Laplacian PCA

Maths Derivation and Explanation

May Yuan-Klitzner

Radar Remote Sensing Group

1 Introduction

From PCA to Graph Laplacian PCA: Preserving Variance and Local Geometry

Dimensionality reduction is a crucial preprocessing step in many machine learning applications. It helps reveal underlying class distributions in low-dimensional spaces, often leading to improved learning outcomes. This document discusses Principal Component Analysis (PCA), Graph Laplacian PCA (gLPCA), and their combined framework.

Principal Component Analysis (PCA) Overview

PCA is the fundamental dimensionality reduction technique that performs the following steps:

1. Centers data points at the origin.
2. Finds orthogonal directions of maximum variance.
3. Projects data onto these directions.
4. Minimizes the reconstruction error for a given dimensionality.

Given a data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ with n samples and d features, we first center it such that $\sum_{i=1}^{n} x_{ij}^{(\text{centered})} = 0$ for each feature j. The sample covariance matrix is computed as:

$$\mathbf{S} = \frac{1}{n-1}\mathbf{X}\mathbf{X}^T \tag{1}$$

The core eigenvalue problem for PCA is:

$$\mathbf{S}\mathbf{w} = \lambda\mathbf{w} \tag{2}$$

where:

- $\mathbf{S}$ is the covariance matrix
- $\mathbf{w}$ are the eigenvectors (principal components)
- λ are the corresponding eigenvalues

The eigenvectors of $\mathbf{S}$ represent the principal components – directions of maximum variance in the data.

Graph Laplacian PCA (gLPCA) Overview

Standard PCA has a limitation: it ignores the local geometric structure of the data manifold. The key insight behind gLPCA is that nearby points in the high-dimensional space should remain nearby in the low-dimensional projection.

Combined PCA and Laplacian Embedding Overview

Both approaches offer complementary advantages:

- PCA naturally has clustering tendencies, as directions that separate data variance often naturally separate clusters.
- Laplacian methods are explicitly designed to optimize cluster separation by preserving manifold structure.

A combined framework aims to leverage both:

- PCA's simplicity and computational efficiency
- Laplacian's ability to capture manifold structure

2 Detailed Mathematical Explanation and Derivation

2.1 Step 1: Initialize Matrices

Let:

- $\mathbf{X}$ = vector data (feature matrix)
- $\mathbf{W}$ = graph data where entries represent distances between nodes or graph edge weights

The input data contains both vector data $\mathbf{X}$ and graph data $\mathbf{W}$.

Task: Learn a low-dimensional data representation of $\mathbf{X}$ that incorporates cluster information encoded in graph data $\mathbf{W}$.

Result: The gLPCA model will then encompass:

1. Data representation
2. Data embedding (low-dimensional representation preserving local neighborhood structure)
3. A closed-form solution that can be efficiently computed

2.2 Step 2: Apply PCA

Let the input data matrix be:

$$\mathbf{X} = (\bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_n) \in \mathbb{R}^{p \times n} \tag{3}$$

where $\bar{\mathbf{x}}_i$ is a column vector and $\mathbf{X}$ contains n data column vectors in p-dimensional space (i.e., $\mathbf{X}$ has p rows):

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{p1} & \cdots & x_{pn} \end{bmatrix} \tag{4}$$

PCA finds the optimal low-dimensional (k-dim) subspace defined by the principal directions:

$$\mathbf{U} = (\bar{\mathbf{u}}_1, \ldots, \bar{\mathbf{u}}_k) \in \mathbb{R}^{p\times k} \tag{5}$$

The projected data points in the new subspace are:

$$\mathbf{V}^T = (\bar{\mathbf{v}}_1, \ldots, \bar{\mathbf{v}}_n) \in \mathbb{R}^{k\times n} \tag{6}$$

where $\mathbf{V}$ is an orthogonal matrix that holds the coordinates of the data points in the new subspace.

2.3 Step 3: PCA finds U and V

PCA finds $\mathbf{U}$ and $\mathbf{V}$ by minimizing the reconstruction error. This becomes an optimization problem that finds the best possible low-rank reconstruction of the vector data matrix $\mathbf{X}$.

The optimization problem is:

$$\min_{\mathbf{U},\mathbf{V}} \|\mathbf{X} - \mathbf{U}\mathbf{V}^T\|_F^2, \quad \text{such that} \quad \mathbf{V}^T\mathbf{V} = \mathbf{I} \tag{7}$$

where:

- $\|\cdot\|_F$ denotes the Frobenius norm of a matrix, which is the square root of the sum of squares of all entries. In this context, it represents the total squared reconstruction error over all entries of the vector data matrix $\mathbf{X}$.
- $\mathbf{V}^T\mathbf{V} = \mathbf{I}$ means $\mathbf{V}$ is an orthogonal matrix, ensuring the principal components are orthogonal.

2.4 Step 4: Manifold (Laplacian) Embedding using Graph Laplacian

PCA provides an embedding for data lying on a linear manifold. However, when data lies on a nonlinear manifold, a popular alternative is graph Laplacian-based embedding.

Given the pairwise similarity matrix:

$$\mathbf{W} \in \mathbb{R}^{n\times n} \tag{8}$$

containing edge weights on a graph with n nodes, Laplacian embedding preserves the local geometrical relationships and maximizes the smoothness with respect to the manifold of the dataset in the low-dimensional embedding space.

Let:

$$\mathbf{Q}^T = (\bar{\mathbf{q}}_1, \ldots, \bar{\mathbf{q}}_n) \in \mathbb{R}^{k\times n} \tag{9}$$

(analogous to the $\mathbf{V}^T$ matrix in PCA) be the embedding coordinates of the n data points, where each column $\bar{\mathbf{q}}_i$ is a low-dimensional coordinate of data point i.

These coordinates are obtained by solving the optimization problem:

$$\min_{\mathbf{Q}} \sum_{i,j=1}^{n} \|\bar{\mathbf{q}}_i - \bar{\mathbf{q}}_j\|^2 \cdot W_{ij} = \mathrm{Tr}(\mathbf{Q}^T(\mathbf{D} - \mathbf{W})\mathbf{Q}), \quad \text{such that} \quad \mathbf{Q}^T\mathbf{Q} = \mathbf{I} \tag{10}$$

where:

- W_{ij} is the similarity weight between points i and j
- Objective: Make similar points (large W_{ij}) have similar coordinates (small $\|\bar{\mathbf{q}}_i - \bar{\mathbf{q}}_j\|^2$)
- $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the Laplacian matrix
- $\mathbf{D}$ is a diagonal matrix where $\mathbf{D} = \mathrm{diag}(d_1, \ldots, d_n)$
- $d_i = \sum_j W_{ij}$ is the sum of similarities for point i
- $\mathrm{Tr}(\cdot)$ denotes the trace operation (sum of diagonal entries)
- Columns of $\mathbf{Q}$ are orthogonal and normalized (orthonormal eigenvectors)

2.5 Step 5: Finding Eigenvectors of $\mathbf{L} = \mathbf{D} - \mathbf{W}$ to Obtain Embedding Coordinates

The objective is to compute the eigenvectors of the Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$. These eigenvectors form the columns of $\mathbf{Q}$, with each row representing an embedding coordinate.

Form the Laplacian matrix:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \tag{11}$$

Solve the eigenvalue problem:

$$\mathbf{L}\bar{\mathbf{q}} = \lambda\bar{\mathbf{q}} \tag{12}$$

where $\bar{\mathbf{q}}$ is an eigenvector and λ is the corresponding eigenvalue.

According to the Rayleigh-Ritz Theorem:

$$\min_{\mathbf{Q}^T\mathbf{Q}=\mathbf{I}} \mathrm{Tr}(\mathbf{Q}^T\mathbf{L}\mathbf{Q}) \tag{13}$$

is solved by taking the eigenvectors corresponding to the smallest eigenvalues of $\mathbf{L}$, given that $\mathbf{L}$ is symmetric (since $\mathbf{W}$ is symmetric).

2.6 Step 6: Form the Graph-Laplacian PCA Model

Suppose we are given vector data $\mathbf{X}$ and pairwise similarity matrix $\mathbf{W}$. We wish to learn a low-dimensional data representation of $\mathbf{X}$ that incorporates the cluster structures inherent in $\mathbf{W}$.

Since $\bar{\mathbf{v}}_i$ in $\mathbf{V}^T$ (from PCA) plays the same role as $\bar{\mathbf{q}}_i$ in $\mathbf{Q}^T$ (from Laplacian embedding), we set them equal and combine Equations (7) and (10) into the following unified model:

$$\min_{\mathbf{U},\mathbf{Q}} J = \|\mathbf{X} - \mathbf{U}\mathbf{Q}^T\|_F^2 + \alpha \operatorname{Tr}\left(\mathbf{Q}^T(\mathbf{D} - \mathbf{W})\mathbf{Q}\right), \quad \text{such that} \quad \mathbf{Q}^T\mathbf{Q} = \mathbf{I} \tag{14}$$

where $\alpha \geq 0$ is a regularization parameter that balances between the two objectives:

1. **PCA Term:** $\|\mathbf{X} - \mathbf{U}\mathbf{Q}^T\|_F^2$
   - Encourages $\mathbf{U}\mathbf{Q}^T$ to reconstruct $\mathbf{X}$ accurately
   - Captures global variance structure of the data

2. **Laplacian/Manifold Smoothness Term:** $\alpha \operatorname{Tr}\left(\mathbf{Q}^T(\mathbf{D} - \mathbf{W})\mathbf{Q}\right)$
   - Encourages similar points (large W_{ij}) to have similar embeddings (small $\|\bar{\mathbf{q}}_i - \bar{\mathbf{q}}_j\|^2$)
   - Preserves local neighborhood structure on the data manifold

Therefore, the final gLPCA model encompasses several important aspects:

- Data representation: $\mathbf{X} \approx \mathbf{U}\mathbf{Q}^T$
- Manifold embedding: using $\mathbf{Q}$ to preserve local structure
- Computational efficiency: has a closed-form solution that can be efficiently computed

2.7 Step 7: Final Closed-Form Solution

The optimal solution $(\mathbf{U}^*, \mathbf{Q}^*)$ of gLPCA is given by:

$$\mathbf{Q}^* = (\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \ldots, \bar{\mathbf{v}}_k) \tag{15}$$

and

$$\mathbf{U}^* = \mathbf{X}\mathbf{Q}^* \tag{16}$$

where $\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \ldots, \bar{\mathbf{v}}_k$ are the eigenvectors corresponding to the first k smallest eigenvalues of matrix $\mathbf{G}$, obtained by solving the eigenvalue problem $\mathbf{G}\bar{\mathbf{v}} = \lambda\bar{\mathbf{v}}$.

Thus, graph Laplacian PCA has the following compact closed-form solution:

$$\mathbf{G} = -\mathbf{X}^T\mathbf{X} + \alpha\mathbf{L}, \quad \text{where} \quad \mathbf{L} = \mathbf{D} - \mathbf{W} \tag{17}$$

Note that the vector data $\mathbf{X}$ is centered as in standard PCA.

2.8 Step 8: Proof for Closed-Form Solution of gLPCA

Let:

$$J(\mathbf{U}) = \|\mathbf{X} - \mathbf{U}\mathbf{Q}^T\|_F^2. \tag{18}$$

2.8.1 Solving for optimal $\mathbf{U}^*$ while fixing Q:

Take the partial derivative with respect to $\mathbf{U}$ and set it to zero:

$$\frac{\partial J}{\partial \mathbf{U}} = -2\mathbf{X}\mathbf{Q} + 2\mathbf{U} = 0 \tag{19}$$

Solving for $\mathbf{U}$ gives:

$$\mathbf{U}^* = \mathbf{X}\mathbf{Q} \tag{20}$$

2.8.2 Solving for optimal $\mathbf{Q}^*$:

Substitute Equation (20) into Equation (14) to obtain:

$$\min_{\mathbf{Q}} J(\mathbf{Q}) = \|\mathbf{X} - \mathbf{X}\mathbf{Q}\mathbf{Q}^T\|_F^2 + \alpha \operatorname{Tr}\left(\mathbf{Q}^T\mathbf{L}\mathbf{Q}\right), \quad \text{such that} \quad \mathbf{Q}^T\mathbf{Q} = \mathbf{I} \tag{21}$$

Using algebraic simplifications:

- Using the identity for the squared Frobenius norm: $\|\mathbf{A}\|_F^2 = \operatorname{Tr}(\mathbf{A}^T\mathbf{A})$
- Applying matrix multiplication algebra

The objective becomes:

$$\min_{\mathbf{Q}} \operatorname{Tr}\left(\mathbf{Q}^T(-\mathbf{X}^T\mathbf{X} + \alpha\mathbf{L})\mathbf{Q}\right) \tag{22}$$

Since $\mathbf{Q}\mathbf{Q}^T = \mathbf{I}$ and $\operatorname{Tr}(x) = x$ when x is a scalar, we define:

$$\mathbf{G} = -\mathbf{X}^T\mathbf{X} + \alpha\mathbf{L} \tag{23}$$

Thus, the optimization reduces to finding the eigenvectors of $\mathbf{G}$ corresponding to the smallest eigenvalues, which yields $\mathbf{Q}^*$ as shown in Equation (15).

3 Conclusion

In this work, we have presented a unified framework that combines Principal Component Analysis (PCA) with graph Laplacian embedding, resulting in the Graph Laplacian PCA (gLPCA) model. The key contributions can be summarized as follows:

3.1 Key Contributions

1. **Unified Framework:** We developed a mathematical formulation (Equation (14)) that seamlessly integrates PCA's global variance preservation with Laplacian embedding's local manifold preservation.

2. **Closed-Form Solution:** We derived an efficient closed-form solution (Equations (15) and (17)) requiring only the computation of eigenvectors of matrix $\mathbf{G} = -\mathbf{X}^T\mathbf{X} + \alpha\mathbf{L}$, making the method computationally efficient.
3. **Flexible Regularization:** The parameter α provides explicit control over the trade-off between data reconstruction fidelity and manifold smoothness preservation.
4. **Interpretability:** The solution maintains clear interpretations from both perspectives: as a low-rank data representation ($\mathbf{X} \approx \mathbf{U}\mathbf{Q}^T$) and as a manifold-preserving embedding.