Semi-Supervised Fuzzy C-Means for Regression

Gabriella Casalino[a], Giovanna Castellano[b] and Corrado Mencar[c]
Dept. of Computer Science, University of Bari Aldo Moro, Italy

Keywords: Fuzzy Clustering, Semi-Supervised Clustering, Regression, Discretization.

Abstract: We propose a method to perform regression on partially labeled data, which is based on SSFCM (Semi-Supervised Fuzzy C-Means), an algorithm for semi-supervised classification based on fuzzy clustering. The proposed method, called SSFCM-R, precedes the application of SSFCM with a relabeling module based on target discretization. After the application of SSFCM, regression is carried out according to one out of two possible schemes: (i) the output corresponds to the label of the closest cluster; (ii) the output is a linear combination of the cluster labels weighted by the membership degree of the input. Some experiments on synthetic data are reported to compare both approaches.

1 INTRODUCTION

One of the methodologies at the heart of machine learning is Semi-Supervised Learning (SSL), a fusion of supervised and unsupervised learning, which was developed as a result of the widespread availability of unlabeled data in many fields and, at the same time, the dearth of labeled data. Indeed, in many real-world applications, a huge amount of data is continuously generated, but only a few of them are labeled. Labeling data is indeed time-consuming, and sometimes it is not possible due to the large volume of data, the speed of acquisition, or both. Cyber attacks, fraudulent transactions, or anomalies in monitoring systems are just a few examples where labeling all data is infeasible.

To overcome these limitations, SSL methods try to use as much unlabeled data as possible while requiring only a small amount of labeled data to drive prediction. Depending on the characteristics of the predicted output, two different approaches can be located under the SSL umbrella: Semi-Supervised Classification and Semi-Supervised Regression (Kostopoulos et al., 2018).

Numerous studies over the past years have dealt with the use of Semi-Supervised Classification approaches in many real-world applications such as text analysis (Duarte and Berton, 2023), e-health (Qayyum et al., 2023; Casalino et al., 2023; Kmita et al., 2022), image analysis (Qiu et al., 2023; Liu et al., 2023b), learning analytics (Liu et al., 2023a), manufacturing (Kim et al., 2023; Leite et al., 2020), energy management (Hao and Xu, 2023), just to mention a few. Moreover, semi-supervised extensions of clustering algorithms are often used for classification by exploiting the information deriving from the few available labels (González-Almagro et al., 2023).

In contrast, few works deal with Semi-Supervised Regression, which is still a lightly touched instance of SSL. Notable SSR techniques are the COREG (Zhou et al., 2005) and the SSKR (Semi-Supervised Kernel Regression) (Wang et al., 2006) algorithms. Kang et al. (Kang et al., 2016) introduce representative SSR algorithms such as Co-training, kernel, and graph-based regression methods.

In this work we propose SSFCM-R, a semi-supervised regression method that leverages the Semi-Supervised Fuzzy C-Means algorithm (SSFCM), previously employed for semi-supervised classification (Pedrycz and Waletzky, 1997). SSFCM-R extends SSFCM by adding some components useful to perform a prediction task linked to regression, starting from partially labeled data. At the core of the SSFCM-R method is a relabeling process based on a discretization of the available target values that enables the application of the SSFCM-based classifier. After the application of SSFCM, regression is carried out according to one out of two possible schemes: (i) the output corresponds to the label of the closest cluster; (ii) the output is a linear combination of the cluster labels weighted by the membership degree of the

[a] https://orcid.org/0000-0003-0713-2260
[b] https://orcid.org/0000-0002-6489-8628
[c] https://orcid.org/0000-0001-8712-023X

Casalino, G., Castellano, G. and Mencar, C.
Semi-Supervised Fuzzy C-Means for Regression.
DOI: 10.5220/0012195100003595
In *Proceedings of the 15th International Joint Conference on Computational Intelligence (IJCCI 2023)*, pages 369-375
ISBN: 978-989-758-674-3; ISSN: 2184-3236
Copyright © 2023 by SCITEPRESS – Science and Technology Publications, Lda. Under CC license (CC BY-NC-ND 4.0)

input. Three different discretization strategies have been compared to identify the most effective. They group data in subsets, called bins, which are assigned to continuous outputs, on the basis of different criteria (equal width distribution, quantiles, k-means). Synthetic data, of different complexity, have been generated to study the robustness of the proposed method. Also, different labeling percentages have been considered to study the algorithm behavior when the number of available labels decreases. Finally, the influence of the size of bins on the regression results has been analyzed.

The paper is organized as follows. In section 2 the proposed algorithm is formalized. The results of the numerical experiments are discussed in Section 3. Section 4 concludes the work and future directions of this research are outlined.

2 THE PROPOSED METHOD

The proposed SSFCM-R extends SSFCM (Semi-Supervised Fuzzy C-Means) (Pedrycz and Waletzky, 1997), which was originally designed for classification, by adding some mechanisms useful to accomplish a regression task.

2.1 SSFCM

SSFCM is a semi-supervised version of the FCM (Fuzzy C-Means) algorithm, which exploits partially labeled data to drive the clustering process. The algorithm generates clusters from a set of data that can be completely or partially labeled, by minimizing the following objective function:

$$J = \sum_{k=1}^{K} \sum_{j=1}^{N} u_{jk}^{m} d_{jk}^{2} + \alpha \sum_{k=1}^{K} \sum_{j=1}^{N} \left(u_{jk} - b_j f_{jk} \right)^{m} d_{jk}^{2} \quad (1)$$

where K is the number of clusters, N is the number of samples, $u_{jk} \in [0,1]$ is the membership degree of sample $\mathbf{x}_j$ in the k-th cluster; d_{jk} is the Euclidean distance between $\mathbf{x}_j$ and the center $\mathbf{c}_k$ of the k-th cluster; m is the fuzzification parameter (we will assume $m = 2$).

Peculiar to SSFCM is the introduction of variables $b_j = b(\mathbf{x}_j)$, where $b : X \mapsto \{0,1\}$ is such that $b(\mathbf{x}) = 1$ iff $\mathbf{x}$ is pre-labeled, i.e., its class value is known, and $f_{jk} = 1$ iff the j-th sample has the k-th class label, 0 otherwise (notice that f_{jk} is undefined when $b_j = 0$). The regularization parameter $\alpha \geq 0$ weights the second term of the objective function, that uses the class information; according to (Pedrycz and Waletzky, 1997), its value is the ratio of unlabeled data over all available data.

The outcome of SSFCM is a partition matrix $U = \left[u_{jk}\right]$ and a set of K cluster centroids

$$\mathbf{c}_k = \frac{\sum_{j=1}^{N} u_{jk}^{2} \mathbf{x}_j}{\sum_{j=1}^{N} u_{jk}^{2}} \quad (2)$$

that minimize (1). The details of the optimization schema are reported in (Pedrycz and Waletzky, 1997).

2.2 SSFCM-R

The core strategy of SSFCM-R is to consider the class label as the output of the function to be approximated. In order to enable regression through SSFCM, we first extend the original algorithm by admitting the possibility that the number of clusters (K) is greater than or equal to the number of class labels (C). In other words, different clusters can be assigned to the same class label. This extension is necessary because a function can have approximately the same value in different regions of the domain.

In this respect, the variables f_{jk} occurring in (1) are re-interpreted as follows: $f_{jk} = 1$ if the j-th sample has the same class label as the k-th cluster prototype, 0 otherwise. This change of interpretation requires evaluating the class label of a cluster prototype. To this pursuit, before starting the SSFCM clustering process, K labeled data are randomly chosen to initialize the prototypes, so that each cluster prototype is associated with a class label.

Once the clustering process is complete, the classification of an unlabeled data sample is based on a matching method using the derived labeled prototypes. Specifically, an unlabeled data sample is assigned the label of the closest prototype, according to the Euclidean distance.

Differently from SSFCM, where class labels do not have any specific structure, in SSFCM-R the class labels are numbers. Thus, SSFCM-R consists of the following three main stages:

1. Pre-processing: a discretization process and a subsequent relabeling process is applied to the target values to reduce the regression problem to one of classification;
2. Clustering is performed as in SSFCM;
3. Post-processing: given the discrete output values provided by SSFCM-based classification, the final predicted output value is computed as either by looking at the closest cluster, or by a linear combination of the discrete output values of all clusters.

2.2.1 Pre-Processing

Suppose that a set D of partially labeled data is available, representing an unknown function $f : \mathcal{X} \rightarrow \mathcal{Y}$. The set D consists of tuples $(\mathbf{x}, y)$, where $\mathbf{x} \in \mathcal{X}$ and $y \in \mathcal{Y} \cup \{\Box\}$. (The tuple $(\mathbf{x}, \Box)$ represents an unlabeled data sample.) The goal of regression is to find a model that approximates f starting from D.

Let

$$L = \{(\mathbf{x}, y) \in D | y \neq \Box\}$$

the subset of labeled data samples of D. We assume that L has cardinality $N_L > 0$. Let $Y = \{y \in \mathcal{Y} | (\mathbf{x}, y) \in L\}$ the set of numerical labels. The set Y is discretized into C intervals; for each interval $[a_i, b_i], i = 1, 2, \ldots, C$ the subset $Y_i = Y \cap [a_i, b_i]$ is computed (i.e., the subset of labels falling in the i-th interval) and the average value $\hat{y}_i$ is considered. The set of labels is therefore $\hat{Y} = \{\hat{y}_i | i = 1, 2, \ldots, C\}$.

The dataset D is then transformed into a new dataset $\hat{D}$ so that each labeled sample $(\mathbf{x}, y)$ is replaced with $(\mathbf{x}, \hat{y}_i)$, where $\hat{y}_i$ is the average of the subset Y_i the label y belongs to. The number C of bins is a hyperparameter that should be fixed in advance.

We consider three different discretization strategies:

D1: Equal-width discretization, separating all possible values into C bins, each having the same width;

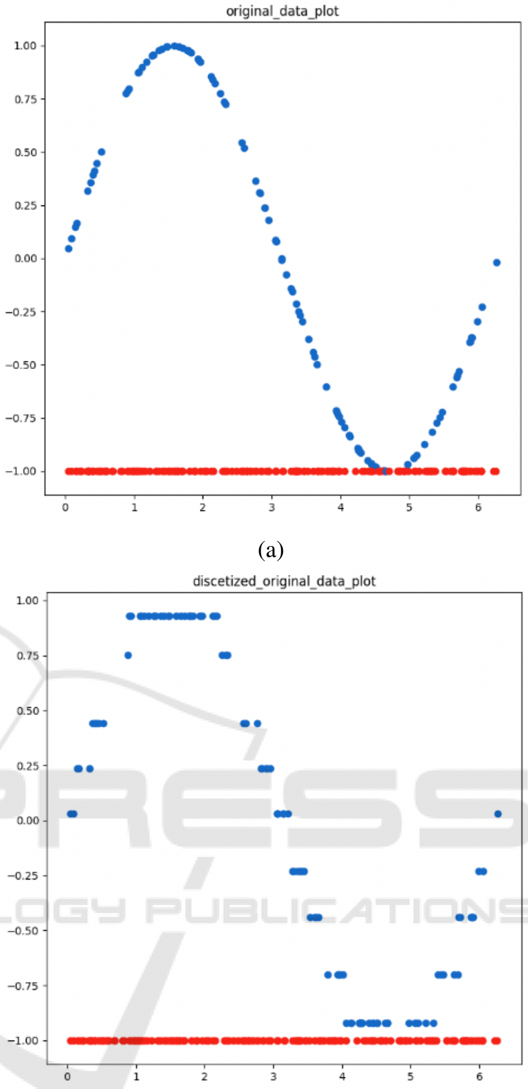
D2: Equal-frequency discretization, separating all possible values into C bins, each having the same amount of observations;

D3: The intervals are defined on the basis of the centroids produced by K-Means clustering.

As an example, fig. 1 shows the values of the sine function in $[0, 2\pi]$, before and after the discretization step. The first plot (fig. 1a) represents the sine function with partially labeled data (red dots correspond to unlabeled data). The second plot (fig. 1b) displays the target values after equal width discretization, with the number of bins equal to the 10% of labeled data (in this case $C = 9$).

2.2.2 Clustering

The pre-processed dataset $\hat{D}$ is used as input to SSFCM, as described in Sec. 2.1. The output is a collection of labeled cluster prototypes $(\mathbf{c_k}, \hat{y}_{i_k})$ where $\hat{y}_{i_k} \in \hat{Y}$ and a partition matrix $U = [u_{jk}]$ of each data sample (either labeled or unlabeled) to each cluster.



(a)

(b)

Figure 1: Original partially labeled data obtained by sine function (a) and equal width discretized data with 9 bins (b).

2.2.3 Post-Processing

Given a new input $\mathbf{x} \in \mathcal{X}$, the estimated value y can be computed according to one out of two possible strategies:

(*max*) The closest prototype $\mathbf{c}_k$ to $\mathbf{x}$ is determined; then, the estimated value $y_{\max}$ corresponds to the class label $\hat{y}_{i_k}$;

(*sum*) The membership degrees of $\mathbf{x}$ to each cluster are determined by using the formula used in SSFCM to compute the membership degrees for unlabeled data (Pedrycz and Waletzky, 1997):

$$u_k(\mathbf{x}) = \frac{1}{\sum_{h=1}^{K} \left(\frac{d(\mathbf{x}, \mathbf{c}_k)}{d(\mathbf{x}, \mathbf{c}_h)} \right)^2}$$

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Table 1: Example of results given by SSFCM-R.

$\mathbf{x}$	Labeled	$\hat{y}$	y_{sum}	y_{max}	y (target)
3.67	No	□	-0.31	-0.70	-0.50
3.95	Yes	-0.70	-0.66	-0.70	-0.72
1.06	Yes	0.93	0.91	0.93	0.87
2.25	Yes	0.75	0.73	0.75	0.77
3.98	No	□	-0.40	-0.70	-0.74

Since $\sum_{k=1}^{K} u_k(\mathbf{x}) = 1$, then the estimated value y corresponds to the weighted average

$$y_{\text{sum}} = \sum_{k=1}^{K} u_k(\mathbf{x}) \hat{y}_{i_k}$$

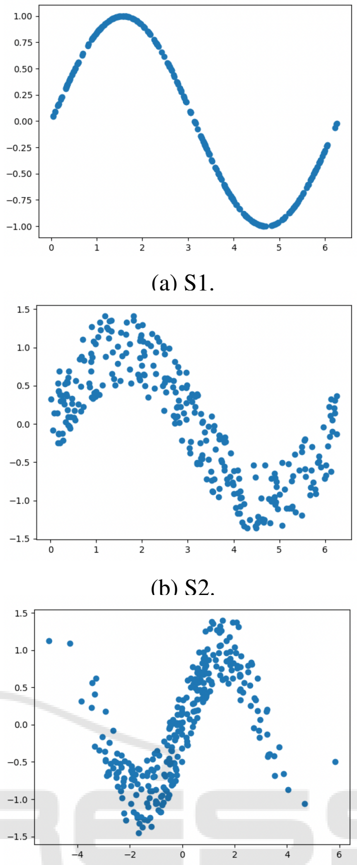
Table 1 shows an example of results obtained by SSFCM-R for five data points (labeled and not labeled). The third column shows the class, in terms of discretized bin value, that has been assigned. The last column indicates the real output for the given input. Two more values are reported: y_{sum} and y_{max}, which are estimated according to the two aforementioned strategies.

3 EXPERIMENTS

Some experiments have been conducted to verify the effectiveness of the proposed approach, by varying the discretization strategy, the percentage of labeled data, and the number of bins used for discretization. Moreover, three synthetic datasets, of different complexity, have been created. For the sake of simplicity, bi-dimensional data have been produced, where the second dimension is the value to predict. Noise has been added to the simplest dataset with different distributions (uniform and normal), thus making the predictive problem more complex to solve.

Partial labeling has been simulated in order to evaluate the robustness of the proposed algorithm in the presence of unlabeled data at varying frequencies. Particularly, eight labeling percentages have been considered, namely: 10%, 30%, 50%, 60%, 70%, 80%, 90%, and 100%. Also, three different bin sizes have been compared by considering the 10%, 20%, and 30% of labeled data. Different bin sizes have been used, from 3 to 90.

The standard Mean Square Error (MSE) and the computational time (TIME) have been used as evaluation metrics for the comparison. We compare the errors obtained with the two post-processing methods: *MSE sum* and *MSE max*.



(a) S1.

(b) S2.

(c) S3.

Figure 2: Synthetic datasets.

3.1 Data

Figure 2 shows the three synthetic dataset *S1*, *S2*, and *S3* created as follows:

- *S1* consists of the evaluation of the sine function on 300 data points generated with a uniform distribution in the interval $[0, 2\pi]$: $Y = \sin X$ where $X \sim U([0, 2\pi])$
- *S2* consists of the evaluation of the sine function on 300 data points generated with a uniform distribution in the interval $[0, 2\pi]$ as in *S1* plus a noise term with uniform distribution: $Y = \sin X + \varepsilon$ where $X \sim U([0, 2\pi])$ and $\varepsilon \sim U([-0.5, 0.5])$
- *S3* consists of the evaluation of the sine function on 300 points generated from a normal distribution plus a noise term with uniform distribution: $Y = \sin X + \varepsilon$ where $X \sim N(0, \pi)$ and $\varepsilon \sim U([-0.5, 0.5])$

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

Table 2: Comparison of different discretization strategies, varying the percentages of adopted bins, with S1, S2, and S3 datasets.

Strategy	Bin %	MSE max	MSE sum	Time
D3	10	0.18	0.12	130.20
	20	0.20	0.12	390.63
	30	0.18	0.10	687.84
D2	10	0.15	0.12	110.80
	20	0.18	0.13	326.19
	30	0.27	0.14	1228.27
D1	10	0.17	0.11	82.33
	20	0.19	0.13	397.84
	30	0.19	0.10	626.00

(a) S1 dataset.

Strategy	Bin %	MSE max	MSE sum	Time
D3	10	0.22	0.19	104.78
	20	0.18	0.18	448.68
	30	0.22	0.18	939.75
D2	10	0.20	0.17	91.56
	20	0.20	0.17	402.34
	30	0.21	0.18	969.61
D1	10	0.21	0.19	106.71
	20	0.18	0.18	385.93
	30	0.20	0.18	724.61

(b) S2 dataset.

Strategy	Bin %	MSE max	MSE sum	Time
D3	10	0.26	0.20	110.07
	20	0.26	0.21	383.59
	30	0.26	0.22	666.51
D2	10	0.29	0.21	97.49
	20	0.30	0.22	323.47
	30	0.30	0.20	833.30
D1	10	0.25	0.19	90.64
	20	0.23	0.19	309.12
	30	0.23	0.20	635.72

(c) S3 dataset.

3.2 Results

Table 2 shows the numerical results obtained by varying the bin percentages, the discretization methods, and the datasets. Average measures over all the labeling percentages have been reported for the three datasets $S1$, $S2$, and $S3$.nExpectedly, as the complexity of data increases, errors also increase, but this is not the only parameter to consider. In fact, the *MSE max* obtained with the discretization $D1$, and bin % 30 on the simplest data S1, is higher than the error obtained with $D1$ on $S3$, the most complex dataset. Thus, different combinations of parameters, affecting the regression results, are analyzed. The computational time is strictly proportional to the number of bins; this is observed for each data and discretization method.

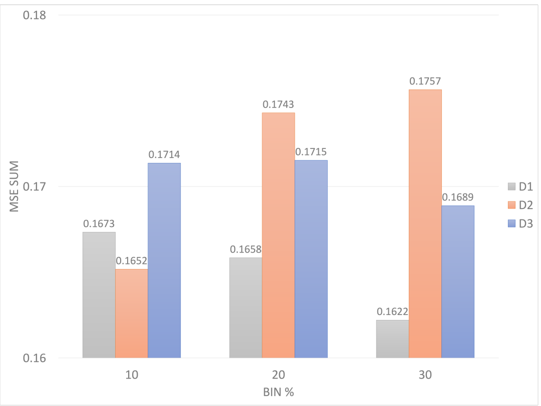


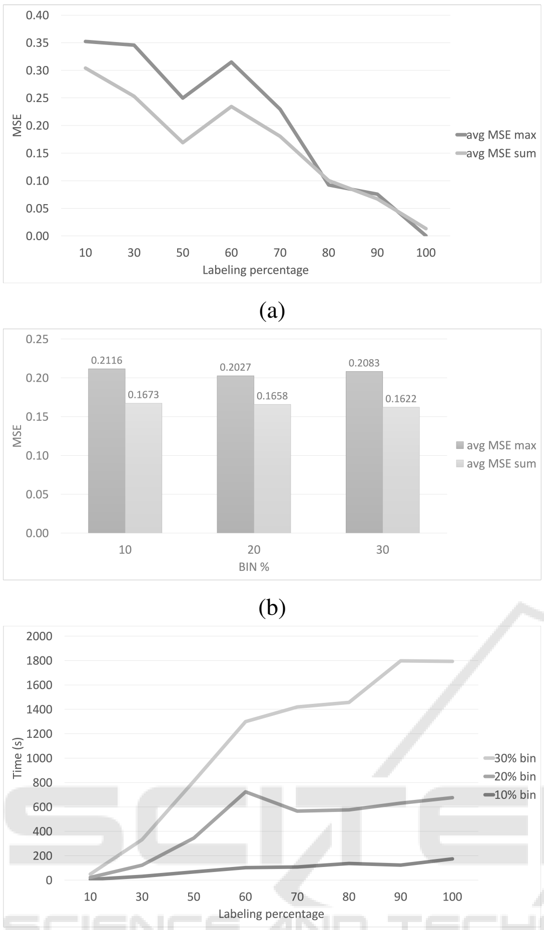
Figure 3: Average MSE values varying the discretization method and the bin percentages, over the labeling percentages and the datasets.

To better analyze the results, charts focusing on each parameter (discretization method, bin percentages, labeling percentages, and post-processing method) and varying the others, have been reported. Figure 3 compares the three discretization methods, by varying the bin percentages. Average values over the labeling percentages and the three datasets have been reported. It is observed that the equal width strategy (D1) has the lowest MSE on average, regardless of the dataset complexity. Also, by increasing the bin percentage, the equal width strategy provides the lowest MSE, with respect to the other two approaches. Moreover, the equal width strategy has also the lowest computational time, in seconds, among all the considered strategies (D1=373.21, D2=487.00, D3=429.12). For this reason, we now focus on the equal width discretization strategy, and analyze the influence of the labeling percentage and the bin size, in terms of *MSE sum* and *MSE max*, averaged on all the remaining parameters.

Figure 4a shows the influence of the labeling percentages by averaging the results from different datasets, and bin percentages. It could be observed that the labeling percentage strongly affects the predictions. Indeed, as expected, as the number of labels increases, the error decreases. However, with a labeling percentage lower than 60% the algorithm is not stable, and peaks could be observed in the graph. As the labeling percentage increases over 60% the error significantly decreases. It is also observed that *MSE sum* is significantly lower than *MSE max* when the percentage of labeled data is low, while both converge to similar values for higher labeling percentages.

We analyzed the influence of the bin size on the results to identify the best percentage. Figure 4b shows the average measures over the three datasets, varying the labeling percentages. It could be observed that the bin percentage does not influence the predictive capability of the algorithm, returning comparable errors.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS



(a)

(b)

(c)

Figure 4: Effectiveness and efficiency of SSCFCM-R with equal width discretization.

Finally, the value (in seconds) of computational time for the equal width discretization, varying the labeling percentage and the bin percentage, are summarized in Fig. 4c, by averaging results over all datasets. It can be seen that computational time increases as the percentage of bins and the labeling percentage increase. Overall, since the bin size does not influence the effectiveness of the methods, whilst it does influence the computational time, a small number of bins (10% of labeled data), is the best choice for both high efficiency and effectiveness.

4 CONCLUSIONS

We have proposed SSFCM-R, an extension of the Semi-Supervised Fuzzy C-Means (SSFCM) algorithm that is suitable for regression. SSFCM-R leverages a discretization mechanism to move from a continuous domain (useful to solve a regression problem) to a discrete one (that SSFCM is able to process). To this aim three different discretization strategies have been compared, based on equally sized bins (subsets of data), percentiles, and k-means. Experiments have been performed to analyze the effectiveness of the proposed approach in different conditions. Particularly, the influence of data complexity, discretization strategy, labeling percentage, and number of bins, on the results, has been studied. In this preliminary work, synthetic data has been produced for controlled experiments. The equal width strategy has been proven to be the more effective, with a lower error if compared with the other discretization strategies. Also, whilst the number of labeled data influences the results, resulting in low performances for labeling percentages lower than 60%, the number of adopted bins does not. Thus, since the computational time is strictly related to the number of bins, a small number is preferable. Finally, the post-processing method *sum* has shown to always achieve lower errors than the *max* method.

Overall, this is the first attempt to modify SSFCM for regression. This study has been useful to identify the parameters that mostly affect the results and which of them allow the algorithm to perform better. Future work will be devoted to studying different discretization strategies, not depending on the labeling percentage and the data complexity. Also, the effectiveness of the proposed approach will be evaluated on real-world applications, and it will be compared with other semi-supervised regression algorithms.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

ACKNOWLEDGEMENTS

G. Casalino acknowledges funding from the European Union PON project Ricerca e Innovazione 2014-2020, DM 1062/2021. G. Castellano's work is partially supported by the FAIR - Future AI Research (PE00000013) project, Spoke 6 - Symbiotic AI (CUP H97G22000210007), under the NRRP MUR program funded by NextGenerationEU. All the authors are members of the INdAM GNCS research group. The authors thank the students Michele Gabriele Addati and Nunziangelo Bruno for their contribution to developing the code and running the experiments.

REFERENCES

Casalino, G., Castellano, G., Hryniewicz, O., Leite, D., Opara, K., Radzieskewska, W., and Kaczmarek-Majer, K. (2023). Semi-supervised vs. supervised learning for mental health monitoring: a case study on the classification of bipolar disorder episodes. *Journal of Applied Mathematics and Computer Science*, 33(3).

Duarte, J. M. and Berton, L. (2023). A review of semi-supervised learning for text classification. *Artificial Intelligence Review*, pages 1–69.

González-Almagro, G., Peralta, D., De Poorter, E., Cano, J.-R., and García, S. (2023). Semi-supervised constrained clustering: An in-depth overview, ranked taxonomy and future research directions. *arXiv preprint arXiv:2303.00522*.

Hao, L. and Xu, Y. (2023). Semi-supervised learning based occupancy estimation for real-time energy management using ambient data. *IEEE Internet of Things Journal*.

Kang, P., Kim, D., and Cho, S. (2016). Semi-supervised support vector regression based on self-training with label uncertainty: An application to virtual metrology in semiconductor manufacturing. *Expert Systems with Applications*, 51:85–106.

Kim, G., Choi, J. G., Ku, M., and Lim, S. (2023). Developing a semi-supervised learning and ordinal classification framework for quality level prediction in manufacturing. *Computers & Industrial Engineering*, 181:109286.

Kmita, K., Casalino, G., Castellano, G., Hryniewicz, O., and Kaczmarek-Majer, K. (2022). Confidence path regularization for handling label uncertainty in semi-supervised learning: use case in bipolar disorder monitoring. In *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8. IEEE.

Kostopoulos, G., Karlos, S., Kotsiantis, S., and Ragos, O. (2018). Semi-supervised regression: A recent review. *Journal of Intelligent & Fuzzy Systems*, 35(2):1483–1500.

Leite, D., Decker, L., Santana, M., and Souza, P. (2020). Egfc: Evolving gaussian fuzzy classifier from never-ending semi-supervised data streams – with application to power quality disturbance detection and classification. In *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–9.

Liu, Z., Kong, W., Peng, X., Yang, Z., Liu, S., Liu, S., and Wen, C. (2023a). Dual-feature-embeddings-based semi-supervised learning for cognitive engagement classification in online course discussions. *Knowledge-Based Systems*, 259:110053.

Liu, Z., Lai, Z., Ou, W., Zhang, K., and Huo, H. (2023b). Discriminative sparse least square regression for semi-supervised learning. *Information Sciences*, 636:118903.

Pedrycz, W. and Waletzky, J. (1997). Fuzzy clustering with partial supervision. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics*, 27(5):787–95.

Qayyum, A., Tahir, A., Butt, M. A., Luke, A., Abbas, H. T., Qadir, J., Arshad, K., Assaleh, K., Imran, M. A., and Abbasi, Q. H. (2023). Dental caries detection using a semi-supervised learning approach. *Scientific Reports*, 13(1):749.

Qiu, L., Cheng, J., Gao, H., Xiong, W., and Ren, H. (2023). Federated semi-supervised learning for medical image segmentation via pseudo-label denoising. *IEEE Journal of Biomedical and Health Informatics*.

Wang, M., Hua, X.-S., Song, Y., Dai, L.-R., and Zhang, H.-J. (2006). Semi-supervised kernel regression. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 1130–1135. IEEE.

Zhou, Z.-H., Li, M., et al. (2005). Semi-supervised regression with co-training. In *IJCAI*, volume 5, pages 908–913.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS