On Modal Clustering with Gaussian Sum-Product Networks

Tiago Madeira[1]   **Denis Deratani Mauá**[1]

[1]Institute of Mathematics and Statistics, University of São Paulo, Brazil

Abstract

Recent research highlights the significance of incorporating density modeling into clustering procedures. While the Sum-Product Networks' ability to compactly represent mixture models has been long noticed, their potential for modal clustering remains largely unexplored. This paper explores the use of Gaussian Sum-Product Networks for semi-parametric density-based clustering via mode association. To associate points to modes, we make use of a recently developed efficient EM-style algorithm. We perform image segmentation experiments to evaluate the (dis)advantages of modal clustering using such models.

1 INTRODUCTION

Clustering plays a pivotal role in data analysis, by which means one can discover segments of homogeneous examples (e.g., similar online purchases), detect outliers and anomalies, fill-in missing values, etc.

Most clustering methods take a distance or dissimilarity function as input, and segment data so as to minimize intra-cluster distance and maximize inter-cluster distance [Aggarwal and Reddy, 2018]. The most prominent example is the k-means algorithm [MacQueen, 1967]. More recently, an increasing number of researchers have emphasized the importance of more explicitly incorporating density modeling into clustering procedures [Carlsson and Mémoli, 2013].

Mixture models form an expressive class of density models, capable of approximating any well-behaved density. A mixture model can be used for clustering either by associating each point to a component, or by associating each point to a mode. The former approach should be favored when one has reason to believe that the true density aligns with the assumed latent-variable model [Chacón, 2019]. The latter is preferred when there is no particular reason to assume a specific parametric model of data. Note that the modes and mixture components can differ significantly in number and location [Améndola et al., 2019].

Sum-Product Networks (SPNs) are a relatively recent class of deep statistical models that leverage arithmetic circuits [Darwiche, 2003] to effectively capture context-sensitive independences and offer reliable and efficient inference, making them a competitive approach for a wide range of demanding machine learning tasks [Poon and Domingos, 2011, Llerena and Maua, 2017, Amer and Todorovic, 2016].

While the connection of SPNs and (hierarchical) mixture models has long been noticed [Peharz et al., 2014, Zhao et al., 2015], the application of SPNs to clustering remains relatively unexplored. This in spite of the fact that the widely-used schema for learning SPNs employs hierarchical clustering to construct the network's structure [Gens and Domingos, 2013, Vergari et al., 2018].

In this paper, we report the first results of our investigation about the effectiveness of modal clustering using Gaussian SPNs. We discuss the advantages and difficulties with such an approach. In particular, we discuss how to employ a recently developed EM-like algorithm for mode finding in Gaussian SPNs [Madeira and Mauá, 2022] to perform modal clustering, and show its application to image segmentation.

2 SUM-PRODUCT NETWORKS

A Sum-Product Network (SPN) is a weighted rooted directed acyclic graph, where each internal is either a sum node or a product node, and each leaf node corresponds is a univariate distribution [Gens and Domingos, 2013]. The edges $u \to v$ from a sum node u are associated with weights $w(u, v) \geq 0$, and the remaining edges have weight 1 (usually omitted). For any node u, we call its scope, denoted as $\mathrm{sc}(u)$, the set of random variables that appear in some distribution at the leaves of the inducing subgraph.

Accepted for the 6th Workshop on Tractable Probabilistic Modeling at UAI (TPM 2023).

An SPN satisfies the properties of *decomposability* and *completeness*, which ensure that certain types of inferences are tractable. Decomposability states that the scopes of any two children of a product node are disjoint (i.e., if u is a product node then $\mathrm{sc}(v) \cup \mathrm{sc}(w) = \emptyset$ for any distinct $v, w \in ch(u)$). Completeness states that the scopes of children of sum nodes are identical (i.e., if u is a sum node then $\mathrm{sc}(v) = \mathrm{sc}(w)$ for any $v, w \in ch(u)$). We also assume w.l.o.g. that the sum of the weights of edges from a sum node u is 1, that is, $\sum_{v \in \mathrm{ch}(u)} w(u, v) = 1$. Together, those three properties ensure that the SPN rooted at any node specifies a joint distribution over its scope. This allow us to refer to nodes and their distribution functions interchangeably.

Given an SPN $\mathcal{S}$, we denote the value of its distribution at a configuration $\mathbf{x}$ of its scope $\mathbf{X}$ by $\mathcal{S}(\mathbf{x})$. This value is obtained inductively as follows. The value of a leaf node is the value of the corresponding distribution. The value of an internal model is $\oplus_{v \in ch(u)} w(u, v)\mathcal{S}_v(\mathbf{x})$, where $\oplus$ is the operation of node u (sum or product) and $\mathcal{S}_v$ is the SPN rooted at v.

Consider a subgraph $\mathcal{T}$ of an SPN $\mathcal{S}$. We say that $\mathcal{T}$ is an *induced tree* if it can be constructed inductively, starting from the root of $\mathcal{S}$, and then including all children of product nodes and exactly one child of any sum node. An induced tree $\mathcal{T}$ is a tree-shaped SPN (i.e., satisfies decomposability and completeness) whose distribution is given by $\mathcal{T}(\mathbf{x}) = \prod_{u \to v \in \mathcal{T}} w(u, v) \prod_j T_j(\mathbf{x})$, where T_j is the SPN/distribution of the leaf of $\mathcal{T}$ whose scope is X_j (note: $\mathcal{T}$ has exactly n such leaves/variables).

Assume an ordering $\mathcal{T}^1, \ldots, \mathcal{T}^\tau$ of the induced trees of $\mathcal{S}$, and let $w_i = \prod_{u \to v \in \mathcal{T}^i} w(u, v)$ be the product of the weights in the i-th induced tree. Then, $\mathcal{S}(\mathbf{x}) = \sum_{i=1}^{\tau} w_i T^i(\mathbf{x})$ for any configuration $\mathbf{x}$ of the scope, where $T^i(\mathbf{x}) = \prod_{j=1}^{n} T_j^i(\mathbf{x})$ is the product of the values of the univariate distributions at the leaves for $X_j = x_j$. Thus, induced trees provide an interesting representation of SPNs as a finite mixture model [Zhao et al., 2016], one where each induced tree represents a different component.

In this study, our emphasis lies on SPNs with Gaussian distributions assigned to their leaves, known as Gaussian SPNs (or GSPNs, for short). For such models, the resulting induced tree representation corresponds to a Gaussian Mixture Model (GMM), one where the variables in each component $\mathcal{T}^i$ are uncorrelated and have mean σ_{k_i} and variance $\sigma^2_{k_i}$.

3 MODAL CLUSTERING IN GSPNS

In modal clustering, data points are segmented by associating each point to a mode of a density model, usually by means of some hill-climbing strategy. Hence, to employ Gaussian SPNs for such intent one needs a method that identifies (some of) its modes.

3.1 FINDING MODES

Li et al. [2007] developed the *Modal EM*, an EM-style method that finds a mode of a GMM $p(\mathbf{x}) = \sum_k^\tau w_k p^k(\mathbf{x})$ by hill-climbing from a given starting point $\mathbf{x}^{(0)}$. The method alternates the following two steps until convergence:

E-step: Let $q_k = \frac{w_k p^k(\mathbf{x}^{(r)})}{p(\mathbf{x}^{(r)})}$, for $k = 1, \cdots, \tau$.

M-step: Compute $\mathbf{x}^{(r+1)} = \arg\max_{\mathbf{x}} \sum_k^\tau q_k \log p^k(\mathbf{x})$.

The direct application of the above algorithm to SPNs is intractable due to the high number of components (given by its number of induced trees). To overcome that limitation, Madeira and Mauá [2022] adapted Modal EM to exploit the recursive character of GSPNs. Their algorithm repeatedly obtains a configuration $\mathbf{x}^{(r+1)}$ from a configuration $\mathbf{x}^{(r)}$ by:

$$x_i^{(r+1)} = \frac{\sum_k^\tau \frac{\mu_{k_i}}{\sigma^2_{k_i}} w_k T^k\left(\mathbf{x}^{(r)}\right)}{\sum_k^\tau \frac{1}{\sigma^2_{k_i}} w_k T^k\left(\mathbf{x}^{(r)}\right)} . \tag{1}$$

The equation above is computed by traversing the network from the leaves to the root, taking time $\Theta(n|\mathcal{S}|)$ per iteration, where n is the number of random variables and $|\mathcal{S}|$ is the number of nodes in the SPN.

Notably, the updates in Equation 1 coincide with the fixed-point iterative scheme proposed by Carreira-Perpiñán [2000] for finding modes of GMMs. In turn, this iterative scheme can be seen as a generalized version of the famous Mean-Shift algorithm Chacón [2019].

3.2 ON THE NUMBER OF CLUSTERS

In modal clustering, the number of clusters corresponds to the number of modes in the model (and not to the number of components). A precise characterization of the relation of number of modes $m(d, k)$ as a function of the number of components k and dimensionality d is still an open problem. Améndola et al. [2019] showed that $m(d, k) \geq \binom{k}{d} + k$ for $d, k \geq 2$,. Additionally, they established that the number of non-degenerate stationary points of a d-dimensional GMM with k components is bounded above by $2^{d+\binom{k}{2}}(5 + 3d)^k$. These bounds suggest that the number of modes can vary greatly inside a class of mixture models of same complexity.

We note that in SPNs the number of components (k) is not usually specified and is rather induced from data.

4 IMAGE SEGMENTATION

As a preliminary investigation of the effectiveness of modal clustering with GPSNs, we performed some experiments with image segmentation of two images, shown in Figure 1.

(a) (b)

Figure 1: (a) Easter Bunny. (b) Tarsila do Amaral's The Family.

Table 1: SPNs Learned for Segmentation (Easter Bunny).

Parameter s	**Nodes**	**Height**	**Clusters**
20,000	13	3	6
15,000	19	3	7
10,000	25	3	10
5,000	50	5	31
2,000	132	5	68
500	528	7	398
200	1,267	9	676

We obtained datasets of 5 variables by considering the RGB intensity values and x and y locations of each pixel in the image. This resulted in $200 \times 144 = 28,800$ data points (instances) for Easter Bunny and $200 \times 158 = 31,600$ for The Family. The datasets were then used to learn GSPNs from data, using the LearnSPN implementation provided by the SPFlow library.[1] Instance splitting was performed using GMM clustering, while variable splitting was accomplished using the Randomized Dependence Coefficient [Lopez-Paz et al., 2013]. We compared our implementation of Modal EM clustering with k-means clustering, as implemented by the scikit-learn library.[2]

We experimented with different GSPNs by varying the minimum number of instances required for slicing in the learning process (s). Tables 1 and 2 show the number of nodes, network height and the number of clusters (modes) obtained for each GSPN as we vary s. One sees the great dependence between those quantities, as well as the quick increase in the number of modes.

The images were segmented by coloring each pixel according to the average color of the pixels in its same cluster, considering a cluster to be formed by points which converge to the same mode. Figures 2 and 3 display a visual comparison of image segmentation by GSPNs and by the k-means algorithm, where k is set to the number of clusters identified by Modal EM in GSPNs trained with different hyperparameters. One can note that GSPN seemingly delivers a worse segmentation than k-means, while taking much more compute. We conjecture that GSPN's worse performance is

Table 2: SPNs Learned for Segmentation (The Family).

Parameter s	**Nodes**	**Height**	**Clusters**
20,000	13	3	4
15,000	25	3	5
10,000	31	3	8
5,000	61	3	19
2,000	163	3	43
500	603	7	187
200	1,504	7	555

GSPN **k-means**

$s = 20000$
$k = 6$

$s = 10000$
$k = 10$

$s = 2000$
$k = 68$

$s = 500$
$k = 398$

Figure 2: Image Segmentation of Easter Bunny.

due to a lack of fit to the model, which could be mitigated by changing the structure learning algorithm, performing fine-tuning of parameters or even using k-means solution as a initial model for refinements.

To conclude, we note that GSPN modal clustering can deliver inferences that standard clustering techniques such as k-means cannot, such as coping with missing values, detecting outliers by probability threshold and scaling up easily to more complex and high-dimensional domains.

5 CONCLUSIONS

In this paper, we have presented an initial exploration of modal clustering using SPNs. Our experiments have highlighted the significant influence of the hyperparameters used to learn the SPN on the resulting number of modes in the model, underscoring the crucial role of parameter selection in achieving desired clustering outcomes.

Furthermore, our findings have demonstrated the applicability of clustering techniques for SPN model analysis. The number of modes in a density distribution serves as an indicator of the complexity of the underlying model, providing valuable insights into its representation capabilities.

[1] Available at `github.com/SPFlow/SPFlow`.
[2] Available at `scikit-learn.org`.

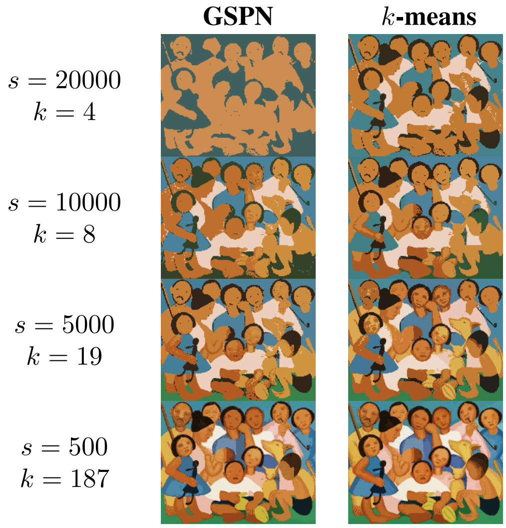

Figure 3: Image Segmentation of The Family.

While image segmentation may not be the optimal application domain for SPNs, our results have shown satisfactory performance comparable to the widely-used k-means algorithm. However, visually, we observed that image segmentation using k-means with k equivalent to the number of modes in the SPN yields more detailed segmentation results.

We acknowledge that there is ample room for future work, particularly in exploring the application of modal clustering in various domains beyond image segmentation. These directions can contribute to expanding the understanding and utility of SPNs in clustering tasks across diverse fields.

Acknowledgements

This work was partially financed by CNPq grant #305136/2022-4, CAPES Finance Code 001 and FAPESP grant #2022/02937-9.

References

Charu C. Aggarwal and Chandan K. Reddy. *Data Clustering: Algorithms and Applications*. Taylor & Francis, 2018.

Mohamed R. Amer and Sinisa Todorovic. Sum product networks for activity recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38:800–813, 4 2016.

Carlos Améndola, Alexander Engström, and Christian Haase. Maximum number of modes of gaussian mixtures. *Information and Inference: A Journal of the IMA*, 2 2019.

Gunnar Carlsson and Facundo Mémoli. Classifying clustering schemes. *Foundations of Computational Mathematics*, 13:221–252, 4 2013. ISSN 1615-3375. doi: 10.1007/s10208-012-9141-9.

Miguel Á Carreira-Perpiñán. Mode-finding for mixtures of gaussian distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1318–1323, 2000.

José E. Chacón. Mixture model modal clustering. *Advances in Data Analysis and Classification*, 13:379–404, 6 2019. ISSN 1862-5347. doi: 10.1007/s11634-018-0308-3.

Adnan Darwiche. A differential approach to inference in bayesian networks. *Journal of the ACM*, 50:280–305, 5 2003.

Robert Gens and Pedro Domingos. Learning the structure of sum-product networks. pages 873–880. PMLR, 2013.

Jia Li, Surajit Ray, and Bruce G Lindsay. A nonparametric statistical approach to clustering via mode identification. *Journal of Machine Learning Research*, 8:1687–1723, 2007.

Julissa Villanueva Llerena and Denis Deratani Maua. On using sum-product networks for multi-label classification. pages 25–30. IEEE, 10 2017. ISBN 978-1-5386-2407-4. doi: 10.1109/BRACIS.2017.34.

David Lopez-Paz, Philipp Hennig, and Bernhard Schölkopf. The randomized dependence coefficient. pages 1–9. Curran Associates Inc., 2013.

J B MacQueen. Some methods for classification and analysis of multivariate observations. volume 1, pages 281–297. University of California Press, 1967.

Tiago Madeira and Denis Mauá. Tractable mode-finding in sum-product networks with gaussian leaves. pages 497–508. SBC, 2022. doi: 10.5753/eniac.2022.227582.

Robert Peharz, Georg Kapeller, Pejman Mowlaee, and Franz Pernkopf. Modeling speech with sum-product networks: Application to bandwidth extension. pages 3699–3703. IEEE, 5 2014.

Hoifung Poon and Pedro Domingos. Sum-product networks: A new deep architecture. pages 689–690. IEEE, 11 2011.

Antonio Vergari, Robert Peharz, Nicola Di Mauro, Alejandro Molina, Kristian Kersting, and Floriana Esposito. Sum-product autoencoding: Encoding and decoding representations using sum-product networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32, 4 2018. ISSN 2374-3468. doi: 10.1609/aaai.v32i1.11734.

Han Zhao, Mazen Melibari, and Pascal Poupart. On the relationship between sum-product networks and bayesian networks. pages 116–124. JMLR.org, 1 2015.

Han Zhao, Pascal Poupart, and Geoff Gordon. A unified approach for learning the parameters of sum-product networks. pages 433–441. Curran Associates, Inc., 2016.