Nonlinear Embedding Methods

For high-dimensional data set \( Y \in \mathbb{R}^{D \times N} \) and \( X \in \mathbb{R}^{d \times N} \), its low-dimensional projection we can formulate nonlinear embedding algorithms as: \( E(X, \lambda) = E'(X) + \lambda E''(X) \), with a trade-off parameter \( \lambda \geq 0 \). For example, in the Elastic Embedding algorithm:

\[
E_{EE}(X, \lambda) = \sum_{n,m=1}^{N} w_{nm} \| x_n - x_m \|^2 + \sum_{n,m=1}^{N} w_{nm} \epsilon \| x_n - x_m \|^2
\]

A nonlinear embedding preserves structure in the high-dimensional data better than linear or spectral methods, but existing training algorithms have quadratic runtime on the number of points \( N \). We address this bottleneck by formulating the optimization as an \( N \)-body problem and using fast multipole methods (FMMs) to approximate the gradient in linear time.

\section*{2 \textit{N}-Body Methods}

1. **Tree-based methods** build a high-dimensional tree around the dataset. Each node contains a subset of the data. Savings occurs by replacing certain point-point interactions with node-point or node-node ones. Typical complexity of these methods is \( O(N \log N) \). Particular case include Barnes-Hut algorithm:

2. **Fast Multipole Methods** do a series expansion (up to \( p \) terms) of the interactions locally around every point. This decouples the computation of each term and reduces a single computation between \( N^2 \) number of terms into a series of computations with \( N \) terms. Overall complexity thus reduces to \( O(N) \).

\section*{3 Accuracy in gradient computation}

Each iteration \( k \) we always incur a small error \( \epsilon_k \). It is better to increase the accuracy with the iterations:
- Cheaper to compute low-accuracy value.
- Analogous to simulated annealing \( \Rightarrow \) gradually increase the accuracy to avoid wandering behavior.
- Assuming \( \epsilon_k \sim \mathcal{N}(0, \sigma^2) \) we show that adding noise is beneficial only where the mean curvature \( \frac{1}{v} (\nabla^2 E(x)) \) is negative, which can happen only in the beginning of the optimization.

\section*{Role of changing the accuracy in FMM optimization}

\section*{4 Experiments}

- 1 000 000 points from infiniteMNIST.
- Elastic Embedding algorithm \((\lambda = 10^{-4})\) optimized with gradient descent (GD), fixed point iterations (FP) and L-BFGS.
- No line search and fixed step size. The accuracy was increased for the first 100 iterations from \( p = 1 \) to \( p = 10 \) terms.

\section*{5 Conclusions}

- \textit{N}-Body methods we can address the main bottleneck of nonlinear embedding methods: quadratic cost of the objective function and the gradient.
- Fast Multipole Methods are more beneficial than Barnes-Hut both theoretically and empirically (\( 4 - 7 \times \) speedup for \( 10^6 \) elements dataset).
- Gradual increase of the accuracy parameter is advisable.