T-SNE

Machine Learning

Denzel Chuc
Andre Dorantes
Leonardo Cañamar

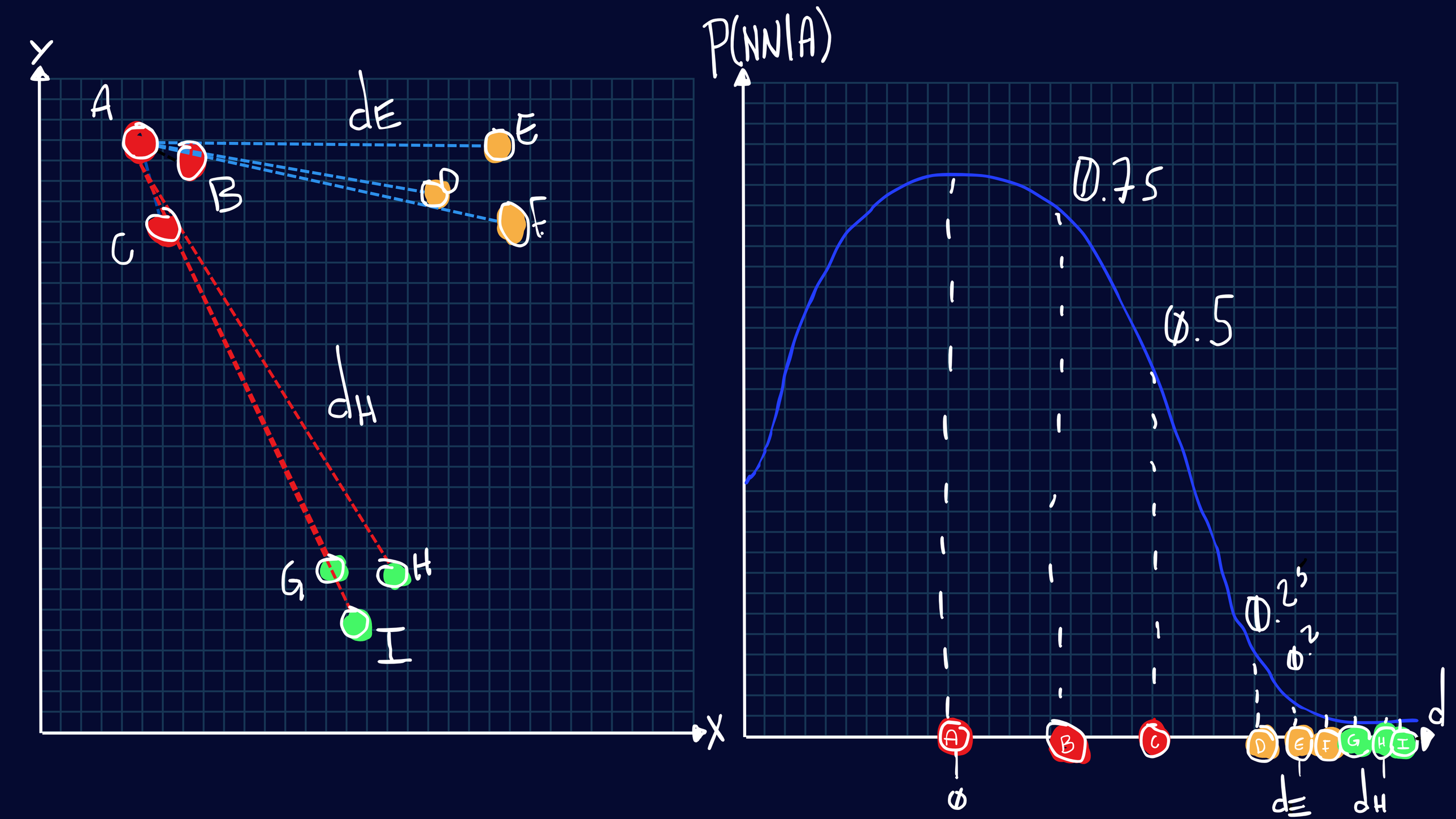
WHAT IS IT?

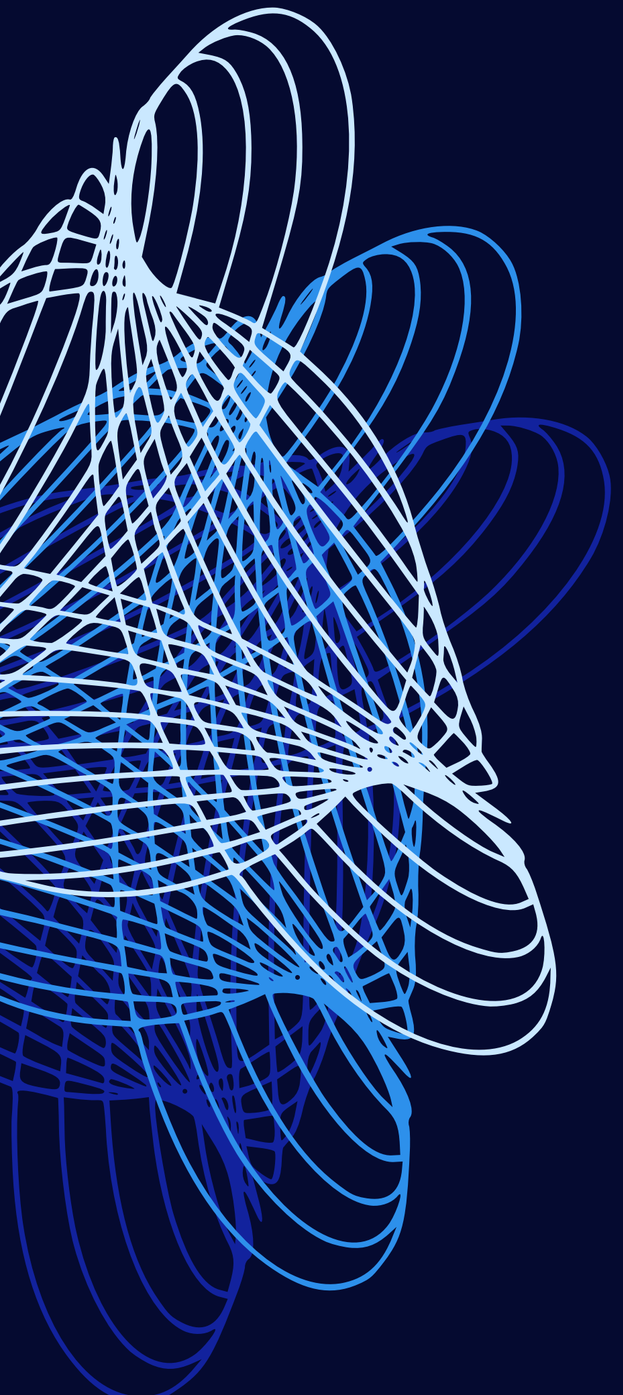
t-distributed Stochastic Neighbour Embedding (t-SNE)

t-SNE is a technique for dimensionality reduction that uses probabilities to encode the properties of the data. These properties are then used in a gradient descent to discover a representation of the data at lower dimensions with the same properties. The properties that are used in this process are called similarity and perplexity.

SIMILARITY

Similarity is a score of how likely any point is to be selected as the nearest neighbour of any given reference point


Y
A
B
C
dE
E
D
F
dH
G
H
I
X
P(NN|A)
0.75
0.5
0.25
0.2
A
B
C
D
E
F
G
H
I
0
dE
dH
d



NORMALISING SIMILARITY

In order to make the similarity easier to work with, we need to mormalise the scores with reference to each other

$$S(P_R) = \frac{P(NN|R)(p)}{\sum P(NN|R)}$$

PERPLEXITY

Perplexity is a relative measure of how packed together the data clusters are, and are typically selected between the values of 5 and 50

According to [3] "Perplexity [as defined] by Van der Maaten & Hinton can be interpreted as a smooth measure of the effective number of neighbours."

GRADIENT DESCENT

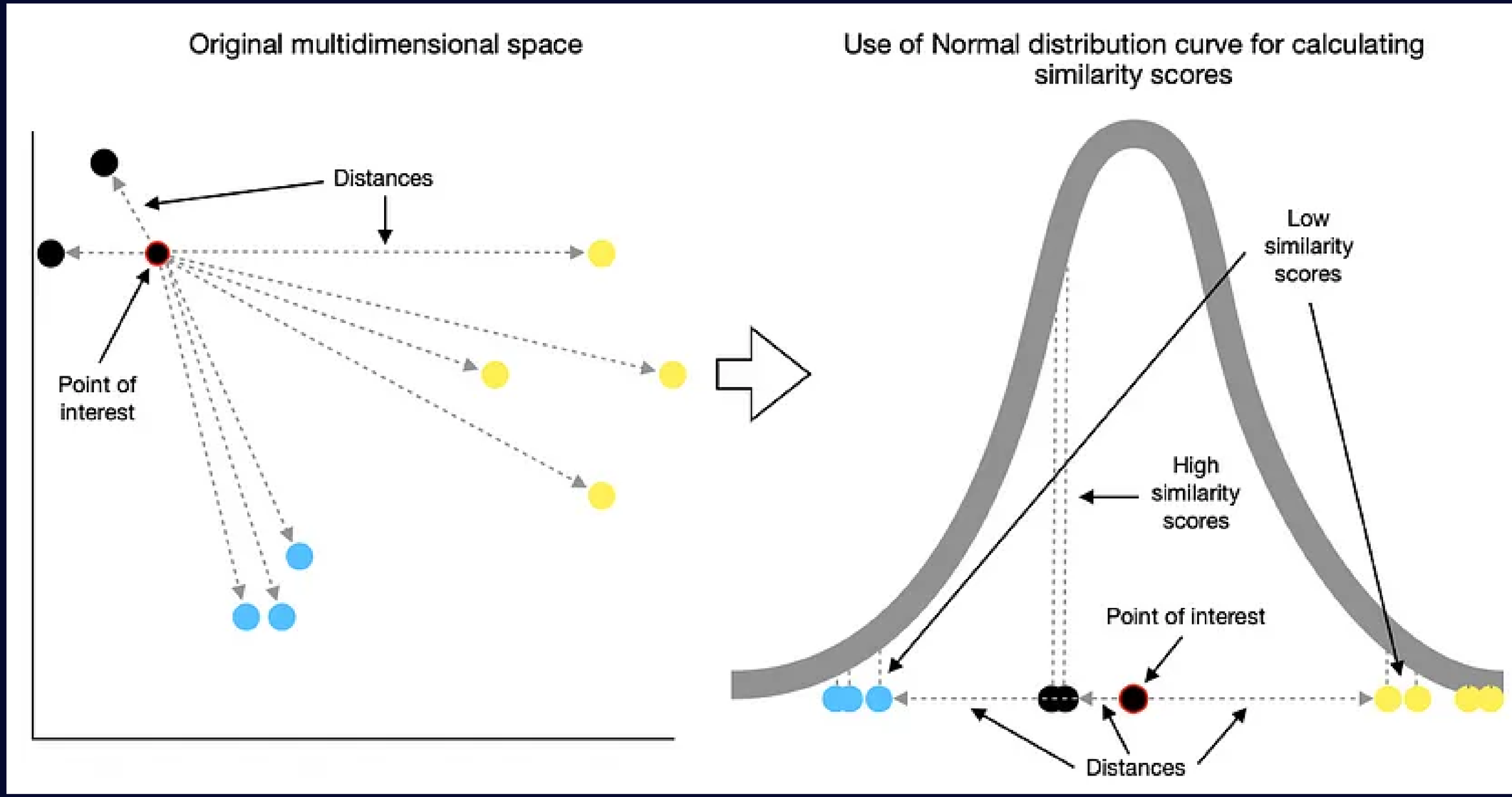
In order to find a lower dimensionality representation of our original data, we can use the gradient descent technique.

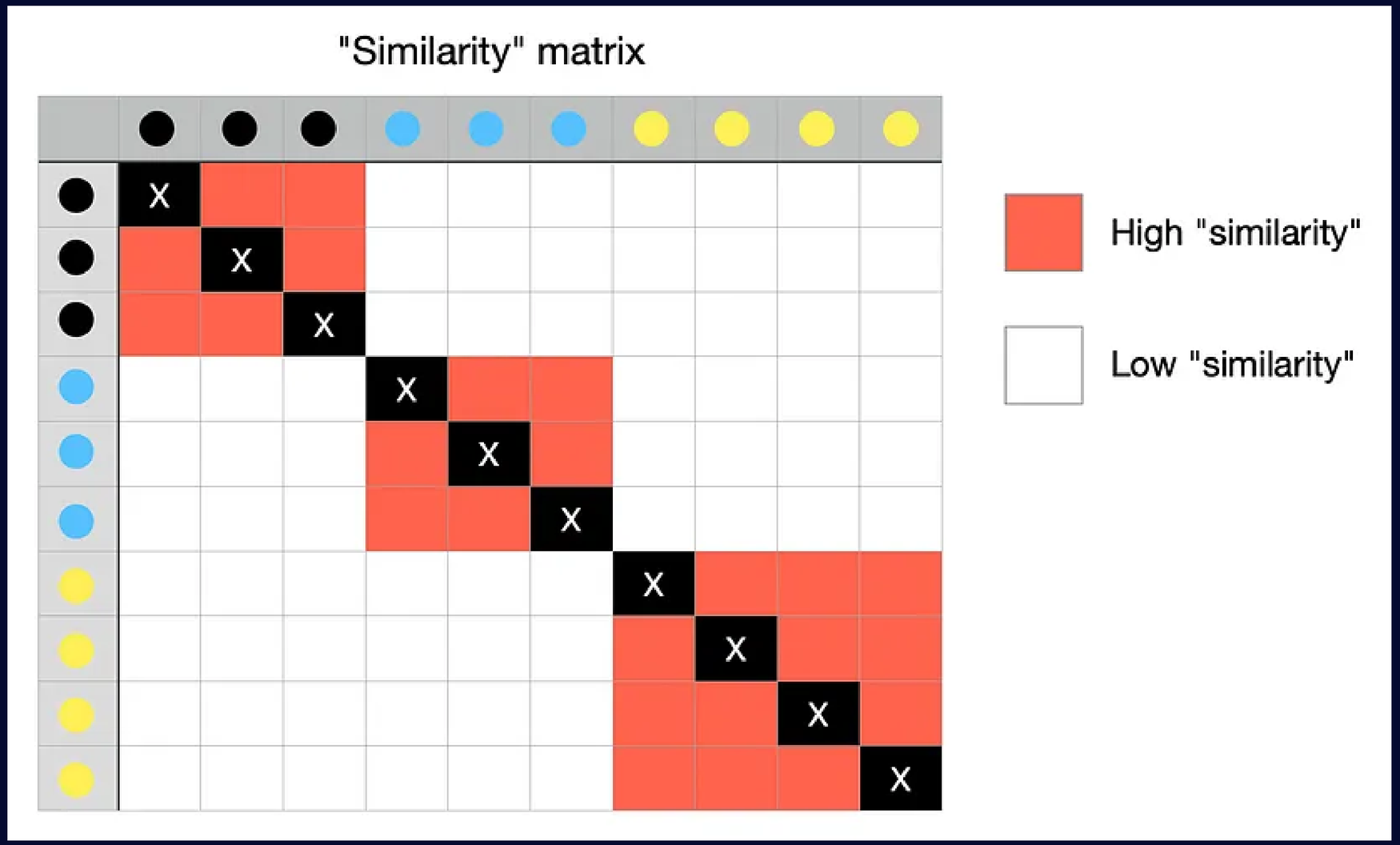
By initialising a random set of data, and calculating how different it is from our goal, we can create figure out how far along the gradient we want to descend, until we reach a point of equilibrium.

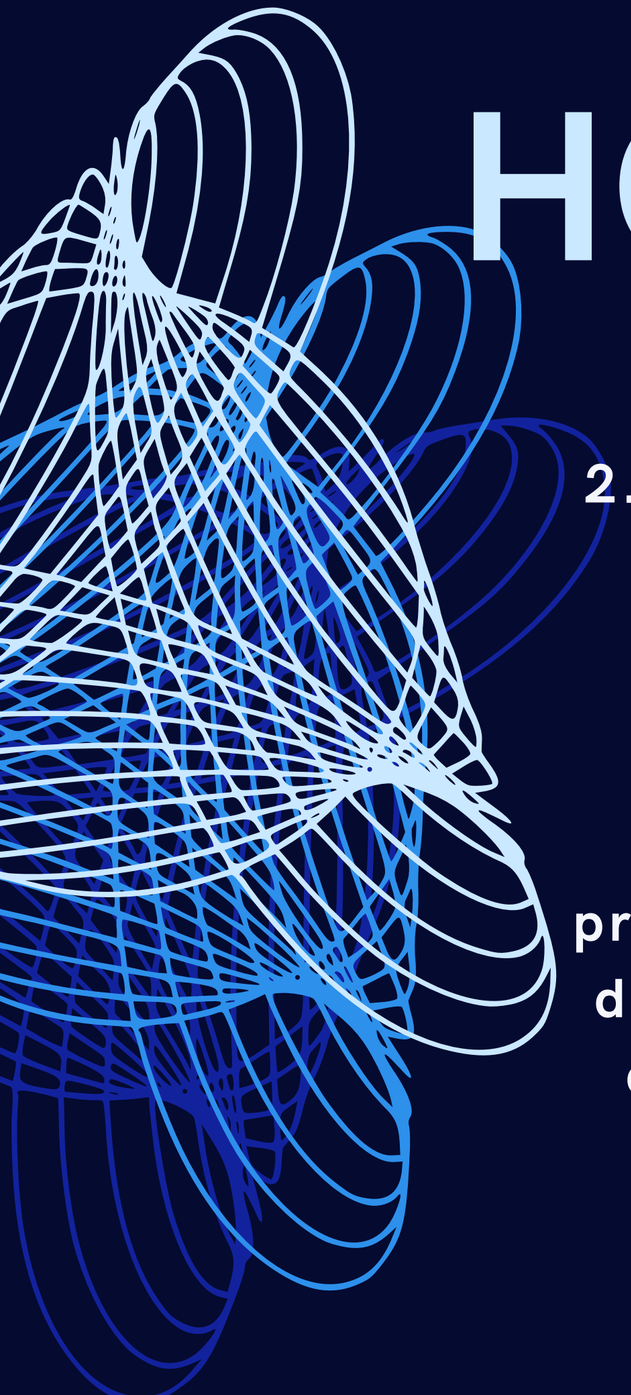
HOW T-SNE WORKS

The t-SNE algorithm finds the similarity measure between pairs of instances in higher and lower dimensional space. After that, it tries to optimize two similarity measures. It does all of that in three steps.

1. t-SNE models a point being selected as a neighbor of another point in both higher and lower dimensions. It starts by calculating a pairwise similarity between all data points in the high-dimensional space using a Gaussian kernel. The points that are far apart have a lower probability of being picked than the points that are close together.


Original multidimensional space
Distances
Point of interest
Use of Normal distribution curve for calculating similarity scores
Low similarity scores
High similarity scores
Point of interest
Distances

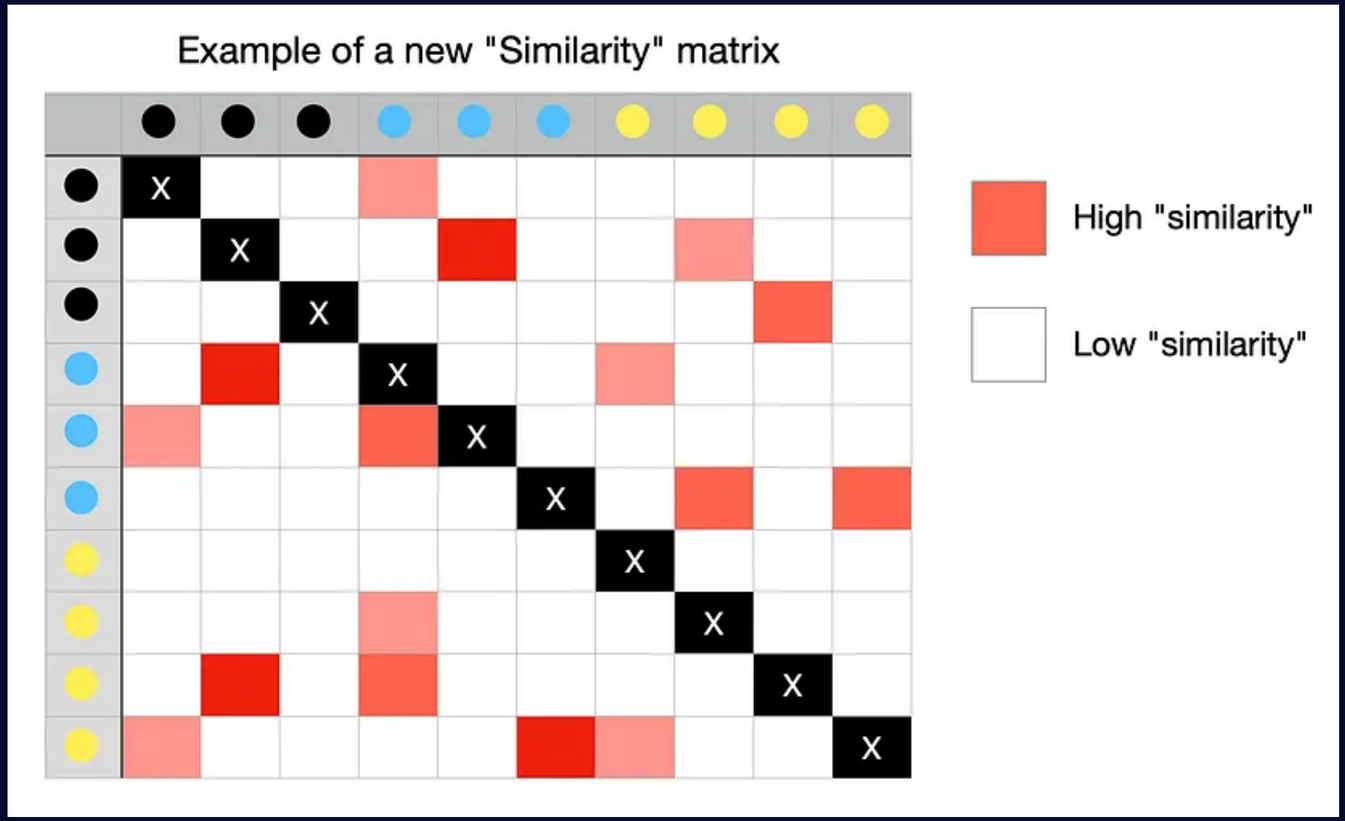

"Similarity" matrix
High "similarity"
Low "similarity"

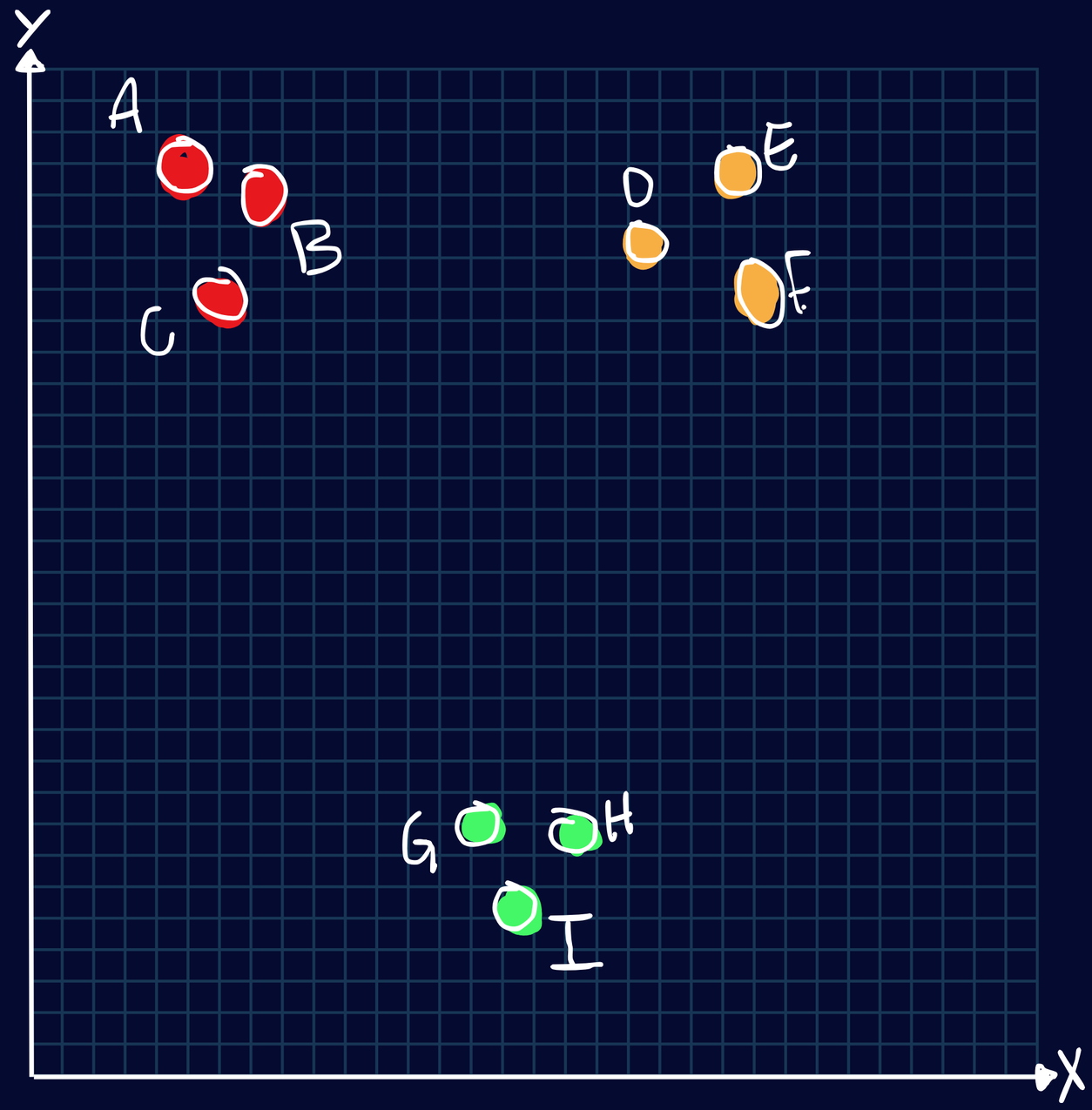


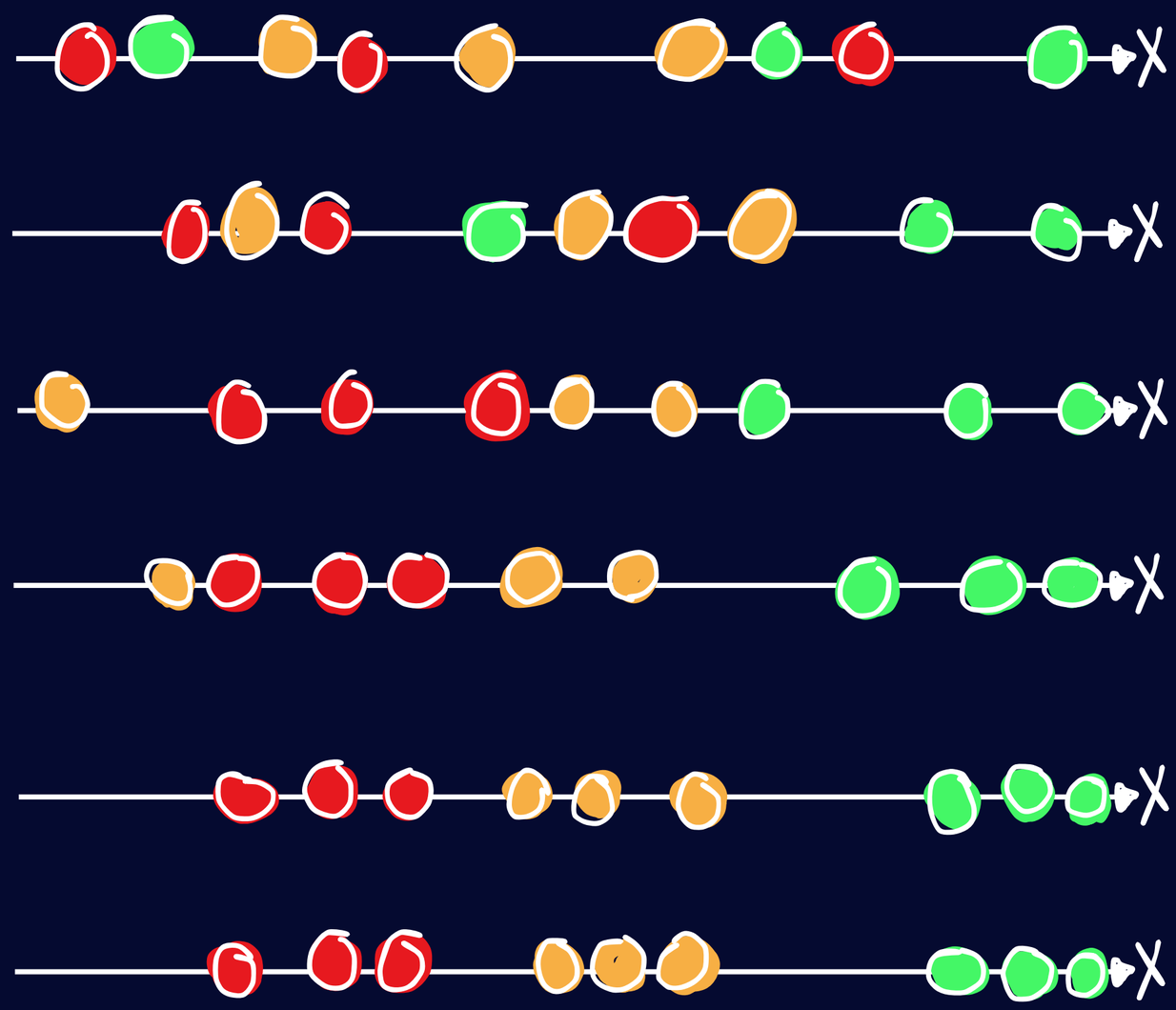
HOW T-SNE WORKS

2. Then, the algorithm tries to map higher dimensional data points onto lower dimensional space while preserving the pairwise similarities.

3. It is achieved by minimizing the divergence between the probability distribution of the original high-dimensional and lower-dimensional. The algorithm uses gradient descent to minimize the divergence. The lower-dimensional embedding is optimized to a stable state.

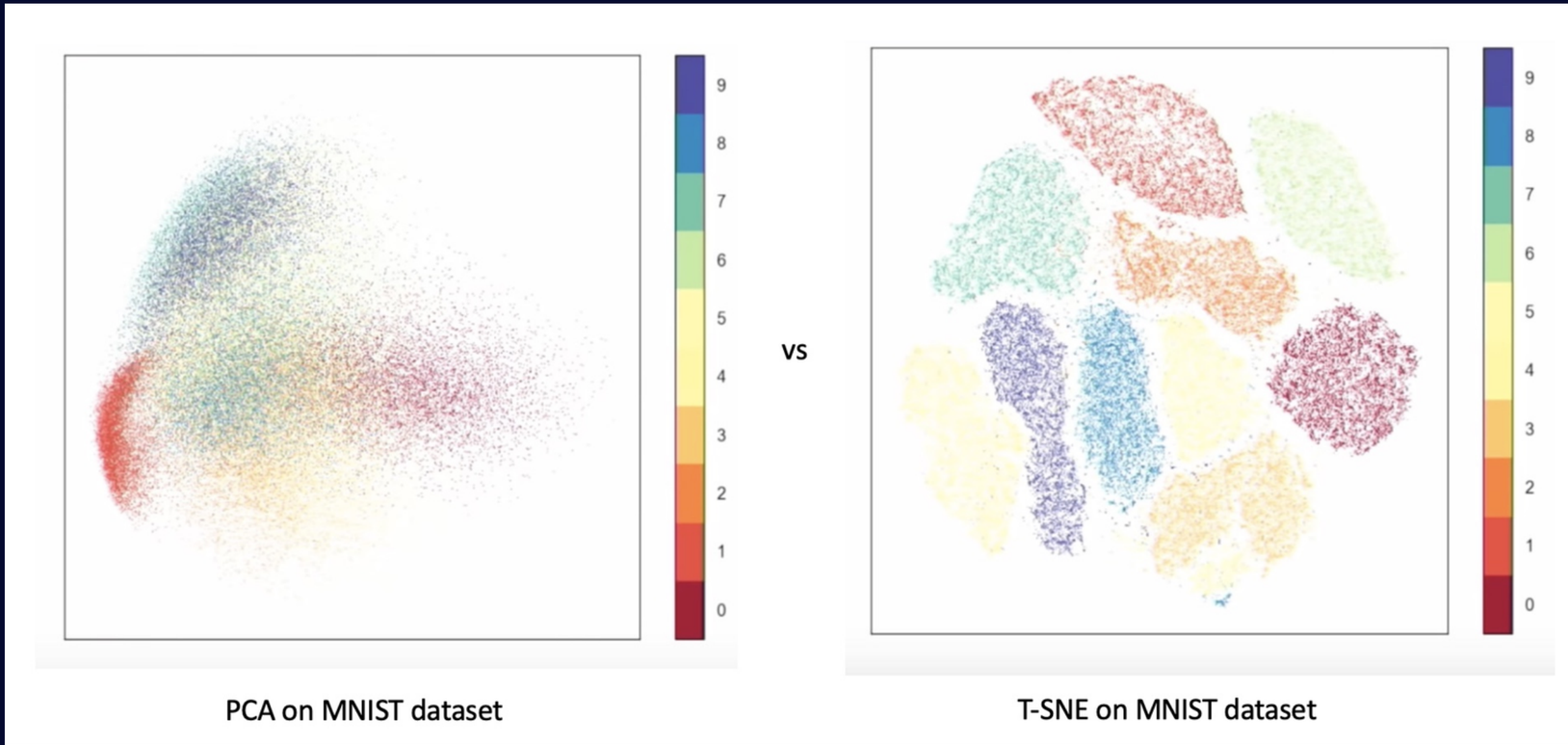

Example of a new "Similarity" matrix
High "similarity"
Low "similarity"


Y
A
B
C
D
E
F
G
H
I
X


X
X
X
X
X
X

COMPARISON WITH PCA

- While PCA is linear, t-SNE is non-linear.
- PCA is deterministic; t-SNE can produce different results for different runs.

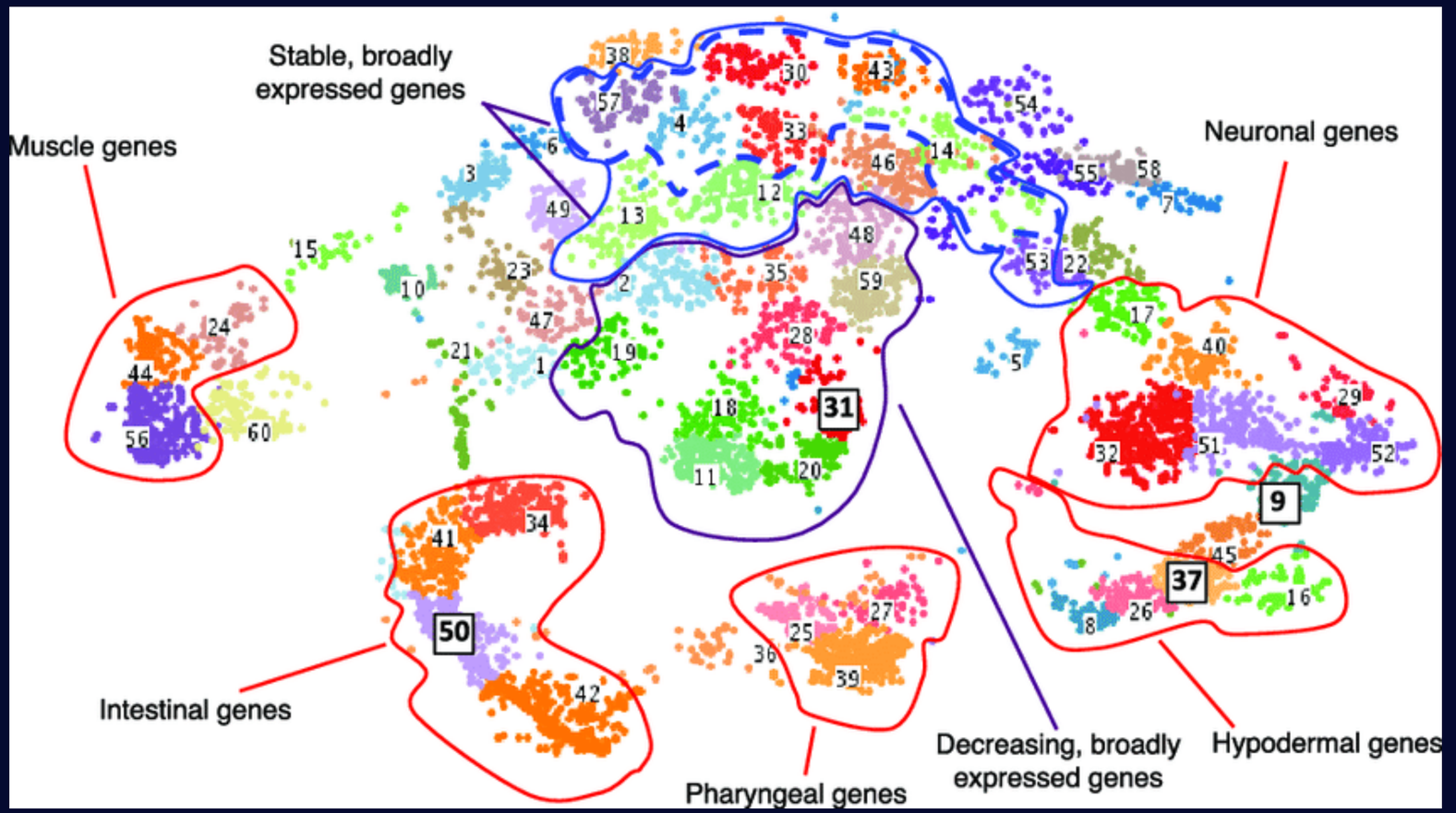


PCA on MNIST dataset

T-SNE on MNIST dataset

APPLICATIONS

- MNIST dataset
- Word embeddings
- Gene expression data



LIMITATIONS

- Computational complexity.
- Randomness: Can produce different results on different runs.
- Might not preserve distances between clusters.
- No guarantee of global structure preservation.

REFERENCES

[1] (burnpiro), K. E. (2022, July 21). T-SNE clearly explained. Medium. https://towardsdatascience.com/t-sne-clearly-explained-d84c537f53a

[2] Agarwal, H. (n.d.). T-SNE (t-distributed stochastic neighbor embedding) algorithm. enjoyalgorithms. https://www.enjoyalgorithms.com/blog/tsne-algorithm-in-ml

[3] Suárez, D. (2022, May 5). Dimensionality reduction - TSNE. Apiumhub. https://apiumhub.com/tech-blog-barcelona/dimensionality-reduction-tsne/#:~:text=A%20second%20feature%20of%20t,effect%20on%20the%20resulting%20pictures.

[4] IBM. (n.d.). What is gradient descent?. IBM. https://www.ibm.com/topics/gradient-descent

[5]Awan, A. A. (2023). Introduction to T-SNE. https://www.datacamp.com/tutorial/introduction-t-sne

[6] T-SNE clearly explained (no date) t-SNE clearly explained - Blog by Kemal Erdem. Available at: https://erdem.pl/2020/04/t-sne-clearly-explained/ (Accessed: 15 September 2023).