Simultaneous learning of fuzzy sets

Luca Cermenati[1], Dario Malchiodi[1], and Anna Maria Zanaboni[1]

Università degli Studi di Milano, Dipartimento di Informatica,
{malchiodi, zanaboni}@di.unimi.it

Abstract. We extend a procedure based on support vector clustering and devoted to inferring the membership function of a fuzzy set to the case of a universe of discourse over which several fuzzy sets are defined. The extended approach learns simultaneously these sets without requiring as previous knowledge either their number or labels approximating membership values. This data-driven approach is completed via expert knowledge incorporation in the form of predefined shapes for the membership functions. The procedure is successfully tested on a benchmark.

Keywords: fuzzy sets, membership inference, modified SV clustering

1 Introduction

Fuzzy sets constitute a sort of backbone for all fuzzy constructs, such as fuzzy models, fuzzy classifiers and fuzzy reasoning schemes. Therefore the quality of the former directly impacts on the performance and readability of such constructs. The design of fuzzy sets is a crucial problem both in the theory and practice of fuzzy methodologies, and indeed there is a broad spectrum of approaches aiming at building fuzzy sets. On one side, fuzzy sets are designed exploiting human knowledge through a mix of different interpretations [9], expert-driven approches [14], predefined shapes for membership functions [15], and specific de-granulation processes [17, 5]. However, the availability of experts in the modeled domain might be a critical aspect, and in any case this kind of estimation has been shown to suffer from incompleteness, inconsistencies or bias linked to the perception of specific concepts captured by humans [18, 16]. For these reasons, on the other extreme of the spectrum of methodologies there are data-driven approaches, relying only on experimental evidence (see for instance [19, 10, 1]). Several strategies actually position themselves between the two extremes, combining them in a hybrid fashion [11, 8, 4, 2].

In this work we propose a technique mixing data-driven and expert-driven approach. The former is used to infer the number of fuzzy sets in a given domain and their approximate localization, while the latter is used to define *a priori* the family shape of such sets. The starting point is a procedure exploiting a modified support vector clustering approach [12] in order to learn the membership function of a single fuzzy set, starting from examples of objects labeled with their membership value (see also [6] for a similar approach based on modified regression). In this paper, such approach is extended in two significant ways: on

the one hand, the need of labels representing membership values of the observed objects is dropped, and on the other one the inference process now concerns *several* fuzzy sets simultaneously. The number and location of such sets is found through application of the original version of the support vector clustering algorithm [7], in order to label objects via approximate membership values; the lables are subsequently used in order to separately learn each fuzzy set.

The paper is structured as follows: Sect. 2 briefly describes the technique used for inferring the membership function of a single fuzzy set on the basis of a sample of objects in the universe of discourse, each one labeled with its membership degree to such set. Sect. 3 exploits the above mentioned technique in order to simultaneously learn several fuzzy sets, starting from a set of unlabeled objects. Sect. 4 describes a preliminary experimental campaign. Some concluding remarks end the paper.

2   Inferring the membership function to a fuzzy set

In this section we briefly recall the procedure used in order to learn the membership function to a fuzzy set starting from a labeled sample $\{(x_1, \mu_1)\ldots, (x_m, \mu_m)\}$, where for each i the value x_i denotes an object in a space X and the label μ_i is the membership grade of x_i to a fixed, yet unknown, fuzzy set A. Readers interested in further details may refer to the original paper [12].

The main component of the learning procedure is a modified version of the support vector clustering algorithm proposed in [7], enhanced in order to deal with labels $\mu_1, \ldots, \mu_m$. Namely, objects are transformed through a nonlinear mapping Φ onto a space within which a sphere S is found such that:

- the higher μ_i, the closer x_i is to the border of S (and when $\mu_i = 1$ the object belongs to S);
- vice versa, as μ_i gets smaller the corresponding object lies farther from S; and
- the radius of S is constrained to be as small as possible.

More precisely, denoting by a and R the center and the radius of S, respectively, this amounts to considering the problem

$$\min R^2 + C \sum (\xi_i + \tau_i) \tag{1}$$
$$\mu_i ||\Phi(x_i) - a||^2 \leq \quad \mu_i R^2 + \xi_i \ , \tag{2}$$
$$(1 - \mu_i)||\Phi(x_i) - a||^2 \geq (1 - \mu_i)R^2 - \tau_i \ , \tag{3}$$
$$\xi_i \geq 0, \tau_i \geq 0 \ , \tag{4}$$

where ξ_i and τ_i denote slack variables allowing the management of possible outliers and $C > 0$ is a hyperparameter defining a trade-off between the two components of the objective function in (1). As usual with support vector methods,

the solution can be found considering the dual version of (1–4), which reads

$$\max \sum_{i=1}^{m} \epsilon_i k(x_i, x_i) - \sum_{i,j=1}^{m} \epsilon_i \epsilon_j k(x_i, x_j) \tag{5}$$

$$\sum_{i=1}^{m} \epsilon_i = 1, \tag{6}$$

$$-C(1 - \mu_i) \le \epsilon_i \le C\mu_i, \tag{7}$$

where $\epsilon_i = \alpha_i \mu_i - \beta_i(1 - \mu_i)$ for each $i = 1, \ldots, m$ (being α_i and β_i the lagrangian multipliers associated to the constraints (2) and (3), respectively) and $k(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$, that is k is the *kernel function* associated to the mapping Φ. Thus the considered objects don't need to be numerical vectors: the only requirement is the existence of a similarity measure k between them. For instance, [13] applies this technique to the problem of detecting a set of reliable axioms starting from a set of OWL formulas.

The experiments shown in Sect. 4 make use of the *gaussian kernel*

$$k(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right),$$

although other choices are possible. Here $\sigma > 0$ is a second hyperparameter to be tuned when performing experiments.

Once the optimal values $\epsilon_1^*, \ldots \epsilon_m^*$ of (5–7) have been computed, it is easy to show that

$$R^2(x) = k(x, x) - 2\sum_{i=1}^{m} \epsilon_i^* k(x, x_i) + \sum_{i,j=1}^{m} \epsilon_i^* \epsilon_j^* k(x_i, x_j) \tag{8}$$

amounts to the squared distance between a and the image through Φ of a generic object x. Moreover, given any k such that $-C(1 - \mu_k) < \epsilon_k^* < C\mu_k$, the quantity $R^{2,*} = R^2(x_k)$ equals the squared radius of S. Thus it is easy to take a further step and induce an approximation $\hat{\mu}_A$ of the membership function of A as follows: having fixed a suitable *fuzzifier* (that is, a nonincreasing function $f : \mathbb{R}^+ \mapsto [0, 1]$ turning the distance of the image of a generic point from S into a membership value), and given a generic object x^N, let $\hat{\mu}_A(x^N) = f(R^2(x^N) - R^{2,*})$. A simple choice for f is that of a piecewise linear function equal to 1 when the image of an object lies within S, equal to 0 when such image is farther from S than the farthest observerd distance $x_{\max}$, and decreases linearly between these extremes in the remaining cases:

$$f(x) = \begin{cases} 1 & \text{if } x < 0, \\ 1 - \frac{x}{x_{\max}} & \text{if } 0 \le x \le x_{\max}, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

More complex choices for f, such as a special exponential decaying function linked with the quantiles of the observed distances of x_is from the border of S, may be considered (see [13] for further details).

Algorithm 1 Procedure for simultaneous learning of memberships to several fuzzy sets, using crisp intermediate membership values

---

Input:

- a set $T = \{x_1, \ldots, x_m\}$ of objects;
- two hyperparameters $C, \sigma > 0$;
- a fuzzifier $f : X \mapsto [0, 1]$.

1. Apply support vector clustering to T using C and σ as hyperparameters, obtaining a partition $T_1, \ldots, T_c$ of T.
2. For each $k = 1, \ldots, c$:
   (a) For each $i = 1, \ldots m$ set

$$\mu_i = \begin{cases} 1 & \text{if } x_i \in T_k, \\ 0 & \text{otherwise.} \end{cases}$$

   (b) Starting from T and $\{\mu_1, \ldots \mu_m\}$, infer $\hat{\mu}_{A_k}$ using the procedure of Sect. 2, the fuzzifier f, and the hyperparameters C and σ.

Output: $\hat{\mu}_{A_1}, \ldots, \hat{\mu}_{A_c}$.

---

3   Simultaneously inferring several membership functions

As a general case, the method outlined in the previous section requires as input a set of objects each labeled with a degree of membership, thus a $[0, 1]$-valued number. However, the procedure can be run even when the information abut membership degrees is not available, yet each object is labeled with a $\{0, 1\}$ value denoting its crisp membership to a (classical) set.

In this section we address the more general case in which even this weaker form of information is missing, that is the only available data is the set $T = \{x_1, \ldots, x_m\}$ of objects, in the idea that *several* fuzzy sets are defined on the universe of discourse X. As a first stage, the original version of the support vector clustering can be applied in order to detect a sort of *core* for each fuzzy set. This step can be descrbed as a simplified version of the procedure described in Sect. 2: now S identifies with the smallest sphere containing most of the images of objects, and an analogous procedure allows to compute:

- a function R^2_{cluster} mapping a generic object to the squared distance of its image through Φ from the center of S, and
- the squared radius $R^{2,*}_{\text{cluster}}$ of S.

Now, let x_a and x_b denote two objects in X belonging to different clusters, and consider the segment joining them. It can be shown that the trajectory described by the images through Φ of all points laying on this segment is not fully contained in S [7]. This fact can be easily checked considering suitable discretizations of the segments joining all possible pairs of objects. As a result, the set T can be partitioned in c subsets, namely $T = \cup_{k=1}^c T_k$ and $T_i \cap T_j = \emptyset$ for each $i \neq j$. These

---

Algorithm 2 Procedure for simultaneous learning of memberships to several fuzzy sets, using fuzzy intermediate membership values

---

Input:

- a set $T = \{x_1, \ldots, x_m\}$ of objects,
- two hyperparameters $C, \sigma > 0$;
- a fuzzifier $f : X \mapsto [0, 1]$.

1. Apply support vector clustering to T using C and σ as hyperparameters, obtaining:
   - a partition $T_1, \ldots, T_c$ of T);
   - a mapping $R^2_{\text{cluster}} : X \mapsto \mathbb{R}^+$;
   - a value $R^{2,*}_{\text{cluster}}$.
2. For each $k = 1, \ldots, c$:
   (a) For each $i = 1, \ldots m$ set $\mu_i = f(R^2_{\text{cluster}}(x_i) - R^{2,*}_{\text{cluster}})$
   (b) Starting from T and $\{\mu_1, \ldots \mu_m\}$, infer $\hat{\mu}_{A_k}$ using the procedure of Sect. 2, the fuzzifier f, and the hyperparameters C and σ.

Output: $\hat{\mu}_{A_1}, \ldots, \hat{\mu}_{A_c}$.

---

subsets can be interpreted as an initial approximation of the localization for c fuzzy sets $A_1, \ldots, A_c$. Thus, for each $k \in \{1, \ldots, c\}$, objects in T_k and $T \backslash T_k$ can be assigned a membership equal to 1 and 0, respectively. The next step consists in applying the procedure of Sect. 2 in order to obtain an approximation $\hat{\mu}_{A_k}$ of the membership function μ_{A_k}. Algorithm 1 formalizes this procedure. It is worth noting that the number c of obtained fuzzy sets is not fixed *a priori*, albeit it is influenced from the choice of hyperparameters, notably C. This means that any preexisting clue about the number of sets can in principle be used in order to restrict the variability of hyperparameters during the model selection phase.

A variant of the proposed technique considers a different way of computing the intermediate values $\mu_1, \ldots, \mu_c$ in step 2a of Algorithm 1. Indeed, a better approximation might be found in terms of:

- the squared distance $R^2_{\text{cluster}}(x_i)$ of the images of each x_i from the center of the sphere learnt during the support vector clustering phase, and
- the squared radius $R^{2,*}_{\text{cluster}}$ of the same sphere.

More precisely, having fixed the fuzzifier f, $f(R^2_{\text{cluster}}(x_i) - R^{2,*}_{\text{cluster}})$ can be used as a guess for the membership value of x_i, as illustrated in Algorithm 2.

4   Experiments

We used as benchmark the Iris dataset, consisting of 150 observations of iris plants in terms of length and width of their petal and sepal. The observations are organized in the three classes *Setosa*, *Virginica*, and *Versicolor*, with only the first class being linearly separable from the remaining ones. As a first experiment, for sake of visualization we extracted the first two principal components from the observations and we iterated for ten times the holdout scheme described below.
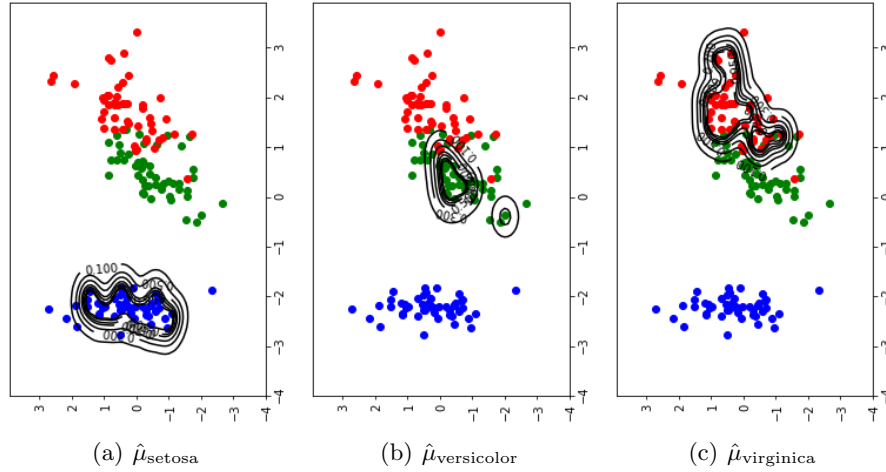
(a) $\hat{\mu}_{\mathrm{setosa}}$ (b) $\hat{\mu}_{\mathrm{versicolor}}$ (c) $\hat{\mu}_{\mathrm{virginica}}$

Fig. 1. Contour plots of the functions $\hat{\mu}_{\mathrm{setosa}}$, $\hat{\mu}_{\mathrm{versicolor}}$, and $\hat{\mu}_{\mathrm{virginica}}$ learnt during one of the ten holdout iterations of the first experiment. Blue, green and red bullets respectively denote observations from the setosa, versicolor, and virginica classes, after the first two principal components have been extracted from the orginal data.

- After having randomly shuffled all data, we partitioned the benchmark into three sets devoted to training, model selection, and model validation (retaining in each one the 80%, 10%, and 10% of available data, respectively).
- For each choice of C and σ in a grid, we applied Algorithm 1 and obtained three membership functions which we called $\hat{\mu}_{\mathrm{setosa}}$, $\hat{\mu}_{\mathrm{virginica}}$, and $\hat{\mu}_{\mathrm{versicolor}}$, and we used them to compute the accuracy in classification of data in the validation set (namely, each item was assigned to the class whose corresponding membership function attained the maximum value). It is worth noting that, being the number c of fuzzy sets learnt by the algorithm, the former could be different from the expected value corrisponding to the three classes in the dataset. We dropped all singleton clusters after the initial phase, and sorted the remaining ones w.r.t. their size. The three biggest resulting clusters were subsequently associated each to the most represented class[1].
- The choice of hyperparameters maximizing the above mentioned accuracy was selected in order to retrain a model, now merging training and validation sets, and the result was scored in terms of accuracy on the test set.

The average accuracy on the ten holdout iterations was 0.8, with a standard deviation of 0.11. Figure 1 shows the contour plots of the inferred membership functions for the three classes in the benchmark in one of the iterations.

---

[1] Note that even with this careful setting, there is no guarantee that the three clusters will get associated injectively to the three available classes. We simply re-executed the iterations in which these cases occurred.

Table 1. Results of ten holdout procedures of the simultaneous fuzzy set learning procedure on the Iris dataset. Each row shows average and standard deviation (columns Avg. and Stdev., respectively) of train and test error, in function of the number of principal components extracted from the original sample.

	SVC				FCM			
N. of principal	Train error		Test error		Train error		Test error	
components	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.
2	0.82	0.05	0.8	0.11	0.83	0.01	0.80	0.05
3	0.87	0.03	0.83	0.07	0.83	0.02	0.84	0.07
4	0.85	0.05	0.89	0.09	0.83	0.01	0.84	0.08

We repeated the experiment considering three and four principal components, obtaining the results shown in Table 1. The accuracy rates are sufficiently high to state that the method succeeds in rebuilding the information about the three classes despite such information has been hidden to the learning procedure. The table also shows the performance of an analogous procedure based on the fuzzy C-means algorithm as base learner. The results, slightly in favour of the proposed methodology, could be improved using a more refined model validation scheme and/or trying different shapes, notably nonlinear ones, for the fuzzifier. We are currently testing Algorithm 2 on the same benchmark.

5   Conclusions

The design of fuzzy sets is an essential component in the search of successful fuzzy models. We considered how to extend an existing learning algorithm for the membership function of a fuzzy set on the basis of objects labeled with the corresponding membership grades. This algorithm was enhanced in order to simultaneoulsy learn several fuzzy sets defined in the considered universe of discourse. The number of sets can in principle be induced directly from data, and the latter do not need to be labeled with any information concerning the membership grades w.r.t. the models to be learnt. We preliminarily tested the proposed approach on the Iris dataset, showing how the three existing clusters can be discovered without using the class information recorded in the benchmark, but only with little post-processing, as mentioned. Besides a more refined experimental campaign, the technique can be further refined analyzing how pre-existing information about the number of fuzzy sets is related to a proper choice of the hyperparameters of the learning algorithm, and by testing the effect of using different nonlinear fuzzifiers. An analysis of the theoretical properties of this approach, for instance exploiting game-based results [3], as well as its extension to the field of type-2 fuzzy sets, can also be envisaged.

Acknowledgments

References

1. Afify, A.: FuzzyRULES-II: A new approach to fuzzy rule induction from numerical data. Frontiers in Artificial Intelligence and Applications **281**, 91–100 (2016)
2. Apolloni, B., Bassis, S., Gaito, S., Malchiodi, D.: Bootstrapping complex functions. Nonlinear Analysis: Hybrid Systems **2**(2), 648–664 (2008)
3. Apolloni, B., Bassis, S., Gaito, S., Malchiodi, D., Zoppis, I.: Controlling the losing probability in a monotone game. Information Sciences **176**(10), 1395–1416 (2006)
4. Apolloni, B., Bassis, S., Malchiodi, D., Pedrycz, W.: Interpolating support information granules. In: Proc. of the 16th Int. Conf. on Artificial Neural Networks - Part II, ICANN'06, pp. 270–281. Springer-Verlag, Berlin, Heidelberg (2006)
5. Apolloni, B., Iannizzi, D., Malchiodi, D., Pedrycz, W.: Granular regression. In: Neural nets: 16th italian workshop on neural nets, WIRN2005 and international workshop on natural and artificial immune systems, NAIS 2005, vol. 3931 LNCS, pp. 147–156. Springer (2006)
6. Apolloni, B., Malchiodi, D., Valerio, L.: Relevance regression learning with support vector machines. Nonlinear Analysis, Theory, Methods and Applications **73**(9), 2855–2864 (2010)
7. Ben-Hur, A., Horn, D., Siegelmann, H.T., Vapnik, V.: Support vector clustering. Journal of machine learning research **2**(Dec), 125–137 (2001)
8. Dubois, D., Hájek, P., Prade, H.: Knowledge-driven versus data-driven logics. Journal of logic, Language and information **9**(1), 65–89 (2000)
9. Dubois, D., Prade, H.: The three semantics of fuzzy sets. Fuzzy Sets and Systems **90**, 141–150 (1997)
10. Hong, T.P., Lee, C.Y.: Induction of fuzzy rules and membership functions from training examples. Fuzzy sets and Systems **84**(1), 33–47 (1996)
11. Lo Presti, M., Poluzzi, R., Zanaboni, A.M.: Synthesis of fuzzy controllers through neural networks. Fuzzy Sets and Systems **71**, 47–70 (1995)
12. Malchiodi, D., Pedrycz, W.: Learning membership functions for fuzzy sets through modified support vector clustering. In: F. Masulli, G. Pasi, R.R. Yager (eds.) Fuzzy Logic and Applications - 10th International Workshop, WILF 2013, Genoa, Italy, November 19-22, 2013. Proceedings, vol. LNCS 8256, pp. 52–59. Springer (2013)
13. Malchiodi, D., Tettamanzi, A.G.B.: Predicting the possibilistic score of OWL axioms through modified support vector clustering. In: SAC 2018: Symposium on Applied Computing, April 9–13, 2018, Pau, France. ACM, New York, NY, USA (2018). DOI 10.1145/3167132.3167345
14. Mottaghi-Kashtiban, M., Khoei, A., Hadidi, K.: Optimization of rational-powered membership functions using extended kalman filter. Fuzzy Sets and Systems **159**(23), 3232–3244 (2008)
15. Nguyen, H., Walker, E.: A First Course in Fuzzy Logic. Chapman Hall, CRC Press, Boca Raton (1999)
16. de Oliveira, J.V.: Semantic constraints for membership function optimization. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans **29**(1), 128–138 (1999)
17. Pedrycz, W.: Granular Computing: Analysis and Design of Intelligent Systems. CRC Press/Francis Taylor, Boca Raton (2013)
18. Setnes, M., Babuska, R., Verbruggen, H.B.: Transparent fuzzy modelling. International Journal of Human-Computer Studies **49**(2), 159–179 (1998)
19. Wu, J.N., Cheung, K.C.: An efficient algorithm for inducing fuzzy rules from numerical data. In: Proceedings of the Eleventh International FLAIRS Conference, pp. 221–224. AAAI (1998)