Dancing Between Phases: Self-Organized Learning Transitions in Competing Associative Networks

Saleh Sargolzaei* and Luis Rueda*
*School of Computer Science, University of Windsor, Windsor, Ontario, Canada
{sargolz, lrueda}@uwindsor.ca

***Abstract*—We investigate emergent learning dynamics through DANCE (Dynamic Associative Network for Cluster Emergence), a clustering algorithm based on competing Hopfield networks. DANCE exhibits a natural transition from exploration to refinement phases without explicit scheduling. This transition emerges from the adaptive repulsion between networks, which scales inversely with the number of prototypes. By tracking energy evolution, cluster formation rates, and network competition, we demonstrate how the system self-organizes its learning behavior through distinct regimes. The dynamics show how adaptive systems can naturally develop a curriculum, first exploring the feature space broadly before shifting to refining established knowledge, offering insights into emergent learning regimes in neural systems.**

***Index Terms*—Hopfield networks, clustering, associative memory, energy-based models, nonparametric clustering, competitive learning, unsupervised learning, adaptive clustering, novelty detection.**

I. Introduction

How adaptive learning systems transition between different regimes remains a key question in understanding high-dimensional learning dynamics [1], [2]. Clustering algorithms offer an accessible framework for studying such dynamics, particularly when they adaptively determine structural complexity. Classical clustering methods often require fixed parameterization [3], [4], while Bayesian nonparametric approaches offer adaptivity at higher computational cost [5]. Recent approaches using associative memory models for clustering [6], [7], [8] have advanced this field, though many still require specifying the number of clusters upfront.

We propose DANCE (Dynamic Associative Network for Cluster Emergence), a clustering algorithm using interacting Hopfield networks [9]. Inspired by ensemble approaches to memory retrieval [10], DANCE leverages competitive dynamics between networks to adaptively determine cluster structure. The primary focus of this paper is on analyzing the learning dynamics that emerge from this competition. By tracking energy landscapes and visualizing network competition, we identify a clear phase transition: a shift from a regime dominated by prototype creation to one focused on refinement. This transition occurs without explicit scheduling, emerging naturally from the adaptive repulsion parameter that evolves with network structure.

Our contribution lies in quantitatively characterizing this phase transition and demonstrating how energy-based competition drives self-organization. This analysis provides insights into how adaptive systems naturally regulate their learning behavior as they accumulate structure. Code is available at https://github.com/salehsargolzaee/dance.

II. DANCE: Dynamic Associative Network for Cluster Emergence

The following presents the mathematical framework and key dynamics properties.

A. Mathematical Framework

At time t, upon observing a data point $\mathbf{x}_t \in [-1,1]^N$, DANCE maintains K_t prototype vectors (memories) $\mathcal{M}_t = \{\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^{K_t}\}$. We instantiate $L_t = K_t + 1$ parallel Hopfield networks: K_t networks for existing prototypes plus a candidate network for testing $\mathbf{x}_t$. Each network has state $\boldsymbol{\sigma}^a \in [-1,1]^N$, with normalized overlaps $m_\mu^a = \frac{1}{N}\sum_{i=1}^N \xi_i^\mu \sigma_i^a$.

The energy function is:

$$E(\boldsymbol{\sigma}) = -N\sum_{a=1}^{L_t}\sum_{\mu=1}^{K_t}(m_\mu^a)^2 - H\sum_{a=1}^{L_t}\sum_{i=1}^{N} h_i^a \sigma_i^a + \lambda N \sum_{\substack{a,b=1 \\ a\neq b}}^{L_t}\Big(\sum_{\mu=1}^{K_t} m_\mu^a m_\mu^b\Big)^2, \quad (1)$$

where $\mathbf{h}^a = \boldsymbol{\xi}^a$ for $a \le K_t$ and $\mathbf{h}^{K_t+1} = \mathbf{x}_t$. This creates competition via: (i) alignment with prototypes, (ii) attraction to external fields, and (iii) inter-network repulsion.

B. Network Dynamics and Adaptive Repulsion

Networks are initialized from a shared mixture state $\boldsymbol{\sigma}_{\text{init}}^a = g(\sum_{b=1}^{L_t}\mathbf{h}^b)$. During each of U update iterations, we compute effective fields and update all neurons synchronously:

$$F_i^a = -2\sum_{\mu=1}^{K_t} m_\mu^a \xi_i^\mu - H h_i^a + 2\lambda \sum_{b\neq a}\left(\sum_{\mu=1}^{K_t} m_\mu^a m_\mu^b\right)\sum_{\mu=1}^{K_t} m_\mu^b \xi_i^\mu, \quad (2)$$

$$\sigma_i^a \leftarrow g(-F_i^a), \quad (3)$$

where g is either sign or $\tanh$. These updates monotonically decrease the energy function (Theorem 1), ensuring convergence.

The key to DANCE's phase transition is the adaptive repulsion $\lambda_t = \frac{2}{L_t^2 \times \text{mult}}$, where "mult" is a hyperparameter

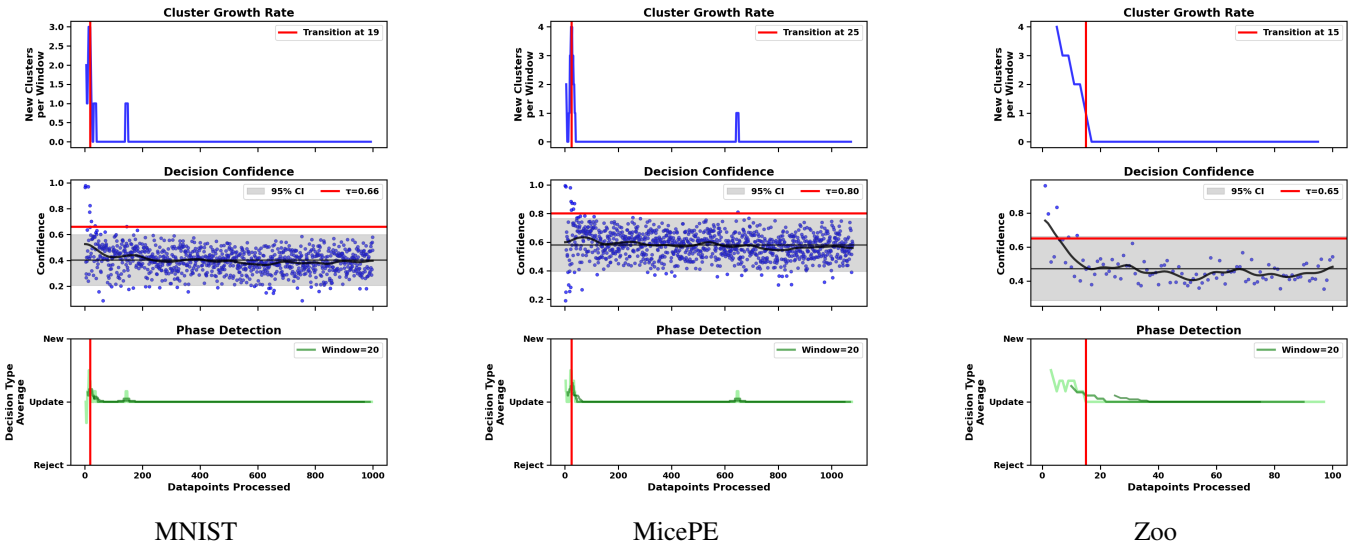


Fig. 1. Phase transition analysis across datasets. Each plot shows the cluster growth rate (top), confidence values (middle), and decision types (bottom) over time. Red vertical lines indicate detected transition points.

controlling the overall repulsion strength without changing its scaling behavior. The adaptive repulsion mechanism creates a natural phase transition, based on two key theoretical results (Theorems 2 and 3):

First, with sufficient repulsion strength, networks naturally separate to represent distinct prototypes, with the candidate network either aligning with an existing prototype or the new data point (Theorem 2). Second, if repulsion strength remained fixed as the network grows, new cluster formation would eventually cease entirely (Theorem 3). This necessitates an adaptive schedule where repulsion scales inversely with network size (Corollary 1). This creates a natural two-phase behavior: (i) exploration phase (small L_t) with strong repulsion creating distinct clusters, and (ii) refinement phase (large L_t) with weaker repulsion favoring prototype updates.

C. Decision Process

After convergence, we evaluate candidate network state $\boldsymbol{\sigma}^{K_t+1,(U)}$ using softmax probabilities:

$$\pi_a = \frac{\exp\left(\beta\langle\boldsymbol{\sigma}^{K_t+1,(U)}, \mathbf{h}^a\rangle\right)}{\sum_{b=1}^{L_t}\exp\left(\beta\langle\boldsymbol{\sigma}^{K_t+1,(U)}, \mathbf{h}^b\rangle\right)}, \tag{4}$$

where $\beta > 0$ is the inverse temperature parameter controlling the sharpness of the probability distribution. If $\pi_{K_t+1} > \tau$, we create a new prototype $\mathcal{M}_{t+1} = \mathcal{M}_t \cup \{\mathbf{x}_t\}$. Otherwise, we update the most likely existing prototype: $\boldsymbol{\xi}^{i_{\max}} \leftarrow \boldsymbol{\xi}^{i_{\max}} + \alpha\pi_{i_{\max}}(\boldsymbol{\sigma}^{K_t+1,(U)} - \boldsymbol{\xi}^{i_{\max}})$, where $i_{\max} = \arg\max_{1\le\mu\le K_t}\pi_\mu$ and α is a learning rate. Points with high entropy in their assignment distribution can be rejected entirely to prevent corruption from ambiguous data (Appendix B). The adaptive repulsion produces a gradual phase transition from exploration to refinement without explicit scheduling, making DANCE an ideal system for studying emergent learning dynamics.

D. Computational Complexity

Per-point complexity is $O(U\,N\,L_t^2)$ for $N \gg L_t$, dominated by computing overlaps and repulsive fields across $L_t = K_t+1$ networks. With cluster cap $K_{\max}$, total stream complexity is $O(U\,T\,K_{\max}^2\,N)$. Space is $O(K_{\max}N)$ for storing prototypes and network states. This is substantially better than spectral/agglomerative clustering's $O(T^2)$ space and $O(T^3)$ time [11], [12].

III. Experimental Analysis of Learning Dynamics

In this section, we analyze the emergent learning dynamics, with particular focus on characterizing the phase transition from exploration to refinement that arises naturally from network competition.

A. Phase Transition Detection

We track cluster growth rate $G_W(t) = (K_{t+W} - K_t)/W$ with $W = 10$, which works across all datasets regardless of size, suggesting scale invariance. Transitions are identified via z-score normalization: $z(t) = (\nabla G_W(t) - \mu_{\nabla G})/\sigma_{\nabla G}$, with peaks below -1.96 (95% confidence) indicating phase transitions. Details in Appendix B.

B. Cross-dataset Analysis

Figure 1 shows phase transitions across MNIST, MicePE, and Zoo datasets [13], [14], [15]. MNIST (left) transitions at datapoint 19, after which cluster formation becomes less frequent. The secondary spike around datapoint 100 demonstrates that post-transition, DANCE can still discover genuinely novel patterns, though with higher novelty requirements. Confidence values after transition rarely exceed 0.6, confirming increased difficulty of cluster formation during refinement.

MicePE (center) transitions at datapoint 25, with a pronounced secondary cluster creation phase around datapoint 600, suggesting complex dataset structure requiring multiple

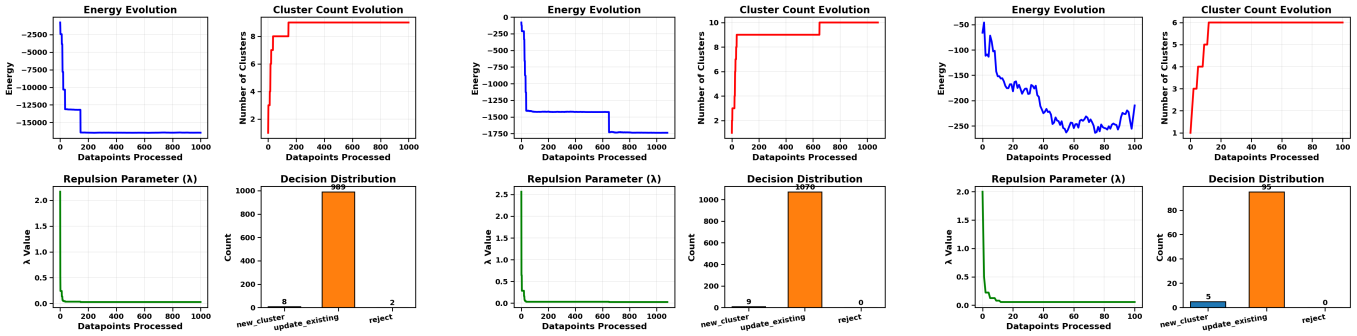


Fig. 2. Energy landscape and parameter evolution across datasets. Top row: energy (left) and cluster count (right) over time. Bottom row: repulsion parameter λ evolution (left) and final decision distribution (right).

exploration phases. Zoo (right) transitions earliest at datapoint 15, with a smoother profile reflecting its smaller size. The adaptive λ (Figure 2) drops rapidly during exploration before stabilizing during refinement, confirming theoretical predictions about adaptive repulsion driving phase transitions.

C. Energy and Repulsion Dynamics

Figure 2 illustrates energy evolution, cluster counts, and repulsion parameter λ across datasets. The energy landscape shows initial rapid drops during exploration as prototypes are established, followed by stabilization during refinement. MicePE exhibits a second energy drop coinciding with its later cluster formation phase. The repulsion parameter λ decays according to our adaptive schedule, with the decay rate correlating with exploration phase length. This decay alters the energy landscape topology, making prototype separation less energetically favorable over time. Decision distributions confirm refinement dominates after transition, demonstrating automatic curriculum emergence.

D. Network Competition Analysis

Figure 3 illustrates competition for an MNIST digit "2". The candidate initially has similar overlap with multiple prototypes, but through updates, overlap with Prototype 1 increases while others decrease—a winner-take-all dynamic resolving ambiguity via energy minimization. The right panel shows pixel-level prototype updates after winning. Post-transition, this competition consistently favors existing prototypes.

IV. Clustering Performance

This section provides comprehensive experimental details supporting the main clustering results presented in Section 4.

1) Experimental Setup: All experiments were conducted on a MacBook Air with Apple M1 chip and 8GB memory without specialized hardware acceleration.

a) Datasets: We evaluated on MNIST [13], F-MNIST [16], Zoo [15], and MicePE [14]. Image data was flattened to 784D; MNIST binarized to $\{-1, 1\}$, F-MNIST normalized to $[-1, 1]$. Biological features mapped to $\{-1, 1\}$ via thresholding. F-MNIST trained on 1K samples, evaluated on 10K. See Table II for details.

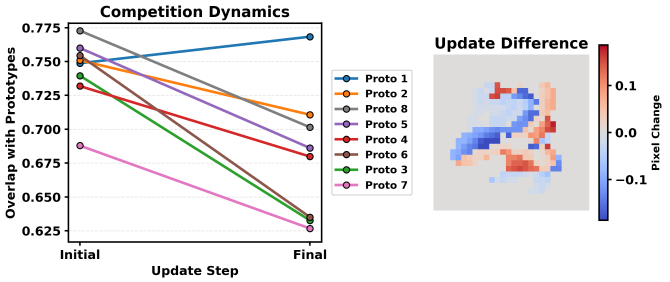


Fig. 3. Left: Competition dynamics showing how prototype overlaps evolve during updates. While the candidate initially has similar overlap with multiple prototypes, the overlap with Prototype 1 increases while others decrease. Right: Pixel-level update difference for Prototype 1, showing strengthened (red) and weakened (blue) features.

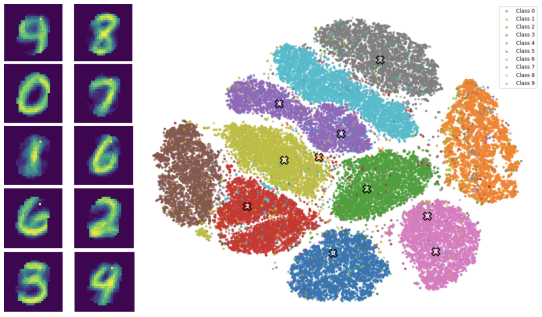


Fig. 4. MNIST results: learned prototypes (left) and t-SNE visualization with data points as circles, prototypes as crosses (right).

b) Evaluation measures: Clustering performance was evaluated using both external and internal metrics. External metrics included the Adjusted Rand Index (ARI)[17] and Normalized Mutual Information (NMI)[18], both of which assess alignment with ground truth labels. Internal validation was performed using the Silhouette Score[19], which assesses the compactness and separation of the clusters.

2) Results and Discussions: Baselines: scikit-learn [20] (K-Means++ [21], Spectral, Agglomerative) and DP-Means[1].

[1]https://pdc-dp-means.readthedocs.io/en/latest/

TABLE I
CLUSTERING PERFORMANCE COMPARISON. BEST SCORES **BOLDED**, RUNNER-UP UNDERLINED.

Dataset	Method	NMI	ARI	Silh.
Zoo	K-Means	0.833	0.737	0.374
	Spectral	0.889	0.866	0.398
	Agglom.	0.843	0.911	0.398
	DP-Means	0.805	0.684	0.375
	CLAM	0.943	0.964	**0.412**
	DANCE	**0.965**	**0.982**	0.382
MicePE	K-Means	0.237	0.126	0.128
	Spectral	0.006	-0.002	0.156
	Agglom.	0.260	0.156	0.193
	DP-Means	**0.548**	0.051	**0.308**
	CLAM	0.311	0.165	0.201
	DANCE	0.489	**0.306**	0.095
F-MNIST	K-Means	0.504	0.346	0.126
	Spectral	**0.643**	**0.431**	0.075
	Agglom.	0.563	0.368	**0.143**
	DP-Means	0.493	0.168	0.066
	CLAM	0.518	0.367	0.138
	DANCE	0.516	0.374	0.141
MNIST	K-Means	0.504	0.346	**0.127**
	Spectral	**0.696**	**0.514**	0.035
	Agglom.	0.664	0.486	0.028
	DP-Means	0.493	0.174	0.065
	CLAM	–	–	–
	DANCE	0.526	0.424	0.071

TABLE II
DATASET SPLITS AND SAMPLE COUNTS USED IN EXPERIMENTS. N/A INDICATES THE TRAINING PORTION WAS USED FOR BOTH TRAINING AND EVALUATION.

Dataset	Training Samples	Testing Samples	Classes	Features
MNIST	60,000	N/A	10	784
F-MNIST	1,000	10,000	10	784
Zoo	101	N/A	7	16
MicePE	1,080	N/A	8	80

CLAM results from [6]. Spectral/Agglomerative used 10K subsets for F-MNIST/MNIST. DANCE hyperparameters in Table III.

TABLE III
HYPERPARAMETERS USED IN TABLE I FOR DANCE

Dataset	mult	H	β^{-1}	τ	α	Updates
Zoo	1.0000	50	0.39	0.65	3.10	1
MicePE	0.7800	0.7	0.08	0.8	1.3	1
F-MNIST	0.1000	1.2	0.08	0.75	0.10	7
MNIST	0.9225	1.2	0.08	0.7	0.8	1

DANCE achieved top or second-best in 7/12 measurements (Table I), the highest among all methods. While Spectral outperformed DANCE on F-MNIST/MNIST, it requires pre-specifying cluster numbers, whereas DANCE automatically infers them. This trade-off is favorable given DANCE's lower space complexity $O(K_{\max}N)$ versus $O(T^2)$ for spectral methods.

DANCE consistently outperformed CLAM [6], the only other memory-based method, using simpler quadratic energy terms versus CLAM's log-sum-exponential function. This suggests that inter-network competition can substitute for increased single-network capacity. Additionally, DANCE avoids the computational challenges of backpropagation through time [22]. DP-Means, the only other non-parametric method, surpassed DANCE only on MicePE NMI/Silhouette, while DANCE achieved better ARI (which adjusts for chance agreement).

Figure 4 shows MNIST prototypes and t-SNE visualization. The learned prototypes provide meaningful digit class summaries, with labels representing majority assignments. Figure 5 shows t-SNE visualizations for Zoo and MicePE, demonstrating excellent prototype separation even on small datasets. Figure 6 shows F-MNIST prototypes using higher-order energy terms (Remark 1).

Remark 1 (Kernelized Energy). For F-MNIST, we used cubic powers for alignment/repulsion and quadratic for external field terms, similar to kernelized Hopfield networks [23]. Higher-order energies produce redundant prototypes requiring merging (Appendix B).

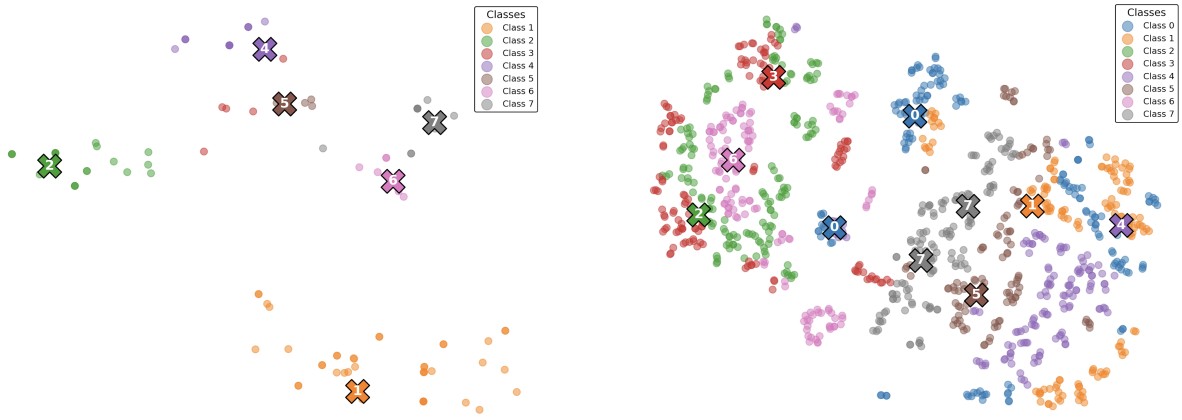
a) Order Sensitivity: DANCE processes data sequentially, raising questions about ordering sensitivity. Table IV shows comprehensive metrics across 10 random permutations. The variation (e.g., Zoo: ARI 0.81 ± 0.09, NMI 0.84 ± 0.05; MNIST: ARI 0.34 ± 0.03) is comparable to K-Means' initialization dependence. Notably, cluster counts remain close to ground truth (6.1 ± 0.3 for Zoo's 7 classes, 10.2 ± 1.1 for MNIST's 10), and DANCE demonstrates consistent performance across all metrics rather than optimizing for specific measures. Entropy-based filtering (Appendix B) helps maintain consistency by rejecting ambiguous points during exploration.

TABLE IV
DANCE PERFORMANCE ACROSS 10 RANDOM PERMUTATIONS. MNIST USES 10K-SAMPLE SUBSETS.

Metric	Zoo	MicePE	MNIST
ARI	$0.811_{\pm 0.09}$	$0.247_{\pm 0.03}$	$0.344_{\pm 0.03}$
NMI	$0.844_{\pm 0.05}$	$0.411_{\pm 0.04}$	$0.482_{\pm 0.02}$
FMI	$0.854_{\pm 0.07}$	$0.343_{\pm 0.03}$	$0.421_{\pm 0.03}$
Silhouette	$0.352_{\pm 0.03}$	$0.091_{\pm 0.00}$	$0.070_{\pm 0.00}$
Duration (s)	$0.047_{\pm 0.01}$	$0.575_{\pm 0.11}$	$16.79_{\pm 4.11}$
Clusters	$6.10_{\pm 0.32}$	$9.30_{\pm 1.06}$	$10.20_{\pm 1.14}$

V. CONCLUSION

We introduced DANCE, a clustering algorithm based on competing Hopfield networks that exhibits emergent phase transitions from exploration to refinement. Our theoretical analysis established that: (1) synchronous updates guarantee energy descent (Theorem 1), (2) sufficient repulsion ensures prototype exclusivity (Theorem 2), and (3) fixed repulsion leads to cluster freezing, necessitating adaptive scheduling (Theorem 3, Corollary 1).



(a) Zoo dataset (b) MicePE dataset

Fig. 5. t-SNE visualizations for Zoo and MicePE. Data points (circles) and prototypes (crosses) colored by ground truth and majority labels respectively.

Experimentally, we demonstrated that DANCE achieves competitive clustering performance while automatically inferring cluster numbers. The phase transition analysis revealed scale-invariant behavior across diverse datasets, with the adaptive repulsion parameter λ directly governing the exploration-to-refinement transition.

This work provides insights into how competitive dynamics in neural systems can naturally produce curriculum-like learning without explicit scheduling.

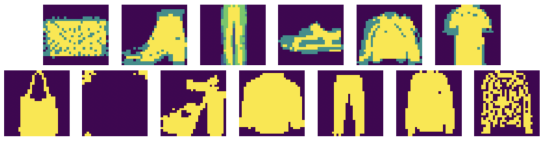


Fig. 6. Fashion-MNIST prototypes after merging redundant patterns from the kernelized energy function (see Appendix B).

Acknowledgments

Partially supported by NSERC and the Vector Institute. The authors used ChatGPT and Claude for formatting mathematical proofs.

References

[1] L. Zdeborová and F. Krzakala, "Statistical physics of inference: Thresholds and algorithms," *Advances in Physics*, vol. 65, no. 5, pp. 453–552, 2016.

[2] M. Advani and S. Ganguli, "Statistical mechanics of optimal convex inference in high dimensions," *Phys. Rev. X*, vol. 6, p. 031034, Aug 2016. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevX.6.031034

[3] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, vol. 5. University of California press, 1967, pp. 281–298.

[4] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 14, 2001.

[5] R. M. Neal, "Markov chain sampling methods for dirichlet process mixture models," *Journal of computational and graphical statistics*, vol. 9, no. 2, pp. 249–265, 2000.

[6] B. Saha, D. Krotov, M. J. Zaki, and P. Ram, "End-to-end differentiable clustering with associative memories," in *International Conference on Machine Learning*. PMLR, 2023, pp. 29 649–29 670.

[7] R. Schaeffer, M. Khona, N. Zahedi, I. R. Fiete, A. Gromov, and S. Koyejo, "Associative memory under the probabilistic lens: Improved transformers & dynamic memory creation," in *Associative Memory & Hopfield Networks in 2023*, 2023. [Online]. Available: https://openreview.net/forum?id=lO61aZlteS

[8] R. Schaeffer, N. Zahedi, M. Khona, D. Pai, S. Truong, Y. Du, M. Ostrow, S. Chandra, A. Carranza, I. R. Fiete *et al.*, "Bridging associative memory and probabilistic modeling," *arXiv preprint arXiv:2402.10202*, 2024.

[9] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities." *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.

[10] E. Agliari, A. Alessandrelli, A. Barra, M. S. Centonze, and F. Ricci-Tersenghi, "Networks of neural networks: more is different," 2025. [Online]. Available: https://arxiv.org/abs/2501.16789

[11] W. H. Day and H. Edelsbrunner, "Efficient algorithms for agglomerative hierarchical clustering methods," *Journal of classification*, vol. 1, no. 1, pp. 7–24, 1984.

[12] L. Ding, C. Li, D. Jin, and S. Ding, "Survey of spectral clustering based on graph theory," *Pattern Recognition*, p. 110366, 2024.

[13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[14] C. Higuera, K. J. Gardiner, and K. J. Cios, "Self-organizing feature maps identify proteins critical to learning in a mouse model of down syndrome," *PLoS ONE*, vol. 10, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:7955336

[15] R. Forsyth, "Zoo," UCI Machine Learning Repository, 1990, DOI: https://doi.org/10.24432/C5R59V.

[16] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[17] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, vol. 2, pp. 193–218, 1985.

[18] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *Journal of machine learning research*, vol. 3, no. Dec, pp. 583–617, 2002.

[19] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[21] S. Vassilvitskii and D. Arthur, "k-means++: The advantages of careful

seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, 2006, pp. 1027–1035.

[22] P. Schnell and N. Thuerey, "Stabilizing backpropagation through time to learn complex physics," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=bozbTTWcaw

[23] G. Iatropoulos, J. Brea, and W. Gerstner, "Kernel memory networks: A unifying framework for memory modeling," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022. [Online]. Available: https://openreview.net/forum?id=px87A_nzK-T

APPENDIX

A. Theoretical Results

Theorem 1 (Convergence)**.** *The synchronous network updates monotonically decrease the energy function. If* $E^{(k)} = E(\Phi^k(\boldsymbol{\sigma}^{(0)}))$*, then* $E^{(k+1)} \leq E^{(k)}$ *for all* k*.*

Proof. Let $\widehat{\boldsymbol{\sigma}} = \Phi_a(\boldsymbol{\sigma})$ denote the state after updating network a. Define $\Delta\sigma_i = \widehat{\sigma}_i^a - \sigma_i^a$, $\Delta m_\mu = \frac{1}{N}\sum_i \xi_i^\mu \Delta\sigma_i$, and let $k = |\{i : \Delta\sigma_i \neq 0\}|$ be the number of flipped spins. The energy change decomposes as:

$$\Delta E = -\sum_i F_i^a \Delta\sigma_i - N\sum_\mu (\Delta m_\mu)^2 + \lambda N \sum_{b\neq a}(\Delta S_{ab})^2 \tag{5}$$

where $\Delta S_{ab} = \sum_\mu \Delta m_\mu m_\mu^b$. The first term is linear and non-positive by the update rule $\widehat{\sigma}_i^a = g(-F_i^a)$; the second is quadratic non-positive; the third is quadratic non-negative but bounded by $O(k^2K_t^3/N)$.

Bounding the quadratic terms: Since $|\Delta\sigma_i| = 2$ for flipped spins, we have $|\Delta m_\mu| \leq 2k/N$ and $|\Delta S_{ab}| \leq 2kK_t/N$. Thus:

$$-\frac{4K_t k^2}{N} \leq -N\sum_\mu (\Delta m_\mu)^2 \leq 0 \tag{6}$$

$$0 \leq \lambda N \sum_{b\neq a}(\Delta S_{ab})^2 \leq \frac{4\lambda k^2 K_t^3}{N} \tag{7}$$

Linear term dominance: For every flipped spin, $-F_i^a\Delta\sigma_i = 2|F_i^a|$, so the linear term satisfies $-\sum_i F_i^a\Delta\sigma_i = 2\sum_{i:\Delta\sigma_i\neq 0}|F_i^a| \geq 2kF_{\min}^a$. Combining yields $\Delta E < 0$ whenever $F_{\min}^a > \frac{2k}{N}(K_t + \lambda K_t^3)$. For large N this is satisfied, and composing updates across all L networks preserves monotonicity. □

Theorem 2 (Prototype Exclusivity)**.** *There exists* $\lambda_0 > 0$ *such that if* $\lambda > \lambda_0$*, every stable fixed point satisfies: (i) each network aligns most strongly with a unique prototype, and (ii) the candidate network aligns with either one existing prototype or the new data point.*

Proof. Let $\boldsymbol{\sigma}^\star$ be a stable fixed point with minimum dominant overlap $C = \min_{a\leq K_t} m_{\mu(a)}^a$.

Part (i): Suppose networks a, b share dominant prototype ν, so $m_\nu^a, m_\nu^b \geq C$. Consider perturbing network a by $\widetilde{\sigma}_i^a = \sigma_i^{a,\star} - 2\varepsilon\xi_i^\nu$ for small $\varepsilon > 0$, which reduces m_ν^a by 2ε. The first-order energy change is:

$$E(\widetilde{\boldsymbol{\sigma}}) - E(\boldsymbol{\sigma}^\star) = 2N\varepsilon\big[2m_\nu^a + H - 2\lambda m_\nu^a (m_\nu^b)^2\big] + O(\varepsilon^2) \tag{8}$$

This is negative when $\lambda > (2m_\nu^a + H)/(2m_\nu^a(m_\nu^b)^2) \geq (2C + H)/(2C^3)$, contradicting stability. Hence networks cannot share prototypes.

Part (ii): For the candidate network c, the proof is identical with $m_{x_t}^c$ playing the role of alignment to the external field weighted by H. If the candidate ties between multiple prototypes, perturbing toward the larger overlap lowers energy under the same λ bound. □

Theorem 3 (Candidate Freezing)**.** *If repulsion* λ *remains fixed while* $L = K_t + 1$ *grows, new cluster formation eventually ceases.*

Proof. For candidate network c with external field alignment $\alpha_{\mathbf{x}}$ and prototype alignment α_ξ, we analyze three energy components:

1. Alignment: Using Cauchy-Schwarz, $\sum_\mu (m_\mu^c)^2 \leq 1 + \delta(K_t - 1)$ where δ is the maximum inter-prototype overlap, giving $-N\sum_\mu (m_\mu^c)^2 \geq -N(1 + \delta(K_t - 1))$.

2. External field: With alignment $\alpha_{\mathbf{x}}$, this contributes $-HN\alpha_{\mathbf{x}}$.

3. Repulsion: For each existing network b with $\sigma^b = \boldsymbol{\xi}^b$, we have $\sum_\mu m_\mu^c m_\mu^b \geq \alpha_\xi - \delta\eta$ where $\eta = \sum_{\mu\neq d}|m_\mu^c| \leq \sqrt{(K_t - 1)(1 + \delta(K_t - 2))}$. Thus the repulsion contributes at least $\lambda N(L-1)(\alpha_\xi - \delta\eta)^2$.

Combining: $E_c^{\min} \geq -N(1+\delta(K_t-1)) - HN\alpha_{\mathbf{x}} + \lambda N(L-1)(\alpha_\xi - \delta\eta)^2$. For existing prototypes, $E(\boldsymbol{\xi}^a) = -(1+H)N$. New cluster formation requires $E_c^{\min} < E(\boldsymbol{\xi}^a)$, which fails when $\lambda(L-1)(\alpha_\xi - \delta\eta)^2 > \delta(K_t - 1) + H(1 - \alpha_{\mathbf{x}})$. For fixed λ and growing L, this eventually occurs. □

Corollary 1 (Necessity of Adaptive Repulsion)**.** *To maintain cluster formation ability,* λ *must scale as* $\lambda \propto 1/(L-1)$*.*

Proof. From Theorem 3, avoiding freezing requires $\lambda(L-1)(\alpha_\xi - \delta\eta)^2 \leq \delta(K_t - 1) + H(1 - \alpha_{\mathbf{x}})$. For novel inputs ($\alpha_\xi \approx 0$, $\alpha_{\mathbf{x}} \approx 1$) with $\eta \sim \sqrt{L-1}$, this simplifies to $\lambda(L-1)\delta^2(L-1) \leq \delta(L-2)$, giving $\lambda \lesssim 1/(\delta(L-1))$. This justifies the adaptive schedule $\lambda_t = 2/(L_t^2 \times \text{mult})$. □

B. Additional Details

Entropy-based filtering: After computing assignment probabilities $p_\mu = \exp(\beta s_\mu)/\sum_\nu \exp(\beta s_\nu)$ where s_μ is cosine similarity to prototype μ, we compute entropy $S = -\sum_\mu p_\mu \log(p_\mu + \epsilon)$. Points exceeding an entropy threshold are rejected to prevent ambiguous data from corrupting prototypes.

Phase detection: We use window size $W = 10$ for growth rate $G_W(t) = (K_{t+W} - K_t)/W$. Transitions are identified via normalized derivative $z(t) = (\nabla G_W(t) - \mu_{\nabla G})/\sigma_{\nabla G}$, with peaks below $-\zeta$ (where $\zeta = 1.96$ for 95% confidence) indicating phase transitions.

Prototype merging (F-MNIST): Higher-order energy functions may produce redundant prototypes. We compute pairwise distances $D_{ij} = d(\boldsymbol{\xi}^i, \boldsymbol{\xi}^j)$, normalize to $[0, 1]$, and apply agglomerative clustering with average linkage and threshold $\tau_{\text{merge}} = 0.5$. Merged prototypes are computed as cluster centroids.