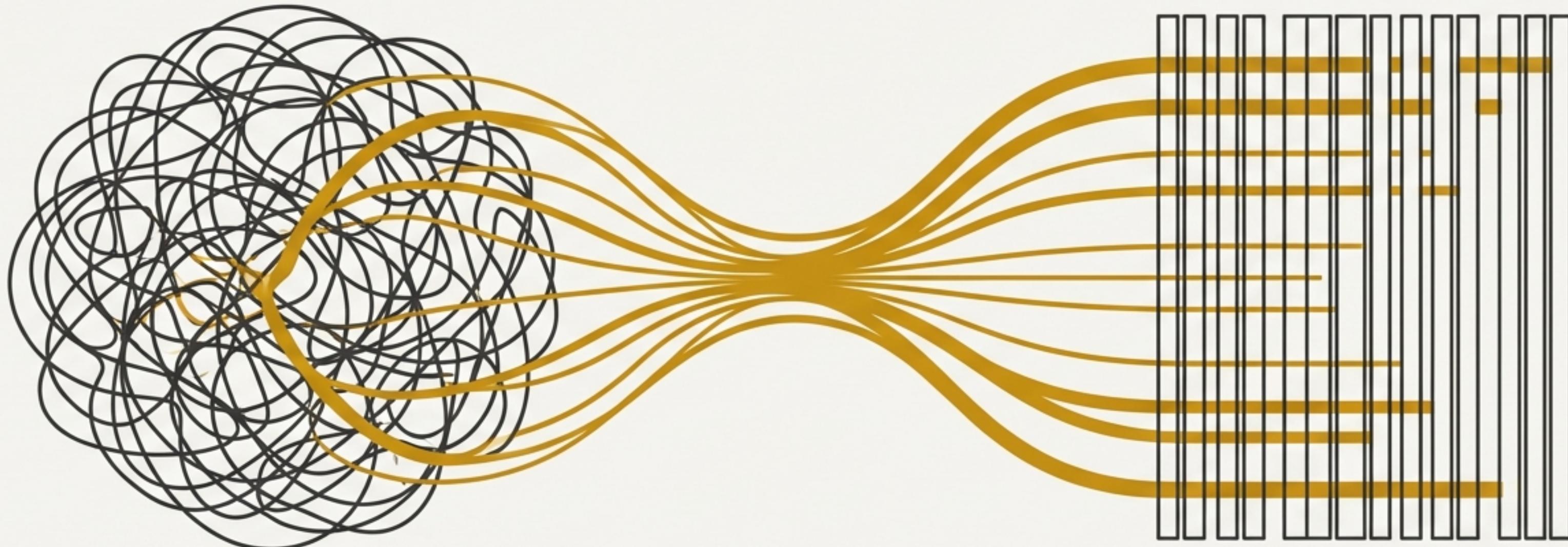


# NSA-Flow: Orthogonalish Matrix Optimization for Interpretable Embeddings

A general-purpose framework that unifies sparse matrix factorization, orthogonalization, and constrained manifold learning to balance reconstruction fidelity and column-wise decorrelation

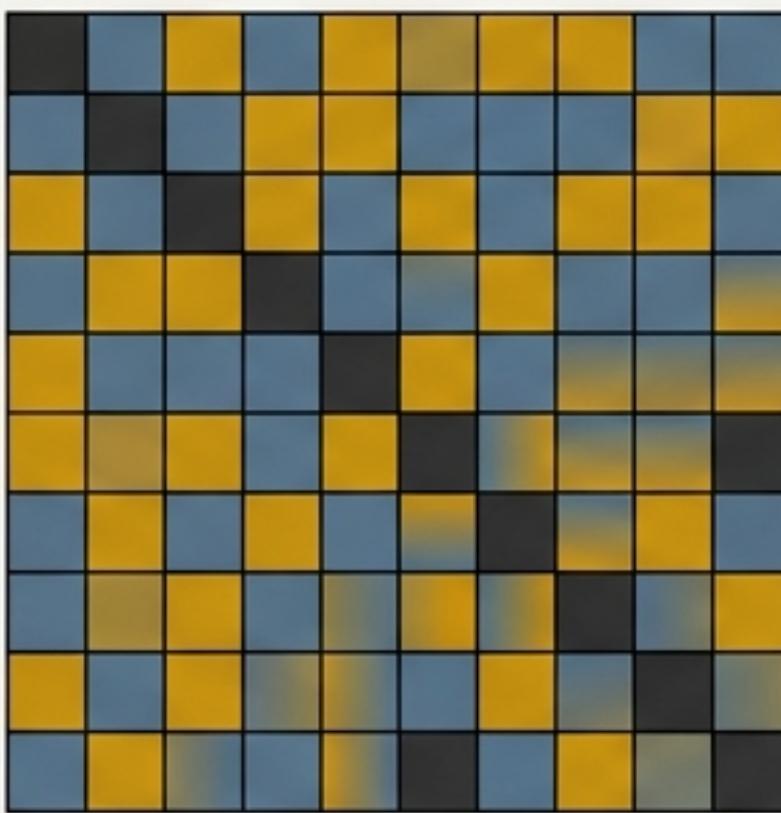


# High-dimensional data is rich with insight, but its complexity hinders interpretation.

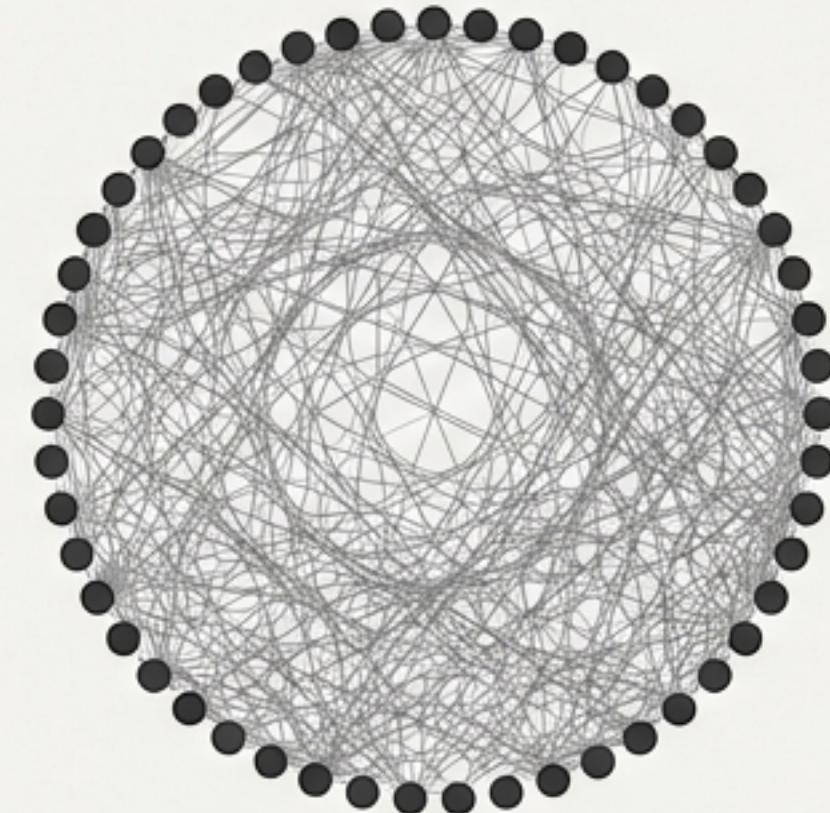
Domains like genomics, neuroscience, and topic modeling produce data where sources of variation are highly correlated and overlapping. This “entanglement” makes it difficult to disentangle meaningful latent factors.

**We need methods to find interpretable, parts-based representations in these complex datasets.**

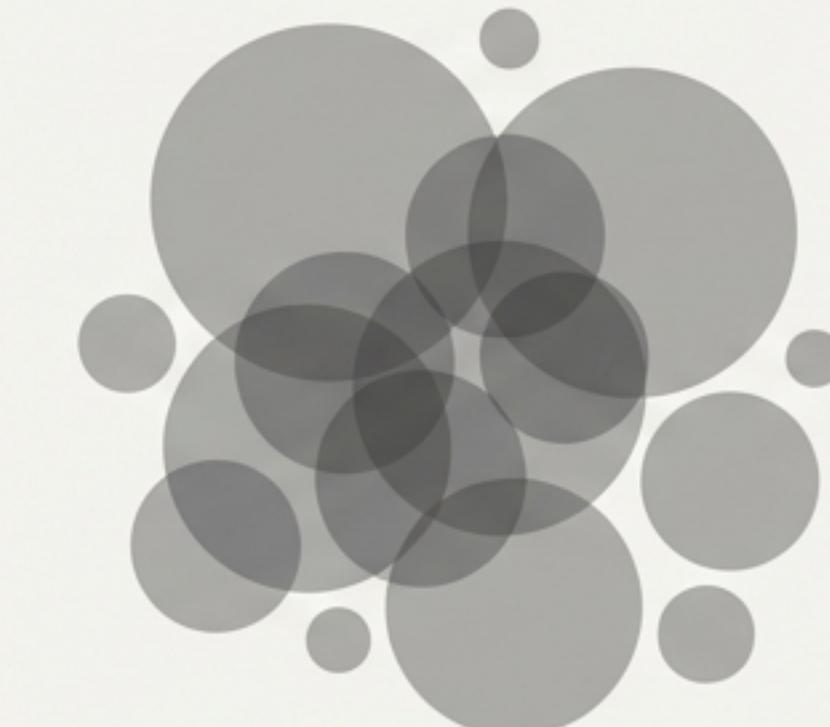
**Genomics**



**Neuroscience**

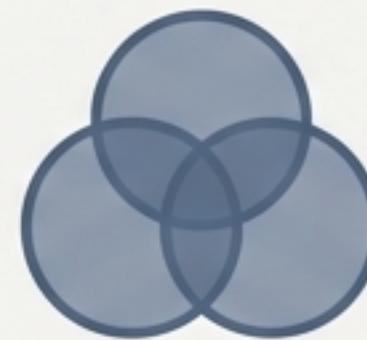


**Topic Modeling**



# Today's methods force a trade-off between model flexibility and interpretability.

FLEXIBILITY



**Powerful but Entangled**

PCA, NMF

These methods offer powerful, low-rank approximations but often produce dense, overlapping factors with rotational ambiguity, making them hard to interpret.

INTERPRETABILITY

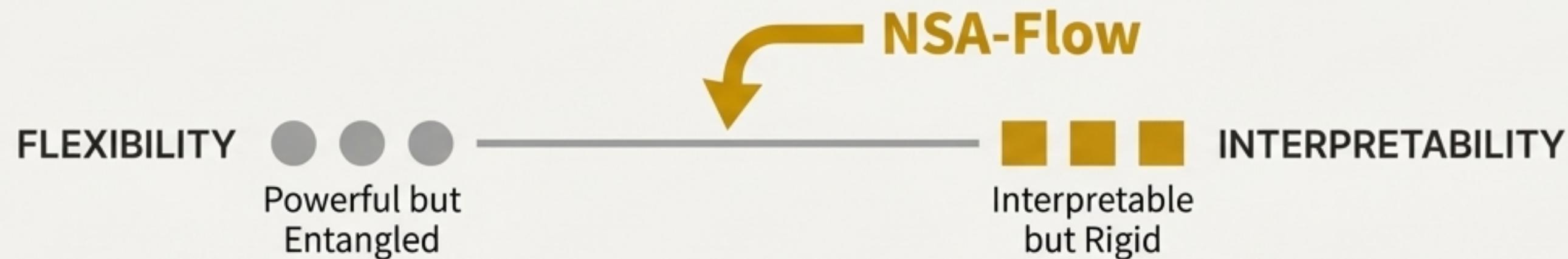


**Interpretable but Rigid**

Orthogonal NMF (ONMF)

Strict orthogonality improves sparsity and identifiability, creating clean factors. However, this rigidity can sacrifice fidelity to the noisy, heterogeneous nature of real-world data.

# NSA-Flow bridges this gap with continuous, tunable control over interpretability.



NSA-Flow is a variational optimization algorithm that finds a non-negative matrix  $\mathbf{Y}$  to approximate a target  $\mathbf{X}_0$ . It balances fidelity, column orthogonality, and non-negativity through a single, intuitive parameter,  $w$ .

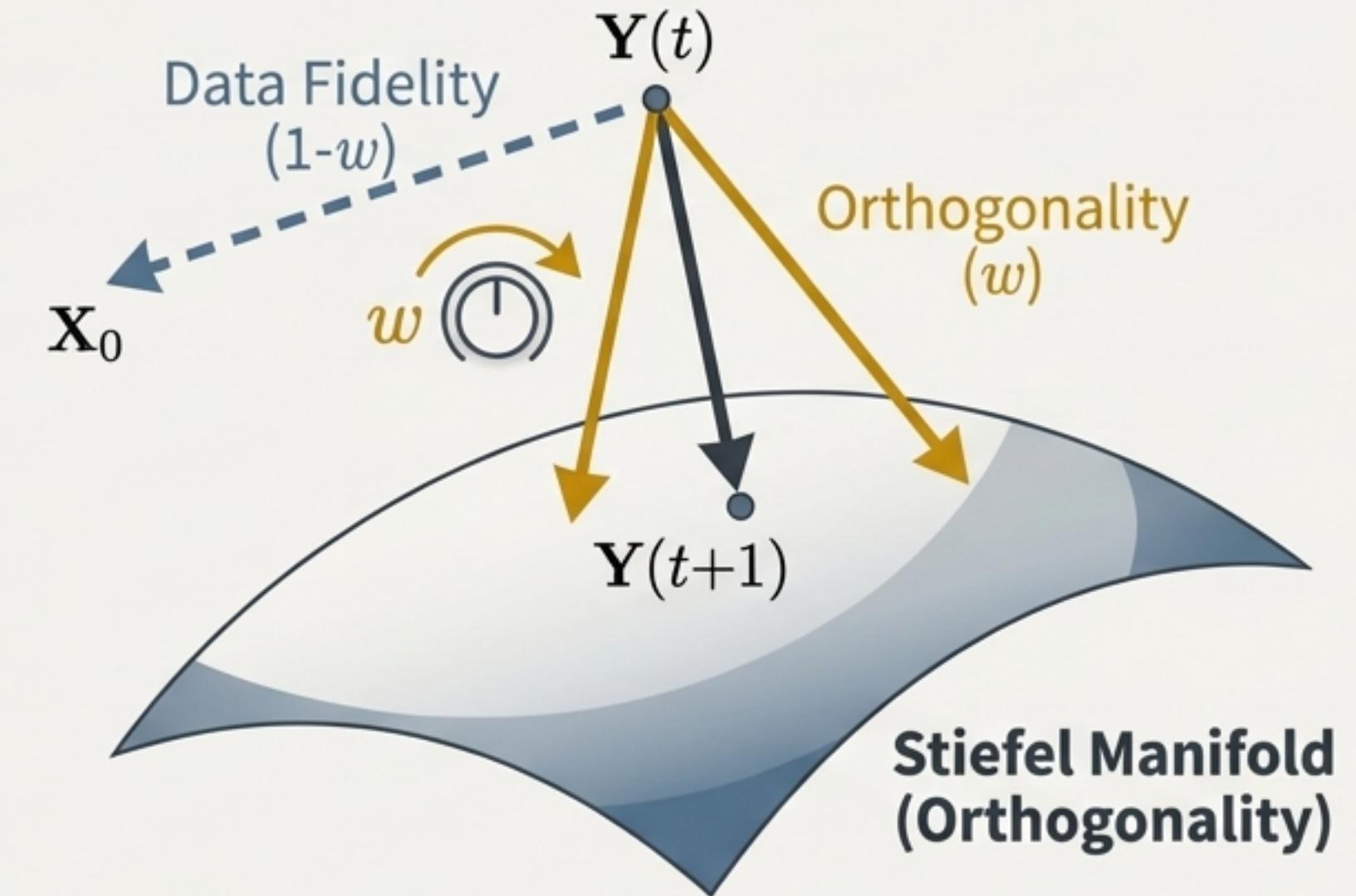
NSA-Flow employs global soft orthogonality constraints to promote disjoint support across columns and foster interpretable bases.

# The core mechanism is a ‘soft-retraction flow’ that guides solutions toward orthogonality.

**Goal:** Find a matrix  $Y$  that is close to the target data  $X_0$  while also being non-negative and having nearly orthogonal columns.

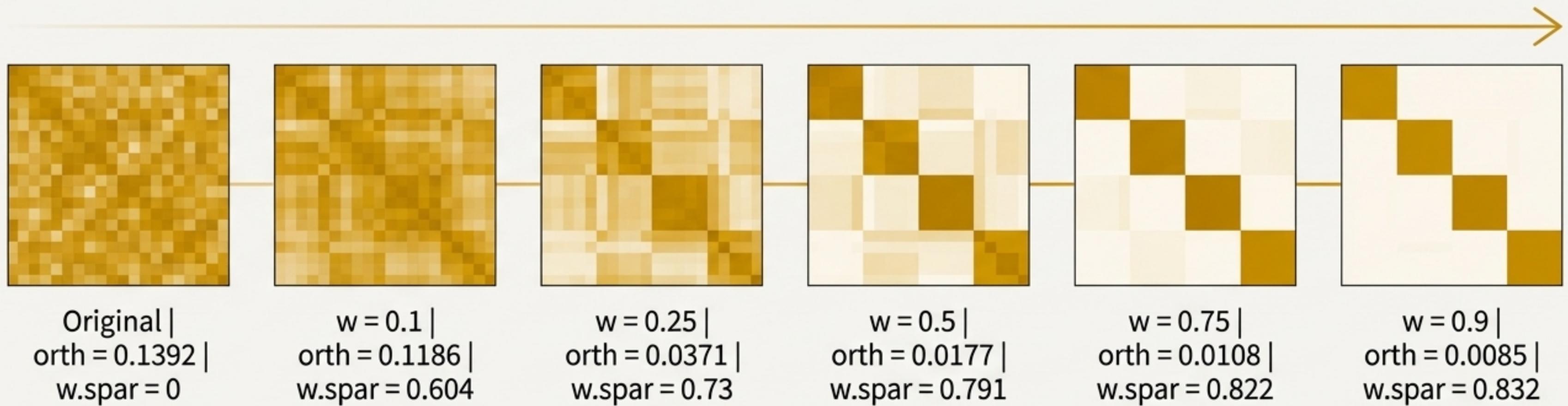
**Challenge:** A standard gradient step ignores the orthogonality constraint.

**NSA-Flow’s Approach:** Instead of a hard projection back to the Stiefel manifold (the space of orthogonal matrices), each update is a blend of the standard gradient step and a step toward the manifold.



$$\mathbf{Y}(t+1) = (1-w) * \mathbf{Y}_{\text{gradient\_step}} + w * \text{Retract}(\mathbf{Y}_{\text{gradient\_step}})$$

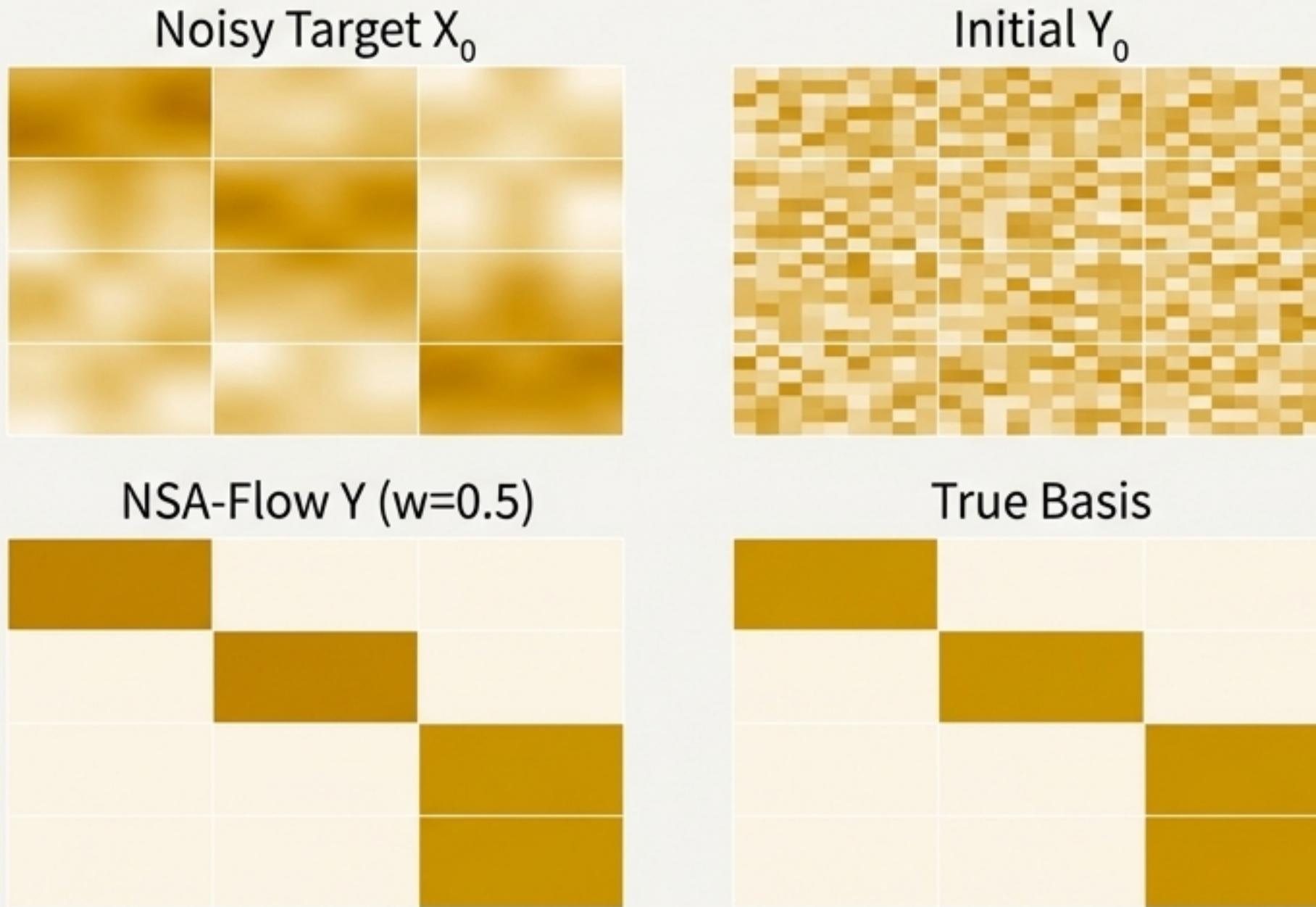
# The `w` parameter provides a dial to tune from blurry and dense to sharp and sparse.



Increasing `w` strengthens orthogonality, which indirectly induces sparsity by concentrating non-zero entries into disjoint patterns.

# On a mixed-signal toy problem, NSA-Flow flawlessly recovers the true, clean basis.

We start with a  $4 \times 3$  matrix  $X_0$  containing noisy, mixed versions of three non-negative, orthogonal patterns.

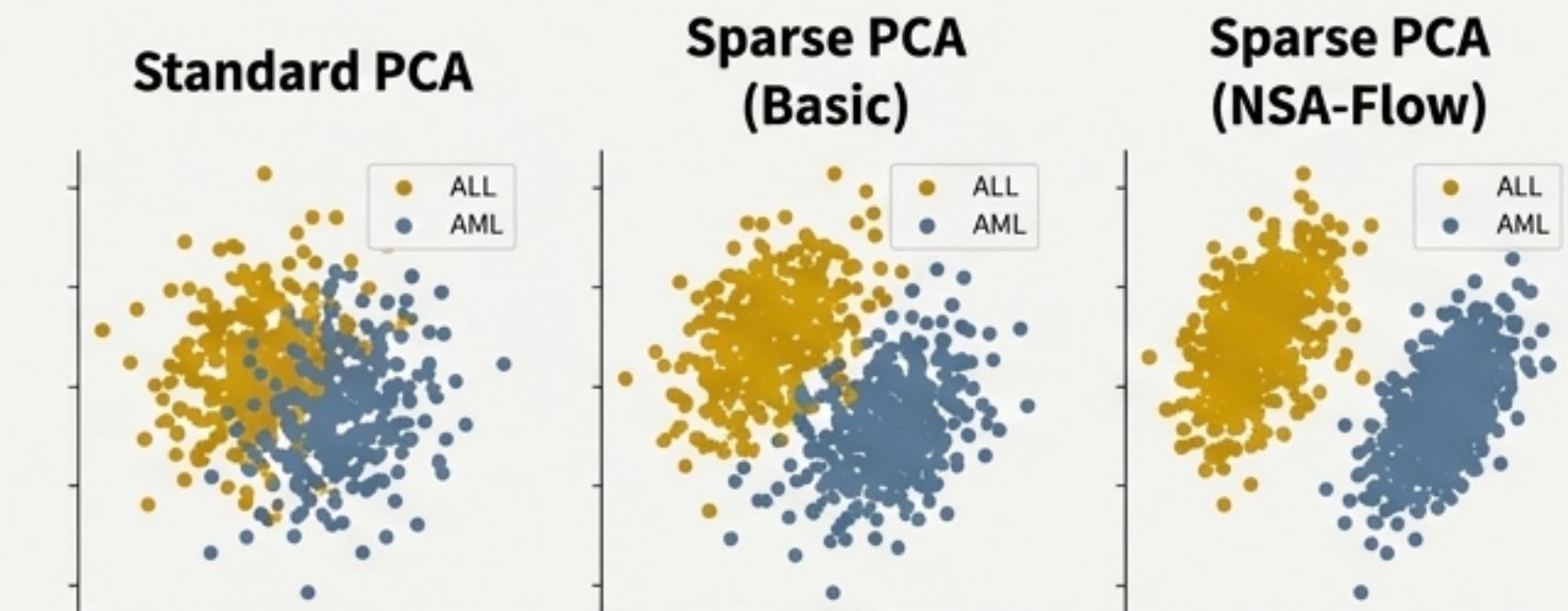


NSA-Flow successfully extracts the interpretable, disjoint components from the noisy data, demonstrating its core capability.

# Application 1: NSA-Flow enhances cancer subtype classification in leukemia data.

Using the classic Golub et al. gene expression dataset to classify leukemia types (ALL vs. AML), we integrate NSA-Flow into a Sparse PCA (SPCA) framework.

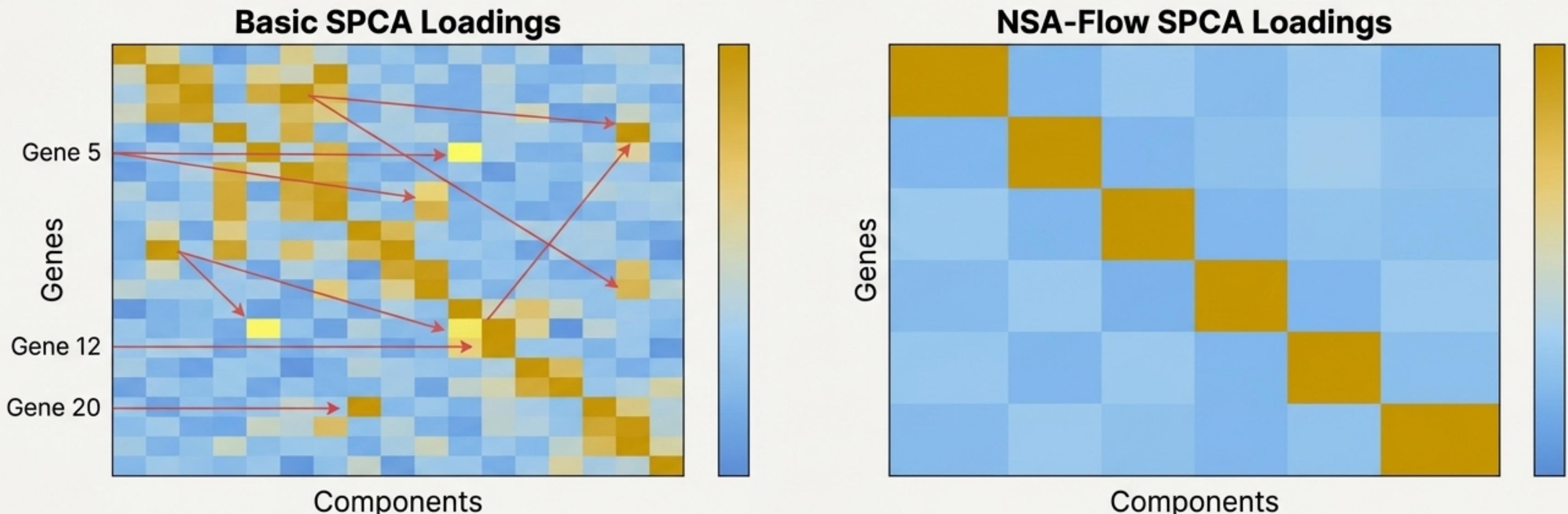
Method	CV Accuracy	Sparsity
Standard PCA	81.9%	0.0%
Sparse PCA (Basic)	86.4%	80.0%
<b>Sparse PCA (NSA-Flow)</b>	<b>88.3%</b>	<b>70.4%</b>



**NSA-Flow improves classification accuracy while generating a highly sparse, interpretable model.**

# The result is a set of clean, disjoint gene signatures, unlike the overlapping features from standard methods.

Standard sparse methods can identify important genes, but they often appear across multiple components, complicating biological interpretation. NSA-Flow's soft-orthogonality produces components with unique, non-overlapping gene signatures.



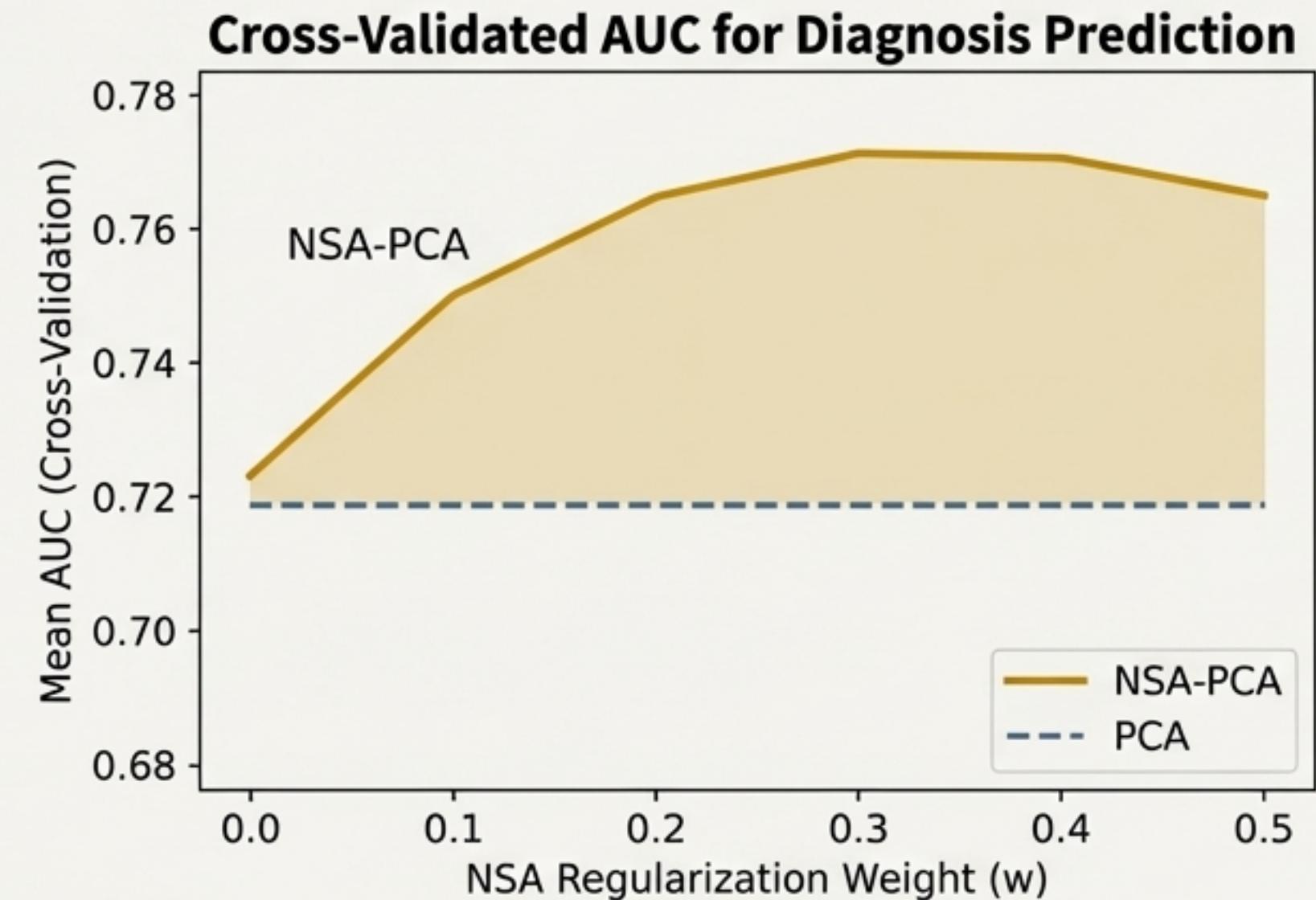
NSA-Flow delivers a clearer identification of relevant biological features by providing (soft) orthogonal and unsigned feature maps.

# Application 2: NSA-Flow uncovers brain networks more predictive of Alzheimer's diagnosis.

We refined PCA components from ADNI cortical thickness data. The goal was to find sparse brain networks that better predict clinical diagnosis (Control vs. MCI vs. AD).

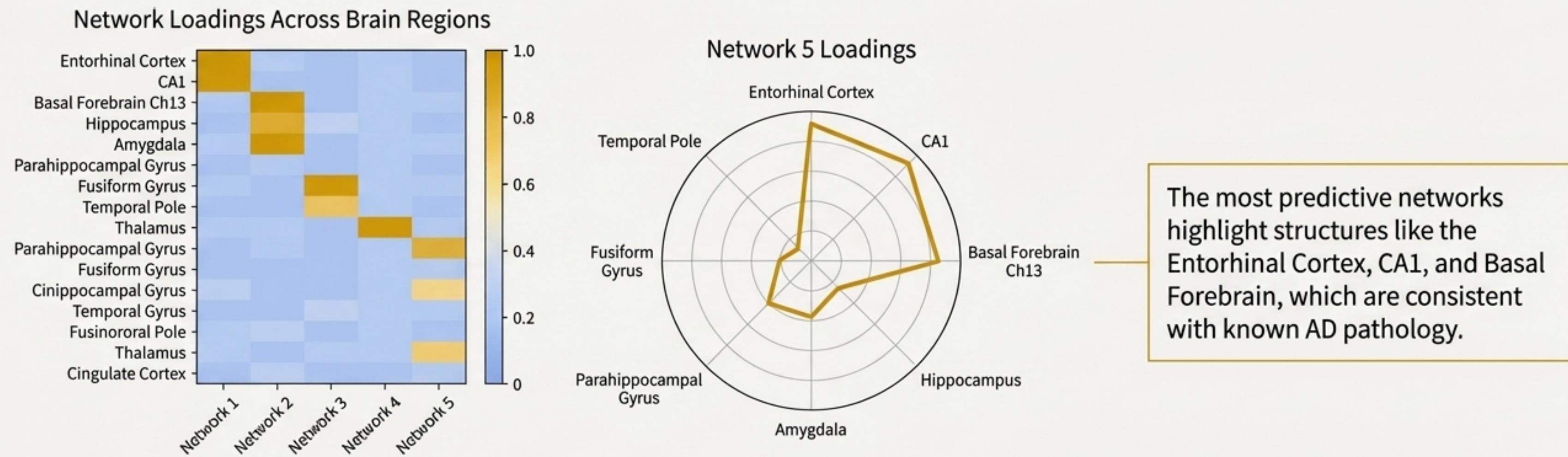
**+4.6%**

absolute improvement in classification AUC over PCA. This gain was statistically stable and most significant in the challenging CN vs. MCI task, **crucial for early detection**.



# The sparse networks highlight specific brain regions known to be involved in Alzheimer's pathology.

The improved predictive power comes from creating more biologically plausible networks. NSA-Flow's components emphasize key regions while filtering out noise that can dominate standard PCA.



NSA-Flow doesn't just improve a number; it extracts more meaningful, disease-relevant features.

# The NSA-Flow Advantage



## Intuitive Control

Provides a single parameter ( $w$ ) to navigate the complex trade-off between data fidelity and interpretability, allowing users to dial in the desired level of sparsity and decorrelation.



## Superior Performance

Delivers sparser, more decorrelated features that consistently maintain or improve predictive power in classification and regression tasks compared to baseline methods.



## A General-Purpose Tool

Functions as a modular, ‘soft projection’ operator that integrates seamlessly into existing ML pipelines (like Sparse PCA) to enhance interpretability across diverse domains.

# Acknowledging Limitations and Charting Future Directions

## Current Challenges



### Scalability

The  $O(k^3)$  cost of matrix inversions in the retraction step can be a bottleneck for problems with a large number of components ( $k$ ).



### Nonconvexity

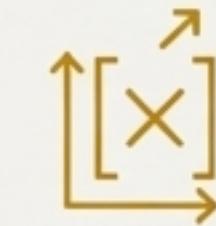
As a nonconvex optimization problem, the algorithm can be sensitive to initialization and may converge to local optima, particularly in high-noise scenarios.

## Future Work



### Sparse & Stochastic Variants

Developing minibatch updates to scale to massive datasets.



### Second-Order Methods

Incorporating Hessian information to accelerate convergence.



### Domain-Specific Adaptations

Tailoring the framework for multi-modal data fusion or graph-structured inputs.

# NSA-Flow is open-source and ready to use.

## Package Information

Python Package: `nsa-flow`  
R Wrapper: Available in `ANTsR`

## Installation

```
pip install nsa-flow
```

## Resources

GitHub Repository: `github.com/stnav/nst\_flow`  
Full Research Paper: `arxiv.org/abs/2511.06425`

