semi-supervised fuzzy clustering definitely shares one aspect with all of the SSL models: the form of supervision available. The two major forms in which it can come are (i) pairwise constraints, and (ii) labels. Pairwise constraints come in two forms: must-link constraints, and cannot-link constraints. Both forms include sets composed of pairs (ω_1, ω_2) that enlist all elements being under the supervision. In case of must-link constraint, the respective set includes observations that should be grouped in the same cluster, whereas in the case of cannot-link constraint, the respective set includes observation that should not be grouped in the same cluster. Labeling supervision assumes availability of a set of labels, representing the classes, organized in a set $\mathcal{Y} = \{y^1, \dots, y^c\}$; it is based on associating each object from Ω^S with one of the labels from $\mathcal{Y}$. In the context of SSFC, we will further treat $\mathcal{Y}$ as a c -tuple, i.e. introduce the order of its elements, whereas in general, this is not required in Semi-Supervised Learning.

The above two types of partial supervision constitute the major criterion in the taxonomy of semi-supervised fuzzy clustering models. In this thesis,

³The original nomenclature used in Chapelle et al. (2006) has been adjusted to be compliant with the nomenclature presented in this thesis.

we study the class of SSFC models that incorporate the labeling partial supervision. Semi-supervised fuzzy clustering with pairwise constraints were studied inter alia in (De Melo & De Carvalho, 2013; Frigui & Hwang, 2008; Grira et al., 2008; Mei et al., 2019; Ren et al., 2018), and a recent work by Z. Wang et al. (2022) proposed – as opposed to the above – to fuzzify the constraints themselves.

It shall be noted that in this thesis we consider the classical, binary supervision approach, where the number of classes defines the number of clusters to be considered, i.e., each cluster corresponds to a specific class. In the literature, modifications of these assumptions have been considered. These include adjusting the cluster’s assumption to allow multiple clusters correspond to a single class (Bouchachia & Pedrycz, 2006), allowing the supervision matrix F to include non-binary entries that are assumed to represent experts’ uncertainty about the validity of a given labeling (Antoine et al., 2022), or introducing the longitudinal aspects, where fuzzy clustering is applied in so-called batches, and a possibility of novel classes emerging with new batches is considered in semi-supervised fuzzy clustering context (Casalino, Castellano, et al., 2020; Casalino, Dominiak, et al., 2020). The motivation for considering the classical assumptions is that all the derivative propositions are reusing the core mechanism of classical SSFC paradigm, and the advancements proposed in this thesis will apply to any modification or adaptation of the classical models, accordingly.

Summary. The class of semi-supervised fuzzy clustering is a distance-based, partitional approach to the task of clustering, just as unsupervised fuzzy clustering. It does not belong to the problem of Semi-Supervised Learning, but through the incorporation of partial supervision, SSFC models can be used as classifiers, and therefore share specific concerns related to the quality of classification. In particular, the taxonomy of SSFC should take into account the type of partial supervision, and a mechanism to incorporate it into the fuzzy clustering problem. The two prevalent types are pairwise constraints, and labeling supervision, and the most prevalent mechanism is to extend the baseline unsupervised optimization problem.

2.2.2 Additive combination technique

As stated above, the canonical model for the class of semi-supervised fuzzy clustering models considered in this thesis is Semi-Supervised Fuzzy c-Means proposed first by Pedrycz and Waletzky (1997), who extended the baseline unsupervised Fuzzy c-Means model by proposing a following semi-supervised objective function J :

$$J_{P97} = \sum_{j=1}^N \sum_{k=1}^c u_{jk}^2 d_{jk}^2 + \alpha \sum_{j=1}^N \sum_{k=1}^c (u_{jk} - b_j \cdot f_{jk})^2 d_{jk}^2, \quad (2.41)$$

where $b_j = \sum_{k=1}^c f_{jk}$ is a binary indicator stating whether j th observation is supervised or not, and F encodes partial supervision, the prior knowledge about the assignment of labels from $\mathcal{Y}$ to the supervised objects from Ω^S . The idea of introducing labeling supervision in semi-supervised fuzzy clustering in a form of a matrix F of the same dimensions as matrix U is to guide the estimation of the memberships matrix in a way that rows corresponding to supervised elements from Ω^S would be closer to the respective rows from supervision matrix F than in the unsupervised case. It is the penalty term $(u_{jk} - b_j \cdot f_{jk})^2$ in Eq. (2.41) that is the specific implementation of this guidance mechanism.

In a binary case, the implementation of this penalty term requires F to be a matrix such that $f_{jk} = 1$ if and only if j th object ω_j is known to be supervised, and the corresponding label is k th element y^k from the c -tuple $\mathcal{Y}$. For example, recall the `iris` dataset introduced in Section 1.4 and the ordered 3-tuple denoting the classes of the objects $\mathcal{Y} = (\text{setosa}, \text{versicolor}, \text{virginica})$. Let us consider four objects; three from Ω^S , and one from Ω^U . Let ω_1 be known to be of `setosa` species, ω_2 be known to be of `versicolor` species, ω_3 be known to be of `virginica` species, and let ω_4 be unsupervised. The corresponding matrix $U_{4 \times 3}$ from a fuzzy 3-partition space, together with the binary supervision matrix F present as follows:

$$U = \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \\ u_{41} & u_{42} & u_{43} \end{pmatrix}, \quad F = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}. \quad (2.42)$$

It is the construction of J_{P97} described above, with the specific idea to introduce partial supervision in a form of the supervision matrix F , that Kmita et al. (2024) called the “additive combination” technique; it was presented schematically in the Section 1.1 in Eq. (1.4) as

$$J = Q(U, V; X, \Theta) + \alpha \cdot Q^S(U, V; X, F, \Theta).$$

The additive combination technique became a platform for numerous advancements in semi-supervised fuzzy clustering. To illustrate how modern, complex models reuse this concept, let us present the objective function proposed by Gan et al. (2023) to handle the problem of label safety⁴ (i.e., a situation where partial supervision may be erroneous for

⁴The original nomenclature used in Gan et al. (2023) has been adjusted to be compliant with the

some of the observations):

$$\begin{aligned}
 J_{\text{GAN}}(U, V, \mathbf{s}) = & \sum_{j=1}^N \sum_{k=1}^c u_{jk}^2 d_{jk}^2 \\
 & + \lambda_1 \sum_{j=1}^N \sum_{k=1}^c b_j s_j (u_{jk} - f_{jk})^2 d_{jk}^2 \\
 & + \lambda_2 \sum_{j=1}^N b_j \left\{ \sum_{t=1}^N \sum_{k=1}^c (1 - b_t) w_{tk} \left(s_j (u_{tk} - f_{jk})^2 + \left(\frac{2}{s_j + 1} - 1 \right)^2 (u_{tk} - u_{jk})^2 \right) \right\},
 \end{aligned} \tag{2.43}$$

where $\mathbf{s} = (s_1, \dots, s_N)$ is an additional variable to be optimized, denoting the label safety, and w_{tk} is an external weight, obtained by constructing a special graph prior to inducing the semi-supervised fuzzy clustering model, and λ_1 and λ_2 are said to be regularization parameters.

The reason to present J_{GAN} is not to introduce in detail the specific model proposed by Gan et al. (2023), but to show that while the above model is much more complex than the SSFCMeans proposed by Pedrycz and Waletzky (1997), it can be clearly categorized as reusing the additive combination technique - in particular, the first two terms in Eq. (2.43) directly correspond to the components Q and Q^S from Eq. (1.4). The penalization factor λ_1 , although named differently, and represented with a different greek letter, is a corresponding concept to the scaling factor α used in the additive combination technique. It is the third term, together with the second penalization factor λ_2 , that introduces the complexity aimed at estimating label safety, but note that it is also introduced in the additive manner, adhering to the additive combination technique.

In Section 1.1 of this thesis, we stressed the special meaning of the scaling factor α - that it is the only endogenous factor (in a sense that the user of the model can freely adjust its value) regulating the impact of partial supervision in the additive combination technique. Since the main topic of this thesis is the explainable impact of partial supervision, understood as the impact of the scaling factor α , our literature review is focused on this topic. Nevertheless, a categorization of the selected works can be made based on the primary focus of each of the reviewed literature. We present it in the following Section 2.2.3, and isolate the review of the scaling factor α , being the primary interest of this thesis, to the separate Section 2.2.4.

nomenclature presented in this thesis.

2.2.3 Related work in semi-supervised fuzzy clustering

The first category of works pertains to introducing partial supervision in the spirit of the additive combination technique to the baseline, unsupervised fuzzy clustering models. A canonical example is the already described Semi-Supervised Fuzzy c-Means model introduced by Pedrycz and Waletzky (1997), who extended the unsupervised Fuzzy c-Means. L. Liu and Wu (2013) extended unsupervised Possibilistic c-Means, and handled the problem of estimating coincident clusters at the same time by proposing the *center maximization* technique; Antoine et al. (2018) decided to introduce partial supervision not to the core PCM, but to the adjusted unsupervised Repulsive Possibilistic c-Means derived by Timm et al. (2004). J. Xu et al. (2019) adapted the unsupervised Interval Type-2 Fuzzy c-Means model to incorporate partial supervision, and Antoine et al. (2022) adapted Possibilistic Fuzzy c-Means model.

Many publications introducing novel models compare their proposition against the selected state of the art (although in a non-reproducible manner, which is described in detail in Section 4.2), but there are not many meta-review publications that are focused solely on comparing various existing models; it is thus worth noting such meta-review publications by Chunfang Li et al. (2008) and Lai and Garibaldi (2011).

The next category of publications addresses the data with known temporal character, the so-called data streams, building on the already proposed SSFC models. Casalino, Castellano, et al. (2020), Casalino, Dominiak, et al. (2020), and Casalino et al. (2019) reuse SSFCMeans and propose Incremental- and Dynamic Incremental Semi-Supervised Fuzzy c-Means models, and Samadi et al. (2025) proposes Weighted Semi-Supervised Possibilistic c-Means model for data stream classification.

A number of publications is focused on similar advances in the field of semi-supervised fuzzy clustering that are considered in the field of unsupervised fuzzy clustering as well. In Section 2.1.4, we described streams of research focused on differentiating features' contribution to clustering by assigning weights; we also presented the idea to use kernel functions to compute distances in high-dimensional feature spaces. In the field of SSFC, the use of kernel functions was introduced to respective models and studied by Bouchachia and Pedrycz (2006), Salehi et al. (2021) and Moghaddam et al. (2025); weighing the features was considered by H. Yu et al. (2024), Golzari Oskouei et al. (2024) and Moghaddam et al. (2025).

The next group includes works that are hard to be classified, but they all treat on various modifications of the objective functions of underlying semi-supervised fuzzy clustering models - mainly the SSFCMeans. Stutz and Runkler (2002) consider the limited range of the scaling factor $\alpha \in [0, 1]$ appropriately modifying the semi-supervised ob-

jective function, Bouchachia and Pedrycz (2003) drop the constraint relating each class to only one cluster, allowing the mapping of multiple clusters to a single class, and J. Gao et al. (2006) address the situation where supervised observations have only partially overlapping features with the unsupervised observations. S. Xu et al. (2024) introduce a pre-processing procedure aiming at filling in missing values in the partial supervision, Wu and Hou (2025) propose an asymmetric penalty in place of quadratic penalty $(u_{jk} - b_j \cdot f_{jk})^2$ from Eq. (2.41); He (2025) propose an adjustment to better handle data with imbalanced distribution of the classes.

Further, two small, separate research field seems to emerge. The first one regards the use of semi-supervised fuzzy clustering in image analysis, specifically - lesion segmentation in brain infarction (B. Zhang, Huang, et al., 2024) and cerebral infarction (B. Zhang, Jiang, & Xia, 2024). The second one is of its own kind - Kaczmarek-Majer et al. (2022) use the estimated clusters' prototypes to create linguistic summaries, which describe the general evolution of temporal numerical datasets in natural language. It is worth stressing that a direct interest in the estimated prototypes has been so far rare in the field of semi-supervised fuzzy clustering.

A separate category consists of works that we call analytical; they are focused on revisiting and improving the understanding and use of the core mechanisms of semi-supervised fuzzy clustering rather than creating novel SSFC models. This group is composed of two publications by the author of this thesis; Kmita et al. (2023) revisited the use of SSFCMeans as a classifier, discussing the matters of training and test error, and Kmita et al. (2024) provided the foundations for the explainability framework for impact of partial supervision in semi-supervised fuzzy clustering that this thesis builds on and extends to the great extent.

The next group includes works treating on the aspect related to the partial supervision introduced in semi-supervised fuzzy clustering - the reliability of labels, that is referred to under different terms by different authors. Antoine and Labroche (2018) review existing classical propositions with respect to how they handle *noisy/erroneous* labels; Kmita et al. (2022) propose a wrapping procedure to estimate *label uncertainty* using the Semi-Supervised Fuzzy c-Means. Apart from these two publications, a number of works is identified with the subfield called Safe Semi-Supervised Clustering (S3C) (Gan et al., 2023). Following publications: Gan et al. (2023, 2019, 2024, 2025, 2018), Guo et al. (2021), Huan et al. (2022), Khezri et al. (2025), and Santos et al. (2024) introduce various models – with the advancements being frequently of incremental nature – aiming at handling *noisy/erroneous* labels by estimating safety degrees associated with the supervised observations.

Last but not least, it is worth listing three publications (Yasunori et al., 2009; Yin et al., 2012; H. Zhang & Lu, 2009) that present alternative formulation of core semi-supervised fuzzy clustering model, adapting the unsupervised Fuzzy c-Means in a different way than Pedrycz and Waletzky (1997). We call this group as *silent α* publications since they also introduce partial supervision in the additive manner, similar to the additive combination technique from Eq. (1.4), but they do not explicitly use the scaling factor α (or any other corresponding hyperparameter); such an approach is in fact implicitly assuming $\alpha = 1$ in light of the additive combination technique.

2.2.4 Related work on the scaling factor α

The synthesis of the literature review on the topic of the presentation of the scaling factor α and the choice of the values to be used has been already presented in Section 1.3, where we defined the research gap. This Section 2.2.4 is devoted to the presentation of the detailed summary based on which the general conclusions presented in Section 1.3 were made. Specifically, Table 2.1 contains the results of the literature review. Its second column provides the values of the scaling factor α used in a given publication, and its third column provides the details on the approach towards describing the role of the scaling factor α ; this third column contains the synthesis of the literature review.

As stressed in Section 1.2, the topic of the scaling factor α is typically raised as a side topic in the literature treating on the semi-supervised fuzzy clustering; in Section 2.2.3, we categorized only two publications as *analytical* in a sense that they focused on revisiting the mechanism of regulating impact of partial supervision by the means of the scaling factor α ; both of the publications (Kmita et al., 2023, 2024) are authored by the author of this thesis and served as first attempts to address the issue of regulating impact of partial supervision in SSFC.

In consequence, there has lacked a unified framework for the discussion and analysis of this impact. The amount of text devoted to the description of the role of the scaling factor α , the degree of detail of respective analyses considering the impact of varying values of it, all differ between the specific works. Moreover, various terms and symbols are used to denote the same concept of a hyperparameter regulating the impact of partial supervision in the additive combination technique, making it difficult to disseminate the results and compare various approaches - for example, Gan et al. (2023) discuss “penalization factor” λ , and Salehi et al. (2021) discuss “impact intensity” β .

For the reason of coherent presentation, we formulate a following interpretation that we call the Canonical Interpretation 1.

Interpretation 1 (Pedrycz and Waletzky (1997)). *A scaling factor's α role is to maintain a balance between the supervised and unsupervised component within the optimization mechanism.*

Note that the choice of the term *interpretation* reflects the introduced definitions of interpretation (Definition 1) and explanation (Definition 2).

To enable synthesis, in the third column of Table 2.1 we mark whether a given description follows the spirit of the Canonical Interpretation 1 provided by Pedrycz and Waletzky (1997), regardless of the naming convention or exact wording used, or deviates from it - only a few descriptions do not directly consider the role of the scaling factor α as balancing the components within the objective function. Stutz and Runkler (2002) describe that the role of α is to express the reliability of the labels, and require that $\alpha \in [0, 1]$, J. Xu et al. (2019) do not interpret the role of α at all (using the symbol λ), providing only the suggested value of it (referring to the original proposition from Pedrycz and Waletzky (1997)), and B. Zhang, Huang, et al. (2024) and B. Zhang, Jiang, and Xia (2024) introduce additional hyperparameter $\eta \in [0, 1]$ to modify the value of $\alpha = N/M$; none of the above deviations proposes any new approach to the interpretation of the role of the scaling factor α . Only Kmita et al. (2023, 2024) provide novel, analytically justified procedures to create grids of α values, but they are not based on a fully explainable impact of partial supervision in semi-supervised fuzzy clustering that this thesis is aiming to provide.

The consequence of the above is that **none of the works deviates from presenting the role of α in a different context than the objective function. Therefore, none of the descriptions can be considered as an explanation in light of Definition 2, which constitutes the research gap formulated in Section 1.3.**

The second column of Table 2.1 displays the actual values of the scaling factor α used in a given publication. The difference between using a single value, or a grid of values, serves as a basis for a distinction between the **balancing** and **grid** approach in Section 1.3. As described in Section 1.3, the balancing approach relies on using a single value that is deemed to ensure the balance between the components of the objective function, and two strategies exist - either treat this value as data-dependent, or use a constant value. With respect to the first approach, Pedrycz and Waletzky (1997) suggest using $\alpha = N/M$, but others decide to use $1 - M/N$ (Chunfang Li et al. (2008) and Lai and Garibaldi (2011)), or N/M (J. Xu et al. (2019)). With respect to the constant value, Antoine and Labroche (2018), Antoine et al. (2018), and Kaczmarek-Majer et al. (2022) choose the value $\alpha = 1$, and Khezri et al. (2025) choose the value $\alpha = 2$.

Table 2.1: Summary of literature review on the topic of the scaling factor α .

publication	α values used	description of α
Pedrycz and Waletzky (1997)	N/M	Canonical Interpretation 1
Chunfang Li et al. (2008)	$1 - M/N$	Canonical Interpretation 1
Lai and Garibaldi (2011)	$1 - M/N$	Canonical Interpretation 1
Stutz and Runkler (2002)	$\alpha \in [0, 1]$	Other: “ $\alpha \in [0, 1]$ is a weighting factor that is chosen corresponding to the reliability of the labels”
Bouchachia and Pedrycz (2003)	$\{0.3, 0.5, 0.7, 0.9, 1, 2\}$	Canonical Interpretation 1
Bouchachia and Pedrycz (2006)	$\{0.3, 0.5, 0.7, 0.9, 1\}$	Canonical Interpretation 1
J. Gao et al. (2006)	<i>Not available</i>	Canonical Interpretation 1
L. Liu and Wu (2013)	$\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 10, 100\}$	Canonical Interpretation 1
Antoine et al. (2018)	1	Canonical Interpretation 1
J. Xu et al. (2019)	M/N	Other: “ λ is the ratio of labeled sample points in the data sample”
Casalino et al. (2019)	N/M	Canonical Interpretation 1
Casalino, Dominiak, et al. (2020)	<i>Not available</i>	Canonical Interpretation 1
Casalino, Castellano, et al. (2020)	<i>Not available</i>	Canonical Interpretation 1
Salehi et al. (2021)	N/M	Canonical Interpretation 1
Antoine et al. (2022)	$\{0.01, 0.05, 0.1, 0.5, 1\}$	Canonical Interpretation 1
S. Xu et al. (2024)	<i>Not available</i>	Canonical Interpretation 1
H. Yu et al. (2024)	$\{3, 3000\}$	Canonical Interpretation 1
B. Zhang, Huang, et al. (2024)	$\alpha = N/M, \eta \in [0, 1]$	Canonical Interpretation 1, η to modify α
B. Zhang, Jiang, and Xia (2024)	$\alpha = N/M, \eta \in [0, 1]$	Canonical Interpretation 1, η to modify α
Golzari Oskouei et al. (2024)	$\{1, 2, 4\}$	Canonical Interpretation 1
Wu and Hou (2025)	$\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$	Canonical Interpretation 1
He (2025)	$\{0.1, 0.5, 0.8, 1, 1.2, 1.5, 2, 2.5, 3\}$	Canonical Interpretation 1
Samadi et al. (2025)	$\{0.01, 0.05, 0.1, 0.5, 1\}$	Canonical Interpretation 1
Moghaddam et al. (2025)	$\{1, 2\}$	Canonical Interpretation 1
Kaczmarek-Majer et al. (2022)	1	Canonical Interpretation 1

Kmita et al. (2023)	$\{0.05, 0.11, 0.18, 0.25, 0.33, 0.43, 0.54, 0.67, 0.82, 1\}$	Other: analytical procedure to derive grid based on the idea of the so-called “margin of making the train error” related to the use of SSFCMeans as a classifier.
Kmita et al. (2024)	$\{0.12, 0.28, 0.49, 0.79, 1.22\}$	Other: analytical procedure to derive grid based on the quantified impact of partial supervision on the decision memberships as a function of α .
Antoine and Labroche (2018)	1	Canonical Interpretation 1
Kmita et al. (2022)	M/N	Canonical Interpretation 1
Gan et al. (2018)	$\{0.001, 0.005, 0.01, 0.1, 0.5, 1, 10, 100\}$	Canonical Interpretation 1
Gan et al. (2019)	$\{0.001, 0.01, 0.1, 1, 10, 100\}$	Canonical Interpretation 1
Guo et al. (2021)	$\{0.001, 0.01, 0.1, 1, 10, 100\}$	Canonical Interpretation 1
Huan et al. (2022)	<i>Not available</i>	Canonical Interpretation 1
Gan et al. (2023)	$\{0.001, 0.005, 0.01, 0.1, 0.5, 1, 10, 100\}$	Canonical Interpretation 1
Gan et al. (2024)	$\{0.001, 0.01, 0.1, 1, 10, 100\}$	Canonical Interpretation 1
Santos et al. (2024)	$\{0.001, 0.01, 0.1, 1, 10, 100\}$	Canonical Interpretation 1
Khezri et al. (2025)	2	Canonical Interpretation 1
Gan et al. (2025)	$\{0.001, 0.01, 0.1, 1, 10, 100\}$	Canonical Interpretation 1
Yasunori et al. (2009)	silent $\alpha = 1$	Silent α
H. Zhang and Lu (2009)	silent $\alpha = 1$	Silent α
Yin et al. (2012)	silent $\alpha = 1$	Silent α

Source: own work.

With respect to the *grid* approach, Table 2.1 shows differentiated approach towards the selection of boundaries a_1 and a_q and selection of the scale of the grid. The scales vary from linear ($\mathcal{A} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ in Wu and Hou (2025)), through mixed ($\mathcal{A} = \{0.01, 0.05, 0.1, 0.5, 1\}$ in Antoine et al. (2022)) to non-linear, of exponential nature ($\mathcal{A} = \{0.001, 0.01, 0.1, 1, 10, 100\}$ in Gan et al. (2025)). Importantly, none of the choices of the scale or boundaries is thoroughly justified in an analytical way.

Finally, all the publications using the grid approach do not study the impact of varying values of α directly on the outcomes of the respective fuzzy clustering model ($\tilde{U}, \tilde{V}$); they all focus on external quality criteria as depicted in Fig. 1.2. For example, Antoine et al. (2022) experimentally study the impact of values from their proposed grid $\{0.01, 0.05, 0.1, 0.5, 1\}$ using the Adjusted Rand Index external quality criterion, and Samadi et al. (2025) consider accuracy, precision, recall, and $F1$ -score in their experimental study of the impact of values from the grid $\{0.01, 0.05, 0.1, 0.5, 1\}$.

Chapter 3

Explanations of the impact of partial supervision

This chapter is devoted to resolving the problem stated in Section 1.2 - to explain the impact of partial supervision in semi-supervised fuzzy clustering. Section 3.1 provides a literature review on the related work in the field of Explainable Artificial Intelligence. A concrete approach towards the matter of explainability was used in this thesis to formulate the problem statement, and identify the research gap. This selected approach is presented in the context of competing approaches to the problem of explaining Artificial Intelligence models or their components. Section 3.2 is focused on the proposed explainability framework, a collection of novel formalism, notation, definitions, discussions, and conceptualizations of certain phenomena that have either not been discussed in the literature before, or discussed to the little extent. Finally, in Sections 3.3 and 3.4, the actual explanations of the impact of the scaling factor α – based on the introduced framework – are provided for the two classical SSFC models: Semi-Supervised Fuzzy c-Means, and Semi-Supervised Possibilistic c-Means.

It shall be noted that this Chapter is a continuation of the initial research conducted by the author of the thesis in publication (Kmita et al., 2024), where the postulate for developing explanations of the impact of partial supervision was first introduced. The scope of discussion in the above-mentioned publication is more limited than in this Chapter, but it served as a building block for the proposed developments. In particular, regarding the limitations, Kmita et al. (2024) focused on explaining the impact of partial supervision on memberships only, whereas in this Chapter we present the significance of considering this impact on the prototypes as well; moreover, Kmita et al. (2024) have not discussed the issues related with the semantics of membership degrees in semi-supervised fuzzy clustering presented in this Chapter.

3.1 Related work in Explainable Artificial Intelligence

Explainable Artificial Intelligence (abbreviated as XAI) is a highly popular research field, with a multitude of researchers with various background and interests contributing to the developments of XAI. Indeed, the volume of academic research on Explainable Artificial Intelligence is huge and increasing. Excellent survey papers proposing a taxonomy of XAI research can be found e.g. in (Ali et al., 2023; Barredo Arrieta et al., 2020; Guidotti et al., 2019; Saeed & Omlin, 2023).

The recent rise of popularity of XAI pertains greatly to the rise of very complex models with millions or more parameters that are referred to as deep learning, and the identification of AI models is frequently associated with these models (Ali et al., 2023). These deep learning models are often referred to as *black-box* models to stress that it is very challenging for a human being to directly comprehend their inner logic. In this context, Explainable AI is a task of enabling the understanding of this logic by human beings (Ali et al., 2023).

Therefore, it may not seem straightforward to consider the problem of regulating impact of partial supervision in semi-supervised fuzzy clustering in the context of XAI. However, the task of explaining black-box models is not the only line of work on the matter of explainability in Artificial Intelligence. A different subfield of XAI that in this thesis we are referring to is focused on a general problem of defining criteria for an explanation of any (broadly understood) AI model. In this approach, ISO/IEC 22989:2022 (2025) define Artificial Intelligence model as “engineered system that generates outputs such as content, forecasts, recommendations or decisions for a given set of human-defined objectives”. Indeed, such a broad definition includes not only deep learning, black-box models, but also classical models, developed in the fields traditionally named as statistical learning, or computational intelligence. Hence, it also includes the class of SSFC that generate content or decisions – in the understanding of ISO/IEC definition – in form of membership matrices and cluster’s prototypes matrices.

In broader sense, the exact definition of what constitutes an explanation of AI system is still an open question, as Confalonieri et al. (2021) note. The way to understand it is that the matters of defining, understanding, and conceptualizing the concept of an explanation have been considered in various fields of science, leading to severe terminological inconsistencies. The need of *explainability* was also phrased as the need of *interpretability*, or even *transparency*, leading to questions how the different terms relate to each other. Ali et al. (2023) note that the choice and popularity of specific term depended on the research field, e.g. the community of machine learning (ML) researchers was more prone

to use the term *interpretability* rather than *explainability* when discussing the same need of conveying safety and trust.

While the terminological inconsistency in the approach treating explanations as methods to shed more light on the inner logic of highly complex black-box models is merely an inconvenience, it becomes a serious challenge in the generic approach to the matter of explainability. This different line of work is associated with defining what an explanation is in a sense of a set of criteria that a given description (understood as a free text treating on a certain subject) of a model – or its parts – must satisfy. In this paradigm, a distinction is made between an interpretation, and an explanation. Scholars differ in defining the relation between these terms, but it is widely accepted that interpretations and explanations are separate concepts. In the remaining of this Section, we present the related work associated with the two approaches to the matter of Explainable Artificial Intelligence discussed above, that we name as the **narrow understanding** and **broad understanding**, accordingly.

3.1.1 Narrow understanding of explanation.

Rudin and Radin (2019) in their paper reflect on XAI following the *Explainable Machine Learning Challenge* organized during a renowned *Neural Information Processing Systems* (NeurIPS) conference in 2018, which they claim to be one of the turning points for the growth of importance of XAI, since it was the earliest data science competition to explicitly emphasize the necessity of understanding the results produced by black-box models, which dominate machine learning-driven decision making processes (Rudin & Radin, 2019). In their reflections, Rudin and Radin (2019) use a nomenclature based on a dichotomy: black-box models, that need to be additionally *explained*, versus white-box models, that are “inherently interpretable models that provide their own explanations”. A similar understanding of interpretability and explainability is also presented in (Phillips et al., 2021), where it is stated that “self-interpretable models are models that are themselves the explanations”, in a sense that one can comprehend the reason for obtaining a given output that a self-interpretable model produces. This approach to the matter of interpretability and explainability is strictly related to these highly complex black-box models.

The task of explaining a model’s output is treated rather specifically in terms of the supervised learning branch, where the problem is to build a decision function $\rho : X \mapsto \mathcal{Y}$, where $\mathcal{Y}$ is a response variable, either a continuous variable, or a set of labels representing the possible classes of the objects represented numerically in the features matrix X . The goal is then to explain an individual prediction $\hat{y} = \rho(\mathbf{x})$.

XAI understood this way can be seen as a toolbox of methods helping to understand

the inner logic of complex models. Indeed, much of the research has been devoted to the development and refining of such toolbox, or creating explainable counterparts, refer to the survey by (Ali et al., 2023). A suite of state-of-the-art methods is described in a textbook by Molnar (2025). Molnar (2025) differentiates between ML models interpretable by-design, or such models that require additional explainability methods to achieve post-hoc interpretability. In terms of the language used in (Rudin & Radin, 2019), these by-design and post-hoc interpretable models correspond to inherently interpretable white-box models, and black-models (that require XAI techniques to become interpretable), respectively.

In this paradigm, the class of semi-supervised fuzzy clustering models falls in the the same category that has been labeled different ways: as white-box models, interpretable by-design, or self-interpretable models, together with such classical models as logistic regression or decision trees.

3.1.2 Broad understanding of explanation.

In the alternate, broad understanding, the need for explainability applies equivocally both to white-box, and black-box models. The building block of this approach to XAI is a mere definition of what Artificial Intelligence – part of the Explainable Artificial Intelligence term – means, that includes a broader scope than sole focus on highly complex, black-box models. In addition to the already introduced definition by ISO/IEC 22989:2022 (2025), we also refer to the unfinished standard by Institute of Electrical and Electronics Engineers (IEEE) that defined AI systems in a similar spirit in their Project Authorization Request of *Standard for Requirements and Classification of Explainable Artificial Intelligence Systems*, where it is motivated that “the aim [of the standard] is to provide researchers, developers and designers of AI (including machine learning, rule-based, neural network and other) systems and industrial applications with a unified and high-level methodology for classification of their products as partially or fully explainable” (IEEE SA, 2025).

The broad range of models included in these definitions requires a different source for the distinction between interpretability and explainability than the white-box versus black-box dichotomy. The approaches presented in the literature within this paradigm can be grouped based on following two ideas:

1. explanations as a concept superior to interpretation,
2. explanations and interpretations as complementary concepts.

Superiority concept. The first approach is represented e.g. by Gilpin et al. (2018), who argue that interpretability and explainability should be distinguished, and claim the superiority of explanations by stating that models that are explainable are also interpretable, whereas the reverse does not always hold. They propose that within the context of AI, an explanation of a model can be assessed due to two dimensions: its interpretability, and completeness, where interpretability aims at describing the internal mechanisms of a system in a form that humans can understand whereas completeness aims at describing the operations of a system in a form that is accurate. While somewhat vague, the above can be understood that interpretations do not need to be complete on their own, can be even inaccurate - the only requirement is that they are understandable to humans. Thus, explanations are superior to interpretations, as they must be also *complete*.

A similar approach is represented by Roscher et al. (2020). Molnar (2025) summarized the definitions from (Roscher et al., 2020), stating that interpretability concerns mapping abstract model-related concepts into representations that humans can understand and explainability is a more demanding notion that extends interpretability and requires to provide additional context.

Finally, the superiority paradigm is also presented by Phillips et al. (2021), although no verbal distinction between an interpretation and an explanation is made. The title of their work – “Four principles of explainable artificial intelligence” – reflects their approach; the four principles are named as *explanation*, *meaningful*, *explanation accuracy*, *knowledge limits*. The explanation itself is defined as “evidence, support, or reasoning related to an outcome from AI system”. As the authors stress, on its own, the explanation principle does not depend on the explanation being correct, informative, or understandable. These are the other principles out of the four principles, namely the principle that requires explanations to be *meaningful*, and a principle that is focused on *explanation accuracy*, that define the quality features or metrics according to which explanations can be compared, or assessed. Even though Phillips et al. (2021) differ in their wording, their approach resembles (Gilpin et al., 2018) or (Roscher et al., 2020) in a sense that one can have superior explanations, and inferior explanations (e.g., not adhering to the principle of being *meaningful*), whereas the other approaches explicitly treat explanations as superior to interpretations.

Complementary concept. The publications treating interpretations and explanations as complementary concepts that will be used to analyze the impact of partial supervision in semi-supervised fuzzy clustering include the work done by International Standards Organisation and International Electrotechnical Commission in (ISO/IEC TS 6254:2025, 2025), and (Broniatowski, 2021).

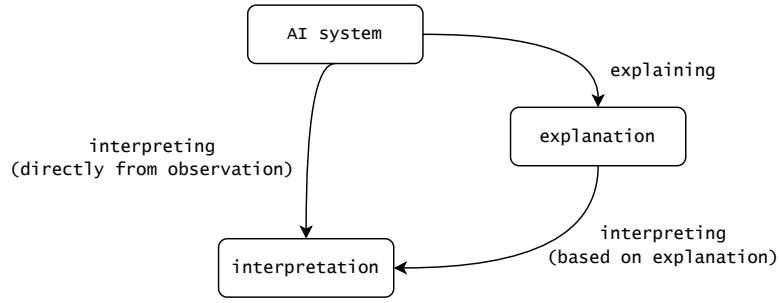
Table 3.1: Selected definitions from the ISO/IEC standard.

AI system	engineered system that generates outputs such as content, forecasts, recommendations or decisions for a given set of human-defined objectives
decision	content or item produced by the AI system as a fulfilment of its task, based on a given input Note 1: The decision can be a class, but also any other form of structured or unstructured data (e.g. a sentence, an image).
interpretation	result of understanding (by a human) a given rationale justification Note 1: Interpretations can pertain to a decision or to an AI system. Note 2: Interpretations can be produced either based on a received explanation or directly from observation without an explicit act of expression.
explanation	result of expressing a given rationale or justification in a way that humans can understand Note 1: Explanations can pertain to a decision or to an AI system.
explainability	property of an AI system that enables a given human audience to comprehend the reasons for the system’s behaviour Note 1: Explainability methods are not limited to the production of explanations, but also include the enabling of interpretations.

Source: (ISO/IEC TS 6254:2025, 2025).

Table 3.1 presents selected definitions from the ISO/IEC Standard “Objectives and approaches for explainability and interpretability of ML models and AI systems”. The definition of *AI system* has already been presented, as it aligns with earlier ISO/IEC 22989:2022 (2025) document. Due to the variability in the structure of what AI systems may produce (content, forecasts, recommendations, or – in the case of SSFC – partitions of observations and prototypes) ISO/IEC TS 6254:2025 (2025) uses the term *decision* to collectively refer to the content provided by AI model based on the input. In the case of semi-supervised fuzzy clustering, the decision of the model are the numerical matrices - estimated memberships and prototypes.

The description of *interpretation* and *explanation* present in Table 3.1 clearly shows the complementary character of these concepts. While the mere definitions do not shed much light on the case of the problem of explainable impact of partial supervision in SSFC (they differentiate between the results of “understanding a given rationale” and “expressing a given rationale”), these are the notes to entries that are of special interest. It is clearly stated that an interpretation does not need to be based on the explanation, and can be made directly based on observation. This is stressed even more in the notes to the definition of *explainability*, where it is directly stated that one of the goals of inventing explainability methods is to enable interpretations. The summary of the approach

Figure 3.1: Schematic approach to two ways of interpreting AI system.

Source: own work based on (ISO/IEC TS 6254:2025, 2025).

undertaken in (ISO/IEC TS 6254:2025, 2025) is presented in Fig. 3.1.

A different work presenting interpretations and explanations as complementary concepts is Broniatowski (2021). It approaches the problem of distinguishing interpretations from explanations in the context of AI models from the perspective of experimental psychology. Broniatowski (2021) postulates that interpretation, and explanation, are separate psychological processes. Broniatowski (2021) does not provide a distinct definition of an *interpretation* in a sense of precise, standards-like definition, similar to the style of (ISO/IEC TS 6254:2025, 2025), but rather describes his approach in a comprehensive, verbose manner, characterizing what an interpretation is and how it differs from an explanation.

According to Broniatowski (2021), interpretations are “simple, yet meaningful, «gist» mental representations” that put a certain stimulus in context and draw upon a person’s prior knowledge whereas an explanation of a model result is “a description of how a model’s outcome came to be”, which consists of a description of the mechanism or steps through which that outcome was produced. Broniatowski (2021) next compares the difference between two terms in a single sentence, stating that while an interpretation aims to help a human make sense of a presented stimulus, an explanation aims to describe the process responsible for generating a result.

Although Broniatowski (2021) precedes the ISO/IEC TS 6254:2025 (2025) by a few years, the understanding of interpretations and explanations are similar - an interpretation, as a mental gist, can be produced by direct observation, or based on an explanation, which provides the ability for a technical practitioner to comprehend the mechanisms of the AI system in greater detail. In this sense, interpretations are necessary for humans to make decisions, and an interpretation is not inferior to an explanation, it is a different concept. Nevertheless, one can postulate that a *good* interpretation should be based on a proper explanation - which is schematically depicted as *interpreting (based on*

explanation) in Fig. 3.1.

3.2 The proposed explainability framework for semi-supervised fuzzy clustering

Recall Definition 2 formulated in the Section 1.2 stating that the explanation of the role of the scaling factor α is a human-understandable, meaningful description of the role of α that describes its impact on the outcome of semi-supervised fuzzy clustering model $(\tilde{U}, \tilde{V})$.

The goal of this definition was to clearly specify the context that an explanation must account for: how the scaling factor α impacts the decision of the SSFC model. This led to the formulation of the research gap, where it was described that none of the approaches in the literature accounted for the entirety of this context related to the decision of the fuzzy clustering model in form of numerical matrices $(\tilde{U}, \tilde{V})$.

While the above definition is clear and concise, the delivery of an actual explanation of the impact of partial supervision is a complex task. As Broniatowski (2021) put it, to achieve an explanation of a decision of a given AI model, the explanations “are detailed, technical, (...) typically more appropriate for technical practitioners, who can rely on extensive background knowledge (...)”. The research gap identified in this thesis, i.e., the lack of explainability of the impact of partial supervision in SSFC, resulted in the lack of appropriate formalism, notation, terminology, and attention to the certain aspects of semi-supervised fuzzy clustering that would constitute the *detailed, technical background knowledge* that Broniatowski (2021) is referring to.

This Section is devoted to filling in this gap. While the actual explanations of the impact of the scaling factor α must be specific to a given model, and thus are presented in separate Sections 3.3 and 3.4, they are based on a generic framework proposed in this Section 3.2. The framework itself is model-agnostic, as it is aimed at enabling the explanations in any semi-supervised fuzzy clustering model in a consistent, unified way, meeting the introduced criteria of an explanation.

3.2.1 Generic fuzzy clustering objective function

In Section 1.1, we stated that the unsupervised and semi-supervised fuzzy clustering models can be treated in general as a single class of fuzzy clustering models, showing that the optimization problem is the same in each case. This approach requires a formulation of a novel, generic representation of the fuzzy clustering objective function, a common

platform for both the unsupervised, and semi-supervised subproblems.

The novel presentation is based on an interpretation of the memberships in fuzzy clustering (in particular, Fuzzy c-Means in the specific reference) provided by J. C. Bezdek (1981). In this approach, considering the objective function of FCM defined in Eq. (2.7) that we reproduce below for convenience:

$$\sum_{j=1}^N \sum_{k=1}^c u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2, \quad (3.1)$$

one can treat u_{jk}^m as a weight for the squared distance between $\mathbf{x}_j$ and $\mathbf{v}_k$, which role is to regulate the contribution of a given j th observation's feature vector in the final shape of the k th cluster's prototype. As J. C. Bezdek (1981) stated, "the squared distance is weighted by $(u_{jk})^m$, the m th power of $\mathbf{x}_j$'s membership in fuzzy cluster u_k ". **In this view, the main mechanism of partial supervision in semi-supervised fuzzy clustering would be to adjust these weights.**

The first step on the way to create a generic objective function is to define a weighing function

$$\phi : u_{jk} \rightarrow \mathcal{R}_+, \quad (3.2)$$

and substitute u_{jk} with $\phi(u_{jk})$ in Eq. (3.1), arriving at

$$\sum_{j=1}^N \sum_{k=1}^c \phi(u_{jk}) \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2. \quad (3.3)$$

This representation handles both Fuzzy c-Means and Semi-Supervised Fuzzy c-Means, where the concrete form of the weighing function defines the particular model. In case of Fuzzy c-Means,

$$\phi_{\text{FCM}}(u_{jk}) = u_{jk}^m, \quad (3.4)$$

and in case of Semi-Supervised Fuzzy c-Means as proposed in Pedrycz and Waletzky (1997) (refer to Eq. 2.41),

$$\phi_{\text{P97}}(u_{jk}) = u_{jk}^2 + \alpha(u_{jk} - b_j f_{jk})^2. \quad (3.5)$$

However, the sole use of the weighing function ϕ is not enough to create a generic objective function, since in many models, starting with the Possibilistic c-Means, the memberships not only weigh the distances, but also are a subject of a certain form of regularization on their own - in case of the objective function of Possibilistic c-Means (refer to Eq. 2.24)

$$\begin{aligned}
 Q_{\text{PCM}}(U, V) &= \sum_{j=1}^N \sum_{k=1}^c u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2 + \sum_{j=1}^N \sum_{k=1}^c \gamma_k (1 - u_{jk})^m \\
 &= \sum_{j=1}^N \sum_{k=1}^c \left(u_{jk}^m \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2 + \gamma_k (1 - u_{jk})^m \right)
 \end{aligned} \tag{3.6}$$

this regularization is implemented by the term $\gamma_k \cdot (1 - u_{jk})^m$. Clearly, the use of weighing function ϕ on its own is not enough for the ability to represent a generic objective function applicable to the entire class of fuzzy clustering models. Rather, one needs an additional function p , and this leads to the truly generic representation

$$G(U, V) = \sum_{j=1}^N \sum_{k=1}^c \left(\phi(u_{jk}) \cdot \|\mathbf{x}_j - \mathbf{v}_k\|^2 + p(u_{jk}) \right). \tag{3.7}$$

The generic representation G from Eq. (3.7) enables clear analysis of the optimal prototypes $\hat{V}$, as for any fuzzy clustering model, the argument minimum of the partial objective function $G(V; U)$ is found by setting the derivative of the function of the k th cluster's prototype $g(\mathbf{v}_k)$ to $\mathbf{0}$. In this action, we will use a simplified notation for easier presentation

$$\phi(u_{jk}) = \phi_{jk}. \tag{3.8}$$

Then, the generic formula for the optimal prototype can be calculated as

$$\frac{\partial g}{\partial \mathbf{v}_k} = -2 \sum_{j=1}^N \phi_{jk} (\mathbf{x}_j - \mathbf{v}_k) = \mathbf{0} \tag{3.9a}$$

$$\sum_{j=1}^N \phi_{jk} \mathbf{x}_j = \left(\sum_{j=1}^N \phi_{jk} \right) \mathbf{v}_k \tag{3.9b}$$

and hence, the optimal cluster's prototype is defined as

$$\hat{\mathbf{v}}_k = \frac{\sum_{j=1}^N \phi_{jk} \mathbf{x}_j}{\sum_{j=1}^N \phi_{jk}} = \sum_{j=1}^N \frac{\phi_{jk}}{\sum_{s=1}^N \phi_{sk}} \mathbf{x}_j. \tag{3.10}$$

3.2.2 Outcome of fuzzy clustering model

In Section 1.2, we introduced the notation $\tilde{U}$ to distinguish the outcome – the numerical, estimated membership matrix – from a conceptual memberships matrix U , which is an element of a certain space, e.g. in case of Fuzzy c-Means U is an element of M_{fc} as defined

in Eq. (2.3) in Definition 3. In this Section, we provide a more detailed presentation of this decision of any fuzzy clustering (unsupervised or semi-supervised) model¹.

A pair (U, V) is an entry to the fuzzy clustering, the variables to be optimized that we will call conceptual memberships and conceptual prototypes, correspondingly; the pair $(\tilde{U}, \tilde{V})$ is the outcome, the output obtained by providing a specific input to the model. In between, as depicted in Fig. 1.1 in Section 1.3, there are a formulation of the optimization problem and a solution of it in form of the algorithm.

Let M_* be a given c -partition space considered in a specific model, e.g., M_{fc} in the case of Fuzzy c-Means. Then, in case of fuzzy clustering, the solution of the optimization problem consists of optimal memberships

$$\hat{U} = \arg \min_{U \in M_*} J(U; V) \quad (3.11)$$

and optimal prototypes

$$\hat{V} = \arg \min_{V \in \mathcal{R}^{c \times p}} J(V; U), \quad (3.12)$$

and an Alternating Optimization algorithm to iterate between these two, described in detail in Section 2.1.2.

Importantly, both optimal memberships $\hat{U}$ and optimal prototypes $\hat{V}$ are matrix functions. Indeed,

$$\hat{U} : \mathcal{R}^{c \times p} \mapsto M_* \quad \text{s.t.} \quad \hat{U}(V) = [\hat{u}_{jk}(V)], \quad (3.13)$$

where $\hat{u}_{jk}$ is an individual optimal membership, a scalar function, and

$$\hat{V} : M_* \mapsto \mathcal{R}^{c \times p} \quad \text{s.t.} \quad \hat{V}(U) = [\hat{v}_k(U)], \quad (3.14)$$

where $\hat{v}_k$ is optimal cluster's prototype, a vector function.

It is important to stress that both $\hat{U}$ and $\hat{V}$ are matrix functions, and a concrete numerical matrix is obtained when providing a respective argument. We will call

- $\hat{U}(V)$ estimated memberships matrix,
- $\hat{V}(U)$ estimated prototypes matrix.

There has not been so far proposed any separate symbol devoted to denote estimated memberships $\hat{U}(V)$ and estimated prototypes $\hat{V}(U)$. Until now, either: (i) symbol U was used in the literature to denote conceptual, optimal and estimated memberships at the same time, and the meaning of the symbol depended on the context, or (ii) symbol $\hat{U}$

¹Recall that in this thesis, the terms decision/outcome are used interchangeably to refer to $\tilde{U}$ and $\tilde{V}$.

Algorithm 3 Alternating Optimization Algorithm for Fuzzy Clustering

Set convergence threshold $\epsilon > 0$
 Set loop counter $l = 0$
 1: Initialize the algorithm with $U^{(0)} \in M_{fc}$
 2: **repeat**
 3: $l \leftarrow l + 1$
 4: $L \leftarrow l$
 5: $\tilde{V}^{(l)} = \hat{V}(\tilde{U}^{(l-1)})$
 6: $\tilde{U}^{(l)} = \hat{U}(\tilde{V}^{(l)})$
 7: **until** $\|\tilde{U}^{(l)} - \tilde{U}^{(l-1)}\| < \epsilon$
 8: **Output:** $\tilde{U} \equiv \tilde{U}^{(L)}, \tilde{V} \equiv \tilde{V}^{(L)}$

was used to denote both optimal memberships (a function), and estimated memberships (a numerical matrix). The same applies to the prototypes, analogously. An appropriate terminology requires such a separate symbol for adequate precision of description.

While it might seem that the **tilde** symbol used in the proposed notion of an outcome of the model $(\tilde{U}, \tilde{V})$ is an adequate one to fill this gap, we cannot associate $\tilde{U}$ with estimated memberships $\hat{U}(V)$ - at least not directly. This is because the outcome of the fuzzy clustering model is a result of the Alternating Optimization algorithm, where estimated memberships and prototypes are iteratively calculated until satisfying the convergence criterion. Thus, the adequate formalism requires a reference to this iterative nature. A respective algorithm using a novel terminology introduced in further paragraphs is presented in Algorithm 3.

In each l th loop, $l = 1, \dots, L$, the algorithm estimates memberships and prototypes based on the estimates obtained in a preceding loop, and at the end of each loop, a convergence criterion is checked. In this sense, the estimated memberships are inherently related to the loop counter of the algorithm, and we will denote them as $\tilde{U}^{(l)}$ and $\tilde{V}^{(l)}$. They are defined as

$$\tilde{V}^{(l)} = \hat{V}(\tilde{U}^{(l-1)}), \quad (3.15)$$

$$\tilde{U}^{(l)} = \hat{U}(\tilde{V}^{(l)}). \quad (3.16)$$

The decision of the fuzzy clustering model is then defined as a pair of the estimated memberships and estimated prototypes in the last, L th loop of the AO algorithm. Since frequently, in practical scenarios, it is only this outcome – not the interim estimates – that is of interest, we will denote for simplicity

Table 3.2: Summary of introduced nomenclature.

symbol	term	description
U	conceptual memberships	a variable to be optimized
V	conceptual prototypes	a variable to be optimized
$\hat{U}$	optimal memberships	a matrix function
$\hat{V}$	optimal prototypes	a matrix function
$\tilde{U}^{(l)}$	estimated memberships	numerical matrix estimated in l th loop of the algorithm
$\tilde{V}^{(l)}$	estimated prototypes	numerical matrix estimated in l th loop of the algorithm
$\tilde{U}$	decision memberships	numerical matrix estimated in the last, L th loop of the algorithm
$\tilde{V}$	decision prototypes	numerical matrix estimated in the last, L th loop of the algorithm

Source: own work.

$$\tilde{U} \equiv \tilde{U}^{(L)}, \quad (3.17)$$

$$\tilde{V} \equiv \tilde{V}^{(L)}. \quad (3.18)$$

To recap, in Table 3.2 a summary of introduced notation and nomenclature is presented.

3.2.3 Impact of partial supervision as a function

The decision of a semi-supervised fuzzy clustering model is a composite outcome of functions $\hat{V}$ and $\hat{U}$ applied L times. To explain the impact of partial supervision, it is necessary to study the role of the scaling factor α as the argument of these functions in the partially supervised case. This exercise must consider the loop iterator l as a factor of the analysis, as potentially, the scaling factor might have varying impact depending on the loop of the algorithm. Moreover, the analysis should also consider the individual character of the impact, i.e., discuss the impact of any j th observation to the k th cluster.

We will denote the result of the analysis specified above as functional impact of the scaling factor α - separately on memberships and prototypes, to reflect the iterative nature of the solution to the optimization problem, where memberships and prototypes are optimized separately. We will use the symbol π that in Section 1.2 was used to denote a general concept of a relation $\pi(\alpha, (\tilde{U}, \tilde{V})) : \alpha \rightarrow (\tilde{U}, \tilde{V})$ in Eq. (1.6); however, having introduced semi-supervised fuzzy clustering in greater detail, we can operate under the more stringent conditions, and define $\pi : \alpha \rightarrow \pi(\alpha)$ to be a function which we will analyze.

Therefore, we will denote

$$\pi_U(\alpha; j, k, l) \quad \text{as functional impact of partial supervision on memberships,} \quad (3.19)$$

$$\pi_V(\alpha; j, k, l) \quad \text{as functional impact of partial supervision on prototypes.} \quad (3.20)$$

As defined in Table 3.1, the decision is *a content or item produced by the AI system as a fulfilment of its task, based on a given input*. The term *functional* used in Eq. (3.19) and Eq. (3.20) is meant to stress the nature of the **task** that SSFC models **fulfill** to obtain the decision: the optimization algorithm presented in Algorithm 3. However, the functional impact does not focus on the other part of the definition of a decision - the **given input**, which in case of SSFC includes all the experimental factors provided to the model: observations X , the differences in the characteristics of the supervised part X^S of the observations, and the unsupervised part X^U , and the configuration of all the remaining hyperparameters other than the scaling factor α . The analysis of the role of the scaling factor α in this context will be called the empirical impact of partial supervision; we will denote it as $\tilde{\pi}(\alpha)$. The empirical impact of partial supervision is analyzed in Section 5.2.

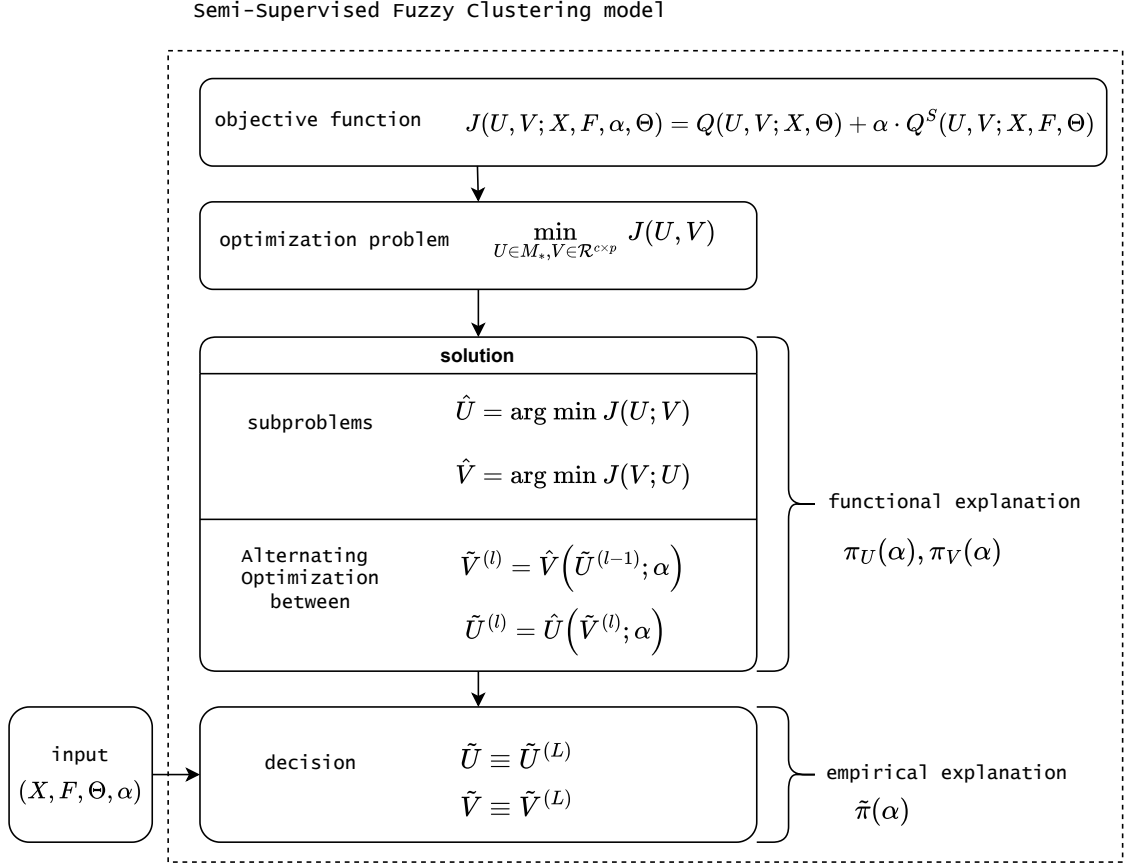
The difference between the functional impact and the empirical impact can be phrased that the empirical analysis of partial supervision requires the assumption of *ceteris paribus* conditions, and studies the impact of α on the numerical results of $(\tilde{U}, \tilde{V})$ in presence of all the other factors fixed; on the other hand, the functional analysis of the impact of partial supervision directly concerns the role of the scaling factor α as an argument of respective functions $\hat{U}$ and $\hat{V}$, and is not focused on the other factors constituting the input to the decision. These two contexts are depicted in Fig. 3.2, which uses the nomenclature established in this Chapter, and thus is an enhanced version of the introductory Fig. 1.1.

Nevertheless, this specific Section is devoted to the functional impact of partial supervision, and as stated above, its analysis must consider the potentially varying impact across different loops l of the Algorithm 3. We will now analyze this matter. For this purpose, a functional representation of the decisions $(\tilde{U}^{(L)}, \tilde{V}^{(L)})$ will be more convenient. Let $\circ$ denote functional composition. Then, we have

$$\begin{aligned} \tilde{U}^{(L)} &= \hat{U}(\tilde{V}^{(L-1)}) \\ &= \hat{U}((\hat{V} \circ \hat{U})^{(L-1)}(\tilde{V}^{(1)})), \end{aligned} \quad (3.21)$$

From Eq. (3.21), we can clearly see that for a specific fuzzy c -partition space M_* , we have

Figure 3.2: Impact of partial supervision, its explanation and the context of semi-supervised fuzzy clustering model.



Source: own work.

$$\hat{U} : \mathcal{R}_{c \times p} \rightarrow M_* \quad \forall l = 1, \dots, L, \quad (3.22)$$

where M_* denotes a certain c -partitions space.

A different case is for the decision $\tilde{V}$. We have

$$\tilde{V}^{(L)} = \hat{V}(\tilde{U}^{(L-1)}) \quad (3.23a)$$

$$= \hat{V}((\hat{U} \circ \hat{V})^{(L-1)}(\tilde{U}^{(1)})) \quad (3.23b)$$

$$= \hat{V}((\hat{U} \circ \hat{V})^{(L-1)}(\hat{U} \circ \hat{V})(U^{(0)})). \quad (3.23c)$$

Eq. (3.23) clearly shows the loop-specific impact of partial supervision on the prototypes. In the first loop of the algorithm, it is the initialization matrix $U^{(0)} \in M_{fc}$ that is fed

into the optimal prototypes function $\hat{V}$ - which is visible in Eq. (3.23c). Note that this initialization matrix is not impacted by partial supervision.

In all the consecutive loops $l > 1$, it is the estimated memberships $\tilde{U}^{(l-1)}$, impacted by the partial supervision, that is fed into the optimal prototypes $\hat{V}$ - which is visible in Eq. (3.23b). In consequence, the domain of the optimal prototypes function changes between the first loop, and all the consecutive loops:

$$\hat{V} : M_{fc} \rightarrow \mathcal{R}_{c \times p} \quad \text{if } l = 1, \quad (3.24a)$$

$$\hat{V} : M_* \rightarrow \mathcal{R}_{c \times p} \quad \text{if } l > 1. \quad (3.24b)$$

The weights ϕ_{jk} described in Eq. (3.8) will refer to the $\hat{V}$ function described in Eq. (3.24a). They are a function of conceptual memberships - any element of the M_{fc} space. To distinguish the weights in all the consecutive loops for the optimal prototypes function $\hat{V}$ described by Eq. (3.24b), we will use the notion of **optimal weights** $\hat{\phi}_{jk}$

$$\hat{\phi}_{jk} = \phi(\hat{u}_{jk}). \quad (3.25)$$

The use of the **hat** symbol in the notion of $\hat{\phi}_{jk}$ is meant to stress that these optimal weights are impacted by the partial supervision, in a way corresponding to the formulation of a specific semi-supervised fuzzy clustering model.

Summary. The functional impact of partial supervision on memberships in the class of SSFC considered in this thesis does not depend on the loop counter of the AO algorithm, but depends on the indices of the observation and cluster:

$$\pi_U(\alpha) = \pi_U(\alpha; j, k), \quad (3.26)$$

whereas the functional impact of partial supervision on prototypes depends on the indices of the observation and cluster, and a distinction whether the loop counter is equal to 1, or greater than it. We will define

$$\pi_V^1(\alpha) = \pi_V(\alpha; j, k, l = 1), \quad (3.27)$$

$$\pi_V^+(\alpha) = \pi_V(\alpha; j, k, l > 1). \quad (3.28)$$

To explain the impact of partial supervision on memberships, and define the functional form of $\pi_U(\alpha)$ from Eq. (3.26), one must analyze how the scaling factor α impacts the

individual optimal membership $\hat{u}_{jk}$. Analogously, to explain the impact of partial supervision on the prototypes, and define the functional form of $\pi_V^+(\alpha)$ from Eq. (3.28) and $\pi_V^-(\alpha)$ from Eq. (3.27), one must analyze how the scaling factor α impacts the individual optimal weight $\hat{\phi}_{jk}$, and how it differs from the unimpacted individual weight ϕ_{jk} .

In both cases, the consideration of indices $j = 1, \dots, N$, and $k = 1, \dots, c$ is not sufficient - the introduction of partial supervision introduces a differentiation between three types of optimal memberships, and weights, that we analyze in the following Section.

3.2.4 Supervised and non-supervised clusters

In the foundational model introducing the additive combination technique, discussed in Section 2.2.2, to distinguish between the supervised and unsupervised observations in the objective function J_{P97} defined in Eq. (2.41) a technical variable $b_j = \sum_{k=1}^c f_{jk}$ was used. While it is convenient for the formulation of objective function and associated derivation of optimal memberships and prototypes, this is not convenient for the proposed explanations of the impact of the scaling factor.

For the matters of explainability, to distinguish between the unsupervised and supervised observations, we will rather use two novel indices in addition to $j = 1, \dots, N$ that is indexing all the observations (regardless whether they are unsupervised, or supervised). Index $i = 1, \dots, M$ will denote supervised observations, and index $h = 1, \dots, H$ will refer to the unsupervised observations. In this case, $N = M + H$.

Moreover, for any supervised observation i we must now distinguish between its membership to the “supervised” cluster (i.e., the column in U corresponding to the label $y(\omega_i) \in \mathcal{Y}$ representing the class of the i th observation ω_i) and the remaining “non-supervised” cluster(s). To address this, we define a function

$$s(i) = \arg \max_k f_{ik} \quad (3.29)$$

which (in the binary supervision case assumed in this thesis) identifies the index of the cluster associated with the supervised class of the i th observation. For convenient notation, we also define a set of indices of non-supervised clusters as

$$\mathcal{E}_i = \{1, \dots, c\} \setminus \{s(i)\}. \quad (3.30)$$

In certain situations, we will also treat $\mathcal{E}_i$ as a $(c - 1)$ -tuple, to be able to refer to the specific indices of non-supervised clusters for a given i th observation. In such case, we will refer to the r th element of $\mathcal{E}_i$ using a notion $\mathcal{E}_{i,r}$.

A following simplified example shows a matrix $U_{3 \times 3}$, and an exemplary corresponding

supervision matrix F .

$$\begin{aligned}
 U &= \begin{matrix} & \begin{matrix} k=1 & k=2 & k=3 \end{matrix} \\ \begin{matrix} (i=1) \\ (h=1) \\ (i=2) \end{matrix} & \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} \end{matrix} \quad F = \begin{matrix} & \begin{matrix} k=1 & k=2 & k=3 \end{matrix} \\ \begin{matrix} (i=1) \\ (h=1) \\ (i=2) \end{matrix} & \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{matrix} \\
 & \hspace{25em} (3.31)
 \end{aligned}$$

In the example presented in Eq. (3.31), we have $M = 2$ supervised observations, and $H = 1$ unsupervised observation. The index of the “supervised cluster” for the first supervised observation $i = 1$ is $s(1) = 1$, and the index of the second supervised observation $i = 2$ is equal to $s(2) = 3$. The corresponding sets of non-supervised cluster indices are $\mathcal{E}_1 = \{2, 3\}$, and $\mathcal{E}_2 = \{1, 2\}$, correspondingly.

Having introduced the above notation, we may define the three types of optimal memberships to be considered in the semi-supervised scenario as:

- $\hat{u}_{hk}$: an optimal membership degree of an unsupervised observation h to any cluster,
- $\hat{u}_{i,k \neq s(i)}$: an optimal membership degree of a supervised observation i to any non-supervised cluster from the set $\mathcal{E}_i$,
- $\hat{u}_{i,k=s(i)}$: an optimal membership degree of a supervised observation i to the supervised cluster $s(i)$.

The study of the impact of partial supervision on the prototypes requires a mechanism of differentiating the impact of unsupervised and supervised observations similar to the three types of memberships $\hat{u}_{hk}$, $\hat{u}_{i,k \neq s(i)}$, $\hat{u}_{i,k=s(i)}$ proposed to be analyzed when considering the impact of partial supervision on the memberships. In light of the generic formula for the optimal prototypes in any SSFC model presented in Eq. (3.10), we propose to focus on the weights ϕ_{jk} , defining three types of such weights:

- ϕ_{hk} : a weight of the unsupervised observation h to any cluster k ,
- $\phi_{i,k \neq s(i)}$: a weight of the supervised observation i to any non-supervised cluster from set $\mathcal{E}_i$,
- $\phi_{i,k=s(i)}$: a weight of the supervised observation i to the supervised cluster $s(i)$.

The study of the above three different weights will be of particular importance for defining the function $\pi_V^1(\alpha)$. Analogously, we will consider three similar types of optimal weights $\hat{\phi}_{hk}$, $\hat{\phi}_{i,k \neq s(i)}$, and $\hat{\phi}_{i,k=s(i)}$ in respective analyses considering the function $\pi_V^+(\alpha)$.

3.2.5 Order of clusters and prototypes

A straightforward, but rarely discussed consequence of introducing partial supervision in form of a supervision matrix F which aims at guiding the values of the decision memberships $\tilde{U}$ pertains to the expected order of rows in the prototypes matrix $\tilde{V}$. To show that, let us consider the unsupervised case first.

The k th column of matrix U , denoted as $U_{:k}$, is interpreted as a certain fuzzy set, for which $\mathbf{v}_k$ is its prototype. The conceptual membership degree $u_k(\omega_j)$ is estimated by calculating the distance between j th observation $\mathbf{x}_j$, and k th cluster's prototype $\mathbf{v}_k$. In the unsupervised case, the order of the rows in the $\tilde{V}$ matrix – each row representing a separate cluster's prototype – does not matter, since nothing is assumed about the clusters a priori.

However, in the semi-supervised case, the supervision matrix F implements an arbitrary, but a fixed order - by associating k th column of matrix F with a k th label from the ordered c -tuple $\mathcal{Y} = (y^k)_{k=1,\dots,c}$. The guiding of memberships of the supervised observations towards the values of 0 and 1 (in case of binary supervision assumed in this thesis) introduced in the description of the partial supervision mechanism in Section 2.2.2 is achieved by introducing a penalty $\mathcal{L}(u_{jk}, f_{jk})$ into the supervised component Q^S of the objective function. A popular penalty introduced in both classical models analyzed in this thesis, Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means, is a quadratic penalty

$$\mathcal{L}(u_{jk}, f_{jk}) = (u_{jk} - f_{jk})^2. \quad (3.32)$$

Since the optimization takes place on the level of entire matrix U , the above equation can be represented column-wise as well, i.e.,

$$\mathcal{L}(U_{:k}, F_{:k}) = (U_{:k} - F_{:k})^2. \quad (3.33)$$

Clearly, the implementation of the quadratic (or any other) loss **introduces a clusters — classes mapping by associating $U_{:k}$, a distribution of memberships to the k th cluster, with the column $F_{:k}$ associated with a concrete label y^k** . The arbitrary order of columns in F becomes expected in U . In consequence, a given k th cluster's prototype $\mathbf{v}_k$, associated with $U_{:k}$, becomes associated in a transitive way with a label y^k encoding the k th class in the ordered set $\mathcal{Y}$,

It is worth noting that this specific mechanism used in semi-supervised fuzzy clustering is a particular realization of a generic assumption undertaken in Semi-Supervised Learning, that Chapelle et al. (2006) called *the cluster's assumption*, and defined it as “if points are in the same cluster, they are likely to be of the same class”. The explicit

reminder that in the semi-supervised case, the estimated k th cluster's prototype $\tilde{\mathbf{v}}_k$ is identified as the cluster of the observations belonging to the y^k class is important for the clarity of explanations provided in this field.

3.3 Explanations in Semi-Supervised Fuzzy c-Means

This Section provides explanations of impact of partial supervision in Semi-Supervised Fuzzy c-Means. While the preceding Section 3.2 was meant to provide the generic explainability framework for all SSFC models, some of the concepts contributing to this framework could not be easily introduced without the context of a specific semi-supervised fuzzy clustering model. Thus, some of the concepts, or contributions presented in this Section will be specific only to Semi-Supervised Fuzzy c-Means, but other – although presented for the first time in the specific context of SSFCMeans – will be applicable to the matters of explainability of partial supervision in the entire class of SSFC.

3.3.1 Semi-Supervised Fuzzy c-Means model

Recall that the original idea, underpinning the entire class of semi-supervised fuzzy clustering models, was to additively combine unsupervised Q and supervised Q^S objective functions' components into a single objective function $J = Q + \alpha \cdot Q^S$. In Semi-Supervised Fuzzy c-Means, the component Q refers to the Fuzzy c-Means objective function Q_{FCM} defined in Eq. (2.7), and the novel idea was to construct Q^S to appropriately account for partial supervision. Partial supervision is introduced by the means of a binary supervision matrix F , as described in Section 2.2.2. To further conveniently distinguish between supervised and unsupervised observations, we introduce an auxiliary binary variable $b_j = \sum_{k=1}^c f_{jk}$.

The idea of an objective function introducing such defined partial supervision was first presented by Pedrycz and Waletzky (1997), but we will use an altered form of the objective function, as introduced in (Kmita et al., 2024). For detailed description of the differences, see the discussion in (Kmita et al., 2024). This altered objective function presents as follows

$$J_{\text{SSFCM}} = \sum_{j=1}^N \sum_{k=1}^c u_{jk}^m d_{jk}^2 + \alpha \sum_{j=1}^N \sum_{k=1}^c b_j (u_{jk} - f_{jk})^m d_{jk}^2. \quad (3.34)$$

The optimization problem posed in Semi-Supervised Fuzzy c-Means is analogous to the Problem (2.14), with the only difference being the different objective function applied.

In the same manner, a relaxation trick is assumed to be able to optimize over independent rows of the matrix U , and apply the Langrange multipliers technique.

The respective Lagrangian is defined as

$$L(\mathbf{u}_k, \lambda) = \sum_{k=1}^c u_{jk}^m d_{jk}^2 + \alpha \sum_{k=1}^c b_j (u_{jk} - f_{jk})^m d_{jk}^2 - \lambda \left(\sum_{k=1}^c u_{jk} - 1 \right). \quad (3.35)$$

Since this is a convex optimization problem, the conditions for a minimum present as $\frac{\partial L}{\lambda} = 0, \frac{\partial L}{\mathbf{u}_k} = \mathbf{0}$.

For the first one,

$$\frac{\partial L}{\partial \lambda} = \sum_{k=1}^c u_{jk} - 1 = 0, \quad (3.36a)$$

$$\sum_{k=1}^c u_{jk} = 1, \quad (3.36b)$$

and for the second one

$$\frac{\partial L}{\partial u_{jk}} = m \cdot u_{jk}^{m-1} \cdot d_{jk}^2 + \alpha \cdot m \cdot b_j \cdot (u_{jk} - f_{jk})^{m-1} - \lambda = 0. \quad (3.37)$$

From Eq. (3.37), it can be clearly seen that “for any value of m different from 2 the optimization conditions call for some extra computational effort as now u_{jk} (and λ) are linked together in the form of a certain polynomial equation whose roots need to be determined numerically” (Pedrycz & Waletzky, 1997). Thus, as Pedrycz and Waletzky (1997) advocate, we will further consider only $m = 2$ for the semi-supervised fuzzy clustering models throughout this thesis. This practice has become a de facto standard in the literature. Furthermore, for easier presentation, let

$$\alpha_j = \alpha \cdot b_j. \quad (3.38)$$

Then, one can calculate u_{jk} as

$$2u_{jk}d_{jk}^2 + 2\alpha_j(u_{jk} - f_{jk})d_{jk}^2 = \lambda \quad (3.39a)$$

$$u_{jk} + \alpha_j u_{jk} d_{jk}^2 - \alpha_j f_{jk} d_{jk}^2 = \frac{\lambda}{2} \quad (3.39b)$$

$$u_{jk} d_{jk}^2 (1 + \alpha_j) = \frac{\lambda}{2} + \alpha_j f_{jk} d_{jk}^2 \quad (3.39c)$$

$$u_{jk} = \left(\frac{\lambda}{2} \cdot \frac{1}{d_{jk}^2} + \alpha_j f_{jk} \right) \left(\frac{1}{1 + \alpha_j} \right). \quad (3.39d)$$

Now, using the equivalence from Eq. (3.36b), formula from Eq. (3.39d), and the fact that

$$\alpha_j \cdot \sum_{g=1}^c f_{jg} = \alpha_j, \quad (3.40)$$

we have

$$\sum_{g=1}^c u_{jg} = \left(\frac{\lambda}{2} \cdot \sum_{g=1}^c \frac{1}{d_{jg}^2} + \alpha_j \sum_{g=1}^c f_{jg} \right) \left(\frac{1}{1 + \alpha_j} \right) = 1 \quad (3.41a)$$

$$\frac{\lambda}{2} \cdot \sum_{g=1}^c \frac{1}{d_{jg}^2} = 1 \cdot (1 + \alpha_j) - \alpha_j \quad (3.41b)$$

$$\frac{\lambda}{2} = \frac{1}{\sum_{g=1}^c \frac{1}{d_{jg}^2}} \quad (3.41c)$$

In the sequel, plugging Eq. (3.41c) into Eq. (3.39d), we have

$$u_{jk} = \frac{1}{\sum_{g=1}^c \frac{1}{d_{jg}^2}} \cdot \frac{1}{d_{jk}^2} \cdot \frac{1}{1 + \alpha_j} + \frac{\alpha_j f_{jk}}{1 + \alpha_j}. \quad (3.42)$$

We can now define the optimal membership $\hat{u}_{jk}$ in SSFCM as

$$\hat{u}_{jk} = \frac{1}{1 + \alpha b_j} \cdot \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}} + \frac{\alpha b_j}{1 + \alpha b_j} f_{jk}. \quad (3.43)$$

Let us also define the term

$$e_{jk} = \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}} \quad (3.44)$$

that we will call **data evidence** to ease the further notation, and comparisons between FCM and SSFCM. Using this notation, and the observation-specific scaling factor $\alpha_j = \alpha \cdot b_j$, we can represent optimal memberships $\hat{u}_{jk}^{\text{FCM}}$ from the Fuzzy c-Means model from Eq. (2.22), and optimal memberships $\hat{u}_{jk}^{\text{SSFCM}}$ from the Semi-Supervised Fuzzy c-Means model from Eq. (3.43) as

$$\hat{u}_{jk}^{\text{FCM}} = e_{jk}, \quad (3.45a)$$

$$\hat{u}_{jk}^{\text{SSFCM}} = \frac{1}{1 + \alpha_j} e_{jk} + \frac{\alpha_j}{1 + \alpha_j} f_{jk}. \quad (3.45b)$$

3.3.2 Impact on memberships in Semi-Supervised Fuzzy c-Means

Let us now present the formulae for the three distinct types of memberships in the semi-supervised scenario. First, consider an unsupervised observation h :

$$\hat{u}_{hk} = \frac{1 + \alpha(0 - 0)}{1 + \alpha \cdot 0} \cdot e_{hk} + \frac{\alpha}{1 + \alpha \cdot 0} \cdot 0 = e_{hk}. \quad (3.46)$$

For the unsupervised observation, there is no direct impact of partial supervision on the value of the membership. It depends only on the data evidence, just as in unsupervised Fuzzy c-Means.

Investigating i th supervised observation and its memberships, we first consider a degree of membership to any non-supervised cluster $k \neq s(i)$

$$\hat{u}_{i,k \neq s(i)} = \frac{1}{1 + \alpha} \cdot e_{i,k \neq s(i)}. \quad (3.47)$$

The data evidence $e_{i,k \neq s(i)}$ is decreased by the factor of $\frac{1}{1+\alpha}$. It is a desired result of the partial supervision mechanism. Even if the data evidence were to support the belonging of the i th observation to the $k \neq s(i)$ cluster, the additional information we possess would decrease this membership.

The equations above clarify the mechanism of the SSFCMeans, but it is the membership of the supervised observation i to the supervised cluster $s(i)$ that is of major interest

$$\hat{u}_{i,s(i)} = \frac{1}{1 + \alpha} \cdot e_{i,s(i)} + \frac{\alpha}{1 + \alpha}. \quad (3.48)$$

We can observe that it includes a data-invariant component $\frac{\alpha}{(1+\alpha)}$ that depends only on the value of the scaling factor.

Explanation 1. *In SSFCM, the scaling factor α sets a lower boundary $\frac{\alpha}{1+\alpha}$ for an optimal membership of a supervised observation to the supervised cluster, and sets an upper boundary for an optimal membership of a supervised observation to the non-supervised cluster at the level of $\frac{1}{1+\alpha}$.*

Semantics of supervised membership degrees

Due to the form of optimal memberships $\hat{U}$, the decision $\tilde{U}$ does not belong to the fuzzy c -partition space M_{fc} , but to its subset. Let us denote this space as $M_{\text{SSFCM}}(\alpha)$, and observe that

$$M_{\text{SSFCM}}(\alpha) \subset M_{fc} \quad \forall \alpha > 0. \quad (3.49)$$

It is up to the user of the model configuring the value of the scaling factor α how much $M_{\text{SSFCM}}(\alpha)$ “shrinks” as compared with M_{fc} , which is regulated by the value of the scaling factor α .

The above has serious consequences with respect to the semantics of the obtained clusters, interpreted as fuzzy sets. As Dubois and Prade (1997) stated, fuzzy sets have been a precise mathematical tool, but an ongoing debate has been on the actual semantics and meaning of the degree of membership. One of the three semantics described in (Dubois & Prade, 1997) is the degree of similarity, where the membership degree “is the degree of proximity of u to the prototype elements of A [a fuzzy set]”. This is the semantics applicable in the unsupervised fuzzy clustering, where a given fuzzy set A is equal to the k th column of $U - U_{:,k}$ – and the prototype of this fuzzy set is equal to $\mathbf{v}_k$. **In Semi-Supervised Fuzzy c-Means, this semantics no longer holds for the supervised observations $i = 1, \dots, M$, where it is a user of the model that can arbitrarily set the upper limit for the estimated membership degree $\tilde{u}_{i,k \neq s(i)}$, and simultaneously a lower bound for $\tilde{u}_{i,k=s(i)}$.** In consequence, the logic of the “similarity to the prototype” semantics is violated.

The above is a result of the optimization procedure common for both unsupervised and semi-supervised models. The $u_{jk} \in [0, 1]$ criterion of M_{fc} is not embedded into the optimization procedure, but it turns out to hold for the result of the optimization procedure in case of unsupervised Fuzzy c-Means, whereas it turns out not to hold for the result of the optimization procedure in case of Semi-Supervised Fuzzy c-Means – in a sense that there exists the inclusion relation stated in Eq. (3.49). Importantly, for other semi-supervised fuzzy clustering models, the fuzzy c -partition space spanned by the specific model may not even be a proper subset of M_{fc} , like in the case of SSFCMeans and M_{SSFCM} .

For example, Wu and Hou (2025) propose a following objective function

$$J_{W25}(U, V) = \beta \sum_{j=1}^N \sum_{k=1}^c u_{jk}^2 d_{jk}^2 + \alpha \sum_{j=1}^N \sum_{k=1}^c (u_{jk}(1 + b_j) - 3b_j f_{jk})^2 d_{jk}^2, \quad (3.50)$$

and derive the following optimal membership formula

$$\hat{u}_{jk}^{W25} = \frac{\beta + \alpha(1 + b_j)^2 - 3\alpha b_j(1 + b_j) \sum_{g=1}^c f_{jk}}{\beta + \alpha(1 + b_j)^2} \cdot \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}} + \frac{3\alpha b_j(1 + b_j)}{\beta + \alpha(1 + b_j)^2} f_{jk}. \quad (3.51)$$

Wu and Hou (2025) do not provide the analysis in terms of three distinct types of memberships, only stating that “without supervised information, $\hat{u}_{jk}^{W25}$ reduces to the standard FCM algorithm, making FCM a special case of the semi-supervised fuzzy clustering method presented”. Denoting that $\hat{u}_{hk}^{W25} = \hat{u}_{jk}^{FCM}$, we analyze the separate supervised memberships as

$$\hat{u}_{i,k \neq s(i)}^{W25} = \frac{\beta - 2\alpha}{\beta + 4\alpha} \cdot \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}}, \quad (3.52a)$$

$$\hat{u}_{i,k=s(i)}^{W25} = \frac{\beta - 2\alpha}{\beta + 4\alpha} \cdot \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}} + \frac{6\alpha}{\beta + 4\alpha} f_{jk}. \quad (3.52b)$$

From the above equations, if $\beta = 2\alpha$, then $\hat{u}_{i,k \neq s(i)}^{W25} = 0$, and $\hat{u}_{i,k=s(i)}^{W25} = 1$, and if $\beta < 2\alpha$, then $\hat{u}_{i,k \neq s(i)}^{W25} < 0$, and $\hat{u}_{i,k=s(i)}^{W25} > 1$! Therefore,

$$M_{W25}(\alpha, \beta) \not\subseteq M_{fc} \quad \forall \alpha > 0, \beta > 0, \quad (3.53)$$

which may cause even more problems with the semantics of the fuzzy sets defined by the decision of the model $\tilde{U}_{W25}$. It is worth noting that Wu and Hou (2025) do not provide any discussion or suggestions regarding the ranges of values of β and α to be used in their model.

Another example is an objective function proposed by Stutz and Runkler (2002)

$$J_{S02}(U, V) = (1 - \alpha) \sum_{k=1}^c \sum_{j=1}^N u_{jk}^2 d_{jk}^2 + \alpha \sum_{k=1}^c \sum_{j=1}^N (u_{jk} - f_{jk})^2 d_{jk}^2. \quad (3.54)$$

Once again, $\hat{u}_{hk}^{S02} = \hat{u}_{jk}^{FCM}$, and for the supervised observations,

$$\hat{u}_{i,k \neq s(i)}^{\text{S02}} = (1 - \alpha) \cdot \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}} \quad (3.55a)$$

$$\hat{u}_{i,k=s(i)}^{\text{S02}} = (1 - \alpha) \cdot \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}} + \alpha. \quad (3.55b)$$

As opposed to Wu and Hou (2025), Stutz and Runkler (2002) mention that within their model, one should only consider $\alpha \in [0, 1]$ to ensure

$$M_{\text{S02}}(\alpha) \subset M_{fc} \quad \text{if } \alpha \in [0, 1]. \quad (3.56)$$

Focus on prototypes in empirical analysis of the impact of partial supervision

Recall the statement from Section 1.1 that partial supervision impacts the outcomes of modeling as compared with no supervision. The problems with the semantics of the decision of the SSFC models $\tilde{U}$ described above lead to a conclusion that with respect to explaining the impact of partial supervision, it can be a better idea to focus on the second element of the decision of the SSFC model, $\tilde{V}$, when studying the empirical impact of partial supervision. This is justified by the following interpretation.

Interpretation 2. *In SSFC, optimal membership does not longer hold semantics of a similarity to a prototype of a fuzzy set, but it holds the interpretation of it as a weight for the squared distance - with partial supervision adjusting these weights for the supervised observations.*

Indeed, a comparison of $\tilde{V}_{\alpha>0}$ with $\tilde{V}_{\alpha=0}$ does not suffer from any semantical issues - it is a straightforward study of how the prototypes changed in result of adjusting the weights ϕ_{jk} .

The idea to consider the decision prototypes as a first class citizen is also justified by their importance in treating the SSFC as a classifier. As stated in Eq. (1.9), the prediction of the label for the given observation $\mathbf{x}$ is based on finding the optimal index k^* such that

$$k^* = \arg \min_{k \in \{1, \dots, c\}} d(\mathbf{x}, \tilde{\mathbf{v}}_k)$$

and predicting $\rho(\mathbf{x}) = y^{k^*}$, the k^* th label from the corresponding c -tuple $\mathcal{Y}$. In result, the analysis of the classification error of any SSFC is focused only on the decision prototypes, without the need to directly consider decision memberships.

Last but not least, with respect to the decision of the SSFCM model $\tilde{U}$, one can consider an alternate decision by calculating

$$\tilde{U}' = \hat{U}_{\text{FCM}}(\tilde{V}_{\text{SSFCM}}) = [\hat{u}_{jk}^{\text{FCM}}(\tilde{V}_{\text{SSFCM}})], \quad (3.57)$$

where $\hat{u}_{jk}^{\text{FCM}}$ is defined by Eq. (2.22). This way, one is assured $\tilde{U}' \in M_{fc}$, and there are no issues with the semantics of the decision memberships. The amount to which semi-supervised $\tilde{u}'_{jk}$ differs from the unsupervised $\tilde{u}_{jk}^{\text{FCM}}$ would not be due to the artificial upper or lower boundary set by the user of the model, but rather due to the ability of the j th observation to influence the shape of the estimated prototypes matrix $\tilde{V}$. The drawback is that formally, $\tilde{U}'$ will not be the output of the Alternating Optimization algorithm, and it is unclear whether convergence theory will apply to this decision memberships without a dedicated study on this matter.

To summarize, all the above arguments support the conclusion that the empirical impact of partial supervision in semi-supervised fuzzy clustering should be measured with respect to the decision prototypes $\tilde{V}$.

3.3.3 Impact on prototypes in Semi-Supervised Fuzzy c-Means

In case of Semi-Supervised Fuzzy c-Means as defined in Eq. (3.34), the weighing function presents as

$$\phi_{jk} = u_{jk}^2 + \alpha b_j (u_{jk} - f_{jk})^2, \quad (3.58)$$

and the three types of weights ϕ_{jk} present as follows:

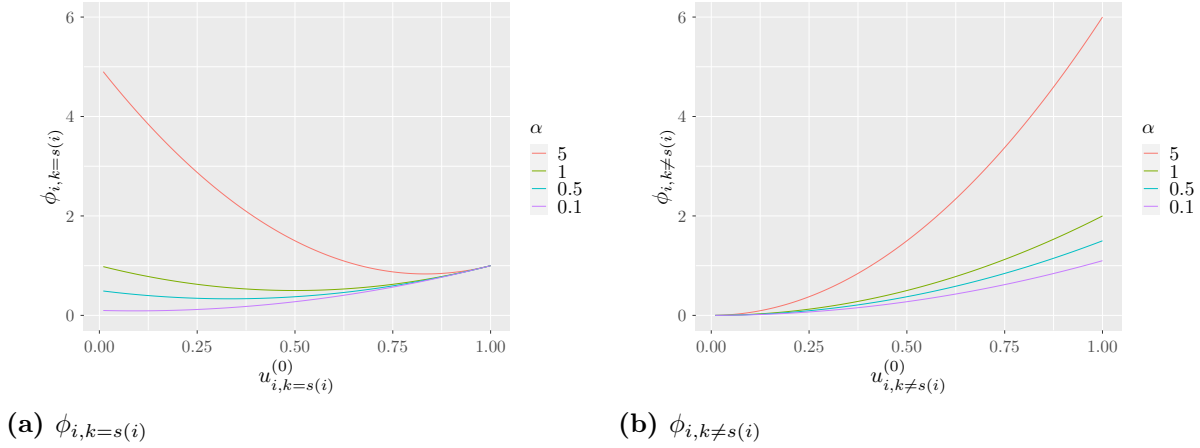
$$\phi_{hk} = u_{hk}^2, \quad (3.59a)$$

$$\phi_{i,k \neq s(i)} = u_{i,k \neq s(i)}^2 + \alpha (u_{i,k \neq s(i)} - 0)^2 = (1 + \alpha) u_{i,k \neq s(i)}^2, \quad (3.59b)$$

$$\phi_{i,k=s(i)} = u_{i,k=s(i)}^2 + \alpha (u_{i,k=s(i)} - 1)^2. \quad (3.59c)$$

For the unsupervised observations, the weight stays the same as in case of unsupervised Fuzzy c-Means, and is not impacted by partial supervision. For the supervised observations, the weight of the supervised observation in the supervised cluster is increased as compared with the unsupervised case, which is expected. However, the weight of the supervised observation in the non-supervised cluster is also increased, which is against the intuition, as one would rather expect to decrease this weight, since it is contradicting the partial supervision. The amount of increasing is presented in Fig. 3.3. Fig. 3.3a presents

Figure 3.3: Weighing functions for weights of supervised observations in non-supervised and supervised cluster in Semi-Supervised Fuzzy c-Means model.



Source: own work.

functional form of $\phi_{i,k=s(i)}$ for $\alpha \in \{0.1, 0.5, 1, 5\}$, and Fig. 3.3b presents functional form of $\phi_{i,k \neq s(i)}$ for the same α values.

The optimal weights $\hat{\phi}_{jk}$ for Semi-Supervised Fuzzy c-Means, based on the formulae for optimal memberships presented in Eq. (3.46), Eq. (3.47), Eq. (3.48), are calculated as follows

$$\hat{\phi}_{hk} = \hat{u}_{hk}^2 = e_{hk}^2, \quad (3.60a)$$

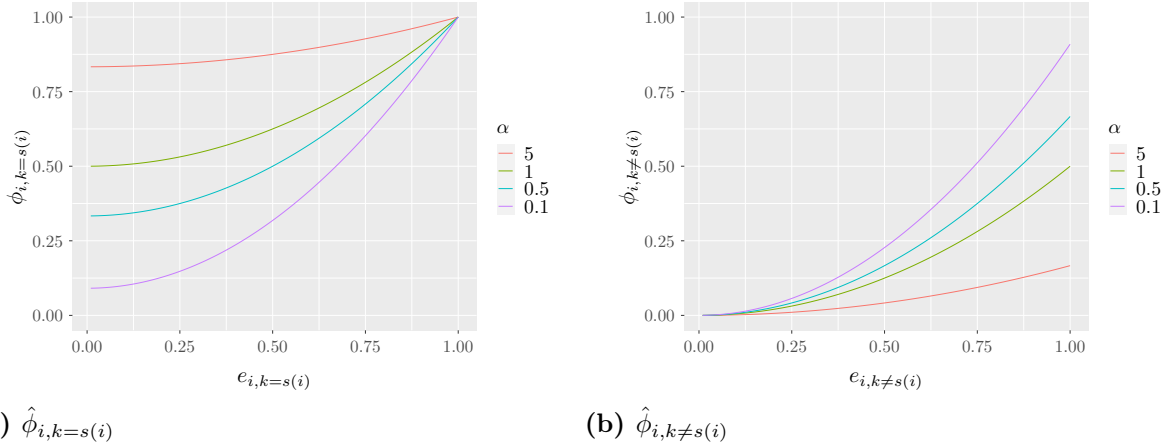
$$\hat{\phi}_{i,k \neq s(i)} = (1 + \alpha) \hat{u}_{i,k \neq s(i)}^2 = \frac{e_{i,k \neq s(i)}^2}{(1 + \alpha)}, \quad (3.60b)$$

$$\begin{aligned} \hat{\phi}_{i,k=s(i)} &= \hat{u}_{i,k=s(i)}^2 + \alpha (\hat{u}_{i,k=s(i)} - 1)^2 \\ &= \left(\frac{e_{i,k=s(i)} + \alpha}{1 + \alpha} \right)^2 + \alpha \cdot \left(\frac{e_{i,k=s(i)} + \alpha}{1 + \alpha} - 1 \right)^2, \end{aligned} \quad (3.60c)$$

where, recall, data evidence e_{jk} is defined in Eq. (3.44).

The nature of the impact of partial supervision on the prototypes is more intuitive in case of optimal weights. For the unsupervised observations, they remain unimpacted by the partial supervision. For the supervised observations, and their weight in the non-supervised cluster $k \neq s(i)$, the weighing function $\hat{\phi}_{i,k \neq s(i)}$ decreases their contribution to the final shape of $\tilde{V}$. For the supervised observations and their weight in the supervised cluster, the weighing function $\hat{\phi}_{i,k=s(i)}$ increases their contribution. Fig. 3.4a presents functional form of $\hat{\phi}_{i,k=s(i)}$ for $\alpha \in \{0.1, 0.5, 1, 5\}$, and Fig. 3.4b presents functional form of $\hat{\phi}_{i,k \neq s(i)}$ for the same α values. Clearly, the images of both of the functions are bounded,

Figure 3.4: Optimal weighing functions for weights of supervised observations in non-supervised and supervised cluster in Semi-Supervised Fuzzy c-Means model.



Source: own work. Note that the argument of the $\hat{\phi}$ function is the data evidence e_{ik} .

and they are equal to $(0, 1)$.

Having presented the above, we can now formulate an appropriate explanation:

Explanation 2. *The impact of partial supervision on prototypes in SSFCMeans is expressed by weighing function ϕ , which weighs the contribution of a given observation $\mathbf{x}_j$ to the k th cluster's prototype $\mathbf{v}_k$ as described in (3.10). The corresponding function π_V is composite, depending on the observation index j , the cluster index k , and the loop l of the optimization algorithm. Specifically, $\pi_V^1(\alpha; j, k)$ is described by equations (3.59a), (3.59b), (3.59c), and $\pi_V^+(\alpha; j, k)$ is described by equations (3.60a), (3.60b), (3.60c).*

This explanation provides a foundation for the analysis of how to construct α grids in Semi-Supervised Fuzzy c-Means in particular, and in semi-supervised fuzzy clustering class of models in general. The observation that in the first loop $l = 1$, $\pi_V^1(\alpha)$ is unbounded leads to thinking about any potential untoward consequences of using a huge value of α , e.g. $1e10$, that will be called as **overimpact of partial supervision** and analyzed in Section 5.2.

Seemingly unsupervised $\hat{V}$

The provided explanation also sheds new light on such semi-supervised fuzzy clustering models where it might seem that partial supervision is impacting only memberships, but not the prototypes. The first of such models is the original formulation of Semi-Supervised Fuzzy c-Means by Pedrycz and Waletzky (1997), that – for clarity – we will refer to as to P97-SSFCM. Pedrycz and Waletzky (1997) optimized $J_{P97}(U; V)$ to obtain $\hat{U}_{P97}$, but they

did not optimize $J_{P97}(V; U)$ - instead, they used a formula corresponding to the optimal $\hat{V}_{FCM}$:

$$\hat{\mathbf{v}}_k^{P97} = \frac{\sum_{j=1}^N u_{jk}^2 \cdot \mathbf{x}_j}{\sum_{j=1}^N u_{jk}^2}. \quad (3.61)$$

Analyzing Eq. (3.61), one may conclude that partial supervision, regulated by the scaling factor α , does not impact the prototypes in this P97-SSFCM model - since α is not a part of the equation. However, the proposed explainability framework requires to distinguish between the impact of partial supervision in the first loop - $\pi_V^1(\alpha)$ from Eq. (3.27) - and further loops - $\pi_V^+(\alpha)$ from Eq. (3.28) - of the algorithm. Indeed, in case of $l = 1$, there is no impact of partial supervision on the prototypes in P97-SSFCM model, but in all the consecutive loops, the corresponding optimal weights present as follows:

$$\hat{\phi}_{hk} = \hat{u}_{hk}^2 = e_{hk}^2, \quad (3.62a)$$

$$\hat{\phi}_{i,k \neq s(i)} = \hat{u}_{i,k \neq s(i)}^2 = \frac{e_{i,k \neq s(i)}^2}{(1 + \alpha)^2}, \quad (3.62b)$$

$$\hat{\phi}_{i,k=s(i)} = \hat{u}_{i,k=s(i)}^2 = \frac{e_{i,k=s(i)}^2 + \alpha^2}{(1 + \alpha)^2}, \quad (3.62c)$$

and are clearly impacted by the scaling factor α regulating the impact of partial supervision. Obviously, $\hat{V}_{P97}$ is not the stationary point of J_{P97} , but this approach was proposed in the original work by Pedrycz and Waletzky (1997), and is sometimes undertaken by authors of modern works referring to their approach, e.g., by Casalino et al. (2019).

The above remarks also apply to these semi-supervised fuzzy clustering models that, through the use of weighing function ϕ and penalty function p as defined in generic model formulation $G(U, V)$ in Eq. (3.7), properly optimize both partial functions, but still arrive at the similar formula for optimal prototypes as in the unsupervised case. In particular, this is the case for semi-supervised models adapting the baseline unsupervised models that introduce the idea of entropy-based penalization. For example, Salehi et al. (2021) proposed a model called Semi-supervised Multiple Kernel Fuzzy Clustering based on Entropy and Relative entropy (SMKFC-ER), with a following objective function

$$J_{S21}(U, V) = \sum_{j=1}^N \sum_{k=1}^c u_{jk} d_{jk}^2 + \eta \sum_{j=1}^N \sum_{k=1}^c u_{jk} \log(u_{jk}) + \alpha \sum_{j=1}^N \sum_{k=1}^c b_j \cdot u_{jk} \log\left(\frac{u_{jk}}{f_{jk}}\right), \quad (3.63)$$

and the optimal memberships and prototypes present as follows

$$\hat{u}_{jk} = \frac{\exp\left((\alpha \cdot b_j \cdot \log(f_{jk}) - d_{jk}^2) \cdot (\eta + \alpha \cdot b_j)^{-1}\right)}{\sum_{g=1}^c \exp\left((\alpha \cdot b_j \cdot \log(f_{jg}) - d_{jg}^2) \cdot (\eta + \alpha \cdot b_j)^{-1}\right)}, \quad (3.64a)$$

$$\hat{\mathbf{v}}_k = \frac{\sum_{j=1}^N u_{jk} \cdot \mathbf{x}_j}{\sum_{j=1}^N u_{jk}}. \quad (3.64b)$$

Similarly to the case of P97-SSFCM model, it might seem that in SMKFC-ER, there is no direct impact of partial supervision on the prototypes, but that also holds only for the first loop of the AO algorithm, with optimal weights in the following loops being equal to

$$\hat{\phi}_{hk} = \hat{u}_{hk} = \frac{\exp(-d_{jk}^2 \cdot \eta^{-1})}{\sum_{g=1}^c \exp(-d_{jg}^2 \cdot \eta^{-1})}, \quad (3.65a)$$

$$\hat{\phi}_{i,k \neq s(i)} = \hat{u}_{i,k \neq s(i)} = \frac{\exp(-d_{jk}^2 \cdot (\eta + \alpha)^{-1})}{\sum_{g=1}^c \exp(-d_{jg}^2 \cdot (\eta + \alpha)^{-1})}, \quad (3.65b)$$

$$\hat{\phi}_{i,k=s(i)} = \hat{u}_{i,k=s(i)} = \frac{\exp((\alpha - d_{jk}^2) \cdot (\eta + \alpha)^{-1})}{\sum_{g=1}^c \exp((\alpha - d_{jg}^2) \cdot (\eta + \alpha)^{-1})}. \quad (3.65c)$$

clearly being impacted by the partial supervision, and the value of the scaling factor α .

3.4 Explanations in Semi-Supervised Possibilistic c-Means

3.4.1 Semi-Supervised Possibilistic c-Means model

We now apply the additive combination technique to introduce partial supervision to the Possibilistic c-Means model, as defined in Eq. (2.24), in a similar way as when creating Semi-Supervised Fuzzy c-Means objective function in the preceding Section. Referring to the discussion presented in Section 2.1.3, we will use the notation t_{jk} , instead of u_{jk} , to stress the interpretation of the degrees of membership as typicalities.

Let the Semi-Supervised Possibilistic c-Means objective function be defined as

$$J_{\text{SSPCM}} = \sum_{j=1}^N \sum_{k=1}^c t_{jk}^2 d_{jk}^2 + \sum_{j=1}^N \sum_{k=1}^c \gamma_k (1 - t_{jk})^2 + \alpha \sum_{j=1}^N \sum_{k=1}^c b_j (t_{jk} - f_{jk})^2 d_{jk}^2. \quad (3.66)$$

The optimization problem posed in Semi-Supervised Possibilistic c-Means is analogous to

the Problem (2.25), with the only difference being the different objective function applied.

Since in Possibilistic c-Means there is no *sum-to-one* constraint, to find optimal typicality $\hat{t}_{jk}$, we calculate the corresponding derivative

$$\frac{\partial J}{\partial t_{jk}} = 2t_{jk}d_{jk}^2 - 2\gamma_k(1 - t_{jk}) + 2\alpha b_j d_{jk}^2(t_{jk} - f_{jk}) \quad (3.67)$$

and set it to 0, obtaining

$$2t_{jk}d_{jk}^2 - 2\gamma_k(1 - t_{jk}) + 2\alpha b_j d_{jk}^2(t_{jk} - f_{jk}) = 0 \quad (3.68a)$$

$$t_{jk}d_{jk}^2 - \gamma_k + \gamma_k t_{jk} + \alpha b_j d_{jk}^2 t_{jk} - \alpha b_j d_{jk}^2 f_{jk} = 0 \quad (3.68b)$$

$$t_{jk}(d_{jk}^2 + \gamma_k + \alpha b_j d_{jk}^2) = \gamma_k + \alpha b_j d_{jk}^2 f_{jk} \quad (3.68c)$$

$$t_{jk}(\gamma_k + (1 + \alpha b_j)d_{jk}^2) = \gamma_k + \alpha b_j d_{jk}^2 f_{jk}. \quad (3.68d)$$

The optimal typicality presents as follows

$$\hat{t}_{jk} = \frac{\gamma_k + \alpha b_j f_{jk} d_{jk}^2}{\gamma_k + (1 + \alpha b_j)d_{jk}^2}. \quad (3.69)$$

3.4.2 Impact on typicalities in Semi-Supervised Possibilistic c-Means

Let us now present the formulae for three distinct types of typicalities in the semi supervised scenario:

$$\hat{t}_{hk} = \frac{\gamma_k}{\gamma_k + d_{hk}^2}, \quad (3.70a)$$

$$\hat{t}_{i,k \neq s(i)} = \frac{\gamma_{k \neq s(i)}}{\gamma_{k \neq s(i)} + (\alpha + 1) \cdot d_{i,k \neq s(i)}^2}, \quad (3.70b)$$

$$\hat{t}_{i,s(i)} = \frac{\gamma_{s(i)} + \alpha \cdot d_{i,s(i)}^2}{\gamma_{s(i)} + (\alpha + 1) \cdot d_{i,s(i)}^2}. \quad (3.70c)$$

Firstly, from Eq. (3.70a), we can see that there is no direct impact of partial supervision on the value of the typicality of any unsupervised observation to any cluster.

Secondly, the analysis of formulae for optimal typicalities of supervised observations is not as straightforward as in the Semi-Supervised Fuzzy c-Means case, since these are not linear transformations of the corresponding unsupervised typicalities. To interpret Eq. (3.70b) and Eq. (3.70c), we need to refer to the explanation of the role of the hy-

perparameter γ_k provided in the original work introducing the unsupervised Possibilistic c-Means model (Krishnapuram & Keller, 1993).

Explanation 3. *The value of γ_k determines the distance at which the typicality value of a point in a cluster becomes 0.5.*

The authors call this value the **3db point**. It comes from the fact that if we consider a distance $d_{hk}^2 := \gamma_k$, then for the typicality of unsupervised observation to any cluster Eq. (3.70a)

$$\hat{t}_{hk} = \frac{\gamma_k}{\gamma_k + d_{hk}^2} \stackrel{d_{hk}^2 := \gamma_k}{=} \frac{\gamma_k}{\gamma_k + \gamma_k} = 0.5. \quad (3.71)$$

With the aim of providing an explanation of the scaling factor α in Semi-Supervised Possibilistic C-Means, we will study the difference between:

- a typicality of a supervised observation to the supervised cluster $\hat{t}_{i,s(i)}$ from (3.70c) and a typicality of unsupervised observation to any cluster $\hat{t}_{hk}$ from (3.70a),
- a typicality of a supervised observation to the non-supervised cluster $\hat{t}_{i,k \neq s(i)}$ from (3.70b) and a typicality of unsupervised observation to any cluster $\hat{t}_{hk}$ from (3.70a).

Let us consider arbitrary observation a and arbitrary cluster b . First, assume $t_{ab}^{(I)}$ is unsupervised typicality to any cluster, as in Eq. (3.70a). We know that if we set $\gamma_b := d_{ab}^2$, then the value of typicality $t_{ab}^{(I)} = \frac{1}{2}$.

Suppose that we obtain the label of the observation a so it becomes supervised, and $b = s(a)$ happens to be the supervised cluster. Then, the typicality value takes the form from Eq. (3.70c) - we will mark it as $t_{ab}^{(II)}$. We have

$$t_{ab}^{(II)} \stackrel{d_{ab}^2 := \gamma_b}{=} \frac{\gamma_b + \alpha \gamma_b}{\gamma_b + (1 + \alpha) \gamma_b} = \frac{1 + \alpha}{2 + \alpha}. \quad (3.72)$$

Now, suppose that the label of observation a is not equal to the cluster b , i.e., $s(a) \neq b$. Then, the typicality value takes the form from Eq. (3.70b) - we will mark it as $t_{ab}^{(III)}$. We have

$$t_{ab}^{(III)} \stackrel{d_{ab}^2 := \gamma_b}{=} \frac{\gamma_b}{\gamma_b + (1 + \alpha) \gamma_b} = \frac{1}{2 + \alpha}. \quad (3.73)$$

To explain the impact of partial supervision on the typicalities in Semi-Supervised Possibilistic c-Means, we need to calculate the differences between

- $t_{ab}^{(II)}$ and $t_{ab}^{(I)}$,
- $t_{ab}^{(III)}$ and $t_{ab}^{(I)}$.

Note that the only change in the above scenario includes the value of the typicality: in each case, this is the same observation a , and the same cluster b , the same hyperparameter γ_b and the same fixed distance $d_{ab}^2 = \gamma_b$. Therefore, we can quantify the impact of partial supervision as:

$$t_{ab}^{(II)} - t_{ab}^{(I)} = \frac{1 + \alpha}{2 + \alpha} - \frac{1}{2} = \frac{2(1 + \alpha)}{2(2 + \alpha)} - \frac{2 + \alpha}{2(2 + \alpha)} = \frac{\alpha}{2(2 + \alpha)} \quad (3.74)$$

in the case of the typicality of the supervised observation to the supervised cluster, and

$$t_{ab}^{(III)} - t_{ab}^{(I)} = \frac{1}{2 + \alpha} - \frac{1}{2} = \frac{2}{2(2 + \alpha)} - \frac{2 + \alpha}{2(2 + \alpha)} = -\frac{\alpha}{2(2 + \alpha)}. \quad (3.75)$$

in the case of the typicality of the supervised observation to the non-supervised cluster. Based on the above equations, we can propose a following explanation.

Explanation 4. *In the supervised case, the scaling factor α increases the typicality of a supervised observation to the supervised cluster $\hat{t}_{i,s(i)}$ by $\frac{\alpha}{2(2+\alpha)}$ and decreases the typicality of a supervised observation to the non-supervised cluster $\hat{t}_{i,k \neq s(i)}$ by $\frac{\alpha}{2(2+\alpha)}$ for the same distance at which the typicality in the unsupervised case would be equal 0.5.*

3.4.3 Impact on prototypes in Semi-Supervised Possibilistic c-Means

In case of Semi-Supervised Possibilistic c-Means as defined in Eq. (3.66), the weighing function presents as

$$\phi_{jk} = t_{jk}^2 + \alpha b_j (t_{jk} - f_{jk})^2, \quad (3.76)$$

and the three types of weights present analogously to the corresponding weights in the Semi-Supervised Fuzzy c-Means case as

$$\phi_{hk} = t_{hk}^2, \quad (3.77a)$$

$$\phi_{i,k \neq s(i)} = t_{i,k \neq s(i)}^2 + \alpha (t_{i,k \neq s(i)} - 0)^2 = (1 + \alpha) t_{i,k \neq s(i)}^2, \quad (3.77b)$$

$$\phi_{i,k=s(i)} = t_{i,k=s(i)}^2 + \alpha (t_{i,k=s(i)} - 1)^2. \quad (3.77c)$$

The interpretation of these weights is the same as in the case of SSFCMeans, as described in Section 3.3.3. The difference between SSPCMeans and SSFCMeans lies in the form of optimal weights $\hat{\phi}_{jk}$, as these are based on optimal typicalities presented in Eq. (3.70a), Eq. (3.70b), Eq. (3.70c), and are calculated as follows

$$\hat{\phi}_{hk} = \hat{t}_{hk}^2 = \left(\frac{\gamma_k}{\gamma_k + d_{hk}^2} \right)^2, \quad (3.78a)$$

$$\hat{\phi}_{i,k \neq s(i)} = (1 + \alpha) \cdot \hat{t}_{i,k \neq s(i)}^2 = (1 + \alpha) \cdot \left(\frac{\gamma_{k \neq s(i)}}{\gamma_{k \neq s(i)} + (1 + \alpha) \cdot d_{i,k \neq s(i)}^2} \right)^2, \quad (3.78b)$$

$$\begin{aligned} \hat{\phi}_{i,k=s(i)} &= \hat{t}_{i,k=s(i)}^2 + \alpha(\hat{t}_{i,k=s(i)} - 1)^2 \\ &= \left(\frac{\gamma_{s(i)} + \alpha \cdot d_{i,s(i)}^2}{\gamma_{s(i)} + (\alpha + 1) \cdot d_{i,s(i)}^2} \right)^2 + \alpha \left(\frac{\gamma_{s(i)} + \alpha \cdot d_{i,s(i)}^2}{\gamma_{s(i)} + (\alpha + 1) \cdot d_{i,s(i)}^2} - 1 \right)^2 \\ &= \left(\frac{\gamma_{s(i)} + \alpha \cdot d_{i,s(i)}^2}{\gamma_{s(i)} + (\alpha + 1) \cdot d_{i,s(i)}^2} \right)^2 + \alpha \left(\frac{d_{i,s(i)}^2}{\gamma_{s(i)} + (\alpha + 1) \cdot d_{i,s(i)}^2} \right)^2. \end{aligned} \quad (3.78c)$$

The form of the optimal weights $\hat{\phi}_{jk}$ presented in Eq. (3.78) is much more complicated and difficult to interpret than in the case of Semi-Supervised Fuzzy c-Means and corresponding optimal weights formulated in Eq. (3.60). While the latter are functions of only the scaling factor α and the bounded data evidence $e_{jk} \in (0, 1)$, the former are functions of the scaling factor α , the unbounded hyperparameter $\gamma > 0$, and potentially unbounded distances $d_{jk} > 0$. In consequence, there does not exist a concise way to visualize functions defined in Eq. (3.78b) and Eq. (3.78c) in a similar way as optimal weights for Semi-Supervised Fuzzy c-Means were presented in Fig. 3.4a and Fig. 3.4b.

The formal explanation of the impact of partial supervision on the prototypes in Semi-Supervised Possibilistic c-Means is formulated as follows.

Explanation 5. *The impact of partial supervision on prototypes in SSPCMean is expressed by weighing function ϕ , which weighs the contribution of a given observation $\mathbf{x}_j$ to the k th cluster's prototype $\mathbf{v}_k$ as described in (3.10). The corresponding function π_V is composite, depending on the observation index j , the cluster index k , and the loop l of the optimization algorithm. Specifically, $\pi_V^1(\alpha; j, k)$ is described by equations (3.77a), (3.77b), (3.77c), and $\pi_V^+(\alpha; j, k)$ is described by equations (3.78a), (3.78b), (3.78c).*

Chapter 4

Software implementation

The proposed explainability framework introduced in the previous Chapter enables a broader analysis of the empirical outcomes of semi-supervised fuzzy clustering models. In particular, a suggested focus on the decision prototypes requires the ability to extract the weights of each observation to each cluster, from each iteration of the Alternating Optimization procedure described in Algorithm 3.

In this Chapter, we propose a novel implementation of four classical fuzzy clustering models: Fuzzy c-Means, Possibilistic c-Means, Semi-Supervised Fuzzy c-Means, and Semi-Supervised Possibilistic c-Means in the R language to enable full investigation of the models in the context of the explainability framework. The proposed implementation is inspired by the novel, generic objective function G presented in Eq. (3.7) that represents any (unsupervised or semi-supervised) model from the class of fuzzy clustering models. This approach serves as a basis for the derived matrix representation of the optimal memberships $\hat{U}$ and optimal prototypes $\hat{V}$ - so far, in the literature the optimal memberships have been presented only on the individual level of optimal membership of j th observation to the k th cluster $\hat{u}_{jk}$, and optimal prototypes have been presented only on the level of individual k th prototype's vector $\hat{\mathbf{v}}_k$; this convention has been used in the preceding Chapter 3 of this thesis as well, as it enables the theoretical analysis of the corresponding models. The importance of representing optimal memberships and prototypes directly in a different, matrix form lies in the ability to efficiently implement corresponding vectorised operations in the R language.

Importantly, the novel proposed implementation adheres to the principles of reproducibility in computational sciences, postulated as a desired standard in this field. Its transparent design allows easy extension for any model that can be classified as reusing the additive combination technique described in Section 2.2.2. We refer the discussion on the matter of reproducibility and replicability in the field of computational sciences,

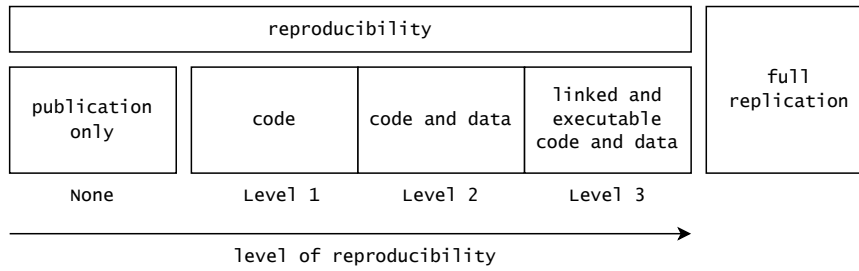
and investigate the related works on the topic of semi-supervised fuzzy clustering from the perspective of whether they enable the reuse of the proposed models in line with the principle of reproducibility and replicability.

4.1 Related work on replicability and reproducibility

The discussion on the issue of replication has been traditionally associated with experimental sciences, where data is collected about observational units, and further analyzed to support or prove some scientific claims. Open Science Collaboration (2015) define *direct replication* as “the attempt to recreate the conditions believed sufficient for obtaining a previously observed finding”. In their work, they replicated a hundred studies conducted in the field of psychology, and found out that many replications resulted in a weaker evidence for the findings included in the original publications.

Direct replication of such nature is, in general, a difficult task. Peng (2011) provide an example of epidemiology sciences, where the effect of environmental pollution is studied on large cohorts of individuals. Replications of such studies would be very costly due to significant time they take, and could be even impossible due to the changing policy decisions. Yet another difficulty applicable for modern studies in various fields is the increasing amount of digital data collected. The discussion around replication as a scientific method, when considered on the grounds of computational sciences, requires a precise setting of the definitions and expectations. In light of increasing difficulties with expecting to replicate the results of studies, the meaning of the reproducibility increased, which is understood by Peng (2011) as a minimal standard for evaluating the credibility of scientific assertions, especially in situations where conducting a fully independent replication is impractical. Peng (2011) stress that the major difference between being able to reproduce or replicate the results of a study lies in the data used: the replication would mean independent data collection, while reproduction means arriving at the same results given availability of the same input data. In their approach, Peng (2011) note that scientific publications vary in the degree of reproducibility provided, and thus they propose to consider a certain spectrum of reproducibility, with clearly distinguished levels, which is visually depicted in Fig. 4.1.

No reproducibility means that certain claims within a given study are published with no associated artifacts. The first Level 1 of reproducibility per (Peng, 2011) is for the code used to analyze the data to be shared. The next level on the reproducibility scale, Level 2, is to share the code and the data that the code was fed with, whereas the greatest reproducibility is attained when the code and data are shared in a linked and

Figure 4.1: Reproducibility spectrum.

Source: own work based on Peng (2011).

executable manner (Level 3). The need for such reproducible research is stressed by various publishers, including IEEE, who in their *IEEE Open Research Statement* say that “IEEE strongly supports the sharing of research processes, methods, data, code, and findings as openly as possible to promote collaboration, accelerate innovation, and advance scientific knowledge”¹.

Problems with replications, and reproducibility, are known to affect the fields of science such as already mentioned psychology, or medicine (Ioannidis, 2005), or politics (Janz, 2016), where measurements are collected on the observational units. As Rougier et al. (2017) argue, it is less obvious to see that similar problems affect the field of computer science, or more broadly *computational sciences*, where scientific research relies on the use of computer and relevant analysis software. Although it relies on code and data, and not on collecting and measuring observational units during a certain experiment, replicability and reproducibility are as well important challenges in this field. However, a clear presentation on these concepts on the ground of computational sciences requires precise definitions that may differ from the nomenclature assumed on the ground of other fields of science.

Rougier et al. (2017) state that “the goal of a reproduction attempt is to verify that the computational protocol leading to the results has been recorded correctly”, and as such, reproducing a result can be understood as executing identical software using the same input data and obtaining matching outputs. Notably, Rougier et al. (2017) do not provide any grading of reproducibility, as Peng (2011) did - in the light of their approach, a certain protocol is either reproducible or non-reproducible. The matter of reproducibility is thus the matter of willingness of the authors to publish all the respective supplemen-

¹<https://journals.ieeeauthorcenter.ieee.org/create-your-ieee-journal-article/research-reproducibility/> (accessed on 31 January 2026).

tary materials in an appropriate form, along with the traditional article describing their achievements.

However, while the absence of supplementary material means no reproducibility for sure, mere existence of appropriate supplementary materials does not guarantee successful reproducibility. It was found in the meta-review by (Collberg & Proebsting, 2016) that out of 512 potentially reproducible studies they had selected based on the literature review from the field of computational sciences, they were able to successfully reproduce only 102 studies, due to the issues with the quality of the source materials. A way to overcome such problems was a primary motivation for the creators of the **ReScience C** peer-reviewed journal, that is focused on replication in scientific computation², as described by Rougier et al. (2017).

To distinguish it from reproducibility, Rougier et al. (2017) propose that replication of a published result includes independently implementing new software using the methodological description given in the original work, executing it, and producing results that are sufficiently similar to be regarded as equivalent, where it is a subjective matter to define the degree of *similarity* that is deemed satisfactory for claiming the replication of the results. In this sense of an independent implementation of a given model, and a verification of the equivalence of the results obtained with the same input data, replication is a stronger quality indicator than reproducibility. Within this logic, one can replicate results of either reproducible, or non-reproducible original results - but the replication itself should adhere to the standards of reproducibility. As Rougier et al. (2017) stated, “the holy grail of computational science is a reproducible replication of reproducible original work”.

The term *reproducibility* appears also in the description of *Bioconductor*, a popular *initiative for the collaborative creation of extensible software for computational biology and bioinformatic* (Gentleman et al., 2004). Gentleman et al. (2004) list *pursuit of reproducibility* as one of the three major motivations behind creating *Bioconductor*, together with the need for *transparency* and *efficiency of development*. Out of these three concepts, it is the matter of *efficiency of development* that is of special interest to us, as it has not been discussed in either (Peng, 2011) or (Rougier et al., 2017). In particular, it is requested that “software and data resources in an open-source environment can be read by interested investigators, and can be modified and extended to achieve new functionalities” (Gentleman et al., 2004). This requires the source code, and accompanying documentation protocols, to be in a form that allows easy way of extending to foster further research, which is not something that is postulated by Rougier et al. (2017) or Peng (2011) as a

²<https://rescience.github.io/> (accessed on 31 January 2026)

standalone requirement in their discussion about reproducibility and replicability.

Similar challenges to these listed by the creators of the *Bioconductor* project have been also identified in the field of pharmaceutical research, where recently the **openstatsware** group was formed, formally a scientific working group in the American Statistical Association (ASA) Biopharmaceutical Section (BIOP)³, and a European Special Interest Group (SIG) sponsored by Statisticians in the Pharmaceutical Industry (PSI)⁴ and the European Federation of Statisticians in the Pharmaceutical Industry (EFSPI)⁵. In one of the presentations⁶, **openstatsware** group describes the situation within the field of biostatistics, where for decades, the proprietary (closed) software solutions provided by commercial companies were used to analyze data, and the development of open source statistical software has been only ongoing since approximately two decades ago - gaining on significance in the last few years. It is stressed that, despite the advantages of this shift to the open source solutions, “users face huge variability in software quality (reliability, efficiency and maintainability)”, that **openstatsware** is aiming to diminish. Even though the nomenclature used in the associated presentation includes terms such as *reliability* and *efficiency* without further detailed definitions, it can be clearly stated that these considerations refer to those already mentioned in this Section.

It is worth stressing that the developments in the infrastructure available to the scientific society since (Peng, 2011) was published make it easy for any researcher to achieve each level of reproducibility described in Fig. 4.1 if they are only willing to. Level 1, code sharing, can be achieved by using highly popular open code repositories, such as e.g. GitHub⁷. Data sharing, with appropriate licensing and necessary formalism, has been enabled by platforms such as IEEE DataPort⁸. Simulated data can be effectively shared via appropriate code sharing as well, by sharing simulation code and the seed used for pseudo-random numbers generation. Finally, achieving Level 3 – linked and executable code and data – can be achieved by using services like CodeOcean that provide the ability to create *Compute Capsules* - “a shareable, traceable, reproducible encapsulation of the code, data, and environments used in computational research, version controlled and linked to the results they produce”⁹. The published capsules are automatically linked to a new Digital Object Identifier¹⁰ (DOI), which improves proper referencing and accelerates

³<https://community.amstat.org/biop/workinggroups/swe-wg> (accessed on 31 January 2026)

⁴<https://www.psiweb.org/about-us> (accessed on 31 January 2026)

⁵<https://www.efspi.org/> (accessed on 31 January 2026)

⁶<https://www.openstatsware.org/slides/r-govys-jan2023.html#/motivation> (accessed on 31 January 2026)

⁷<https://github.com/> (accessed on 31 January 2026)

⁸<https://ieee-dataport.org/> (accessed on 31 January 2026)

⁹<https://codeocean.com/product/compute-capsules> (accessed on 31 January 2026)

¹⁰<https://www.doi.org/> (accessed on 31 January 2026)

scientific discourse.

An important concept which is related to more than one of the above discussed concepts is an idea of a software package, or a library¹¹. A specific implementation of this idea, called a packaging protocol in R language, is central to the *Bioconductor* project - Gentleman et al. (2004) state that the package system enables the development of separate software modules and their distribution under well-defined standards. The idea of a package, and abilities to create it and distribute it, are at the heart of the *efficiency of development* requirement stressed by the creators of the *Bioconductor*, as sharing an implementation of a given algorithm in a form of a script uploaded to a certain repository, or possibly a multitude of interdependent scripts, without a structured connection made between them greatly reduces the ability of other researches to efficiently test and possibly extend the source code behind the given method. Packaging ones code is also related to the Level 1 and Level 3 of reproducibility as defined by Peng (2011). Firstly, by default, sharing a package means sharing a code, which means meeting the Level 1 of reproducibility requirements. With respect to Level 3, a created package, that must meet the well-established quality criteria to be published, also meets the description of linked & executable code, and can be understood as a means of implementing full reproducibility in the computational sciences. The description of the role of R packages in assuring reproducibility in detailed, technical way – as a “unified model for installing, managing, and publishing software contexts in R” – can be found in (Becker et al., 2017).

4.2 Review of reproducibility in semi-supervised fuzzy clustering

Based on the above discussion, we now present a review of the reproducibility levels associated with the publications from the literature review on the topic of semi-supervised fuzzy clustering. Table 4.1 presents a full categorization of the reviewed publications into each of the three categories:

1. no reproducibility,
2. reproducibility Level 1 - code available,
3. reproducibility Level 3 - linked & executable code.

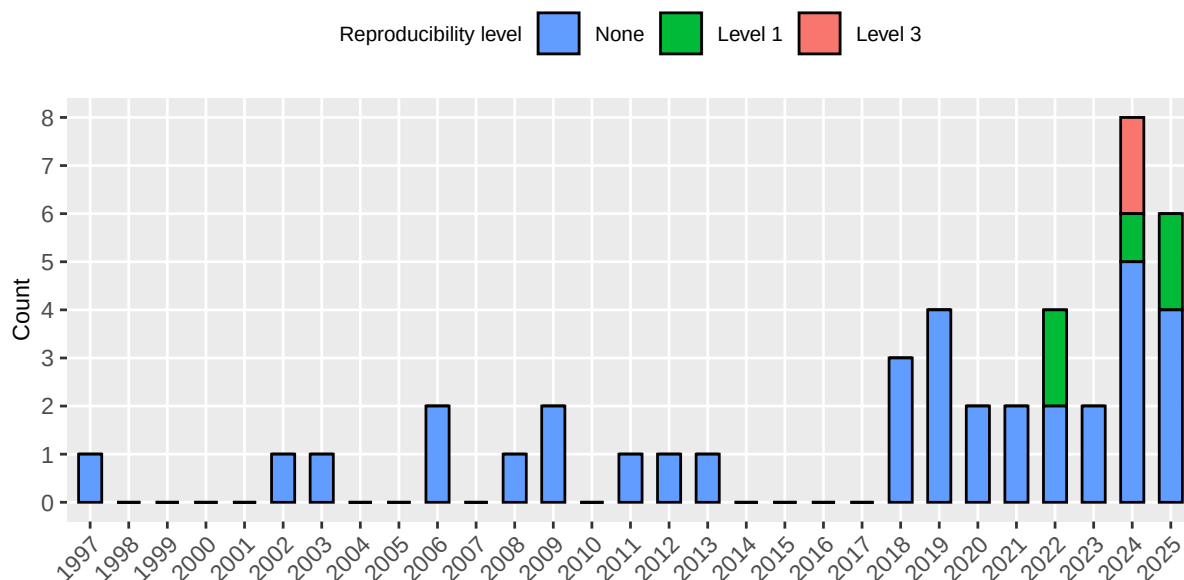
In our categorization, we omit the *code and data* reproducibility requirement from the original work of Peng (2011) categorized as Level 2, since – as we can see from Table 4.1

¹¹Two popular open source programming languages, **Python** and **R**, use different notions for the similar concept. **R** uses the term *package*, while **Python** uses the term *library*.

– the main issue is with no code shared at all, which is a fundamental requirement for any reproducibility in a sense of Peng (2011).

Fig. 4.2 presents the trend over the years, which aligns with the increasing consciousness about the need for reproducibility in the scientific community, and the availability and popularity of tools enabling achieving each level of reproducibility. None of the reviewed publications that were published before 2022 achieved any level of reproducibility, with 7 publications categorized as satisfying some level of reproducibility published post 2021. The trend shows also the increasing popularity of the topic of semi-supervised fuzzy clustering, with the increasing number of publications treating on this topic – and selected for the literature review – starting from the year 2018.

Figure 4.2: The number of reviewed publications per year, per reproducibility category.



Source: own work. The categorization of reproducibility levels is a custom approach based on (Peng, 2011): *Level 1* refers to only code being available, and *Level 3* refers to the availability of linked & executable code.

Table 4.1 lists specific publications according to their categorized level of reproducibility they present. Out of the 35 publications categorized as *not reproducible*, only one claimed availability of respective code and data. J. Xu et al. (2019) stated that they “have uploaded the computer code of all the algorithms in the submission system”, and that “Supplementary data to this article can be found online at <https://doi.org/10.1016/j.cageo.2019.06.005>”, but the investigation of this specific DOI revealed no code available at this website, and the data being inaccessible - an error message appears upon trying to download supplementary data stating that “The data that has been used is confidential”¹².

¹²<https://doi.org/10.1016/j.cageo.2019.06.005> (accessed on 31 January 2026)

Table 4.1: Literature review on the matter of reproducibility in semi-supervised fuzzy clustering.

Not reproducible (35 publications)
Antoine and Labroche (2018), Antoine et al. (2018), Bouchachia and Pedrycz (2006, 2003), Casalino, Castellano, et al. (2020), Casalino, Dominiak, et al. (2020), Casalino et al. (2019), Chunfang Li et al. (2008), Gan et al. (2023, 2019, 2024, 2025, 2018), J. Gao et al. (2006), Guo et al. (2021), He (2025), Huan et al. (2022), Kmita et al. (2023, 2022), Lai and Garibaldi (2011), L. Liu and Wu (2013), Mei (2019), Pedrycz and Waletzky (1997), Salehi et al. (2021), Samadi et al. (2025), Stutz and Runkler (2002), Wu and Hou (2025), J. Xu et al. (2019), S. Xu et al. (2024), Yasunori et al. (2009), Yin et al. (2012), H. Yu et al. (2024), B. Zhang, Huang, et al. (2024), B. Zhang, Jiang, and Xia (2024), and H. Zhang and Lu (2009)
Reproducible Level 1 - code available (5 publications)
Antoine et al. (2022), Kaczmarek-Majer et al. (2022), Khezri et al. (2025), Moghaddam et al. (2025), and Santos et al. (2024)
Reproducible Level 3 - linked & executable code (2 publications)
Golzari Oskouei et al. (2024) and Kmita et al. (2024)

Source: own work.

This finding aligns with the conclusion from the above mentioned broad reproducibility check performed by Collberg and Proebsting (2016) who found many publications claiming reproducibility being actually not reproducible.

A few of the publications, although not providing any reproducibility materials, include the specific description of the computers specification, and programming language used to perform their experiments. Among them, He (2025) claims to have use **MATLAB** software, and Samadi et al. (2025) and B. Zhang, Jiang, and Xia (2024) claim to have used **Python** programming language.

Interestingly, many publications that were categorized in this thesis as not satisfying any level of reproducibility present experimental results that require applying methods and models introduced beyond a specific publication - this applies to Gan et al. (2023, 2024, 2025, 2018), He (2025), Mei (2019), Salehi et al. (2021), Samadi et al. (2025), Wu and Hou (2025), Yin et al. (2012), B. Zhang, Huang, et al. (2024), B. Zhang, Jiang, and Xia (2024), and H. Zhang and Lu (2009). The authors of the above publications do not specify how they used the other models, whether they replicated them on their own, or used existing scripts, packages, or computing capsules. It is worth stressing that a similar observation applies to these works that were categorized in this thesis as achieving reproducibility Level 1 or reproducibility Level 3 - the authors of these publications provided implementations of their proposed algorithms, but did not replicate the compared models, and did not state how they were able to use them.

With respect to the 5 publications that achieved Level 1 of reproducibility by sharing

code, but not sharing any linked & executable code and data, the efficiency of development remains limited. Although the code was shared, the documentation and readability of it does not adhere to the best practices, and in particular - none of the publications shared the code in a form of an organized package. Regarding the chosen programming language, Kaczmarek-Majer et al. (2022) and Santos et al. (2024) used `Python`, and Antoine et al. (2022), Khezri et al. (2025), and Moghaddam et al. (2025) used `MATLAB` software.

Finally, two publications were categorized as fully reproducible (achieving Level 3), since they published associated CodeOcean *Compute Capsules*. Golzari Oskouei et al. (2024) publication was followed by the associated Compute Capsule (Oskouei et al., 2024), which implemented their method in `MATLAB`, and Kmita et al. (2024) publication was followed by the associated (Kmita, 2024) - the Compute Capsule reproducing the results of their analyses performed in the `R` language.

The conclusion from the review of reproducibility of results in the field of semi-supervised fuzzy clustering is that it remains very limited, even though it seems a trend is emerging for meeting the Level 1 reproducibility criterion in the last years. Surprisingly, the problem with reproducibility does not limit only to the novel models, but applies to the use of core models that are treated as benchmark methods in many comparisons - moreover, due to the nature of the additive combination technique described in Section 2.2.2, many of the implementations should be straightforward extensions of the core models. The reason for such situation in this field may be due to the phenomenon described by Rougier et al. (2017), who stress that there is little pay off for the researchers in spending time on assuring reproducibility in an environment requiring the researchers to publish. Until journals explicitly require certain materials to be provided together with the publication, there is no hard requirement and no clear incentive for providing Level 3 reproducibility. The above remark was made with respect to novel publications, whereas for replicating the software in a form a reproducible material for the already published results the pay off is even less, and there are no other incentives than an ethical care for a “better science” (Rougier et al., 2017). As Rougier et al. (2017) pointed, “the incentives for doing such time-consuming work are low or nonexistent”. This was a major motivation for creating the *ReScience C* journal concentrated only on replications of existing software. **The lack of a reproducible replication of core fuzzy clustering models is also a key motivation for the reproduction of Fuzzy c-Means, Semi-Supervised Fuzzy c-Means, Possibilistic c-Means, and Semi-Supervised Possibilistic c-Means that is undertaken in this thesis.**

4.3 Selected programming language

The reproducible replications of the four classical fuzzy clustering models are provided in this thesis in a form of a package written in the R programming language¹³ that is called **fussclust** (***F**uzzy **U**nsupervised and **S**emi-**S**upervised **C**lustering*). It is hosted at GitHub platform, at <https://github.com/kkmita/fussclust>, achieving Level 1 and Level 3 of reproducibility - and can be easily downloaded into the R session by using the syntax presented in Code Snippet 1.

Code Snippet 1 Installing **fussclust** package.

```
library(remotes)
remotes::install_github("kkmita/fussclust")
```

The chosen programming language – R – is highly popular especially in the field of statistical learning, data analysis and alike. Wickham (2019) lists the following advantages of R: it is open-source, has a strong, diverse community, and a huge number of existing packages and functionalities supporting activities related to the analysis of the data. The popularity of R, especially in the academic world, is also visible by its choice in popular textbooks. For example, a series *Introduction to Statistical Learning* had its two first editions published in 2013 and 2021 (James et al. (2013) and James et al. (2021), accordingly) using R language in the practical sections of the textbooks, and it was only in 2023 when a corresponding Python version of the textbook was published (James et al., 2023). Also, the popular in-depth textbook *Elements of Statistical Learning* had all its accompanying analyses performed in R (Hastie et al., 2009).

From the perspective of this thesis, and the proposed implementations, a particularly important feature of R is its prototyping capabilities, which are enabled by the so-called *vectorisation*, that is critical to the proficient use of this programming language. This aspect is stressed by Wickham (2019), who describes that “vectorising is about taking a whole-object approach to a problem, thinking about vectors, not scalars”. This feature of the R language aligns more closely with the mathematical notation of statistical computing, centered on linear algebra operations, which supports prototyping - implementing mathematically sound solutions in R.

However, with R being a high-level interpreted language, its rapid prototyping capabilities are typically balanced by a rather slow performance compared to other programming languages (Wickham, 2019). Since it is an extensible system, allowing augmenting the

¹³The specific version of R used was version 4.5.1, with a nickname **Great Square Root** (R Core Team, 2025).

core R capabilities with compiled code, a popular extension was implemented in the form of `Rcpp` package (Eddelbuettel & François, 2011) that greatly simplifies integrating C++ code with R code. Many existing R packages with computational methods are thus only interfaces to the underlying compiled C++ code, which greatly reduces the *efficiency of development* postulated by Gentleman et al. (2004).

In this thesis we propose an implementation of fuzzy clustering models based on core R that is efficient due to the appropriate use of the *vectorisation* paradigm mentioned above. The idea is to create a package that is easily extensible for other fuzzy clustering models, adhering to the principle of *efficiency of development*. We compare the efficiency of our solution with existing unsupervised fuzzy clustering packages that exhibit various approaches to the optimization of R code: `ppclust`, `fclust`, and `e1071`. **We compare only the unsupervised models due to lack of any packages devoted to the topic of semi-supervised fuzzy clustering that could be compared against.**

The first package to compare is `ppclust`: “Probabilistic and Possibilistic Cluster Analysis”¹⁴ (Cebeci, 2019), which implements Fuzzy c-Means as described in (Bezdek†, 1973) and Possibilistic c-Means as described in (Krishnapuram & Keller, 1993). The authors of `ppclust` use core R, and importantly, do not fully use the idea of vectorization, which can be checked by investigating the source code of the package, e.g. the `fcm` function implementing the Fuzzy c-Means model¹⁵ presented in Code Snippet 2.

Code Snippet 2 Part of `ppclust` source code.

```
for(j in 1:k)
  for(i in 1:n)
    if(any(d[i,] == 0))
      u[i,] <- rep(1/k, k)
    else
      u[i, j] <- 1/(sum((d[i, j]/d[i, ])^((2/dt)/(m - 1))))
```

This is the explicit use of double `for` loops that is known to frequently being an inefficient programming paradigm in the R language. As Burns (2011) describes, “A double for loop is often the result of a function that has been directly translated from another language. Translations that are essentially verbatim are unlikely to be the best thing to do. Better is to rethink what is happening with R in mind”.

The second package to compare is `fclust` (Ferraro et al., 2019), which provides algorithms for fuzzy clustering, cluster validity indices and plots for cluster validity and visualizing fuzzy clustering results. This package provides a function `FKM`, which imple-

¹⁴<https://cran.r-project.org/web/packages/ppclust/index.html> (accessed on 31 January 2026)

¹⁵<https://github.com/cran/ppclust/blob/master/R/fcm.R> (accessed on 31 January 2026)

ments a Fuzzy c-Means model as described in J. C. Bezdek (1981). An investigation of the source code reveals that the authors of `fclust` used the `Rcpp++` package to move the computations to the C++ language, treating R as an interface, which decreases the efficiency of development capabilities of this package¹⁶.

The last package to compare is `e1071`: “Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien”(Meyer et al., 2024), which provides a `cmeans` function implementing Fuzzy c-Means as described in (J. C. Bezdek, 1981). An investigation of the source code reveals that the respective function is also implemented in the C++ language, similarly to `fclust` package¹⁷.

None of the above packages fulfils all the desired features of a fully replicable, vectorised implementation supporting efficiency of development. Moreover, while all three packages provide implementations of the Fuzzy c-Means model, only `ppclust` provides an implementation of the Possibilistic c-Means model in form of a function `pcm`, whereas packages `fclust` and `e1701` do not. Therefore, the aim of the novel implementation is to provide vectorized, modular, packaged R code based on precise mathematical description that will also serve as a platform for extending the core fuzzy clustering models - either by proposing new models or replicating those already proposed in the literature albeit delivered in a non-reproducible manner.

4.4 Proposed vectorised implementations

In this Section, we provide extensive comments on the derivation of the R code used in the proposed `fussclust` package. The idea of the novel implementation of the four fuzzy clustering models is based on proprietary code apart from one aspect, the vectorized calculation of the distances matrix $D = [d_{jk}]$, where $d_{jk} = d(\mathbf{x}_j, \mathbf{v}_k)$ denotes the distance between j th observation and k th cluster’s prototype. Since the aspect of calculating such distances is not specific only to the fuzzy clustering problem, we will make use of the existing packages - in particular, a package `rdist` “Calculate Pairwise Distances” (Blaser, 2020) has a number of optimized functions to calculate selected distances, including the Euclidean distance.

¹⁶Compare https://github.com/cran/fclust/blob/master/src/FKM_rcpp.cpp that implements the `mainFKM` function (accessed on 31 January 2026).

¹⁷Compare <https://github.com/cran/e1071/blob/master/src/cmeans.c> that implements the respective functions (accessed on 31 January 2026).

4.4.1 Notation

Existing R implementations of unsupervised fuzzy clustering models follow the double for loop programming paradigm due to the nature of the formula for optimal membership in the reference paper by J. C. Bezdek (1981) that `ppclust`, `fclust`, and `e1071` are referring to. For convenience, we recall below this formula that was already presented in Eq. (2.20).

$$\hat{u}_{jk} = \left(\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2} \right)^{-1} = \left(\frac{d_{jk}^2}{d_{j1}^2} + \dots + \frac{d_{jk}^2}{d_{jc}^2} \right)^{-1} = \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}}. \quad (4.1)$$

Recall that in Eq. (3.44), we defined the data evidence

$$e_{jk} = \frac{1}{\sum_{g=1}^c \frac{d_{jk}^2}{d_{jg}^2}} \quad (4.2)$$

to simplify the notation. We will use this notation throughout the remainder of this Chapter.

Indeed, the scalar representation of the formula of the optimal membership degree at the level of a specific j th observation and a specific k th cluster intuitively calls for the programming of a double for loop. While an approach heavily based on explicit for loops can be efficient in low-level language like C++, it is by design inefficient in a high-level language such as R - recall the suggestion from Wickham (2019) who states that *vectorisation*, the key to efficient programming in R language, is about “taking a whole-object approach to a problem, thinking about vectors, not scalars”.

Therefore, we propose to represent the formulae for $\hat{U} = [\hat{u}_{jk}]$ and $\hat{V} = [\hat{v}_k]$ in a matrix form, which requires a novel, vectorized mathematical description of the operations required to calculate the decision $(\tilde{U}, \tilde{V})$ that has not been presented before. The notation used for this purpose is as follows.

An *array* is the most generic concept: an arranged collection of numbers from $\mathcal{R}$.

A *vector* $\mathbf{x} = [x_1, x_2, \dots, x_p]^T$ is an 1D-array of length p . The vector lies in the set formed by taking the Cartesian product of $\mathcal{R}$ a number of p times, denoted as $\mathcal{R}^p$. By default, we refer to the vector as to the column vector

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_p \end{bmatrix}.$$

We denote vectors with square brackets. If we want to treat them as single-row (or

single-column) matrices, we use round brackets that are reserved to denote matrices.

A *matrix* $B = (b_{ij})$ is a 2D-array of numbers $b_{ij} \in \mathcal{R}$ indexed by two indices $i = 1, \dots, N$ and $j = 1, \dots, M$, e.g. a 2×2 matrix B is represented as

$$B = \begin{pmatrix} b_{11} & b_{21} \\ b_{21} & b_{22} \end{pmatrix}.$$

An i th row of a matrix B is denoted by $B_{i:} = (b_{i1}, \dots, b_{iM})$.

A z -*multimatrix* B composed of z submatrices B_k , $k = 1 \dots, z$, is created by vertically stacking a number of z matrices $B_1, \dots, B_z$ of the same dimension $N \times M$. We denote a multimatrix by $B = (B_1|B_2|\dots|B_z)$; note that $B \in \mathbb{R}^{zN \times M}$. For example, if each $B_k \in \mathbb{R}^{3 \times 2}$, then

$$B = (B_1|B_2|\dots|B_z) = \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_z \end{pmatrix} = \begin{pmatrix} b_{11}^{(1)} & b_{21}^{(1)} \\ b_{21}^{(1)} & b_{22}^{(1)} \\ b_{31}^{(1)} & b_{32}^{(1)} \\ b_{11}^{(2)} & b_{21}^{(2)} \\ b_{21}^{(2)} & b_{22}^{(2)} \\ b_{31}^{(2)} & b_{32}^{(2)} \\ \dots & \dots \\ b_{11}^{(z)} & b_{21}^{(z)} \\ b_{21}^{(z)} & b_{22}^{(z)} \\ b_{31}^{(z)} & b_{32}^{(z)} \end{pmatrix}. \quad (4.3)$$

Further, it will be also convenient to denote by $\oslash$ the Hadamard division of two matrices, and by $\odot$ the Hadamard product of two matrices.

Definition 7. Let A and B be two matrices of the same dimensions $N \times M$. Then, the Hadamard division is a matrix given by

$$(A \oslash B)_{ij} = (A)_{ij} / (B)_{ij}. \quad (4.4)$$

Remark 1. Note than in R , for two matrices A and B , the Hadamard division is obtained by a following syntax: A / B .

Definition 8. Let A and B be two matrices of the same dimensions $N \times M$. Then, the Hadamard product is a matrix given by

$$(A \odot B)_{ij} = (A)_{ij} \cdot (B)_{ij}. \quad (4.5)$$

Remark 2. Note than in R , for two matrices A and B , the Hadamard product is obtained by a following syntax: $A * B$.

4.4.2 Fuzzy c-Means

Memberships

The main challenge in the efficient implementation of the Fuzzy c-Means algorithm is the construction of the data evidences matrix $E = [e_{jk}]$, since by Eq. (3.45a), the optimal membership $\hat{u}_{jk} = e_{jk}$ in case of Fuzzy c-Means. In particular, calculating a single e_{jk} requires retrieval and processing of the distances of a given observation between all c clusters' prototypes, i.e.

$$D_{j\cdot} = (d_{j1}, \dots, d_{jc}).$$

In the description of the proposed approach, we will be using a following simple exemplary distances matrix

$$D^* = \begin{pmatrix} 5 & 145 \\ 25 & 85 \\ 61 & 41 \end{pmatrix}. \quad (4.6)$$

The vectorised approach can be achieved using following auxiliary matrices. Let us consider the first row of the exemplary D^* , namely $D_{1\cdot}^* = (d_{11} = 5, d_{12} = 145)$. Let us define two functions: χ_H and χ_V , and the resulting matrices $DHE = \chi_H(D_{1\cdot}^*)$, and $DVE = \chi_V(D_{1\cdot}^*)$, where DHE is meant to denote *horizontally exploded* D^* , and DVE is meant to denote *vertically exploded* D^* , accordingly. They present as follows

$$DHE = \begin{pmatrix} d_{11} = 5 & d_{11} = 5 \\ d_{12} = 145 & d_{12} = 145 \end{pmatrix}, \quad DVE = \begin{pmatrix} d_{11} = 5 & d_{12} = 145 \\ d_{11} = 5 & d_{12} = 145 \end{pmatrix}. \quad (4.7)$$

If we now apply Hadamard division $\oslash$, we get

$$DHE \oslash DVE = \begin{pmatrix} \frac{d_{11}}{d_{11}} & \frac{d_{11}}{d_{12}} \\ \frac{d_{12}}{d_{11}} & \frac{d_{12}}{d_{12}} \end{pmatrix}. \quad (4.8)$$

By multiplying the resulting 2×2 matrix by the unit vector $\mathbf{1} = [1, 1]^T$, we obtain

$$(DHE \oslash DVE)\mathbf{1} = \begin{pmatrix} \frac{d_{11}}{d_{11}} + \frac{d_{11}}{d_{12}} \\ \frac{d_{12}}{d_{11}} + \frac{d_{12}}{d_{12}} \end{pmatrix} = \begin{pmatrix} e_{11}^{-1} \\ e_{12}^{-1} \end{pmatrix}. \quad (4.9)$$

Note that we now almost have a first row of the data evidence matrix $E_{1:}^*$! Treating the (column) vectors as matrices with one column, we apply Hadamard division and get

$$\left[\mathbf{1} \oslash \begin{pmatrix} e_{11}^{-1} \\ e_{12}^{-1} \end{pmatrix} \right]^T = (e_{11}, e_{12}) = E_{1:}^* \quad . \quad (4.10)$$

Let us now generalize this concept to the level of entire matrix D . A function $\chi_H : D_{j:} \mapsto \mathcal{R}^{c \times c}$ denoting the horizontal exploding operation is defined as

$$\chi_H(D_{j:}) = \begin{pmatrix} d_{j1} & \cdots & d_{jc} \\ \vdots & \ddots & \vdots \\ d_{jc} & \cdots & d_{jc} \end{pmatrix}. \quad (4.11)$$

Similarly, a function $\chi_V : D_{j:} \mapsto \mathcal{R}^{c \times c}$ denoting the vertical exploding operation is defined as

$$\chi_V(D_{j:}) = \begin{pmatrix} d_{j1} & \cdots & d_{jc} \\ \vdots & \ddots & \vdots \\ d_{j1} & \cdots & d_{jc} \end{pmatrix}. \quad (4.12)$$

Let us further define two N -multimatrices

$$\mathcal{R}^{Nc \times c} \ni \text{DHE} = \begin{pmatrix} \chi_H(D_{1:}) \\ \chi_H(D_{2:}) \\ \vdots \\ \chi_H(D_{N:}) \end{pmatrix} \quad (4.13)$$

and

$$\mathcal{R}^{Nc \times c} \ni \text{DVE} = \begin{pmatrix} \chi_V(D_{1:}) \\ \chi_V(D_{2:}) \\ \vdots \\ \chi_V(D_{N:}) \end{pmatrix}. \quad (4.14)$$

The first step towards obtaining the data evidences matrix E is to apply the following transformations, denoted by the function $\gamma : \mathcal{R}^{Nc \times c} \rightarrow \mathcal{R}^{Nc \times 1}$

$$\Gamma = \gamma(\text{DHE}, \text{DVE}) = \mathbf{1} \oslash \left([\text{DHE} \oslash \text{DVE}] \mathbf{1} \right) = \begin{pmatrix} E_{1:}^T \\ E_{2:}^T \\ \vdots \\ E_{N:}^T \end{pmatrix}. \quad (4.15)$$

We then only need to rearrange the elements of the resulting matrix Γ , by applying a function $\xi : \mathcal{R}^{Nc \times 1} \mapsto \mathcal{R}^{N \times c}$ defined as

$$\xi\left(\begin{pmatrix} E_{1.}^T \\ E_{2.}^T \\ \vdots \\ E_{N.}^T \end{pmatrix}\right) = \begin{pmatrix} E_{1.} \\ E_{2.} \\ \vdots \\ E_{N.} \end{pmatrix} = E = [e_{jk}]. \quad (4.16)$$

A R function implementing functions χ_H and χ_V is a function `dheve_fcm` presented below:

Code Snippet 3

```
dheve_fcm <- function(D, vertical) {
  if (vertical == TRUE) {    # chi_V
    elements = D[rep(1:nrow(D), each=ncol(D)), ]
  } else {    # chi_H
    elements = matrix(c(t(D))), [, rep(1, ncol(D))]
  }
  return(elements)
}
```

The vectorisation is achieved by appropriate subscripting, as according to Burns (2011), “subscripting in R is extremely powerful, and is often a key part of effective vectorization”.

Next, the mathematical operation of multiplying matrices - a part of the γ function:

$$[\text{DHE} \oslash \text{DVE}] \mathbf{1}$$

can be implemented using the `rowSums` function, which is specifically listed by Wickham (2019) as one the vectorised functions that can help in overcoming typical performance bottlenecks in computations performed by the means of R language. The reason for that is a general approach to the vectorisation in R: “vectorisation means finding the existing R function that is implemented in C and most closely applies to your problem” (Wickham, 2019). Therefore, the entire γ function defined in Eq. (4.15) can be implemented as presented in Code Snippet 4, and the reshaping function ξ can be implemented as in Code Snippet 5.

Code Snippet 4

```
gamma_fcm <- function(dhe, dve) {
  1 / rowSums(dhe/dve)
}
```

Code Snippet 5

```
xi_fcm <- function(E, c) {
  matrix(E, ncol = c, byrow = TRUE)
}
```

The ultimate R code to calculate evidences matrix E , which is equal to the optimal memberships matrix $\hat{U}$, is presented in Code Snippet 6.

Code Snippet 6

```
dve = dheve_fcm(D, vertical = TRUE)
dhe = dheve_fcm(D, vertical = FALSE)
E = xi_fcm(E = gamma_fcm(dhe, dve), c = ncol(dhe))
```

Prototypes

With respect to the optimal prototypes matrix $\hat{V}$, one can construct the entire weights matrix $\Phi = [\phi_{jk}]$, and perform appropriate operations on Φ and features matrix X . In case of Fuzzy c-Means, $\phi_{jk} = \hat{u}_{jk}^2 = e_{jk}^2$, but for any other model the following R code will apply as well - it will be the creation of the respective Φ matrix that will be model-specific, which is an input to this implementation of calculating $\tilde{V}$.

Let us consider the following matrix multiplication

$$\Phi^T X = \begin{pmatrix} \sum_j \phi_{j1} x_{j1} & \sum_j \phi_{j1} x_{j2} & \dots & \sum_j \phi_{j1} x_{jp} \\ \sum_j \phi_{j2} x_{j1} & \sum_j \phi_{j2} x_{j2} & \dots & \sum_j \phi_{j2} x_{jp} \\ \vdots & \vdots & \vdots & \vdots \\ \sum_j \phi_{jc} x_{j1} & \sum_j \phi_{jc} x_{j2} & \dots & \sum_j \phi_{jc} x_{jp} \end{pmatrix} = \begin{pmatrix} \sum_j \phi_{j1} \mathbf{x}_j^T \\ \sum_j \phi_{j2} \mathbf{x}_j^T \\ \vdots \\ \sum_j \phi_{jc} \mathbf{x}_j^T \end{pmatrix}_{[c \times p]}. \quad (4.17)$$

Clearly, $\Phi^T X$ is close to the generic formula of $\hat{V} = [\hat{\mathbf{v}}_k]$ from Eq. (3.10) derived in Section 3.2.1. What is missing is the division by the corresponding column-wise sums $\sum_{s=1}^N \phi_{sk}$ of the weights matrix Φ . This could be applied in a vectorised form by a Hadamard division

$$\tilde{V} = (\Phi^T X) \oslash \Sigma, \quad (4.18)$$

where

$$\Sigma = \begin{pmatrix} \sum_{s=1}^N \phi_{s1} & \sum_{s=1}^N \phi_{s1} & \cdots & \sum_{s=1}^N \phi_{s1} \\ \sum_{s=1}^N \phi_{s2} & \sum_{s=1}^N \phi_{s2} & \cdots & \sum_{s=1}^N \phi_{s2} \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{s=1}^N \phi_{sc} & \sum_{s=1}^N \phi_{sc} & \cdots & \sum_{s=1}^N \phi_{sc} \end{pmatrix}. \quad (4.19)$$

is a matrix of dimensions $c \times p$. However, in R language there is no need for an explicit creation of the entire matrix Σ . One can use the recycling rule: “if one tries to add two structures with a different number of elements, then the shortest is recycled to length of longest” (R Core Team, 2025). If the two structures involved include a matrix $(A)_{N \times M}$, and a N -vector, the recycling operation is performed column-wise, because in R, the contents of matrices are stored in the column-major order (R Core Team, 2025). Thus, a following R code presented in Code Snippet 7 will implement the vectorised Eq. (4.18).

Code Snippet 7

```
# `Phi` is a weights matrix of dimensions N x c
# `X` is a features matrix of dimensions N x p
V = (t(Phi) %*% X) / colSums(Phi)
```

The R code proposed in the above Code Snippet 7 is a generalization of the proposition on how to calculate prototypes in the **ppclust** package’s source code¹⁸, presented in Code Snippet 8. It shall be noted, though, that the authors of **ppclust** did not present the mathematical description of their approach, and their solution is specific to the Fuzzy c-Means model only.

Code Snippet 8 Part of **ppclust** source code.

```
v = t(u~m) %*% x / colSums(u~m)
```

4.4.3 Semi-Supervised Fuzzy c-Means

Memberships

To derive the vectorised implementation of optimal memberships in Semi-Supervised Fuzzy c-Means model, let us begin with presenting an alternate form of the formula for the optimal membership Eq. (3.45b) in Semi-Supervised Fuzzy c-Means as

$$\hat{u}_{jk}^{\text{SSFCM}} = \frac{1}{1 + \alpha_j} e_{jk} + \frac{\alpha_j}{1 + \alpha_j} f_{jk} = \frac{e_{jk} + \alpha_j f_{jk}}{1 + \alpha_j}. \quad (4.20)$$

¹⁸<https://github.com/cran/ppclust/blob/master/R/fcm.R> (accessed on 31 January 2026)

Further, let us consider an exemplary conceptual memberships matrix U , and an accompanying supervision matrix F

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \end{pmatrix}, \quad F = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}. \quad (4.21)$$

By Eq. (4.20), the optimal memberships matrix $\hat{U}$ treated as entirety in this example presents as follows

$$\hat{U} = \begin{pmatrix} \frac{e_{11} + \alpha}{1 + \alpha} & \frac{e_{12}}{1 + \alpha} \\ e_{21} & e_{22} \\ \frac{e_{31}}{1 + \alpha} & \frac{e_{32} + \alpha}{1 + \alpha} \end{pmatrix}. \quad (4.22)$$

Let us decompose this matrix. First, consider that

$$E + \alpha F = \begin{pmatrix} e_{11} + \alpha & e_{12} \\ e_{21} & e_{22} \\ e_{31} & e_{32} + \alpha \end{pmatrix}. \quad (4.23)$$

The vectorised matrix $\hat{U}$ from Eq. (4.22) can be represented as

$$\hat{U} = (E + \alpha F) \oslash \begin{pmatrix} 1 + \alpha & 1 + \alpha \\ 1 & 1 \\ 1 + \alpha & 1 + \alpha \end{pmatrix}. \quad (4.24)$$

Thus, calculation of memberships in Semi-Supervised Fuzzy c-Means can be achieved by transforming the data evidences matrix E – which implementation is described in the preceding Sections – with the user-provided, external supervision matrix F and the scaling factor α . By using the concept of recycling implemented in **R**, there is no need to explicitly create the entire matrix

$$\begin{pmatrix} 1 + \alpha & 1 + \alpha \\ 1 & 1 \\ 1 + \alpha & 1 + \alpha \end{pmatrix}$$

from Eq. (4.24), since its columns are idempotent; it suffices to create a vector of length N (the number of observations in the general case).

The generic implementation corresponding to Eq. (4.24) case is presented in Code

Snippet 9. We will refer to the supervision matrix F via a variable `superF`, since `F` is a reserved symbol in R referring to the `FALSE` logical value.

Code Snippet 9

```
U = (E + alpha * superF) / (1 + alpha * rowSums(superF))
```

Prototypes

Let us recall the formula for the weights in the Semi-Supervised Fuzzy c-Means model proposed in this thesis, presented originally in Eq. (3.58):

$$\phi_{jk} = u_{jk}^2 + \alpha b_j (u_{jk} - f_{jk})^2. \quad (4.25)$$

Let us continue with the exemplary matrices introduced in Eq. (4.21). In this case, the weights matrix would take a form

$$\Phi = \begin{pmatrix} u_{11}^2 + \alpha(u_{11} - 1)^2 & u_{12}^2 + \alpha(u_{12})^2 \\ u_{21}^2 & u_{22}^2 \\ u_{31}^2 + \alpha(u_{31})^2 & u_{32}^2 + \alpha(u_{32} - 1)^2 \end{pmatrix} \quad (4.26)$$

and can be thus represented as a sum of following matrices

$$U^2 + \underbrace{\begin{pmatrix} \alpha(u_{11} - 1)^2 & \alpha(u_{12})^2 \\ 0 & 0 \\ \alpha(u_{31})^2 & \alpha(u_{32} - 1)^2 \end{pmatrix}}_Z, \quad (4.27)$$

where $U^2 = [u_{jk}^2]$, and Z matrix can be represented as

$$Z = (U - F)^2 \odot \begin{pmatrix} \alpha & \alpha \\ 0 & 0 \\ \alpha & \alpha \end{pmatrix}, \quad (4.28)$$

where $\odot$ denotes the Hadamard product.

Matrix Z can be efficiently implemented in R by using the recycling feature as presented in Code Snippet 10, and then the weights matrix Φ can be calculated as as presented in Code Snippet 11. The final calculation of optimal prototypes $\hat{V}$ in SSFCMeans is implemented by Code Snippet 12

Code Snippet 10

```
Z = (U - superF)^2 * alpha * rowSums(superF)
```

Code Snippet 11

```
Phi_SSFCM = U^2 + (U - superF)^2 * alpha * rowSums(superF)
```

Code Snippet 12

```
# `Phi` is a weights matrix of dimensions N x c
# `X` is a features matrix of dimensions N x p
V = (t(Phi_SSFCM) %*% X) / colSums(Phi_SSFCM)
```

4.4.4 Possibilistic c-Means

Typicalities

The formula for the optimal membership, called a typicality, in unsupervised Possibilistic c-Means was provided in Eq. (2.34). Let us recall it below for convenience:

$$\hat{t}_{jk} = \frac{\gamma_k}{\gamma_k + d_{jk}^2}. \quad (4.29)$$

The vectorized optimal typicality matrix presents as follows:

$$\hat{T} = \begin{pmatrix} \frac{\gamma_1}{\gamma_1 + d_{11}^2} & \frac{\gamma_2}{\gamma_2 + d_{12}^2} & \cdots & \frac{\gamma_c}{\gamma_c + d_{1c}^2} \\ \frac{\gamma_1}{\gamma_1 + d_{21}^2} & \frac{\gamma_2}{\gamma_2 + d_{22}^2} & \cdots & \frac{\gamma_c}{\gamma_c + d_{2c}^2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\gamma_1}{\gamma_1 + d_{N1}^2} & \frac{\gamma_2}{\gamma_2 + d_{N2}^2} & \cdots & \frac{\gamma_c}{\gamma_c + d_{Nc}^2} \end{pmatrix} = G \oslash (G + D), \quad (4.30)$$

where

$$G = \begin{pmatrix} \gamma_1 & \gamma_2 & \cdots & \gamma_c \\ \gamma_1 & \gamma_2 & \cdots & \gamma_c \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_1 & \gamma_2 & \cdots & \gamma_c \end{pmatrix}. \quad (4.31)$$

The R code implementing $\hat{V}$ according to Eq. (4.30), using the concepts of vectorising by subscripting is presented in Code Snippet 13.

Code Snippet 13

```
# gammas is a vector of length c provided externally to the model
G = matrix(gammas, nrow = 1)[rep(1, nrow(D)), ]
Tp = G / (G + D)
```

Prototypes

Similarly to Fuzzy c-Means, the R code to calculate prototypes in the case of Possibilistic c-Means is

Code Snippet 14

```
# Phi is a weights matrix of dimensions N x c
# X is a features matrix of dimensions N x p
V <- (t(Phi) %*% X) / colSums(Phi)
```

where $\Phi = [\phi_{jk}] = [t_{jk}^2]$ in Possibilistic c-Means.

4.4.5 Semi-Supervised Possibilistic c-Means**Typicalities**

The formula for the optimal typicality in Semi-Supervised Possibilistic c-Means was provided in Eq. (3.69). Let us recall it below for convenience:

$$\hat{t}_{jk} = \frac{\gamma_k + \alpha_j d_{jk}^2 f_{jk}}{\gamma_k + (1 + \alpha_j) d_{jk}^2}, \quad (4.32)$$

where $\alpha_j = b_j \cdot \alpha$. To achieve the vectorised form, one should consider two modified distances matrices D_1 and D_2 :

$$D_1 = [\alpha_j f_{jk} d_{jk}^2], \quad (4.33a)$$

$$D_2 = [(1 + \alpha_j) d_{jk}^2], \quad (4.33b)$$

and have the optimal typicalities represented as

$$\hat{T} = (G + D_1) \oslash (G + D_2). \quad (4.34)$$

Let us consider an exemplary distances matrix D^* , and an accompanying supervision matrix F^*

$$D^* = \begin{pmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \\ d_{31} & d_{32} \end{pmatrix}, \quad F^* = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}. \quad (4.35)$$

In such case, matrix D_1 can be represented as

$$D_1 = \begin{pmatrix} d_{11}^2 & d_{12}^2 \\ d_{21}^2 & d_{22}^2 \\ d_{31}^2 & d_{32}^2 \end{pmatrix} \odot \begin{pmatrix} \alpha & 0 \\ 0 & 0 \\ 0 & \alpha \end{pmatrix}, \quad (4.36)$$

which can be computed in R using the recycling feature as

Code Snippet 15

```
D1 = D^2 * alpha * superF
```

and matrix D_2 can be represented as

$$D_2 = \begin{pmatrix} d_{11}^2 & d_{12}^2 \\ d_{21}^2 & d_{22}^2 \\ d_{31}^2 & d_{32}^2 \end{pmatrix} \odot \begin{pmatrix} 1 + \alpha & 1 + \alpha \\ 1 & 1 \\ 1 + \alpha & 1 + \alpha \end{pmatrix}, \quad (4.37)$$

and can be computed in R using the recycling feature as

Code Snippet 16

```
D2 = D^2 * (1 + alpha * rowSums(superF))
```

The final R code to calculate typicalities $\hat{T}$ from Eq. (4.34) in a general case is thus

Code Snippet 17

```
G = matrix(gammas, nrow = 1)[rep(1, nrow(D), )]  
Tp = (G + D^2 * alpha * superF) / (G + D^2 * (1 + alpha * rowSums(superF)))
```

Prototypes

The formula for the weights in Semi-Supervised Possibilistic c-Means is analogous to the one in Semi-Supervised Fuzzy c-Means and presents as

$$\phi_{jk} = t_{jk}^2 + \alpha b_j (t_{jk} - f_{jk})^2. \quad (4.38)$$

Therefore, the entire prototypes matrix $\hat{V}$ can be calculated using the R code from Code Snippet 18 analogous to the Code Snippets 11 and 12.

Code Snippet 18

```
Phi_SSPCM = Tp^2 + (Tp - superF)^2 * alpha * rowSums(superF)
V = (t(Phi_SSPCM) %*% X) / colSums(Phi_SSPCM)
```

4.5 Comparison of implementations

This section is devoted to the comparison of the proposed implementation in **fussclust** package against the selected **ppclust** (version 1.1.0.1), **e1071** (version 1.7.16), and **fclust** (version 2.1) state-of-the-art packages.

Numerical equivalence

The first step is to compare whether **fussclust** correctly implemented the two classical unsupervised models, Fuzzy c-Means and Possibilistic c-Means. We will achieve it by comparing the decision matrices $\tilde{V}$ obtained by running each of the models on the entire **iris** dataset, that was used throughout this thesis. The features matrix **Xiris** will contain information about sepal length and sepal width for all 150 observations available in the dataset.

To compare the results of Fuzzy c-Means model, we will create a random initialization $U^{(0)}$, denoted in the R program as **U0**, and call the code presented in Code Snippet 19. Note that out of the four implementations, only the function **e1071::cmeans** does not allow to provide the initialization memberships matrix, hence a different, random initialization will be used for this function (according to its implementation in the **e1071** package).

The results obtained are presented in Eq. (4.39).

Code Snippet 19

```

create_random_U <- function(nobs, C) {
  .U0 <- matrix(runif(nobs*C), ncol=C)
  .U0 <- t(apply(.U0, 1, function(x) x / sum(x)))

  return(.U0)
}

data(iris)
Xiris <- iris[, c("Sepal.Length", "Sepal.Width")] |> as.matrix()

U0 <- create_random_U(nrow(iris), 2)

fcm_ppclust = ppclust::fcm(x = Xiris, centers = 2, memberships = U0)
fcm_fclust = fclust::FKM(X = Xiris, k = 2, startU = U0)
fcm_e1071 = e1071::cmeans(x = Xiris, centers = 2)
fcm_fussclust = fussclust::FCM(X = Xiris, C = 2, U = U0)

```

$$\tilde{V}_{\text{ppclust}} = \begin{pmatrix} 6.515 & 2.966 \\ 5.121 & 3.202 \end{pmatrix}, \quad (4.39a)$$

$$\tilde{V}_{\text{fclust}} = \begin{pmatrix} 5.121 & 3.202 \\ 6.515 & 2.966 \end{pmatrix}, \quad (4.39b)$$

$$\tilde{V}_{\text{e1071}} = \begin{pmatrix} 6.515 & 2.966 \\ 5.121 & 3.202 \end{pmatrix}, \quad (4.39c)$$

$$\tilde{V}_{\text{fussclust}} = \begin{pmatrix} 5.121 & 3.202 \\ 6.515 & 2.966 \end{pmatrix}. \quad (4.39d)$$

The decision matrices presented in the Eq. (4.39) – adjusted for the order of rows in the matrix – indicate that the various implementations lead to the same results in case of Fuzzy c-Means.

The comparison of implementations of unsupervised Possibilistic c-Means requires more preparations. Firstly, according to Eq. (2.30), we will use the results of `fcm_fussclust` model to calculate values $\gamma_1 = 0.655$, $\gamma_2 = 0.614$.

Next, function `ppclust::pcm` has a parameter `fcmrun`, which is by default set to the value `TRUE`, triggering the action of calculating initial memberships and centers from a corresponding run of Fuzzy c-Means model. This is unnecessary, as `ppclust::pcm` allows to provide arbitrary initialization matrix $U^{(0)}$ through the use of `memberships`

argument. To achieve the same initialization mechanism between `ppclust::pcm` and the proposed implementation `fussclust::PCM`, a following syntax will be used described in Code Snippet 20.

Code Snippet 20

```
pcm_ppclust = ppclust::pcm(x = Xiris, centers = 2, memberships = U0,
                          fcmrun = FALSE,
                          omega = c(0.655, 0.614))

pcm_fussclust = fussclust::PCM(X = Xiris, C = 2, U = U0,
                              gammas = c(0.655, 0.614))
```

The following results are obtained as follows in Eq. (4.40).

$$\tilde{V}_{\text{ppclust}} = \begin{pmatrix} 5.958 & 2.93 \\ 5.966 & 2.926 \end{pmatrix}, \quad \tilde{V}_{\text{fussclust}} = \begin{pmatrix} 5.958 & 2.93 \\ 5.966 & 2.926 \end{pmatrix}. \quad (4.40)$$

The decision matrices obtained indicate that the two implementations lead to the same results in case of Possibilistic c-Means initialized in the same way.

Efficiency of implementation

The second step to compare the selected implementations is to measure the efficiency of underlying code. To achieve that, comparisons were run on a *AMD Ryzen 5 PRO 4650U with Radeon Graphics* machine with 12 cores and 25.1 GB of RAM using the `bench` package (Hester, Jim & Vaughan, Davis, 2023) - the toolset to accurately analyze execution times for R expressions. Specifically, a function `bench::mark` was used.

For the comparisons, we will use simulated datasets with the same structure, but growing number of observations N . The number of observations will be denoted using the letter k to mark the thousand unit, i.e., $1k$ denotes a thousand observations, and $10k$ denotes ten thousands observations. The data are simulated from multivariate normal distribution

$$\mathcal{N}_4(\mu = [0, 1, 2, 3]^T, \Sigma = \text{diag}(1)),$$

and the number of observations will include: $0.1k$, $1k$, $10k$, $20k$, $30k$, $40k$, $50k$, $100k$ for the Fuzzy c-Means implementation comparison, and $0.1k$, $1k$, $10k$ for Possibilistic c-Means comparison.

To assure a fair comparison, the parameters controlling the value of the convergence threshold and the maximal number of iterations to perform if convergence was

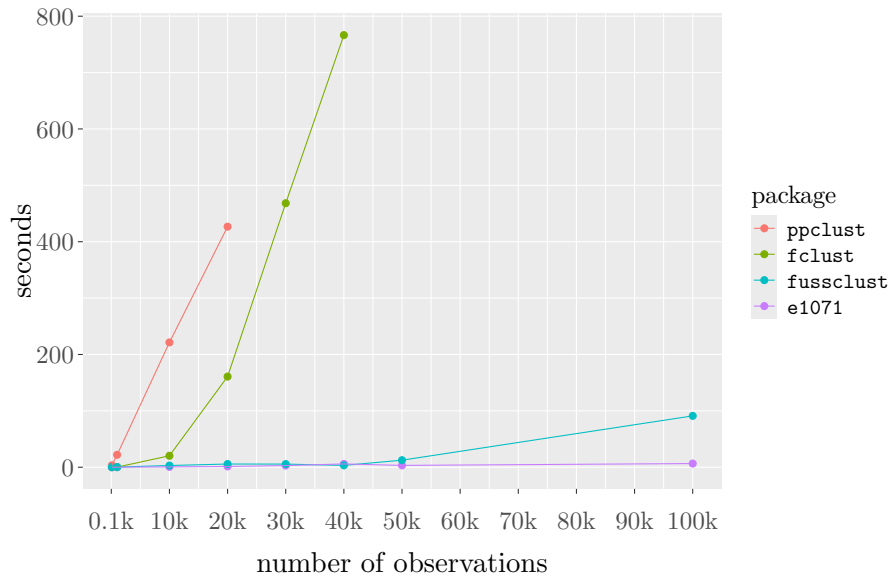
not achieved are set equal between each of the implementations. The appropriate settings are provided in Code Snippet 21.

Code Snippet 21 Fair comparison of FCM implementations efficiency

```
res <- bench::mark(
  ppclust = ppclust::fcm(x = X, centers = 4,
    con.val = 1e-9, iter.max = 100),
  fclust = fclust::FKM(X = X, k = 4,
    conv = 1e-9, maxit = 100),
  e1071 = e1071::cmeans(x = X, centers = 4,
    iter.max = 100, control = list(reltol = 1e-9)),
  fussclust = fussclust::FCM(X = X, C = 4,
    conv_criterion = 1e-9, max_iter = 100),
  check = FALSE,
  iterations = 3
)
```

The full results of the total time of execution (converted to seconds) of Fuzzy c-Means implementations calculated by the `bench::mark` function are provided in Table 4.2, and visualised in Fig. 4.3. In further analysis, we will use the convention $t(X_k)$ to denote the time in seconds required to accomplish the calculations for the dataset X_k .

Figure 4.3: The results of efficiency comparison of Fuzzy c-Means implementation.



Source: own work.

Firstly, analysing the results for datasets $X_{0.1k}$, X_{1k} , X_{10k} , and X_{20k} , one can see the rapidly growing time required to accomplish the calculations for both `ppclust` and `fclust` packages, and minimal increases in the required time for both `e1071` and `fussclust`

Table 4.2: The results of efficiency comparison of Fuzzy c-Means implementation.

package	size	seconds	package	size	seconds
ppclust	0.1k	3.448	ppclust	30k	NA
fclust	0.1k	0.015	fclust	30k	468.293
e1071	0.1k	0.007	e1071	30k	2.900
fussclust	0.1k	0.066	fussclust	30k	5.565
ppclust	1k	22.033	ppclust	40k	NA
fclust	1k	0.248	fclust	40k	766.459
e1071	1k	0.062	e1071	40k	5.851
fussclust	1k	0.260	fussclust	40k	3.257
ppclust	10k	221.388	ppclust	50k	NA
fclust	10k	20.312	fclust	50k	NA
e1071	10k	0.634	e1071	50k	3.273
fussclust	10k	3.028	fussclust	50k	12.495
ppclust	20k	426.718	ppclust	100k	NA
fclust	20k	160.746	fclust	100k	NA
e1071	20k	1.337	e1071	100k	6.496
fussclust	20k	5.680	fussclust	100k	91.071

Source: own work.

packages. Clearly, the implementation in **ppclust** package is the least efficient, with $t(X_{10k}) = 221.388$, and $t(X_{20k}) = 426.718$. For X_{10k} , all the three remaining packages were much faster, but for X_{20k} , the **fclust** implementation started to achieve much worse results than for X_{10k} , achieving $t(X_{20k}) = 160.746$. It was still a result nearly three times lower than the corresponding time achieved by the **ppclust** package, but an exponential trend has emerged.

For the above reasons, due to computational cost, it was decided that for the datasets with the number of observations greater than $20k$, further **ppclust** models will not be fitted, as the respective comparison for four above mentioned datasets provided enough evidence to claim that **ppclust** is the least efficient implementation. For **fclust**, this package was also used to fit models to the datasets X_{30k} and X_{40k} to test the further behaviour of the observed exponential trend. The results visible in Fig. 4.3 confirm the rapidly growing time required to complete calculations for the bigger datasets, and allow to treat **fclust** as the second least efficient implementation of Fuzzy c-Means model.

Finally, the difference between the proposed **fussclust** and **e1071** package is negligible – compared to the efficiency of the other packages – for datasets with number of observations less than or equal $50k$. An additional step was undertaken for these two packages to test their efficiency on the dataset with the number of observations equal to $100k$. For this dataset, $t(X_{100k}) = 91.071$ for **fussclust**, and $t(X_{100k}) = 6.496$ for **e1071**. At the dimensionality of $100k$ observations, the difference between the two implementations becomes clearly visible, with **e1071** being the most efficient implementation.

Last but not least, we also test the efficiency of the implementations of Possibilistic c-Means that are available in `ppclust` and the proposed `fussclust` packages. The results provided in Table 4.3 are in line with the results obtained for Fuzzy c-Means, confirming the supremacy of the proposed `fussclust` package in terms of efficiency of implementation.

Table 4.3: The results of efficiency comparison of Possibilistic c-Means implementation.

package	size	seconds
<code>ppclust</code>	0.1k	2.346
<code>fussclust</code>	0.1k	0.022
<code>ppclust</code>	1k	8.254
<code>fussclust</code>	1k	0.066
<code>ppclust</code>	10k	62.906
<code>fussclust</code>	10k	0.569

Source: own work.

Conclusions

The proposed implementation of classical fuzzy clustering models, Fuzzy c-Means and Possibilistic c-Means, remains efficient as compared with other state-of-the-art R implementations. It achieves superior results to `ppclust` and `fclust` packages, and comparable results to `e1071` package for datasets of dimensionality lower than 100,000 observations.

In addition to the efficient implementation, it satisfies the *efficiency of development* principle - the mathematical operations, and corresponding R code are documented in a traceable, transparent way, and enable easy extensions of the core models. In this thesis, such extensions were provided in a form of Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means implementations.

Chapter 5

Empirical analysis of impact of partial supervision and experiments

The proposed explainability framework for impact of partial supervision in semi-supervised fuzzy clustering, together with the actual explanations of the impact of partial supervision on memberships and prototypes in classical Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means models delivered in Chapter 3 of this thesis, showed **the importance of considering the impact of partial supervision specifically with respect to the prototypes** - supported by Interpretation 2 treating optimal memberships as adjusted weights in the semi-supervised case.

The analyses performed in Chapter 3 were of theoretical nature, whereas this Chapter 5 is devoted to the empirical impact of partial supervision. We study how the various values of the scaling factor α impact the outcome of a model – specifically, the decision prototypes $\tilde{V}_\alpha$ – under *ceteris paribus* conditions, i.e., for the other experimental factors (including a specific dataset, ratio $\frac{M}{N}$ and concrete values of initialization matrix $U^{(0)}$) fixed.

The empirical analysis requires the availability of the appropriate software. We build on the achievements from Chapter 4, and use the proposed implementations of Fuzzy c-Means, Possibilistic c-Means, Semi-Supervised Fuzzy c-Means, and Semi-Supervised Possibilistic c-Means available in the `fussclust` package created as a part of this thesis, and described in Chapter 4. In particular, the availability to extract the weights matrix $\Phi^{(l)}$ from each estimated model is critical to the analyses performed in this Chapter 5.

All the experimental results presented in Chapter 1, Chapter 2 and this Chapter 5 were created by running respective `R Markdown notebooks`; the notebooks containing the code, and the corresponding formatted `GitHub-flavoured markdown` outputs rendering the obtained results, are shared on the author’s personal GitHub account: <https://github.com/kkmita/phd-thesis>¹.

¹The files include `phd_analyses_chapters_1_2_5.Rmd` and associated

As concluded in Section 1.3, the ultimate interest of researchers is frequently to consider grids of values of the scaling factor α , and the challenge is to construct *good* grids. The empirical explanation of the impact of partial supervision delivered in this Chapter leads to creating the guidance on how to construct a *good* grid $\mathcal{A} = (a_1, \dots, a_q)$ when using the class of semi-supervised fuzzy clustering models.

The guidance is underpinned by identification and a detailed analysis of two phenomena: underimpact and overimpact of partial supervision. The former is related to using too small values of α ; the occurrence of underimpact can corrupt the inference drawn from the specific SSFC model. In Section 5.1, we propose a diagnostic tool in form of a visual display that we call the underimpact-plot (U-plot in short) aimed at detecting underimpact. It is followed by an auxiliary plot called the Φ -plot that enables the analysis of the adjusted weights. Moreover, as we show, underimpact is of stochastic nature. In Section 5.4, we implement experimental simulation protocol to estimate the frequency of underimpact occurrence in selected cases.

The latter phenomenon – overimpact – concerns the upper limit a_q of the grid. Theoretically, there does not exist an upper bound for the value of the scaling factor α to be used; the issue of overimpact is related to the considerations about the value of the empirical impact of partial supervision $\tilde{\pi}(\alpha)$ when $\alpha \rightarrow \infty$. In Section 5.2, we propose a diagnostic tool in form of a visual display that we call the overimpact-plot (O-plot in short) aimed at detecting overimpact. As we argue, the O-plot is a better tool to estimate empirical impact of partial supervision $\tilde{\pi}(\alpha)$ than the U-plot.

Last but not least, in Chapter 3 theoretical explanations of the impact of partial supervision were provided for both Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means, whereas in this Chapter 5, we introduce and analyze the underimpact and overimpact of partial supervision using Semi-Supervised Fuzzy c-Means only. The justification for that decision is presented in Section 5.1.2.

5.1 Underimpact of partial supervision

5.1.1 Explanation of differences in accuracy

We start our analysis of the phenomenon called the underimpact of partial supervision by revisiting the experiment introduced in Section 1.4, where two characteristics of iris – sepal length and sepal width – were used to cluster the observations from the `iris`

`phd_analyses_chapters_1_2_5.md` containing results presented in Chapters 1, 2 and 5 apart from Section 5.4, and `phd_analyses_chapters_5_simulations.Rmd` and associated `phd_analyses_chapters_5_simulations.md` containing results presented in Section 5.4.

dataset. Recall that this dataset comes with full supervision, which provides the reference partitioning of the observations, and the semi-supervised scenario was simulated by random sampling 20% of observations to remain supervised, treating artificially the remaining observations as unsupervised. The observations matrix was further randomly split into X_{train} and X_{test} (10% of supervised observations were moved to the X_{test}), and three Semi-Supervised Fuzzy c-Means models were estimated based on the X_{train} data - each model M_α was estimated with a different value of the scaling factor $\alpha \in \{0.1, 1, 5\}$.

The three classes (referring to the species of iris) were organized in an arbitrary order in the labels 3-tuple $\mathcal{Y}$ as

$$\mathcal{Y} = (y^1 = \text{setosa}, y^2 = \text{versicolor}, y^3 = \text{virginica}). \quad (5.1)$$

Further, external quality criteria were calculated for the above scenario and presented in Table 1.1. We observed stable results for external clustering quality criteria (partition coefficient PC, Xie-Bieni measure XB and Silhouette index), but a huge variation in the results for accuracy – the external classification quality criterion – measured on the test dataset. The specific results included

$$\text{acc}(M_{0.1}) = 0.083, \quad \text{acc}(M_1) = 0.833, \quad \text{acc}(M_5) = 0.833. \quad (5.2)$$

The surprising difference in the quality of classification can be explained when we focus on the relation $\tilde{V} \sim \alpha$, per the requirements of the explainability framework introduced in this thesis. We will denote the decision from a model estimated with a given value of the scaling factor α as $\tilde{V}_\alpha$, or $\tilde{V}(\alpha)$ - the latter convention will allow the use of lower subscript for other purposes.

Recall that using SSFCMeans as a classifier is based on a following decision rule

$$\rho(\mathbf{x}) = y^{k^*}, \quad \text{where } k^* = \arg \min_{k \in \{1, \dots, c\}} d(\mathbf{x}, \tilde{\mathbf{v}}_k). \quad (5.3)$$

Since all the models were fit with the same initialization matrix $U^{(0)}$, on the same dataset X_{train} , and the accuracy was measured on the same dataset X_{test} , these are the actual values of the matrices $\tilde{V}_{0.1}$, $\tilde{V}_1$ and $\tilde{V}_5$ that must have contributed to the differences in accuracy. Let us examine these matrices presented in Eq. (5.4).

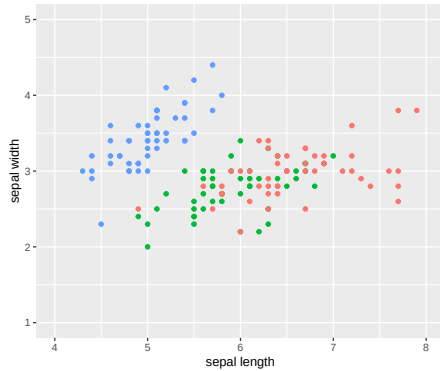
$$\tilde{V}_{0.1} = \begin{pmatrix} 5.75 & 2.757 \\ 6.776 & 3.057 \\ 5.04 & 3.341 \end{pmatrix}, \quad \tilde{V}_1 = \begin{pmatrix} 4.982 & 3.351 \\ 5.8 & 2.746 \\ 6.779 & 3.041 \end{pmatrix}, \quad \tilde{V}_5 = \begin{pmatrix} 4.984 & 3.355 \\ 5.797 & 2.747 \\ 6.749 & 3.023 \end{pmatrix}. \quad (5.4)$$

The order of rows in matrix $\tilde{V}_{0.1}$ differs from the order of rows in matrices $\tilde{V}_1$ and $\tilde{V}_5$, but with respect to the nominal values of the prototypes, they all have estimated approximately similar prototypes - it can be seen in Eq. (5.5), where we present the reordered matrix $\tilde{V}'_{0.1}$.

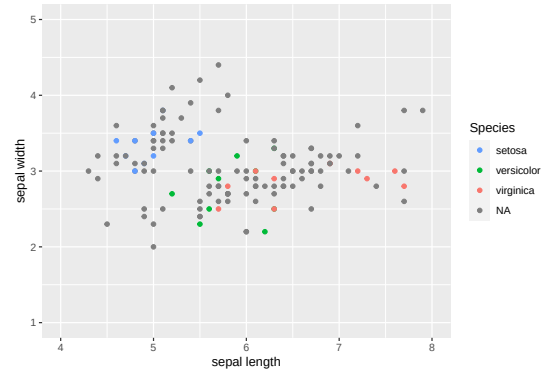
$$\tilde{V}'_{0.1} = \begin{pmatrix} 5.04 & 3.341 \\ 5.75 & 2.757 \\ 6.776 & 3.057 \end{pmatrix}, \quad \tilde{V}_1 = \begin{pmatrix} 4.982 & 3.351 \\ 5.8 & 2.746 \\ 6.779 & 3.041 \end{pmatrix}, \quad \tilde{V}_5 = \begin{pmatrix} 4.984 & 3.355 \\ 5.797 & 2.747 \\ 6.749 & 3.023 \end{pmatrix}. \quad (5.5)$$

The difference in the order of rows in the matrices presented in Eq. (5.4) corresponds to the difference in the values of accuracy. In the case of this specific, fully supervised scenario analyzed in this Subsection 5.1.1 – where we artificially simulate partial supervision – we can unequivocally determine which order of rows is the expected one, given the established order of the labels $\mathcal{Y}$ from Eq. (5.1). Let us recall for convenience Fig. 1.3, presented in the introductory Section 1.4, in the Fig. 5.1 presented below.

Figure 5.1: Dataset *iris* (R. A. Fisher, 1936).



(a) Fully supervised data.



(b) Simulated partial supervision ($\frac{M}{N} = 0.2$).

Source: own work.

Looking at the Fig. 5.1b, one cannot unequivocally establish the expected order of prototypes, but making use of the full supervision presented in Fig. 5.1a, it is possible. Each observation $\mathbf{x}_j \in \mathcal{R}^2$ from the X matrix is a two-element vector $\mathbf{x}_j = [x_{j1}, x_{j2}]^T$, where the first coordinate corresponds to sepal length, and the second coordinate corresponds to sepal width. Analyzing sepal length, we can see from Fig. 5.1a that on average, iris of *virginica* species have the highest value of this sepal length, and iris of *setosa* species have the lowest value. It is reflected in the actual average values of x_{j1} calculated on the X_{train} dataset, that are presented in Table 5.1.

Table 5.1: Average values of sepal length for different species of iris.

species	average sepal length
setosa	5.004
versicolor	5.913
virginica	6.617

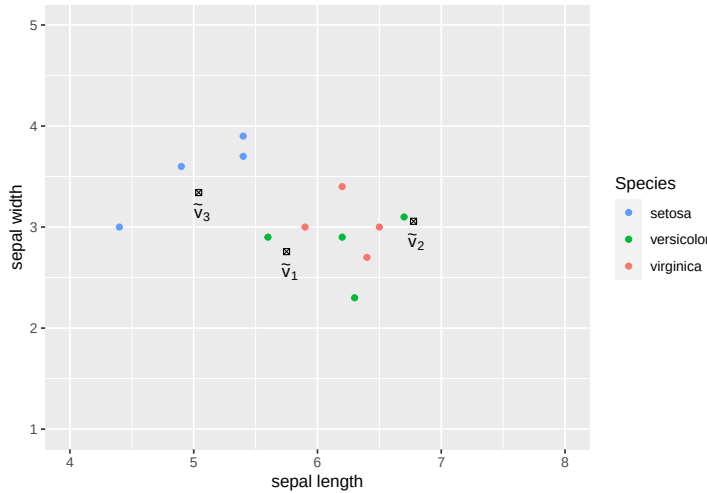
Source: own work.

By the means of the established order of labels in $\mathcal{Y}$, we associate the first prototype with the class of **setosa** iris, the second prototype with the class of **versicolor** iris, and the third prototype with the class of **virginica** iris². Let $\tilde{v}_{k,1}$ denote the first coordinate of the k th prototype. Knowing the average values of sepal length (i.e., the first coordinate of the observation's vector) in the X_{train} , we expect a similar order in any decision prototypes matrix:

$$\tilde{v}_{1,1} < \tilde{v}_{2,1} < \tilde{v}_{3,1}. \quad (5.6)$$

This expected order is achieved in decisions $\tilde{V}_1$ and $\tilde{V}_5$, but not in $\tilde{V}_{0.1}$. This deviation from the expected order has consequences for the predictions made on the X_{test} dataset, when applying the decision rule defined in Eq. (5.3). To visualize that, in Fig. 5.2 we present the test observations coloured accordingly to their true class, and the prototypes $\tilde{V}_{0.1}$ estimated by the model $M_{0.1}$.

Figure 5.2: Test dataset X_{test} and the decision prototypes.



Source: own work. The black squares correspond to the prototypes from the decision $\tilde{V}_{0.1}$ from Eq. (5.4).

As depicted in Fig. 5.2, the closest prototype for the test observations from class

²For the detailed description of the expected order of rows in the decision $\tilde{V}$, refer to the Section 3.2.5 of this thesis.

setosa (coloured in blue) is the third prototype $\tilde{\mathbf{v}}_3 = [5.04, 3.341]^T$, which is associated with the third label $y^3 = \text{virginica}$. As a result, applying the ρ decision function defined in Eq. (5.3) leads to the missclassification of all of the four observations as **virginica**. A similar situation occurs for all test observations of the class **virginica** (coloured in orange), for which the closest prototype is either the first prototype $\tilde{\mathbf{v}}_1 = [5.75, 2.757]^T$ associated with the first label $y^1 = \text{setosa}$, or the second prototype $\tilde{\mathbf{v}}_2 = [6.776, 3.057]^T$ associated with the second label $y^2 = \text{vericolor}$. Analogously, a similar situation happens for all but one test observations of the class **versicolor** (coloured in green), which are closest to the prototype $\tilde{\mathbf{v}}_1 = [5.75, 2.757]^T$, associated with the class $y^1 = \text{setosa}$.

These repercussions in a form of underestimated quality of classification when using a decision rule ρ based on $M_{0.1}$ Semi-Supervised Fuzzy c-Means model are a consequence of the phenomenon that we introduce in this thesis and propose to call the underimpact of partial supervision.

As we shall show in the following Subsection 5.1.3, the detection of underimpact does not require a split into the training dataset, and a test dataset - it is specific to any dataset $X \in \mathcal{R}^p$ used to estimate a semi-supervised fuzzy clustering model. Nevertheless, the use of the test dataset allows to clearly show the serious consequences for the use of semi-supervised fuzzy clustering models as classifiers, where the result of underimpact is the underestimation of classification quality of the model in a form of artificially low values of metrics such as accuracy.

In our example, the underimpact occurred for $\alpha = 0.1$, but did not occur for $\alpha \in \{1, 5\}$. This suggests that underimpact is related with the use of *too small values* of α , and the task of detection of underimpact is finding a threshold $\alpha^\dagger$ such that for $\alpha \geq \alpha^\dagger$ the underimpact is not occurring, and for $\alpha < \alpha^\dagger$, the underimpact is occurring. However, before we proceed to the general task of detecting underimpact, we first need to show the conditions of underimpact to support the claim that underimpact is related to using *too small values* of α - it is the topic of Subsection 5.1.3.

5.1.2 Coincident prototypes in Semi-Supervised Possibilistic c-Means

Before we proceed to the analysis of the conditions of underimpact, in this Subsection we justify the sole focus on Semi-Supervised Fuzzy c-Means model in the entire Chapter 5. Let us start with fitting three Semi-Supervised Possibilistic c-Means models, analogous to the SSFCMeans models presented in the previous Subsection. The decision prototypes present as follows:

$$\tilde{V}_{0.1}^{\text{SSPCM}} = \begin{pmatrix} 5.895 & 2.915 \\ 5.936 & 2.901 \\ 5.967 & 2.905 \end{pmatrix}, \quad \tilde{V}_1^{\text{SSPCM}} = \begin{pmatrix} 5.367 & 3.134 \\ 5.908 & 2.872 \\ 6.229 & 2.92 \end{pmatrix}, \quad \tilde{V}_5^{\text{SSPCM}} = \begin{pmatrix} 5.061 & 3.304 \\ 5.86 & 2.811 \\ 6.559 & 2.903 \end{pmatrix}. \quad (5.7)$$

Firstly, let us notice that $\tilde{V}_1^{\text{SSPCM}}$ and $\tilde{V}_5^{\text{SSPCM}}$ adhere to the expected order of prototypes from Eq. (5.6), and thus do not suffer from the underimpact phenomenon, similarly to the corresponding decision prototypes for Semi-Supervised Fuzzy c-Means presented in Eq. (5.4). The numerical values differ between the corresponding Semi-Supervised Possibilistic c-Means and Semi-Supervised Fuzzy c-Means models, exhibiting the different approaches to modelling uncertainty.

Secondly, in case of the decision prototypes $\tilde{V}_{0.1}^{\text{SSPCM}}$, it also adheres to the expected order of prototypes from Eq. (5.6), but the differences between each of the prototypes are of very little magnitude. Indeed, it can be argued that introducing partial supervision with the value of the scaling factor $\alpha = 0.1$ did not counteract the problem of coincident clusters' prototypes described in Section 2.1.3, whereas higher values of $\alpha = 1$ and $\alpha = 5$ clearly did counteract this problem. Therefore, introducing partial supervision with adequately high value of the scaling factor α in this case can be viewed as a remedy to the coincident clusters' prototypes issue, but the focus of this Chapter is on the novel phenomenon of underimpact of partial supervision.

With regard to the underimpact, comparing $\tilde{V}_{0.1}$ from Eq. (5.4) obtained from Semi-Supervised Fuzzy c-Means model, and $\tilde{V}_{0.1}^{\text{SSPCM}}$ from Eq. (5.7), it can be seen that **the analysis of underimpact phenomenon should be preceded by the analysis of coincident clusters' prototypes phenomenon in case of Semi-Supervised Possibilistic c-Means, whereas it can be directly studied in case of Semi-Supervised Fuzzy c-Means.** Although the order of prototypes in $\tilde{V}_{0.1}^{\text{SSPCM}}$ adheres to the expected order of prototypes from Eq. (5.6), the minimal magnitude of differences does not allow to unequivocally state whether this specific outcome of the SSPCMean model is free from underimpact or not, as it can be argued that it suffers from the problem of coincident clusters' prototypes.

Due to the above reason, and the lack of space, this Chapter 5 introduces, describes, and empirically analyzes the issue of underimpact of partial supervision using Semi-Supervised Fuzzy c-Means class of models only. Nevertheless, the entire discussion, together with the proposed diagnostic tools, applies to the Semi-Supervised Possibilistic c-Means case – after excluding the models affected by the coincident clusters' prototypes issue – and to any other semi-supervised fuzzy clustering model using the additive com-

bination technique as introduced in Section 2.2.2.

5.1.3 Conditions of underimpact

In this thesis, we claim that the underimpact of partial supervision occurs when two conditions are satisfied:

1. the random initialization $U^{(0)} = \Phi^{(0)}$ is untoward partial supervision,
2. the value of the scaling factor α is too low to counteract the untoward $\Phi^{(0)}$.

In further part of this Subsection, we will discuss each of the conditions separately, but a few remarks applicable to each of the conditions will ease the discussion. Firstly, recall from Eq. (3.10) that the form of the generic optimal prototypes applicable to any semi-supervised fuzzy clustering model presents as

$$\hat{\mathbf{v}}_k = \frac{\sum_{j=1}^N \phi_{jk} \mathbf{x}_j}{\sum_{j=1}^N \phi_{jk}} = \sum_{j=1}^N \frac{\phi_{jk}}{\sum_{s=1}^N \phi_{sk}} \mathbf{x}_j.$$

and that ϕ is a weighing function. We will adhere to Interpretation 2 and treat the optimal memberships as weights, where the scaling factor α is adjusting the weights of the supervised observations.

Secondly, recall that we will distinguish between the impact of partial supervision on the prototypes in the first loop of the Alternating Optimization (AO) algorithm (function $\pi_V^1(\alpha)$ defined in Eq. 3.27), and any loop $l > 1$ of the AO algorithm (function $\pi_V^+(\alpha)$ defined in Eq. 3.28). The distinction between the loops is reflected in the notation, where ϕ_{jk} was used to denote weights in the $\pi_V^1(\alpha)$ function, and $\hat{\phi}_{jk}$ was used to denote weights in the $\pi_V^+(\alpha)$ function.

For the analysis of underimpact, we will also need to refer to the random initialization $U^{(0)}$, and would like to present it as random initialization of weights - for the convenience of the presentation. For this reason, we will refer to it as to the *zero* loop of the algorithm $l = 0$.

Taking all the above into account, we will define a common notation for a weight of j th observation to the k th cluster in any loop as

$$\phi_{jk}^{(l)} = \begin{cases} u_{jk}^{(0)} & \text{if } l = 0, \\ \phi_{jk} & \text{if } l = 1, \\ \hat{\phi}_{jk} & \text{if } l > 1. \end{cases} \quad (5.8)$$

and define a respective weights matrix as

$$\Phi^{(l)} = [\phi_{jk}^{(l)}]. \quad (5.9)$$

It is now clear why we used the notation $U^{(0)} = \Phi^{(0)}$ in the definition of the first condition of underimpact.

Condition 1: the random initialization $\Phi^{(0)}$ is untoward partial supervision

Regarding the first condition, let us analyze the initial random weights $\Phi^{(0)}$ for a few selected supervised observations from the X_{train} observations matrix that we present in Table 5.2. The initial memberships towards the supervised cluster $\phi_{i,k=s(i)}^{(0)}$ were highlighted in bold for easier reading of Table 5.2. Recall, though, that these values are not adjusted by the partial supervision mechanism, and the use of $\phi_{i,k=s(i)}^{(0)}$ convention is to enable the analysis of the distribution of the initial weights across the clusters for each i th selected observation.

Table 5.2: Values of initialization matrix $\Phi^{(0)}$ for the selected observations.

i	$s(i)$	$\phi_{i,1}^{(0)}$	$\phi_{i,2}^{(0)}$	$\phi_{i,3}^{(0)}$
1	1	0.361	0.352	0.287
2	1	0.055	0.439	0.506
11	2	0.42	0.008	0.573
13	2	0.092	0.456	0.452
21	3	0.526	0.057	0.418
27	3	0.044	0.325	0.631

Source: own work.

The definition of what constitutes an untoward partial supervision will require the use of the notion of a set of indices of non-supervised clusters $\mathcal{E}_i$ for i th supervised observation defined in Eq. (3.30); refer to Section 3.2.4 for more detailed discussion.

A specific distribution of weights for a given i th object is untoward partial supervision when

$$\phi_{i,k=s(i)}^{(0)} \ll \max \left(\{ \phi_{ig}^{(0)} \}_{g \in \mathcal{E}_i} \right), \quad (5.10)$$

that is, when the weight of the i th supervised observation to the supervised cluster $\phi_{i,k=s(i)}^{(0)}$ is much less than the maximum of the weights of this observation to the non-supervised cluster. In such case, this observation $\mathbf{x}_i$ contributes to the supervised cluster to the very small degree, but contributes to the non-supervised cluster to a much higher degree, which contradicts partial supervision and may contribute to the materializing of the underimpact. For example, this is a case of the observation $\mathbf{x}_{i=11}$, where the corresponding $\phi_{i,k=s(i)}^{(0)} = \phi_{11,2}^{(0)} = 0.008$ and is much less than the maximum of the weights to the

non-supervised clusters being equal to $\phi_{11,3}^{(0)} = 0.573$.

As opposed to being untoward, a specific distribution of weights for the i th observation may be also toward partial supervision, when

$$\phi_{i,k=s(i)}^{(0)} \gg \max \left(\{\phi_{ig}^{(0)}\}_{g \in \mathcal{E}_i} \right), \quad (5.11)$$

which is for example the case of $\mathbf{x}_{i=27}$, where $\phi_{i,k=s(i)}^{(0)} = \phi_{27,3}^{(0)} = 0.631$ is much greater than the maximum of the weights to the non-supervised clusters being equal to $\phi_{27,2}^{(0)} = 0.325$.

A specific distribution may be also neutral, when the weight of the supervised observation to the supervised cluster is approximately similar to its weights to the non-supervised cluster. This is the case of the first $\mathbf{x}_{i=1}$ observation.

So far, we have discussed the matter of the distribution of weights of a single observation being toward partial supervision, untoward it, or neutral with respect to it. Condition 1, stating that the random initialization $\Phi^{(0)}$ is untoward partial supervision, is satisfied **when on average, the distributions of the supervised observations in $\Phi^{(0)}$ tend to be untoward partial supervision**. This condition is of stochastic nature, and its occurrence is a first step towards materialization of the underimpact of partial supervision.

The analysis of the initial distributions of weights, and how partial supervision adjusts the weights in these distributions, will be more convenient when we refer to the notion of $\mathcal{E}_i$ as the $(c-1)$ -tuple of indices of non-supervised clusters for the i th observation, not as a set. In our case, this will be the 2-tuple, and we will refer to the r th element of it by $\mathcal{E}_{i,r}$. Table 5.2 represents the results from Table 5.2 using this alternate convention.

Table 5.3: Values of initialization matrix $\Phi^{(0)}$ for the selected observations - alternate convention.

i	$s(i)$	$\mathcal{E}_i$	$\phi_{i,k=s(i)}^{(0)}$	$\phi_{i,\mathcal{E}_{i,1}}^{(0)}$	$\phi_{i,\mathcal{E}_{i,2}}^{(0)}$
1	1	(2, 3)	0.361	0.352	0.287
2	1	(2, 3)	0.055	0.439	0.506
11	2	(1, 3)	0.008	0.42	0.573
13	2	(1, 3)	0.456	0.092	0.452
21	3	(1, 2)	0.418	0.526	0.057
27	3	(1, 2)	0.631	0.044	0.325

Source: own work.

This convention used in Table 5.2 is more convenient for two reasons. Firstly, it allows easier detection of untoward weights' distributions, as the weights of the supervised observations to the supervised clusters are now displayed in the same column. Secondly, it enables transparent data analysis when we analyze entire distributions of the weights

in the following Subsections, as the information about the distribution of weights of supervised observations to the supervised clusters is stored in a single column of a table.

When investigating whether the first condition of the underimpact is satisfied, we will be interested in the differences between the distributions of the weights to the supervised clusters

$$\left\{ \phi_{i,k=s(i)}^{(l)} \right\} \quad (5.12)$$

and the distributions of the weights to the non-supervised clusters

$$\left\{ \phi_{ig}^{(l)} : g \in \mathcal{E}_i \right\} \quad (5.13)$$

without a deeper analysis in the differences between the specific non-supervised clusters $\mathcal{E}_{i,r}$. From the computational point of view, presentation of the data in the form visible in Table 5.3 will enable convenient aggregation and analysis of the data.

Finally, the empirical impact of partial supervision, understood as the degree to which partial supervision affects the values of the decision matrix $\tilde{V}_\alpha$ by the means of adjusting the weights, can also be limited by the distributions of the unsupervised weights

$$\left\{ \phi_{hg}^{(l)} : g \in \{1, \dots, c\} \right\}. \quad (5.14)$$

The distribution of these weights can be of deciding importance especially in a case $M \ll N$, since even if the distribution of the supervised weights is toward partial supervision, the sheer number of unsupervised observations may by chance counteract this effect.

Summary. When studying the phenomenon of underimpact in detail, we will be comparing the distributions of three sets of weights defined in Eq. (5.12), Eq. (5.13) and Eq. (5.14). Note that these sets differ in their cardinalities:

$$\# \left\{ \phi_{i,k=s(i)}^{(l)} \right\} = M, \quad (5.15)$$

$$\# \left\{ \phi_{ig}^{(l)} : g \in \mathcal{E}_i \right\} = M \cdot (c - 1), \quad (5.16)$$

$$\# \left\{ \phi_{hg}^{(l)} : g \in \{1, \dots, c\} \right\} = H \cdot c. \quad (5.17)$$

Also, for convenience of notation, we will be using following abbreviations

$$\left\{ \phi_{ig}^{(l)} : g \in \mathcal{E}_i \right\} \equiv \left\{ \phi_{i,\mathcal{E}_i}^{(l)} \right\}, \quad (5.18)$$

$$\left\{ \phi_{hg}^{(l)} : g \in \{1, \dots, c\} \right\} \equiv \left\{ \phi_{hk}^{(l)} \right\}. \quad (5.19)$$

Condition 2: the value of the scaling factor α is too low to counteract untoward $\Phi^{(0)}$.

In Section 3.2.4, it was established that the impact of partial supervision on the prototypes depends on the loop of the AO algorithm, and – by Eq. (3.27) and Eq. (3.28) – we distinguish between this impact in the first loop $\pi_V^1(\alpha)$, and the impact in any loop $l > 1$ in form of a function $\pi_V^+(\alpha)$.

These functions $\pi_V^1(\alpha)$ and $\pi_V^+(\alpha)$ are defined for the Semi-Supervised Fuzzy c-Means model used in our experiment in Explanation 2, by referring to the appropriate functional form of the weights $\phi_{jk}^{(1)}$ and $\phi_{jk}^{(l>1)}$, correspondingly. In particular, function $\pi_V^1(\alpha)$ is unbounded in case of SSFCMeans, whereas the function $\pi_V^+(\alpha)$ is bounded, and its image is a unit interval $[0, 1]$.

Therefore, in our analysis of the models $M_{0.1}$, M_1 and M_5 – where we study if the value of α was sufficient enough to counteract the untoward $\Phi^{(0)}$ initialization – we will pay special attention to the weights matrix $\Phi^{(1)}$, as high values of α in this case have the potential to successfully counteract the effect of untoward random initialization $\Phi^{(0)}$. Moreover, we will additionally analyze $\Phi^{(L)}$ as an additional step to show the values of the weights for the final decision of the model. We chose the last step L since the full analysis of $\{\Phi^{(l)}\}_{l=1,\dots,L}$ would require substantial additional space, as for the models $M_{0.1}$, M_1 and M_5 , the number of loops L was equal to 21, 17, and 14, correspondingly.

In each case, we will statistically compare the distributions of the weights between the different sets by analyzing boxplots visually summarizing these distributions. We will call the respective plots the Φ -plots.

Analysis of $\Phi^{(1)}$. The analysis of the first loop $l = 1$ will be discussed based on Fig. 5.3, where colours of the boxplots refer to the three sets defined in Eq. (5.12) (orange), Eq. (5.13) (green), and Eq. (5.14) (blue). In addition to the results from respective semi-supervised models, presented in Fig. 5.3b, Fig. 5.3c, and Fig. 5.3d, it will also include the weights $\Phi^{(1)}$ from a corresponding unsupervised Fuzzy c-Means model denoted as M_0 , presented in Fig. 5.3a. The goal of this addition is twofold. Firstly, to assess whether the initial weights distribution $\Phi^{(0)}$ is untoward or toward partial supervision. This is possible since in unsupervised Fuzzy c-Means

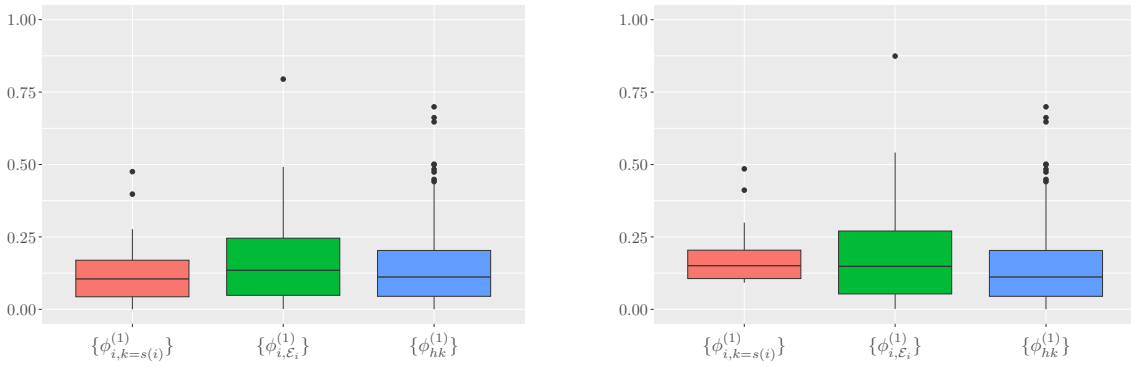
$$\phi_{jk}^{(1)} = \left(u_{jk}^{(0)}\right)^2 = \left(\phi_{jk}^{(0)}\right)^2. \quad (5.20)$$

Hence, the boxplots in Fig. 5.3a display squared initialization weights unimpacted by partial supervision. The split into the three groups in Fig. 5.3a is applied to enable easy comparison with other subfigures, since in this analysis, the weights are not adjusted by

partial supervision.

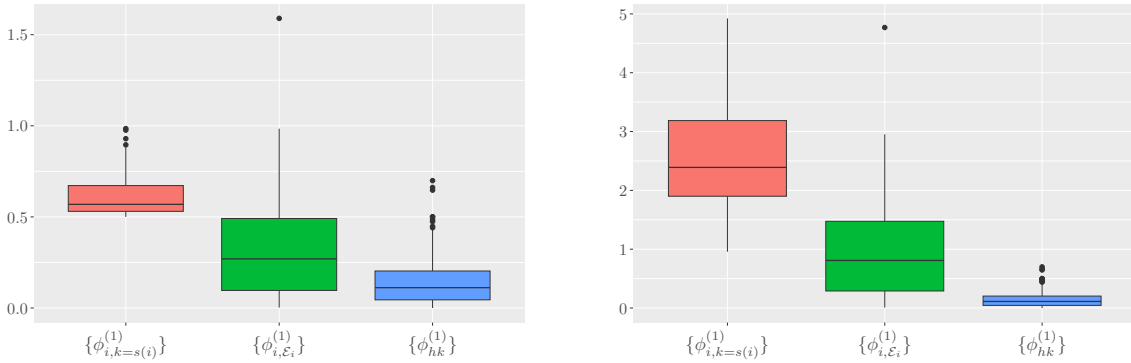
The second aspect of introducing the analysis of $\Phi^{(1)}$ from unsupervised M_0 model is to compare weights $\Phi^{(1)}$ for the semi-supervised models M_α against this baseline, unsupervised weights, to assess how much a given value of the scaling factor α adjusted the respective distributions of the weights. This comparison must be made with reference to the squared initialization weights available in $\Phi^{(1)}$ of the unsupervised Fuzzy c-Means model, since respective formulae for weights in the first loop of Semi-Supervised Fuzzy c-Means model estimation, defined in Eq. (3.59), are transformations of the squared initialization weights.

Figure 5.3: Boxplots of weights $\phi_{jk}^{(1)}$ from the first loop of each of the models.



(a) The initial, random weights $\Phi^{(1)} = (\Phi^{(0)})^2$ from unsupervised M_0 model.

(b) $\Phi^{(1)}$ from semi-supervised $M_{0.1}$ model.



(c) $\Phi^{(1)}$ from semi-supervised M_1 model.

(d) $\Phi^{(1)}$ from semi-supervised M_5 model.

Source: own work. In each of the subfigures, distributions are shown based on the split of observations from X_{train} used to estimate the models into three groups sets defined in Eq. (5.12), Eq. (5.13), Eq. (5.14). Note that the split into the three groups in Fig. 5.3a is applied to enable easy comparison with other subfigures, since in this analysis, the weights are not adjusted by partial supervision.

In Fig. 5.3a, we can observe that on average, the distribution of initial weights $\Phi^{(0)}$ – in a form of squared values in $\Phi^{(1)}$ – is untoward partial supervision. In general, these are the weights from the set $\{\phi_{i,E_i}^{(1)}\}$ that have – on average – higher values, and there are a number of highly influential outliers in the distribution of $\{\phi_{hk}^{(1)}\}$. The distribution of $\{\phi_{i,k=s(i)}^{(1)}\}$

remains concentrated, with less spread around the central tendency measure than in case of the two other sets of weights. Overall, this presents the limited impact of supervised observations on the numerical values of $\tilde{V}^{(l=1)}$, the prototypes matrix estimated in the first loop of the AO algorithm, **that leads to meeting the first condition of the underimpact of partial supervision** (as confirmed in the previous Subsection 5.1.1 when analyzing the iris data).

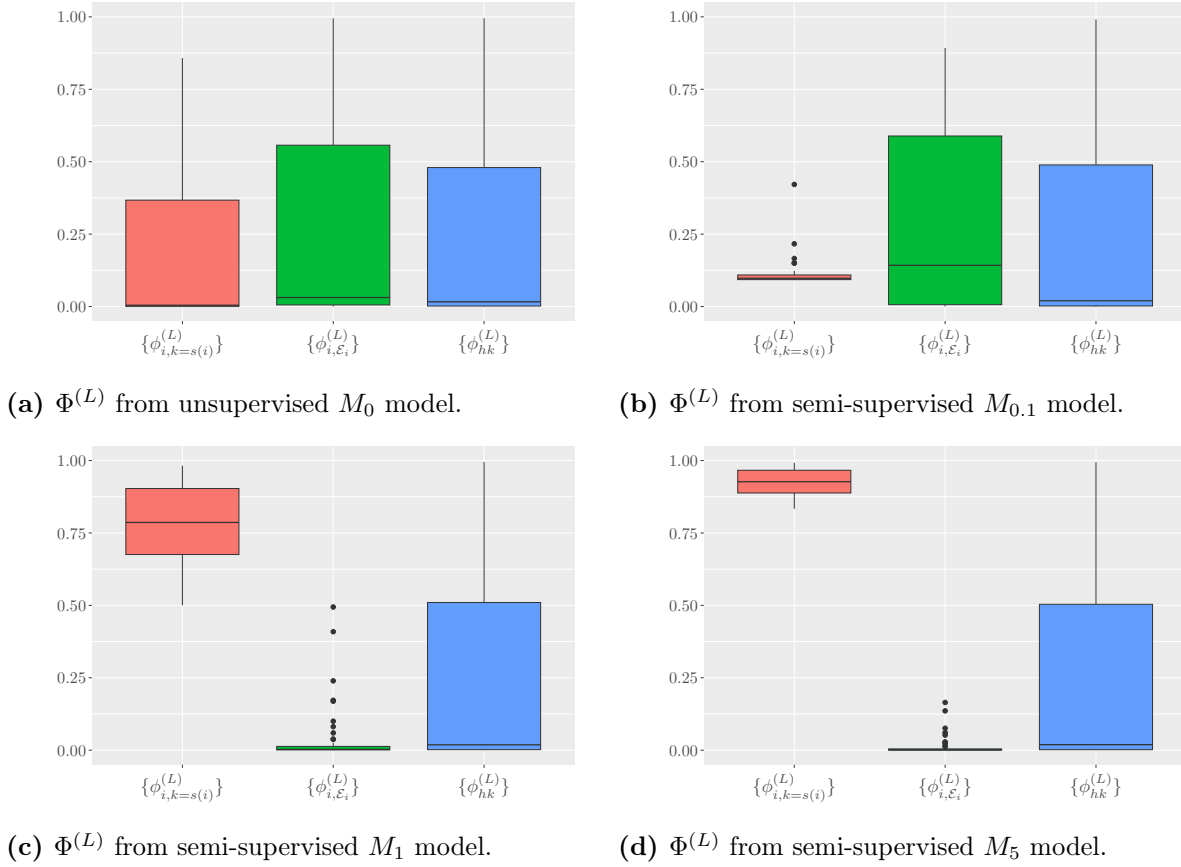
The second Fig. 5.3b shows that the introduction of partial supervision, with the value of the scaling factor $\alpha = 0.1$, changes little. The distribution of $\{\phi_{i,k=s(i)}^{(1)}\}$ has slightly moved towards higher values, but so has the distribution of $\{\phi_{i,\mathcal{E}_i}^{(1)}\}$ as well, and in result the overall picture remains similar. **The value of the scaling factor $\alpha = 0.1$ is too low to counteract the untoward $\Phi^{(0)}$ - in consequence, the second condition for the underimpact of partial supervision is met, and we observe this phenomenon in our experimental results.**

The counteracting effect of higher α values against the untoward realization $\Phi^{(0)}$ is visible in Fig. 5.3c and Fig. 5.3d. It shows a trend where with increasing α , the distribution of $\{\phi_{i,k=s(i)}^{(1)}\}$ starts to dominate the distribution of $\{\phi_{i,\mathcal{E}_i}^{(1)}\}$, which in turn dominates the distribution of $\{\phi_{hk}^{(1)}\}$. This leads to the supervised observations contributing greatly to the numerical values of the corresponding $\tilde{\mathbf{v}}_{i,k=s(i)}^{(l=1)}$ cluster prototypes, which assures the expected order of rows in the final prototypes decision matrix $\tilde{V}$ is kept.

Analysis of $\Phi^{(L)}$. Fig. 5.4 displays the Φ -plots of the final weights $\Phi^{(L)}$ that ultimately decide on the values of the decision matrix $\tilde{V} \equiv \tilde{V}^{(L)}$. The displays for the M_1 and M_5 models – Fig. 5.4c and Fig. 5.4d, accordingly – confirm that the initial domination of $\{\phi_{i,k=s(i)}^{(1)}\}$ over $\{\phi_{i,\mathcal{E}_i}^{(1)}\}$ and $\{\phi_{hk}^{(1)}\}$ that was already established in the first loop of the algorithm (refer to Fig. 5.3c and Fig. 5.3d) was not only sustained, but magnified to the great extent for these models. Indeed, the weights of the supervised observations to the non-supervised clusters $\{\phi_{i,\mathcal{E}_i}^{(L)}\}$ are so small that the corresponding observations nearly do not contribute to the respective clusters at all. This is a desired behaviour, as we want these weights to be as low as possible in order not to contradict the partial supervision.

It is worth noting that the increase of the value of the scaling factor α from $\alpha = 1$ to $\alpha = 5$ between the models M_1 and M_5 did not change the distribution of the weights $\{\phi_{hk}^{(L)}\}$. In both cases, the unsupervised observations contribute to the similar degree - the median value is very close to 0, but the distribution is highly skewed, with the third quartile close to the value of 0.5, and some outlier observations reaching the values of 1. Considering that the increase in the value of the scaling factor did not result in the change of this distribution, a question arises about the limit of the empirical impact of partial supervision that we will be analyzing in Section 5.2.

Figure 5.4: Boxplots of weights $\phi_{jk}^{(L)}$ from the last loop of each of the models.



Source: own work. In each of the subfigures, distributions are shown based on the split of observations from X_{train} used to estimate the models into three groups sets defined in Eq. (5.12), Eq. (5.13), Eq. (5.14). Note that the split into the three groups in Fig. 5.4b is applied to enable easy comparison with other subfigures, since in this analysis, the weights are not adjusted by partial supervision.

Finally, for the $M_{0.1}$ model we can clearly see the ultimate results of the insufficient adjustment of the weights in the first loop of the algorithm. In Fig. 5.4b, it is shown that the distribution $\{\phi_{i,\mathcal{E}_i}^{(L)}\}$ dominates the distribution $\{\phi_{i,k=s(i)}^{(L)}\}$. Although the median values are similar (with the median of the former distribution slightly higher than the median of the latter), the much higher values of the third quartile and the upper whisker of the distribution $\{\phi_{i,\mathcal{E}_i}^{(L)}\}$ than in the case of the distribution $\{\phi_{i,k=s(i)}^{(L)}\}$ confirm the occurrence of the underimpact of partial supervision.

5.1.4 The proposed visualization of underimpact

In the experimental scenario using `iris` dataset presented so far, we made use of the availability of full supervision to derive the expected order of rows in the decision matrix $\tilde{V}$, which is not available in the truly semi-supervised scenario. However, the knowledge about the true prototypes (as a function of the reference partition available in a fully

supervised scenario) is not necessary for detecting the underimpact. **The underimpact of partial supervision is not about quality of clustering, but about the unexpected reordering of rows in the decision matrices $\tilde{V}_{\alpha_1}$, $\tilde{V}_{\alpha_2}$ between certain values of the scaling factor $\alpha_2 > \alpha_1$, as described in the previous Subsection 5.1.3,** since the reordering of rows – a manifestation of counteracting the underimpact – happens when the value of $\alpha_2 > \alpha_1$ becomes sufficiently big to counteract the untoward initialization $\Phi^{(0)}$ by adjusting the weights $\{\phi_{i,k=s(i)}^{(l)}\}$ and $\{\phi_{i,\mathcal{E}_i}^{(l)}\}$ in a way that the distribution of the weights of the supervised observations to the supervised clusters dominates the distribution of the corresponding weights to the non-supervised clusters $\mathcal{E}_i$.

In either case (fully supervised or truly semi-supervised) the detection of underimpact requires numerical investigation of the decision matrices $\tilde{V}_\alpha$ from the models fitted with the values of the scaling factor α from a specific grid $\mathcal{A}$. A direct investigation of such matrices would be a tedious task, especially in cases when $c \gg 2$ and/or $p \gg 2$. For this reason, we introduce an easy to read diagnostic tool in form of a plot that we call the **underimpact-plot**, or **U-plot** in short.

The main assumption of the proposed approach is that small changes in the value of the scaling factor α are expected to cause small changes in the decision prototypes $\tilde{V}_\alpha$, *ceteris paribus* (i.e., with all other experimental factors unchanged). Underimpact is detected when the change between the two consecutive decision prototypes matrices is much bigger – of a clearly higher magnitude – than the other changes; this is due to the reordering of the rows between these prototypes matrices.

Formalizing this idea, let α_1, α_2 s.t.

$$\alpha_2 = \alpha_1 + \epsilon, \quad \epsilon > 0, \quad (5.21)$$

denote the two consecutive values of the scaling factor α , where the change between the values of α_2 and α_1 is arbitrarily small.

We will measure the change between the corresponding decision matrices $\tilde{V}_{\alpha_1}$ and $\tilde{V}_{\alpha_2}$ using the Frobenius matrix norm defined as

$$\|V\|_F = \sqrt{\sum_{k=1}^c \sum_{r=1}^p v_{kr}^2}, \quad (5.22)$$

expecting that for the above α_1, α_2 , we obtain a correspondingly arbitrarily small change

$$\|\tilde{V}_{\alpha_2} - \tilde{V}_{\alpha_1}\|_F = \varepsilon, \quad \varepsilon > 0. \quad (5.23)$$

Any difference between the two consecutive matrices $\tilde{V}_{\alpha_1}, \tilde{V}_{\alpha_2}$ being of a clearly higher

magnitude, i.e.,

$$\|\tilde{V}_{\alpha_2} - \tilde{V}_{\alpha_1}\|_F \gg \varepsilon, \quad \varepsilon > 0, \quad (5.24)$$

would indicate that the models fitted with values of the scaling factor $\alpha \leq \alpha_1$ may have suffered from the underimpact of partial supervision, and the use of the value α_2 reordered the rows in $\tilde{V}_{\alpha_2}$ as compared with $\tilde{V}_{\alpha \leq \alpha_1}$.

In practice, one does not consider a continuous spectrum of α values, but operates on the finite grid $\mathcal{A} \subset \mathcal{R}_+$, formally a q -tuple $\mathcal{A} = (\alpha_r)_{r=1,\dots,q}$ with r indexing its elements. To simplify notation, we will define the Frobenius norm calculated for two consecutive elements of the grid as

$$\delta_r = \|\tilde{V}_{\alpha_r} - \tilde{V}_{\alpha_{r-1}}\|_F, \quad r = 2, \dots, q. \quad (5.25)$$

In this scenario with a finite grid $\mathcal{A}$, the differences between two consecutive values α_r, α_{r-1} are not arbitrarily small, and hence, the differences between consecutive $\tilde{V}_{\alpha_{r-1}}, \tilde{V}_{\alpha_r}$ expressed by δ_r are not expected to be arbitrarily small, either. In fact, the less dense the grid, the higher the potential differences δ_r might be. Nevertheless, they are expected to be of similar magnitude, whereas the occurrence of underimpact is exhibited by the value of δ_r of a much higher magnitude, that can be treated as anomaly or an outlier with respect to the entire distribution $\{\delta_r\}$.

The proposed diagnostic plot called U-plot is a graphical display of pairs $(\alpha_r, \delta_r), r > 1$, aimed at finding a threshold value $\alpha^\dagger$ such that for $\alpha \geq \alpha^\dagger$, the underimpact is not occurring, and for $\alpha < \alpha^\dagger$, the underimpact is occurring. The investigation of the U-plot may also reveal no underimpact at all, as recall this phenomenon is of stochastic nature related to the random initialization of the weights $\Phi^{(0)}$.

In the next Section 5.1.5, we present three concrete U-plots for three different datasets, each of these underimpact-plots contributing some unique considerations to the problem of the underimpact and its detection. Two of the datasets are fully supervised (with the simulated partial supervision), and one is truly semi-supervised. Even though the U-plot does not require the availability of full supervision to detect underimpact, the knowledge about reference partitioning will ease the discussion on certain aspects of detecting underimpact that would be possible, but much harder to establish considering only partially supervised data.

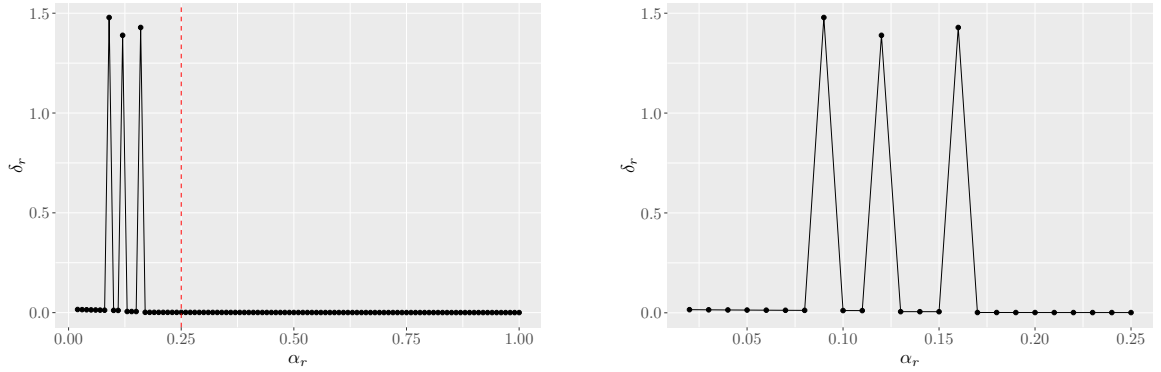
5.1.5 Empirical analysis of datasets with U-plots

Underimpact plot for iris dataset

We will start with creating the U-plot for the `iris` data used throughout this thesis. Recall that in Subsection 5.1.3 we checked that the first condition needed for the underimpact to materialize, the initialization $\Phi^{(0)}$ being untoward partial supervision, was satisfied. Underimpact was detected between $\tilde{V}_{0,1}$ and $\tilde{V}_1$ matrices, in a sense that the value $\alpha = 1$ counteracted the untoward weights initialization $\Phi^{(0)}$, whereas the value $\alpha = 0.1$ did not. This leaves an open question whether the matrices $\tilde{V}_{0.1 \leq \alpha < 1}$ were affected by underimpact of partial supervision.

To address this question, we fit 100 models for the grid $\mathcal{A} = (0.01, 0.02, 0.03, \dots, 0.99, 1)$, and create the U-plot displayed in Fig. 5.5a. For the majority of the values of the scaling factor we observe $\delta_r \approx 0$, but for the three values, the results are much bigger, approaching the value of 1.5. For convenience, in Fig. 5.5b, we display the results from the experiment only for $\mathcal{A}' = (\alpha_r \in \mathcal{A} : \alpha_r \leq 0.25)$. Note that while the first element of $\mathcal{A}$ is the value $\alpha_{r=1} = 0.01$, the first pair in the U-plot is a pair $(\alpha_{r=2}, \delta_{r=2})$ due to the nature of the δ_r defined in Eq. (5.25) (it is a difference between two consecutive values of the scaling factor in the respective grid $\mathcal{A}$).

Figure 5.5: Underimpact plot for the matrix X_{train} , `iris` dataset.



(a) Values of δ_r for $\mathcal{A} = (0.02, 0.03, \dots, 0.99, 1)$. (b) Values of δ_r for $\mathcal{A}' = (\alpha_r \in \mathcal{A} : \alpha_r \leq 0.25)$.

Source: own work. The red, dashed vertical line in Fig. 5.5a denotes the upper limit defining the range of (α_r, δ_r) pairs displayed in Fig. 5.5b.

In Fig. 5.5b, we can see that there are three candidate values for $\alpha^\dagger$, for which $\delta_r \gg 0$; indeed, each of the candidate values $\alpha_r \in \{0.09, 0.11, 0.16\}$ corresponds to a particular reordering of the rows of the prototypes, but without an investigation of the corresponding decision matrices no further conclusions can be drawn. In general, for c clusters, there are $c!$ possible permutations of the order of prototypes in the $\tilde{V}$. For the `iris` data, there

are specifically $3! = 6$ permutations:

$$(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1).$$

Making use of the availability of the full supervision, and the associated knowledge of the expected order of the prototypes defined in Eq. (5.6), we can analyze the order of the prototypes in each of the estimated models from the grid $\mathcal{A}'$. Let $\text{Ord}(\cdot)$ be a function of the decision prototypes matrix $\tilde{V}_\alpha$ such that

$$\text{Ord}(\tilde{V}) = (s_1, s_2, s_3) \quad \text{s.t.} \quad \tilde{v}_{s_1,1} < \tilde{v}_{s_2,1} < \tilde{v}_{s_3,1}. \quad (5.26)$$

From Eq. (5.6) we know that the expected order is equal to

$$(1, 2, 3).$$

Table 5.4 presents the full analysis of the considered scenario, displaying the pairs (α_r, δ_r) obtained for the models fitted to the grid $\mathcal{A}'$, the order in each estimated decision matrix (according to the function $\text{Ord}(\cdot)$ defined in Eq. 5.26), the information whether a given α_r is considered a candidate for the threshold value $\alpha^\dagger$, and the information whether the order of prototypes in the decision matrix $\tilde{V}_\alpha$ complies with the expected order of the prototypes.

Table 5.4: The analysis of the order of first coordinates of the prototypes in $\tilde{V}_{\alpha_r}$ for iris dataset.

α_r	δ_r	$\text{Ord}(\tilde{V}_{\alpha_r})$	$\alpha^\dagger$ cand.	compliant	α_r	δ_r	$\text{Ord}(\tilde{V}_{\alpha_r})$	$\alpha^\dagger$ cand.	compliant
0.01	NA	3, 2, 1			0.14	0.006	1, 3, 2		
0.02	0.016	3, 2, 1			0.15	0.006	1, 3, 2		
0.03	0.015	3, 2, 1			0.16	1.429	1, 2, 3	✓	✓
0.04	0.014	3, 2, 1			0.17	0.001	1, 2, 3		✓
0.05	0.014	3, 2, 1			0.18	0.001	1, 2, 3		✓
0.06	0.013	3, 2, 1			0.19	0.001	1, 2, 3		✓
0.07	0.013	3, 2, 1			0.20	0.001	1, 2, 3		✓
0.08	0.012	3, 2, 1			0.21	0.001	1, 2, 3		✓
0.09	1.478	3, 1, 2	✓		0.22	0.001	1, 2, 3		✓
0.10	0.011	3, 1, 2			0.23	0.001	1, 2, 3		✓
0.11	0.011	3, 1, 2			0.24	0.001	1, 2, 3		✓
0.12	1.389	1, 3, 2	✓		0.25	0.001	1, 2, 3		✓
0.13	0.006	1, 3, 2			0.26	0.001	1, 2, 3		✓

Source: own work. Header “ $\alpha^\dagger$ cand.” means “Candidate value for the threshold $\alpha^\dagger$ ”. Header “compliant” means “ $\text{Ord}(\tilde{V}_{\alpha_r})$ as in Eq. (5.6)”.

In Table 5.4, we observe that the first candidate value, $\alpha_r = 0.09$, refers to the re-ordering

$$\text{Ord}(\tilde{V}_{0.08}) = (3, 2, 1) \rightarrow \text{Ord}(\tilde{V}_{0.09}) = (3, 1, 2),$$

the second candidate value, $\alpha_r = 0.12$, refers to the reordering

$$\text{Ord}(\tilde{V}_{0.11}) = (3, 1, 2) \rightarrow \text{Ord}(\tilde{V}_{0.12}) = (1, 3, 2),$$

and it is only the last candidate value $\alpha_r = 0.16$ that refers to the reordering

$$\text{Ord}(\tilde{V}_{0.15}) = (1, 3, 2) \rightarrow \text{Ord}(\tilde{V}_{0.16}) = (1, 2, 3)$$

that is the expected order of the prototypes and that remains for any $\tilde{V}_{\alpha_r \geq 0.16}$.

Based on the above empirical analysis, we formulate a following general rule.

Rule 1 (Selection of the threshold value $\alpha^\dagger$). *To avoid underimpact, select the threshold value $\alpha^\dagger$ to be the biggest value of the scaling factor α out of potential candidates, i.e. such values α_r that the corresponding δ_r remains outlier with respect to the entire distribution $\{\delta_r\}$.*

In a truly semi-supervised case, one cannot establish the expected order of prototypes, and the decision whether a given value δ_r is exhibiting the occurrence of underimpact may not be unequivocal, since only one out of lc possible permutations of the order of prototypes is the desired one. Nevertheless, taking into account that the underimpact occurs for *too small* values of the scaling factor α , applying Rule 1 and selecting the biggest value out of the potential candidates for the threshold value $\alpha^\dagger$ maximizes the chances of selecting such value $\alpha^\dagger$, that there are no further reordering of rows for any $\tilde{V}_{\alpha > \alpha^\dagger}$ (which means a successful counteracting of underimpact).

Having performed the above analysis, **we apply Rule 1** and decide that

$$\alpha_{\text{iris}}^\dagger = 0.16. \quad (5.27)$$

Underimpact plot for bipolar dataset

Let us now consider a truly partially supervised dataset **bipolar**, which is described in detail by Kmita et al. (2022). It contains data for a single patient diagnosed with bipolar disorder that was participating in a prospective observational study. The unsupervised data about acoustic features of patient's speech were collected from a dedicated mobile phone application installed on the patient's smartphone. In particular, **bipolar** dataset contains five voice features that describe different types of jitter, shimmer, and spectrum of the signal (spectral flux and spectral centroid). Unsupervised observations $\mathbf{x}_j \in \mathcal{R}^5$ summarise specific calls (a mean value of each voice feature measured throughout the concrete phone call).

Labels for partial supervision were obtained from scheduled stationary psychiatric assessments. At each visit, psychiatrist assigned one of the four labels denoting the phase of the disease:

$$\mathcal{Y} = \{y^1 = \text{depression}, y^2 = \text{mixed}, y^3 = \text{euthymia}, y^4 = \text{dysfunction}\}, \quad (5.28)$$

where **euthymia** refers to the good condition of the patient, and **mixed** state is a disease phease combining the features of **depression** and **dysfunction** states. This obtained label is extrapolated onto the unsuperivsed data by the means of the so-called *ground truth extrapolation*: each observation $\mathbf{x}_j$ recorded in the timeframe spanning from 7 days before the visit up until 2 days after the visit is assigned the label obtained during the visit. The summary of the number of observations in this dataset is presented in Table 5.5.

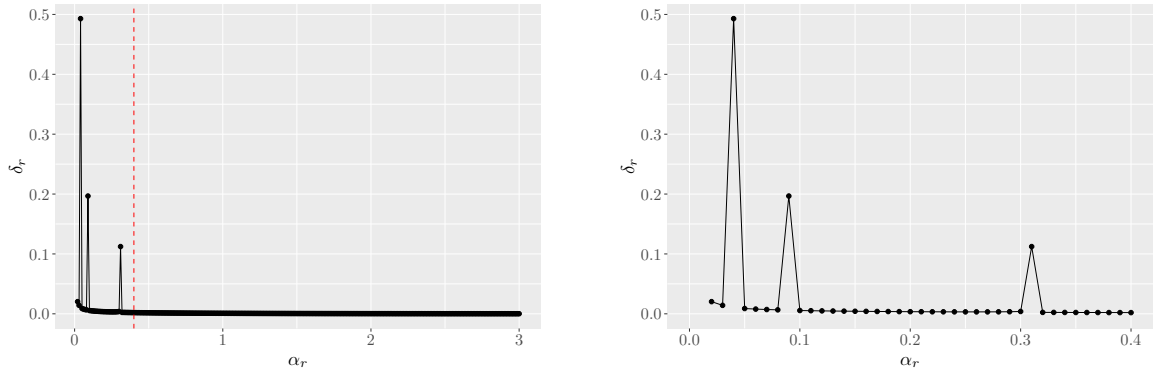
Table 5.5: Total number of unsupervised and supervised observations, by label, in **bipolar** dataset.

labeling	number
unsupervised	1034
$y^1 = \text{depression}$	58
$y^2 = \text{mixed}$	55
$y^3 = \text{euthymia}$	85
$y^4 = \text{dysfunction}$	63
total supervised	261

Source: own work based on (Kmita et al., 2022).

The underimpact analysis of the truly semi-supervised **bipolar** dataset will be very similar to the one performed for the fully supervised **iris** dataset, for which partial supervision was artificially simulated. Fig. 5.6a displays the underimpact-plot for the SSFCMeans models fitted to the grid $\mathcal{A} = (0.01, 0.02, \dots, 2.99, 3)$ for the **bipolar** dataset. For the majority of the values of the scaling factor we observe $\delta_r \approx 0$, but for the three values, the results are much bigger, achieving the values of approximately 0.5, 0.2, and 0.1, respectively. For convenience, in Fig. 5.6b, we display the results from the experiment only for $\mathcal{A}' = (\alpha_r \in \mathcal{A} : \alpha_r \leq 0.4)$.

The three candidate values include $\alpha_{r=4} = 0.04$, $\alpha_{r=9} = 0.09$, and $\alpha_{r=31} = 0.31$. Let us investigate the first candidate value, and the difference between the matrices $\tilde{V}_{\alpha_r=0.03}$ and $\tilde{V}_{\alpha_r=0.04}$.

Figure 5.6: Underimpact plot for the **bipolar** dataset.

(a) Values of δ_r for $\mathcal{A} = (0.01, 0.02, \dots, 2.99, 3)$. (b) Values of δ_r for $\mathcal{A}' = (\alpha_r \in \mathcal{A} : \alpha_r \leq 0.4)$.

Source: own work. The red, dashed vertical line in Fig. 5.6a denotes the upper limit defining the range of (α_r, δ_r) pairs displayed in Fig. 5.6b.

$$\tilde{V}_{0.03} = \begin{pmatrix} 0.273 & 0.194 & 0.369 & 0.033 & 0.523 \\ 0.165 & 0.108 & 0.31 & 0.021 & 0.51 \\ 0.112 & 0.069 & 0.222 & 0.013 & 0.454 \\ 0.514 & 0.383 & 0.481 & 0.09 & 0.426 \end{pmatrix}, \tilde{V}_{0.04} = \begin{pmatrix} 0.513 & 0.383 & 0.481 & 0.091 & 0.425 \\ 0.165 & 0.108 & 0.31 & 0.021 & 0.51 \\ 0.114 & 0.07 & 0.226 & 0.013 & 0.455 \\ 0.268 & 0.19 & 0.366 & 0.032 & 0.524 \end{pmatrix}. \quad (5.29)$$

In Eq. (5.29), we can see a clear reordering that affected the first and the fourth rows of the matrices. It becomes even more visible if we present the same matrices with the precision up to two decimal places in Eq. (5.30).

$$\tilde{V}_{0.03} = \begin{pmatrix} 0.27 & 0.19 & 0.37 & 0.03 & 0.52 \\ 0.17 & 0.11 & 0.31 & 0.02 & 0.51 \\ 0.11 & 0.07 & 0.22 & 0.01 & 0.45 \\ 0.51 & 0.38 & 0.48 & 0.09 & 0.43 \end{pmatrix}, \quad \tilde{V}_{0.04} = \begin{pmatrix} 0.51 & 0.38 & 0.48 & 0.09 & 0.43 \\ 0.17 & 0.11 & 0.31 & 0.02 & 0.51 \\ 0.11 & 0.07 & 0.23 & 0.01 & 0.46 \\ 0.27 & 0.19 & 0.37 & 0.03 & 0.52 \end{pmatrix}. \quad (5.30)$$

Investigating Eq. (5.30), we can see that the order of the prototypes can be in this case identified by considering only the first coordinate $\tilde{v}_{k,1}(\alpha)$, since the differences between the values 0.17, 0.11, 0.27, 0.51 (occurring in both matrices) are of sufficient magnitude to approximately sustain similar between the decision matrices corresponding to the $\alpha_r = 0.04$ and the remaining $\alpha_r = 0.09$ and $\alpha_r = 0.31$ candidate values. Therefore, in this case, we opt for identifying the order of the prototypes by analyzing only the first coordinate of each prototypes matrix. We thus define $\text{Ord}(\cdot)$ function

$$\text{Ord}(\tilde{V}) = (s_1, s_2, s_3, s_4) \quad \text{s.t.} \quad \tilde{v}_{s_1,1} < \tilde{v}_{s_2,1} < \tilde{v}_{s_3,1} < \tilde{v}_{s_4,1}, \quad (5.31)$$

and display the results for the models fitted to the grid $\mathcal{A}'$ in Table 5.6, which looks analogous to the Table 5.4 with the difference that this time, we do not know the expected order of the prototypes.

Table 5.6: The analysis of the order of first coordinates of the prototypes in $\tilde{V}_{\alpha_r}$ for bipolar dataset.

α_r	δ_r	$\text{Ord}(\tilde{V}_{\alpha_r})$	$\alpha^\dagger$ cand.	α_r	δ_r	$\text{Ord}(\tilde{V}_{\alpha_r})$	$\alpha^\dagger$ cand.
0.01	NA	3, 2, 1, 4		0.21	0.003	3, 4, 2, 1	
0.02	0.020	3, 2, 1, 4		0.22	0.003	3, 4, 2, 1	
0.03	0.014	3, 2, 1, 4		0.23	0.003	3, 4, 2, 1	
0.04	0.493	3, 2, 4, 1	✓	0.24	0.003	3, 4, 2, 1	
0.05	0.009	3, 2, 4, 1		0.25	0.003	3, 4, 2, 1	
0.06	0.008	3, 2, 4, 1		0.26	0.003	3, 4, 2, 1	
0.07	0.007	3, 2, 4, 1		0.27	0.003	3, 4, 2, 1	
0.08	0.007	3, 2, 4, 1		0.28	0.003	3, 4, 2, 1	
0.09	0.197	3, 4, 2, 1	✓	0.29	0.003	3, 4, 2, 1	
0.10	0.005	3, 4, 2, 1		0.30	0.004	3, 4, 2, 1	
0.11	0.005	3, 4, 2, 1		0.31	0.112	4, 3, 2, 1	✓
0.12	0.005	3, 4, 2, 1		0.32	0.002	4, 3, 2, 1	
0.13	0.005	3, 4, 2, 1		0.33	0.002	4, 3, 2, 1	
0.14	0.004	3, 4, 2, 1		0.34	0.002	4, 3, 2, 1	
0.15	0.004	3, 4, 2, 1		0.35	0.002	4, 3, 2, 1	
0.16	0.004	3, 4, 2, 1		0.36	0.002	4, 3, 2, 1	
0.17	0.004	3, 4, 2, 1		0.37	0.002	4, 3, 2, 1	
0.18	0.004	3, 4, 2, 1		0.38	0.002	4, 3, 2, 1	
0.19	0.004	3, 4, 2, 1		0.39	0.002	4, 3, 2, 1	
0.20	0.004	3, 4, 2, 1		0.40	0.002	4, 3, 2, 1	

Source: own work. Header “ $\alpha^\dagger$ cand.” means “Candidate value for the threshold $\alpha^\dagger$ ”.

In Table 5.6, we observe that the first candidate value, $\alpha_r = 0.04$, refers to the re-ordering

$$\text{Ord}(\tilde{V}_{0.03}) = (3, 2, 1, 4) \rightarrow \text{Ord}(\tilde{V}_{0.04}) = (3, 2, 4, 1),$$

the second candidate value, $\alpha_r = 0.09$, refers to the reordering

$$\text{Ord}(\tilde{V}_{0.08}) = (3, 2, 4, 1) \rightarrow \text{Ord}(\tilde{V}_{0.09}) = (3, 4, 2, 1),$$

and it is only the last candidate value $\alpha_r = 0.31$ that refers to the reordering

$$\text{Ord}(\tilde{V}_{0.30}) = (3, 4, 2, 1) \rightarrow \text{Ord}(\tilde{V}_{0.31}) = (4, 3, 2, 1)$$

which remains stable for any $\tilde{V}_{\alpha_r \geq 0.31}$. We confirm the difference in the order of the prototypes between matrices $\tilde{V}_{\alpha_r=30}$ and $\tilde{V}_{\alpha_r=31}$ in Eq. (5.32).

$$\tilde{V}_{0.30} = \begin{pmatrix} 0.44 & 0.33 & 0.44 & 0.09 & 0.44 \\ 0.26 & 0.18 & 0.36 & 0.03 & 0.52 \\ 0.14 & 0.09 & 0.27 & 0.02 & 0.46 \\ 0.16 & 0.11 & 0.3 & 0.02 & 0.52 \end{pmatrix}, \quad \tilde{V}_{0.31} = \begin{pmatrix} 0.44 & 0.33 & 0.44 & 0.09 & 0.44 \\ 0.26 & 0.18 & 0.36 & 0.03 & 0.52 \\ 0.17 & 0.11 & 0.31 & 0.02 & 0.51 \\ 0.13 & 0.08 & 0.25 & 0.01 & 0.47 \end{pmatrix}. \quad (5.32)$$

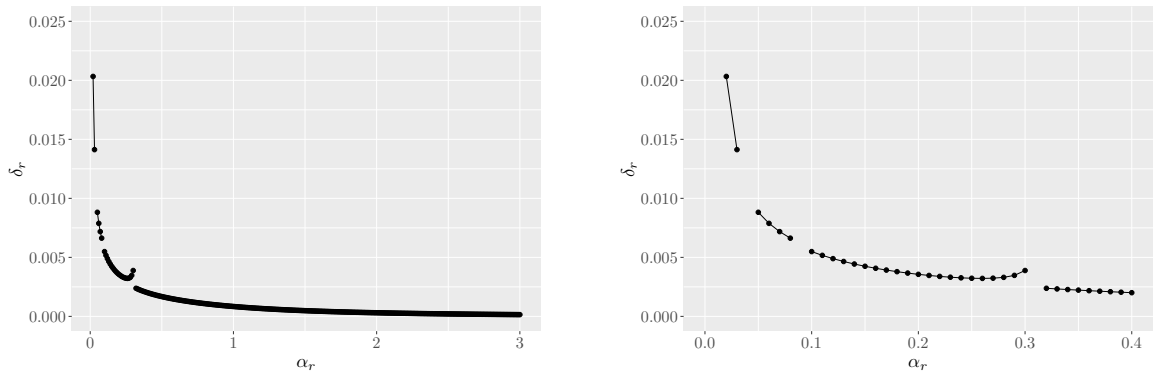
Having performed the above analysis, **we apply Rule 1** and decide that

$$\alpha_{\text{bipolar}}^\dagger = 0.31. \quad (5.33)$$

The trend in the U-plot for bipolar dataset.

Let us have a closer look at the values of pairs (α_r, δ_r) from Fig. 5.6, excluding the abnormally high values for the three candidate values for the threshold $\alpha^\dagger$. We display such data with the capped value of δ_r at the value of 0.025 in Fig. 5.7. Fig. 5.7a presents the values for $\mathcal{A}$, and Fig. 5.7b presents the values for the models fitted to the $\mathcal{A}'$ grid. In both figures, there are three visible gaps which correspond to the excluded data for the $\alpha^\dagger$ candidate values as presented in Table 5.6.

Figure 5.7: Underimpact plot for the bipolar dataset capped at $\max(\delta_r) = 0.025$.



(a) Values of δ_r for $\mathcal{A} = (0.01, 0.02, \dots, 2.99, 3)$. (b) Values of δ_r for $\mathcal{A}' = (\alpha_r \in \mathcal{A} : \alpha \leq 0.4)$.

Source: own work.

Recall that the idea for the U-plot is based on the assumption that arbitrarily small changes in α will result in correspondingly arbitrarily small changes in δ_r , and that the underimpact would result in the value of δ_r being of a much higher magnitude. Clearly, in Fig. 5.7b the changes are small in their magnitude (as compared with the values of δ_r for the candidates to be $\alpha^\dagger$), but there seems to be an evident non-linear, decreasing trend visible in the ordered pairs $(\alpha_r, \delta_r)_{r=1, \dots, q}$. There also seems to be a limit to the potential

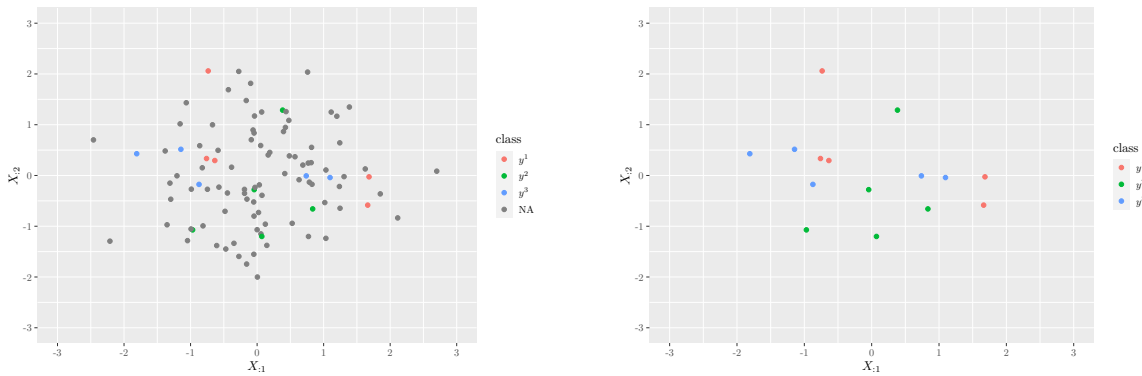
of the scaling factor α to change the values of decision matrix, as the differences nearly vanish around the $\max(\alpha_r) = 3$ displayed in Fig. 5.7a.

A question naturally arises about the functional form of the empirical impact of partial supervision, and whether it has an upper limit, given there is no upper bound for the values of the scaling factor $\alpha > 0$. We analyze these issues in the following Section 5.2 where we argue that the U-plot is not the perfect tool for the analysis of the empirical impact of partial supervision $\tilde{\pi}(\alpha)$ defined in Section 3.2.3, and the limit of this impact.

Underimpact plot for chaos dataset.

Last but not least, let us consider a proprietary, simulated scenario that we refer to as the **chaos** dataset to stress its nature. We will simulate $N = 100$ observations from a multivariate normal distribution $\mathcal{N}_2(\mu = [0, 0]^T, \Sigma = \text{diag}(1))$, and create random supervision for 15 observations, labeling 5 observations as y^1 , 5 observations as y^2 , and 5 observations as y^3 . This scenario is called **chaos** since the simulated partial supervision does not reflect any prior knowledge about the true class of the considered objects. All the objects supervised to various classess – reflected in the difference of the assigned labels y^1, y^2, y^3 – are in fact generated from the same, compact multivariate probability distribution. Fig. 5.8a displays all the observations together with the information about the label, and Fig. 5.8b displays only the 15 supervised observations for convenience – it is clearly visible that one cannot easily separate the supervised observations in the considered $\mathcal{R}^2$ space.

Figure 5.8: Simulated **chaos** dataset.



(a) Unsupervised and supervised observations. (b) Supervised observations.

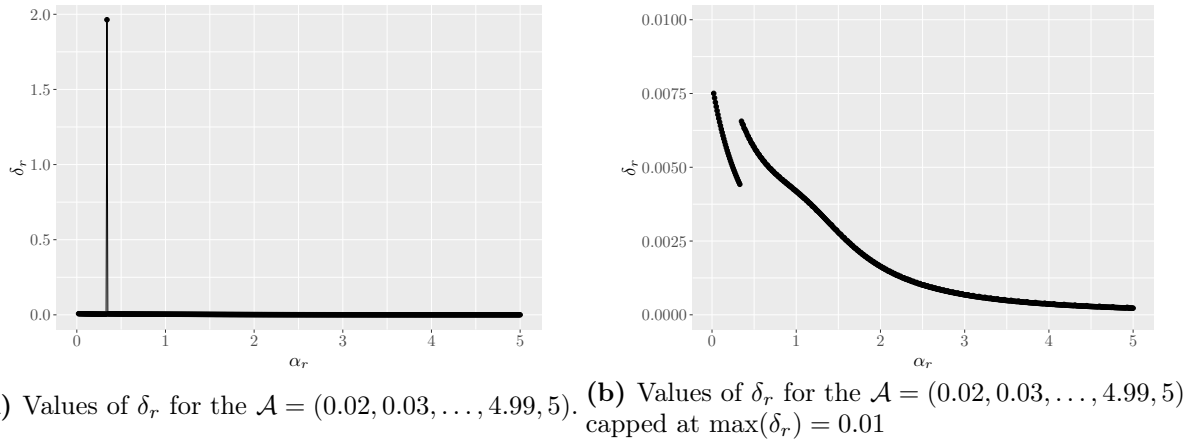
Source: own work. Fig. 5.8b is a variant of Fig. 5.8a with unsupervised observations removed from the display.

This curated scenario aims at stressing that the underimpact phenomenon is not related to the quality of clustering – it can occur in a situation where we do not expect the

partial supervision to improve the results of clustering in any way as compared with the unsupervised case. Underimpact of partial supervision is a phenomenon resulting from the value of the scaling factor α being too small to counteract the random initialization of weights $\Phi^{(0)}$ being untoward partial supervision - *whatever* this partial supervision is. **Therefore, the task of detecting underimpact, and selecting the threshold $\alpha^\dagger$ should be considered as in initial, pre-processing step in semi-supervised fuzzy clustering.**

To analyze the **chaos** data, we fitted models for the grid $\mathcal{A} = (0.01, 0.02, \dots, 4.99, 5)$. Fig. 5.9a presents the corresponding U-plot.

Figure 5.9: Underimpact plot for the **chaos** dataset.



Source: own work.

In Fig. 5.9a we can clearly see a single candidate for the $\alpha^\dagger$ being equal to $\alpha_{r=34} = 0.34$, for which the associated $\delta_{r=34} \approx 2$. When we investigate the corresponding matrices $\tilde{V}_{\alpha_r=0.33}$ and $\tilde{V}_{\alpha_r=0.34}$

$$\tilde{V}_{0.33} = \begin{pmatrix} -0.16 & 0.93 \\ -0.39 & -0.8 \\ 0.83 & 0.01 \end{pmatrix}, \quad \tilde{V}_{0.34} = \begin{pmatrix} 0.89 & 0.11 \\ -0.32 & -0.85 \\ -0.36 & 0.82 \end{pmatrix}, \quad (5.34)$$

we can confirm that a reordering occurred between the first and the third row of each matrix. In this scenario, the underimpact analysis is meaningless, since we do know the dummy character of the partial supervision, but hadn't we known that, we should have decided according to Rule 1 that $\alpha_{\text{chaos}}^\dagger = 0.34$.

The second presented Fig. 5.9b displays the same data as in Fig. 5.9a with one exception - the $\alpha_r = 0.34$ corresponding to the single candidate for the threshold value was excluded, and thy y axis was capped at the level of 0.01. We can see a similar

non-linear, decreasing, gradually vanishing trend in the plotted pairs of (α_r, δ_r) as in the case of **bipolar** data. The interpretation of this pattern is easier than in the case of the corresponding Fig. 5.7b, since in the case of **chaos** dataset, we do know the underlying data generation process, and the fact that partial supervision does not reflect any prior knowledge about the supervised observations. **Introducing partial supervision in any form makes the SSFC optimization algorithm try to adapt to the supervision structure - with increasing value of α , the consecutive differences are smaller, converging towards 0, which suggests there is a limit of this adaptation.** The matter of this limit is analyzed in more detail in the following Section 5.2.

5.2 Overimpact and empirical impact of partial supervision function

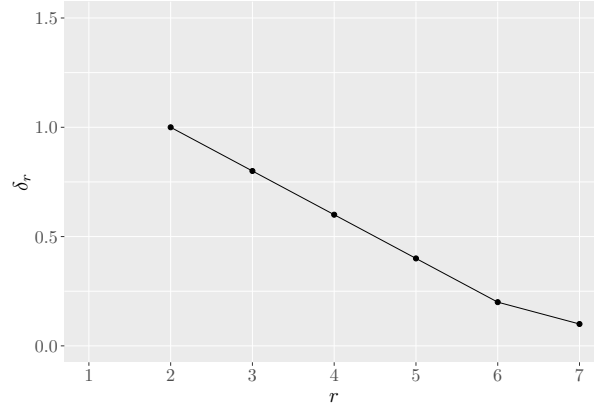
In the previous Section, some of the presented underimpact-plots showed clear non-linear, decreasing, vanishing trend in the values of $(\delta_r)_{r=2,\dots,q}$. However, it was stated that underimpact-plot is not an adequate tool to estimate the empirical impact of partial supervision $\tilde{\pi}(\alpha)$ defined in Section 3.2.3. The justification for this statement is that the U-plot is a display of pairs (α_r, δ_r) , where r is indexing the elements of the given grid $\mathcal{A} = (\alpha_r)_{r=1,\dots,q}$. Each r th point in the display thus directly relates only to the models M_{α_r} and $M_{\alpha_{r-1}}$; in the extreme scenario, we could have a decreasing trend in the U-plot without the actual changes in the decision matrix. To illustrate this, let us consider an exemplary scenario presented in Table 5.7.

Table 5.7: Exemplary scenario showing that underimpact plot is not suited to analyze empirical impact of partial supervision.

$\tilde{V}_{r=1}$	$\tilde{V}_{r=2}$	$\tilde{V}_{r=3}$	$\tilde{V}_{r=4}$	$\tilde{V}_{r=5}$	$\tilde{V}_{r=6}$	$\tilde{V}_{r=7}$
$\begin{pmatrix} 3 & 3 \\ 3 & 3 \end{pmatrix}$	$\begin{pmatrix} 3.5 & 3.5 \\ 3.5 & 3.5 \end{pmatrix}$	$\begin{pmatrix} 3.1 & 3.1 \\ 3.1 & 3.1 \end{pmatrix}$	$\begin{pmatrix} 2.8 & 2.8 \\ 2.8 & 2.8 \end{pmatrix}$	$\begin{pmatrix} 3 & 3 \\ 3 & 3 \end{pmatrix}$	$\begin{pmatrix} 3.1 & 3.1 \\ 3.1 & 3.1 \end{pmatrix}$	$\begin{pmatrix} 3.05 & 3.05 \\ 3.05 & 3.05 \end{pmatrix}$

Source: own work.

Fig. 5.10 presents the associated underimpact-plot. Using the underimpact-plot to estimate the empirical impact of partial supervision $\tilde{\pi}(\alpha)$, one could conclude from the Fig. 5.10 that it is a decreasing function of α , representing similar trends as shown in Fig. 5.7b for **bipolar** dataset and Fig. 5.9b for **chaos** dataset. However, from Table 5.7, we know that the decision matrix $\tilde{V}_{r=7}$ is very similar to $\tilde{V}_{r=1}$, and the sequence $(\tilde{V}_r)_{r=1,\dots,7}$ resembles a random walk. Thus, U-plot can lead to incorrect conclusions in such extreme cases.

Figure 5.10: U-plot for the exemplary scenario presented in Table 5.7.

Source: own work.

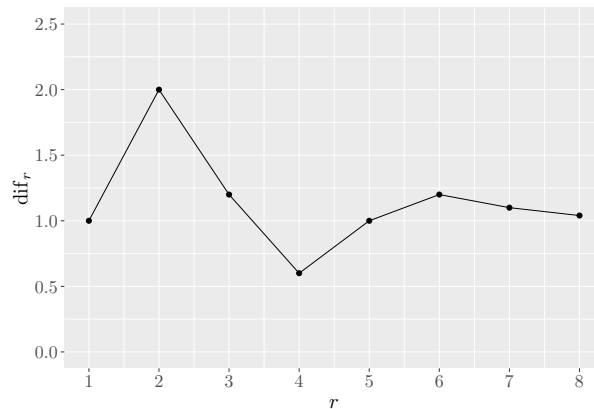
For this reason, **we need a different kind of a plot, called the overimpact plot, or O-plot in short.** The idea behind this new diagnostic tool is that we should compare each $\tilde{V}_r$ against the same, fixed matrix V_{FIXED} , calculating the difference as

$$\text{dif}_r(V_{\text{FIXED}}) = \|\tilde{V}_{\alpha_r} - V_{\text{FIXED}}\|_F. \quad (5.35)$$

In the above exemplary scenario, if we chose – for example –

$$V_{\text{FIXED}} = \begin{pmatrix} 2.5 & 2.5 \\ 2.5 & 2.5 \end{pmatrix}, \quad (5.36)$$

we would obtain a following O-plot presented in Fig. 5.11.

Figure 5.11: O-plot for the exemplary scenario presented in Table 5.7.

Source: own work.

We can see that even in such extreme, artificially designed scenario, the O-plot is not leading to a false conclusion about the supposedly decreasing nature of the empirical

impact of partial supervision as a function of the scaling factor α – as opposed to conclusions one could draw from the analysis of the U-plot – but it reflects the random-walk-like properties of the process. **Thus, we advocate to use the O-plots to estimate the empirical impact of partial supervision $\tilde{\pi}(\alpha)$.** The underimpact-plots are designed specifically to detect outliers, anomalies in the entire distribution $\{\delta_r\}$ to counteract underimpact. Having justified the choice of O-plots over U-plots to estimate empirical impact of partial supervision, we now need to account for the explanations provided in this thesis, and choose the appropriate V_{FIXED} argument of the corresponding dif_r function from Eq. (5.35) in the semi-supervised scenario.

Overimpact of partial supervision

The proposed value for the V_{FIXED} is associated with the full name of the O-plot introduced above, namely **the overimpact-plot**. The proposed definition of overimpact is analogous to the phenomenon of underimpact - while the latter considered small values of the scaling factor α (precisely - too small to counteract the initialization being untoward partial supervision), the former considers big values of α . Since in the additive combination technique there is no upper bound on the value of α , a question appears about the limit of the impact of partial supervision. **It shall be stressed that such a matter has not been precisely defined, or studied, in the literature so far, and the provided considerations constitute a novel proposition.**

We propose to relate the overimpact to the maximal possible impact. Let $\bar{X}$ be a class-average features matrix defined as

$$\bar{X} = \begin{pmatrix} \frac{1}{\sum_{j=1}^N f_{j1}} \sum_{j=1}^N f_{j1} \mathbf{x}_j \\ \vdots \\ \frac{1}{\sum_{j=1}^N f_{jc}} \sum_{j=1}^N f_{jc} \mathbf{x}_j \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^N \left(\frac{f_{j1}}{\sum_{j'=1}^N f_{j'1}} \right) \mathbf{x}_j \\ \vdots \\ \sum_{j=1}^N \left(\frac{f_{jc}}{\sum_{j'=1}^N f_{j'c}} \right) \mathbf{x}_j \end{pmatrix}. \quad (5.37)$$

In $\bar{X}$, each supervised observation $\mathbf{x}_i$ contributes only to its supervised cluster $s(i)$. This follows from (3.29), as in the binary supervision case, the function $s(i)$ assigns such value that $f_{i,s(i)} = 1$. **In the semi-supervised case, $\bar{X}$ refers to the extreme situation where both (a) impact of unsupervised observations and (ii) impact of supervised observations to non-supervised clusters are excluded.** Therefore, the overimpact of partial supervision can be defined as follows.

Definition 9 (Overimpact of partial supervision (I)). *The overimpact of partial supervision occurs when*

$$\tilde{V}_\alpha^{(L)} \xrightarrow{\alpha \rightarrow \infty} \bar{X}. \quad (5.38)$$

In Definition 9, the proposed V_{FIXED} is associated with $\bar{X}$. Such a choice would not only allow to build easy to interpret O-plots, but would also enable the immediate investigation of the potential occurrence of the overimpact. The proposed definition of overimpact is a theoretical construct, and whether the overimpact can occur at all is a separate task requiring an adequate mathematical proof that is beyond the scope of this thesis. Nevertheless, in a concrete semi-supervised scenario, on a concrete dataset, one can empirically check if overimpact occurs by examining a corresponding O-plot.

In light of Definition 9, the empirical study of overimpact would require a calculation of $\bar{X}$, which is an average of the features matrix limited to the supervised observations. We can avoid this step by using the alternate definition of overimpact that is focused on the weights matrix $\Phi^{(L)}$ from the final, L th loop of the Alternating Optimization algorithm. Such an alternate characterization clearly shows the importance of treating memberships as weights, according to the proposed Interpretation 2. It builds on Eq. (3.10), referring to the generic representation of the formula for optimal prototypes in SSFC, that allows to represent $\hat{V}$ in a similar convention as $\bar{X}$ in Eq. (5.37)

$$\hat{V} = \begin{pmatrix} \frac{1}{\sum_{j=1}^N \phi_{j1}} \sum_{j=1}^N \phi_{j1} \mathbf{x}_j \\ \vdots \\ \frac{1}{\sum_{j=1}^N \phi_{jc}} \sum_{j=1}^N \phi_{jc} \mathbf{x}_j \end{pmatrix} \quad (5.39)$$

and clearly, by Eq. (5.39) and Eq. (5.37), the definition of overimpact presented in Eq. (5.38) can be represented also as

$$\Phi_\alpha^{(L)} \xrightarrow{\alpha \rightarrow \infty} F, \quad (5.40)$$

i.e., the overimpact occurs when the weights $\Phi_\alpha^{(L)}$ used to calculate the decision prototypes converge to the binary supervision matrix F .

The final step to the alternate definition of the overimpact, and the ultimate definition of the O-plot, is to consider a difference between the two matrices. Let us define the function ν_r – analogous to the function δ_r used in the underimpact-plot – in the creation of overimpact-plot in the semi-supervised fuzzy clustering as

$$\nu_r = \|\Phi_{\alpha_r}^{(L)} - F\|_F, \quad (5.41)$$

The choice of the symbol ν_r is to distinguish this function from function dif_r defined in Eq. (5.35), which did not assume any specific form of V_{FIXED} , whereas ν_r builds on the concept of overimpact of partial supervision analyzed using the weights matrix $\Phi_{\alpha}^{(L)}$ and supervision matrix F . The overimpact plot will be a display of pairs (α_r, ν_r) , and the alternate definition of overimpact, based on Eq. (5.40) and Eq. (5.41), is defined as follows.

Definition 10 (Overimpact of partial supervision (II)). *The overimpact of partial supervision occurs when*

$$\|\Phi_{\alpha}^{(L)} - F\|_F \xrightarrow{\alpha \rightarrow \infty} 0. \quad (5.42)$$

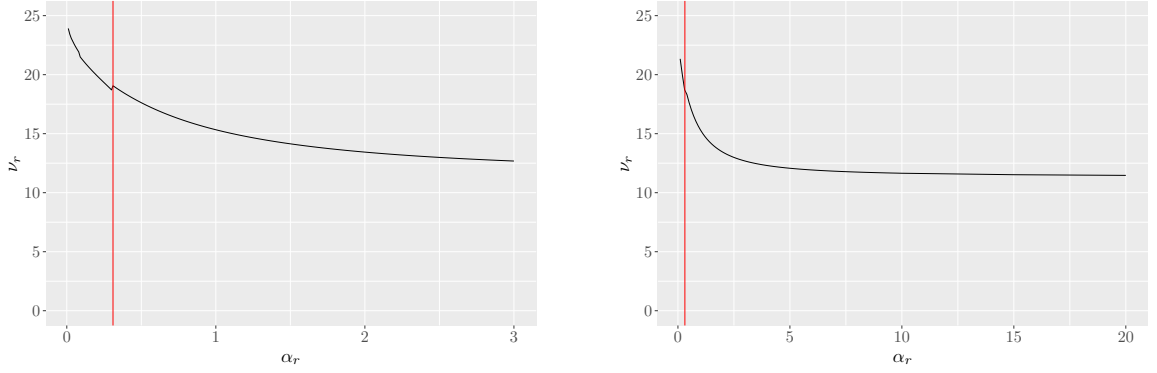
Overimpact plots for considered datasets and empirical impact of partial supervision

The overimpact plot serves two purposes: it enables to detect the occurrence of overimpact, as defined in Eq. (5.42), and estimates the empirical impact of partial supervision $\tilde{\pi}(\alpha)$, defined in Section 3.2.3, as it is free from the drawback of the underimpact-plot discussed above in this Section 5.2. Let us now analyze the O-plots for the **bipolar**, **iris**, and **chaos** datasets presented already throughout this Chapter.

Fig. 5.12 presents the O-plot for the outcomes of Semi-Supervised Fuzzy c-Means models fitted to the **bipolar** data. Fig. 5.12a presents the O-plot for the same grid $\mathcal{A}$ as in the case of underimpact analysis performed in Fig. 5.6 for the matching models, and Fig. 5.12b presents the O-plot for the extended grid $\mathcal{A} = (0.01, 0.02, \dots, 20)$. The red, vertical line in each of the figures denotes the threshold value $\alpha_{\text{bipolar}}^{\dagger} = 0.31$ defined in Eq. (5.33).

In Fig. 5.12a, we can see that the empirical impact of partial supervision $\tilde{\pi}(\alpha)$ is a non-linear, decreasing function of α - as α grows, the values of ν_r take smaller values. The impact seems not to flatten out at the value of $\alpha_r = 3$, therefore we analyze the results for the extended grid presented in Fig. 5.12b. This figure provides clear insight that the empirical impact of partial supervision does converge to a certain limit, but this limit is not equal to 0 - and hence, overimpact as defined in Definition 10 does not occur.

To confirm the above conclusion, and establish the limit of impact of partial supervision, we estimate a single Semi-Supervised Fuzzy c-Means model, using an arbitrarily big value of $\alpha = 1e10$. The resulting difference

Figure 5.12: Overimpact plot for the `bipolar` dataset.**(a)** Values of ν_r for the $\mathcal{A} = (0.01, 0.02, \dots, 2.99, 3)$. **(b)** Values of ν_r for the $\mathcal{A} = (0.01, 0.02, \dots, 20)$.

Source: own work. The red vertical line denotes the value $\alpha_{\text{bipolar}}^\dagger = 0.31$.

$$\|\Phi_{\alpha=1e10}^{(L)} - F\|_F = 11.32 \quad (5.43)$$

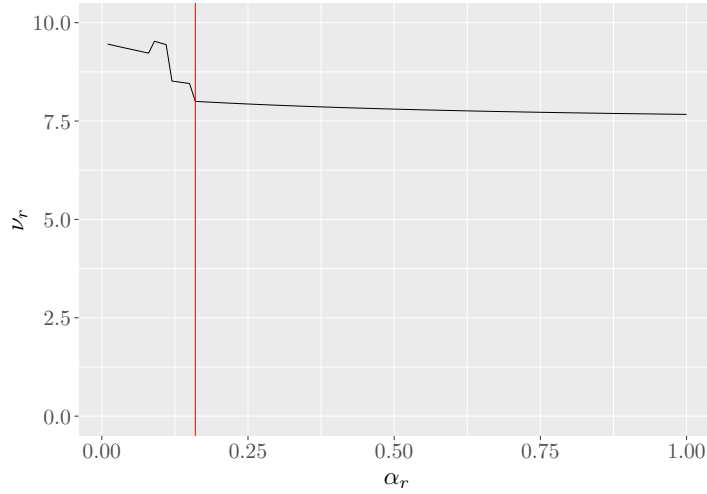
suggests that overimpact of partial supervision is impossible in this case. Even though the above results constitute a strong suggestion that overimpact may not be possible, the confirmation of this statement would require a formal proof that is beyond the scope of this thesis.

Fig. 5.13 presents the O-plot for the `iris` data. It is clearly visible how underimpact affected the models fitted with the values $\alpha < \alpha_{\text{iris}}^\dagger$, where $\alpha_{\text{iris}}^\dagger = 0.16$ according to Eq. (5.27). The black line estimating the empirical impact of partial supervision $\tilde{\pi}(\alpha)$ is chaotically moving up and down, reflecting the changing order of prototypes, and gets stable only for values $\alpha \geq \alpha_{\text{iris}}^\dagger$. After excluding the values of the scaling factor α affected by underimpact, one can conclude that the empirical impact of partial supervision in this case is a linear function, exhibiting minor changes in the impact as α grows towards the value of $\alpha_r = 1$.

Finally, Fig. 5.14 presents the underimpact plot for curated, artificially created `chaos` dataset. It presents a clear, non-linear trend; as opposed to `iris` data, the threshold value $\alpha_{\text{chaos}}^\dagger = 0.34$ does not change the functional character of $\tilde{\pi}(\alpha)$ in this case. Starting from $\alpha_{r=1} = 0.01$, the impact is steadily decreasing in a non-linear fashion, converging towards the value of $\nu_r = 5$.

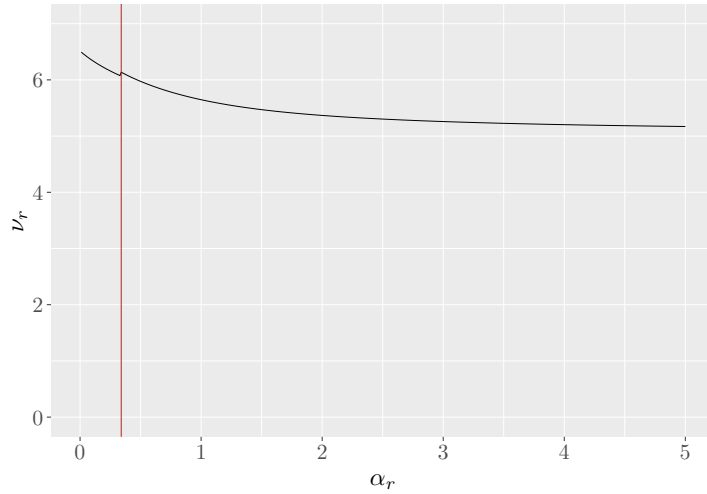
Summary. The overimpact analysis for the three datasets – `bipolar`, `iris`, and `chaos` – suggests that in each case, the overimpact as defined in Definition 10 cannot occur. As stated above, a firm confirmation of such suggestion requires a formal proof, but the results of the empirical analysis constitute a motivation to undertake such an effort, which

Figure 5.13: Overimpact plot for the `iris` dataset.



Source: own work. The red vertical line denotes the value $\alpha_{\text{iris}}^\dagger = 0.16$.

Figure 5.14: Overimpact plot for the `chaos` dataset.



Source: own work. The red vertical line denotes the value $\alpha_{\text{chaos}}^\dagger = 0.34$.

remains beyond the scope of this thesis.

Moreover, the three O-plots shown in Fig. 5.12, Fig. 5.13, and Fig. 5.14 present varying functional form of impact of partial supervision $\tilde{\pi}(\alpha)$ - from a strong, non-linear decrease for `bipolar` data in Fig. 5.12, through a mild, non-linear decrease for `chaos` data in Fig. 5.14, to a linear, very mild decrease for `iris` data in Fig. 5.13. This confirms that SSFC models try to adapt to the assumed partial supervision, regardless of its actual quality (refer to the case of the `chaos` dataset), and that this impact has a decreasing character, suggesting there exists a limit of it. Nevertheless, the functional form of the empirical impact, and the extent to which SSFC models can adapt to partial supervi-

sion, vary between the datasets, and hence, **the study of empirical impact of partial supervision should always consider the characteristic of a given experimental scenario.** There does not exist a single, universal functional form of the empirical impact of partial supervision.

5.3 Guidance on creating grids of scaling factor α

Based on the detailed analyses performed in Section 5.1 on the topic of underimpact of partial supervision, and in Section 5.2 on the topics of overimpact of partial supervision and empirical impact of partial supervision, we can formulate the guidance on how to create the grids of the scaling factor α .

As stated in Section 1.3 – when introducing the *grid* approach towards the interpretation of the scaling factor α in the literature – the task of creating any grid involves decisions on: (i) the grid’s boundaries: α_1 and α_q , (ii) the number of elements of the grid being equal to q , and (iii) the scale of the grid, i.e. the spacing between consecutive elements of the grid. This Section provides guidance with respect to each of these matters.

(iii) Firstly, we will start with the third aspect - the scale of the grid. Since the experiments performed in Section 5.2 showed that the functional form of impact of partial supervision $\tilde{\pi}(\alpha)$ depends on the dataset, and given experimental scenario, there does not exist a universal functional form that could be recommended in every case. Therefore, we opt for the safe choice to start with a linear grid such that

$$\alpha_r - \alpha_{r-1} = \Delta \quad \forall r = 2, 3, \dots, q, \quad (5.44)$$

where Δ is a constant value. Upon fitting the overimpact plot, one can examine if the actual empirical impact of partial supervision fits to the assumption of linear impact, or not. If, as in case of **bipolar** data and the associated O-plot presented in Fig. 5.12, $\tilde{\pi}(\alpha)$ is non-linear, and one needs for certain reasons a grid that is linear in ν_r - they can construct an alternate grid by modifying the spacing between consecutive elements that exhibits the required characteristics.

(ii) The second aspect of creating grids, the number of elements q , is subject only to the computational cost that the experimenters can afford. It is suggested that the number of elements is sufficient enough to be able to construct underimpact plots - the detection of anomalies requires an adequate number of regular values of δr in the distribution of $\{\delta_r\}$. Unfortunately, there cannot be any specific recommendations made here besides

sharing the number of elements q used in the experiments performed in this thesis. For example, the underimpact analysis for the `iris` dataset presented in Fig. 5.5 was based on a grid composed of $q = 100$ elements. Note that the consequences of choosing an insufficient value of q for the purposes of automated detection of underimpact in the task of estimating frequency of this phenomenon are discussed in Section 5.4.1.

(i) The most important aspect of the proposed guidance is the approach to selecting the grid's boundaries a_1 and a_q , as this aspect pertains directly to the consideration of the novel phenomena formulated and analyzed in this thesis in Sections 5.1 and 5.2.

With respect to the lower boundary, it is recommended that a_1 should be equal to $\alpha^\dagger$ obtained during the underimpact analysis, according to the Rule 1. Section 5.1 provides detailed examples of such analyses.

The selection of upper boundary α_q is a more difficult task. It is recommended to select a *big* value, e.g. $\alpha_q = 10$, and construct the overimpact plot, based on which further actions should be taken. If, as in case of the O-plot presented for `bipolar` dataset in Fig. 5.12a, one is not certain whether the curve of the estimated empirical impact of partial supervision is flattening out, a second O-plot should be constructed for a higher value of a_q – as in case of Fig. 5.12b in the referred analysis – to achieve the flattening out of the curve. When the overimpact plot exhibits the *flattening out of the curve* estimating the empirical impact of partial supervision, it is recommended to choose the smallest $\alpha_r \in \mathcal{A}$ beyond which the curve flattens out.

The use of the term *empirical* with regard to the title of this Chapter *Empirical analysis of impact of partial supervision* was also supposed to stress that creating α grids is to some extent an art, not a strict science, as there does not exist a formula for the optimal choice - it involves arbitrary decisions about the precise threshold exhibiting the *flattening out* of the curve (α_r, ν_r) . In particular, this involves selecting the upper boundary a_q of the grid; the most important aspect of this decision is not to lose any information, and thus to choose such a_q beyond which the increases in the value of the scaling factor α do not significantly alter the empirical impact of partial supervision $\tilde{\pi}(\alpha)$ (the *flattening out* of the empirical impact of partial supervision curve). For example, in case of the above-mentioned analysis of `bipolar` data and the associated O-plot from Fig. 5.12b, the choice could be to select $a_q = 8$.

5.4 Simulations - frequency of underimpact

In the introductory Section 1.4 it was stressed that the underimpact phenomenon is of stochastic nature. In general, Section 5.1 provided detailed analysis of underimpact;

in particular, Subsection 5.1.3 explained the stochastic character of underimpact - this phenomenon occurs when two conditions are satisfied: the random initialization $U^{(0)} = \Phi^{(0)}$ is untoward partial supervision, and the value of the scaling factor α is too low to counteract the untoward initialization $\Phi^{(0)}$.

The examples analyzed throughout entire Section 5.1 were carefully chosen to exhibit the occurrence of underimpact, but in general – due to its stochastic nature – underimpact does not have to occur. This Section 5.4 is devoted to studying the frequency of underimpact materializing in the selected cases. It provides a simulation protocol and tools to study the frequency of underimpact both in fully supervised scenario – when partial supervision is simulated by removing the labels for selected observations – and in a truly semi-supervised scenario.

In Section 5.2, we stated that *the study of empirical impact of partial supervision should always consider the characteristic of a given experimental scenario*, stressing that there does not exist a universal functional form of the empirical impact of partial supervision applicable to every semi-supervised fuzzy clustering problem. The same applies to the problem of estimating frequency of underimpact considered in this Section - any change in the data analyzed (a different dataset, or a different subset of variables from a given dataset) or in the partial supervision should be accounted for in the separate analysis. **Thus, the goal of this Section is to estimate the frequency of underimpact in the specific examples considered in this thesis, and to provide the toolkit for the practitioners to estimate frequency of underimpact in their cases.**

5.4.1 Automated detection of underimpact

The problem of detecting underimpact of partial supervision was discussed in Section 5.1.5, where we introduced the underimpact-plot, and the associated Rule 1 defining how to establish a threshold value $\alpha^\dagger$ such that underimpact does not occur for $\alpha \geq \alpha^\dagger$. The use of Rule 1 in Section 5.1.5 relied on visual inspection of the associated U-plot, whereas in the task of estimating frequency of underimpact we would like to automate this activity. Below we present an idea of an approach to this problem that takes inspiration from the field of Statistical Process Control (Montgomery, 2020), and specifically - control charts that aim at defining quality thresholds to monitor the stability of the processes.

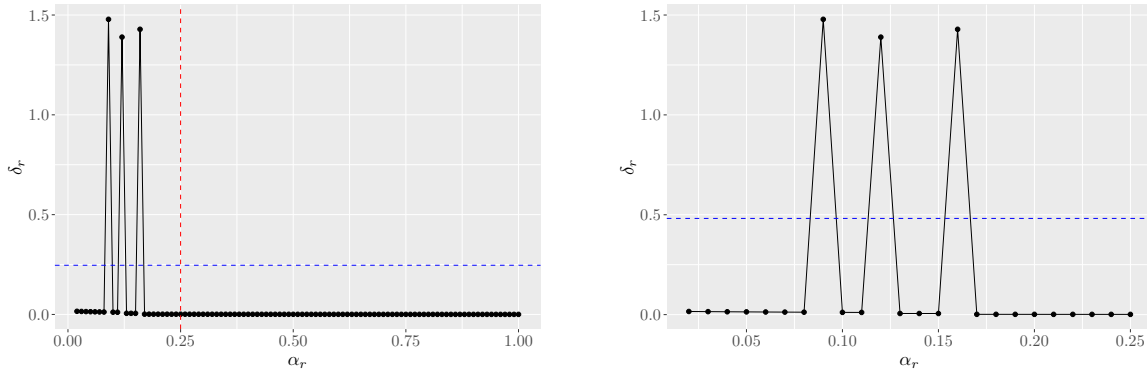
Recall that Rule 1 defines candidate values for the threshold value $\alpha^\dagger$ as such that they remain outliers with respect to the entire distribution $\{\delta_r\}$, i.e. as values of clearly higher magnitude than the typical value of $\delta_r \approx 0$. The simplest algorithm to automatically detect such candidate values – and then choose the greatest of them as $\alpha^\dagger$ – is to compare each value δ_r from the respective U-plot against the measure of dispersion of

this distribution; in this thesis, we choose to identify the candidate values by comparing if the respective values are greater or lower than the standard deviation $\text{sd}(\{\delta_r\})$. The associated Rule 2 presents as follows.

Rule 2 (Automated selection of the threshold value $\alpha^\dagger$). *To detect underimpact, select the threshold value $\alpha^\dagger$ to be the biggest value of the scaling factor α out of potential candidates, i.e. such values α_r that the corresponding δ_r is greater than the standard deviation of $\{\delta_r\}$.*

Let us illustrate this approach on the example of detecting underimpact of partial supervision in the analyses performed on `iris` dataset. Specifically, in Fig. 5.15 we replot Fig. 5.5 by adding horizontal lines denoting the standard deviations of the respective distributions $\{\delta_r\}$.

Figure 5.15: Underimpact plot for the matrix X_{train} , `iris` dataset, with standard deviation of δ_r marked.



(a) Values of δ_r for $\mathcal{A} = (0.02, 0.03, \dots, 0.99, 1)$. (b) Values of δ_r for $\mathcal{A}' = (\alpha_r \in \mathcal{A} : \alpha_r \leq 0.25)$.

Source: own work. This is a replot of Fig. 5.5. The red, dashed vertical line in Fig. 5.15a denotes the upper limit defining the range of (α_r, δ_r) pairs displayed in Fig. 5.15b. The blue, dashed horizontal lines denote the respective values of $\text{sd}(\{\delta_r\})$.

For each grid – $\mathcal{A} = (0.02, 0.03, \dots, 0.99, 1)$ presented in Fig. 5.15a and $\mathcal{A}' = (\alpha_r \in \mathcal{A} : \alpha_r \leq 0.25)$ presented in Fig. 5.15b – the application of Rule 2 leads to a correct identification of the three candidate values for the threshold value $\alpha^\dagger$.

Nevertheless, the above example shows how critical is a choice of the number of elements in the grid q in the task of estimating the frequency of underimpact. Standard deviation of values $\{\delta_r\}$ calculated for the entire grid $\mathcal{A}$ was equal to 0.25, whereas the associated standard deviation calculated for the subset $\mathcal{A}' \subset \mathcal{A}$ was equal to 0.48; the exclusion of a number of elements with $\delta_r \approx 0$ inflated the standard deviation. **In certain scenarios, such inflation of the value of standard deviation may lead to incorrect exclusion of candidate values for threshold $\alpha^\dagger$ that remain outliers in the spirit of Rule 1.** Therefore, for the purpose of estimating frequency of underimpact, we recommend creating dense grids of the scaling factor α values.

5.4.2 Analyses of datasets

Fully supervised iris data

The `iris` dataset used throughout this thesis to illustrate and analyze the impact of partial supervision in semi-supervised fuzzy clustering is a fully supervised dataset; moreover, in Section 5.1.5 it was established that considering two characteristics of iris – sepal length and sepal width – we can derive the expected order of prototypes presented using the $\text{Ord}(\tilde{V})$ defined in Eq. (5.26) as $(1, 2, 3)$; denoting by $\tilde{v}_{1,k}$ the first coordinate of k th decision cluster's prototype, we expect that

$$\tilde{v}_{1,1} < \tilde{v}_{1,2} < \tilde{v}_{1,3}. \quad (5.45)$$

The detection of underimpact in this case does not require automated detection as defined in Rule 2.

The simulation protocol is presented in Algorithm 4. It relies on modifying the following experimental factors:

- partial supervision will be simulated by keeping the labels only for a fraction of observations; the considered fractions will include 0.05, 0.10, 0.15, 0.20, in such a way that

$$\Omega_{0.05}^S \subset \Omega_{0.10}^S \subset \Omega_{0.15}^S \subset \Omega_{0.20}^S,$$
 where $\Omega_{\text{fraction}}^S$ denotes the set of elements to remain supervised in each fraction scenario,
- for each ratio, the selection of observations to remain supervised will be random sampled 10 times, sustaining the class balance of the selected observations,
- for each ratio, for each simulated supervision matrix F , 10 random initializations of $U^{(0)}$ matrix will be simulated.

Finally, for each combination of the fraction $\frac{M}{N}$, simulated supervision matrix F , and initialization matrix $U^{(0)}$, Semi-Supervised Fuzzy c-Means models will be fitted to the grid $\mathcal{A} = \{0.1, 0.2, \dots, 9.9, 10\}$.

The results of simulations are presented in Fig. 5.16; Fig. 5.16a presents the entirety of the results, and Fig. 5.16b presents the results for the subgrid $\mathcal{A}' = (\alpha \in \mathcal{A} : \alpha \leq 1)$.

First and foremost, in Fig. 5.16b we can see the clear impact of fraction $\frac{M}{N}$ on the frequency of underimpact of partial supervision occurring; the curves representing the fractions are clearly separated, sustaining the order in which **the greater the fraction of observations supervised, the less frequent the materialization of underimpact of partial supervision.**

Algorithm 4 Protocol for simulations on `iris` data, for a single ratio $\frac{M}{N}$

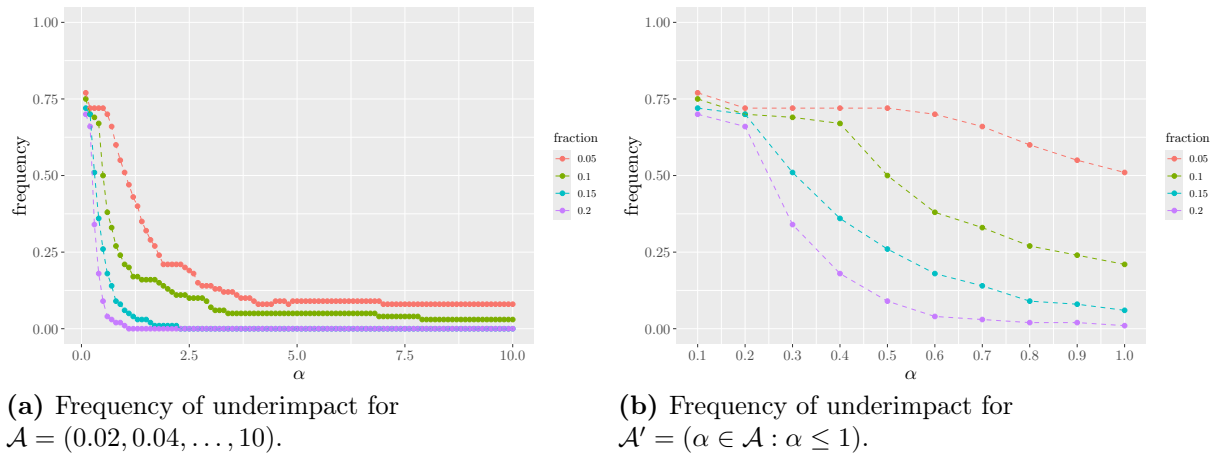
```

1: for  $\alpha \in \mathcal{A}$  do
2:    $\text{out}_\alpha \leftarrow 0$ 
3: end for
4: for  $j = 1, \dots, 10$  do
5:   randomly sample the supervised observations  $\Omega_j^S$ 
6:   for  $i = 1, \dots, 10$  do
7:     randomly sample an initialization matrix  $U_i^{(0)} \in M_{fc}$ 
8:     for  $\alpha \in \mathcal{A}$  do
9:       estimate  $M(\alpha, F(\Omega_j^S), U_i^{(0)})$  using:  $\Omega_j^S, U_i^{(0)}, \alpha$ 
10:      if not  $(\tilde{v}_{1,1} < \tilde{v}_{1,2} < \tilde{v}_{1,3})$  then
11:         $\text{out}_\alpha \leftarrow \text{out}_\alpha + 1$ 
12:      end if
13:    end for
14:  end for
15: end for
16: for  $\alpha \in \mathcal{A}$  do
17:    $\text{frac}_\alpha \leftarrow \text{out}_\alpha / 100$ 
18: end for

```

Secondly, in Fig. 5.16b we observe that for fractions 0.05 and 0.1 the underimpact seems to be persistent for all the values of the scaling factor α ; the flat, non-changing shape of the curve estimating the frequency of underimpact for values $\alpha > 5$ indicates that in this case, the underimpact is detected due to a specific realization of supervision matrix F . Recall the structure of the data presented in Fig. 5.1. The uncertainty in the data may lead to a particular realization of random sample of observation such that the expected order of prototypes defined in Eq. (5.45) is impossible to achieve.

Figure 5.16: Results of estimating frequency of underimpact for the matrix X_{train} , `iris` dataset



Source: own work. Results of simulation per simulation protocol described in Algorithm 4.

Finally, regarding the actual frequencies themselves, Fig. 5.16b presents the results for the grid $\mathcal{A}' = (\alpha \in \mathcal{A} : \alpha \leq 1)$ selected to aid reading and interpretation of the results. We can see that for the values of the scaling factor $\alpha \in \{0.1, 0.2\}$, the frequency of underimpact remains high for each fraction 0.05, 0.1, 0.15, 0.2, achieving the values of ≈ 0.75 . The frequency starts to decrease beyond $\alpha = 0.2$ for fractions 0.15, 0.2, converging to values close to 0 for $\alpha = 1$; it starts to decrease beyond $\alpha = 0.4$ for the fraction 0.1, converging to value close to 0.25 for $\alpha = 1$; finally, the frequency of underimpact starts to decrease beyond $\alpha = 0.6$ for the smallest fraction 0.05, converging to value close to 0.5 for $\alpha = 1$.

Regarding the interpretation of the values, **every occurrence of underimpact of partial supervision is undesired as it can compromise the inference drawn from a specific model**. Considering that in practical scenarios, the fraction $\frac{M}{N}$ may remain low (especially in situations when annotating data has a high associated cost, e.g. annotating medical images by professional healthcare providers), the results from Fig. 5.16b show how important in each case is to analyze the results of a given semi-supervised fuzzy clustering model to exclude from considerations such models that were impacted by the phenomenon of underimpact of partial supervision.

Semi-supervised bipolar data

The semi-supervised **bipolar** dataset is described in detail in Section 5.1.5. The estimation of frequency of underimpact for this dataset will differ from the approach towards fully supervised **iris** dataset - the only experimental factor that will be changing across each of the simulations will be the initialization matrix $U^{(0)}$, since partial supervision is given; hence the supervision matrix F and the fraction $\frac{M}{N} = \frac{1295}{261} = 0.2$ are fixed. The simulation protocol is described in Algorithm 5.

The results of simulations are presented in Fig. 5.17; Fig. 5.17a presents the entirety of the results, and Fig. 5.17b presents the results for the subgrid $\mathcal{A}' = (\alpha \in \mathcal{A} : \alpha \leq 1)$. The precise values of non-zero frequencies of underimpact of partial supervision are presented in Table 5.8.

Table 5.8: Non-zero frequencies of underimpact of partial supervision in **bipolar** data simulations.

α	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7
freq.	0.95	0.8	0.75	0.6	0.55	0.35	0.25	0.15	0.15	0.15	0.1	0.1	0.1

Source: own work. Freq. = frequency.

The interpretation of the results is similar to the one presented above for **iris** data simulations - every occurrence of underimpact is undesired. For **bipolar** data – under the

Algorithm 5 Protocol for simulations on **bipolar** data.

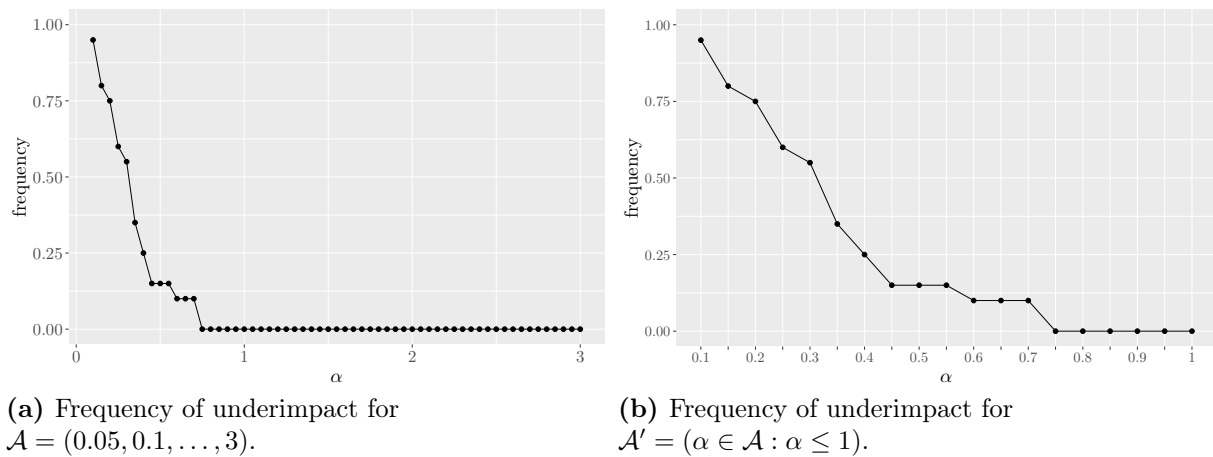
```

1: for  $\alpha \in \mathcal{A}$  do
2:    $\text{out}_\alpha \leftarrow 0$ 
3: end for
4: for  $i = 1, \dots, 20$  do
5:   randomly sample an initialization matrix  $U_i^{(0)} \in M_{fc}$ 
6:   for  $\alpha \in \mathcal{A}$  do
7:     estimate  $M(\alpha, F, U_i^{(0)})$  using:  $F, U_i^{(0)}, \alpha$ 
8:   end for
9:   create underimpact plot, denote pairs  $(\alpha_r, \delta_r)$ 
10:  apply Rule 2 to select  $\alpha^\dagger$ 
11:  for  $\alpha \in \mathcal{A}$  do
12:    if  $\alpha < \alpha^\dagger$  then
13:       $\text{out}_\alpha \leftarrow \text{out}_\alpha + 1$ 
14:    end if
15:  end for
16: end for
17: for  $\alpha \in \mathcal{A}$  do
18:    $\text{frac}_\alpha \leftarrow \text{out}_\alpha / 20$ 
19: end for

```

simulation protocol described in Algorithm 5 – the underimpact is occurring for $\alpha \leq 0.75$, suggesting that values $\alpha > 0.75$ may be sufficient enough to counteract any untoward initialization of underimpact; however, this remains a suggestion only due to the limited number of iterations run (20), which is associated with a high computational cost of running extensive simulations using complex datasets such as **bipolar**.

Figure 5.17: Results of estimating frequency of underimpact for the **bipolar** dataset



Source: own work. Results of simulation per simulation protocol described in Algorithm 5.

It is worth noting that while the frequency of underimpact is decreasing fast, its

frequency amounts to as much as 0.95 for $\alpha = 0.1$. Referring to Section 2.2.4, this value of the scaling factor α is being used by researchers and practitioners. In light of the above, the simulation results for **bipolar** data – although formally limited to the specific settings of this concrete simulation protocol from Algorithm 5 – lead to a conclusion that one can expect a frequent occurrence of underimpact for values of the scaling factor α as low as 0.1.

Chapter 6

Conclusions

The detailed literature review on the scaling factor α , presented in Section 2.2.4, led to the conclusion that, despite decades of advances in semi-supervised fuzzy clustering, no dedicated studies have focused solely on the critical mechanism of regulating the impact of partial supervision through this specific hyperparameter α . The brief original interpretation of the role of α – referred to in this thesis as the Canonical Interpretation 1 – proposed by Pedrycz and Waletzky (1997), who introduced the classical Semi-Supervised Fuzzy c-Means model, has remained prevalent and unchallenged. However, this interpretation focused only on the context of the objective function; the lack of analysis of the role of α in the fuzzy clustering model treated as a whole — including the objective function, the optimization problem, the solution algorithm, and the model outcomes — results in a reduced trust in the modeling results. Therefore, the key motivation and aim of this thesis was to revisit this core mechanism and extend the discussion on the impact of partial supervision in semi-supervised fuzzy clustering, and on the crucial role of the scaling factor α in regulating this impact, in order to achieve an explainable impact of partial supervision that conveys trust in the modeling results.

A consequence of treating the impact of partial supervision as a side topic in the literature was the lack of uniform terminology and notation. Therefore, the first step was to establish appropriate language to precisely formulate the problem. Supported by precise definitions of *interpretation* and *explanation* for Artificial Intelligence models (with fuzzy clustering models treated as AI), coined on the grounds of Explainable Artificial Intelligence, the propositions made in this thesis enabled a clear formulation of the problem addressed in this thesis, stated in Section 1.2 as the need to *explain the role of the scaling factor α in regulating the impact of partial supervision in the class of semi-supervised fuzzy clustering models adapting the additive combination technique (as defined in Eq. 1.4)*. The research gap analysis in Section 1.3 confirmed that this problem remains unsolved in the

literature, and thus, its solution is addressed through two co-primary objectives of this thesis, formulated in Section 1.5:

- Primary Objective 1: *to introduce explainability framework for impact of partial supervision in semi-supervised fuzzy clustering,*
- Primary Objective 2: *to explain the impact of partial supervision in two classical SSFC models: Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means.*

Formally, Primary Objective 1 has been achieved in Section 3.1, which provides the source for the definition of an explanation, and Section 3.2, which is composed of five subsections, each treating on a different aspect of the explainability framework. Together, the five subsections form a **coherent and exhaustive framework for explaining the impact of partial supervision in semi-supervised fuzzy clustering**, where a framework is understood as an arbitrary collection of formalisms, notation, definitions, and conceptualizations of certain phenomena devoted specifically to explaining the impact of partial supervision (as defined in Eq. 1.6) in any semi-supervised fuzzy clustering model that adapts the additive combination technique (as defined in Eq. 1.4).

Primary Objective 2 has been achieved in Section 3.3 and Section 3.4, respectively, where Explanation 1 explains the impact of α on decision memberships $\tilde{U}$ and Explanation 2 explains the impact of α on decision prototypes $\tilde{V}$ in Semi Supervised Fuzzy c-Means, and Explanation 4 explains the impact of α on decision memberships $\tilde{U}$ and Explanation 5 explains the impact of α on decision prototypes $\tilde{V}$ in Semi Supervised Possibilistic c-Means.

The theoretical advances proposed in this thesis led to novel practical considerations that constitute four secondary objectives of this thesis. In particular, the alternative representation of the fuzzy clustering objective function presented in Section 3.2.1 was of key importance. To date, publications in the field of semi-supervised fuzzy clustering have represented the objective function of a given model as a sum of components; this approach was termed the *additive combination* technique by Kmita et al. (2024), as described in Section 2.2.2 and represented schematically in Eq. (1.4), which is reproduced below for convenience:

$$J = Q(U, V; X, \Theta) + \alpha \cdot Q^S(U, V; X, F, \Theta). \quad (6.1)$$

In particular situations, there may be more than one supervised component Q^S , refer for example to J_{GAN} from Eq. (2.43).

In Section 3.2.1, a novel **generic fuzzy clustering objective function** G is proposed; it relies on noticing that each component element of the composite semi-supervised objective function J is a double sum over j , indexing the observations, and k , indexing the clusters; the idea is to move all calculations under a single double sum. The generic objective function defined in Eq. (3.7) is reproduced below for convenience:

$$G(U, V) = \sum_{j=1}^N \sum_{k=1}^c (\phi(u_{jk}) \cdot d_{jk}^2 + p(u_{jk})) , \quad (6.2)$$

where $\phi_{jk} = \phi(u_{jk})$ defined in Eq. (3.2) is a weighing function that transforms the memberships, and $p(u_{jk})$ is a penalty function. This novel representation allowed to represent the formula for the optimal prototypes in **any fuzzy clustering – unsupervised or semi-supervised – model** in Eq. (3.10), which is reproduced below for convenience:

$$\hat{\mathbf{v}}_k = \sum_{j=1}^N \frac{\phi_{jk}}{\sum_{s=1}^N \phi_{sk}} \mathbf{x}_j. \quad (6.3)$$

In this view, the models differ in the exact form of the weights ϕ_{jk} that they define.

The importance of this novel, alternative representation is that it refocuses attention on the impact of α on the prototypes, treating memberships as weights adjusted by partial supervision that – together with the data X – define the values of the decision prototypes $\tilde{V}$. The interpretation of memberships as weights in unsupervised Fuzzy c-Means was already proposed by J. C. Bezdek (1981), but it was specific to this model and not further investigated; the Interpretation 2 proposed in this thesis is model-agnostic and applicable to any unsupervised or semi-supervised fuzzy clustering model. Interpretation 2 also discusses the semantical issues with interpreting memberships in the semi-supervised case in light of fuzzy set theory.

The shift in focus to the relation between the decision prototypes $\tilde{V}$ and the scaling factor α was central to the achievements presented in Chapter 4 and Chapter 5. Identifying a common notion for the calculation of prototypes in any fuzzy clustering model was a motivation to formulate Secondary Objective 1 – *to create novel modular open-source software implementation of selected semi-supervised fuzzy clustering models* – which was achieved in Chapter 4. The establishment of a novel implementation was necessary for the author to conduct the analyses presented in Chapter 5, as none of the state-of-the-art implementations enabled the extraction of the weights matrix $\Phi = [\phi_{jk}]$ from the model, which is crucial for defining and analyzing the phenomena of underimpact and overimpact of partial supervision studied in Chapter 5. Moreover, it was found during the literature review presented in Section 4.2 that the SSFC field generally lacks reproducible software

implementations of core models that follow the efficiency of development rule defined in Section 4.1. Therefore, Secondary Objective 1 was achieved by providing a formal mathematical description of the vectorized operations for calculating optimal memberships $\hat{U}$ and optimal prototypes $\hat{V}$ for four classical models: FCMeans, PCMeans, SSFCMeans, and SSPCMeans, and sharing the code in form of a dedicated R package `fussclust` (*Fuzzy Unsupervised and Semi-Supervised Clustering*), available at GitHub platform at <https://github.com/kkmita/fussclust>.

Finally, the remaining three secondary objectives: Secondary Objective 2 *to identify and provide empirical analysis of underimpact of partial supervision*, Secondary Objective 3 *to identify and provide empirical analysis of overimpact of partial supervision*, and Secondary Objective 4 *to provide guidance on how to create grids of the scaling factor α values* have been achieved in Chapter 5. Both the underimpact of partial supervision, the stochastic untoward phenomenon that occurs when using too small values of the scaling factor α analyzed in Sections 5.1 and 5.4, and the overimpact of partial supervision, the potential untoward phenomenon that occurs when using too big values of α analyzed in Section 5.2, are defined using the notion of the generic objective function G and the weights ϕ_{jk} . Section 5.1.3 defines that underimpact occurs when the random initialization weights $\Phi^{(0)}$ of the Alternating Optimization (AO) algorithm (defined in Algorithm 3) are untoward partial supervision, and the value of α is too low to counteract this situation; Definition 10 states that overimpact occurs when the final weights $\Phi^{(L)}$, estimated at the last L th step of the AO algorithm, converge towards the supervision matrix F as α approaches infinity.

As demonstrated in this thesis, the underimpact of partial supervision, in particular, can compromise the validity of inferences drawn from affected models. A detailed, in-depth analysis of how its occurrence can inflate the classification error, as measured by metrics such as accuracy, is presented in Section 5.1.1. As a remedy to these identified negative phenomena, two two-dimensional diagnostic plots are proposed: the underimpact-plot (U-plot) to detect the occurrence of underimpact, and the overimpact-plot (O-plot) to detect the potential occurrence of overimpact. Although the construction and interpretation of the plots, together with the practical analysis of the selected datasets, were conducted specifically using Semi-Supervised Fuzzy c-Means model, **the two diagnostic plots are model-agnostic and can serve to detect underimpact or overimpact in any semi-supervised fuzzy clustering model**, since they only require the availability of the decision prototypes $\tilde{V}_\alpha$, which are the outcome of any SSFC model. All presented results can be reproduced using the R `Markdown` notebooks shared in the author’s personal GitHub repository (<https://github.com/kkmita/phd-thesis>); the rendered

results are also available at the same location.

In practice, when using the estimated semi-supervised fuzzy clustering model as a classifier, the user’s interest frequently lies in optimizing an external quality criterion; specifically, they aim to select a value of α that maximizes this criterion. The quality of this procedure depends on the quality of the finite set of candidate values – called a grid – of the scaling factor α used in the procedure. The final Secondary Objective 4 of this thesis, associated with this topic, has been achieved in Section 5.3 *Guidance on creating grids of scaling factor α* . Concrete recommendations regarding each aspect of creating the grid $\mathcal{A} = (a_r)_{r=1,\dots,q}$ – (i) the grid’s boundaries: α_1 and α_q , (ii) the number of elements of the grid being equal to q , and (iii) the scale of the grid, i.e. the spacing between consecutive elements of the grid – were made using the introduced phenomena of underimpact and overimpact, and the diagnostic tools proposed to detect them. In particular, the U-plot and O-plot are used to select the grid boundaries, and the overimpact-plot is additionally used to determine the scale of the grid. As demonstrated in Section 5.2, the overimpact-plot can serve not only to identify the occurrence of overimpact but also to estimate the empirical impact of partial supervision, $\tilde{\pi}(\alpha)$ – a function quantifying the effect of the scaling factor α with all other experimental factors fixed.

The formulation of the problem addressed in this thesis was initially presented in the author’s article (Kmita et al., 2024). The scope of this thesis is greatly extended compared to the referenced paper, but that paper laid the foundations for further research. Apart from (Kmita et al., 2024), the author published two additional articles related to the contents of this thesis, which further confirm the need for the explainability framework developed in this dissertation. Kmita et al. (2022) represents an initial study aimed at estimating label uncertainty using Semi-Supervised Fuzzy c-Means in the context of remote monitoring of patients diagnosed with bipolar disorder¹. In that article, a linear form of the empirical impact of partial supervision, $\tilde{\pi}(\alpha)$, was assumed. Furthermore, Kmita et al. (2023) investigated in more detail the mechanics of using semi-supervised fuzzy clustering models as classifiers. The considerations presented in this paper led to a refocus on the expected order of cluster prototypes, which is relevant to the use of SSFC in classification tasks.

Regarding future work, the author is hopeful that the presented results will be disseminated in the field of semi-supervised fuzzy clustering, as the novel Interpretation 2, which refocuses attention on the decision prototypes $\tilde{V}$ together with the identified phenomena of underimpact and overimpact are applicable to any semi-supervised fuzzy clustering model. One research stream is to apply the derived explainability framework and provide

¹The article received the Best Paper Award at the 2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE) conference held in Padua, Italy.

formal explanations of the impact of α on decision memberships and decision prototypes in the SSFC models other than the classical Semi-Supervised Fuzzy c-Means and Semi-Supervised Possibilistic c-Means. The second research stream includes an empirical analysis of underimpact and overimpact using U-plots and O-plots in SSFC models other than Semi-Supervised Fuzzy c-Means, to empirically assess their sensitivity to these negative phenomena. In particular, this involves an empirical study of Semi-Supervised Possibilistic c-Means model, which is prone to the negative phenomenon of coincident prototypes – first ensuring that this phenomenon has not occurred – or an empirical study of derivative models, such as Semi-Supervised Repulsive Possibilistic c-Means (Antoine et al., 2018), which directly aim to address the problem of coincident prototypes.

Bibliography

1. Aggarwal, C. C., & Reddy, C. K. (Eds.). (2018, September 3). *Data Clustering: Algorithms and Applications* (1st ed.). Chapman and Hall/CRC. <https://doi.org/10.1201/9781315373515>
2. Albert, B., Antoine, V., & Koko, J. (2025). ECM+: An improved evidential c-means with adaptive distance. *Fuzzy Sets and Systems*, 498, 109168. <https://doi.org/10.1016/j.fss.2024.109168>
3. Ali, S., Abuhmed, T., El-Sappagh, S., Muhammad, K., Alonso-Moral, J. M., Confalonieri, R., Guidotti, R., Del Ser, J., Díaz-Rodríguez, N., & Herrera, F. (2023). Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence. *Information Fusion*, 99, 101805. <https://doi.org/10.1016/j.inffus.2023.101805>
4. Antoine, V., Guerrero, J. A., Boone, T., & Romero, G. (2018). Possibilistic clustering with seeds. *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 1–7. <https://doi.org/10.1109/FUZZ-IEEE.2018.8491655>
5. Antoine, V., Guerrero, J. A., & Romero, G. (2022). Possibilistic fuzzy c-means with partial supervision. *Fuzzy Sets and Systems*, 449, 162–186. <https://doi.org/10.1016/j.fss.2022.08.003>
6. Antoine, V., & Labroche, N. (2018). Semi-supervised Fuzzy c-Means Variants: A Study on Noisy Label Supervision. In J. Medina, M. Ojeda-Aciego, J. L. Verdegay, D. A. Pelta, I. P. Cabrera, B. Bouchon-Meunier, & R. R. Yager (Eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Foundations* (pp. 51–62, Vol. 854). Springer International Publishing. https://doi.org/10.1007/978-3-319-91476-3_5
7. Arthur, D., & Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1027–1035.

8. Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barredo, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58, 82–115. <https://doi.org/10.1016/j.inffus.2019.12.012>
9. Becker, G., Barr, C., Gentleman, R., & Lawrence, M. (2017). Enhancing Reproducibility and Collaboration via Management of *R* Package Cohorts. *Journal of Statistical Software*, 82(1). <https://doi.org/10.18637/jss.v082.i01>
10. Bezdek, J. C. (1974). Numerical taxonomy with fuzzy sets. *Journal of Mathematical Biology*, 1(1), 57–71. <https://doi.org/10.1007/BF02339490>
11. Bezdek†, J. C. (1973). Cluster Validity with Fuzzy Sets. *Journal of Cybernetics*, 3(3), 58–73. <https://doi.org/10.1080/01969727308546047>
12. Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Springer US. <https://doi.org/10.1007/978-1-4757-0450-1>
13. Bezdek, J. C. (2025). A Short Tutorial: Some Things You Might Not Know About *k*-Means Clustering. *IEEE Systems, Man, and Cybernetics Magazine*, 11(1), 23–33. <https://doi.org/10.1109/MSMC.2024.3443535>
14. Bezdek, J. C., Coray, C., Gunderson, R., & Watson, J. (1981). Detection and Characterization of Cluster Substructure I. Linear Structure: Fuzzy *c*-Lines. *SIAM Journal on Applied Mathematics*, 40(2), 339–357. <https://doi.org/10.1137/0140029>
15. Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3), 191–203. [https://doi.org/10.1016/0098-3004\(84\)90020-7](https://doi.org/10.1016/0098-3004(84)90020-7)
16. Blaser, N. (2020). *Rdist: Calculate pairwise distances*. <https://CRAN.R-project.org/package=rdist>
17. Bouchachia, A., & Pedrycz, W. (2006). Enhancement of fuzzy clustering by mechanisms of partial supervision. *Fuzzy Sets and Systems*, 157(13), 1733–1759. <https://doi.org/10.1016/j.fss.2006.02.015>
18. Bouchachia, A., & Pedrycz, W. (2003). A Semi-supervised Clustering Algorithm for Data Exploration. In T. Bilgiç, B. De Baets, & O. Kaynak (Eds.), *Fuzzy Sets and Systems — IFSA 2003* (pp. 328–337). Springer Berlin Heidelberg.

19. Broniatowski, D. A. (2021, April 12). *Psychological Foundations of Explainability and Interpretability in Artificial Intelligence*. National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.IR.8367>
20. Burns, P. (2011). *The R inferno*. Patrick Burns Consulting. https://www.burns-stat.com/pages/Tutor/R_inferno.pdf
21. Casalino, G., Castellano, G., Galetta, F., & Kaczmarek-Majer, K. (2020). Dynamic Incremental Semi-supervised Fuzzy Clustering for Bipolar Disorder Episode Prediction. In A. Appice, G. Tsoumakas, Y. Manolopoulos, & S. Matwin (Eds.), *Discovery Science* (pp. 79–93, Vol. 12323). Springer International Publishing. https://doi.org/10.1007/978-3-030-61527-7_6
22. Casalino, G., Castellano, G., & Mencar, C. (2019). Data Stream Classification by Dynamic Incremental Semi-Supervised Fuzzy Clustering. *International Journal on Artificial Intelligence Tools*, 28(08), 1960009. <https://doi.org/10.1142/S0218213019600091>
23. Casalino, G., Dominiak, M., Galetta, F., & Kaczmarek-Majer, K. (2020). Incremental Semi-Supervised Fuzzy C-Means for Bipolar Disorder Episode Prediction. *2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, 1–8. <https://doi.org/10.1109/EAIS48028.2020.9122748>
24. Cebeci, Z. (2019). Comparison of Internal Validity Indices for Fuzzy Clustering. *Journal of Agricultural Informatics*, 10(2). <https://doi.org/10.17700/jai.2019.10.2.537>
25. Chapelle, O., Schölkopf, B., & Zien, A. (Eds.). (2006). *Semi-supervised learning*. MIT Press.
26. Chunfang Li, Lianzhong Liu, & Wenli Jiang. (2008). Objective function of semi-supervised Fuzzy C-Means clustering algorithm. *2008 6th IEEE International Conference on Industrial Informatics*, 737–742. <https://doi.org/10.1109/INDIN.2008.4618199>
27. Collberg, C., & Proebsting, T. A. (2016). Repeatability in computer systems research. *Communications of the ACM*, 59(3), 62–69. <https://doi.org/10.1145/2812803>
28. Confalonieri, R., Coba, L., Wagner, B., & Besold, T. R. (2021). A historical perspective of explainable Artificial Intelligence. *WIREs Data Mining and Knowledge Discovery*, 11(1), e1391. <https://doi.org/10.1002/widm.1391>

29. De Melo, F. M., & De Carvalho, F. D. A. (2013). Semi-supervised fuzzy c-medoids clustering algorithm with multiple prototype representation. *2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 1–7. <https://doi.org/10.1109/FUZZ-IEEE.2013.6622374>
30. Denceux, T., & Kanjanatarakul, O. (2016). Evidential Clustering: A Review. In V.-N. Huynh, M. Inuiguchi, B. Le, B. N. Le, & T. Denoeux (Eds.), *Integrated Uncertainty in Knowledge Modelling and Decision Making* (pp. 24–35, Vol. 9978). Springer International Publishing. https://doi.org/10.1007/978-3-319-49046-5_3
31. Dey, L., & Mukhopadhyay, A. (2011). Microarray gene expression data clustering using PSO based K-means algorithm. *Proceedings of the International Conference on Advanced Computing, Communication and Networks*, 1, 587–591.
32. Drineas, P., Frieze, A., Kannan, R., Vempala, S., & Vinay, V. (2004). Clustering Large Graphs via the Singular Value Decomposition. *Machine Learning*, 56(1-3), 9–33. <https://doi.org/10.1023/B:MACH.0000033113.59016.96>
33. Dubois, D., & Prade, H. (2001). Possibility Theory, Probability Theory and Multiple-Valued Logics: A Clarification. *Annals of Mathematics and Artificial Intelligence*, 32(1-4), 35–66. <https://doi.org/10.1023/A:1016740830286>
34. Dubois, D., & Prade, H. (1988). *Possibility theory: An approach to computerized processing of uncertainty*. Plenum Press.
35. Dubois, D., & Prade, H. (1997). The three semantics of fuzzy sets. *Fuzzy Sets and Systems*, 90(2), 141–150. [https://doi.org/10.1016/S0165-0114\(97\)00080-8](https://doi.org/10.1016/S0165-0114(97)00080-8)
36. Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3(3), 32–57. <https://doi.org/10.1080/01969727308546046>
37. Eddelbuettel, D., & François, R. (2011). **Rcpp** : Seamless R and C++ Integration. *Journal of Statistical Software*, 40(8). <https://doi.org/10.18637/jss.v040.i08>
38. Ferraro, M., Brigida, Giordani, P., & Serafini, A. (2019). Fclust: An R Package for Fuzzy Clustering. *The R Journal*, 11(1), 198. <https://doi.org/10.32614/RJ-2019-017>
39. Ferraro, M. B. (2024). Fuzzy k -Means: History and applications. *Econometrics and Statistics*, 30, 110–123. <https://doi.org/10.1016/j.ecosta.2021.11.008>

40. Ferraro, M. B., & Giordani, P. (2020). Soft clustering. *WIREs Computational Statistics*, 12(1). <https://doi.org/10.1002/wics.1480>
41. Frigui, H., & Hwang, C. (2008). Fuzzy Clustering and Aggregation of Relational Data With Instance-Level Constraints. *IEEE Transactions on Fuzzy Systems*, 16(6), 1565–1581. <https://doi.org/10.1109/TFUZZ.2008.2005692>
42. Gagolewski, M. (2022). A framework for benchmarking clustering algorithms. *SoftwareX*, 20, 101270. <https://doi.org/10.1016/j.softx.2022.101270>
43. Gan, H., Fan, Y., Luo, Z., Huang, R., & Yang, Z. (2019). Confidence-weighted safe semi-supervised clustering. *Engineering Applications of Artificial Intelligence*, 81, 107–116. <https://doi.org/10.1016/j.engappai.2019.02.007>
44. Gan, H., Fan, Y., Luo, Z., & Zhang, Q. (2018). Local homogeneous consistent safe semi-supervised clustering. *Expert Systems with Applications*, 97, 384–393. <https://doi.org/10.1016/j.eswa.2017.12.046>
45. Gan, H., Gan, W., Yang, Z., & Zhou, R. (2024). Discrimination-aware safe semi-supervised clustering. *Information Sciences*, 676, 120798. <https://doi.org/10.1016/j.ins.2024.120798>
46. Gan, H., Yang, Z., Shi, M., Ye, Z., & Zhou, R. (2025). Improved safe semi-supervised clustering based on capped ℓ_{21} norm. *Fuzzy Sets and Systems*, 505, 109276. <https://doi.org/10.1016/j.fss.2025.109276>
47. Gan, H., Yang, Z., & Zhou, R. (2023). Adaptive safety-aware semi-supervised clustering. *Expert Systems with Applications*, 212, 118751. <https://doi.org/10.1016/j.eswa.2022.118751>
48. Gao, J., Tan, P.-N., & Cheng, H. (2006). Semi-Supervised Clustering with Partial Background Information. *Proceedings of the 2006 SIAM International Conference on Data Mining*, 489–493. <https://doi.org/10.1137/1.9781611972764.46>
49. Gao, Y., Wang, X., Pan, J., Zhu, Q., Cao, C., & Shao, G. (2024). Possibilistic c-Means Clustering Algorithms Based on Adaptive Nearest Neighbors. *2024 36th Chinese Control and Decision Conference (CCDC)*, 4011–4016. <https://doi.org/10.1109/CCDC62350.2024.10588186>

50. Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Leisch, F., Li, C., Maechler, M., Rossini, A. J., ... Zhang, J. (2004). Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5(10), R80. <https://doi.org/10.1186/gb-2004-5-10-r80>
51. Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining Explanations: An Overview of Interpretability of Machine Learning. *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, 80–89. <https://doi.org/10.1109/DSAA.2018.00018>
52. Golzari Oskouei, A., Samadi, N., & Tanha, J. (2024). Feature-weight and cluster-weight learning in fuzzy c-means method for semi-supervised clustering. *Applied Soft Computing*, 161, 111712. <https://doi.org/10.1016/j.asoc.2024.111712>
53. Gray, R. M. (2011). *Entropy and Information Theory*. Springer US. <https://doi.org/10.1007/978-1-4419-7970-4>
54. Grira, N., Crucianu, M., & Boujemaa, N. (2008). Active semi-supervised fuzzy clustering. *Pattern Recognition*, 41(5), 1834–1844. <https://doi.org/10.1016/j.patcog.2007.10.004>
55. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2019). A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys*, 51(5), 1–42. <https://doi.org/10.1145/3236009>
56. Guo, L., Gan, H., Xia, S., Xu, X., & Zhou, T. (2021). Joint exploring of risky labeled and unlabeled samples for safe semi-supervised clustering. *Expert Systems with Applications*, 176, 114796. <https://doi.org/10.1016/j.eswa.2021.114796>
57. Gustafson, D., & Kessel, W. (1978). Fuzzy clustering with a fuzzy covariance matrix. *1978 IEEE Conference on Decision and Control Including the 17th Symposium on Adaptive Processes*, 761–766. <https://doi.org/10.1109/CDC.1978.268028>
58. Hall, L., Ozyurt, I., & Bezdek, J. (1999). Clustering with a genetically optimized approach. *IEEE Transactions on Evolutionary Computation*, 3(2), 103–112. <https://doi.org/10.1109/4235.771164>
59. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer New York. <https://doi.org/10.1007/978-0-387-84858-7>

60. Hathaway, R. J., & Bezdek, J. C. (1988). Recent convergence results for the fuzzy c-means clustering algorithms. *Journal of Classification*, 5(2), 237–247. <https://doi.org/10.1007/BF01897166>
61. Hathaway, R. J., Bezdek, J. C., & Tucker, W. T. (1987). An improved convergence theory for the fuzzy c-means clustering algorithms. In J. C. Bezdek (Ed.), *Analysis of fuzzy information* (pp. 123–132, Vol. 3). CRC Press.
62. Hathaway, R., & Bezdek, J. (1995). Optimization of clustering criteria by reformulation. *IEEE Transactions on Fuzzy Systems*, 3(2), 241–245. <https://doi.org/10.1109/91.388178>
63. He, Y. (2025). Semi-supervised equilibrium K-means for imbalanced data clustering. *Knowledge-Based Systems*, 326, 113990. <https://doi.org/10.1016/j.knosys.2025.113990>
64. Hester, Jim & Vaughan, Davis. (2023). *Bench: High Precision Timing of R Expressions* (R package version 1.1.3). <https://CRAN.R-project.org/package=bench>
65. Hofmann, T., Schölkopf, B., & Smola, A. J. (2008). Kernel methods in machine learning. *The Annals of Statistics*, 36(3). <https://doi.org/10.1214/0090536070000000677>
66. Hsin-Chien Huang, Yung-Yu Chuang, & Chu-Song Chen. (2012). Multiple Kernel Fuzzy Clustering. *IEEE Transactions on Fuzzy Systems*, 20(1), 120–134. <https://doi.org/10.1109/TFUZZ.2011.2170175>
67. Huan, P. T., Thong, P. H., Tuan, T. M., Hop, D. T., Thai, V. D., Minh, N. H., Giang, N. L., & Son, L. H. (2022). TS3FCM: Trusted safe semi-supervised fuzzy clustering method for data partition with high confidence. *Multimedia Tools and Applications*, 81(9), 12567–12598. <https://doi.org/10.1007/s11042-022-12133-6>
68. IEEE SA. (2025). *Standard for Requirements and Classification of Explainable Artificial Intelligence Systems* (Standard). IEEE Standards Association. <https://standards.ieee.org/ieee/2976/10522/>
69. Ioannidis, J. P. A. (2005). Why most published research findings are false. *PLOS Medicine*, 2(8). <https://doi.org/10.1371/journal.pmed.0020124>
70. ISO/IEC 22989:2022. (2025). *Artificial intelligence concepts and terminology* (Standard). International Organization for Standardization. <https://www.iso.org/standard/74296.html>

71. ISO/IEC TS 6254:2025. (2025). *Objectives and approaches for explainability and interpretability of ML models and AI systems* (Standard). International Organization for Standardization. <https://www.iso.org/standard/82148.html>
72. Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8), 651–666. <https://doi.org/10.1016/j.patrec.2009.09.011>
73. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer New York. <https://doi.org/10.1007/978-1-4614-7138-7>
74. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning: With Applications in R*. Springer US. <https://doi.org/10.1007/978-1-0716-1418-1>
75. James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). *An Introduction to Statistical Learning: With Applications in Python*. Springer International Publishing. <https://doi.org/10.1007/978-3-031-38747-0>
76. Janz, N. (2016). Bringing the gold standard into the classroom: Replication in university teaching. *International Studies Perspectives*, 17(4), 392–407. <https://doi.org/10.1111/insp.12104>
77. Jayaram, B., & Klawonn, F. (2013). Can Fuzzy Clustering Avoid Local Minima and Undesired Partitions? In C. Moewes & A. Nürnberger (Eds.), *Computational Intelligence in Intelligent Data Analysis* (pp. 31–44, Vol. 445). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-32378-2_3
78. Jebari, K., Elmoujahid, A., & Ettouhami, A. (2019). Automatic Genetic Fuzzy c-Means. *Journal of Intelligent Systems*, 29(1), 529–539. <https://doi.org/10.1515/jisys-2018-0063>
79. Jiang, G. C., & Sun, Z. J. (2014). Improvement on a Fuzzy C-Means Algorithm Based on Genetic Algorithm. *Applied Mechanics and Materials*, 614, 385–388. <https://doi.org/10.4028/www.scientific.net/AMM.614.385>
80. Kaczmarek-Majer, K., Casalino, G., Castellano, G., Hryniewicz, O., & Dominiak, M. (2022). Explaining smartphone-based acoustic data in bipolar disorder: Semi-supervised fuzzy clustering and relative linguistic summaries. *Information Sciences*, 588, 174–195. <https://doi.org/10.1016/j.ins.2021.12.049>

81. Kannan, S., Ramathilagam, S., & Chung, P. (2012). Effective fuzzy c-means clustering algorithms for data clustering problems. *Expert Systems with Applications*, 39(7), 6292–6300. <https://doi.org/10.1016/j.eswa.2011.11.063>
82. Khezri, S., Aghazadeh, N., Hashemzadeh, M., & Oskoue, A. G. (2025). FW-S3KIFCM: Feature Weighted Safe-Semi-Supervised Kernel-Based Intuitionistic Fuzzy C-Means Clustering Method. *Fuzzy Information and Engineering*, 17(2), 227–260. <https://doi.org/10.26599/FIE.2025.9270061>
83. Klawonn, F., & Hoffmann, G. (2023). Using Fuzzy Cluster Analysis to Find Interesting Clusters. In L. A. García-Escudero, A. Gordaliza, A. Mayo, M. A. Lubiano Gomez, M. A. Gil, P. Grzegorzewski, & O. Hryniewicz (Eds.), *Building Bridges between Soft and Statistical Methodologies for Data Science* (pp. 231–239, Vol. 1433). Springer International Publishing. https://doi.org/10.1007/978-3-031-15509-3_31
84. Klawonn, F., & Höppner, F. (2003). What Is Fuzzy about Fuzzy Clustering? Understanding and Improving the Concept of the Fuzzifier. In M. R. Berthold, H.-J. Lenz, E. Bradley, R. Kruse, & C. Borgelt (Eds.), *Advances in Intelligent Data Analysis V* (pp. 254–264). Springer Berlin Heidelberg.
85. Klawonn, F., Kruse, R., & Winkler, R. (2015). Fuzzy clustering: More than just fuzzification. *Fuzzy Sets and Systems*, 281, 272–279. <https://doi.org/10.1016/j.fss.2015.06.024>
86. Kmita, K. (2024). *Source Code for Explainable Impact of Partial Supervision on Memberships in Semi-Supervised Fuzzy Clustering* (Version 2.1). <https://doi.org/10.24433/CO.3328061.V2>
87. Kmita, K., Casalino, G., Castellano, G., Hryniewicz, O., & Kaczmarek-Majer, K. (2022). Confidence path regularization for handling label uncertainty in semi-supervised learning: Use case in bipolar disorder monitoring. *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 1–8. <https://doi.org/10.1109/FUZZ-IEEE55066.2022.9882759>
88. Kmita, K., Kaczmarek-Majer, K., & Hryniewicz, O. (2023). Classification Error in Semi-Supervised Fuzzy C-Means. In S. Massanet, S. Montes, D. Ruiz-Aguilera, & M. González-Hidalgo (Eds.), *Fuzzy Logic and Technology, and Aggregation Operators* (pp. 725–736, Vol. 14069). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-39965-7_60

89. Kmita, K., Kaczmarek-Majer, K., & Hryniewicz, O. (2024). Explainable Impact of Partial Supervision in Semi-Supervised Fuzzy Clustering. *IEEE Transactions on Fuzzy Systems*, 1–10. <https://doi.org/10.1109/TFUZZ.2024.3370768>
90. Krishnapuram, R., Frigui, H., & Nasraoui, O. (1995). Fuzzy and possibilistic shell clustering algorithms and their application to boundary detection and surface approximation. I. *IEEE Transactions on Fuzzy Systems*, 3(1), 29–43. <https://doi.org/10.1109/91.366564>
91. Krishnapuram, R., & Keller, J. (1993). A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2), 98–110. <https://doi.org/10.1109/91.227387>
92. Lai, D. T. C., & Garibaldi, J. M. (2011). A comparison of distance-based semi-supervised fuzzy c-means clustering algorithms. *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, 1580–1586. <https://doi.org/10.1109/FUZZY.2011.6007562>
93. Liangqun, L., Hongbing, J., & Xinbo, G. (2006). Maximum entropy fuzzy clustering with application to real-time target tracking. *Signal Processing*, 86(11), 3432–3447. <https://doi.org/10.1016/j.sigpro.2006.03.007>
94. Lingras, P., & West, C. (2004). Interval Set Clustering of Web Users with Rough K-Means. *Journal of Intelligent Information Systems*, 23(1), 5–16. <https://doi.org/10.1023/B:JIIS.0000029668.88665.1a>
95. Liu, L., & Wu, X.-J. (2013). Semi-Supervised Possibilistic Fuzzy c-Means Clustering Algorithm on Maximized Central Distance. *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*. <https://doi.org/10.2991/iccsee.2013.342>
96. Liu, Y., Jiang, Y., Hou, T., & Liu, F. (2021). A new robust fuzzy clustering validity index for imbalanced data sets. *Information Sciences*, 547, 579–591. <https://doi.org/10.1016/j.ins.2020.08.041>
97. Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2), 129–137. <https://doi.org/10.1109/TIT.1982.1056489>
98. Masson, M.-H., & Dencœux, T. (2008). ECM: An evidential version of the fuzzy c-means algorithm. *Pattern Recognition*, 41(4), 1384–1397. <https://doi.org/10.1016/j.patcog.2007.08.014>

99. McLachlan, G., & Peel, D. (2000, September 18). *Finite Mixture Models* (1st ed.). Wiley. <https://doi.org/10.1002/0471721182>
100. Mei, J.-P. (2019). Semisupervised Fuzzy Clustering With Partition Information of Subsets. *IEEE Transactions on Fuzzy Systems*, 27(9), 1726–1737. <https://doi.org/10.1109/TFUZZ.2018.2889010>
101. Mei, J.-P., Lv, H., Cao, J., & Gong, W. (2019). Pairwise Constrained Fuzzy Clustering: Relation, Comparison and Parallelization. *International Journal of Fuzzy Systems*, 21(6), 1938–1949. <https://doi.org/10.1007/s40815-019-00683-1>
102. Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2024). *E1071: Misc functions of the department of statistics, probability theory group (formerly: E1071), TU wien*. <https://CRAN.R-project.org/package=e1071>
103. Moghaddam, A. N., Aghazadeh, N., Hashemzadeh, M., & Oskoue, A. G. (2025). A robust possibilistic semi-supervised fuzzy clustering algorithm with neighborhood-aware feature weighting. *International Journal of Machine Learning and Cybernetics*. <https://doi.org/10.1007/s13042-025-02731-9>
104. Molnar, C. (2025). *Interpretable machine learning: A guide for making black box models explainable* (3rd ed.). Christoph Molnar.
105. Montgomery, D. C. (2020). *Introduction to statistical quality control* (Eighth edition, EMEA edition). Wiley.
106. Nayak, J., Naik, B., & Behera, H. S. (2015). Fuzzy C-Means (FCM) Clustering Algorithm: A Decade Review from 2000 to 2014. In L. C. Jain, H. S. Behera, J. K. Mandal, & D. P. Mohapatra (Eds.), *Computational Intelligence in Data Mining - Volume 2* (pp. 133–149, Vol. 32). Springer India. https://doi.org/10.1007/978-81-322-2208-8_14
107. Open Science Collaboration. (2015). Estimating the reproducibility of psychological science. *Science*, 349(6251), aac4716. <https://doi.org/10.1126/science.aac4716>
108. Oskoue, A. G., Samadi, N., & Tanha, J. (2024, March). Feature-weight and cluster-weight learning in fuzzy c-means method for semi-supervised clustering. <https://doi.org/10.24433/CO.1422782.v1>
109. Ouchicha, C., Ammor, O., & Meknassi, M. (2020). A New Validity Index in Overlapping Clusters for Medical Images. *Automatic Control and Computer Sciences*, 54(3), 238–248. <https://doi.org/10.3103/S0146411620030050>

110. Pacheco, J., & Valencia, O. (2003). Design of hybrids for the minimum sum-of-squares clustering problem. *Computational Statistics & Data Analysis*, 43(2), 235–248. [https://doi.org/10.1016/S0167-9473\(02\)00224-4](https://doi.org/10.1016/S0167-9473(02)00224-4)
111. Pal, N., & Bezdek, J. (1995). On cluster validity for the fuzzy c-means model. *IEEE Transactions on Fuzzy Systems*, 3(3), 370–379. <https://doi.org/10.1109/91.413225>
112. Pal, N., Pal, K., & Bezdek, J. (1997). A mixed c-means clustering model. *Proceedings of 6th International Fuzzy Systems Conference*, 1, 11–21. <https://doi.org/10.1109/FUZZY.1997.616338>
113. Pal, N., Pal, K., Keller, J., & Bezdek, J. (2005). A possibilistic fuzzy c-means clustering algorithm. *IEEE Transactions on Fuzzy Systems*, 13(4), 517–530. <https://doi.org/10.1109/TFUZZ.2004.840099>
114. Pawlak, Z. (1982). Rough sets. *International Journal of Computer & Information Sciences*, 11(5), 341–356. <https://doi.org/10.1007/BF01001956>
115. Pawlak, Z. (1991). *Rough Sets*. Springer Netherlands. <https://doi.org/10.1007/978-94-011-3534-4>
116. Pedrycz, W., & Waletzky, J. (1997). Fuzzy clustering with partial supervision. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 27(5), 787–795. <https://doi.org/10.1109/3477.623232>
117. Peng, R. D. (2011). Reproducible Research in Computational Science. *Science*, 334(6060), 1226–1227. <https://doi.org/10.1126/science.1213847>
118. Peters, G. (2006). Some refinements of rough -means clustering. *Pattern Recognition*, 39(8), 1481–1491. <https://doi.org/10.1016/j.patcog.2006.02.002>
119. Phillips, P. J., Hahn, C. A., Fontana, P. C., Yates, A. N., Greene, K., Broniatowski, D. A., & Przybocki, M. A. (2021, September 29). *Four principles of explainable artificial intelligence* (NIST IR 8312). National Institute of Standards and Technology (U.S.) Gaithersburg, MD. <https://doi.org/10.6028/NIST.IR.8312>
120. R. A. Fisher. (1936). *Iris*. UCI Machine Learning Repository. <https://doi.org/10.24432/C56C76>
121. R Core Team. (2025). *R: A language and environment for statistical computing*. Vienna, Austria, R Foundation for Statistical Computing. <https://www.R-project.org/>

122. Ren, Y., Hu, X., Shi, K., Yu, G., Yao, D., & Xu, Z. (2018). Semi-supervised DenPeak Clustering with Pairwise Constraints. In X. Geng & B.-H. Kang (Eds.), *PRICAI 2018: Trends in Artificial Intelligence* (pp. 837–850, Vol. 11012). Springer International Publishing. https://doi.org/10.1007/978-3-319-97304-3_64
123. Roscher, R., Bohn, B., Duarte, M. F., & Garcke, J. (2020). Explainable Machine Learning for Scientific Insights and Discoveries. *IEEE Access*, 8, 42200–42216. <https://doi.org/10.1109/ACCESS.2020.2976199>
124. Rougier, N. P., Hinsén, K., Alexandre, F., Arildsen, T., Barba, L. A., Benureau, F. C., Brown, C. T., de Buyl, P., Caglayan, O., Davison, A. P., Delsuc, M.-A., Detorakis, G., Diem, A. K., Drix, D., Enel, P., Girard, B., Guest, O., Hall, M. G., Henriques, R. N., ... Zito, T. (2017). Sustainable computational science: The ReScience initiative. *PeerJ Computer Science*, 3, e142. <https://doi.org/10.7717/peerj-cs.142>
125. Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20, 53–65. [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7)
126. Rudin, C., & Radin, J. (2019). Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson From An Explainable AI Competition. *Harvard Data Science Review*, 1(2). <https://doi.org/10.1162/99608f92.5a8a3a3d>
127. Rui-Ping Li & Mukaidono, M. (1995). A maximum-entropy approach to fuzzy clustering. *Proceedings of 1995 IEEE International Conference on Fuzzy Systems. The International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium*, 4, 2227–2232. <https://doi.org/10.1109/FUZZY.1995.409989>
128. Ruspini, E. H. (1991). On the semantics of fuzzy logic. *International Journal of Approximate Reasoning*, 5(1), 45–88. [https://doi.org/https://doi.org/10.1016/0888-613X\(91\)90006-8](https://doi.org/https://doi.org/10.1016/0888-613X(91)90006-8)
129. Ruspini, E. H. (1990). Possibility as similarity; the semantics of fuzzy logic. *Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence*, 271–280.
130. Ruspini, E. H., Bezdek, J. C., & Keller, J. M. (2019). Fuzzy Clustering: A Historical Perspective. *IEEE Computational Intelligence Magazine*, 14(1), 45–55. <https://doi.org/10.1109/MCI.2018.2881643>

131. Saeed, W., & Omlin, C. (2023). Explainable AI (XAI): A systematic meta-survey of current challenges and future opportunities. *Knowledge-Based Systems*, 263, 110273. <https://doi.org/10.1016/j.knosys.2023.110273>
132. Sahbi, H., & Boujemaa, N. (2005). Validity of Fuzzy Clustering Using Entropy Regularization. *The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ '05.*, 177–182. <https://doi.org/10.1109/FUZZY.2005.1452389>
133. Salcedo-Sanz, S., Del Ser, J., & Geem, Z. W. (2014). An Island Grouping Genetic Algorithm for Fuzzy Partitioning Problems. *The Scientific World Journal*, 2014, 1–15. <https://doi.org/10.1155/2014/916371>
134. Salehi, F., Keyvanpour, M. R., & Sharifi, A. (2021). SMKFC-ER: Semi-supervised multiple kernel fuzzy clustering based on entropy and relative entropy. *Information Sciences*, 547, 667–688. <https://doi.org/10.1016/j.ins.2020.08.094>
135. Samadi, N., Tanha, J., & Jalili, M. (2025). A Weighted Semi-supervised Possibilistic Fuzzy c-Means algorithm for data stream classification and emerging class detection. *Knowledge-Based Systems*, 309, 112831. <https://doi.org/10.1016/j.knosys.2024.112831>
136. Santos, G. M., Julia, R. M. S., & Do Nascimento, M. Z. (2024). K-GBS3FCM - KNN Graph-Based Safe Semi-Supervised Fuzzy C-Means. *2024 IEEE International Conference on Big Data (BigData)*, 3498–3507. <https://doi.org/10.1109/BigData62323.2024.10825467>
137. Seeger, M. (2006). A taxonomy for semi-supervised learning methods. In O. Chapelle, B. Schölkopf, & A. Zien (Eds.), *Semi-supervised learning* (pp. 15–31). MIT Press.
138. Silva, L., Moura, R., Canuto, A. M. P., Santiago, R. H. N., & Bedregal, B. (2015). An Interval-Based Framework for Fuzzy Clustering Applications. *IEEE Transactions on Fuzzy Systems*, 23(6), 2174–2187. <https://doi.org/10.1109/TFUZZ.2015.2407901>
139. Simões, E. C., & T. De Carvalho, F. D. A. (2023). Gaussian kernel fuzzy c-means with width parameter computation and regularization. *Pattern Recognition*, 143, 109749. <https://doi.org/10.1016/j.patcog.2023.109749>
140. Stutz, C., & Runkler, T. (2002). Classification and prediction of road traffic using application-specific fuzzy clustering. *IEEE Transactions on Fuzzy Systems*, 10(3), 297–308. <https://doi.org/10.1109/TFUZZ.2002.1006433>

141. Tang, Y., Pan, Z., Hu, X., Pedrycz, W., & Chen, R. (2023). Knowledge-Induced Multiple Kernel Fuzzy Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12), 14838–14855. <https://doi.org/10.1109/TPAMI.2023.3298629>
142. Timm, H., & Kruse, R. (2002). A modification to improve possibilistic fuzzy cluster analysis. *2002 IEEE World Congress on Computational Intelligence. 2002 IEEE International Conference on Fuzzy Systems. FUZZ-IEEE'02. Proceedings (Cat. No.02CH37291)*, 2, 1460–1465. <https://doi.org/10.1109/FUZZ.2002.1006721>
143. Timm, H., Borgelt, C., Döring, C., & Kruse, R. (2001). Fuzzy cluster analysis with cluster repulsion. *Euro. Symp. Intelligent Technologies (EUNITE), Tenerife, Spain*.
144. Timm, H., Borgelt, C., Döring, C., & Kruse, R. (2004). An extension to possibilistic fuzzy cluster analysis. *Fuzzy Sets and Systems*, 147(1), 3–16. <https://doi.org/10.1016/j.fss.2003.11.009>
145. Tran, D., & Wagner, M. (2000). Fuzzy entropy clustering. *Ninth IEEE International Conference on Fuzzy Systems. FUZZ- IEEE 2000 (Cat. No.00CH37063)*, 1, 152–157. <https://doi.org/10.1109/FUZZY.2000.838650>
146. Van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, 109(2), 373–440. <https://doi.org/10.1007/s10994-019-05855-6>
147. Wang, H.-Y., Wang, J.-S., & Wang, G. (2022). A survey of fuzzy clustering validity evaluation methods. *Information Sciences*, 618, 270–297. <https://doi.org/10.1016/j.ins.2022.11.010>
148. Wang, Z., Wang, S.-S., Bai, L., Wang, W.-S., & Shao, Y.-H. (2022). Semisupervised Fuzzy Clustering With Fuzzy Pairwise Constraints. *IEEE Transactions on Fuzzy Systems*, 30(9), 3797–3811. <https://doi.org/10.1109/TFUZZ.2021.3129848>
149. Wickham, H. (2019, May 24). *Advanced R* (2nd ed.). Chapman and Hall/CRC. <https://doi.org/10.1201/9781351201315>
150. Wittgenstein, L. (1981). *Tractatus logico-philosophicus*. Routledge, Taylor & Francis Group.
151. Wu, C., & Hou, J. (2025). New semi-supervised fuzzy C-means clustering with asymmetric deviation constraints and fast algorithm. *Expert Systems with Applications*, 129648. <https://doi.org/10.1016/j.eswa.2025.129648>

152. Xie, X., & Beni, G. (1991). A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8), 841–847. <https://doi.org/10.1109/34.85677>
153. Xu, J., Feng, G., Zhao, T., Sun, X., & Zhu, M. (2019). Remote sensing image classification based on semi-supervised adaptive interval type-2 fuzzy c-means algorithm. *Computers & Geosciences*, 131, 132–143. <https://doi.org/10.1016/j.cageo.2019.06.005>
154. Xu, S., Hao, Z., Zhu, Y., Wang, Z., Xiao, Y., & Liu, B. (2024). Semi-supervised fuzzy clustering algorithm based on prior membership degree matrix with expert preference. *Expert Systems with Applications*, 238, 121812. <https://doi.org/10.1016/j.eswa.2023.121812>
155. Yang, D.-R., Lan, L.-S., & Pao, W.-C. (2006). A New Fuzzy Entropy Clustering Method with Controllable Membership Characteristics. *2006 49th IEEE International Midwest Symposium on Circuits and Systems*, 187–191. <https://doi.org/10.1109/MWSCAS.2006.382028>
156. Yasunori, E., Yukihiro, H., Makito, Y., & Sadaaki, M. (2009). On semi-supervised fuzzy c-means clustering. *2009 IEEE International Conference on Fuzzy Systems*, 1119–1124. <https://doi.org/10.1109/FUZZY.2009.5277177>
157. Yin, X., Shu, T., & Huang, Q. (2012). Semi-supervised fuzzy clustering with metric learning and entropy regularization. *Knowledge-Based Systems*, 35, 304–311. <https://doi.org/10.1016/j.knosys.2012.05.016>
158. Yu, H., Xu, X., Li, H., Wu, Y., & Lei, B. (2024). Semi-supervised possibilistic c-means clustering algorithm based on feature weights for imbalanced data. *Knowledge-Based Systems*, 286, 111388. <https://doi.org/10.1016/j.knosys.2024.111388>
159. Yu, S., Wei, Y.-M., Fan, J., Zhang, X., & Wang, K. (2012). Exploring the regional characteristics of inter-provincial CO2 emissions in China: An improved fuzzy clustering analysis based on particle swarm optimization. *Applied Energy*, 92, 552–562. <https://doi.org/10.1016/j.apenergy.2011.11.068>
160. Zadeh, L. (1965). Fuzzy sets. *Information and Control*, 8(3), 338–353. [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X)
161. Zadeh, L. (1999). Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100, 9–34. [https://doi.org/10.1016/S0165-0114\(99\)80004-9](https://doi.org/10.1016/S0165-0114(99)80004-9)

162. Zarinbal, M., Fazel Zarandi, M., & Turksen, I. (2014). Relative entropy fuzzy c-means clustering. *Information Sciences*, 260, 74–97. <https://doi.org/10.1016/j.ins.2013.11.004>
163. Zhang, B., Huang, L., Wang, J., Zhang, L., Wu, Y., Jiang, Y., & Xia, K. (2024). Semi-supervised fuzzy C means based on membership integration mechanism and its application in brain infarction lesion segmentation in DWI images. *Journal of Intelligent & Fuzzy Systems*, 46(1), 2713–2726. <https://doi.org/10.3233/JIFS-234148>
164. Zhang, B., Jiang, Y., & Xia, K. (2024). Interclass Balance Factor-Based Membership Fusion Semi-Supervised Fuzzy Clustering Algorithm for Lesion Segmentation in Cerebral Infarction Images. *IEEE Access*, 12, 107077–107088. <https://doi.org/10.1109/ACCESS.2024.3438122>
165. Zhang, D., Ji, M., Yang, J., Zhang, Y., & Xie, F. (2014). A novel cluster validity index for fuzzy clustering based on bipartite modularity. *Fuzzy Sets and Systems*, 253, 122–137. <https://doi.org/10.1016/j.fss.2013.12.013>
166. Zhang, H., & Lu, J. (2009). Semi-supervised fuzzy clustering: A kernel-based approach. *Knowledge-Based Systems*, 22(6), 477–481. <https://doi.org/10.1016/j.knosys.2009.06.009>
167. Zhang, J., & Shen, L. (2014). An Improved Fuzzy c -Means Clustering Algorithm Based on Shadowed Sets and PSO. *Computational Intelligence and Neuroscience*, 2014, 1–10. <https://doi.org/10.1155/2014/368628>
168. Zhong-dong Wu, Wei-xin Xie, & Jian-ping Yu. (2003). Fuzzy C-means clustering algorithm based on kernel method. *Proceedings Fifth International Conference on Computational Intelligence and Multimedia Applications. ICCIMA 2003*, 49–54. <https://doi.org/10.1109/ICCIMA.2003.1238099>
169. Zhu, L. F., Wang, J. S., & Wang, H. Y. (2019). A Novel Clustering Validity Function of FCM Clustering Algorithm. *IEEE Access*, 7, 152289–152315. <https://doi.org/10.1109/ACCESS.2019.2946599>
170. Zimmermann, H.-J. (2001). *Fuzzy Set Theory—and Its Applications*. Springer Netherlands. <https://doi.org/10.1007/978-94-010-0646-0>

Internet Sources

1. <https://codeocean.com/product/compute-capsules> (accessed on 31 January 2026)
2. <https://cran.r-project.org/web/packages/ppclust/index.html>
(accessed on 31 January 2026)
3. <https://doi.org/10.1016/j.cageo.2019.06.005> (accessed on 31 January 2026)
4. <https://github.com/> (accessed on 31 January 2026)
5. <https://github.com/cran/e1071/blob/master/src/cmeans.c> (accessed on 31 January 2026)
6. https://github.com/cran/fclust/blob/master/src/FKM_rcpp.cpp
(accessed on 31 January 2026)
7. <https://github.com/cran/ppclust/blob/master/R/fcm.R> (accessed on 31 January 2026)
8. <https://github.com/kkmita/fussclust> (accessed on 31 January 2026)
9. <https://github.com/kkmita/phd-thesis/tree/main> (accessed on 31 January 2026)
10. <https://ieee-dataport.org/> (accessed on 31 January 2026)
11. <https://www.doi.org/> (accessed on 31 January 2026)
12. <https://www.efspi.org/> (accessed on 31 January 2026)
13. <https://www.openstatsware.org/slides/r-govys-jan2023.html#/motivation>
(accessed on 31 January 2026)
14. <https://www.psiweb.org/about-us> (accessed on 31 January 2026)

List of Figures

1.1	The entire context of the fuzzy clustering model.	8
1.2	The contexts of balancing and grid approaches.	10
1.3	Dataset iris (R. A. Fisher, 1936).	12
2.1	An example of an unsupervised dataset with atypical observations.	32
3.1	Schematic approach to two ways of interpreting AI system.	57
3.2	Impact of partial supervision, its explanation and the context of semi-supervised fuzzy clustering model.	65
3.3	Weighing functions for weights of supervised observations in non-supervised and supervised cluster in Semi-Supervised Fuzzy c-Means model.	78
3.4	Optimal weighing functions for weights of supervised observations in non-supervised and supervised cluster in Semi-Supervised Fuzzy c-Means model.	79
4.1	Reproducibility spectrum.	88
4.2	The number of reviewed publications per year, per reproducibility category.	92
4.3	The results of efficiency comparison of Fuzzy c-Means implementation.	113
5.1	Dataset iris (R. A. Fisher, 1936).	119
5.2	Test dataset X_{test} and the decision prototypes.	120
5.3	Boxplots of weights $\phi_{jk}^{(1)}$ from the first loop of each of the models.	128
5.4	Boxplots of weights $\phi_{jk}^{(L)}$ from the last loop of each of the models.	130
5.5	Underimpact plot for the matrix X_{train} , iris dataset.	133
5.6	Underimpact plot for the bipolar dataset.	137
5.7	Underimpact plot for the bipolar dataset capped at $\max(\delta_r) = 0.025$	139
5.8	Simulated chaos dataset.	140
5.9	Underimpact plot for the chaos dataset.	141
5.10	U-plot for the exemplary scenario presented in Table 5.7.	143
5.11	O-plot for the exemplary scenario presented in Table 5.7.	143
5.12	Overimpact plot for the bipolar dataset.	147

5.13	Overimpact plot for the iris dataset.	148
5.14	Overimpact plot for the chaos dataset.	148
5.15	Underimpact plot for the matrix X_{train} , iris dataset, with standard deviation of δ_r marked.	152
5.16	Results of estimating frequency of underimpact for the matrix X_{train} , iris dataset	154
5.17	Results of estimating frequency of underimpact for the bipolar dataset . .	156

List of Tables

1.1	Values of the considered external quality criteria.	13
2.1	Summary of literature review on the topic of the scaling factor α	49
3.1	Selected definitions from the ISO/IEC standard.	56
3.2	Summary of introduced nomenclature.	63
4.1	Literature review on the matter of reproducibility in semi-supervised fuzzy clustering.	93
4.2	The results of efficiency comparison of Fuzzy c-Means implementation. . .	114
4.3	The results of efficiency comparison of Possibilistic c-Means implementation.	115
5.1	Average values of sepal length for different species of iris.	120
5.2	Values of initialization matrix $\Phi^{(0)}$ for the selected observations.	124
5.3	Values of initialization matrix $\Phi^{(0)}$ for the selected observations - alternate convention.	125
5.4	The analysis of the order of first coordinates of the prototypes in $\tilde{V}_{\alpha_r}$ for iris dataset.	134
5.5	Total number of unsupervised and supervised observations, by label, in bipolar dataset.	136
5.6	The analysis of the order of first coordinates of the prototypes in $\tilde{V}_{\alpha_r}$ for bipolar dataset.	138
5.7	Exemplary scenario showing that underimpact plot is not suited to analyze empirical impact of partial supervision.	142
5.8	Non-zero frequencies of underimpact of partial supervision in bipolar data simulations.	155

List of Pseudocodes

1	Alternating Optimization Algorithm for k -means (Jain, 2010)	23
2	Alternating Optimization Algorithm for Fuzzy c-Means (J. C. Bezdek, 1981)	29
3	Alternating Optimization Algorithm for Fuzzy Clustering	62
4	Protocol for simulations on <code>iris</code> data, for a single ratio $\frac{M}{N}$	154
5	Protocol for simulations on <code>bipolar</code> data.	156

List of Author's Publications

1. K. Kmita, K. Kaczmarek-Majer and O. Hryniewicz, "Explainable Impact of Partial Supervision in Semi-Supervised Fuzzy Clustering," in *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 5, pp. 3189-3198, May 2024, doi: 10.1109/TFUZZ.2024.3370768
2. Kmita, K., Kaczmarek-Majer, K., Hryniewicz, O. (2023). Classification Error in Semi-Supervised Fuzzy C-Means. In: Massanet, S., Montes, S., Ruiz-Aguilera, D., González-Hidalgo, M. (eds) *Fuzzy Logic and Technology, and Aggregation Operators. EUSFLAT AGOP 2023* 2023. Lecture Notes in Computer Science, vol 14069. Springer, Cham. https://doi.org/10.1007/978-3-031-39965-7_60
3. K. Kmita, G. Casalino, G. Castellano, O. Hryniewicz and K. Kaczmarek-Majer, "Confidence path regularization for handling label uncertainty in semi-supervised learning: use case in bipolar disorder monitoring," *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Padua, Italy, 2022, pp. 1-8, doi: 10.1109/FUZZ-IEEE55066.2022.9882759.
4. Kmita, K., Kaczmarek-Majer, K., Hryniewicz, O. (2023). Learning Control Limits for Monitoring of Multiple Processes with Neural Network. In: García-Escudero, L.A., et al. *Building Bridges between Soft and Statistical Methodologies for Data Science . SMPS 2022. Advances in Intelligent Systems and Computing*, vol 1433. Springer, Cham. https://doi.org/10.1007/978-3-031-15509-3_32