

UMAP

What is Dimension reduction?

Dimension reduction, is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some **meaningful properties** of the original data

Dimension reduction algorithms tend to fall into two categories;

- those that seek to preserve the pairwise distance structure amongst all the data samples.

Eg. PCA , MDS, and Sammon mapping.

- those that favor the preservation of local distances over global distance.

Eg. t-SNE, Isomap, LargeVis, Laplacian eigenmaps, UMAP

Why not PCA?

For **visualization**, our humans could only visualize 2 or 3-dimensional plot. Since PCA preserves the global picture of the dataset, if the dataset is complex, it normally remains complex in 2 or 3 PCs space.

Three steps of UMAP:

- 1.Constructs a high dimensional graph
- 2.Constructs a low dimensional graph
- 3.Optimizes the low-dimensional graph to be high dimensional one as similar as possible

Algorithm 1 UMAP algorithm

function UMAP($X, n, d, \text{min-dist}, \text{n-epochs}$)

①

Construct the relevant weighted graph

for all $x \in X$ **do**

fs-set[x] \leftarrow LOCALFUZZYSIMPLICIALSET(X, x, n)

②

Perform optimization of the graph layout

$Y \leftarrow$ SPECTRALEMBEDDING(top-rep, d)

③

$Y \leftarrow$ OPTIMIZEEMBEDDING(top-rep, $Y, \text{min-dist}, \text{n-epochs}$)

return Y

1.Constructs a high dimensional graph

What is the desired graph? A particular weighted k-neighbour Graph

With **neighbors** and **distance (weight)**

How to construct this graph?

- Let $X = \{x_1, \dots, x_N\}$ be the input dataset with metric d
- Find set of nearest k -neighbours $\{x_{i_1}, \dots, x_{i_k}\}$ of x_i under metric d
- Weight function:

$$w((x_i, x_{i_j})) = \exp\left(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right)$$

- $d_{\mathbb{R}^N}(x_i, x_{i_j})$ is the Euclidean distance in \mathbb{R}^N

- $\rho_i = \min\{d(x_i, x_{i_j}) \mid 1 \leq j \leq k, d(x_i, x_{i_j}) > 0\}$ ensures that x_i connects to at least one other data point with an edge of weight 1

- normalisation factor σ_i such that

$$\sum_{j=1}^k \exp\left(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right) = \log_2(k)$$

14

What is the property of the final umap high dimensional graph?

1. Explore final weighted graph

Why ?

- Undirected graph
- The largest weight is always 1
- Sum of weights is no longer $\log_2(k)$
- New sum $\geq \log_2(k)$
- Each point is now connected to at least $k-1$ other points



	A	B	C
A	0	1	0.8
B	1	0	1
C	0.8	1	0

why?

Approximating underlying manifold

- Assume D is uniformly distributed on the manifold M $D \in \mathbb{R}^N$
 - Then a ball of fixed volume V on M should contain the same number of points
 - Conversely a ball centred on point x that contains its k -nearest neighbours has fixed volume regardless of the choice of x



	A	B	C
A	0	1	0.8
B	1	0	1
C	0.8	1	0

\mathbb{R}^N is the high dimensional space

2. Standard steps of optimization:

Initialize low dimension distribution

Sample high and low dimension pairs

Calculate distances of sampled pairs in both high and low dimensions

Calculate the cost of dis-similarity of distances

Minimize the cost

Two different steps in the UMAP to do optimization.

How to Sample?

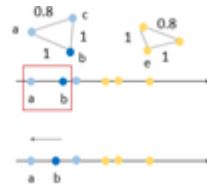
UMAP selects a pair of points **within** a cluster **proportionally** to their high-dimensional weight $P(ab) > P(ac)$

How to Calculate distances in two dimensions ?

- High dimension: weighted graph
- Low dimension: Some form of Euclidean distance

How to calculate the cost of dis-similarity?

- Optimisation problem of finding the low dimensional representation
- $w_h(e)$ weight of edge e in high dimensional case
- $w_l(e)$ weight in low dimensional case
- Cross entropy



$$w_h(e) \log \left(\frac{w_h(e)}{w_l(e)} \right)$$

Take limits as $w_h \rightarrow 1$: w_l will be large to minimise the first term

23

How to Sample?

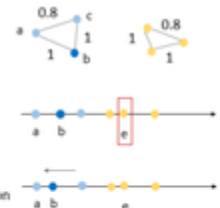
we randomly sample potential edges and assume them to be a negative example (i.e. with weight in high dimension equals to 0)

How to Calculate distances in two dimensions ?

- High dimension: weighted graph
- Low dimension: Euclidean distance

How to calculate the cost of dis-similarity?

- Optimisation problem of finding the low dimensional representation
- Cross entropy



$$(1 - w_h(e)) \log \left(\frac{1 - w_h(e)}{1 - w_l(e)} \right)$$

Take limits as $w_h \rightarrow 0$: w_l is forced to be small to minimise the second term

23