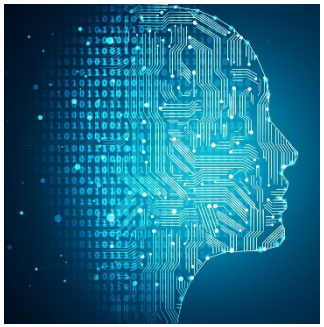
Principles of Machine Learning

Diffusion-based methods for learning and visualizing structure

Kevin Moon (kevin.moon@usu.edu)

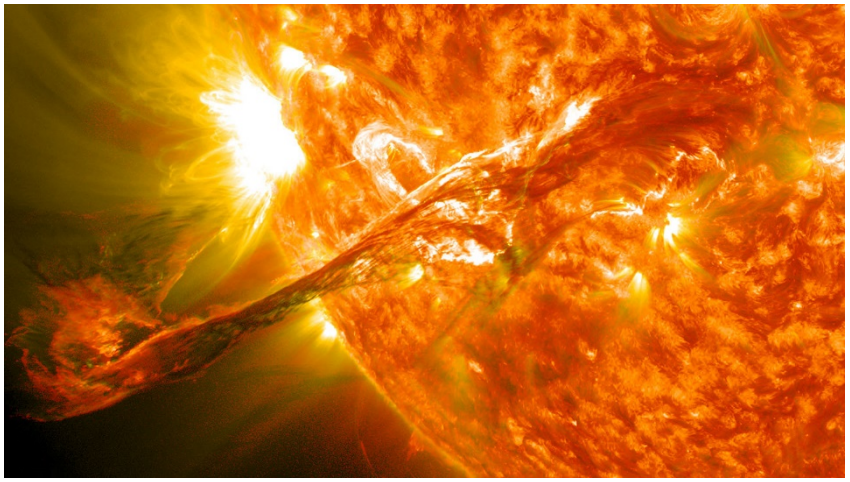
STAT 6910-001

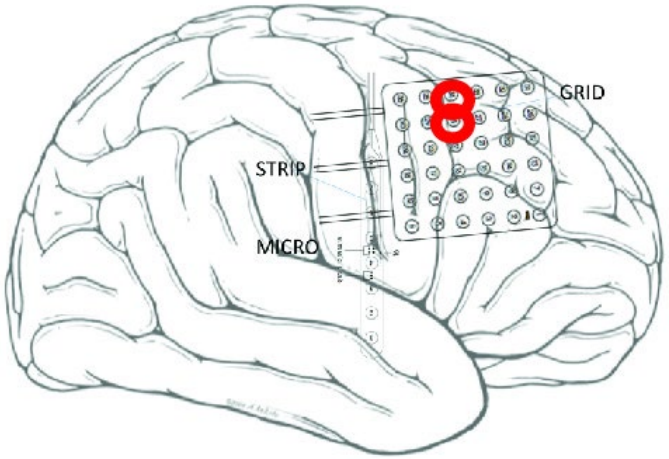


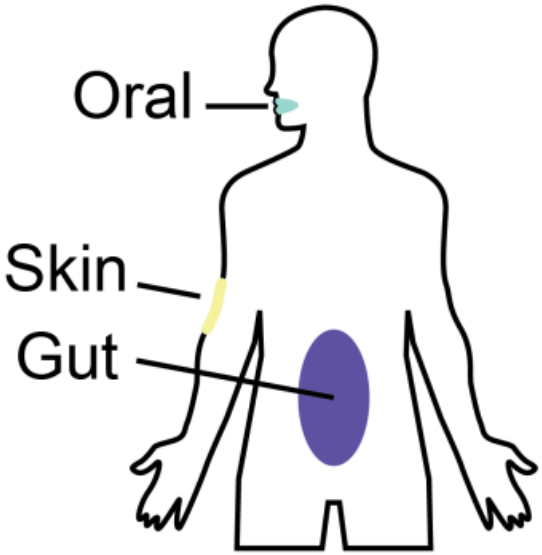
Outline

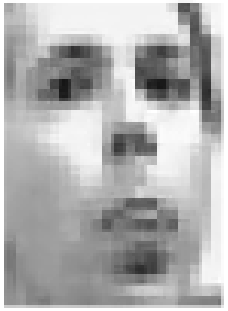
1. Motivation
2. Data Visualization
3. PHATE
4. MAGIC and Data Imputation

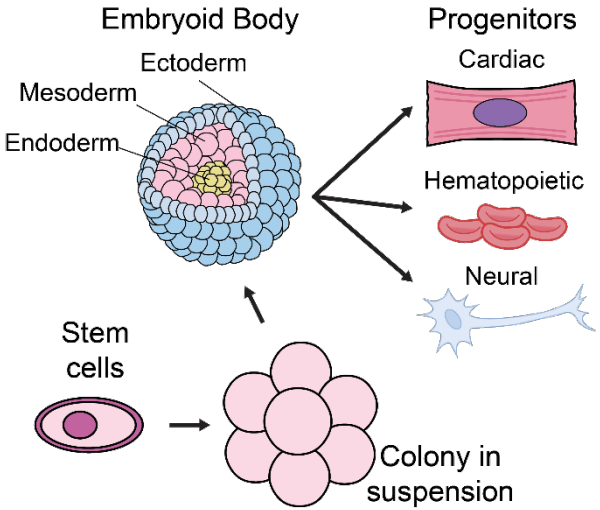
The Era of Big Data





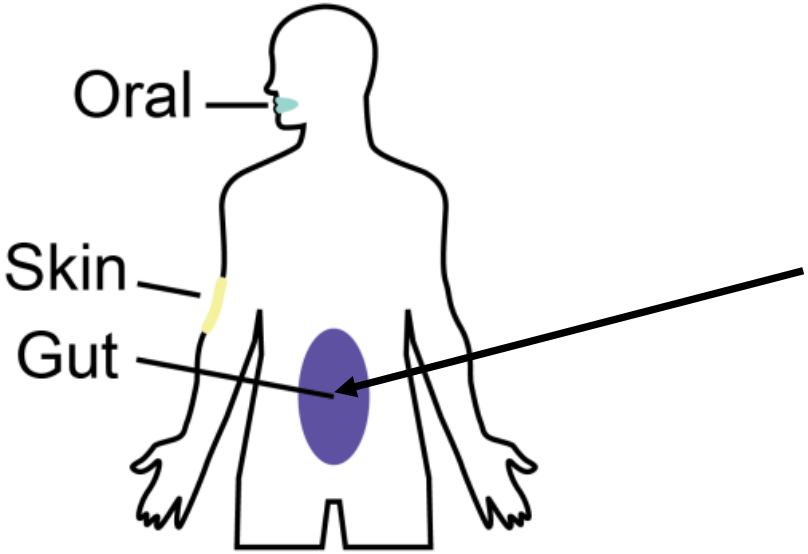


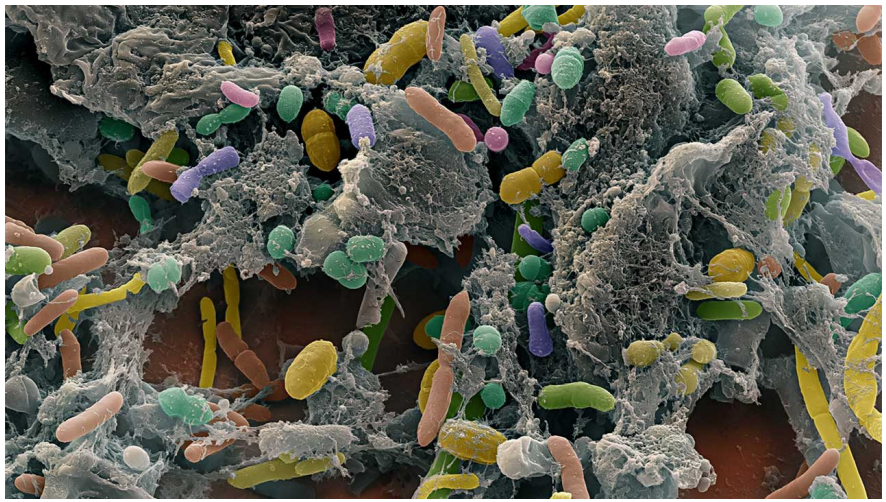




Human microbiome

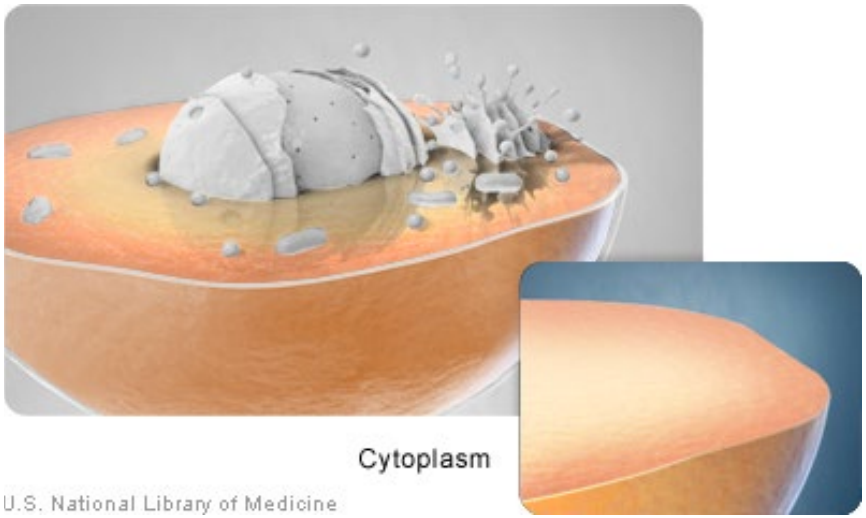
- Microbiome linked to various diseases
  - e.g. autism (Li et al., 2017), arthritis (Scher et al., 2016)

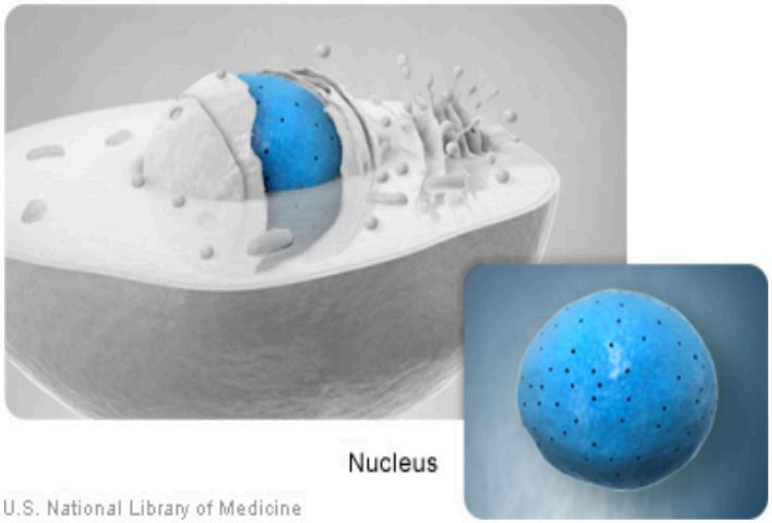




(Pennisi, 2016)

Parts of Cell

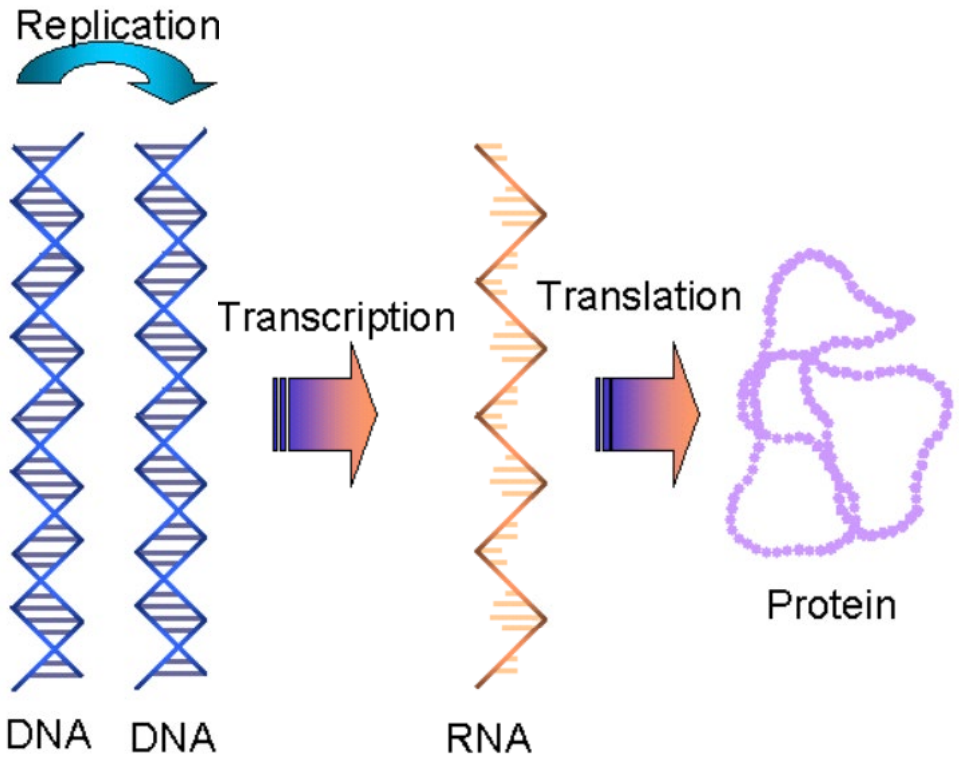




https://ghr.nlm.nih.gov/primer/basics.pdf

Central Dogma of Molecular Biology

- DNA
  - Information across generations
- RNA
  - Information around the cell
- Protein
  - Functional gene product



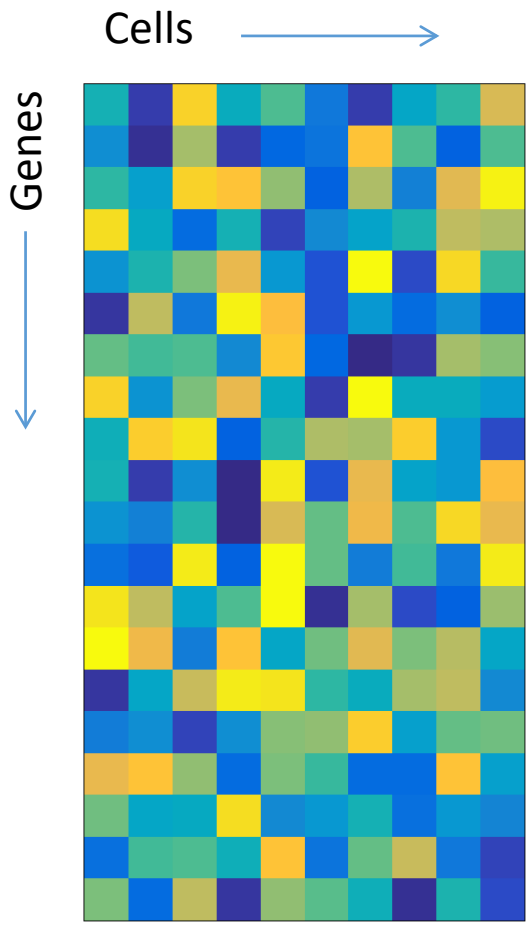
Potential Applications

Goal: infer cell states, functions, and responses from RNA expression measurements to improve our understanding of biology:

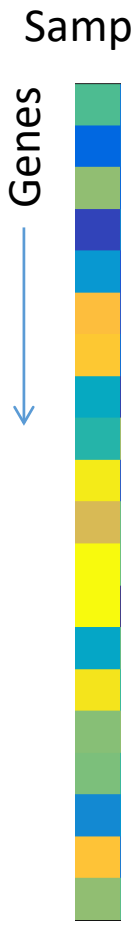
- Cancer metastasis
- Autoimmune diseases
- Immunological responses to diseases and treatment
- Effects of aging
- Effects of genetic mutations
- Etc.

Bulk vs. Single-Cell Data

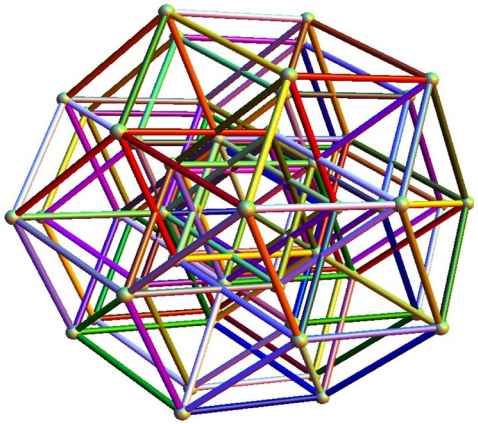
Single-cell Measurement



Bulk Measurement



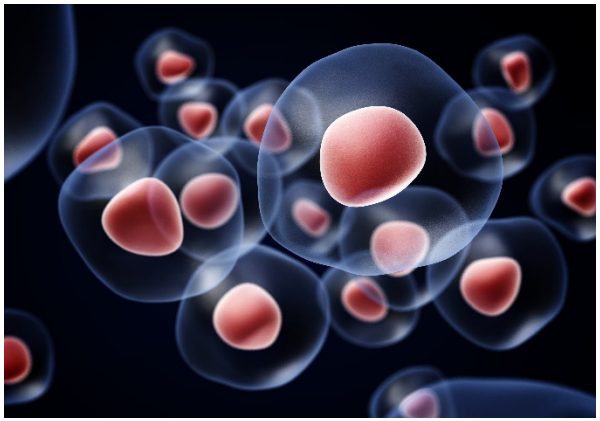
Single-Cell Data



High Dimensional

High Throughput

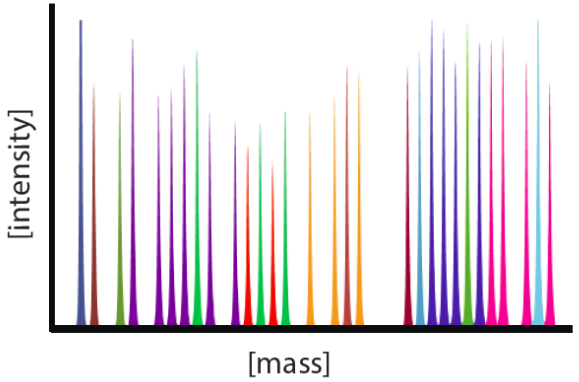
1000s to 100,000s of cells



Heterogeneous

Current Single-Cell Measurement Technologies

- Mass Cytometry (CyTOF)
  - 30 to 40 targets
  - 10,000 to 100,000s of cells
- Single-cell RNA-sequencing (scRNA-seq)
  - 2000+ genes
  - 100 to 10,000s of cells



Challenges in Big Data Analysis

- Interpretability
- Labels or samples may be expensive
- Often, little is known about the data
  - Data exploration is necessary

→ Supervised methods are insufficient in many cases

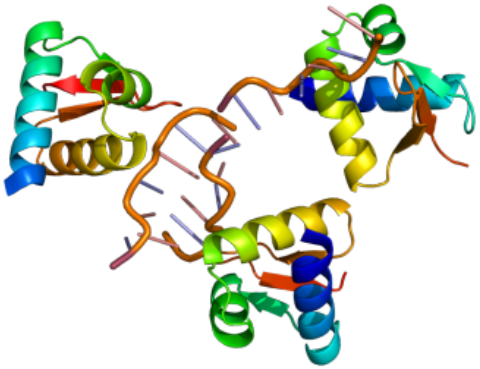
xkcd.com/1838

Exploratory Data Analysis

Descriptive Exploration

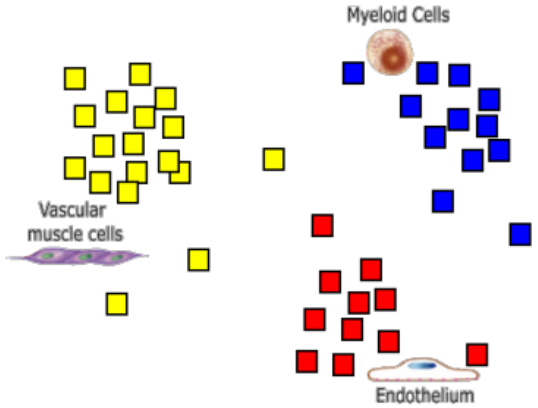
Single-cell data:

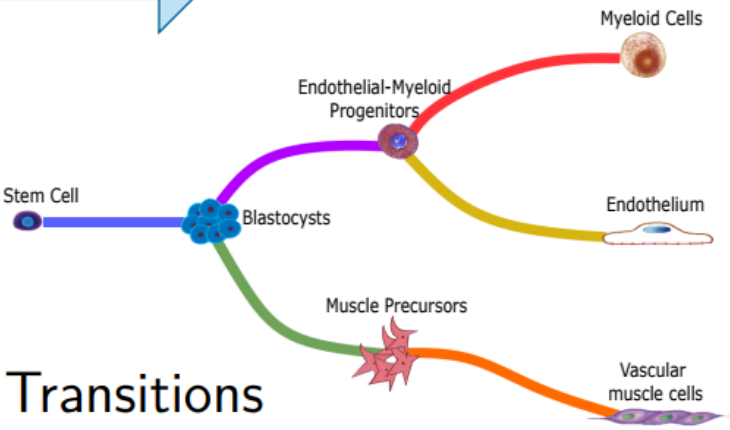
Gene counts



Protein counts

Clusters





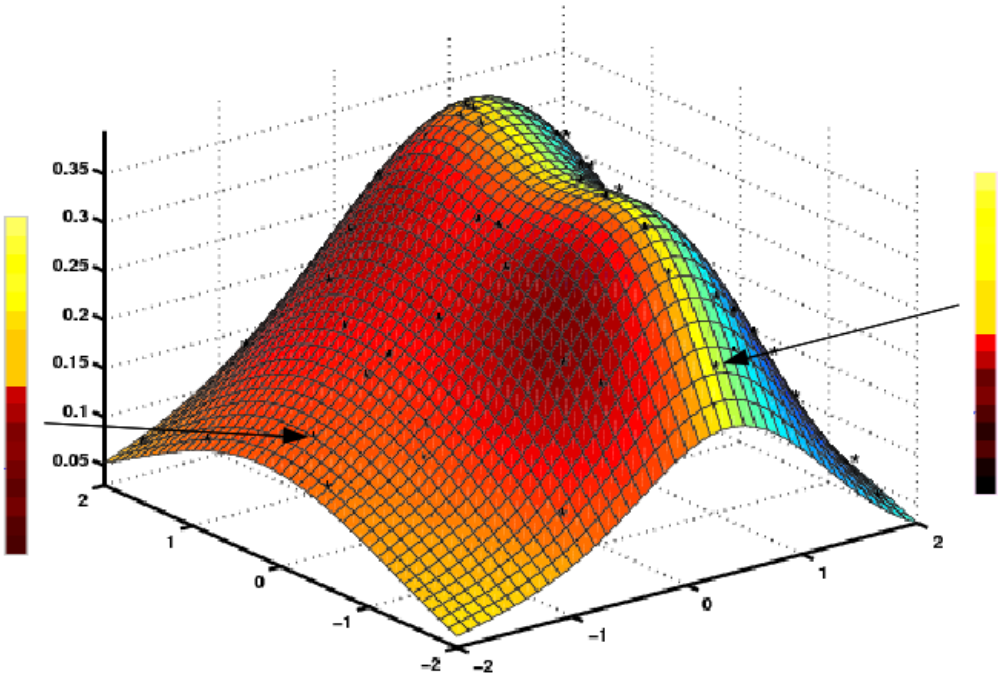
Transitions

Data Visualization

- Humans are very visual
- Data visualization is a necessary tool for exploration
  - Develop intuitive understanding of the structure
  - Generate hypotheses

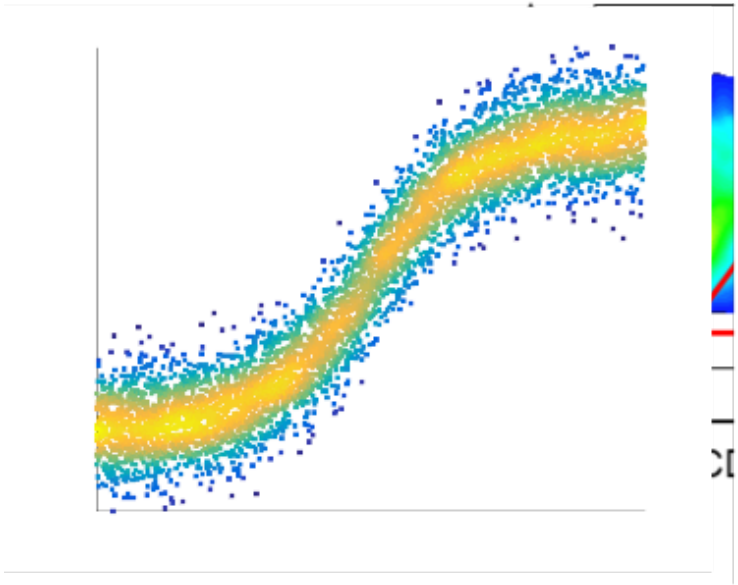
High dimensionality

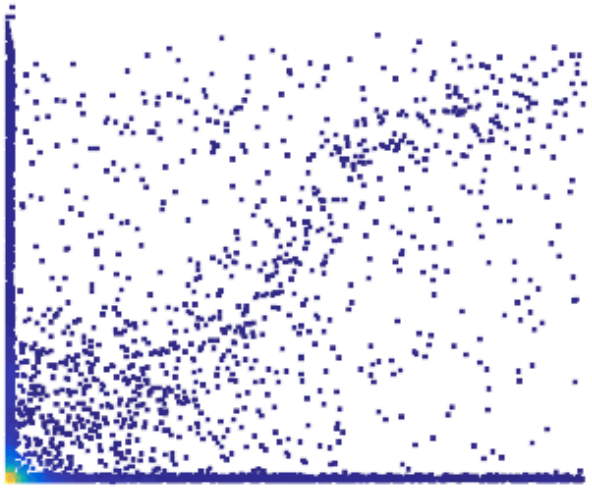
- High dimensional data can often be modeled with lower dimensional manifolds
  - Ex: facial expressions (high dimensional images) are controlled by only a few muscles
- Goal: learn the manifold from the data (manifold learning)



Challenges in Visualizing (Biomedical) Data

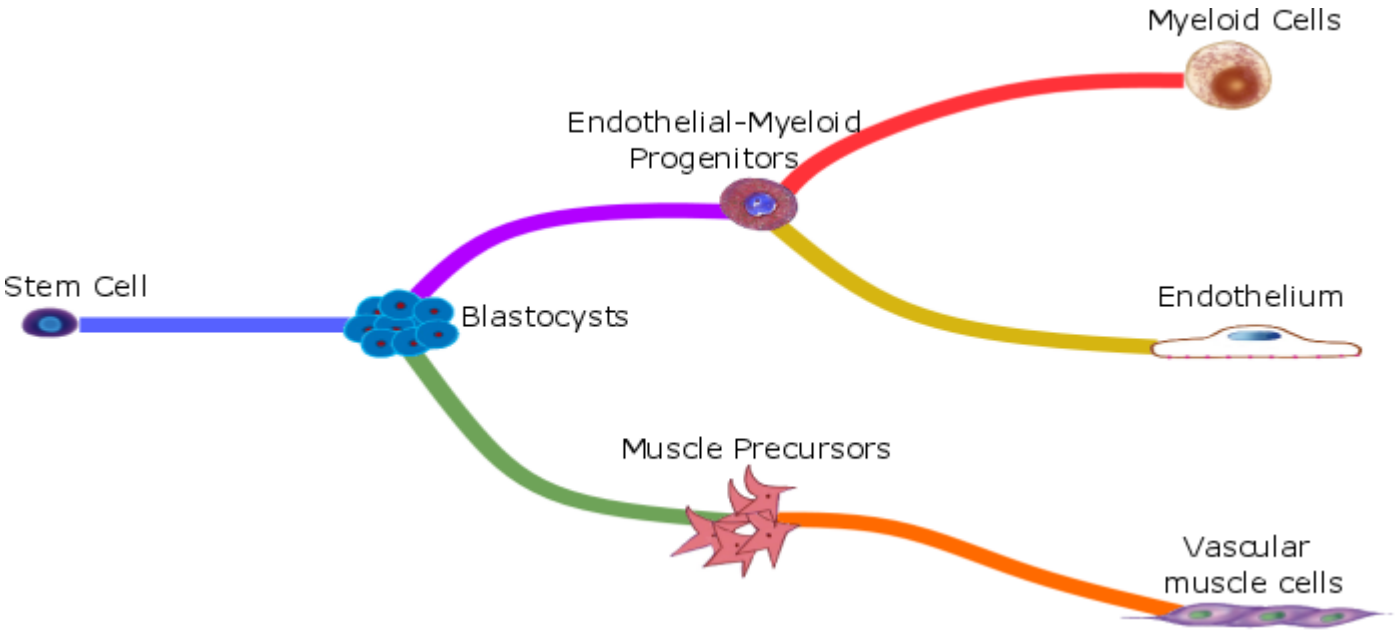
- High dimensions
- Noise/artifacts
- Nonlinearities
  - Most biological processes are NOT linear



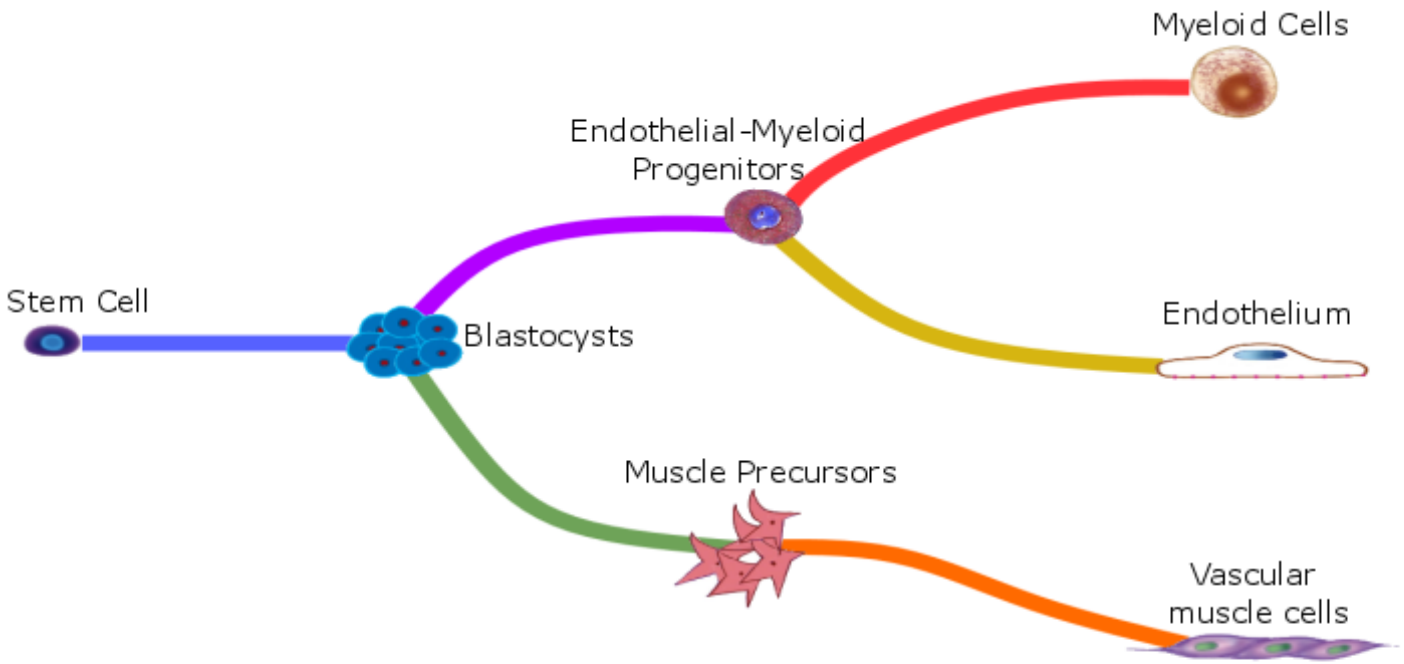


Example: Transitional Structure

- Many biological systems have a dominant transitional structure
  - Developmental branches
  - Cancer
  - Sleep stage



Visualizing Structure



How?

- How will we do this?
- Answer:
  1. Use a crafted version of a robust dimensionality reduction method known as diffusion maps to learn the manifold.
  2. Transform the learned structure into low dimensions for visualization

Existing Methods: PCA

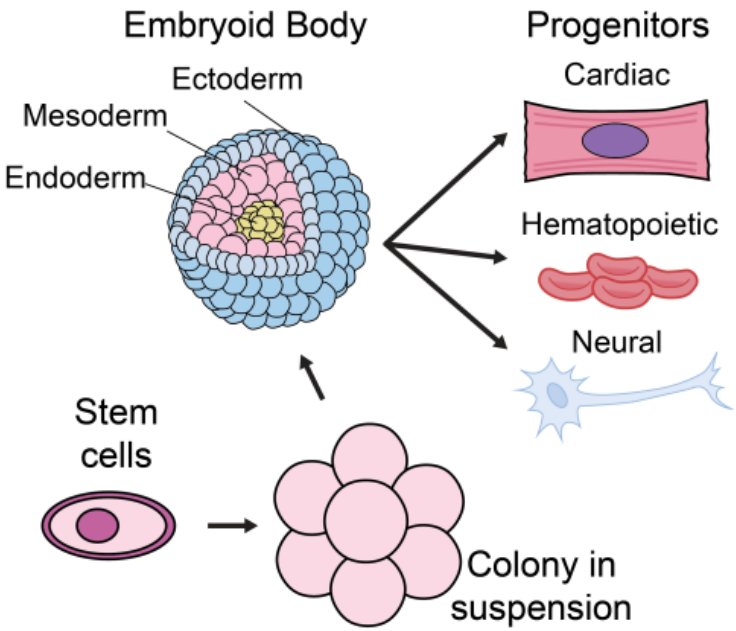
Artificial Tree Data

PCA

1400 points, 60 dimensions

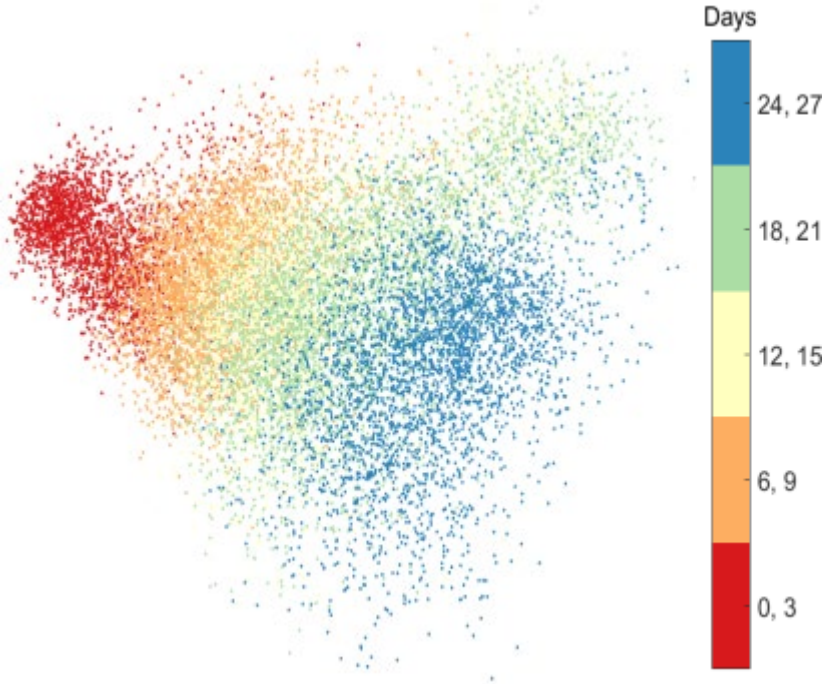
Existing Methods: PCA

Newly generated scRNA-seq data (27 days)



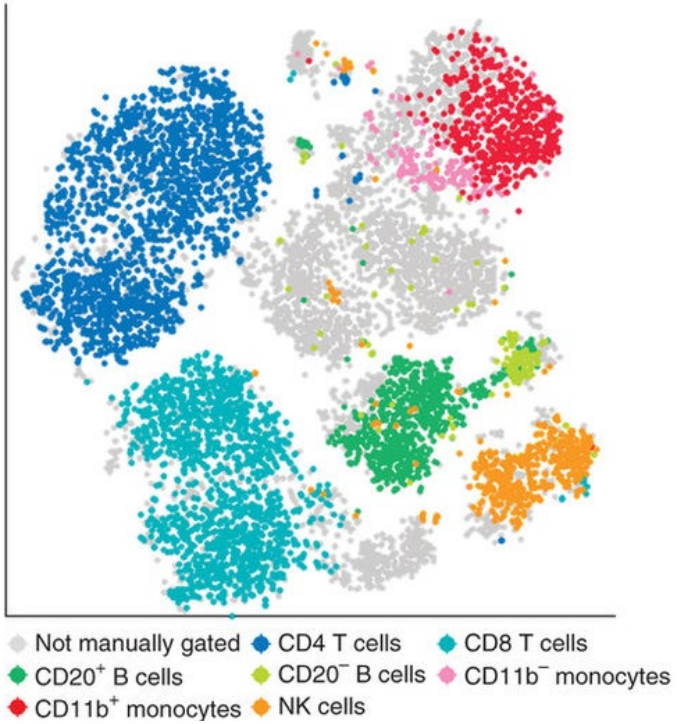
≈31k cells, more than 17k genes

PCA



Existing Methods: t-SNE (van der Maaten & Hinton, *JMLR*, 2008)

- Widely used on single cell data
  - Biology version (Amir et al, *Nature Biotech*, 2013) has 500+ citations
- Attempts to preserve local relationships in both the high and low-dimensional spaces
  - Designed for separating clusters



(Amir et al, 2013)

Existing Methods: t-SNE (van der Maaten & Hinton, *JMLR,* 2008)

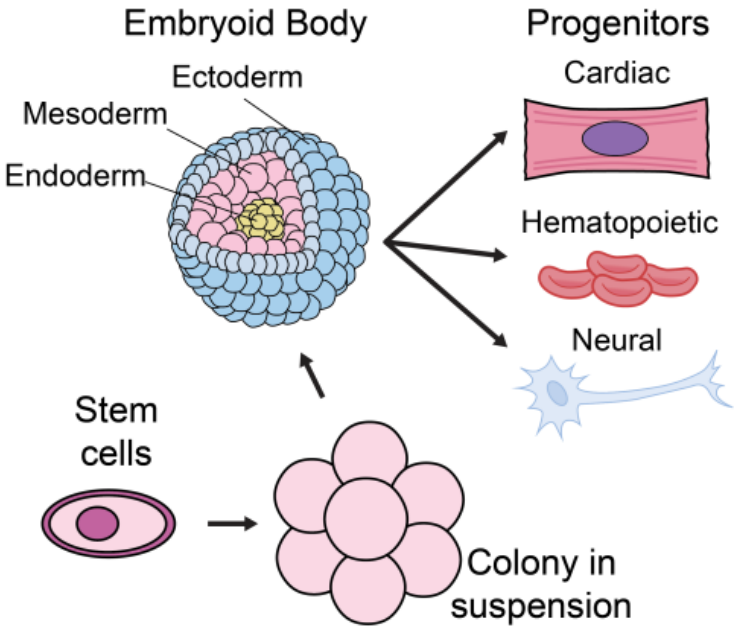
Artificial Tree Data

t-SNE

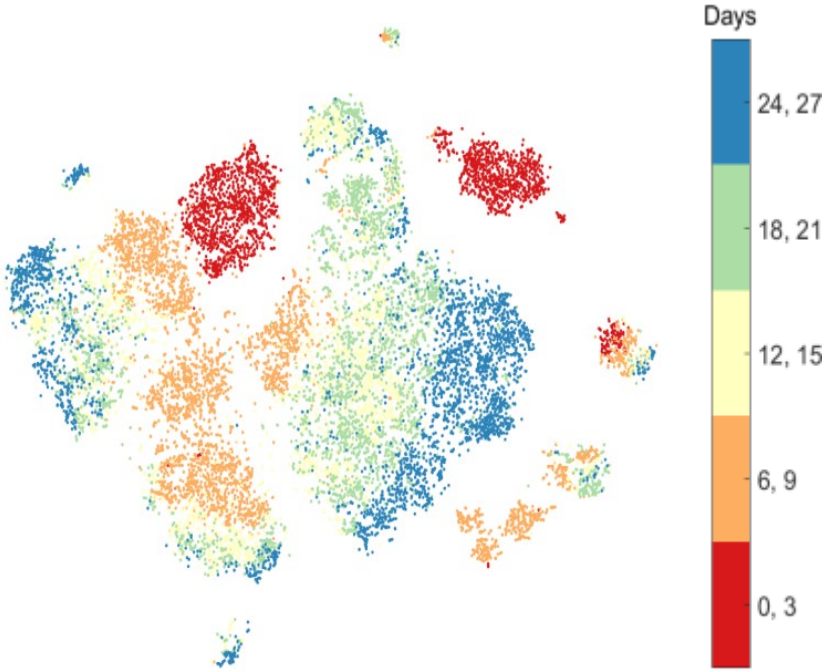
1400 points, 60 dimensions

Existing Methods: t-SNE (van der Maaten & Hinton, *JMLR*, 2008)

Newly generated scRNA-seq data (27 days)



t-SNE



≈31k cells, more than 17k genes

Visualizing Structure and Transitions in High-Dimensional Biological Data

Preprint (bioRxiv): https://doi.org/10.1101/120378

Code (GitHub): See final slide

Joint work with:

Yale Applied Math & Genetics

Guy Wolf,

David van Dijk,

Smita Krishnaswamy,

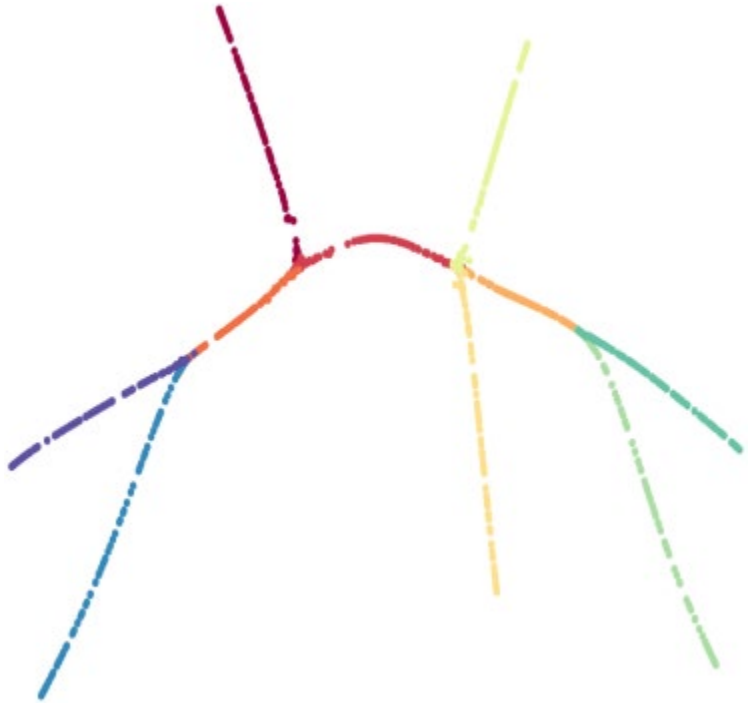
Z. Wang, W. Chen, M.J. Hirn, R.R. Coifman, N.B. Ivanova et al.

Accepted at *Nature Biotechnology (2019)*

PHATE on Artificial Tree Data

Artificial Tree Data

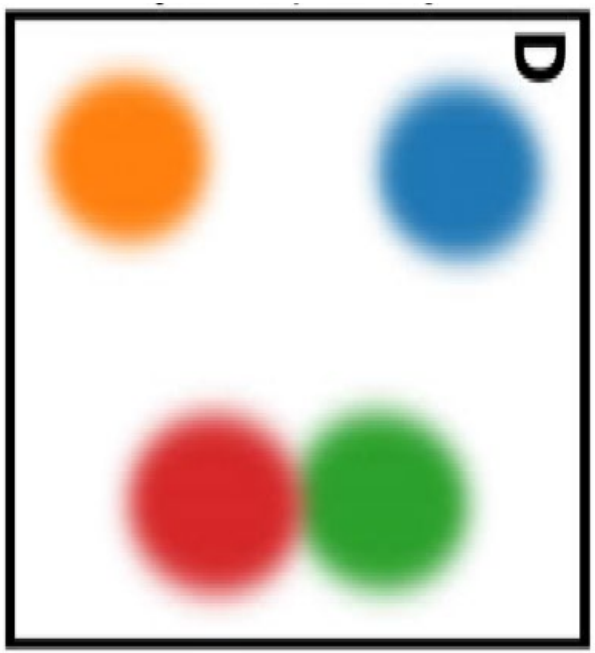
PHATE



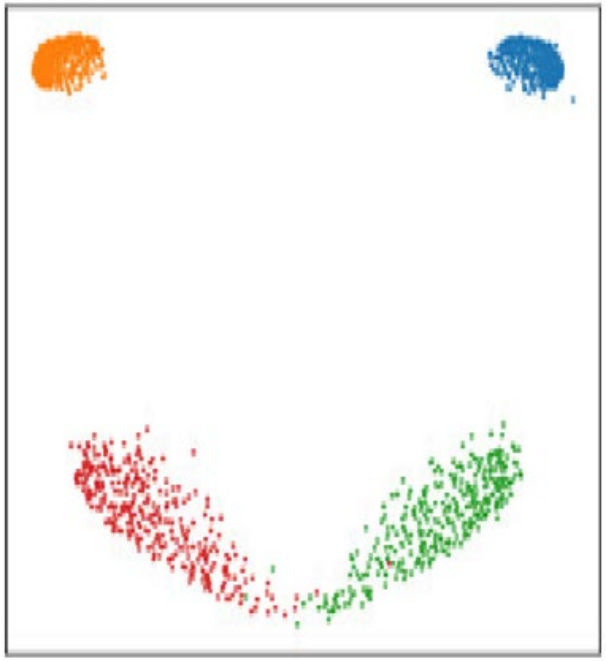
1400 points, 60 dimensions

PHATE on GMM

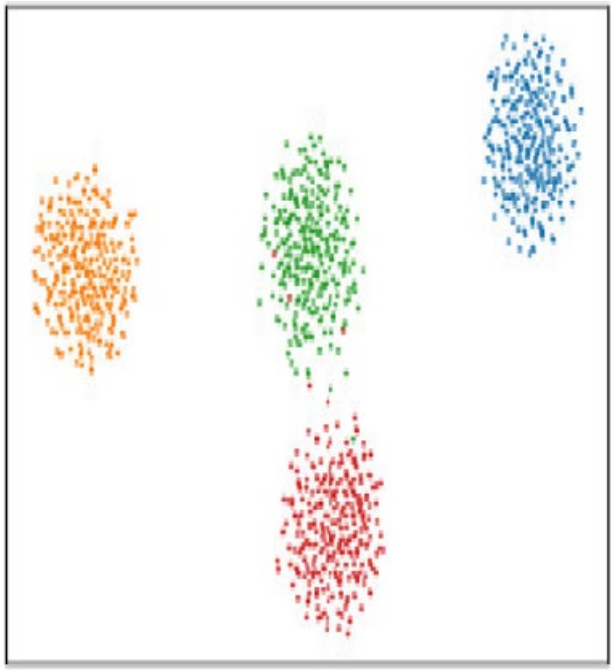
Truth



PHATE

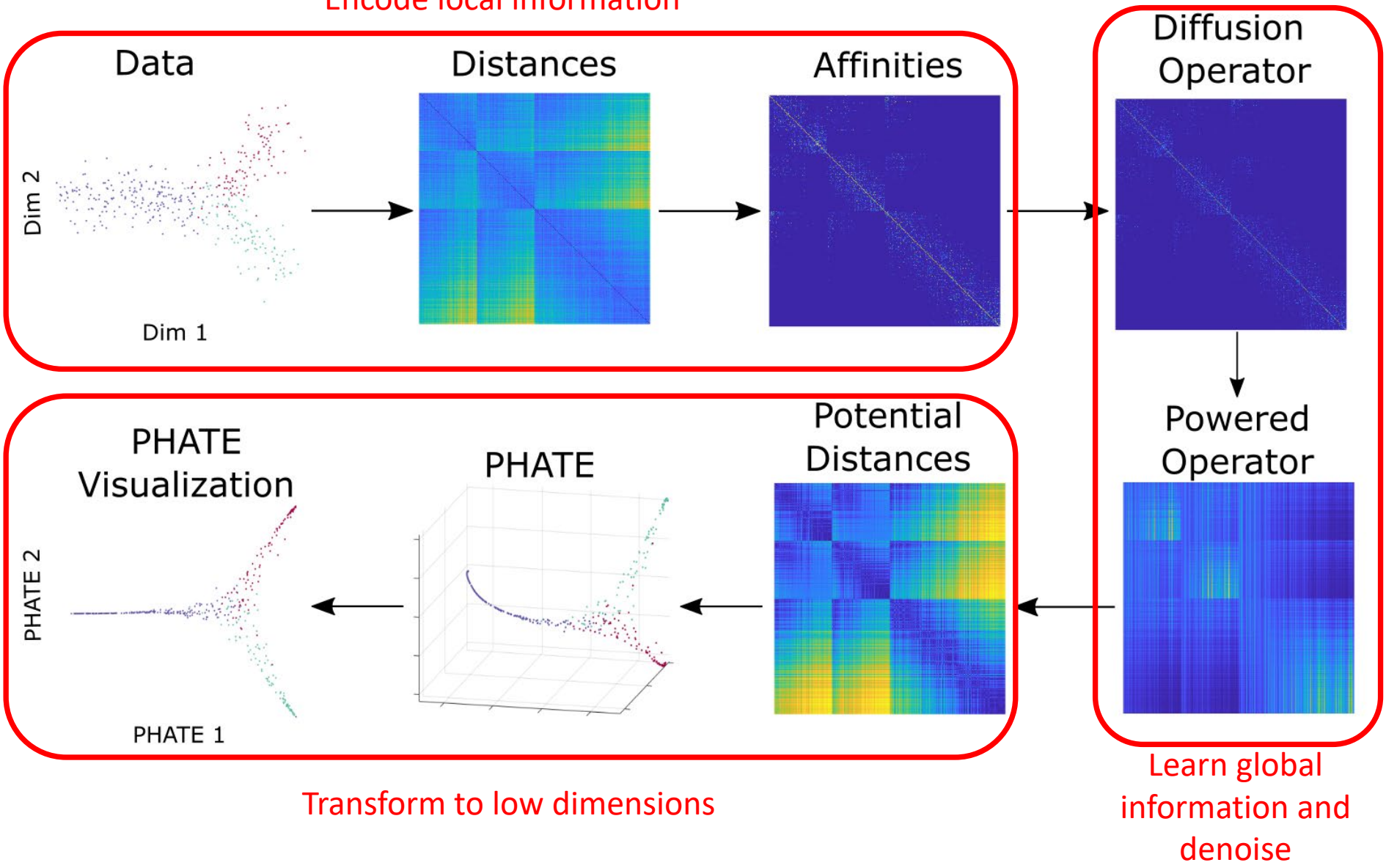


t-SNE



The PHATE Algorithm

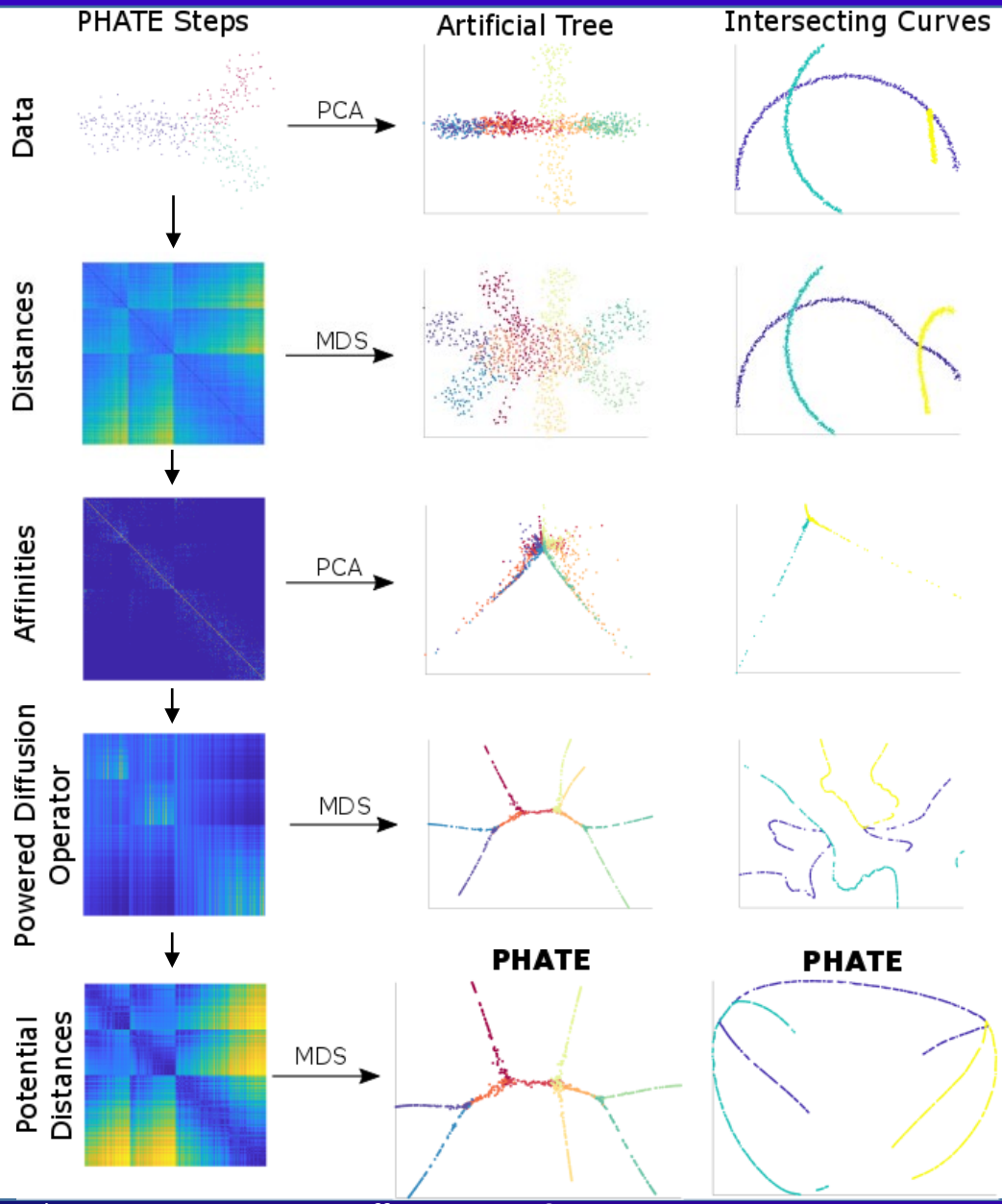
Encode local information



Learn global information and denoise

Transform to low dimensions

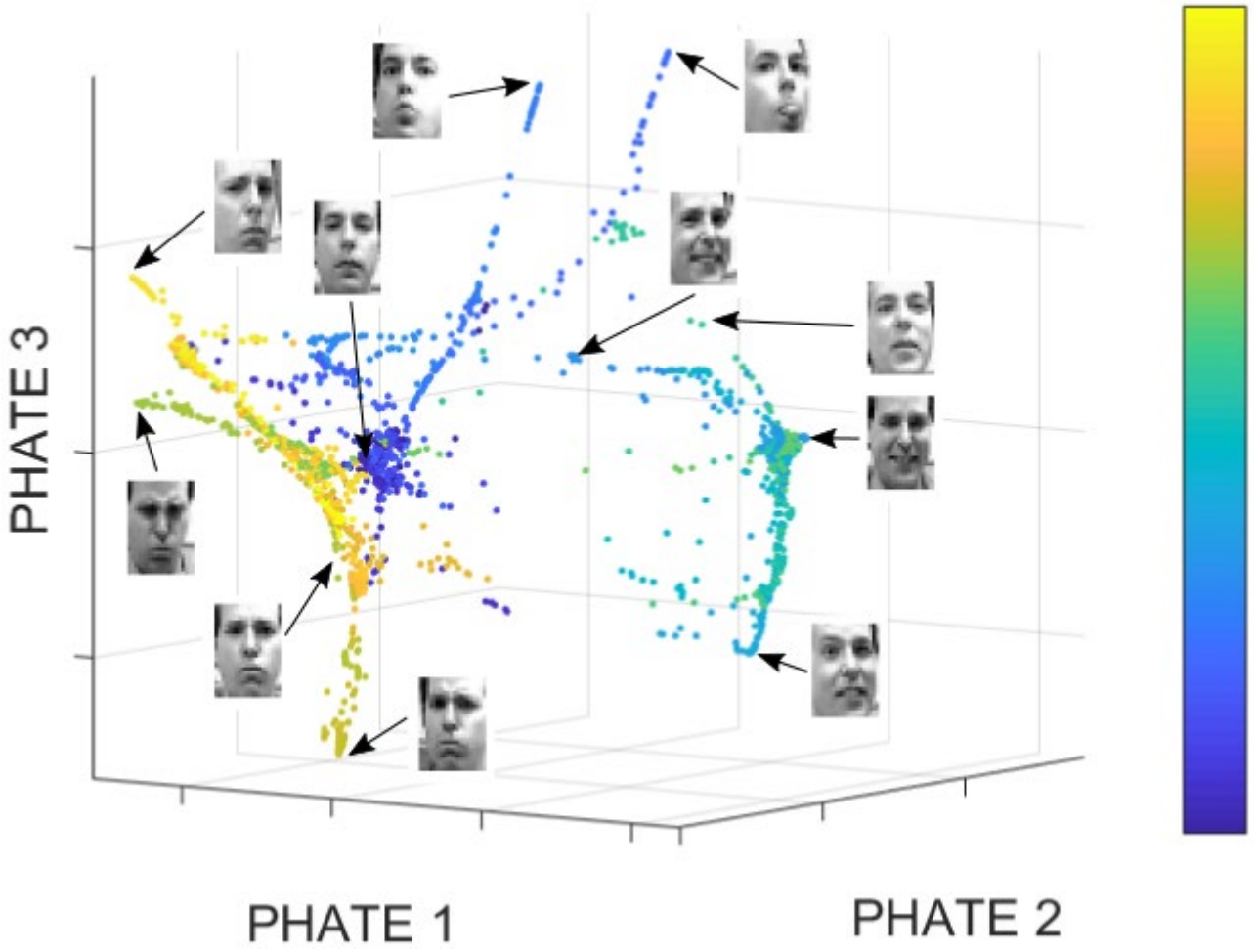
The PHATE Algorithm



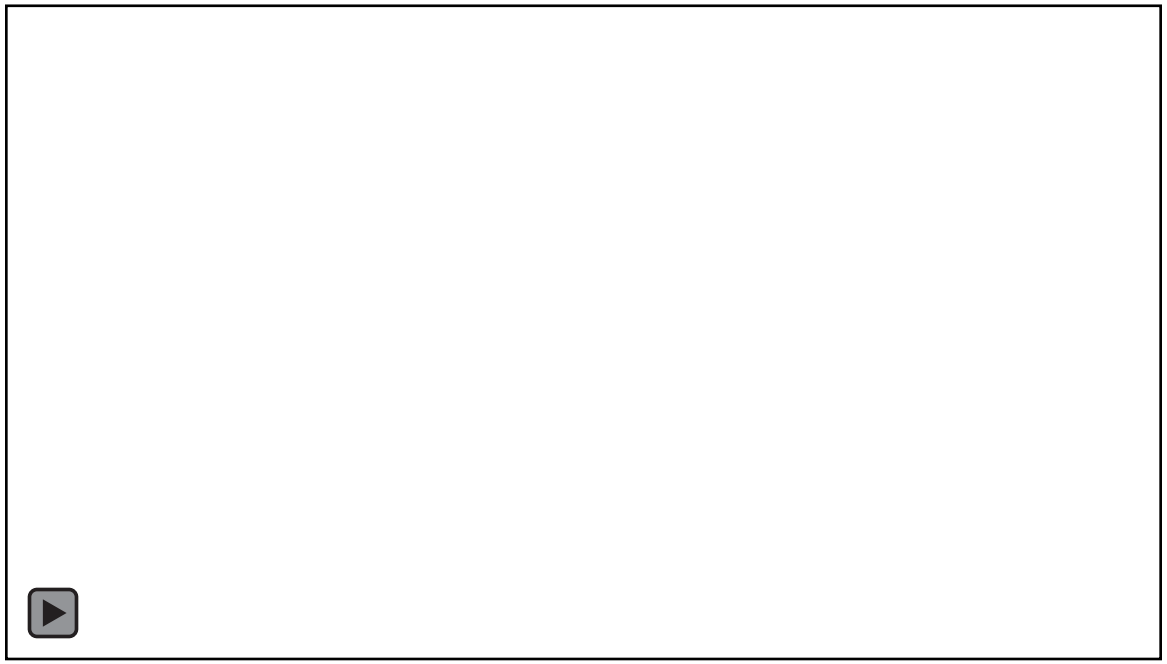
PHATE on Frey Faces

- Frey Faces dataset
  - 1965 frames taken from a video of Brendan Frey making various faces in front of a camera
  - Resolution: 20x28
- Frames are given out of order to PHATE w/o information about sequential ordering

PHATE on Frey Faces

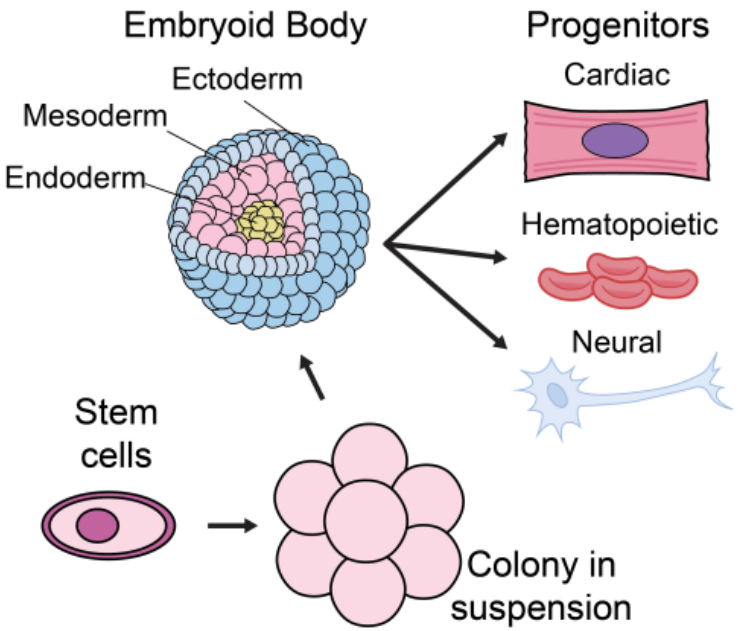


PHATE on Frey Faces



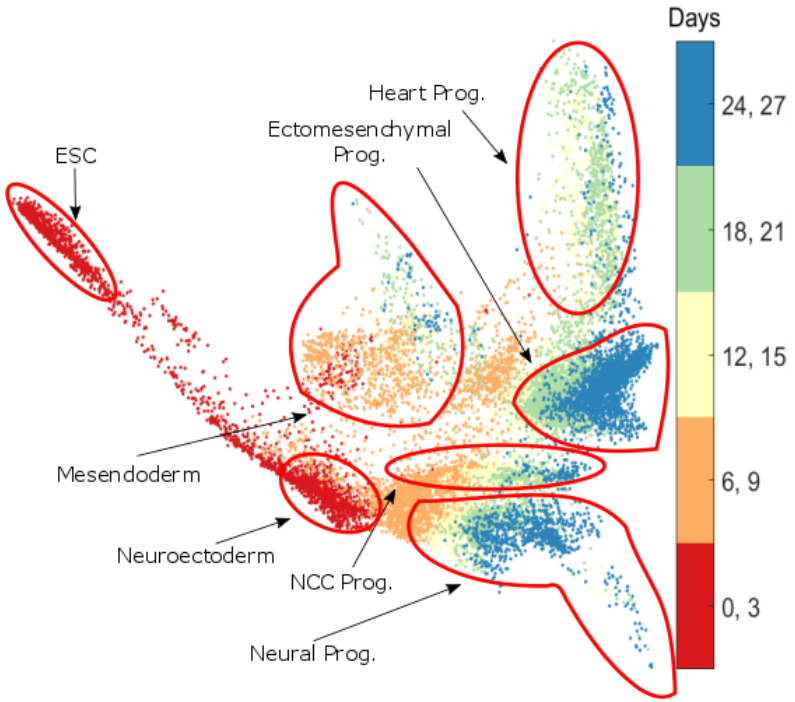
PHATE on EB scRNA-seq Data

Newly generated scRNA-seq data (27 days)



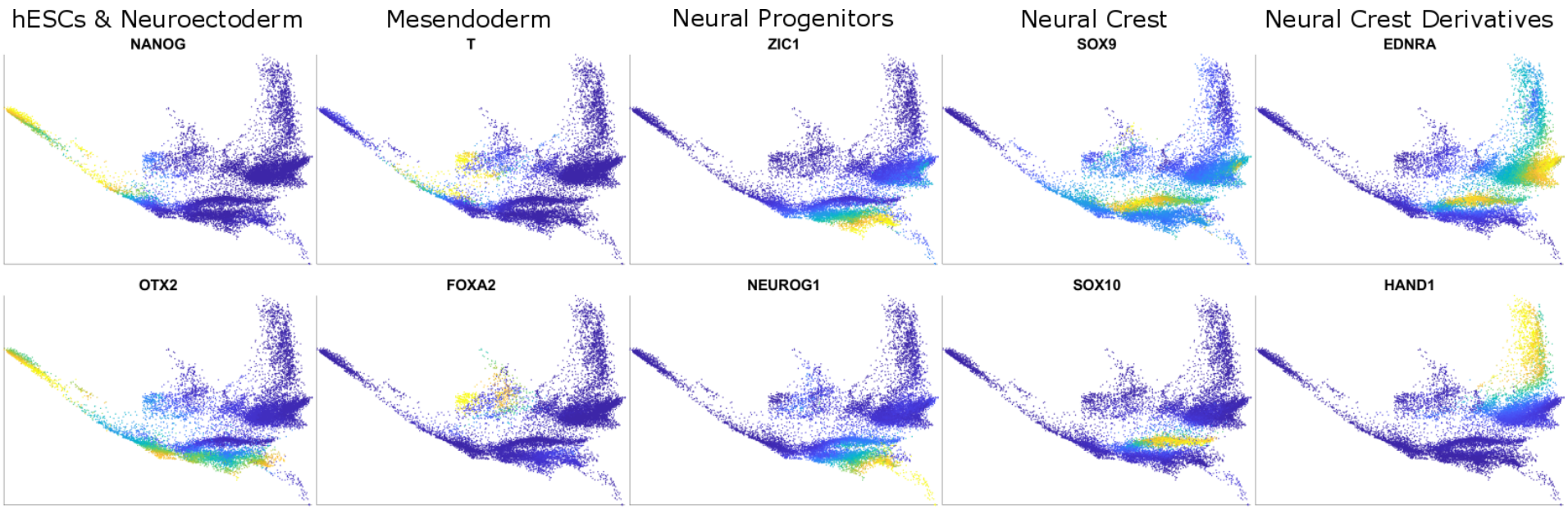
≈31k cells, more than 17k genes

PHATE



Exploratory Data Analysis with PHATE

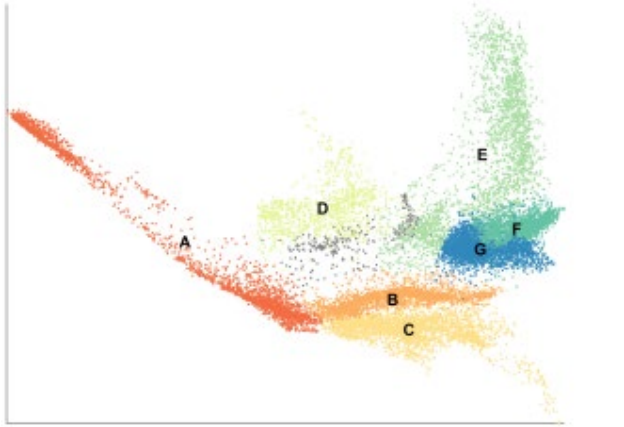
- Coloring the embedding by gene expression after MAGIC* (van Dijk,…Moon et al., 2018) reveals lineages



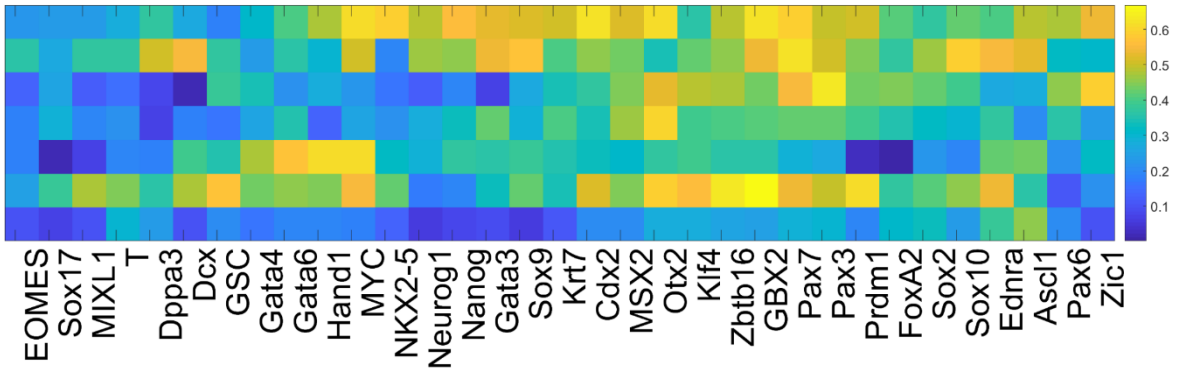
- Interactive web tool: krishnaswamylab.org/phatewebtool

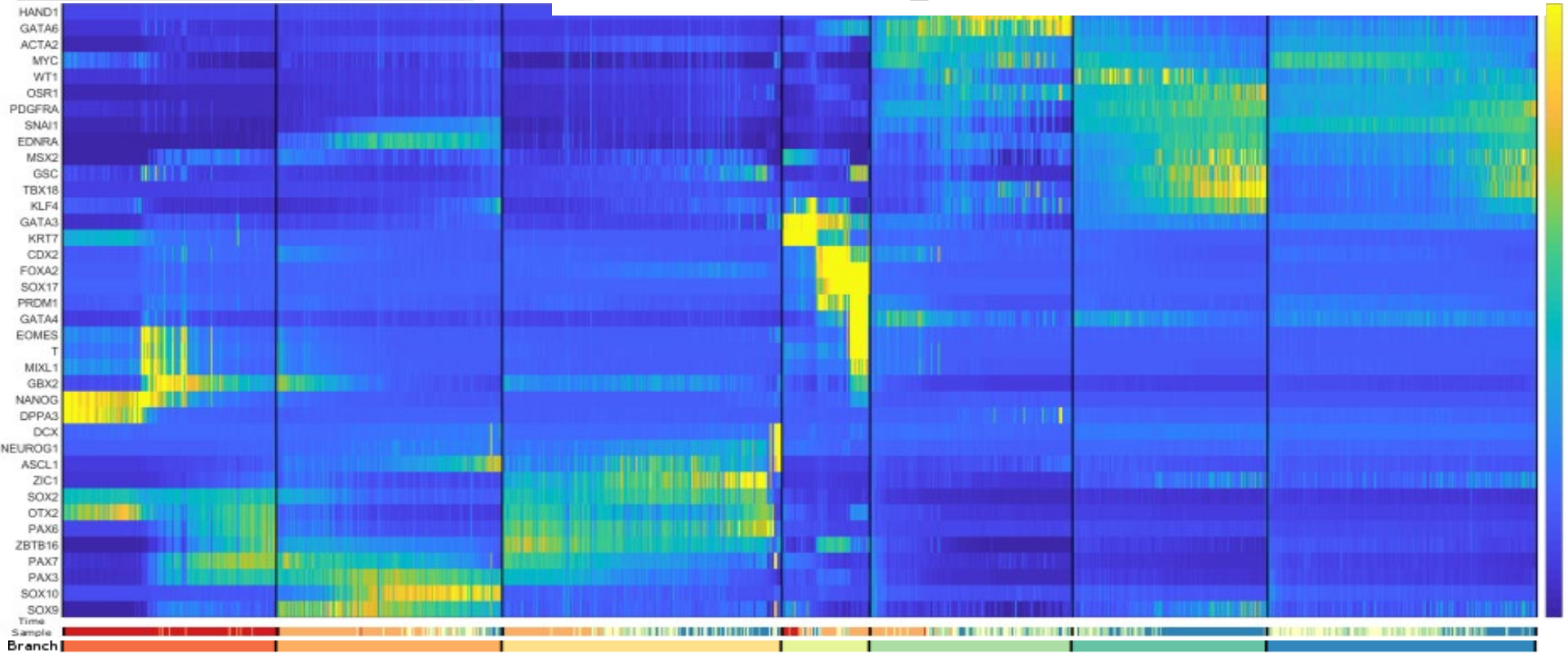
Published in *Cell

Exploratory Data Analysis with PHATE

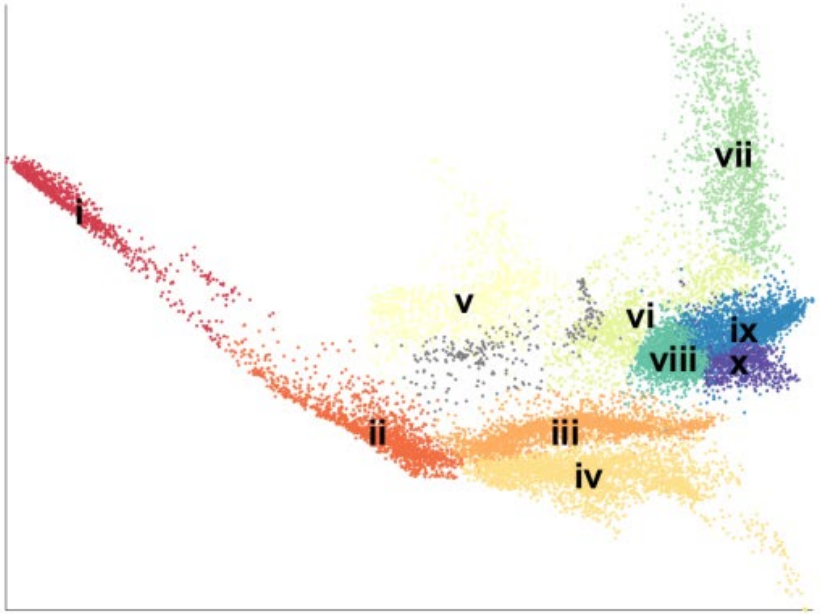


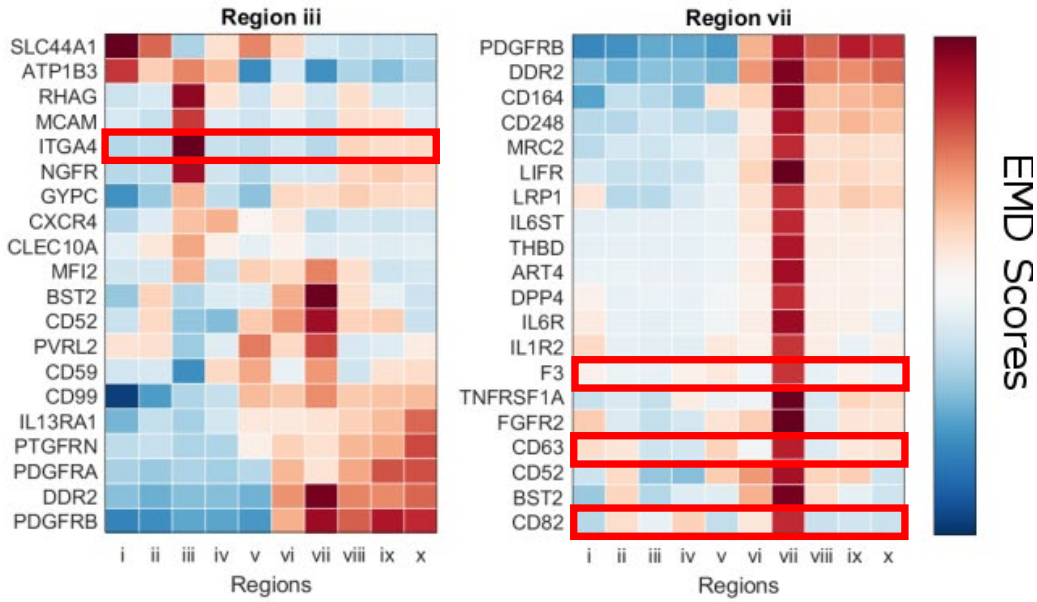
DREMI Score (Krishnaswamy et al., 2014)





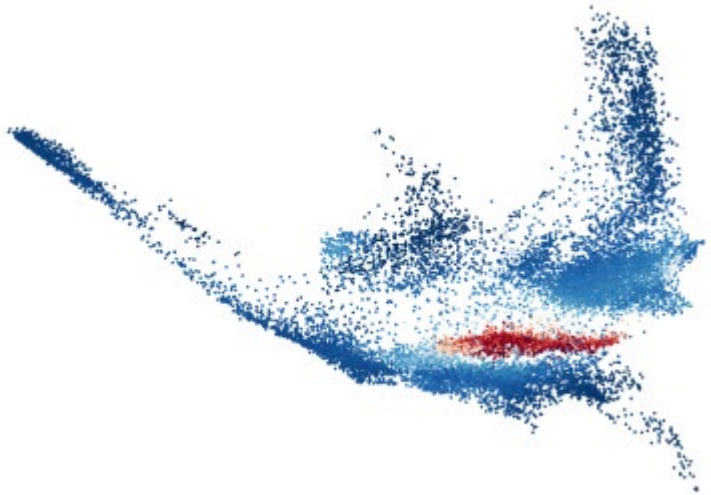
Discovering New Surface Markers for Sorting Populations

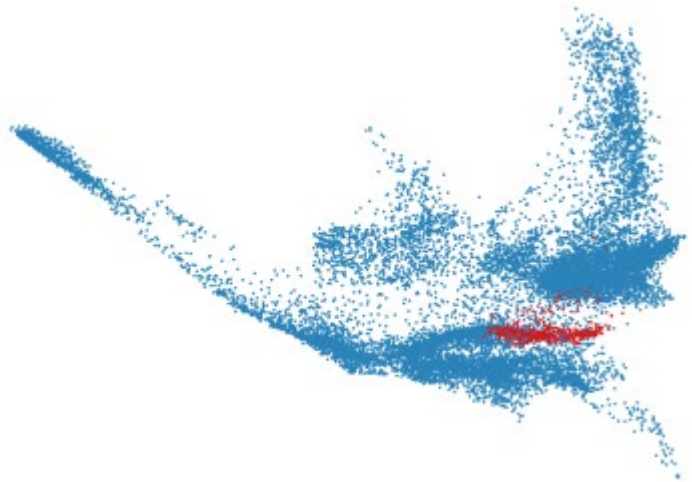




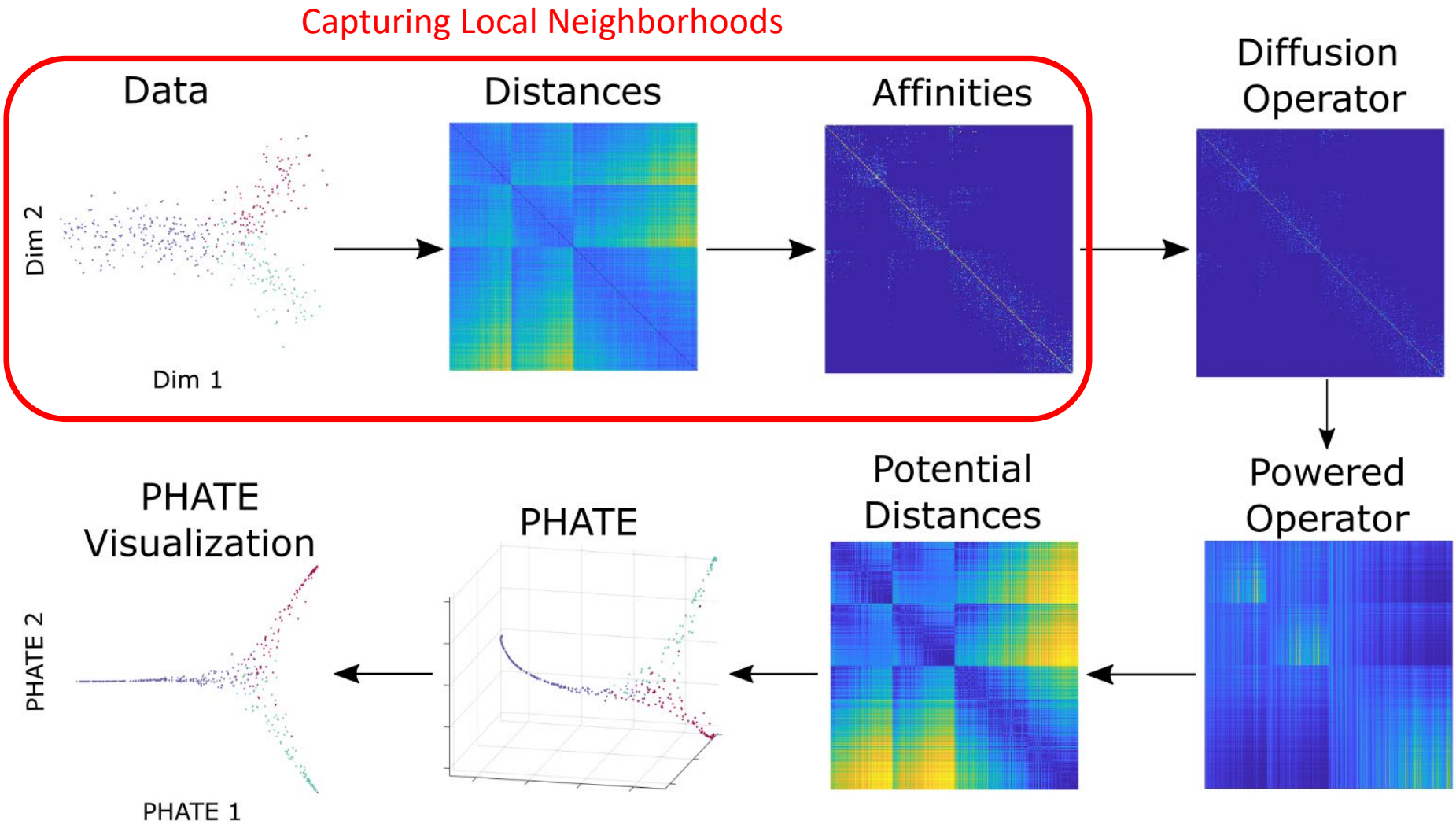
ITGA4

ITGA4 sorted bulk RNA-seq correlation

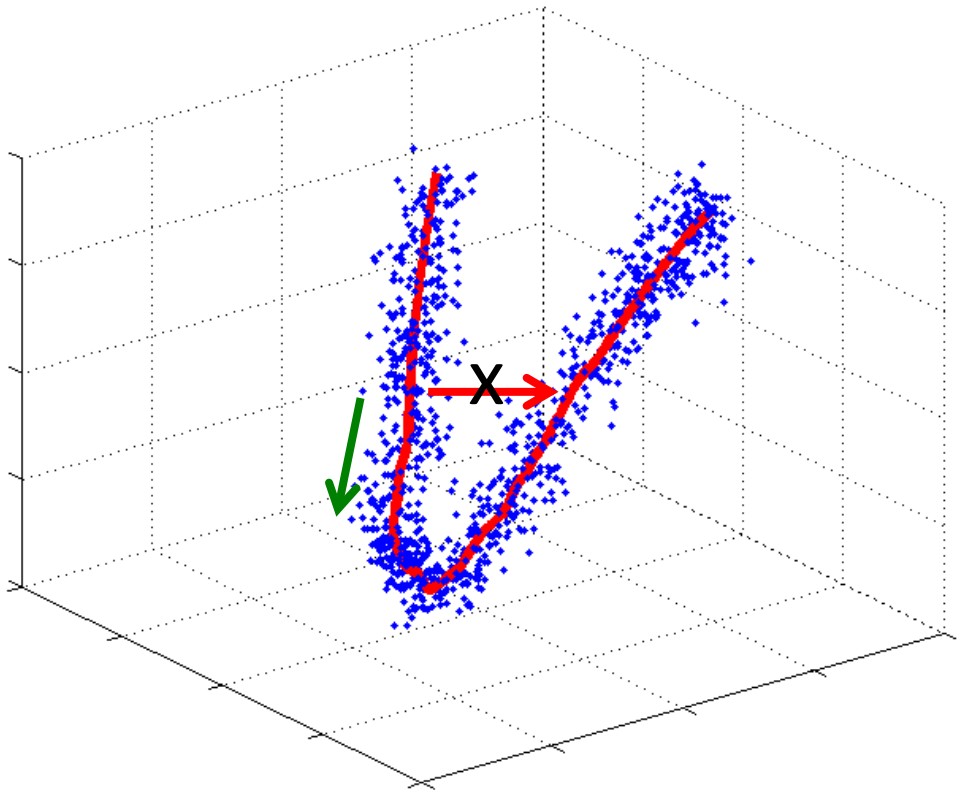




The PHATE Algorithm



Capturing local neighborhoods



- Large Euclidean distance gives wrong global structure for visualizing
- Small Euclidean distances ok
  - Encodes local structure

From distances to affinities

- Step 1: Calculate all pairwise Euclidean distances
- Step 2: Convert the distances to pairwise affinities using a kernel function
  - E.g. the Gaussian kernel
  - Kernel function must be near zero for large distances and nonzero for small distances
  - Affinity: a measure of similarity between points
- **Problem**: many data have both dense and sparse regions
- A kernel bandwidth fixed for dense regions doesn't work well for sparse regions and vice versa
- Idea: use locally adaptive bandwidth

Capturing Local Neighborhoods

- **Adaptive α-decaying kernel**

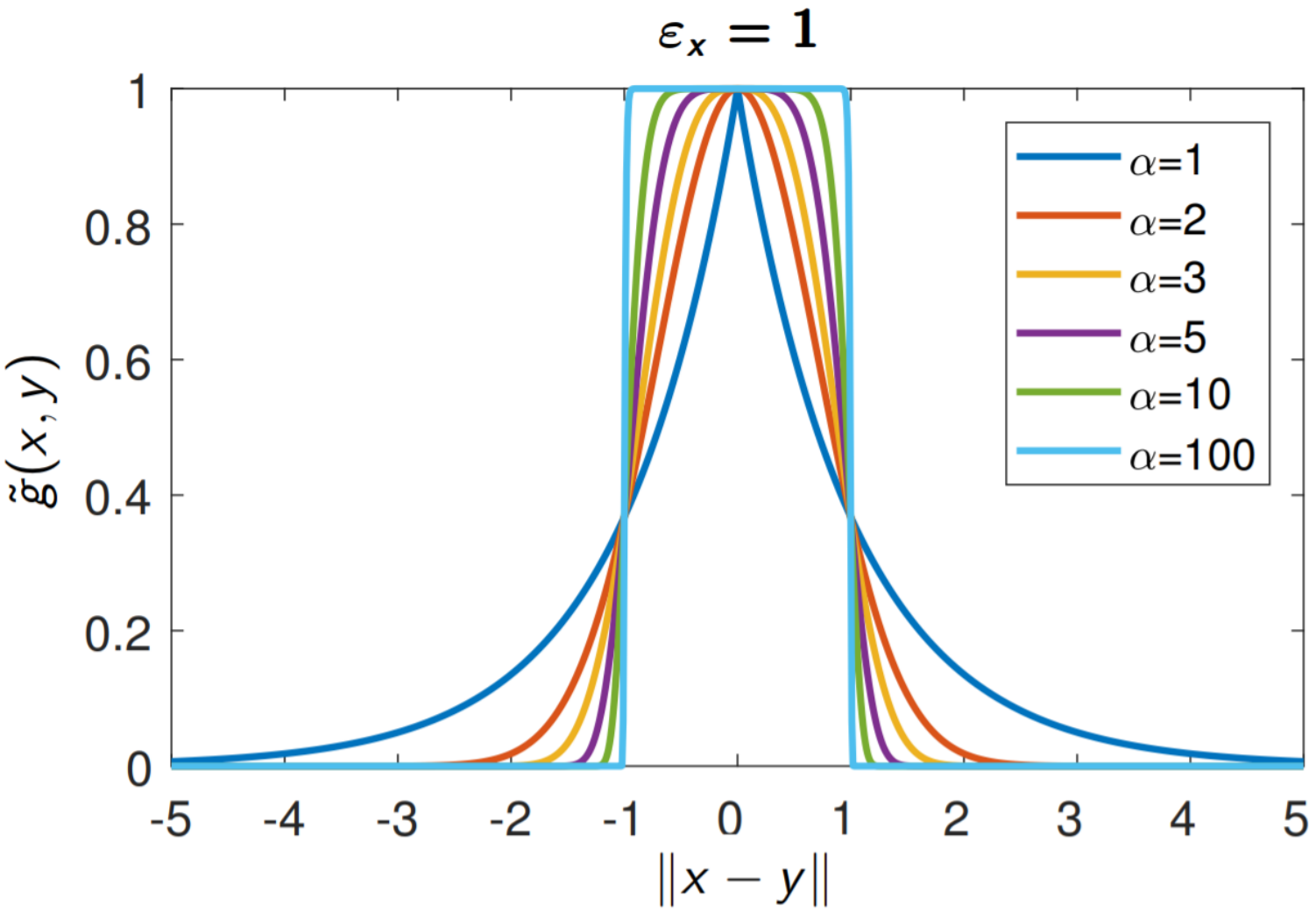
$$\tilde{g}(x,y) = \exp\left(-\left(\frac{||x-y||}{\epsilon_x}\right)^{\alpha}\right), \rightarrow g(x,y) = \frac{\tilde{g}(x,y)+\tilde{g}(y,x)}{2}$$

Where

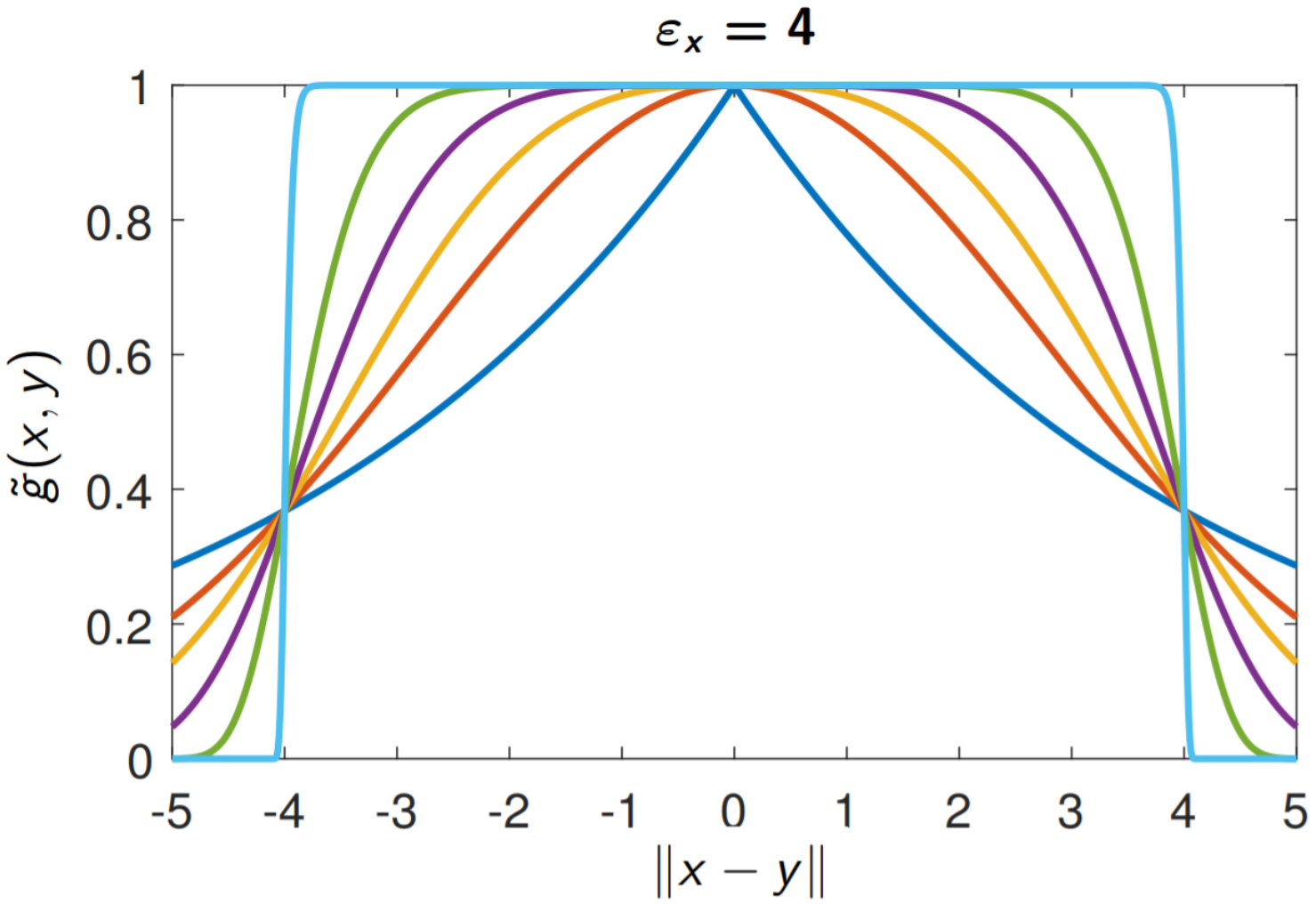
- ϵ_x = distance from x to its kth nearest neighbor
- α controls the decay rate of $\tilde{g}$

- Provides a robust notion of adaptive locality

Capturing Local Neighborhoods



Capturing Local Neighborhoods

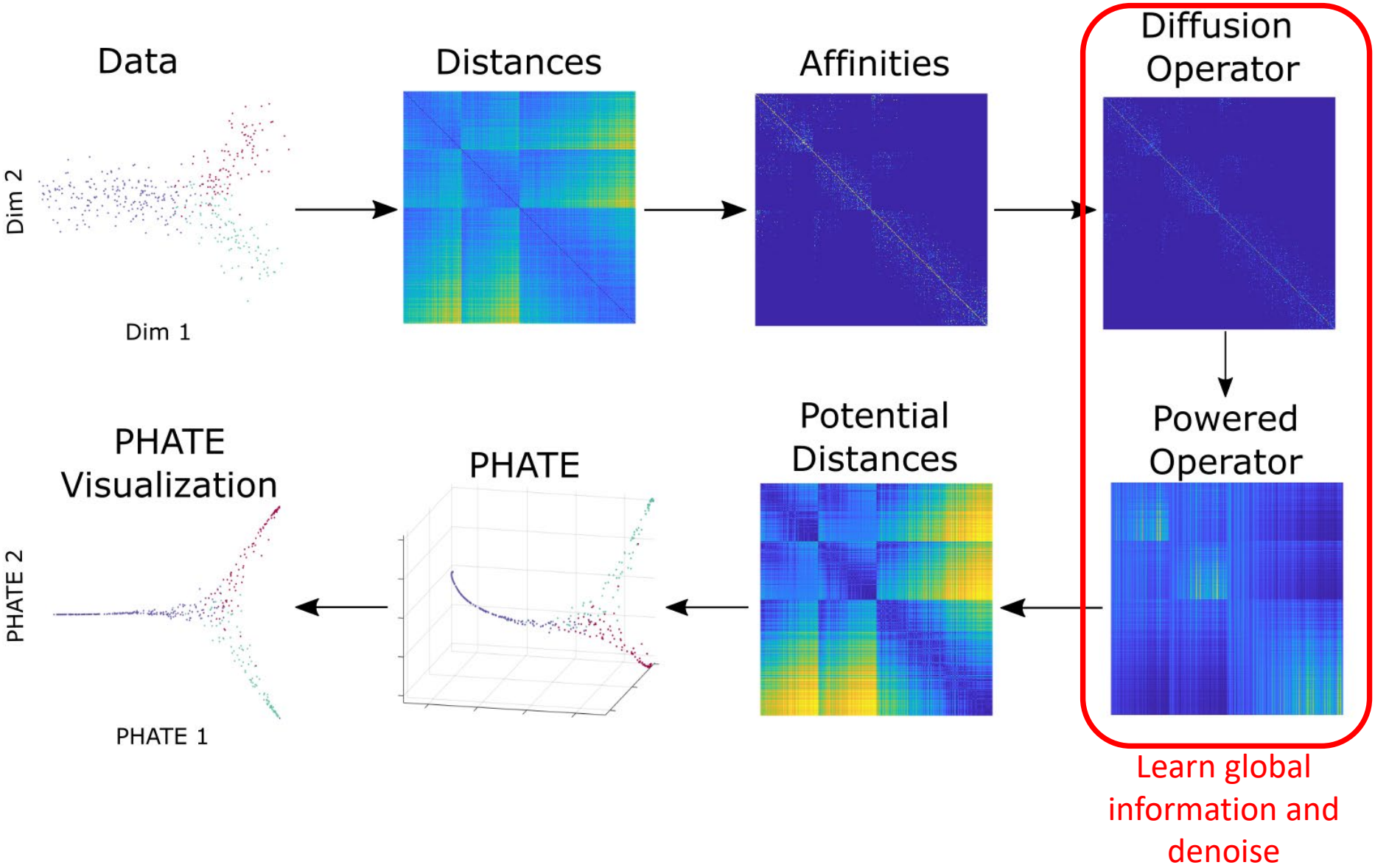


Group Exercises

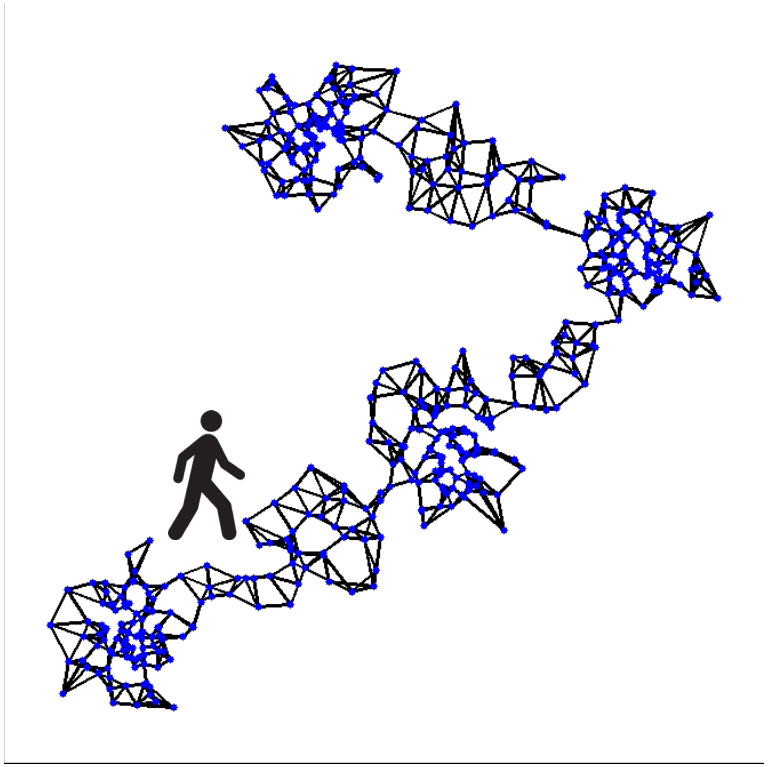
1. Give an example of data where taking the Euclidean distance between data points does not make sense.
2. How would you apply PHATE to this data?

1. Images where small rotations and/or shifts do not matter. Also network data with a built-in notion of similarity.
2. Give some measure of distance or similarity that ignores shifts and/or rotations. For network data, you can simply use the existing notion of similarity and diffuse on that.

The PHATE Algorithm

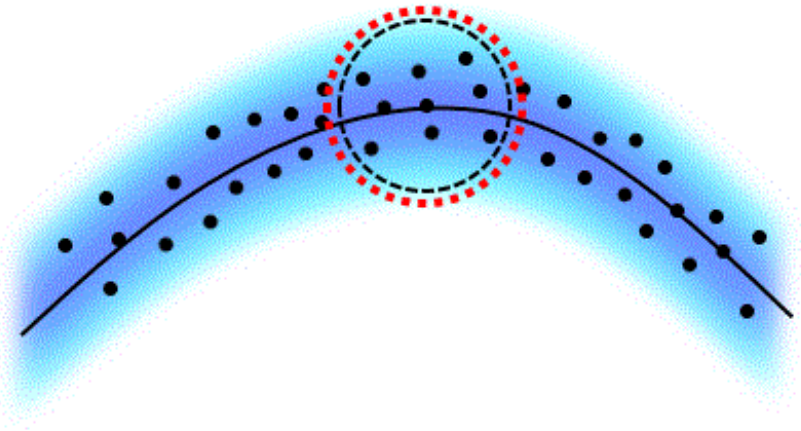


Diffusion Denoises and Recovers Global Structure



- Big Euclidean steps bad, likely to exit structure
- Small steps good, likely to stay within structure
- To learn global structure via small local steps we use random walk (diffusion)
  - Normalize affinity matrix to create Markov transition matrix (the diffusion operator) P (Coifman & Lafon, *ACHA*, 2006)
  - Power P by time step t

Diffusion Denoises and Recovers Global Structure



How much diffusion?

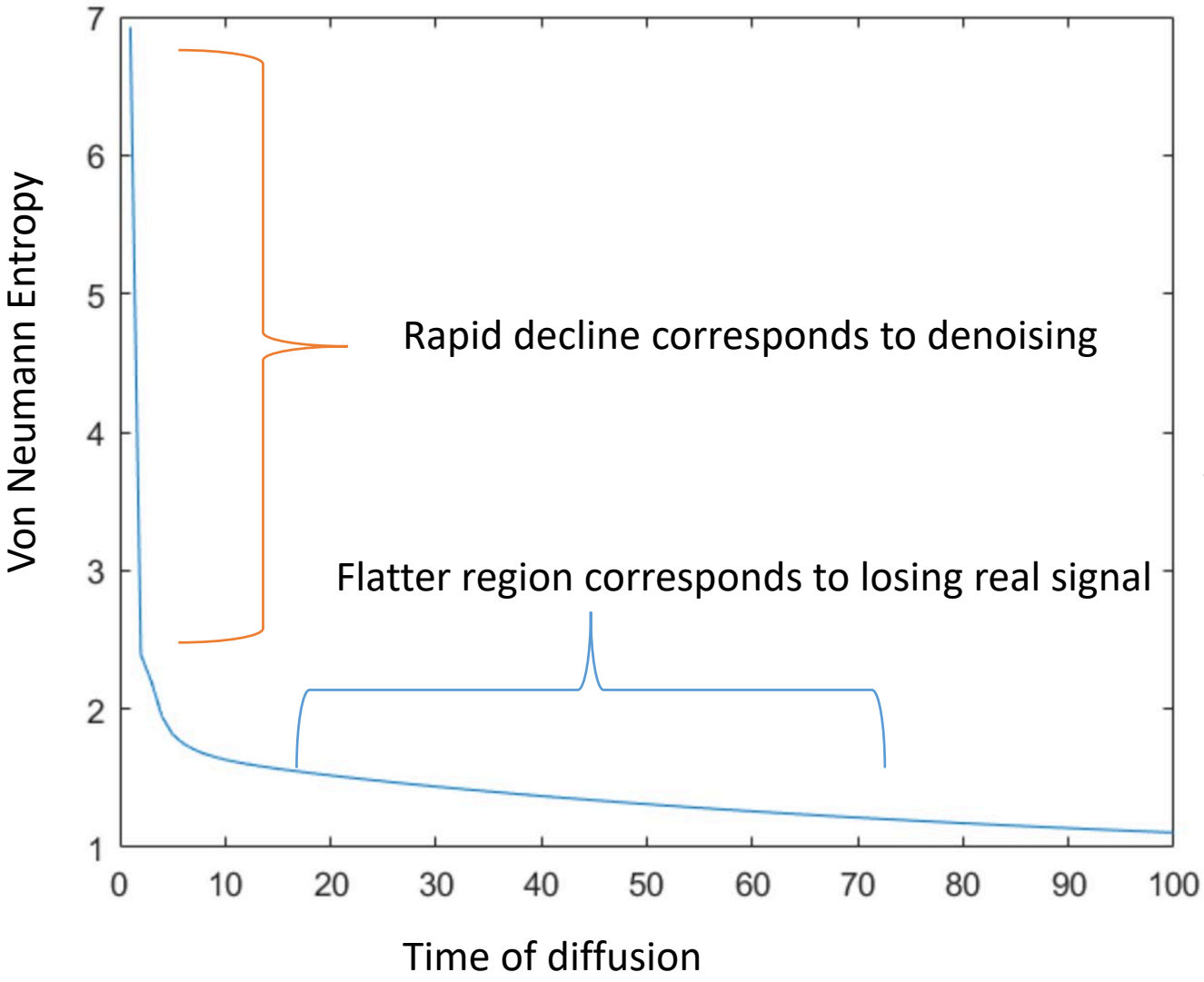
- Von Neumann entropy (von Neumann, 1932) of diffused operator P^t

$$VNE(P^t) = -\sum_j \eta_j \log \eta_j \, , \eta_j = \lambda_j^t / \left|\left|\boldsymbol{\lambda}^t\right|\right|_1$$

Where $\boldsymbol{\lambda}^t = \{\lambda_0^t, \lambda_1^t, \dots\}$ are the eigenvalues of P^t

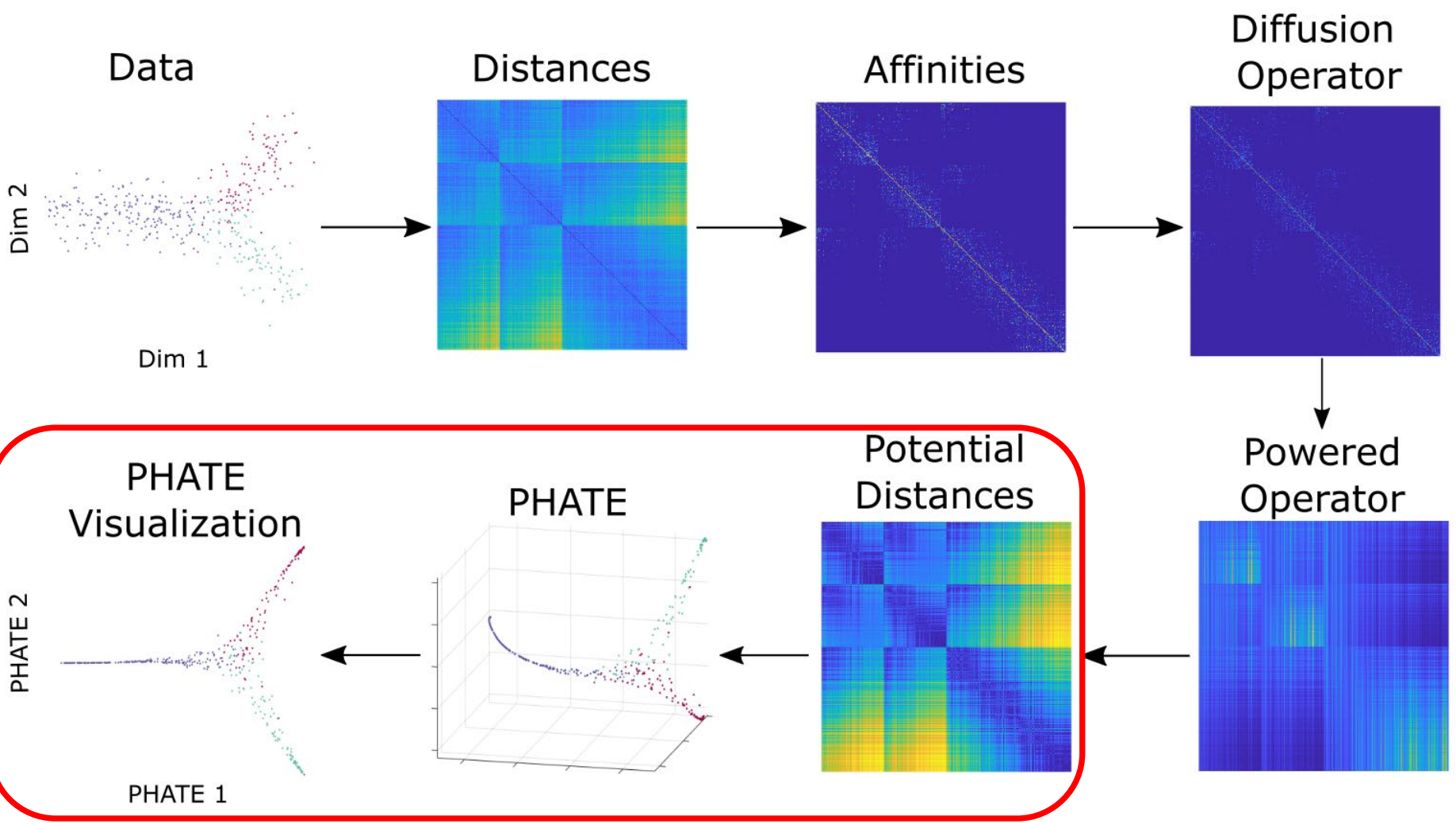
- VNE is a soft proxy of numerical rank
- Decays as diffusion time increases, $\lim_{t\to\infty} VNE(P^t) = 0$
- Use rate of decay to choose time scale

How much diffusion?



Embedding is visually robust in the flatter region

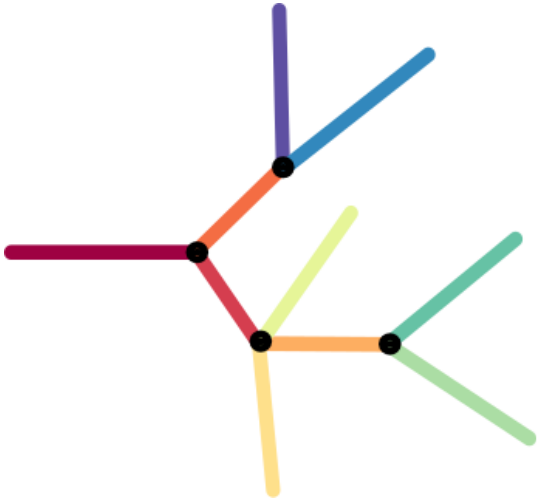
The PHATE Algorithm

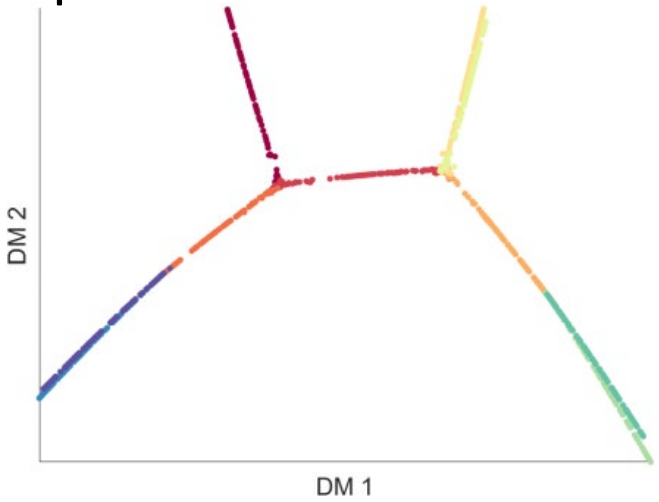


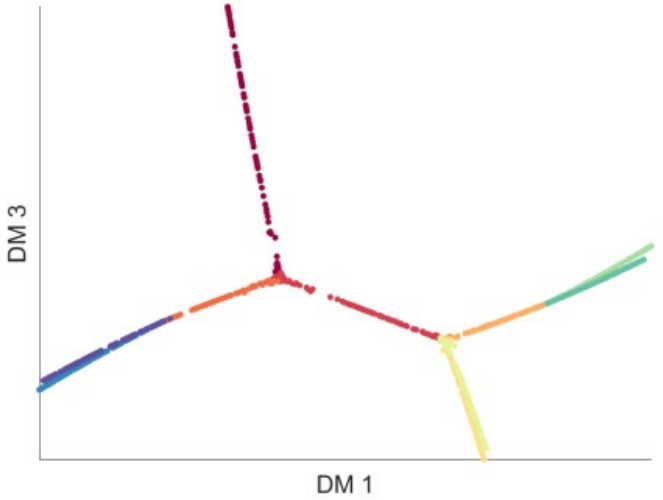
Transform to low dimensions

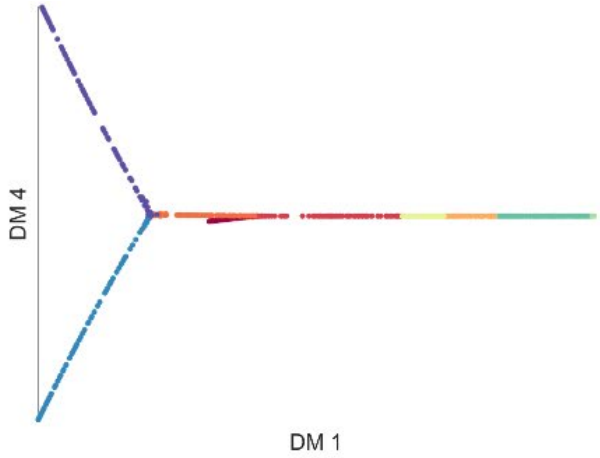
Potential Distances

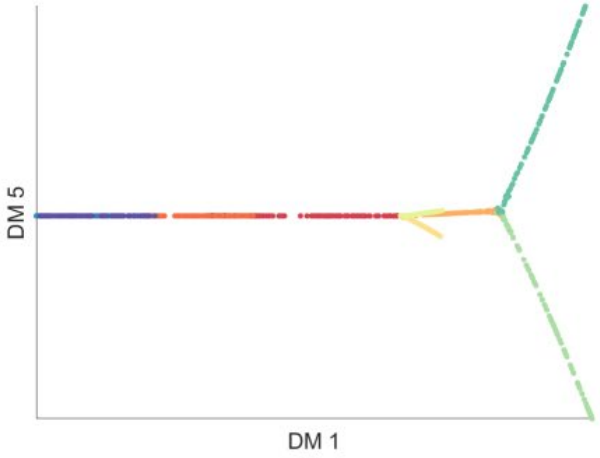
- Diffusion operator contains global and local structure
  - Encoded in multiple dimensions

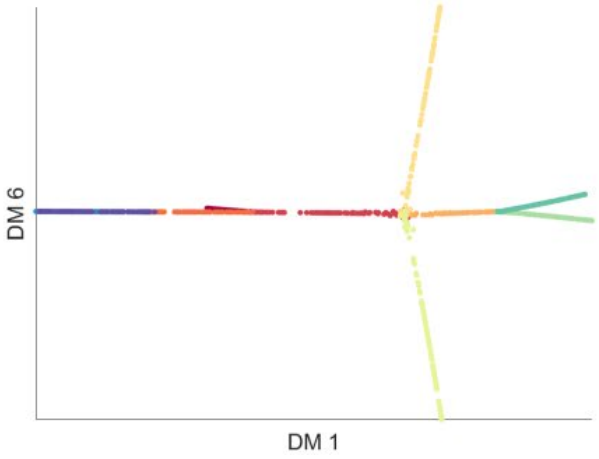






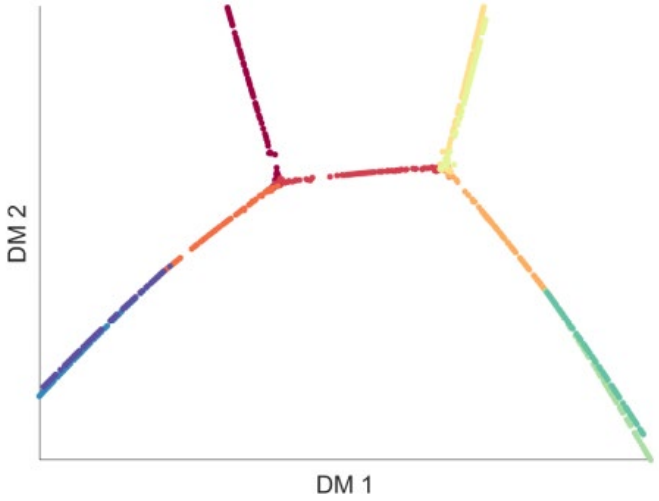


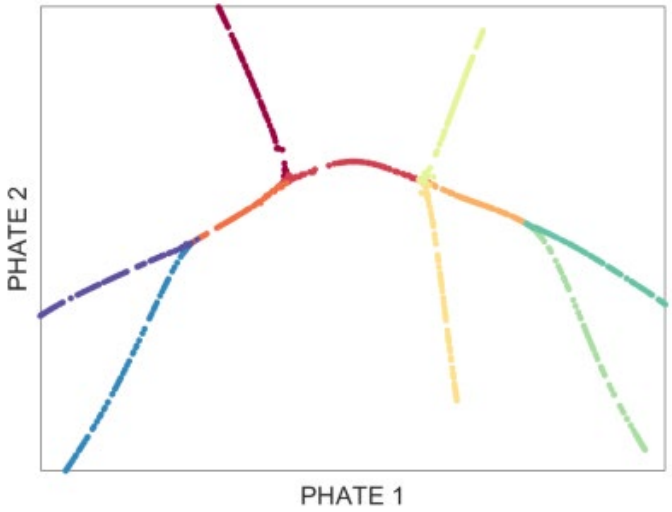




Potential Distances

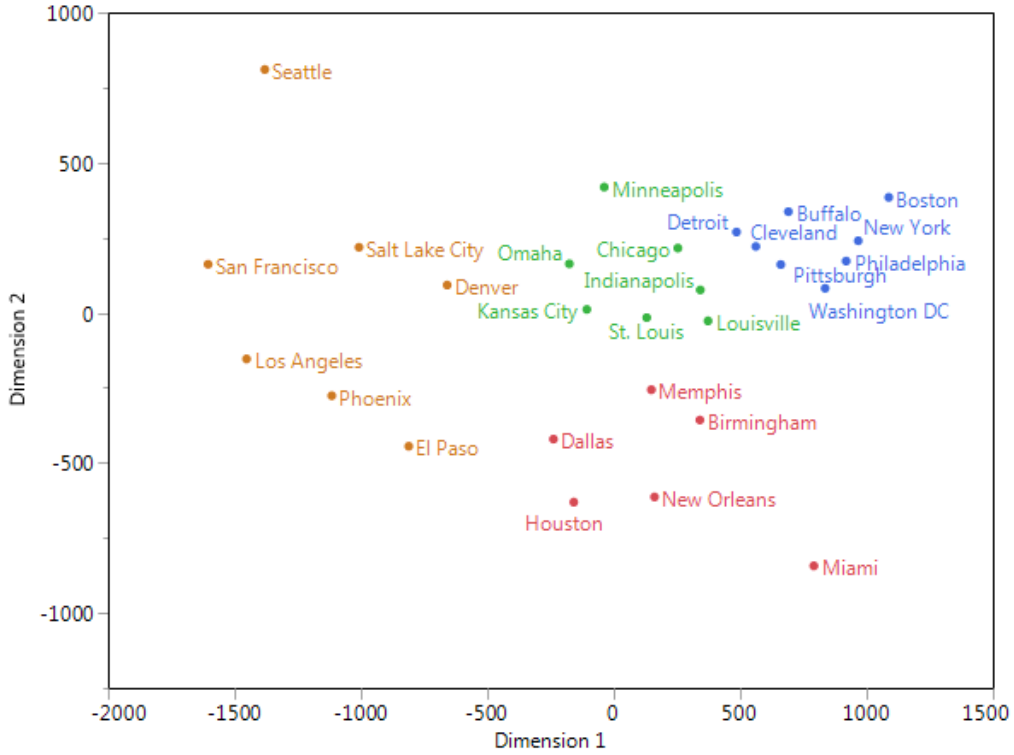
- Diffusion operator contains global and local structure
  - Encoded in multiple dimensions
- To extract this information, we transform diffused probabilities using a ***potential transformation***
  - Forms an information distance between diffused probabilities
  - Connected to heat potential
- Embed for visualization using multidimensional scaling (MDS)





MDS Review

- MDS is a class of dimensionality reduction methods
- **Goal**: minimize the "difference" between corresponding distances in the high and low dimensional spaces
- **Example**: projecting cities on a 3D Earth to a 2D map



MDS Review

- Different measures of "difference" give different algorithms
- Let $D^{(2)}$ be a matrix of squared pairwise distances between n points in the high-dimensional space
- Let $J = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$
  - $I =$ identity matrix, $\mathbf{1} = n$-dimensional vector of ones
- Set $B = -\frac{1}{2}JD^{(2)}J$
- Classical MDS (CMDS) minimizes the following:

$$Strain(\hat{\boldsymbol{x}}_1, \dots, \hat{\boldsymbol{x}}_n) = \sqrt{\frac{\sum_{i,j}\left(B_{ij} - \langle \hat{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_j \rangle\right)^2}{\sum_{i,j} B_{ij}^2}}$$

  - The $\hat{\boldsymbol{x}}_i$ are the low-dimensional coordinates
  - Can be solved using an eigendecomposition of B

Metric MDS Review

- Let D be a matrix of pairwise distances between n points in the high-dimensional space
- Metric MDS has a different "stress" function to minimize:

$$Stress(\hat{\boldsymbol{x}}_1, \dots, \hat{\boldsymbol{x}}_n) = \sqrt{\frac{\sum_{i,j}\left(D_{ij} - \left\|\hat{\boldsymbol{x}}_i - \hat{\boldsymbol{x}}_j\right\|\right)^2}{\sum_{i,j} D_{ij}^2}}$$

- Solving this requires an iterative approach (like gradient descent)
- Generally gives better visualization than CMDS for PHATE
- Nonmetric MDS also exists
  - Input "distances" do not need to be a true distance

Full PHATE Algorithm

Input: Data matrix X, neighborhood size k, locality scale α, desired embedding dimension m (usually 2 or 3 for visualization)

Output: The PHATE embedding Y_m

1: $D \leftarrow$ compute pairwise distance matrix from X
2: Compute the k-nearest neighbor distance $\varepsilon_k(x)$ for each column x of X
3: $K_{k,\alpha} \leftarrow$ compute local affinity matrix from D and ε_k (see Eq. 3)
4: $P \leftarrow$ normalize $K_{k,\alpha}$ to form a Markov transition matrix (diffusion operator; see Eq. 2)
5: $t \leftarrow$ compute time scale via Von Neumann Entropy (see Eq. 5)
6: Diffuse P for t time steps to obtain P^t
7: Compute potential representations: $U_t \leftarrow -\log(P^t)$
8: $\mathfrak{V}^t \leftarrow$ compute potential distance matrix from U_t (see Eq. 6)
9: $Y_{class} \leftarrow$ apply classical MDS to $\mathfrak{V}^t$
10: $Y_m \leftarrow$ apply metric MDS to $\mathfrak{V}^t$ with Y_{class} as an initialization

Supplemental Table S1: Detailed steps in the PHATE algorithm.

Comparison to Diffusion Maps

Algorithm for Diffusion Maps

1. Compute Euclidean distances
2. Convert distances to local affinity matrix K
   - Classically, the Gaussian kernel with fixed bandwidth was used
3. Create degree matrix D from K (sum the rows of K and create a diagonal matrix D from the sums)
4. Normalize: $M = D^{-1/2} K D^{-1/2}$
5. Diffuse: M^t
6. Obtain diffusion coordinates by performing eigenvalue decomposition of M^t

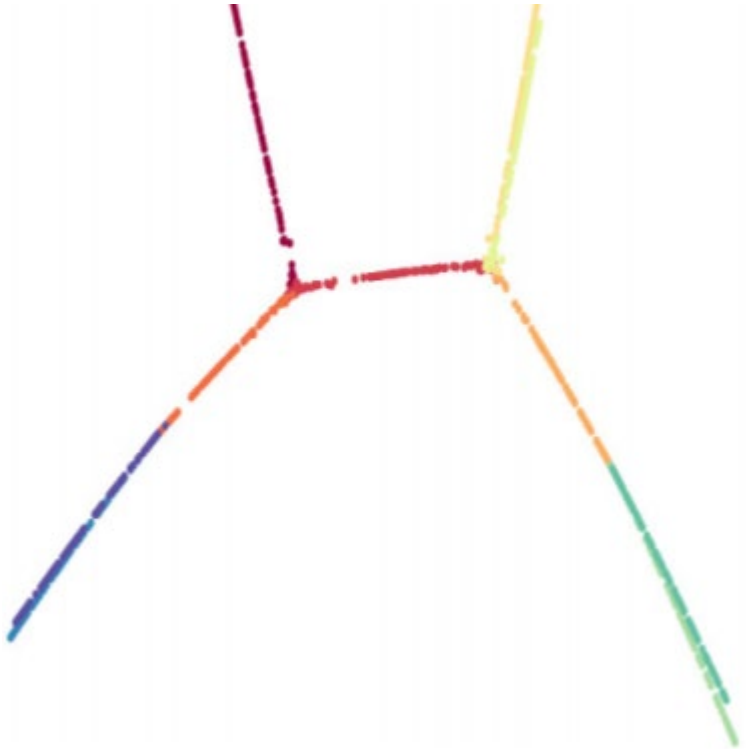
Comparison to Diffusion Maps

- DM captures the information accurately and robustly, but not for visualization
- Thus the PHATE modifications are necessary for visualization

Artificial Tree Data

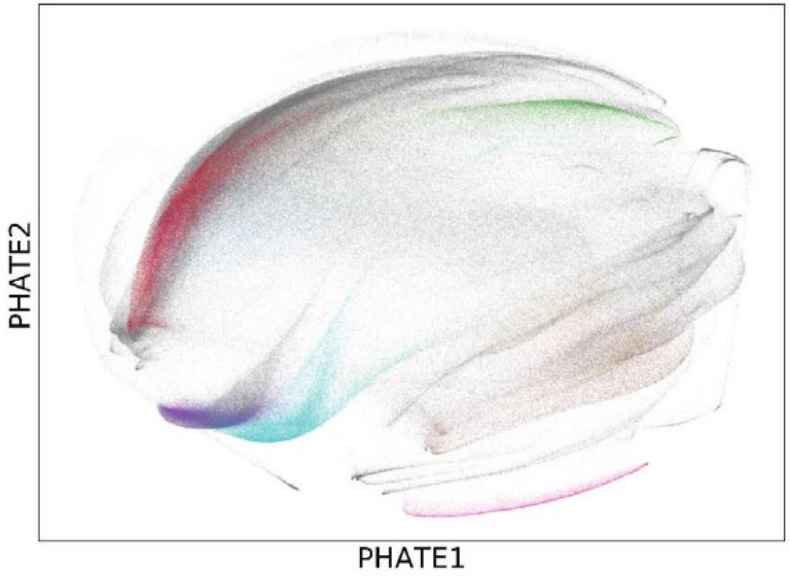
1400 points, 60 dimensions

Diffusion Maps



Scalability of PHATE

- Storing and performing operations on the diffusion matrix can be difficult for large samples
- Can reduce computation by diffusing through "landmarks"
  - Landmarks are chosen by clustering
  - Obtain an embedding of all points by projection

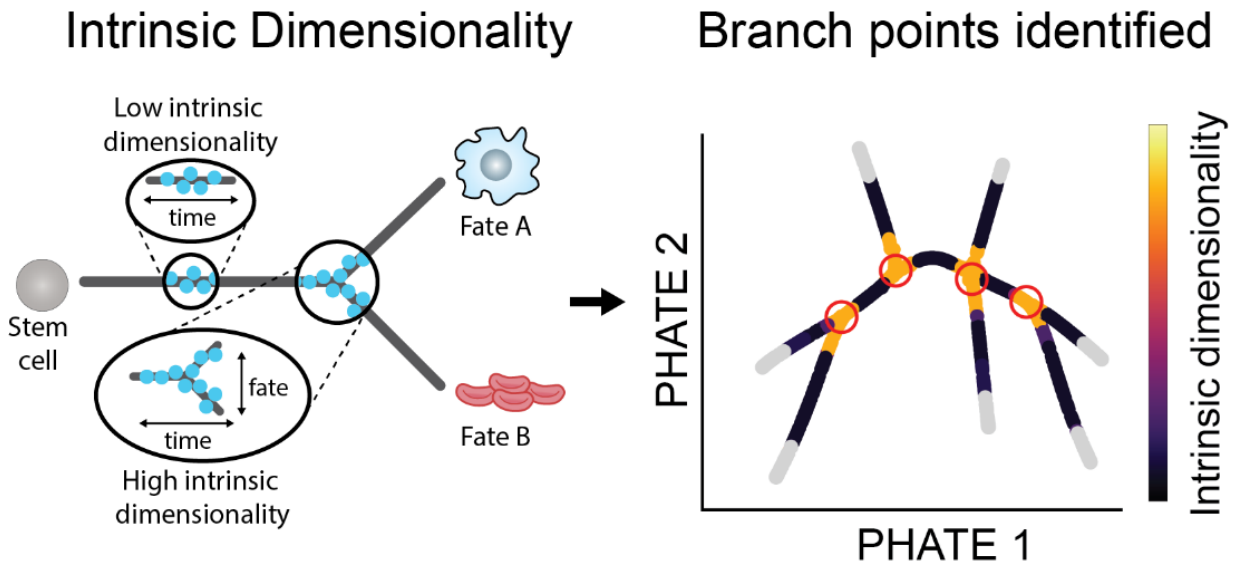


PHATE applied to the 10x megacell mouse brain data (>1 million cells)

Branch Extraction

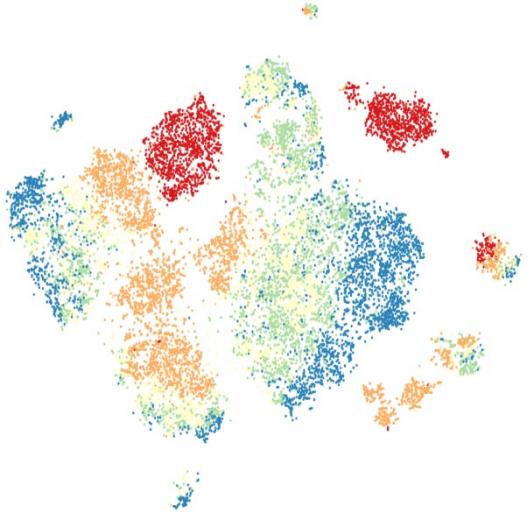
Branch-point detection:

- Local intrinsic dimensionality estimate (Carter et al., 2010)



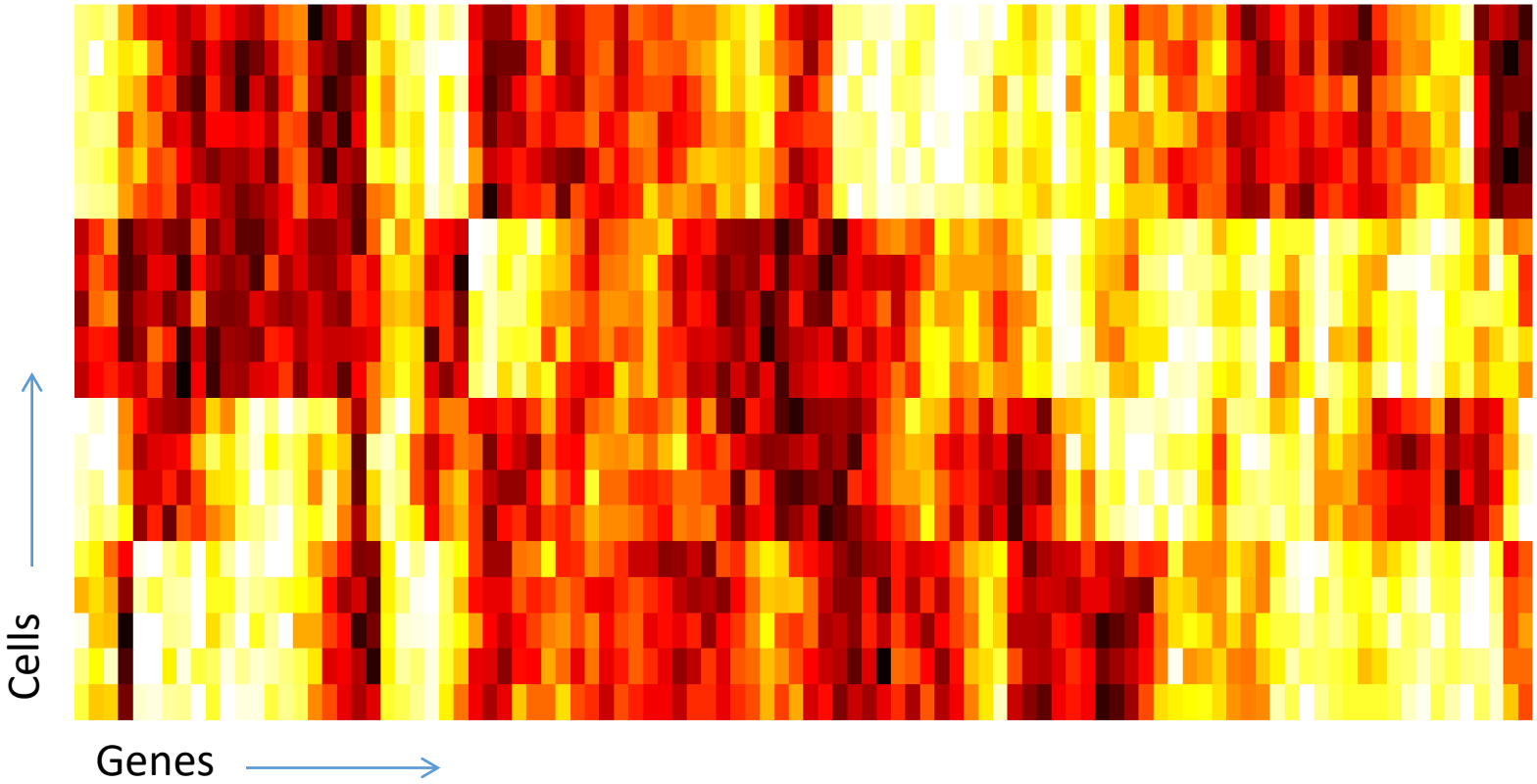
PHATE Summary

- Data have structure at different scales
  - Local branching structure
  - Global relationship between branches
- Existing visualization methods fail to account for all scales
- PHATE captures both global and local structure
  - Innovative diffusion process
  - Selection of t
  - Potential transformation
  - Branch detection
- PHATE reveals new biology

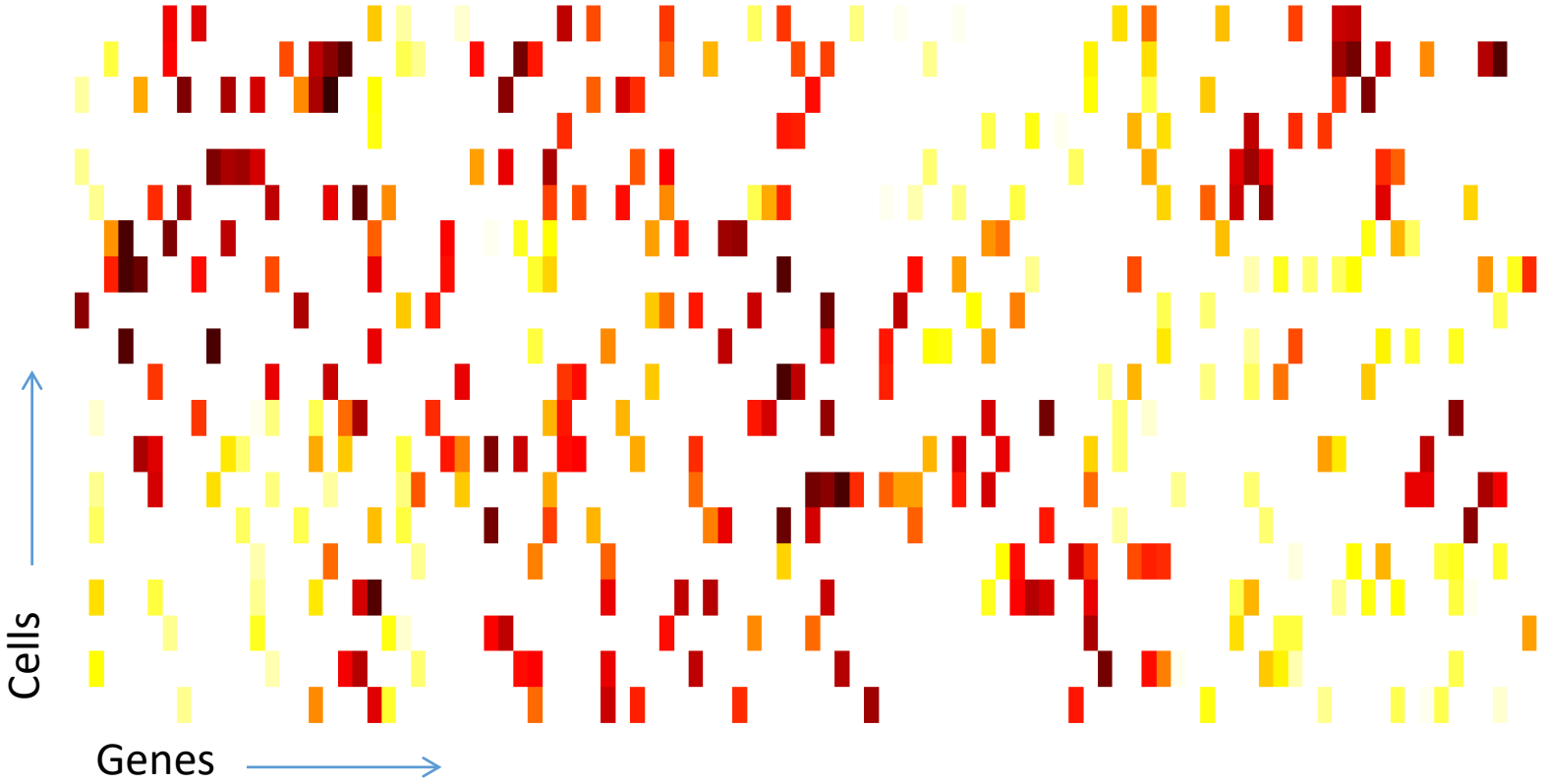


Data Imputation with MAGIC

scRNA-seq data: High Dimensional but Noisy

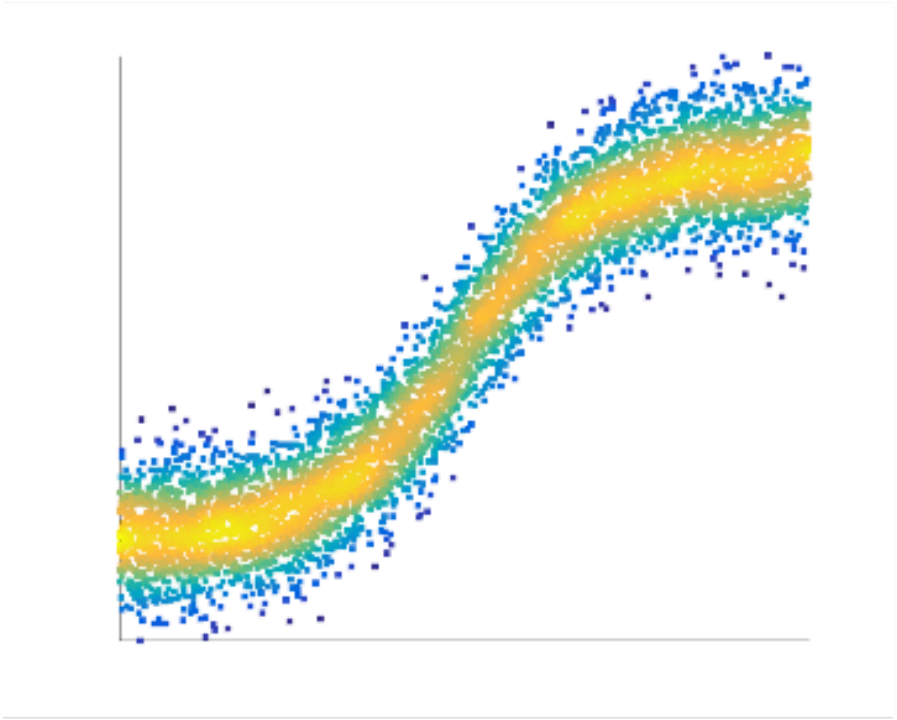


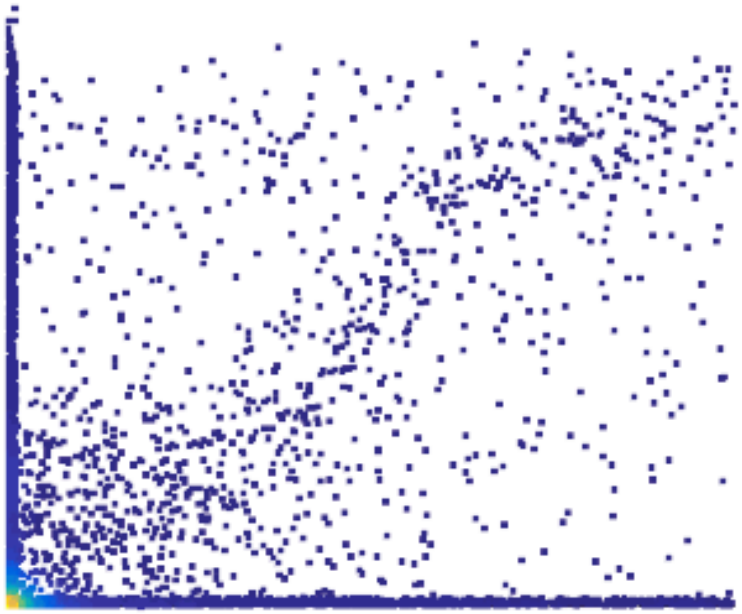
Dropout in scRNA-seq data



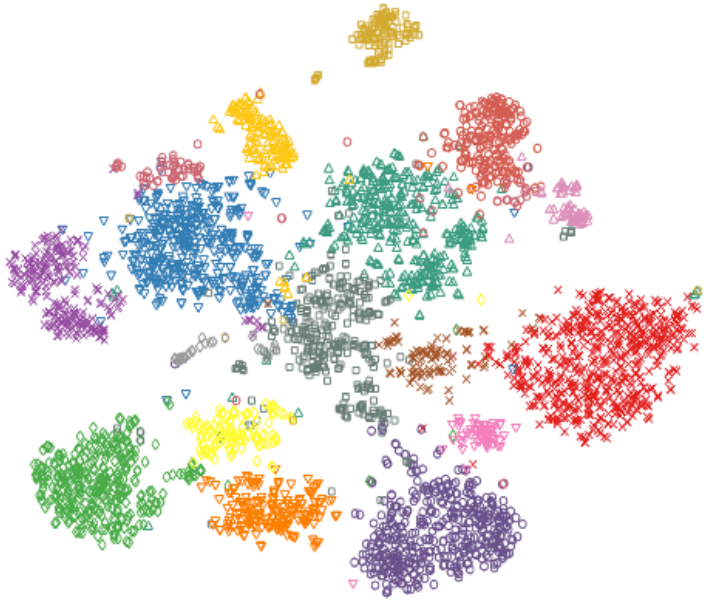
- Obscuring biological structure
- Dropout creates sparse data
- Hard to learn gene interactions

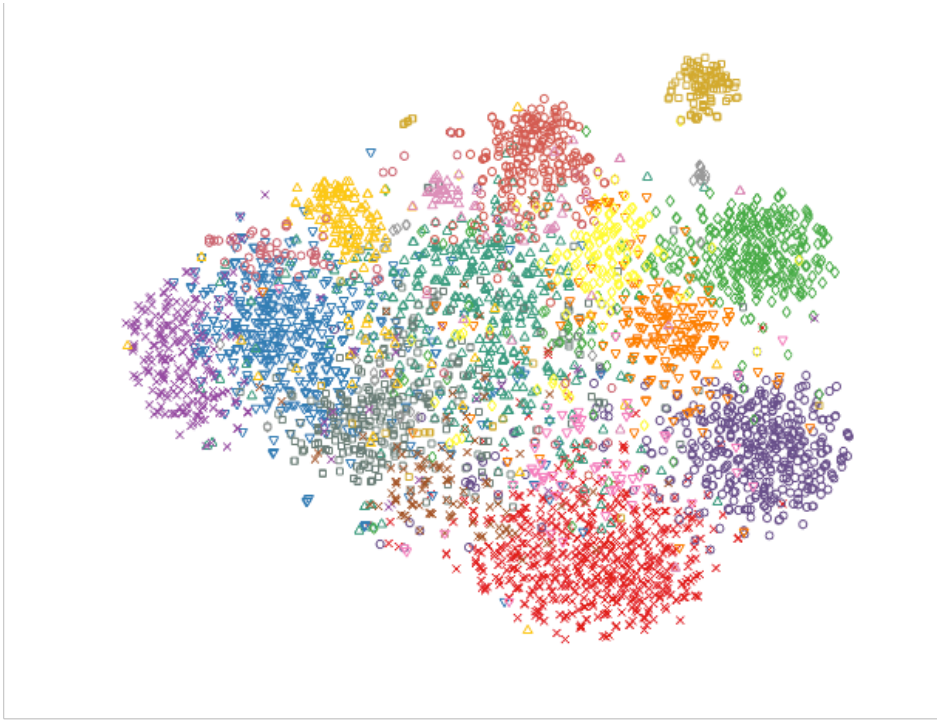
Dropout destroys gene-gene relationships





Dropout destroys clusters





Imputation in other settings: Matrix Completion

- Goal: fill in the missing entries of a partially observed matrix M
- **Example**: Netflix movie-ratings matrix
  - Movies along one dimension, users along the other
  - Can we predict the unobserved rating of a movie by a given user?
    - Used to recommend movies
- **Low rank matrix completion**: find the lowest rank matrix X which matches the matrix M in the set E of the observed entries

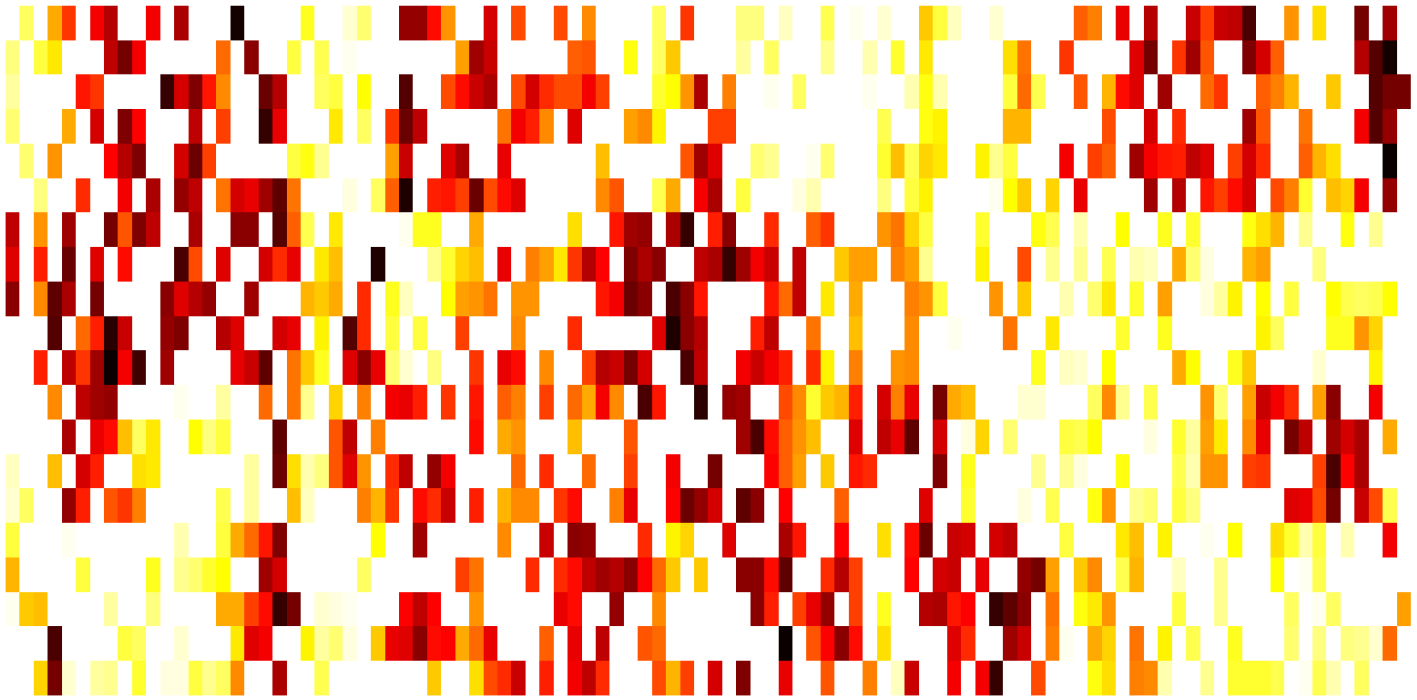
$$\min_{X} \quad rank(X)$$
$$s.t. \quad X_{ij} = M_{ij} \quad \forall i,j \in E$$

Low rank matrix completion

$$\min_{X} \quad rank(X)$$
$$s.t. \quad X_{ij} = M_{ij} \quad \forall i,j \in E$$

- Assumptions:
  - Uniform sampling of observed entries
  - Lower bound on number of observed entries
  - Incoherence (singular vectors of M are not too sparse)
- Problem: this optimization problem is NP-hard
  - Can get around this by approximating the optimization problem with a convex relaxation (e.g. replace the non-convex functions with convex functions that are upper bounds)
- Another problem: this is a linear method
  - Most biological processes are nonlinear

MAGIC: Impute by learning from similar cells



- Biologically similar cells will have similar transcriptomes
- Cells “exchange” information

How MAGIC works

- Compute the diffusion operator $P_{k,\alpha}$ from the scRNA-seq data matrix X
  - Each row of X is a cell and each column of X is a gene
- Diffuse the operator with time step t
- Multiply the data by the diffused operator:

$$X_{MAGIC} = P_{k,\alpha}^{t} X$$

  - Simultaneously imputes and denoises the data

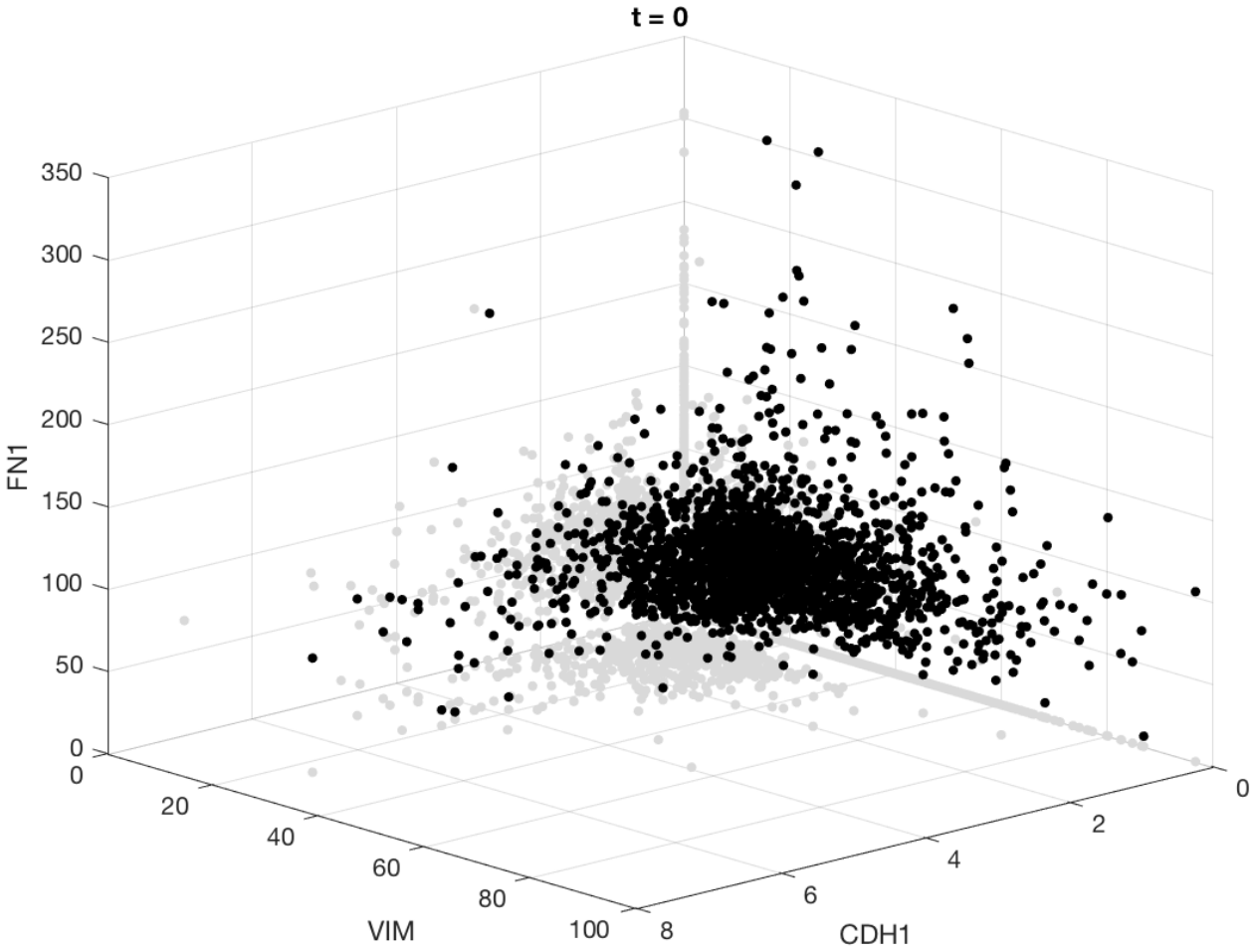
- Published at *Cell* (van Dijk et al., 2018)

Group Exercise

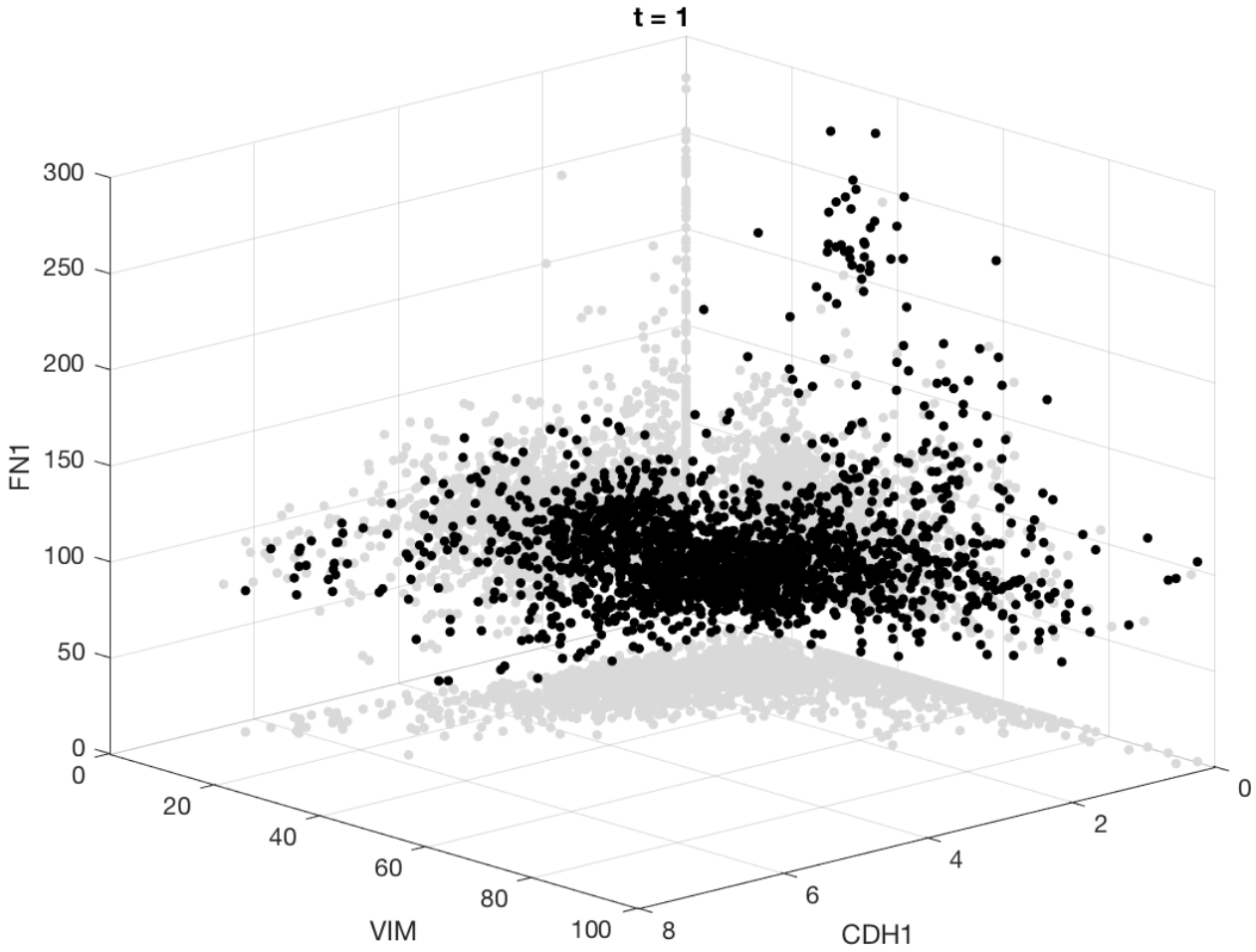
1. Explain why multiplying the data matrix X by the diffused operator $P_{k,\alpha}^{t}$ is equivalent to replacing a cell with a weighted average of its neighbors.

1. Each row of $P_{k,\alpha}^{t}$ sums to 1 with high values assigned to cells that are similar to the cell in the corresponding row by design (remember $P_{k,\alpha}^{t}$ is constructed based on a measure of similarity). Multiplying by $P_{k,\alpha}^{t}$ then replaces each row with a weighted average of the expression levels of the most similar cells in the dataset.

Example

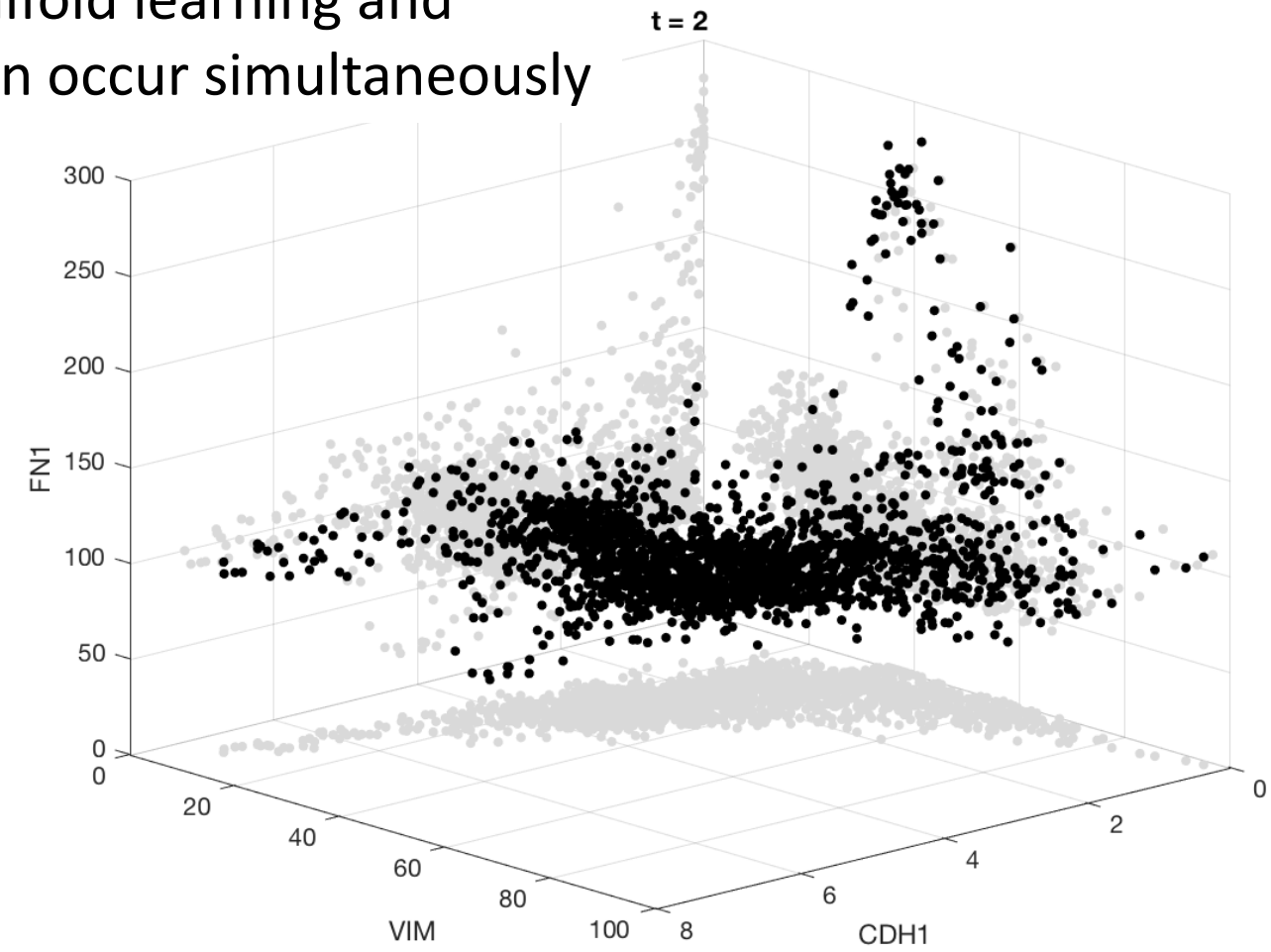

t = 0
FN1
350
300
250
200
150
100
50
0
0
20
40
60
80
100
VIM
8
6
4
2
0
CDH1

Example

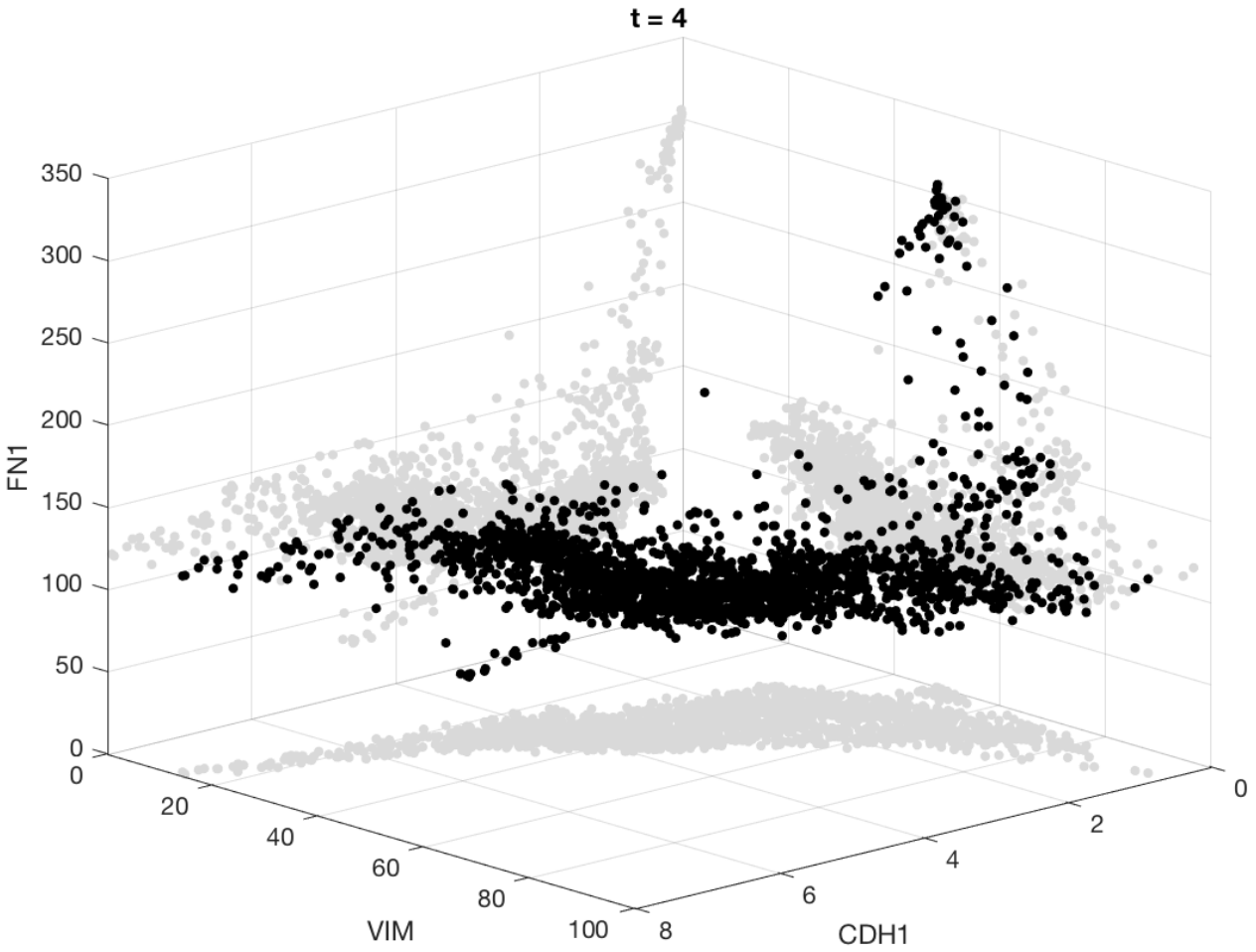


Example

Manifold learning and imputation occur simultaneously

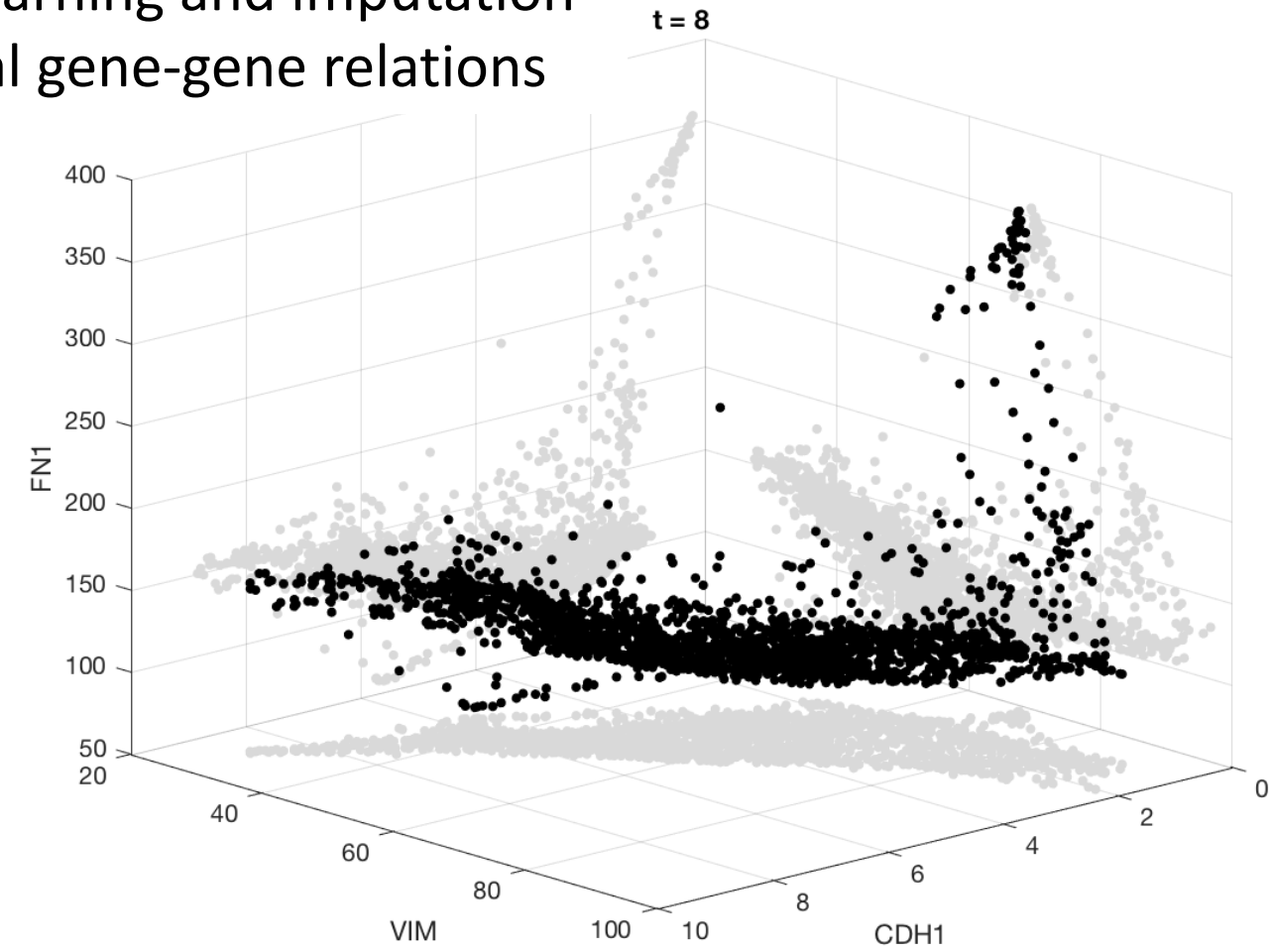


Example


t = 4
FN1
350
300
250
200
150
100
50
0
0
20
40
60
80
100
VIM
8
6
4
2
0
CDH1

Example

Manifold learning and imputation help reveal gene-gene relations



Validate MAGIC by artificial dropout

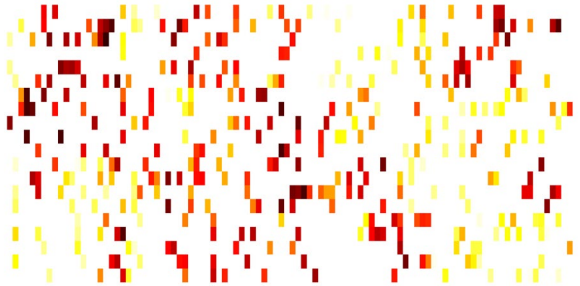
- Test case where we know "ground truth" that we can artificially dropout and recover
- Bulk data that lies on a manifold
- Simulate drop out by uniformly randomly reducing molecule count per matrix entry
- MAGIC recovers 60% in the presence of 80% dropout

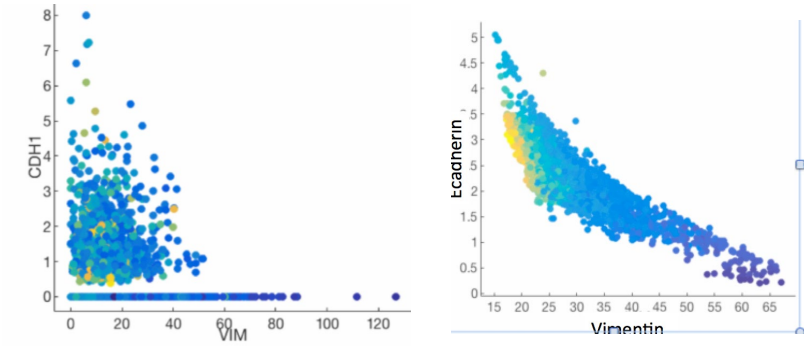
- MAGIC was also used to discover new gene-gene interactions in breast cancer tissue
  - The EMT continuum

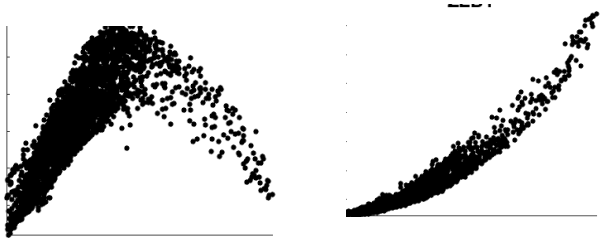
- See the *Cell* paper for details

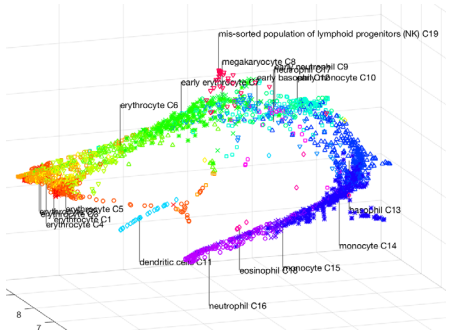
MAGIC Summary

- scRNA-seq data is sparse
- MAGIC recovers structure:
  - Data diffusion process
  - Imputation from robust neighbors
- MAGIC reveals biology:
  - EMT continuum









Final comments on Diffusion methods

- Diffusion-based methods are very useful for learning the underlying structure of the data
  - There is a lot of good theory backing them up
  - Worth digging into if you're interested in manifold learning techniques

Further reading

- PHATE
  - Paper: https://doi.org/10.1101/120378
  - Code: https://github.com/KrishnaswamyLab/PHATE
- MAGIC
  - Paper: https://doi.org/10.1016/j.cell.2018.05.061
  - Code: https://github.com/KrishnaswamyLab/MAGIC
- Diffusion Maps
  - https://en.wikipedia.org/wiki/Diffusion_map
  - https://doi.org/10.1016/j.acha.2006.04.006
- Matrix Completion
  - https://en.wikipedia.org/wiki/Matrix_completion