Uniform Manifold Approximation and Projection (UMAP) is a dimension reduction technique that can be used for visualisation similarly to t-SNE, but also for general non-linear dimension reduction. The algorithm is founded on three assumptions about the data:

1. The data is uniformly distributed on Riemannian manifold;
2. The Riemannian metric is locally constant (or can be approximated as such);
3. The manifold is locally connected. (No isolated points)

From these assumptions it is possible to model the manifold with a fuzzy topological structure. The embedding is found by searching for a low dimensional projection of the data that has the closest possible equivalent fuzzy topological structure.

Maths behind that algorithm :

algebraic topology, Riemannian geometry, fuzzy logic

Video (https://www.youtube.com/watch?v=nq6iPZVUxZU):

UMAP is based on Neighbour Graphs Techniques (state-of-the-art technique for neighbour graphs techniques is t-SNE : capture local structures rather than global structure of the data) but can also capture global and local structure of the data.

UMAP is bases on a strong mathematical theories (hard to understand, can’t find the official paper) such as :

1. Topological data analysis : if we built simplices complex in a topological space in a certain way 🡪 rebuild these simplices as combinatorial things without losing information : which is way easier to manpulate (recover all the important topology of that space.
2. Uniform distribution assumption : if the data is not uniformly distributed on the manifolds 🡪 need to define a Riemannian metric on the manifolds to make the assumption True (by varying the notions of distance on the manifolds)

Algorithm wise :

1. When we know the manifolds but not the correct nearest neighbour distance 🡪 use of cross entropy (goal : first get the clumps right (local structure) and then get the gaps right (global structure)). Implemented thanks to RP-trees and NN-descent (so this is efficient and fast even in high dimension)
2. Pace of the algorithm : SGD + negative sampling

Advantages :

1. Faster than t-SNE
2. Can combine spaces with different metrics (can do UMAP for generic Pandas Dataframes)
3. Can do both supervised (if we give the label vector) and unsupervised classification

* See <https://pair-code.github.io/understanding-umap/> pour une explication peut être plus simple et surtout pour des représentations visuelles 3D sur les données Fashion MNIST