

# Clustering Multi-Vues en utilisant la Représentation des Coordonnées Barycentriques

Xiaotong Qian\*, Lili Jin\*\*, Guénaël Cabanes\*\*,  
Parisa Rastin\*\*\*, Nistor Grozavu\*

\*CY Cergy Paris Université, Cergy, France  
xiaotong.qian@ensea.fr  
nistor.grozavu@cyu.fr

\*\*Université Sorbonne Paris Nord, Villetaneuse, France  
lili.jin@edu.univ-paris13.fr  
guenael.cabanes@lipn.univ-paris13.fr

\*\*\*Université de lorraine, Vandoeuvre-lès-Nancy, France  
parisa.rastin@loria.fr

**Résumé.** Dans cet article, nous abordons le problème du clustering multi-vues, où les attributs sont décomposés en groupes qui fournissent des informations complémentaires. Nous présentons une nouvelle approche de clustering multi-vues BCMvlearn qui combine la représentation par coordonnées barycentriques (CB), utilisée dans de précédents travaux de clustering, avec le clustering multi-vues RMKMC basé sur KMeans, qui permet une mise à jour automatique des poids des vues. Cette approche réduit la complexité sans sacrifier la qualité du clustering. En outre, notre algorithme ne dépend pas de la forme vectorielle des données originales, ce qui le rend applicable au clustering multimodal.

## 1 Introduction

Dans le domaine de l'apprentissage non supervisé, le clustering consiste à diviser les objets en groupes homogènes en fonction de leurs similarités, sans connaissance préalable de l'affectation des groupes. Au lieu de ne traiter qu'un sur un seul aspect de la représentation des données, le clustering multi-vues implique le regroupement d'objets présente par plusieurs ensembles de caractéristiques ou de descriptions. Par exemple, des articles de presse dans différentes langues peuvent être considérés comme des vues distinctes. L'analyse multi-vues permet d'exploiter des informations complémentaires pour une meilleure compréhension de la diversité des données (Luo et al., 2007). Malgré des progrès récents en clustering multi-vues, de nombreux algorithmes rencontrent des problèmes de passage à l'échelle avec des données volumineuses ou un grand nombre de vues.

Inspirés par le succès d'approches de clustering se basant sur des coordonnées barycentriques pour des ensembles de données à vue unique (Rastin et al., 2019), l'approche présentée dans cet article étend cette idée au clustering multi-vues. Les coordonnées barycentriques simplifient la représentation des objets en utilisant des moyennes pondérées basées sur les

distances entre les objets, réduisant ainsi la complexité de calcul. Nous adaptons également le cadre du RMKMC (Cai et al., 2013) pour mettre automatiquement à jour l'importance de chaque vue. Notre approche prend aussi en charge le clustering multimodal sans nécessiter une représentation vectoriel pour l'ensemble de données d'origine, car elle se base uniquement sur les distances entre les objets. De plus, elle tient compte de multiples mesures de similarité entre différentes vues, élargissant son applicabilité à divers types de données et modalités. Ce travail a été publiée dans la conférence IJCNN 2023 (Xiaotong et al., 2023).

Dans la suite de cet article, nous introduisons le formalisme des coordonnées barycentriques et présentons un bref aperçu du clustering multivues dans la section 2. Nous présentons ensuite l'approche proposée, BCmvlern, dans la section 3. Nous détaillons et discutons les résultats expérimentaux et l'application au clustering multimodal dans la section 4, avant de conclure dans la section 5.

## 2 État de l'Art

### 2.1 Coordonnées Barycentriques (CB)

Les coordonnées barycentriques d'un point  $x^i$  dans un simplexe (comme un triangle ou un tétraèdre) sont un ensemble de valeurs  $(\beta_1^i, \dots, \beta_p^i)$  qui décrivent la position d'un point par rapport aux sommets du simplexe  $(s^1, \dots, s^p)$ . Un espace de dimension  $p - 1$  peut être défini en sélectionnant  $p$  points de support d'un ensemble de données  $X$ . Dans cet espace, chaque élément de  $X$  est défini par l'équation 1.

$$x^i = \sum_{j=1}^p \beta_j^i s^j \text{ où } \sum_{j=1}^p \beta_j^i = 1. \quad (1)$$

Pour obtenir les coordonnées  $(\beta_1^i, \dots, \beta_p^i)$  d'un objet  $x^i$ ,  $s^p$  n'est pas essentiel, il suffit de définir la matrice  $A$  de taille  $p \times p$  (voir eq. 2), où  $d$  est une mesure de dissimilarité qui doit posséder des propriétés similaires à celles d'une distance euclidienne, et un vecteur  $M^i = [d(x^i, s^1) - d(x^i, s^2), \dots, d(x^i, s^1) - d(x^i, s^p), 1]$  pour enfin obtenir les coordonnées :  $\beta^i = A^{-1} * M^i$ .

$$A = \begin{pmatrix} d(s^1, s^1) - d(s^2, s^1) & \dots & d(s^1, s^p) - d(s^2, s^p) \\ \dots & \dots & \dots \\ d(s^1, s^1) - d(s^p, s^1) & \dots & d(s^1, s^p) - d(s^p, s^p) \\ 1 & \dots & 1 \end{pmatrix}. \quad (2)$$

Basé sur ce formalisme, un algorithme relationnel de clustering K-means utilisant la représentation CB a été proposé par Rastin et al. (2019). L'équation permettant de calculer la distance entre un objet  $x^i$  et un centre de cluster  $\mu^k$  en utilisant leurs coordonnées respectives est la suivante :

$$d^2(x^i, \mu^k) = -\frac{1}{2}(\beta^i - \beta^k)^T * D_s * (\beta^i - \beta^k), \quad (3)$$

où  $D_s$  est la matrice de dissimilarité entre les points de support. Les mises à jour des centres de K-means sont calculées comme la moyenne des CBs des membres du cluster. Les résultats présentés dans Rastin et al. (2019) sur des ensembles de données à vue unique ont montré que cet algorithme a une complexité linéaire en temps et en mémoire, sans perte significative de qualité.

## 2.2 Clustering Multi-Vues

Divers algorithmes de clustering multi-vues, tels que mvKMeans et mvSphKMeans (Bickel et Scheffer, 2004), étendent KMeans pour gérer deux vues conditionnellement indépendantes. Ils fonctionnent en effectuant de manière itérative les étapes de maximisation et d'espérance d'un algorithme EM traditionnel sur une des vues, puis en utilisant les variables cachées calculées comme entrée pour l'étape de maximisation de l'autre vue. Cependant, ils sont limités à seulement deux vues. D'autres algorithmes, tels que mvSpecClus (Kumar et Daumé, 2011) et mvCRSpecClus (Kumar et al., 2011), étendent Spectral clustering en appliquant un co-entraînement et une co-régularisation pour assurer la cohérence des relations entre les vues. Néanmoins, les deux méthodes reposent sur le clustering spectral et présentent une complexité significative en temps de calcul.

De plus, toutes ces approches négligent l'influence du poids des vues (souvent noté  $\alpha$ ) sur la fonction objectif. D'autres approches, telles que celles décrites dans Pedrycz (2002), Benlamine et al. (2019) et Wiswedel et Berthold (2007), considèrent  $\alpha$  comme un paramètre, ce qui permet des applications mieux adaptées aux cas réels où toutes les vues ne sont pas équivalentes.

Dans les méthodes mentionnées précédemment, la valeur de  $\alpha$  est spécifiée par l'utilisateur, mais dans le contexte du clustering multi-vues, ce n'est pas toujours évident. Pour résoudre ce défi, RMKMC (Cai et al., 2013) a été développé. Il met automatiquement à jour les poids  $\alpha$  et est réglé par un paramètre  $\gamma$  pour contrôler la distribution des poids. La nouvelle fonction objectif vise à minimiser la somme pondérée des distances entre les données et les centres de cluster. Grâce à cette adaptation, RMKMC donne des résultats prometteurs par rapport à d'autres méthodes de clustering multi-vues. En combinant RMKMC avec le système de représentation des CB, notre algorithme cherche à améliorer l'efficacité en temps de calcul sans sacrifier la qualité des résultats.

## 3 Approche proposée

Nous proposons une nouvelle approche pour le clustering multi-vues utilisant la représentation des CBs. Dans ce contexte, chaque objet (élément de données) est représenté par une CB tel que défini dans l'équation 1 et les distances entre les centres de cluster et les éléments de données sont calculées avec l'équation 3. La fonction objectif de cette approche est définie comme suit :

$$J = \min_{(\beta^\mu)^{(v)}, G, \alpha^{(v)}} - \frac{1}{2} \sum_{v=1}^V (\alpha^{(v)})^\gamma H^{(v)}, \quad (4)$$

où  $H^{(v)} = \sum_{i=1}^N \Phi_i^T D_s^{(v)} \Phi_i$  avec  $\Phi_i = (\beta^i)^{(v)} - (\beta^\mu)^{(v)} G_i^T$ ,  $\beta^\mu \in R^{p \times k}$  représente les CBs de tous les centres  $\mu$ , et  $N$  représente le nombre de données. Par ailleurs,  $\sum_{v=1}^V \alpha^{(v)} = 1$  et  $\sum_{k=1}^K G_{i,k} = 1$ , où  $G \in R^{N \times K}$  est une matrice de cluster binaire qui indique l'appartenance de chaque élément de données à un cluster spécifique.

Sur la base de l'idée de Cai et al. (2013), l'optimisation de la fonction objective peut être divisée en trois parties.

### 3.1 Optimisation

1. Mise à jour  $(\beta^\mu)^{(v)}$  en calculant la dérivée partielle de  $J$  par rapport à  $(\beta^\mu)^{(v)}$ .

$$(\beta^\mu)^{(v)} = \sum_{i=1}^N (\beta^i)^{(v)} \left( \sum_{n=1}^N G_i^T G_i \right)^{-1} \quad (5)$$

2. Mise à jour  $G$  en calculant la matrice de distance de la somme pondérée de chaque vue.

$$i = \underset{k \in \{1, \dots, K\}}{\operatorname{argmin}} \left\{ -\frac{1}{2} \sum_{v=1}^V (\alpha^{(v)})^\gamma \Psi_k^T D_s^{(v)} \Psi_k \right\} \quad (6)$$

Avec  $\Psi_k = (\beta^i)^{(v)} - (\beta^\mu)_k^{(v)}$ , la différence entre l'élément de données  $n$  et le centre de cluster  $k$  dans la vue  $v$ , de sorte que  $\forall i \in \{1, \dots, N\}$ ,  $G_{i,a} = 1$  et  $G_{i,b} = 0$  où  $b \in \{b \neq a | 1, \dots, k\}$ .

3. En mettant à jour  $\alpha^{(v)}$ , nous utilisons le multiplicateur de Lagrange  $\lambda$  et obtenons :

$$\alpha^{(v)} = \frac{(\gamma H^{(v)})^{\frac{1}{1-\gamma}}}{\sum_{v=1}^V (\gamma H^{(v)})^{\frac{1}{1-\gamma}}} \quad (7)$$

### 3.2 Algorithme

La mise en œuvre de l'approche commence par la projection de chaque donnée dans un espace de CBs, créé à partir d'un ensemble de points de support sélectionnés à partir des données. Ensuite, les CB  $\beta^k$  de chaque centre de cluster  $\mu^k$  sont initialisées aléatoirement. Enfin, la matrice de clusters  $G$  est calculée, attribuant un poids égal à chaque vue. Ces trois composantes sont mises à jour itérativement par optimisation jusqu'à convergence de la fonction objectif. Pour plus de détails, consultez l'algorithme 1.

---

#### Algorithme 1 : BCMvlearn

---

**entrée** : Ensemble des données séparées par vue  $X^{(1)}, \dots, X^{(v)}, P, K$ ,  
Liste des fonctions de mesure de distance par vue  $[d^{(1)}, \dots, d^{(v)}]$

**sortie** : Matrice commune d'affectation des clusters  $G$

Choisir aléatoirement  $p$  points de support

**pour**  $i = 1, \dots, v$  **faire**

- $O^{(i)} \leftarrow$  Matrice de distance  $O^{(i)}$  entre les données et les points de support par  $d^{(i)}$ .
- $D_s^{(i)} \leftarrow$  Extraire la matrice de distance entre les points de support de  $O^{(i)}$ .
- Calculer la nouvelle représentation  $\beta^{(i)}$  de  $X^{(i)}$  dans le système CB (Eq. (3)).

**fin**

Initialiser  $K$  centres  $\mu^{(v)}$  à partir de  $\beta^{(i)}$ , et initialiser  $\alpha^{(i)} = \frac{1}{v}, \forall i \in [1, \dots, v]$ .

Définir la matrice de cluster commune  $G$  par  $\mu^{(v)}$ .

**tant que non convergence faire**

- Mettre à jour les centres de cluster  $\mu^{(i)}$  de chaque vue par (5),  $\forall i \in [1, \dots, v]$ .
- Mettre à jour  $G$  à l'aide de (6).
- Calculer  $H^{(i)}, \forall i \in [1, \dots, v]$ , et mettre à jour  $\alpha^{(i)}$  par (7),  $\forall i \in [1, \dots, v]$ .

**fin**

---

## 4 Expérience

Nous suivons la même configuration que celle décrite dans Cai et al. (2013). Nous utilisons ARI (Hubert et Arabie, 1985), NMI (Vinh et al., 2009), DBI (Davies et Bouldin, 1979), et SI (Rousseeuw, 1987) comme indices internes et externes, ainsi que la complexité en temps de calcul pour l'évaluation. Ces indices sont estimés par une moyenne obtenus par chaque algorithme, calculés sur 20 répétitions pour 10 jeux de données multi-vues réelles. La description des données est présentée dans le tableau 1.

Nom de données	Nb données	Nb cluster	Nb vue	Nb variable par vue
MSRCv1 (Winn et Jojic, 2005)	210	7	5	[24, 576, 512, 256, 254]
SensITVehicle (Chih-Chung, 2011)	300	3	2	[50,50]
Movies (Bisson et Grimal, 2012)	617	17	2	[1878, 1398]
Caltech101-7 (Fei-Fei et al., 2004)	1474	7	6	[48, 40, 254, 1984, 512, 928]
Small-Reuters (Frank, 2010)	1500	6	5	[21531, 24892, 34251, 15506, 11547]
100-leaves (Frank, 2010)	1599	100	3	[64, 64, 64]
Handwritten (Frank, 2010)	2000	10	6	[76, 216, 64, 240, 47, 6]
Small-NUS (Chua et al., 2009)	2000	31	5	[65, 226, 145, 74, 129]
Caltech101-20 (Fei-Fei et al., 2004)	2386	20	6	[48, 40, 254, 1984, 512, 928]
Waveform (Frank, 2010)	5000	3	2	[21, 19]

TAB. 1 – Résumé des données utilisées

### 4.1 Discussion sur les paramètres

Dans notre approche, il existe deux hyperparamètres : le nombre de points de support  $p$ , et  $\gamma$ , qui contrôle le poids de chaque vue. Selon notre étude, on fait varier  $p$  entre 10 et 200 sur toutes les données, et on trouve que l'augmentation de  $p$  n'améliore généralement pas de manière significative la qualité des résultats, mais augmente le temps de calcul. En outre, nous avons testé la valeur de  $\gamma$  entre 0.2 et 2, et avons constaté que lorsque  $\gamma$  varie de 0,9 à 1,1, l'indice externe diminue tandis que l'indice interne augmente. Cependant, il n'y a pas beaucoup de changement dans les autres intervalles. Nous fixons donc pour la suite des expériences  $nb\_SP = 30$  et  $\gamma = 0,5$ .

### 4.2 Performances pour le clustering multi-vue

Dans cette section, nous visons à évaluer la performance globale de BCmlearn. Nous testons aussi une version BCmlearn++ qui utilise KMeans++ (Arthur et Vassilvitskii, 2007) pour améliorer le processus d'initialisation de KMeans en sélectionnant stratégiquement les centres initiaux. Ces approches sont comparées avec des algorithmes de clustering multi-vue tels que décrits dans la section 2.2.

Les expériences montrent que BCmlearn et BCmlearn++ surpassent mvKMeans et mvS-phKMeans en termes de qualité externe moyenne. Ils sont également comparables en performance à RMKMC, mvSpecClus et mvCRSpecClus. Bien que BCmlearn++ montre une légère baisse en termes d'indices externes par rapport à BCmlearn, il obtient de bons résultats en termes de qualité interne moyenne. En résumé, les expériences suggèrent que BCmlearn et BCmlearn++ sont des choix efficaces pour le clustering multi-vue, offrant des performances

## Clustering Multi-Vues en utilisant la Représentation CB

similaires ou supérieures à d'autres méthodes, tout en étant plus flexibles et sans limitation sur le nombre de vues (voir Fig. 1a à Fig. 1c).

De nombreux algorithmes adaptés aux données multi-vues ne fonctionnent que si toutes les vues ont la même représentation (en particulier vectorielle). Nous souhaitons mettre en avant la capacité de notre approche à traiter des vues de différentes modalités, à condition qu'une distance puisse être définie pour chaque vue. Par exemple, nous avons testé l'approche sur l'ensemble de données IEMOCAP (Busso et al., 2008), composé de différents modalités. En employant des mesures de distance appropriées pour chaque modalité (par exemple, Euclidean distance (ED) pour l'audio, CosED (Schubert, 2021) pour le texte, et IMED (Sun et al., 2015) pour les images), nous montrons que l'approche proposée présente de meilleures performances sur les ensembles de données multimodales, mettant en évidence son efficacité avec diverses mesures de distance (voir Fig. 1d).

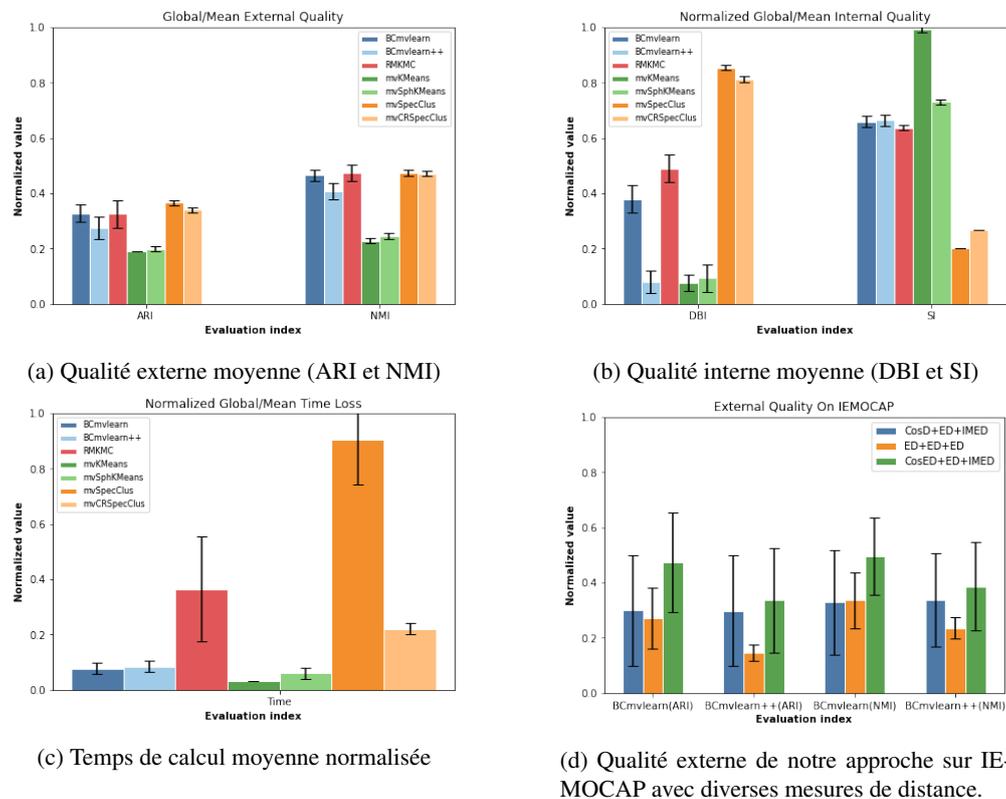


FIG. 1 – Résultats des test de performance pour le clustering multi-vue.

## 5 Conclusion

Cet article présente BCmvlearn, une nouvelle approche de clustering multi-vue basée sur les coordonnées barycentriques. Elle réduit la dimensionnalité des représentations en projetant

les données dans un espace barycentrique, optimise le calcul des prototypes des clusters, et construit un modèle consensus pour le clustering multi-vue. Les résultats montrent sa stabilité, sa convergence rapide, et sa compétitivité en termes de qualité et de temps de calcul. BCmv-learn se distingue par son efficacité pour la découverte de clusters homogènes, tels que définis par des indices de qualité internes et externes. Cette approche s'adapte à diverses métriques de distance et présente un potentiel pour les applications de clustering multimodal. L'objectif futur est d'étendre cette approche au traitement de flux de données dynamiques multi-vues.

## Remerciements

Ce travail a été financé par le projet ANR Pro-TEXT (project N° ANR-18-CE23-0024-01). Plus de détails sont disponibles à l'adresse suivante : <https://pro-text.huma-num.fr/le-projet/>

## Références

- Arthur, D. et S. Vassilvitskii (2007). K-means++ the advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035.
- Benlamine, K., N. Grozavu, Y. Bennani, et B. Matei (2019). Collaborative non-negative matrix factorization. In *International Conference on Artificial Neural Networks*, pp. 655–666. Springer.
- Bickel, S. et T. Scheffer (2004). Multi-view clustering. In *ICDM*, Volume 4, pp. 19–26. Citeseer.
- Bisson, G. et C. Grimal (2012). Co-clustering of multi-view datasets : a parallelizable approach. In *2012 IEEE 12th International Conference on Data Mining*, pp. 828–833. IEEE.
- Busso, C., M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, et S. S. Narayanan (2008). Iemocap : Interactive emotional dyadic motion capture database. *Language resources and evaluation* 42, 335–359.
- Cai, X., F. Nie, et H. Huang (2013). Multi-view k-means clustering on big data. In *Twenty-Third International Joint conference on artificial intelligence*.
- Chih-Chung, C. (2011). Libsvm : a library for support vector machines. *ACM transactions on intelligent systems and technology* 2, 27–1.
- Chua, T.-S., J. Tang, R. Hong, H. Li, Z. Luo, et Y. Zheng (2009). Nus-wide : a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, pp. 1–9.
- Davies, D. L. et D. W. Bouldin (1979). A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence* (2), 224–227.
- Fei-Fei, L., R. Fergus, et P. Perona (2004). Learning generative visual models from few training examples : An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pp. 178–178. IEEE.
- Frank, A. (2010). Uci machine learning repository. irvine, ca : University of california, school of information and computer science. <http://archive.ics.uci.edu/ml>.

- Hubert, L. et P. Arabie (1985). Comparing partitions. *Journal of classification* 2, 193–218.
- Kumar, A. et H. Daumé (2011). A co-training approach for multi-view spectral clustering. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 393–400.
- Kumar, A., P. Rai, et H. Daume (2011). Co-regularized multi-view spectral clustering. *Advances in neural information processing systems* 24.
- Luo, H., F. Kong, et Y. Li (2007). An analysis of diversity measures in clustering ensembles. *CHINESE JOURNAL OF COMPUTERS-CHINESE EDITION-* 30(8), 1315.
- Pedrycz, W. (2002). Collaborative fuzzy clustering. *Pattern Recognition Letters* 23(14), 1675–1686.
- Rastin, P., G. Cabanes, B. Matei, Y. Bennani, et J.-M. Marty (2019). A new sparse representation learning of complex data : Application to dynamic clustering of web navigation. *Pattern Recognition* 91, 291–307.
- 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.
- Schubert, E. (2021). A triangle inequality for cosine similarity. In *International Conference on Similarity Search and Applications*, pp. 32–44. Springer.
- Sun, B., J. Feng, et G. Wang (2015). On the translation-invariance of image distance metric. In *Applied Informatics*, Volume 2, pp. 1–12. SpringerOpen.
- Vinh, N. X., J. Epps, et J. Bailey (2009). Information theoretic measures for clusterings comparison : is a correction for chance necessary ? In *Proceedings of the 26th annual international conference on machine learning*, pp. 1073–1080.
- Winn, J. et N. Jojic (2005). Locus : Learning object classes with unsupervised segmentation. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, Volume 1, pp. 756–763. IEEE.
- Wiswedel, B. et M. R. Berthold (2007). Fuzzy clustering in parallel universes. *International Journal of Approximate Reasoning* 45(3), 439–454.
- Xiaotong, Q., J. Lili, G. CABANES, P. RASTIN, et N. GROZAVU (2023). Multi-view clustering using barycentric coordinate representation. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE.

## Summary

In this article, we address the problem of multi-view clustering, where attributes are decomposed into groups that provide complementary information. We present a new multi-view clustering approach BCMvlearn that combines the barycentric coordinate (BC) representation used in previous clustering work with the KMeans-based multi-view clustering RMKMC, which allows automatic updating of view weights. This approach reduces complexity without sacrificing clustering quality. In addition, our algorithm does not depend on the vector form of the original data, making it applicable to multimodal clustering.