Linear and Non-linear Dimensionality Reduction applied to gene expression data of cancer tissue samples

Franck Olivier Ndjakou Njeunje

Applied Mathematics, Statistics, and Scientific Computation

Advisers
Wojtek Czaja
John J. Benedetto
Norbert Wiener Center for Harmonic Analysis
Department of Mathematics
University of Maryland - College Park

October 9, 2014
Outline

Introduction
  Background
  Approach

Software and Hardware
  Principal component Analysis [1]
  Laplacian Eigenmap [2]
  Hierarchical Clustering
  K-means Clustering
  Implementation

Validation and Test Problem

Extra

Bibliography
Gene expression data:
- interaction between genes and environment
- High-density micro-array
- Similarity learning
- Gene function
- Disease mechanism
- High-dimension
Dimension reduction

- Linear (LDR)
- Non-linear (NDR)

Expression of a single gene/variable

Gene expression level in the third sample

Variable \( x \) with dimension \( M \)

Variable \( y \) with dimension \( m \)

\[
\begin{array}{cccccc}
5 & 0 & 7 & 5 & 9 & 0 \\
6.2 & 3.14 & 7.5
\end{array}
\]
Similarity learning: Clustering

- **Clustering**: Elements in the same cluster are more similar than elements in other cluster.

- **Example**: *K-means* and *Expectation-Maximization* clustering are applied on an artificial dataset of mouse.\(^1\)

\(^{1}\)Image by Chire Wikipedia
For this project I will be considering the following methods:

1. Dimension reduction algorithms
   - LDR: Principal Component Analysis (PCA)
   - NDR: Laplacian Eigenmap (LE)

2. Clustering algorithm
   - Hierarchical clustering (HC)
   - K-means (KM)
► **Principal Component Analysis:**
  ▶ Does not do well on data with non-linear structure.
  ▶ Preserve the most variance from data.

► **Step 1:** Compute the standardized matrix $\tilde{X}$ of the original matrix $X$,

$$
\tilde{X} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_M) \\
= \left( \frac{x_1 - \bar{x}_1}{\sqrt{\sigma_{11}}}, \frac{x_2 - \bar{x}_2}{\sqrt{\sigma_{22}}}, \ldots, \frac{x_M - \bar{x}_M}{\sqrt{\sigma_{MM}}} \right).
$$

Here, $\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_M$ and $\sigma_{11}, \sigma_{22}, \ldots, \sigma_{MM}$ are respectively the mean values and the variances for corresponding variable vectors.
Step 2: Compute the covariance matrix of $\tilde{X}$, then make spectral decomposition to get the eigenvalues and its corresponding eigenvectors.

$$C = \tilde{X}'\tilde{X} = XX'.$$

Here $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_M)$, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M$, $U = (u_1, u_2, \ldots, u_M)$. $\lambda_i$ and $u_i$ are separately the $i$th eigenvalue corresponding eigenvector for covariance matrix $C$.

Step 3: Determine the number of principal components based on the preconcerted value. Supposing the number to be $m$, the $i^{th}$ principal component can be computed as $\tilde{X}u_i$, and the reduced dimensional \((N \times m)\) subspace is $\tilde{X}U_m$.\(^2\)

\(^2\)Jinlong Shi, Zhigang Luo, Nonlinear dimensionality reduction of gene expression data for visualization and clustering analysis of cancer tissue samples.
Laplacian Eigenmap:
- Non-linear
- Preserve natural geometrical structure

Step 1: Given a set of $N$ points $x_1, x_2, \ldots, x_N$ in $\mathbb{R}^M$, construct a weighted graph with $N$ nodes.
- $\epsilon$-neighborhoods: Link nodes that are $\epsilon$ away from each other. This could lead to a disconnected graph.
- $k$ nearest neighbors: Each node is linked to the $k^{th}$ nearest neighbors.

Step 2: Choose the weight for the edges and construct the weight matrix $W$.\(^3\)

\(^3\)Mikhail Belkin, Partha Niyogi, Laplacian Eigenmaps for Dimensionality Reduction and Data Representation
Step 3: For each connected sub-graph(s), solve the following generalized eigenvector problem,

$$Lf = \lambda Df,$$  \hspace{1cm} (4)\]

where $$D_{ii} = \sum_j W_{ji}$$, the diagonal matrix; and $$L = D - W$$, the Laplacian matrix.

Let $$f_0, f_1, \ldots, f_{N-1}$$ be the solutions of (4) with corresponding $$\lambda_0, \lambda_1, \ldots, \lambda_{N-1}$$ such that, $$Lf_i = \lambda_i Df_i$$ for $$i$$ going from 0 to $$N-1$$ and $$0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_{N-1}$$. Then the $$m$$-dimensional Euclidean space embedding is given by:

$$x_i \rightarrow y_i = (f_1(i), \ldots, f_m(i)).$$  \hspace{1cm} (5)
Hierarchical Clustering

Choose a metric:

- Euclidean distance:

\[ \|a - b\|_2 = \sqrt{\sum_i (a_i - b_i)^2} \]  

- Manhattan distance:

\[ \|a - b\|_1 = \sum_i |a_i - b_i| \]
Linkage Criteria:

- Maximum or CLINK (complete linkage clustering)
  \[ \text{max}\{d(a, b) : a \in A, b \in B\}. \quad (8) \]

- Minimum or SLINK (single linkage clustering)
  \[ \text{min}\{d(a, b) : a \in A, b \in B\}. \quad (9) \]

- Mean or average linkage clustering
  \[ \frac{1}{|A||B|} \sum_{a \in A} \sum_{b \in B} d(a, b). \quad (10) \]
K-means Clustering

- Initialized a set of $k$ means $m_1^{(1)}, m_2^{(1)}, \ldots, m_k^{(1)}$.
- Assignment step: Assign each observation $x_p$ to exactly one set $S_i$ containing the nearest mean to $x_p$.

$$S_i^t = \{x_p : \|x_p - m_i^{(t)}\|^2 \leq \|x_p - m_j^{(t)}\|^2 \ \forall j, 1 \leq j \leq k\}. \quad (11)$$

- Update step: update the mean within each cluster,

$$m_i^{t+1} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i(t)} x_j. \quad (12)$$

- Repeat the two previous steps.
- Stop when no new assignments are made.
- **K-means illustration:**

  \( k \) initial means, \( k = 3 \)

  ![K-means illustration diagram](image)

  **Assignment step**

  **Update step**

  **Repeat**
1. **platform**
   - Matlab
   - Personal laptop
   - Norbert Wiener Center lab

2. **Database**
   - NIC-60 [3, p. 95-96]
   - Computer generated data for test run

3. **Complexity: Large matrix operations**

4. **Algorithm implemented**
   - PCA from the Singular Value Decomposition.
   - Laplacian Eigenmaps
DRtoolbox\textsuperscript{4} [4] contains:
\begin{itemize}
  \item Implementation of the PCA and LE methods describe above, they would be used for comparison.
  \item Well understood data such as the \textit{Swiss roll} and the \textit{Twin peaks}
\end{itemize}

\textbf{Swiss roll} \hspace{1cm} \textbf{Twin peaks}

\begin{itemize}
  \item Clustering algorithms will be borrowed and used as a tool to compare the outputs from the DR methods.
\end{itemize}

\textsuperscript{4}Laurens van der Maaten, Delft University of Technology
October - November:
  - Implementation of PCA algorithm.
  - Resolve issues that come up (storage and memory).
  - Testing and validating.

December: Mid-year presentation.

January: First semester progress report.

February - April:
  - Implementation of LE algorithm.
  - Testing and validating.

April - May:
  - Implementation of a clustering algorithm (if time permits).

May: Final report
Dimensionality Reduction Techniques

Franck Njeunje

Introduction
Background
Approach
Software and Hardware
Principal component Analysis [1]
Laplacian Eigenmap [2]
Hierarchical Clustering
K-means Clustering Implementation
Validation and Test Problem
Validation Methods
Extra
Timeline
Deliverable
Bibliography

- Weekly Report
- Self Introduction
- Project Proposal
- First-Semester Progress Report
- Mid-year Status Report
- Final Report
- Code for PCA implementation
- Code for LE implementation
- NIC-60 data set


Vinodh N. Rajapakse (2013). Data Representation for Learning and Information Fusion in Bioinformatics. Digital Repository at the University of Maryland, University of Maryland (College Park, Md.)

Laurens van der Maaten, Affiliation: Delft University of Technology. Matlab Toolbox for Dimensionality Reduction (v0.8.1b) March 21, 2013.