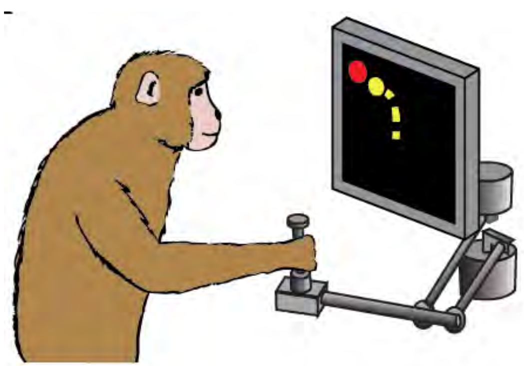
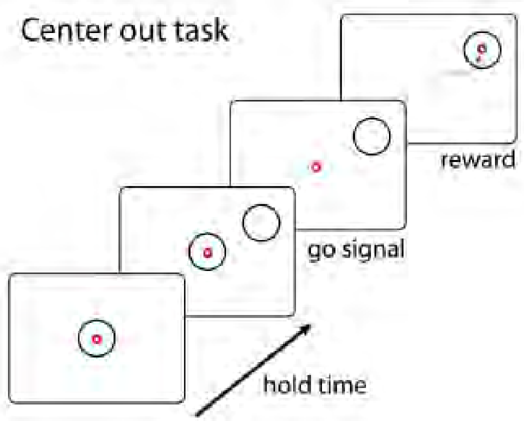
Linear and nonlinear dimensionality reduction: applications to neural data

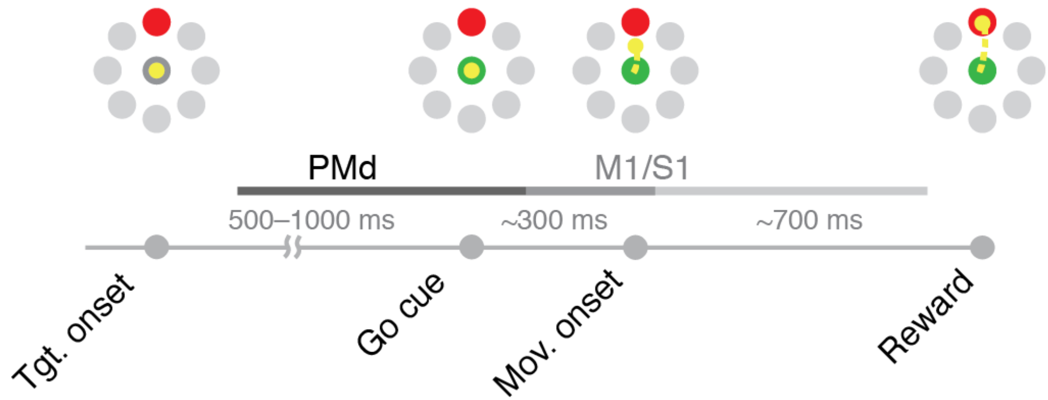
Sara A. Solla
Northwestern University

A simple motor task: center-out reaches

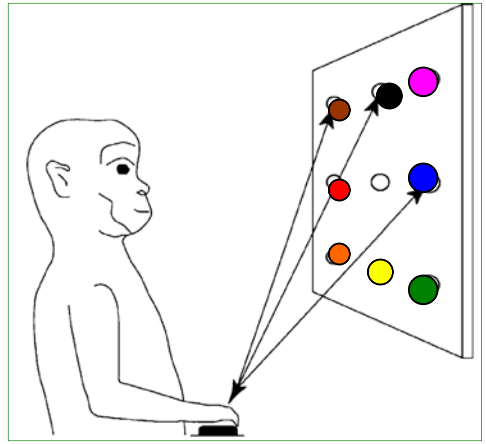
Instructed delay center-out reaching task

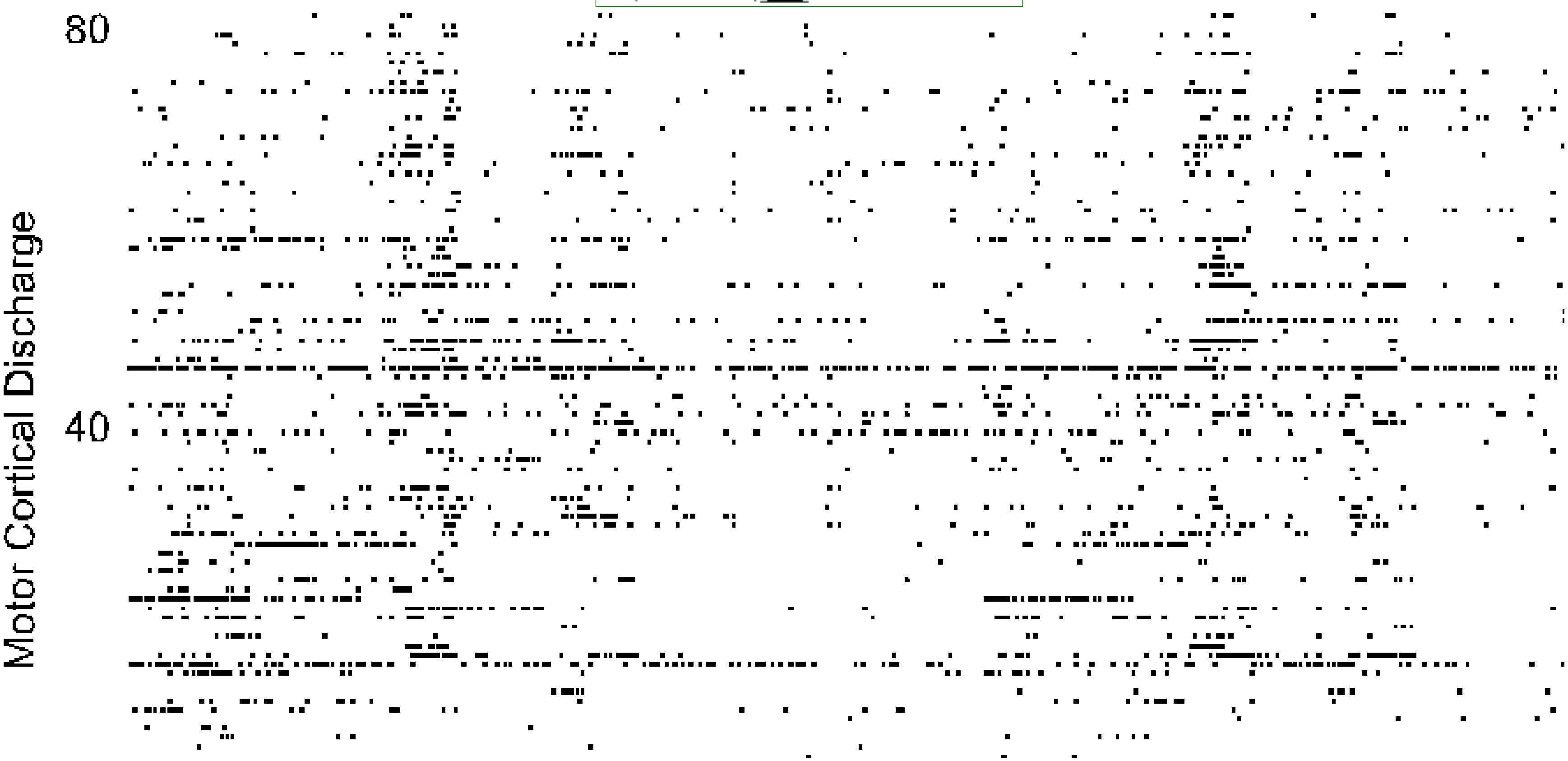




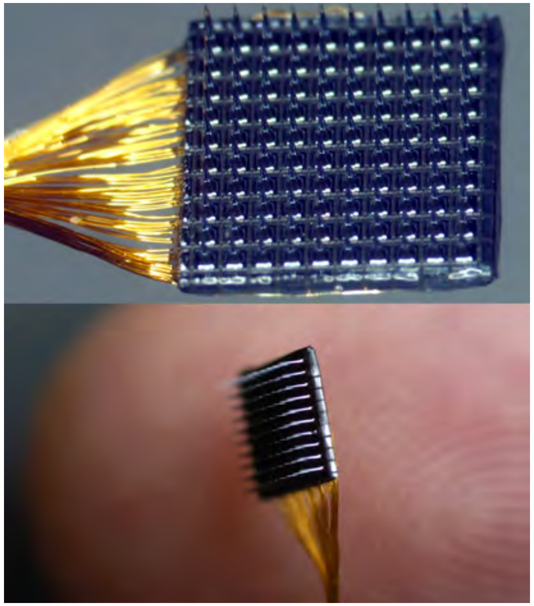


Neural recordings: center-out reaches



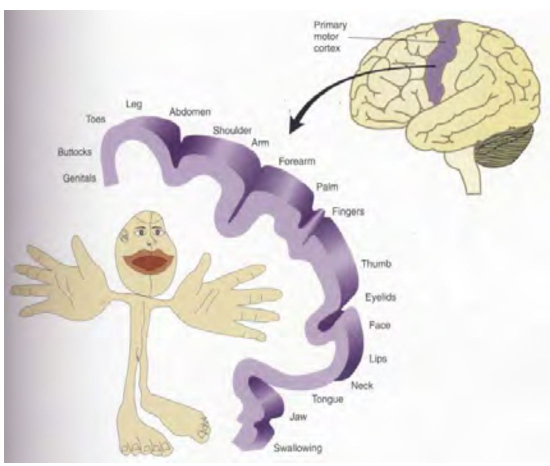


Neural recordings: population activity



$$X = \begin{bmatrix} n_1^t & n_1^{t+1} & \cdots & n_1^{t+T} \\ n_2^t & n_2^{t+1} & & n_2^{t+T} \\ \vdots & \vdots & & \vdots \\ n_N^t & n_N^{t+1} & \cdots & n_N^{t+T} \end{bmatrix}$$

Data matrix X has N rows and $(T+1)$ columns

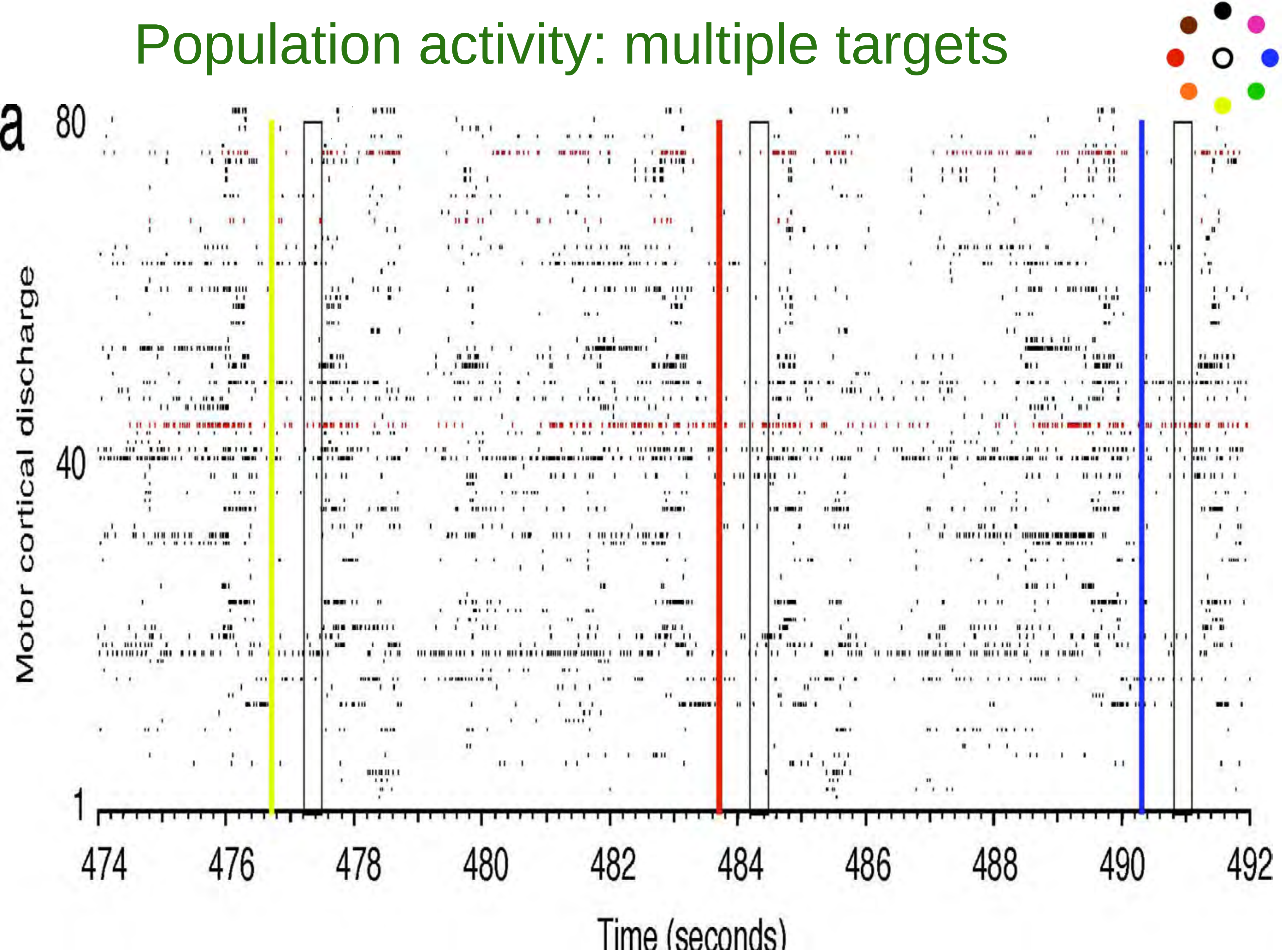


$N \approx 10^2$ for Multi-Electrode Arrays (MEAs)

$N \approx 10^3$ for Neuropixels

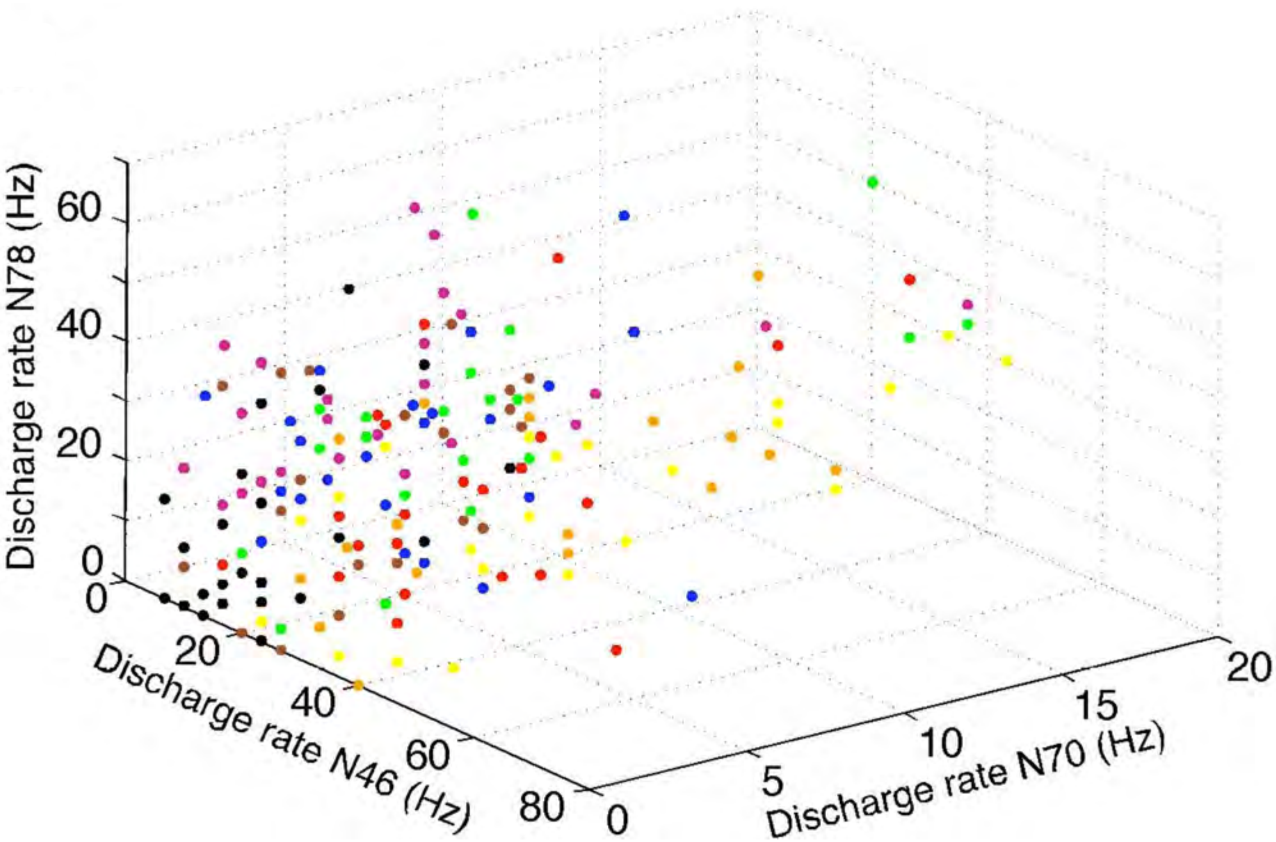
N is the number of recorded neurons

Population activity: multiple targets



Target-dependent population activity

$$\vec{f} = (f_1, f_2, \cdots, f_N) \qquad f_i = \frac{n_i}{\Delta}$$



Principal Components Analysis (PCA)

Consider data in the form of N-dimensional vectors. Here, the data is the N-dimensional vector of firing rates associated with each reach.

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_N \end{pmatrix}$$

Data for M reaches result in an N x M matrix:

$$\boldsymbol{X} = (\vec{x_1}|\vec{x_2}|\ldots|\vec{x_M})$$

Estimate the mean firing rate of each neuron:

$$\hat{\mu}_i = \frac{1}{M}\sum_{k=1}^{M} x_{ik} \qquad 1 \leq i \leq N$$

and subtract it from the corresponding row.

Principal Components Analysis (PCA)

Next, estimate the covariance of the data:

$$\hat{C} = \frac{1}{(M-1)} \; X \, X^T$$

$$\hat{C}_{ij} \; = \frac{1}{(M-1)} \sum_{k=1}^{M} (x_{ik} - \hat{\mu}_i)(x_{jk} - \hat{\mu}_j)$$

$$1 \leq i, j \leq N$$

The diagonalization of the covariance matrix yields eigenvectors and eigenvalues: the principal components

$$\widehat{C} \, \vec{u}_\nu = \lambda_\nu \, \vec{u}_\nu$$

$$1 \leq \nu \leq N$$

Principal Components Analysis (PCA): relation to Singular Value Decomposition (SVD)

Consider the singular value decomposition of the data matrix X:

$$X = U\,\Sigma\,V^T$$

The columns of the N x N orthonormal matrix U provide a basis for the neural space.

The columns of the M x M orthonormal matrix V provide a basis for the space of samples.

The N x M matrix sigma Σ consists of an N x N diagonal block and and N by $(M - N)$ block of zeros.

Principal Components Analysis (PCA): relation to Singular Value Decomposition (SVD)

Given the singular value decomposition of the data matrix X:

$$X = U\,\Sigma\,V^T$$

$$\begin{aligned} X\,X^T &= (U\,\Sigma\,V^T)\,(V\,\Sigma^T U^T\,) \\ &= U\,(\Sigma\,\Sigma^T)\,U^T \end{aligned}$$

$$\hat{C} = \frac{1}{(M-1)}\;X\,X^T = U\Lambda\,U^T$$

$$\Lambda = \frac{1}{(M-1)}\;\Sigma\,\Sigma^T$$

Principal Components Analysis (PCA): dimensionality reduction

$$\Lambda = \frac{1}{(M-1)}\ \Sigma\ \Sigma^T$$

$$\Lambda = \begin{bmatrix} \lambda_1 & \cdots & 0 & \cdots & 0 \\ 0 & \cdots & \lambda_K & \dots & 0 \\ 0 & \cdots & 0 & \cdots & \lambda_N \end{bmatrix}$$

with $\lambda_1 \geq \cdots \geq \lambda_K \geq \cdots \geq \lambda_N$

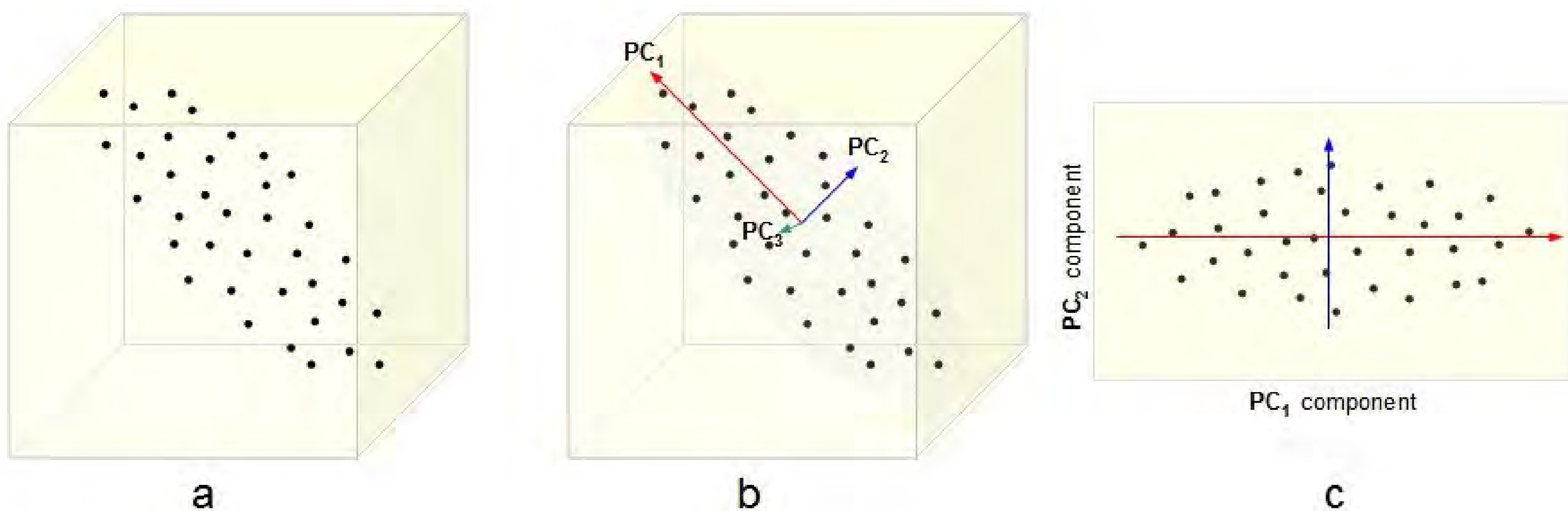
Dimensionality reduction: keep only the K leading eigenvalues

$$\hat{C} = U \Lambda\ U^T$$

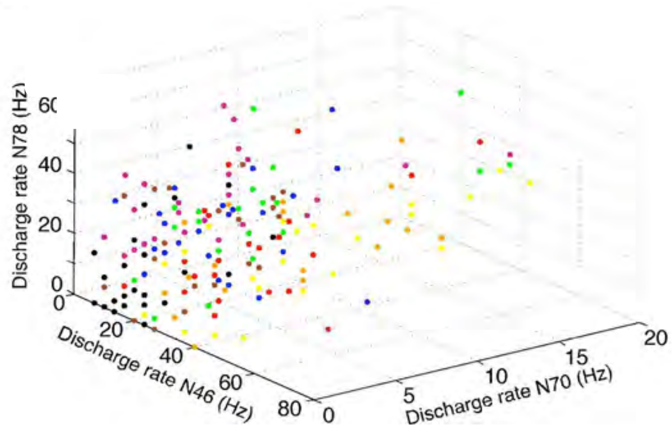
$$\hat{C} = \sum_{\mu=1}^{N} \lambda_\mu\ \vec{u}_\mu\ \vec{u}_\mu^T \qquad \Longrightarrow \qquad \hat{C} = \sum_{\mu=1}^{K} \lambda_\mu\ \vec{u}_\mu\ \vec{u}_\mu^T$$

Principal Components Analysis (PCA): dimensionality reduction

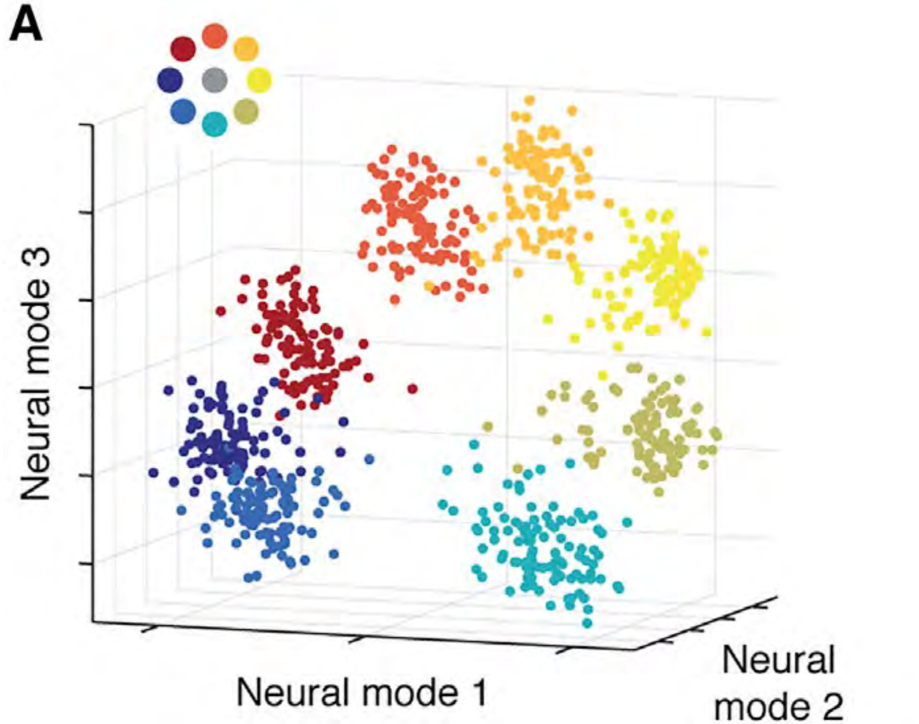
$$\hat{C} = \sum_{\mu=1}^{N} \lambda_\mu \ \vec{u}_\mu \ \vec{u}_\mu^T \qquad \Longrightarrow \qquad \hat{C} = \sum_{\mu=1}^{K} \lambda_\mu \ \vec{u}_\mu \ \vec{u}_\mu^T$$



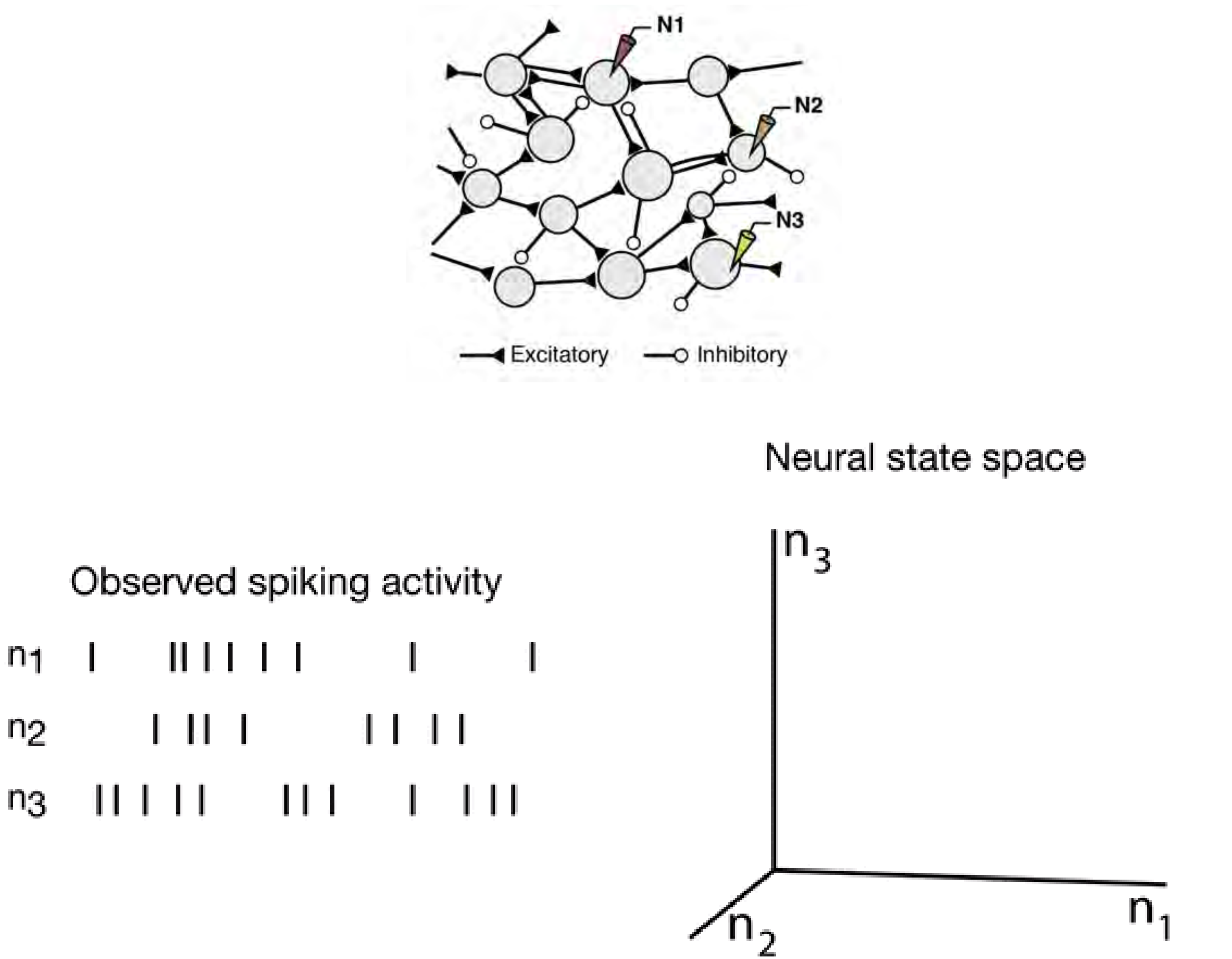
Target-dependent population activity



$$\vec{f} = (f_1, f_2, \ldots, f_N)$$

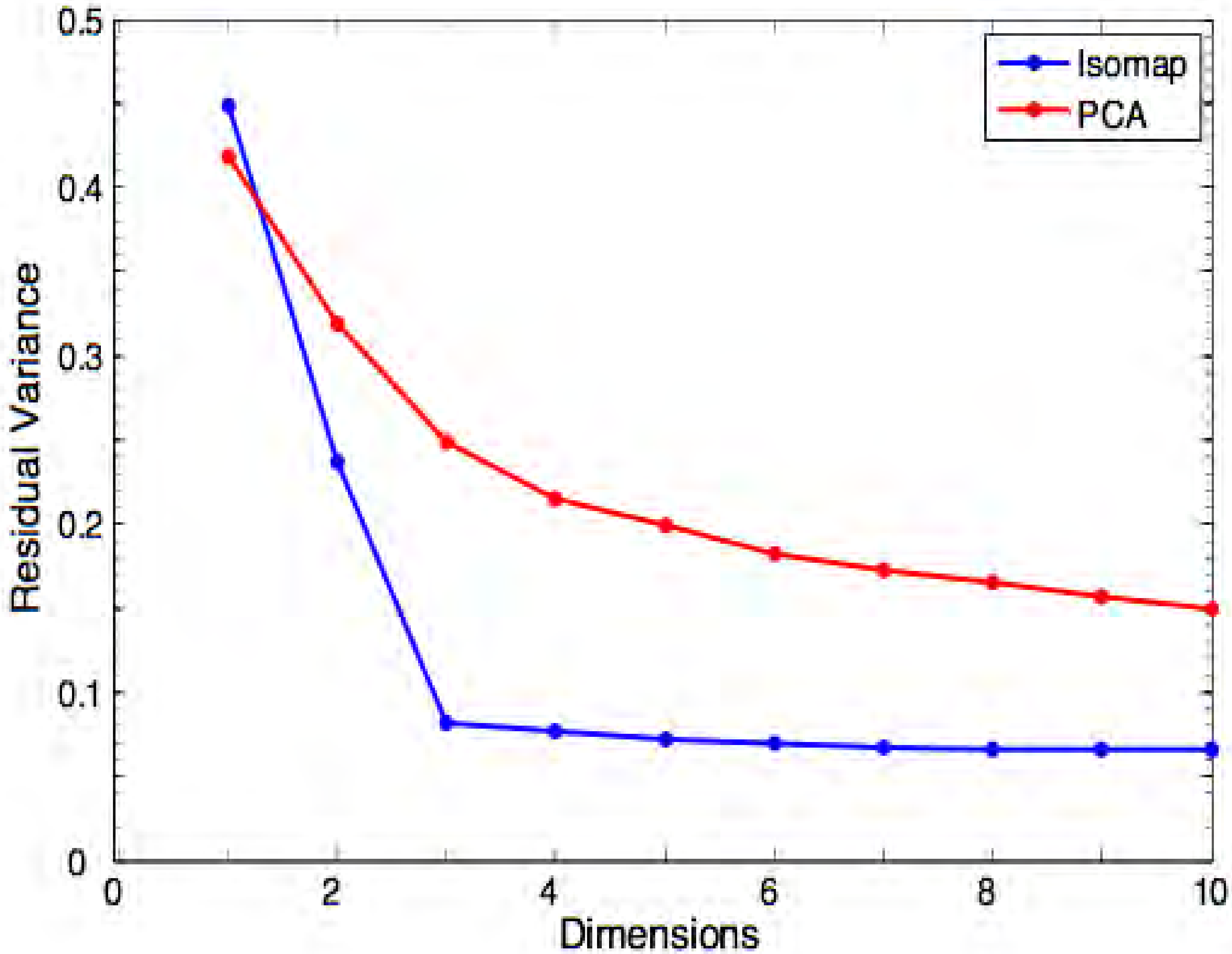


Population dynamics: the empirical neural space

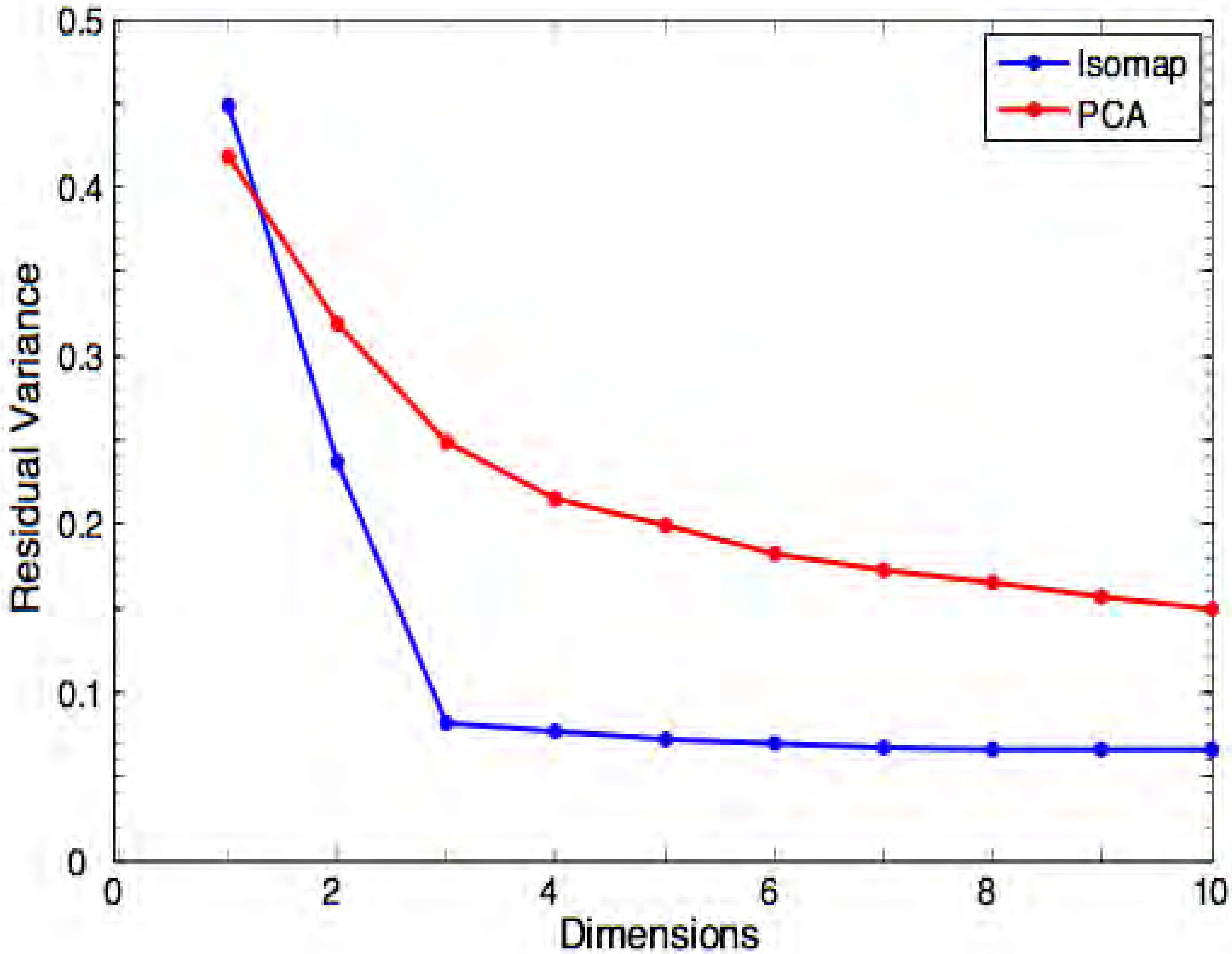


Cunningham & Yu, *Nat Neuro* 2014

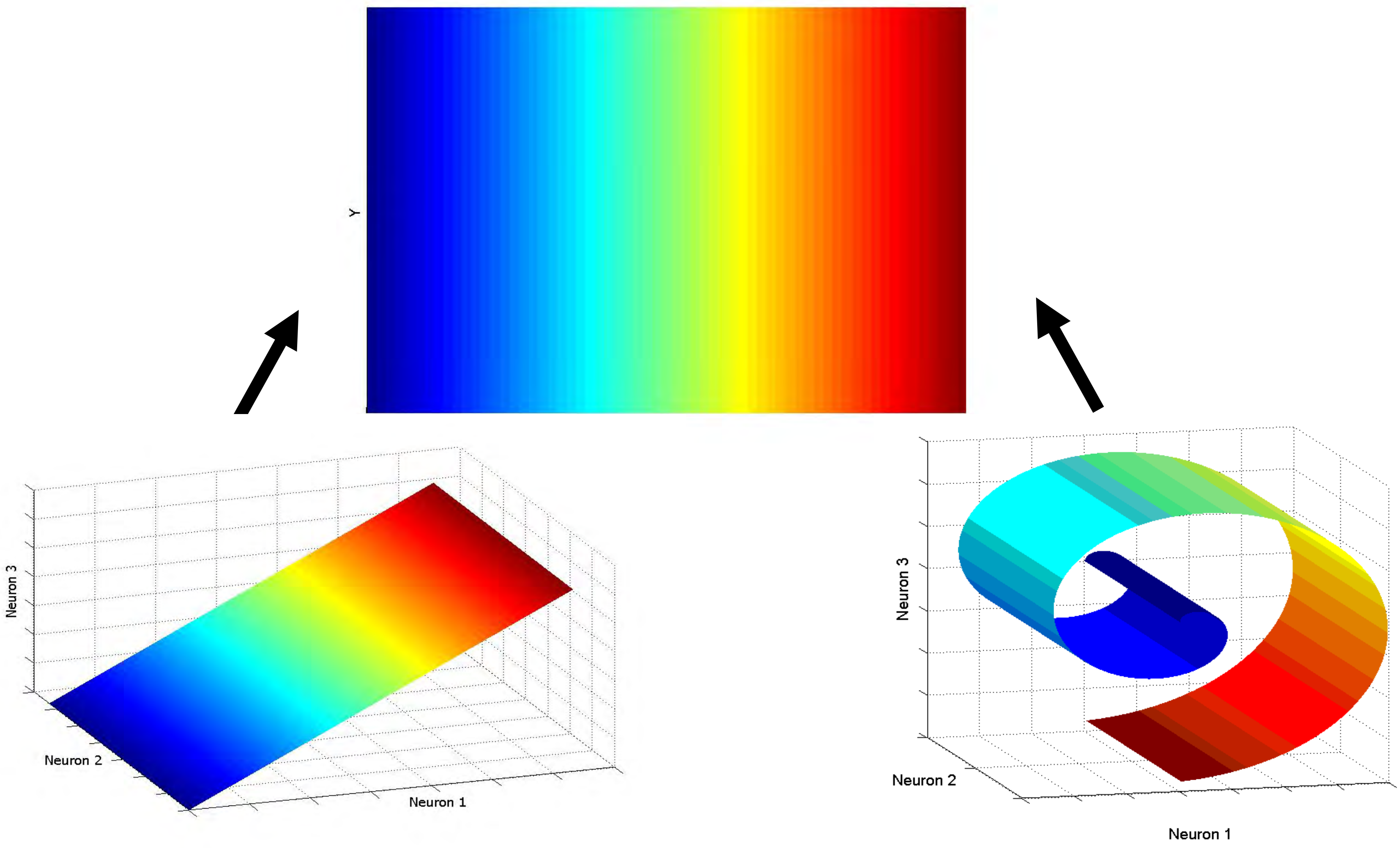
PCA eigenvalues



Isomap eigenvalues



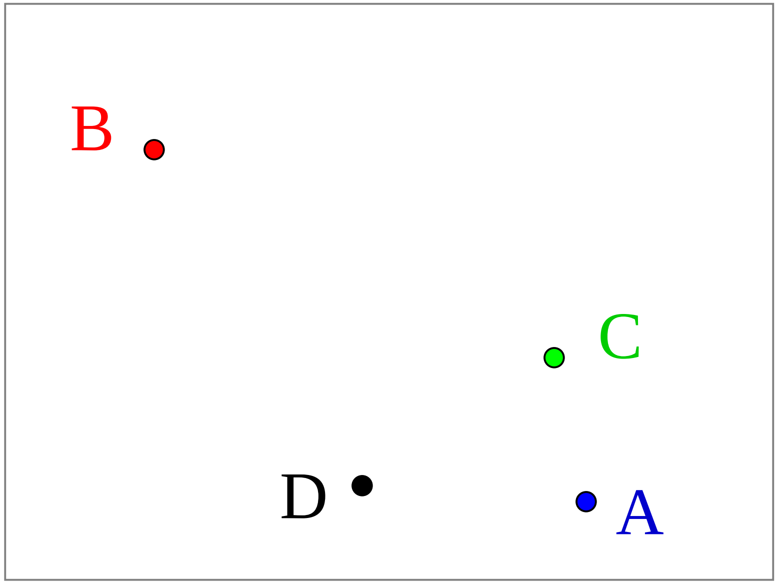
ISOMAP
nonlinear dimensionality reduction



Multidimensional scaling

Represent objects as points in a low dimensional space:
Euclidean distances between the corresponding points reproduce as well as possible an empirical matrix of distances or dissimilarities.

	A	B	C	D
A	0	7	2	3
B	7	0	4.5	6
C	2	4.5	0	5
D	3	6	5	0



Multidimensional scaling

Consider data in the form of N-dimensional vectors. Here, the data is the N-dimensional vector of firing rates associated with each reach.

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_N \end{pmatrix}$$

Data for M reaches result in an N x M matrix:

$$\boldsymbol{X} = (\vec{x_1}|\vec{x_2}|\ldots|\vec{x_M})$$

If the matrix $\boldsymbol{X}$ is hidden from us, but we are given instead an M x M matrix $\boldsymbol{S}$ of squared distances between the points, can we reconstruct the matrix $\boldsymbol{X}$?

Multidimensional scaling

If the distances are Euclidean:

$$S_{ij} = d_{ij}^2 = (\vec{x}_i - \vec{x}_j)^T(\vec{x}_i - \vec{x}_j)$$

the scalar product between data points can be written as:

$$\vec{x}_i^T \vec{x}_j = -(1/2)(S_{ij} - ||\vec{x}_i||^2 - ||\vec{x}_j||^2)$$

In matrix form, $\vec{x}_i^T \vec{x}_j = (X^T X)_{ij}$

and $S_{ij} - ||\vec{x}_i||^2 - ||\vec{x}_j||^2 = (JSJ)_{ij}$

where J is the M x M centering matrix $J = I - (1/M)\, \mathbf{e}\, \mathbf{e}^T$

$$X^T X = -(1/2) JSJ$$

Multidimensional scaling

In matrix form: $$\boldsymbol{X}^T\boldsymbol{X} = -(1/2)\boldsymbol{JSJ}$$

From this equation the data matrix $\boldsymbol{X}$ can be easily obtained:

$$\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T \Longrightarrow \boldsymbol{X} = \boldsymbol{\Lambda}^{1/2}\boldsymbol{U}^T$$

- If the distance matrix to which this calculation is applied is based on Euclidean distances, this process allows us to recover the data matrix $\boldsymbol{X}$ from the distance matrix $\boldsymbol{S}$.
- A reduction of the dimensionality of the original data space follows from truncation of the number of eigenvalues from M to K, and the corresponding restriction in the number of eigenvectors used to reconstruct $\boldsymbol{X}$.
- It can be proved that this truncation is equivalent to PCA, which is based on the diagonalization of $\boldsymbol{X}\boldsymbol{X}^T$.

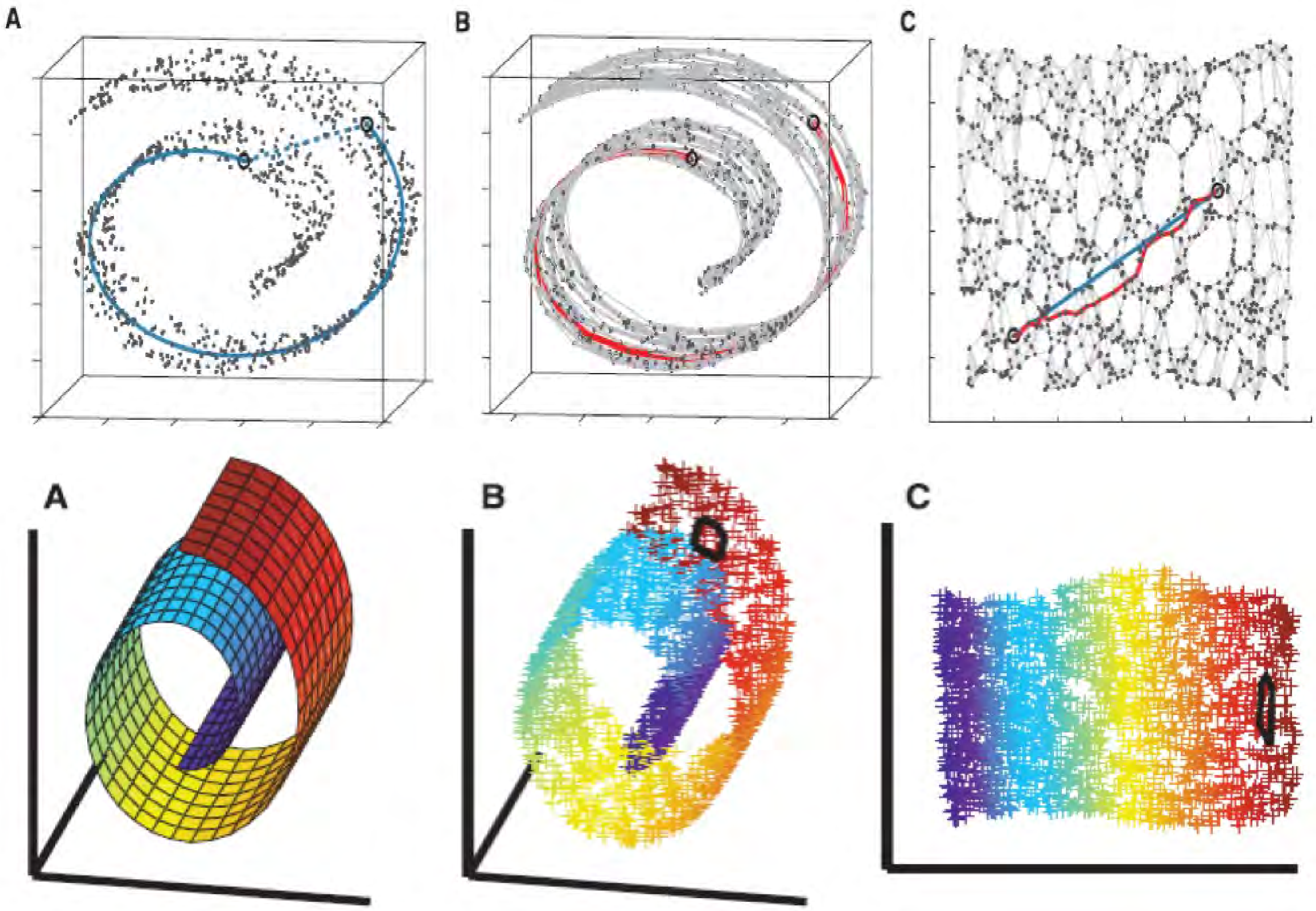
Multidimensional scaling

When applied to an arbitrary matrix ***S*** of ‘squared distances’, the method still implies defining an ‘inner product’ matrix Y through the centering operation: $Y = -(1/2)\, JSJ$, followed by the diagonalization of Y: $Y = U \Lambda U^T$ and the identification of the data matrix X as $X = \Lambda^{1/2} U^T$.

This procedure minimizes a cost function E that measures the Frobenius norm of the difference between two matrices: the original matrix Y and the inner product matrix $X^T X$ obtained from the Euclidean representation of the data:

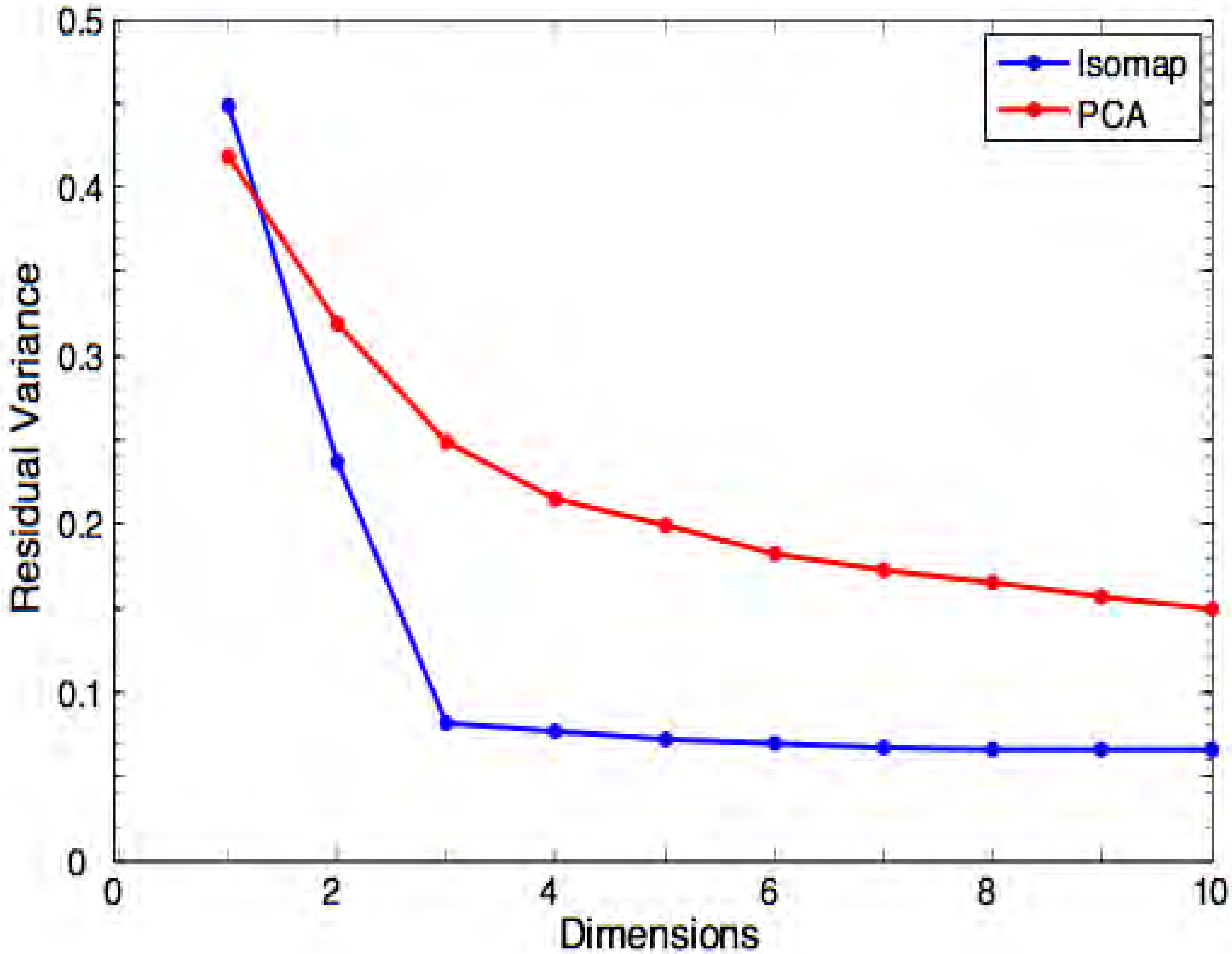
$$E(X) = \| X^T X - Y \|_F$$

ISOMAP: nonlinear embedding

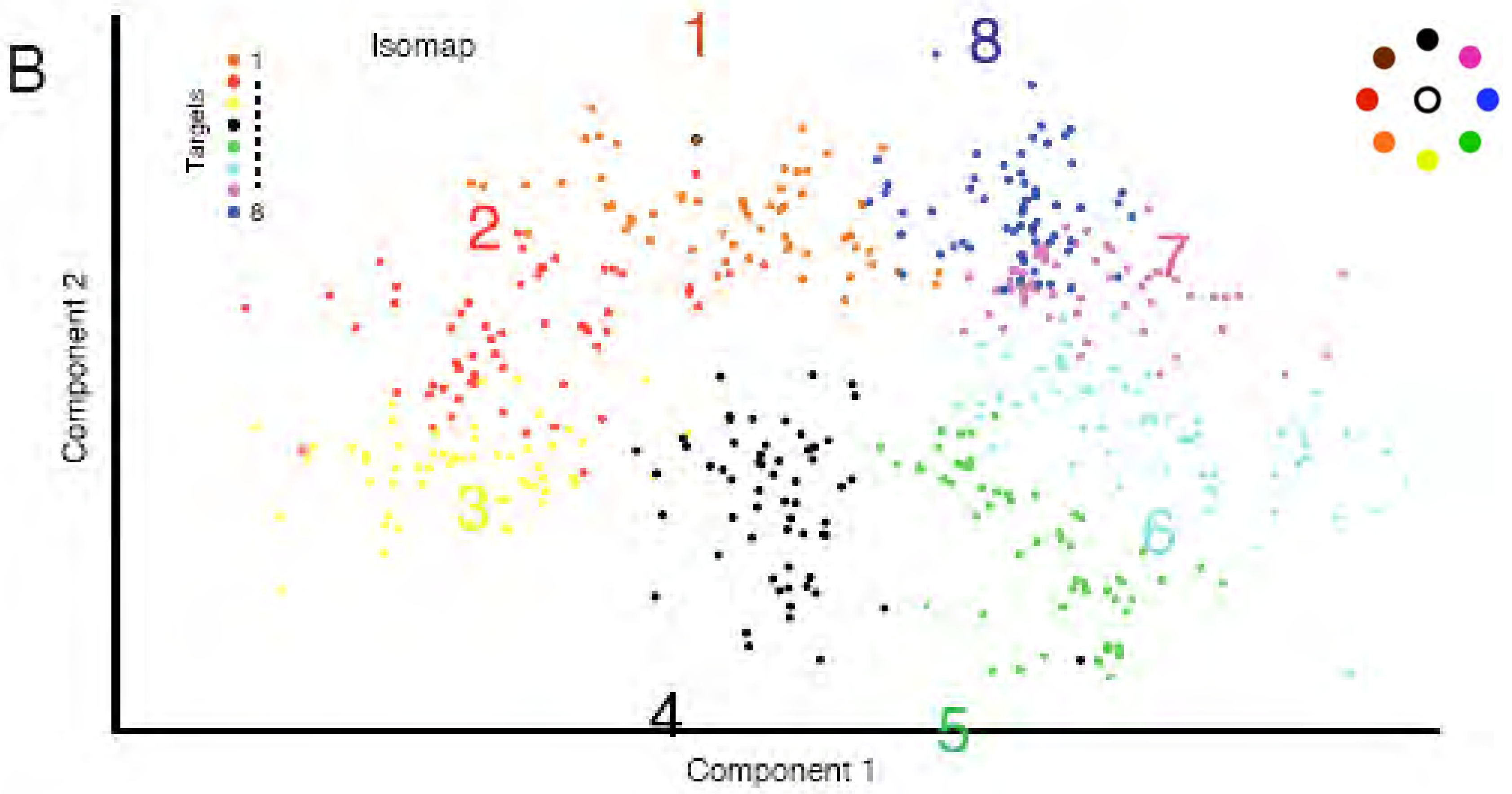


Tenenbaum, de Silva, Langford, *Science* (2000)

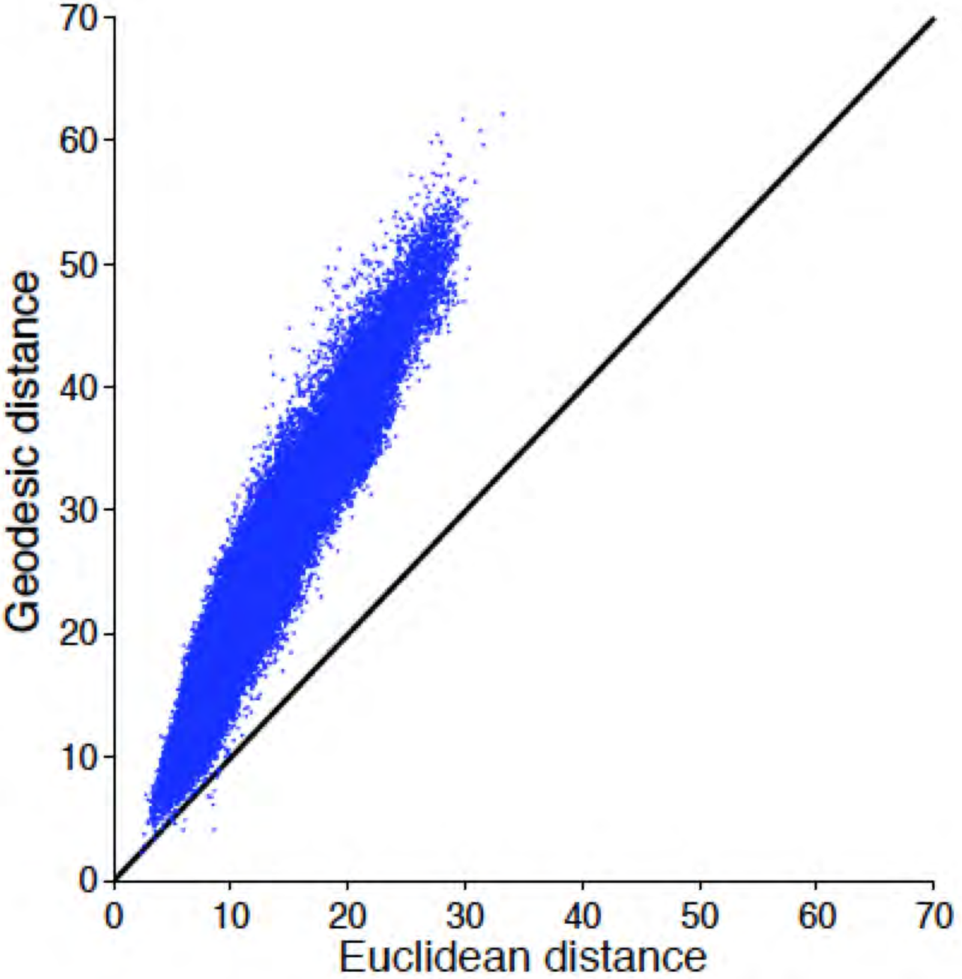
ISOMAP eigenvalues



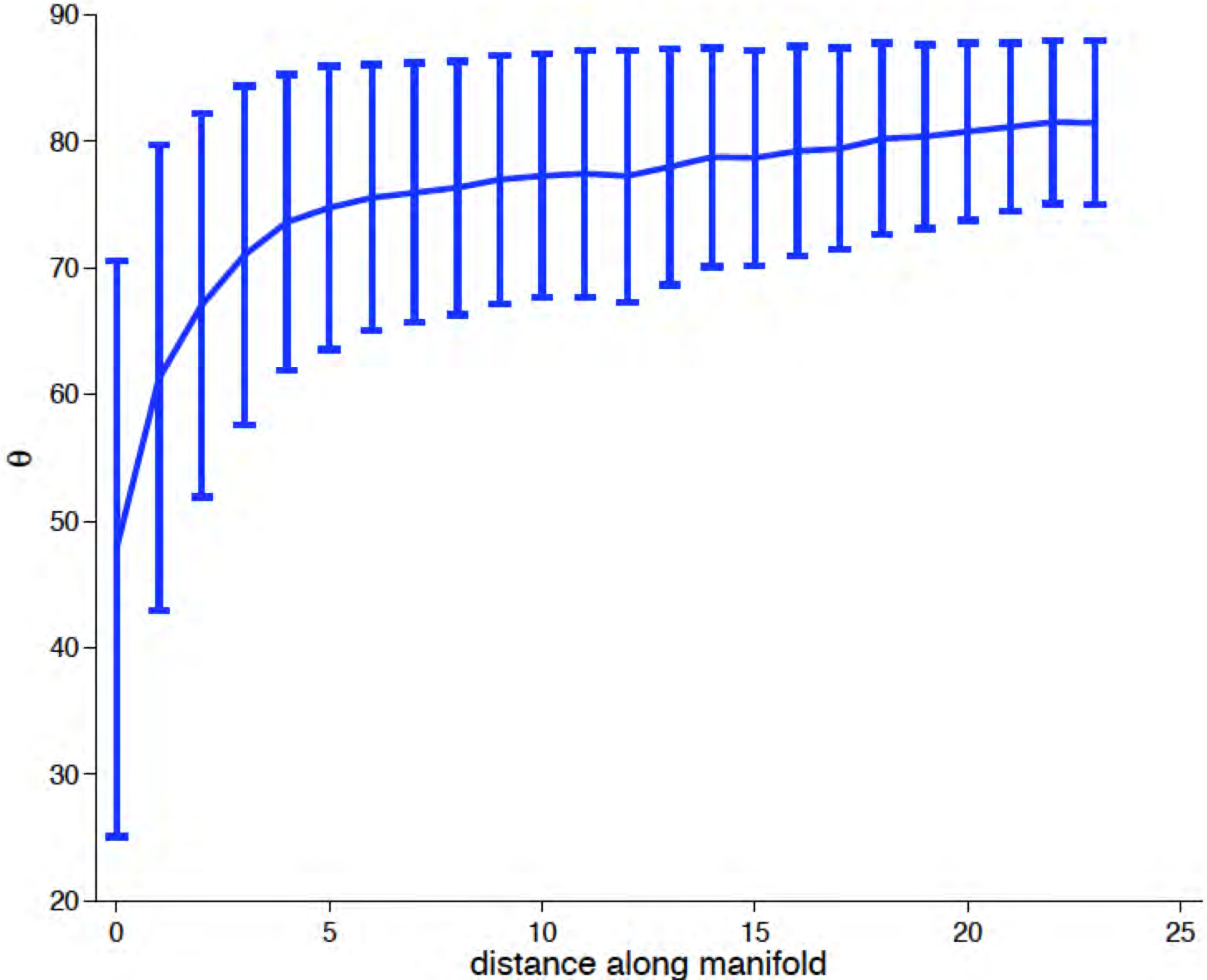
ISOMAP: two-dimensional projection



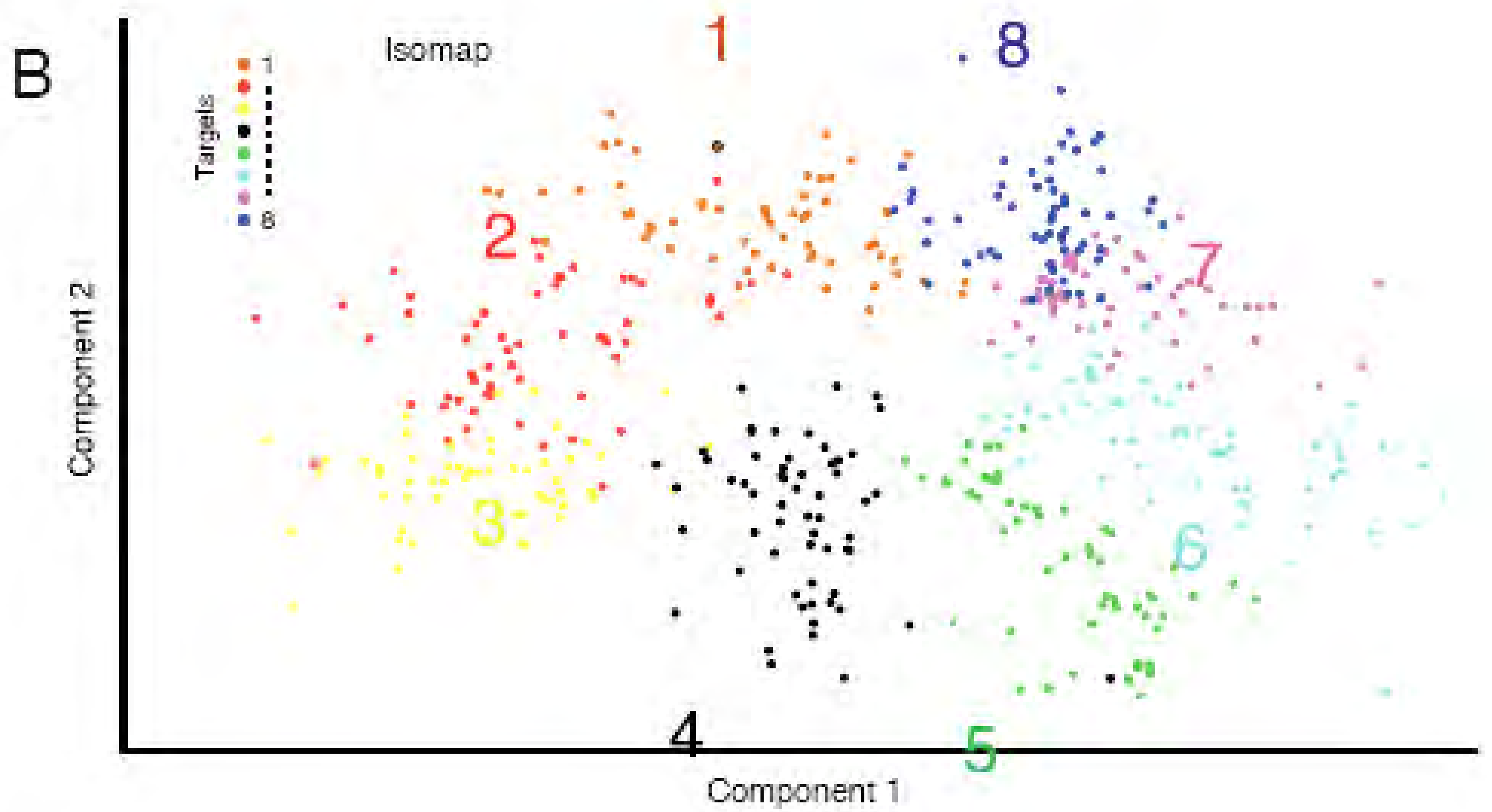
Euclidean vs Geodesic distances



Geodesics on a curved manifold



ISOMAP: two-dimensional projection



Endpoint predictions from M1 activity

- Simultaneous recordings of population activity can be analyzed with fundamentally different techniques than those used for single neurons.

- For limb reaches, the population activity defines a nonlinear low dimensional manifold whose intrinsic coordinates capture task-relevant dimensions.

- The curvature of the manifold is a network effect and arises from the interaction among neurons, while coordinates within the manifold are task-specific.