(Semi-Supervised) Fuzzy Clustering

Kamil Kmita

Analysis of Uncertain Data
Research Workshop @ MiNI PW

Outline

Technicalities

- we will use Rmd (R Markdown) from now on,
- install `devtools` and try to install the `ssfclust` library.

```
devtools::install_github("ITPsychiatry/ssfclust@refactor")
```

Fuzzy C-Means

The roots of Fuzzy C-Means

[Bez] in Chapter 2:

- p. 18, defines hard 2-partition,
- p. 20, defines fuzzy 2-partition.

Fuzzy C-Means, because the direct inspiration is taken from the fuzzy set theory to represent the degree of belonging of object x to cluster k with a characteristic function $\mu_k(x) \in [0, 1]$.

The "sum-to-one" condition, treated as de facto *probabilistic constraint*, is discussed on the above pages. See also [RBK].

Hard 2-partition

Note the distinction between *set-theoretic* and *functional-theoretic* approaches. In fact, u_{jk} is short for $u_k(x_j)$.

In terms of their function-theoretic duals, $\mathbb{0} \leftrightarrow \varnothing$ and $\mathbb{1} \leftrightarrow X$, properties (4.7) are equivalent to

$$u_A \vee \tilde{u}_A = \mathbb{1} \tag{4.8a}$$

$$u_A \wedge \tilde{u}_A = \mathbb{0} \tag{4.8b}$$

$$\mathbb{0} < u_A < \mathbb{1} \tag{4.8c}$$

Figure: [Bez, p. 18].

Fuzzy 2-partition

(D4.1) *Fuzzy 2-Partition.* Let X be any set, and $P_f(F)$ be the set of all fuzzy subsets of X. The pair $(u_A, \tilde{u}_A)$ is a *fuzzy 2-partition* of X if

$$u_A + \tilde{u}_A = \mathbb{1} \tag{4.11a}$$

$$\mathbb{0} < u_A < \mathbb{1} \tag{4.11b}$$

Figure: [Bez, p. 20]

Fuzzy clustering - finding good $c-$partitions

Clustering: partitioning data set X into c clusters that contain observations **similar** to each other and dissimilar to the rest of the data,

Fuzzy clustering: uses a soft assignment of each observation to each cluster (**a membership degree** u_{jk}) that is grounded in fuzzy set theory.

Fuzzy c-partition space[1]

Let X be any finite set, c a number of clusters $2 \leq c < N$, W_{Nc} a set of real matrices of $N \times c$ dimension. Then a **fuzzy** $c-$**partition space** for X is the set

$$M_{fc} = \left\{ U \in W_{Nc} \mid u_{jk} \in [0,1]; \quad \sum_{k=1}^{c} u_{jk} = 1 \, \forall j; \quad 0 < \sum_{j=1}^{N} u_{jk} < n \, \forall k \right\} \tag{1}$$

[1] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Springer US

Fuzzy clustering - finding good $c-$partitions

The classical Fuzzy C-Means [Bez] is based on a following objective function

$$Q_{\mathrm{FCM}}(U, V; X, m) = \sum_{k=1}^{c} \sum_{j=1}^{N} u_{jk}^{m} \cdot d_{jk}^{2}. \tag{2}$$

Let us recall that c denotes a fixed number of clusters. Note that the fuzzifier m is the only hyperparameter of the algorithm, so $\Theta = \{m\}$.

The minimization problem to solve is

$$\arg\min_{U,V} \quad \sum_{k=1}^{c}\sum_{j=1}^{N} u_{jk}^2 \cdot d_{jk}^2 \tag{3a}$$

$$\text{s.t.} \quad \sum_{k=1}^{c} u_{jk} = 1 \quad \forall j = 1, \dots, N, \tag{3b}$$

$$0 < \sum_{j=1}^{N} u_{jk} < N \quad \forall k = 1, \dots, c, \tag{3c}$$

$$u_{jk} \in [0, 1]. \tag{3d}$$

The formulae for optimal $\hat{u}_{jk}$ and $\hat{\mathrm{v}}_k$ are

$$\hat{u}_{jk} = \frac{1}{\sum_{g=1}^{c}(d_{jk}^2/d_{jg}^2)} = e_{jk} \qquad \text{(the data evidence)}, \tag{4a}$$

$$\hat{\mathrm{v}}_k = \frac{\sum_{j=1}^{N} u_{jk}^2 \cdot \mathrm{x}_j}{\sum_{j=1}^{N} u_{jk}^2}. \tag{4b}$$

Why did we call the outcome in Eq. 4a the *data evidence*?

In general, finding optimal $(U^\star, V^\star)$ is intractable and approximation algorithms are often used. A typical optimization procedure for fuzzy clustering is described in [Bez]. It relies on fixing one variable and optimizing the other at a time. Such an iterative procedure is performed until a convergence criterion is met. The formulae for two variables $\hat{U}$ and $\hat{V}$ are obtained by studying first-order necessary conditions for a global minimizer $(U^\star, V^\star)$ of a respective objective function. Note that this minimization procedure yields equations for standalone $\hat{u}_{jk}$ or $\hat{t}_{jk}$ variables (see [Bez] and [KK] for details).

The generic algorithm can be summarized in four steps:

1. Initiate matrix $U^{(0)}$ e.g. by random sampling. Set the counter $l = 1$.
2. Calculate prototypes $V^{(l)}$ using the formula for $\hat{\mathrm{v}}_k$ and values from $U^{(l-1)}$.
3. Update matrix $U^{(l)}$ using the formula for $\hat{u}_{jk}$ and values from $V^{(l)}$.
4. Compare $U^{(l)}$ to $U^{(l-1)}$ in a chosen matrix norm and stop if the difference is less than a chosen convergence criterion. Otherwise, increase the counter l by 1 and go back to step S2.

Possibilistic C-Means

$$Q_{\mathrm{PCM}}(T, V; X, \Theta) = \sum_{k=1}^{c} \sum_{j=1}^{N} t_{jk}^{m} d_{jk}^{2} + \sum_{k=1}^{c} \gamma_k \sum_{j=1}^{N} (1 - t_{jk})^{m}. \tag{5}$$

$T = [t_{jk}]$ is a typicalities matrix. Q_{PCM} is parametrized by $\Theta = \{m, \Gamma\}$. Vector $\Gamma = (\gamma_1, \ldots, \gamma_c)^T$ contains cluster-specific scalars $\gamma_k > 0$.

The minimization problem becomes

$$
\begin{aligned}
\underset{T,V}{\arg\min} \quad & Q_{\mathrm{PCM}}(T, V; X, \Gamma) && (6a)\\
\text{s.t.} \quad & 0 < \sum_{j=1}^{N} t_{jk} < N \quad \forall k = 1, \ldots, c, && (6b)\\
& t_{jk} \in [0, 1]. && (6c)
\end{aligned}
$$

Krishnapuram and Keller [KK] prove that the optimal solution of the minimization problem in (6) is

$$\hat{t}_{jk} = \frac{1}{1 + \left(d_{jk}^2/\gamma_k\right)} = \frac{\gamma_k}{\gamma_k + d_{jk}^2}, \tag{7}$$

and the optimal value for kth cluster's prototype is

$$\hat{\mathrm{v}}_k = \frac{\sum_{j=1}^{N} t_{jk}^2 \cdot \mathrm{x}_j}{\sum_{j=1}^{N} t_{jk}^2}. \tag{8}$$

- How does PCM differs from FCM in terms of *data evidence*?
- How to set and what is the meaning of γ_k? Read R. Krishnapuram and J.M. Keller. A possibilistic approach to clustering.
  1(2):98–110
- Why "Nothing about PCM is possibilistic in the true sense of possibility theory" [RBK, Sec. 4]? What is the one-line summary of the key aspect of the possibility theory?

Exercices

Ex. 1 [0-2 pkt.]

Recreate dataset from Figure 1a in *R. Krishnapuram and J.M. Keller. A possibilistic approach to clustering*. We will refer to it as to `diamonds`

Apply Fuzzy C-Means and Possibilistic C-Means to reproduce the results of the authors and confirm their conclusions.
Visualize the distribution of memberships with appropriate visualization techniques.

Ex. 2 [0-2 pkt.]

One can choose different distances than Euclidean distance. In particular, we can use the Mahalanobis distance to avoid spherical clusters produced by the algorithms using Euclidean distance.

- by what name goes the appropriate fuzzy clustering model? (Last names of the authors of the paper),
- use FCM either or PCM with Euclidean and Mahalanobis distances (so 2 models in tota: FCM-Euclid & FCM-Mah, or PCM-Euclid and PCM-Mah) to experiment with the `diamonds` dataset. Do the conclusions change?

Ex. 3 [0-1 pkt.]

Choose one option from the list below and compare the appropriate model with the previously fitted models on the dataset `diamonds`:

- yet another distance: kernelized methods,
- yet another fuzzy model: a hybrid of PCM/FCM, evidential clustering.

Semi-Supervised Fuzzy C-Means

Fuzzy clustering - finding good $c-$partitions

Clustering: partitioning data set X into c clusters that contain observations **similar** to each other and dissimilar to the rest of the data,

Fuzzy clustering: uses a soft assignment of each observation to each cluster (**a membership degree** u_{jk}) that is grounded in fuzzy set theory.

Fuzzy c-partition space[2]

Let X be any finite set, c a number of clusters $2 \leq c < N$, W_{Nc} a set of real matrices of $N \times c$ dimension. Then a **fuzzy** $c-$**partition space** for X is the set

$$M_{fc} = \left\{ U \in W_{Nc} \mid u_{jk} \in [0,1]; \quad \sum_{k=1}^{c} u_{jk} = 1 \, \forall j; \quad 0 < \sum_{j=1}^{N} u_{jk} < n \, \forall k \right\} \tag{9}$$

[1] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Springer US

An illustrative example of a fuzzy 2-partition

$X = \{x_1, x_2, x_3\}$, $x_j \in R^p$.

$j = 1, \ldots, 3$; $N = 3$.

$k \in \{1, 2\}$; $c = 2$.

A possible fuzzy 2−partition:

$$U = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \end{array} \begin{array}{cc} k=1 & k=2 \\ \end{array} \begin{pmatrix} 0.98 & 0.02 \\ 0.6 & 0.4 \\ 0.06 & 0.94 \end{pmatrix}$$

Observation x_1 belongs strongly to cluster 1, observation x_3 belongs strongly to cluster 2, while observation x_2 seems to be a "hybrid": it belongs to both clusters to similar degree.

Struggling with imagining a “hybrid”?

A classical example from [Bez]:

- x_1: a `peach`,
- x_3: a `plum`,
- x_2: a `nectarine`, **supposedly** a hybrid of a `peach` and a `plum`.

Supposedly...

Struggling with imagining a “hybrid”?

A classical example from [Bez]:

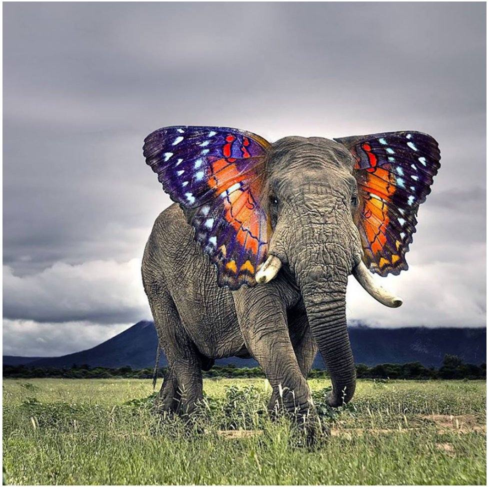
- x_1: a `peach`,
- x_3: a `plum`,
- x_2: a `nectarine`, **supposedly** a hybrid of a `peach` and a `plum`.

Supposedly... because it turns out to be a controversial topic, e.g. `http://www.bctreefruits.com/fruits/other-fruits/detail/0/Nectarines/` state “There is some misconception that nectarines are a cross between a peach and a plum, **but this is not the case. They’re simply a fuzzless peach**.”

Unreal, but proper hybrid

- x_1: a butterfly,
- x_3: an elephant,
- x_2: a butterphant

Unreal, but proper hybrid



- x_1: a butterfly,
- x_3: an elephant,
- x_2: a butterphant

Figure: A butterphant. Source: https://www.boredpanda.com/animals-hybrids-photoshop/?media_id=321587

Introducing partial supervision

Partial supervision (*a type* of semi-supervision): only M observations out of all available N data ($M < N$) are labeled, the rest remains unsupervised.

indices

- index j denotes all available observations, i.e. $j = 1, \ldots, N$,
- index i denotes all supervised observations, i.e. $i = 1, \ldots M$; $\quad M < N$.

Semi-supervised fuzzy clustering

- Semi-Supervised Learning (SSL)[3]: labels $y_j \in Y$ are available for a part of observations M out of all N observations ($M < N$),
- an arbitrary 1-1 mapping must be established between clusters (columns of U) and classes (columns of F).

$$U = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \end{array} \begin{array}{c} \begin{array}{cc} k=1 & k=2 \end{array} \\ \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \end{bmatrix} \end{array} \qquad F = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \end{array} \begin{array}{c} \begin{array}{cc} k=1 & k=2 \end{array} \\ \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \end{array} \begin{array}{c} \mathrm{s(i)} \\ s(1)=1 \\ \\ s(3)=2 \end{array}$$

Function $s(i)$ retrieves the index of the class (a column in F) associated with i-th supervised observation.

[2]Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien, editors. *Semi-Supervised Learning*. Adaptive Computation and Machine Learning. MIT Press

Semi-Supervised Fuzzy C-Means (SSFCMeans) model

Objective function J based on [PW][4] introducing *partial supervision*

$$J_{\mathrm{SSFCM}} = \sum_{k=1}^{c} \sum_{j=1}^{N} u_{jk}^2 \cdot d^2(x_j, v_k) + \alpha \sum_{k=1}^{c} \sum_{j=1}^{N} \underbrace{(u_{jk} - b_j f_{jk})^2}_{\text{penalization}} \cdot d^2(x_j, v_k).$$

- $u_{jk} \in [0, 1]$ is a membership degree
- $d_{jk} = d(x_j, v_k)$ is a Euclidean distance between jth observation and kth prototype v_k (k-th cluster is associated with its prototype $v_k \in R^p$),

---

- $F = [f_{jk}]$ is a matrix introducing partial supervision with binary entries $f_{jk} \in \{0, 1\}$,
- $b_j \in \{0, 1\}$ is an indicator variable equal to 1 iff x_j is labeled,
- $\alpha \geq 0$ **is a scaling factor that weighs the strength of partial supervision**.

[3] W. Pedrycz and J. Waletzky. Fuzzy clustering with partial supervision.
27(5):787–795

Finding optimal c-partitions

Notation:

- $X = [x_j],\ x_j \in R^p$
- $U \in M_{fc}$: a memberships matrix,
- $V \in W_{cp}$: a prototypes matrix ($V = [v_k]$),
- Θ: a set of hyper-parameters.

Task:

$$(U^\star, V^\star) = \underset{U,V}{\arg\min}\ \ J(U, V; X, \Theta), \tag{10}$$

where objective function J quantifies a notion of similarity between observations and prototypes (*typically, using a distance function such as e.g. Euclidean distance*).

Optimal $\hat{U}$

An iterative optimization algorithm is frequently performed. Optimal $\hat{U} = [\hat{u}_{jk}]$ matrix is obtained by considering first-order necessary conditions of a global minimizer, leading to

$$\hat{u}_{jk} = \frac{1}{1+\alpha} \cdot \left(\frac{1 + \alpha \cdot \left(1 - b_j \sum_{s=1}^{c} f_{js}\right)}{\sum_{s=1}^{c} \left(d_{jk}^2 / d_{js}^2\right)} + \alpha f_{jk} b_j \right). \tag{11}$$

In a case of a supervised observation i and its membership degree to the supervised cluster $s(i)$

$$\hat{u}_{i,s(i)} = \frac{1}{1+\alpha} \cdot \frac{1}{\sum_{s=1}^{c} \left(d_{ik}^2 / d_{is}^2\right)} + \frac{\alpha}{1+\alpha}. \tag{12}$$

Interpretations of the scaling factor α

objective function	$\sum_{k=1}^{c}\sum_{j=1}^{N} u_{jk}^2 d_{jk}^2 + \boldsymbol{\alpha} \sum_{k=1}^{c}\sum_{j=1}^{N} \underbrace{\left(u_{jk} - b_j f_{jk}\right)^2}_{\text{penalization}} d_{jk}^2.$
optimal membership $\hat{u}_{i,s(i)}$	$\frac{1}{1+\alpha} \cdot \frac{1}{\sum_{s=1}^{c}\left(d_{ik}^2/d_{is}^2\right)} + \frac{\alpha}{\underbrace{1+\alpha}_{\text{ALB}}}$

- [PW, p. 788] "a scaling factor whose role is **to maintain a balance** between the supervised and unsupervised component",
- "The scaling factor α quantifies the impact of partial supervision as $\text{IPS}(\alpha) = \frac{\alpha}{1+\alpha}$, and establishes an Absolute Lower Bound for a membership of a supervised observation to the supervised cluster $u_{i,s(i)} > \text{IPS}(\alpha)$"[5].

[4] K. Kmita, K. Kaczmarek-Majer, O. Hryniewicz, Explainable Impact of Partial Supervision in Semi-Supervised Fuzzy Clustering, *manuscript under review*

What about the prototypes?

Optimizing $J_{\mathrm{SSFCM}}(V)$ shall raise

$$\mathrm{v}_k = \frac{\sum_{j=1}^{N} \left(u_{jk}^2 + b_j \cdot \alpha \cdot (u_{jk} - f_{jk})^2 \right) \cdot \mathrm{x}_j}{\sum_{j=1}^{N} \left(u_{jk}^2 + b_j \cdot \alpha \cdot (u_{jk} - f_{jk})^2 \right)}, \tag{13}$$

but in the literature frequently the non-α-impacted prototypes are used:

$$\hat{\mathrm{v}}_k = \frac{\sum_{j=1}^{N} t_{jk}^2 \cdot \mathrm{x}_j}{\sum_{j=1}^{N} t_{jk}^2}. \tag{14}$$

Ex. 1 [0-2 pkt.]

Recreate data (and figure) from Fig. 2 in the followign publication: Violaine Antoine, Jose A. Guerrero, and Gerardo Romero. Possibilistic fuzzy c-means with partial supervision. 449:162–186.

Note the authors open-sourced the computational programs to recreate this data.

[AGR, p. 172]. describe their idea to apply partial superivsion to the dataset. Reconstruct their idea, i.e., enhance your dataset with partial supervision.

Ex. 2 [0-2 pkt.]

Run respective SSFCM model from `ssfclust` library with these settings:

- $\alpha = 1$,
- impact of partial superivsion increased twice,
- impact of partial supervision decreased twice.

Plot the results of the respective models, similarly as in the Figure 3a in [AGR].

Ex. 3 [0-1 pkt.]

Answer the questions:

- what distance function is used in `ssfclust`?
- what formula for prototypes is used in the library (the non-α FCM-like, or the α-corrected one)?

Ex. 4 [0-1 pkt.]

Provide an idea to include Mahalanobis distance in `ssfclust`. Refer to [PW] for the formulae to include Mahalanobis distance.

Bibliography I

[AGR] Violaine Antoine, Jose A. Guerrero, and Gerardo Romero.
Possibilistic fuzzy c-means with partial supervision.
449:162–186.

[Bez] James C. Bezdek.
Pattern Recognition with Fuzzy Objective Function Algorithms.
Springer US.

[CSZ] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien, editors.
Semi-Supervised Learning.
Adaptive Computation and Machine Learning. MIT Press.

[KK] R. Krishnapuram and J.M. Keller.
A possibilistic approach to clustering.
1(2):98–110.

Bibliography II

[PW] W. Pedrycz and J. Waletzky.
Fuzzy clustering with partial supervision.
27(5):787–795.

[RBK] Enrique H. Ruspini, James C. Bezdek, and James M. Keller.
Fuzzy Clustering: A Historical Perspective.
14(1):45–55.