Generative Kernel Spectral Clustering

David Winant*, Sonny Achten*†, Johan A. K. Suykens ‡

ESAT-Stadius, KU Leuven
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

Abstract. Modern clustering approaches often trade interpretability for performance, particularly in deep learning-based methods. We present Generative Kernel Spectral Clustering (GenKSC), a novel model combining kernel spectral clustering with generative modeling to produce both well-defined clusters and interpretable representations. By augmenting weighted variance maximization with reconstruction and clustering losses, our model creates an explorable latent space where cluster characteristics can be visualized through traversals along cluster directions. Results on MNIST and FashionMNIST datasets demonstrate the model's ability to learn meaningful cluster representations.

1 Introduction

Clustering is a key technique in data analysis, used to uncover patterns in unlabeled data by grouping similar instances. While modern neural network-based clustering methods often achieve impressive performance, they frequently lack interpretability, making it difficult to understand the characteristics that define each cluster. This limitation is especially concerning in sensitive domains—such as healthcare, finance, and security—where understanding the basis of clustering results is critical for transparency, trust, and informed decision-making.

Deep clustering methods often lack interpretability [1], while interpretable methods rarely use deep architectures and rely on post hoc explanations, limiting their ability to capture complex patterns [2]. This gap in the literature shows a need for models that combine the representational power of deep learning with the transparency of interpretable clustering. To address this challenge, we propose GenKSC, a novel interpretable clustering model that combines clustering with generative modeling. GenKSC produces well-defined clusters while allowing users to interpret the distinguishing features of each group. Our approach leverages the latent structure of a kernel spectral clustering (KSC) framework, integrating it with a generative restricted kernel machine. Augmented loss terms further guide the model to form clear, interpretable clusters, ensuring the learned representations are both accurate and explorable. By bridging the gap between

*Equal contribution.

†Corresponding author: sonny.achten@kuleuven.be

‡The research leading to these results has received funding from the European Research Council under the European Union's Horizon 2020 research and innovation program / ERC Advanced Grant E-DUALITY (787960). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information. This research received funding from the Flemish Government (AI Research Program); iBOF/23/064; KU Leuven C1 project C14/24/103. Johan Suykens, David Winant, and Sonny Achten are also affiliated to Leuven.AI - KU Leuven institute for AI, B-3000, Leuven, Belgium.

clustering and interpretability, GenKSC advances explainable AI, offering a valuable tool for applications where *understanding* the clustering result is critical.

2 Preliminaries and Related Work

Restricted Kernel Machines (RKM) [3], introduced conjugate feature duality in a kernel-based setting, facilitating both supervised and unsupervised learning and supporting deep kernel learning. The Stiefel-RKM [4] is a generative model that achieves interpretability through a disentangled latent space within a kernel principal component analysis framework.

Exploring latent spaces has been enabled in other generative models, such as variational autoencoders [5], as well as the combination with a clustering model in ClusterGAN [6] where an adversarial loss was used along with a latent space clustering objective to preserve clustering. However, ClusterGAN does not inherently provide interpretability of the learned clusters.

Alzate and Suykens [7] introduced kernel spectral clustering (KSC)—a nonlinear extension to spectral clustering—by framing it as a weighted principal component analysis (PCA) problem in a (implicit) feature space. The solution to KSC is formulated as an eigendecomposition problem:

$$\boldsymbol{D}^{-1}\boldsymbol{K}\boldsymbol{H} = \boldsymbol{H}\boldsymbol{\Lambda}, \tag{1}$$

where $\boldsymbol{K}$ represents the kernel matrix with $K_{ij} = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}$, and $\mathcal{H}$ denotes the reproducing kernel Hilbert space associated with the kernel. Here, $\boldsymbol{D}$ is the diagonal degree matrix with elements $D_{ii} = \sum_j K_{ij}$, $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n]^\top$ represents the spectral embeddings, and $\boldsymbol{\Lambda}$ is a diagonal matrix containing the corresponding eigenvalues along its diagonal. As illustrated in Fig. 1, distinct linear structures emerge in the $(k-1)$ dimensional eigenspace spanned by the highest principal components when data are clustered into k groups. Given this distinct line structure, cosine similarity is highly suitable for both cluster assignments and cluster quality evaluation [8].

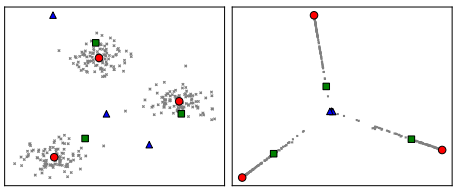

Fig. 1: Visualization of latent structure with KSC [7]. **Left**: The data in its original space. **Right**: The spectral embeddings in the eigenspace of the first two components. Observe that the cluster prototypes align at the tips of the lines. A radial basis function kernel is used in this example.

3 The Generative Kernel Spectral Clustering Model

The proposed model leverages a weighted variance maximization framework, which shares foundational connections with the kernel spectral clustering model (1). We use a parametric feature map that is trained simultaneously with the spectral clustering problem, rather than relying on a predefined kernel function. To enable both clustering and generative tasks, the model incorporates a reconstruction loss along with an unsupervised clustering loss.

3.1 Spectral Clustering Loss

For a dataset $\{\boldsymbol{x}_i\}_{i=1}^n$, $\boldsymbol{x}_i \in \mathbb{R}^d$, and feature map $\phi : \mathcal{I} \subseteq \mathbb{R}^d \mapsto \mathcal{F} \subseteq \mathbb{R}^{d_f}$, the weighted variance maximization problem in $\mathcal{F}$ can be formulated as:

$$\max_{\boldsymbol{U}} \frac{1}{2} \sum_{i=1}^{n} D_{ii}^{-1} \left\| \boldsymbol{U}^\top \phi(\boldsymbol{x}_i) \right\|_2^2 \qquad \text{s.t. } \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_s, \tag{2}$$

where the weighting scalars are the inverse degrees of the kernel matrix, $D_{ii} = \sum_j \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j)$, and $\boldsymbol{U} \in \mathbb{R}^{d_f \times s}$ is a projection matrix with $s < d_f$.[1] The optimal $\boldsymbol{U}^*$ spans the eigenspace of the top s components of this weighted PCA problem. In traditional KSC problems, s is set to $k-1$, where k is the number of clusters to infer. In our model, we allow $s > k-1$, creating a richer latent representation for generation. The stationarity conditions of the optimization problem in (2) demonstrate the equivalence between the problems (1) and (2). The kernel is defined as $K_{ij} = \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j)$, and the principal component score vectors $\boldsymbol{e}_i = \boldsymbol{U}^\top \phi(\boldsymbol{x}_i)$ relate to the spectral embeddings $\boldsymbol{h}_i$ as: $\boldsymbol{e}_i = D_{ii} \boldsymbol{h}_i \boldsymbol{\Lambda}$.[2]

3.2 Augmented Losses

Problem (2) formulates the KSC problem for a given feature map. A novel aspect of this work is the use of learnable feature mappings, such as neural networks, in the KSC framework. This requires the addition of augmented loss terms. Following Pandey et al. [4], we incorporate an inverse mapping $\psi : \mathcal{F} \mapsto \mathcal{I}$, enabling an encoder-decoder architecture that facilitates representation learning. We denote the parametric feature map and its approximate inverse as $\phi(\cdot; \boldsymbol{\theta}_\phi)$ and $\psi(\cdot; \boldsymbol{\theta}_\psi)$, respectively. To optimize the parameters $\boldsymbol{\theta}_\phi$ and $\boldsymbol{\theta}_\psi$, we introduce a reconstruction error term. Since feature representations are projected onto the eigenspace through $\boldsymbol{U}$, this reconstruction depends on the KSC problem (3):

$$\mathcal{L}_{\text{rec}} = \sum_{i=1}^{n} \left\| \boldsymbol{x}_i - \psi\left(\boldsymbol{U}\boldsymbol{U}^\top \phi(\boldsymbol{x}_i; \boldsymbol{\theta}_\phi); \boldsymbol{\theta}_\psi \right) \right\|_2^2 .$$

Additionally, we introduce a cluster loss term. As depicted in Fig. 1, effective clustering in the KSC framework produces a line-structured distribution in the

[1] We assume that the feature map is centered with respect to the weighting scheme, which can be implemented by updating $\phi(\boldsymbol{x}) \leftarrow \phi(\boldsymbol{x}) - \sum_i D_{ii}^{-1} \phi(\boldsymbol{x}_i) / \sum_i D_{ii}^{-1}$.

[2] Refer to [9] for a detailed mathematical comparison between equivalent primal and dual formulations within a PCA framework.

score vector space. We predefine k directions for these lines using cluster codes ($\{\boldsymbol{s}_c\}_{c=1}^k$) and minimize the cosine distance of each representation to its closest cluster code. We set these cluster codes as the vertices of a regular $(k-1)$-simplex ensuring maximal angular separation between the equidistant vertices, leading to clear and distinct cluster directions. The cosine distance of a point to a cluster code, along with the total cluster loss, is given by:

$$d_{ic}^{\cos} = 1 - \frac{\boldsymbol{e}_i'^{\top} \boldsymbol{s}_c}{\|\boldsymbol{e}_i'\|_2 \, \|\boldsymbol{s}_c\|_2} \qquad \mathcal{L}_{\text{cl}} = \sum_{i=1}^{n} \min_c d_{ic}^{\cos},$$

where $\boldsymbol{e}_i' = \boldsymbol{e}_{i,1:k-1}$ represents the first $(k-1)$ elements of the score vectors. Note that the solution of (2) yields an arbitrary rotation of the first s components, distributing cluster information across all s components rather than solely within the first $(k-1)$. The proposed cosine distance loss encourages optimal rotation in the first $(k-1)$ components, and further enhances linearity within this subspace.

3.3 The GenKSC Model

Combining the above loss terms and adding regularization on feature representations, we arrive at the optimization problem for GenKSC:

$$\begin{aligned} \min_{\boldsymbol{U},\boldsymbol{\theta}_\phi,\boldsymbol{\theta}_\psi} \sum_{i=1}^{n} \Bigg(& -\frac{1}{2} D_{ii}^{-1} \left\| \boldsymbol{U}^\top \phi(\boldsymbol{x}_i;\boldsymbol{\theta}_\phi) \right\|_2^2 + \left\| \phi(\boldsymbol{x}_i;\boldsymbol{\theta}_\phi) \right\|_2^2 \\ & + \eta_{\text{rec}} \left\| \boldsymbol{x}_i - \psi\left(\boldsymbol{U}\boldsymbol{U}^\top \phi(\boldsymbol{x}_i;\boldsymbol{\theta}_\phi); \boldsymbol{\theta}_\psi \right) \right\|_2^2 + \eta_{\text{cl}} \min_c d_{ic}^{\cos} \Bigg) \qquad \text{s.t. } \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_s, \end{aligned} \tag{3}$$

where η_{rec} and η_{cl} are hyperparameters balancing the contributions of the respective loss terms; and where feature map parameters $\boldsymbol{\theta}_\phi$, inverse feature map parameters $\boldsymbol{\theta}_\psi$, and projection matrix $\boldsymbol{U}$ are the training parameters.

This formulation effectively constructs a spectral clustering problem within a feature space, while simultaneously learning the feature representations. After training, a new point $\boldsymbol{e}^*$ in the score variable space can be selected by targeting a specific cluster center to generate a representative datapoint, or sampled randomly to explore the latent space. The corresponding datapoint is then computed as $\boldsymbol{x}^* = \psi(\boldsymbol{U}\boldsymbol{e}^*)$.

4 Experiments

4.1 Datasets and Model Details

For clarity, we select a subset of the MNIST dataset, containing only the first three digit classes (0, 1, and 2), for a total of 18,732 images; termed MNIST012. For a more challenging experiment, we use the FashionMNIST dataset. For both datasets, convolutional neural networks were selected as parametric feature maps $\phi(\cdot;\boldsymbol{\theta}_\phi)$ with the approximate inverse feature map $\psi(\cdot;\boldsymbol{\theta}_\psi)$ using a mirrored architecture with transposed convolutions. For MNIST012, latent

space dimensions were set to $s = 10$ with $k = 3$, using three convolutional and two linear layers in the encoder. For FashionMNIST, we used $s = 40$, $k = 10$, with similar architectures as in ClusterGAN. To avoid clustering on arbitrary features, cluster loss was excluded from the objective function for the first 10 epochs on MNIST012 and 32 epochs on FashionMNIST, allowing the model to develop meaningful representations before clustering. We use Cayley ADAM [10] to enforce the orthonormal constraint on $\boldsymbol{U}$. Loss weights η_{rec} and η_{cl} were set to 1 for MNIST012, while for FashionMNIST, the values $\eta_{\text{rec}} = 0.001$ and $\eta_{\text{cl}} = 0.008$ were determined through hyperparameter tuning based on the average membership strength criterion, as in [8].

4.2 Results

In Fig. 2, the spectral embedding space for MNIST012 shows well-separated clusters where the traversals along the cluster directions give us an indication on which features the model has clustered the data. Compared to the line structure of classical KSC in Fig. 1, the GenKSC model has enabled the generation of new points, even beyond the farthest point on the cluster line representing the cluster prototype, allowing us to exaggerate the characteristic feature in the cluster revealing that thinner digits are harder to cluster.

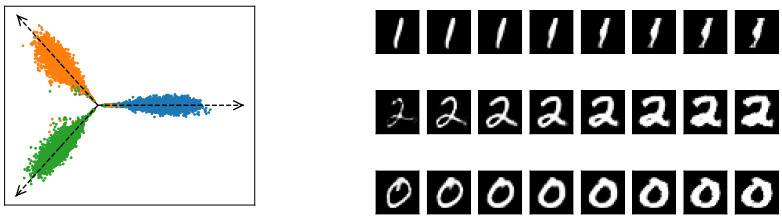

Fig. 2: Generated images along indicated cluster directions of the first two dimensions of the latent space for MNIST012.

The model and line structures generalizes to clusters $k > 3$. Fig. 3 shows the traversals along the high-dimensional cluster directions for 6 clusters of the FashionMNIST dataset. Again, the extrapolations in the latent space yield characterizations of the features that are indicative for the clustering. For example, in rows 1 and 5, two pant legs become more distinct, and the shoulder straps of the dress become more prominent. Additionally, generated points along higher components, like shown on the right, enable us to observe intra-cluster variations, such as the distinction between sleeveless and T-shirt sleeves.

5 Conclusion

The GenKSC model has demonstrated its ability to combine representational learning with a clustering objective to yield an explorable latent clustering space.

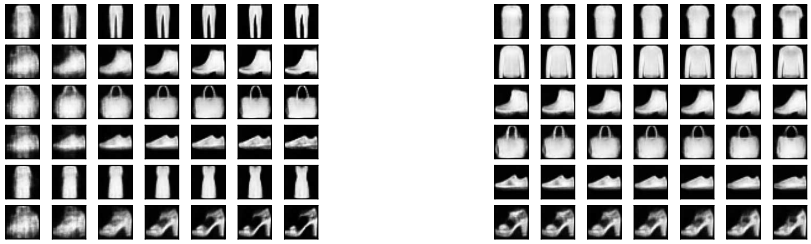

Fig. 3: Latent space traversals for the FashionMNIST dataset. **Left**: The traversals along the cluster directions in the 9-dimensional latent subspace. **Right**: Traversals along the k-th dimension of the latent space.

The key idea is that by extrapolating in the latent space, we generate new data points that emphasize or exaggerate distinctive cluster features—something that existing methods cannot achieve. Future work can include generalizing to a semi-supervised setting and even a fully supervised setting where the cluster labels could be given by another clustering model, potentially creating an interpretable clustering latent space from any clustering model.

References

[1] Yazhou Ren, Jingyu Pu, Zhimeng Yang, Jie Xu, Guofeng Li, Xiaorong Pu, Philip S. Yu, and Lifang He. Deep Clustering: A Comprehensive Survey. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–21, 2024. Early Access.

[2] Lianyu Hu, Mudi Jiang, Junjie Dong, Xinying Liu, and Zengyou He. Interpretable Clustering: A Survey, 2024. arXiv:2409.00743 [cs].

[3] Johan A. K. Suykens. Deep Restricted Kernel Machines Using Conjugate Feature Duality. *Neural Computation*, 29(8):2123–2163, 2017.

[4] Arun Pandey, Michaël Fanuel, Joachim Schreurs, and Johan A. K. Suykens. Disentangled Representation Learning and Generation With Manifold Optimization. *Neural Computation*, 34(10):2009–2036, 09 2022.

[5] Diederik P. Kingma and Max Welling. Auto-encoding Variational Bayes. In *International Conference on Learning Representations*, 2014.

[6] Sudipto Mukherjee, Himanshu Asnani, Eugene Lin, and Sreeram Kannan. ClusterGAN: Latent Space Clustering in Generative Adversarial Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):4610–4617, Jul. 2019.

[7] Carlos Alzate and Johan A. K. Suykens. Multiway Spectral Clustering with Out-of-sample Extensions through Weighted Kernel PCA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):335–347, 2010.

[8] Rocco Langone, Raghvendra Mall, and Johan A. K. Suykens. Soft kernel spectral clustering. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2013.

[9] Sonny Achten, Arun Pandey, Hannes De Meulemeester, Bart De Moor, and Johan A. K. Suykens. Duality in Multi-View Restricted Kernel Machines. ICML Workshop on Duality for Modern Machine Learning, 2023. arXiv:2305.17251 [cs].

[10] Jun Li, Fuxin Li, and Sinisa Todorovic. Efficient Riemannian Optimization on the Stiefel Manifold via the Cayley Transform. In *International Conference on Learning Representations*, 2019.