0.1 TODO List

0.1.1 Classic Dimensionality Reduction

1. Learn about the following:
   (a) PCA
   (b) MDS
   (c) isomap
2. Maybe need to implement some of them
3. Know the difference from a practical aspect

0.1.2 Clustering

1. Need to learn about the different approaches
   (a) Variational
   (b) Spectral
   (c) Hierarchical
   (d) Density thresholding
   (e) Mode seeking
   (f) Valley Seeking
2. Need to know the specific applications
3. Need to delve into one of them to see if possible for improvement
4. Mathematical details about Mode Seeking
   (a) Hypotheses on the function of f and estimator $\hat{f}$
   (b) Exact conclusions about the prominence gap.
   (c) How to specify prominence parameter τ
   (d) Algorithm details

Persistent Homology

1. Need to delve into the details of the PH algorithms
   (a) Like reading off the intervals.
   (b) Different implementations.

0.2 Clustering

Point cloud (with coordinates)

Distance / dissimilarity matrix

Note: this seems to be a different idea of the distance function used in TDA, which is called lens in that context.

Barcode →merge tree →dendrogram

0.2.1 Mode Seeking Paradigm

Problems:

1. Noisy estimator
2. Neighborhood graph

Solutions:

1. Be proactive: smooth
2. Be reactive: merge clusters after clustering
   This leads to "topological persistence"

Persistence for Model Seeking:
Probability density function f

- Nested family (filtration) of of inverse images, or superlevel-sets $f^{-1}([t, +\infty))$ for t from $+\infty$ to $-\infty$
- Track evolution of "topology"

Similar stability theorem.
Seems to have relation with Morse theory. "If f is Morse, then..."

0.3 Topological Persistence

Persistence diagram shows the "persistence" of the topological features. Slight perturbation causes slight difference in persistence diagrams.

0.4 Homology

Definition: $h : X \mapsto Y$ is a homeomorphism if there exists a map $h^{-1} : Y \mapsto X$, s.t.

- both are continuous
-

0.5 Computation of Persistence Homology

This part is largely cited from [1].
How to keep track of how one feature "merges" to another?
Boundary matrix: the matrix representation of boundary operator.
We also need a total ordering compatible with the "filtration" in the following sense:

- a face of a simplex precedes the simplex.
- a simplex in K_i precedes simplices in K_j for $j > i$, and not in K_i.

– this essentially means that we place the simplices by the order of "appearing".

Standard Algorithm:

- Form the boundary matrix from the ordering.
- Reduction, which is essentially Gaussian elimination.
- Reading off intervals.
  1. some details to do.
  2. degree: $\mathrm{dg}(\sigma) =$ smallest number l s.t.$\sigma \in K_l$
-

0.6 Mapper

Bibliography

[1] N. Otter, M. A. Porter, U. Tillmann, P. Grindrod, and H. A. Harrington, "A roadmap for the computation of persistent homology," *EPJ Data Science*, vol. 6, p. 17, Aug 2017.