1. Abstract

Gaussian affinities are commonly used in graph-based methods such as spectral clustering or nonlinear embedding. Hinton and Roweis (2003) introduced a way to set the scale individually for each point so that it has a distribution over neighbors with a desired perplexity, or effective number of neighbors. This gives very good affinities that adapt locally to the data but are harder to compute.

We study the mathematical properties of these entropic affinities and show that they implicitly define a continuously differentiable function in the input space and give bounds for it. We then devise a fast algorithm to compute the widths and affinities, based on robustified, quickly convergent root-finding methods combined with a tree- or density-based initialization scheme that exploits the slowly-varying behavior of this function. This algorithm is nearly optimal and much more accurate and fast than the existing bisection-based approach, particularly with large datasets, as we show with image and text data.

2. Motivation

The bandwidth \( \sigma \) in Gaussian affinities is data-dependent and usually set using some rule-of-thumb. Too high values result in almost identical interaction between the neighboring points, while too low values result in almost zero interaction. Good results are achieved when:

- **distinct** for every data point,
- **takes into account whole distribution of distances**.

Example: 2D embedding of COIL object images

Entropic Affinities, Constant \( \sigma \), Dist. to 7th neighbor

3. Entropic affinities

In the entropic affinities, the bandwidth is set individually for each point such that it has a distribution over neighbors with fixed perplexity \( \beta \) (Hinton & Roweis, 2003). For a point \( x_i \in \mathbb{R}^d \), consider the posterior distribution of an isotropic Gaussian kernel density estimator of width \( \sigma \) defined on a set \( x_1, \ldots, x_N \). We have a discrete distribution \( p(x, \sigma) \) with probabilities for \( s = 1, \ldots, N \)

\[
p_k(x, \sigma) = \frac{K_k(||x - x_k||/\sigma)}{\sum_{k=1}^{N} K_k(||x - x_k||/\sigma)},
\]

where \( K_k \) is the kernel function. The entropy of the distribution is defined as

\[
H(x, \sigma) = - \sum_{k=1}^{N} p_k(x, \sigma) \log(p_k(x, \sigma)) = - \sum_{k=1}^{N} p_k(x, \sigma) \log K_n + \log \sum_{k=1}^{N} K_k.
\]

For the Gaussian kernel it becomes (define \( \beta = 1/(2\sigma^2) \)):

\[
H(x, \beta) = \beta \sum_{n=1}^{N} p_n(x, \beta) \log(\exp(-d^2_n/\beta)) = \sum_{n=1}^{N} p_n(x, \beta) \exp(-d^2_n/\beta) - \log \sum_{n=1}^{N} \exp(-d^2_n/\beta).
\]

Define partition function \( Z(\beta) = \sum_{n=1}^{N} \exp(-d^2_n/\beta) \) and moments

\[
m_k(\beta) = \sum_{n=1}^{N} p_n(x, \beta) d^2_n,
\]

Then:

\[
H(x, \beta) = 3m_1 + \log Z, \quad H_2(x, \beta) = -m_2 - m^2,
\]

\[
H_3(x, \beta) = \beta(m_3 - m_{2}m_2 + 2m_1^2) + m_1^3 - m_2^2.
\]

Consider the search for \( \beta \) (or \( \sigma \)) given the perplexity \( K \):

\[
F(x, K) := H(x, \beta) - \log K = 0.
\]

We propose modified derivative-based methods that achieve global convergence from any starting point:

- **Input:** initial \( \beta \), perplexity \( K \), distances \( d_1, \ldots, d_N \), compute bounds \( B \).
- **while** true do
  - for \( k = 0 \) to maxit do
    - compute \( \beta \) using any derivative-based method
    - if tolerance achieved return \( \beta \)
  - end
  - if \( \beta \in \mathcal{B} \) exit for loop
  - update \( \mathcal{B} \)
  - end
- **end while**

2. Initialization

The initialization should be close to the root:

- **Precomputed initialization.** For example, middle of the bounds, or distance to i-th nearest neighbor.

3. Computation of entropic affinities

For each point, we run a quickly-convergent root-finding algorithm with a clever initialization.

1. **Choice of root-finding algorithm**

We focus of the following methods:

- **1. Derivative-free methods.** Define an interval around the root and iteratively shrinking it (convergence is slow, but guaranteed.
- **2. Bisection - first-order method.**
- **3. Brent - superlinear convergence.**
- **4. Ridder - quadratic convergence.**

2. **Derivative-based methods.** Generally do not have convergence guarantees. High-order convergence:
- **Newton - second order, approximation with a line.**
- **Euler - third order, approximation with a parabola.**
- **Halley - third order, approximation with a hyperbola.**

3. **Experimental evaluation**

1. 512 x 512 Lena image. Each data point is a pixel represented by spatial and range features \((x, y, L, u, v) \in \mathbb{R}^5\) where \((x, y)\) is the pixel location and \((L, u, v)\) the pixel value (overall \( N \approx 202,144 \) points in \( D = 5 \) dimensions).

2. 60,000 handwritten digits from the MNIST dataset. Each datapoint is a 28 x 28 grayscale image.

3. 30,000 articles from Grolier’s encyclopedia. Each point is a word count of the most popular words from 30,000 articles.

4. **Conclusions**

We studied the behavior of entropic affinities function and its properties. The only parameters of the algorithms is the single perplexity value \( K \). Using (1) root-finding methods with second- or third-order convergence, (2) warm-start initialization based on local or density properties of the entropy, and (3) tight bounds around the root we were able to solve the problem all the way down to the machine precision in just a few iterations per point. MATLAB code: [http://eecs.ucmerced.edu](http://eecs.ucmerced.edu) Run it simply like \( \text{[x, a]} = \text{a}(x, K) \).