

CS-667 Advanced Machine Learning

Nazar Khan

PUCIT

Principal Component Analysis

Principal Component Analysis

- ▶ Widely used technique for
 - ▶ dimensionality reduction
 - ▶ data compression (lossy)
 - ▶ feature extraction
 - ▶ data visualisation
- ▶ Can be defined in 2 ways
 - ▶ *Orthogonal projection* of data onto lower dimensional *linear* space (*principal subspace*) such that *variance of projected data is maximised*.
 - ▶ Linear projection that minimises average projection cost.
- ▶ Also called Karhunen-Loeve transform.

Principal Component Analysis

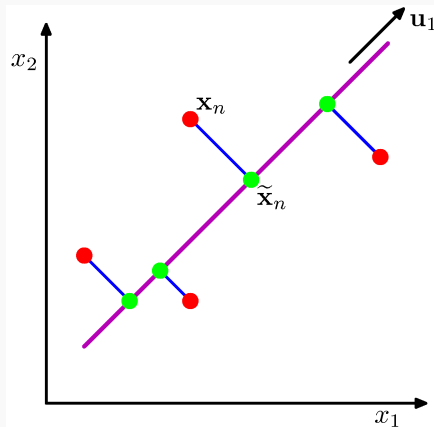


Figure: The two views of PCA. In this example for 2D data (in red), we want to find the direction vector $\mathbf{u}_1$ (in magenta) for which (1) the projections (in green) have maximum variance, or (2) the projection costs (lengths of blue lines) are minimum.

Maximum Variance Formulation of PCA

- ▶ Consider a set of signals $X = [\mathbf{x}_1, \dots, \mathbf{x}_N]$ where each $\mathbf{x}_i \in \mathbb{R}^D$.
- ▶ We have to find a vector $\mathbf{u} \in \mathbb{R}^D$ such that the variance of the projected data onto $\mathbf{u}$ is maximum.
- ▶ Projections of a data points $\mathbf{x}_i$ onto $\mathbf{u}$ are obtained via dot-products $\mathbf{u}^T \mathbf{x}_i$ for $i = 1, \dots, N$.
- ▶ Mean of projected data is computed as $\mathbf{u}^T \bar{\mathbf{x}}$ where $\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$.
- ▶ Therefore, variance of projected data along direction $\mathbf{u}$ is computed as

$$\text{Var}(\mathbf{u}) = \frac{1}{N} \sum_{i=1}^N (\mathbf{u}^T \mathbf{x}_i - \mathbf{u}^T \bar{\mathbf{x}})^2$$

Maximum Variance Formulation of PCA

- Variance along $\mathbf{u}$ can be rewritten as the quadratic form

$$\begin{aligned}\text{Var}(\mathbf{u}) &= \frac{1}{N} \sum_{i=1}^N (\mathbf{u}^T \mathbf{x}_i - \mathbf{u}^T \bar{\mathbf{x}})^2 = \frac{1}{N} \sum_{i=1}^N (\mathbf{u}^T \mathbf{x}_i - \mathbf{u}^T \bar{\mathbf{x}})(\mathbf{u}^T \mathbf{x}_i - \mathbf{u}^T \bar{\mathbf{x}})^T \\ &= \frac{1}{N} \sum_{i=1}^N (\mathbf{u}^T \mathbf{x}_i - \mathbf{u}^T \bar{\mathbf{x}})(\mathbf{x}_i^T \mathbf{u} - \bar{\mathbf{x}}^T \mathbf{u}) = \frac{1}{N} \sum_{i=1}^N \mathbf{u}^T (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i^T - \bar{\mathbf{x}}^T) \mathbf{u} \\ &= \mathbf{u}^T \underbrace{\frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i^T - \bar{\mathbf{x}}^T)}_{S_{D \times D}} \mathbf{u} = \mathbf{u}^T S \mathbf{u}\end{aligned}$$

- We want to find the direction vector $\mathbf{u}$ that maximises the quadratic form $\mathbf{u}^T S \mathbf{u}$ where S is the *data covariance matrix*.
- **Take-home Quiz 3:** Prove that $\mathbf{u}^* = \arg \max_{\mathbf{u}} \mathbf{u}^T S \mathbf{u}$ is the eigenvector of S corresponding to the largest eigenvalue. (Hint: This is a constrained optimisation problem.)

Maximum Variance Formulation of PCA

- ▶ The eigenvector of S corresponding to the largest eigenvalue is called the *first principal component*.
- ▶ Additional principal components can be defined incrementally by *choosing each new projection direction as the one with maximum projected variance among all directions orthogonal to those already considered*.
- ▶ First M principal components correspond to the eigenvectors $\mathbf{u}_1, \dots, \mathbf{u}_M$ of S corresponding to the M largest eigenvalues $\lambda_1, \dots, \lambda_M$. (Proof by induction in Exercise 12.1)
- ▶ Eigen-decomposition of $D \times D$ matrix has $O(D^3)$ complexity.
- ▶ For finding the first M eigenvectors only, there exist alternative methods such as the *power method* with $O(MD^2)$ complexity.

Choosing M

- ▶ Total variance of the data is given by the sum $V(D) = \sum_{i=1}^D \lambda_i$.
- ▶ By using the first M principal components, we capture variance amounting to $V(M) = \sum_{i=1}^M \lambda_i$.
- ▶ The remaining, uncaptured variance is called the *distortion measure* and is given by $J = \sum_{i=M+1}^D \lambda_i$.
- ▶ M can be chosen as the smallest integer for which $\frac{V(M)}{V(D)} > \tau$ where $0 < \tau \leq 1$.
- ▶ For example, $\tau = 0.95$ corresponds to retaining 95% of the total variance after projection.

Choosing M

- ▶ Even for $\tau = 1$, it is often observed that $M < D$.
- ▶ This shows that the *intrinsic dimensionality* of D -dimensional data is often less than D .
- ▶ Therefore, by working in this lower-dimensional space we do not lose any variations in the data.

Assignment 5

Principal Component Analysis

- ▶ Dimensionality reduction via PCA.
 - ▶ Code up a generic implementation of PCA in function `[evecs, evals]=compute_pca(X)` where X is a $D \times N$ data matrix.
 - ▶ Regenerate Figures 12.3, 12.4 and 12.5 in Bishop's book.
- ▶ Submit your `_roll_number_PCA.zip` containing
 - ▶ code,
 - ▶ generated images, and
 - ▶ `report.txt/pdf` explaining your results.
- ▶ Due Tuesday, April 25, 2016 before 5:30 pm on `\\printsrv.`

PCA for high-dimensional data

- ▶ N points in $\mathbb{R}^D$ define an $N - 1$ dimensional linear subspace.
- ▶ If $N < D$, the $D \times D$ covariance matrix S will have rank (= number of non-zero eigenvalues) at most $N - 1$.
- ▶ The remaining $D - (N - 1)$ eigenvalues of S will all be 0.
- ▶ So we should not compute more than $N - 1$ eigenvectors.
- ▶ Projecting onto $M > N - 1$ eigenvectors *does not imply* dimensionality reduction.
- ▶ The $N < D$ scenario occurs often. For example, in a dataset of $N = 100000$ RGB images of size 640×480 ,
 $D = 640 * 480 * 3 = 921600 \gg N$.
- ▶ The $O(D^3)$ scaling also makes computing the eigenvectors of S impractical for large D .

PCA for high-dimensional data

- ▶ So we use a clever trick.
- ▶ Let $\tilde{\mathbf{X}}$ be the *data centered design matrix*.

$$\tilde{\mathbf{X}} = \begin{bmatrix} (\mathbf{x}_1 - \bar{\mathbf{x}})^T \\ (\mathbf{x}_2 - \bar{\mathbf{x}})^T \\ \vdots \\ (\mathbf{x}_N - \bar{\mathbf{x}})^T \end{bmatrix}$$

- ▶ We can write the data covariance matrix as

$$S = \sum_{i=1}^N (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i^T - \bar{\mathbf{x}}^T) = \frac{1}{N} \tilde{\mathbf{X}}^T \tilde{\mathbf{X}}$$

PCA for high-dimensional data

- ▶ The eigenvector equation can be written as

$$\begin{aligned} S\mathbf{u}_i &= \lambda_i \mathbf{u}_i \implies \frac{1}{N} \tilde{\mathbf{X}}^T \tilde{\mathbf{X}} \mathbf{u}_i = \lambda_i \mathbf{u}_i \\ &\implies \frac{1}{N} \tilde{\mathbf{X}} \tilde{\mathbf{X}}^T \tilde{\mathbf{X}} \mathbf{u}_i = \lambda_i \tilde{\mathbf{X}} \mathbf{u}_i \\ &\implies \frac{1}{N} \tilde{\mathbf{X}} \tilde{\mathbf{X}}^T \mathbf{v}_i = \lambda_i \mathbf{v}_i \end{aligned} \tag{1}$$

which shows that λ_i and $\mathbf{v}_i$ are eigenvalues and eigenvectors of the *smaller* $N \times N$ matrix $\tilde{\mathbf{X}} \tilde{\mathbf{X}}^T$.

- ▶ But notice that λ_i was also the eigenvalue of the original covariance matrix S . So we have found the eigenvalues of S in $O(N^3)$.

PCA for high-dimensional data

- ▶ To obtain the eigenvectors $\mathbf{u}_i$, pre-multiply both sides of Equation (1) by $\tilde{\mathbf{X}}^T$ to obtain

$$\left(\frac{1}{N} \tilde{\mathbf{X}}^T \tilde{\mathbf{X}} \right) \left(\tilde{\mathbf{X}}^T \mathbf{v}_i \right) = \lambda_i \left(\tilde{\mathbf{X}}^T \mathbf{v}_i \right)$$

which shows that $\tilde{\mathbf{X}}^T \mathbf{v}_i$ is an eigenvector of S with eigenvalue λ_i .

- ▶ So the original eigenvectors are obtained as

$$\mathbf{u}_i = \frac{\tilde{\mathbf{X}}^T \mathbf{v}_i}{\|\tilde{\mathbf{X}}^T \mathbf{v}_i\|} = \frac{\tilde{\mathbf{X}}^T \mathbf{v}_i}{\sqrt{N\lambda_i}}$$

Show that $\|\tilde{\mathbf{X}}^T \mathbf{v}_i\| = \sqrt{N\lambda_i}$.

- ▶ So the eigen-decomposition of the $D \times D$ covariance matrix S can be achieved in $O(N^3)$.

PCA for high-dimensional data

Summary

- ▶ When $N < D$, construct the $N \times N$ matrix $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T$ and compute its eigenvalues λ_i and eigenvectors $\mathbf{v}_i$.
- ▶ Eigenvalues of S are also λ_i .
- ▶ Eigenvectors of S are obtained as

$$\mathbf{u}_i = \frac{\tilde{\mathbf{X}}^T \mathbf{v}_i}{\sqrt{N\lambda_i}}$$

Applications

- ▶ We now look at some applications of PCA.
- ▶ These include
 - ▶ Compression
 - ▶ Pre-processing of data
 - ▶ Visualization of data
 - ▶ Classification

Compression

- ▶ When data point $\mathbf{x}$ is projected onto the i -th principal component, coefficient of projection is given by

$$\alpha_i = (\mathbf{x} - \bar{\mathbf{x}})^T \mathbf{u}_i$$

- ▶ Consider projections $\alpha_1, \dots, \alpha_M$ onto the first M principal components where $M < D$.
- ▶ Reconstruction $\hat{\mathbf{x}}$ from these M scalar coefficients can be obtained as

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \sum_{i=1}^M \alpha_i \mathbf{u}_i$$

Compression

- ▶ This dimensionality reduction represents *compression* from $\mathbb{R}^D$ to $\mathbb{R}^M$.
- ▶ In $\mathbb{R}^D$, N data points require storing ND values.
- ▶ After compression using the first M principal components, the N data points require storing $NM + D$ values. (Why $+D$?)
- ▶ You will implement compression via PCA in Project 4a when you regenerate Bishop's Figure 12.5.

Data pre-processing

- ▶ Sometimes different dimensions of data have different units or significantly different variability.
 - ▶ $\mathbf{x} = [\text{time (seconds), speed (mph), fuel consumption (liters)}]^T$.
 - ▶ $\mathbf{x} = [\text{time between earthquakes, duration of earthquake}]^T$.
- ▶ Averaged over the whole dataset, every component of $\mathbf{x}$ will have a different mean and different variance.
- ▶ Effectiveness of subsequent algorithms can be diminished due to such variability.
- ▶ Non-PCA solution: *Standardize* the data using $y_{ni} = \frac{x_{ni} - \bar{x}_i}{\sigma_i}$.
- ▶ Individual components of the transformed data $\mathbf{y}_1, \dots, \mathbf{y}_N$ will now have *zero-mean and unit-variance*.
- ▶ However different components y_{ni} and y_{nj} can still be correlated.

Data pre-processing

Whitening

- ▶ A better PCA-based solution, known as *whitening* or *sphereing* transforms the data as

$$\mathbf{y}_n = \mathbf{L}^{-\frac{1}{2}} \mathbf{U}^T (\mathbf{x}_n - \bar{\mathbf{x}})$$

where $\mathbf{L}$ is a $D \times D$ diagonal matrix of D eigenvalues λ_i of S and $\mathbf{U}$ is an orthogonal $D \times D$ matrix with columns given by the corresponding eigenvectors $\mathbf{u}_i$.

- ▶ Easy to show that transformed data $\mathbf{y}_1, \dots, \mathbf{y}_N$ has zero-mean and its covariance matrix $\frac{1}{N} \sum_{n=1}^N \mathbf{y}_n \mathbf{y}_n^T$ equals $\mathbf{I}_{D \times D}$. **Show it.**
- ▶ So, individual components of the transformed data $\mathbf{y}_1, \dots, \mathbf{y}_N$ will now have *zero-mean and unit-covariance*.

Visualization

- ▶ Project data onto the first 1, 2, or 3 principal components and visualise these projected coefficients.

Classification via PCA

► Training

1. Compute eigen-decomposition of the complete training data $\mathbf{x}_1, \dots, \mathbf{x}_N$.
2. Form orthogonal eigen-basis from the first M principal components.
3. Project each mean-subtracted training sample $\mathbf{x}_n - \bar{\mathbf{x}}$ onto the eigen-bases to obtain projected coefficients $\phi_n \in \mathbb{R}^M$.

► Testing

1. Project mean-subtracted test sample $\mathbf{x} - \bar{\mathbf{x}}$ onto the eigen-bases to obtain projected coefficients $\phi \in \mathbb{R}^M$.
2. Compute Euclidean distance of coefficients ϕ from each of the coefficients ϕ_n of the training samples.
3. Class of $\mathbf{x}$ is the class of the nearest neighbour nn from the training samples where

$$nn = \arg \min_n ||\phi - \phi_n||^2$$

- This is essentially nearest neighbour classification in $\mathbb{R}^M$ instead of $\mathbb{R}^D$.

Assignment 6

Classification via Principal Component Analysis

- ▶ Classification via PCA.
 - ▶ Compute eigen-basis of a suitable size M for the 10 classes from the MNIST digits training set using the function `[evecs,evals]=compute_pca(X)` from Assignment 5.
 - ▶ Classify digits in the testing set and compute testing accuracy.
- ▶ Submit your `_roll_number_PCA_Classify.zip` containing
 - ▶ code,
 - ▶ `report.txt/pdf` explaining your results.
- ▶ **Please do not include the MNIST dataset in your .zip file.**
- ▶ Due Tuesday, May 2, 2016 before 5:30 pm on `\\printsrv`.