Principal Component Analysis: A Gentle Tutorial

Roger Jang
MATLAB code & examples
Last update: 2012/01/16

Prinbcipal component analysis (PCA) is an effective statistical technique for reducing the dimensions of a given unlabeled high-dimensional dataset while keeping its spatial characteristics as much as possible. It has found immense applications in image compression, pattern recognition (face recognition in particular) and data clustering. Depending on the field of application, PCA is also known as the discrete Karhunen-Loeve transformation, or the Hotelling transform.

More specifically, PCA transforms the dataset into a new coordinate system such that the projection onto the first coordinate have the greatest variance among all possible projections, and the projection onto the second coordinate have the second greatest variances, and so on. By finding these successive coordinates (or principal components), we can visualize the distribution of the original dataset after projecting it onto a low-dimensional space. In other words, PCA provides a best meaningful viewing angle that can disperse the dataset as much as possible. We shall have a gentle walk-through of the mathematics underlying PCA as follows.

Assume that our dataset is composed of m d-dimensional column vectors $\mathbf{x}_i$, where $i=1,\ldots,n$. Further, we also assume the dataset is zero justified. That is, the average across each dimension is zero:

$$\sum_{i=1}^{n}\mathbf{x}_i = 0.$$

If the original dataset does not satisfy this constraint, we can simply subtract the mean of each dimension from the original dataset. Our goal is now to find a unity vector $\mathbf{u}$ such that the squared sum of the dataset's projection onto this direction is the maximum. For simplicity, we can use an $d \times n$ matrix $\mathbf{X}$ to represent the dataset:

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_n \\ | & | & & | \end{bmatrix}$$

Then the projection of each column of A onto $\mathbf{u}$ can be represented by the following column vector:

$$\mathbf{p} = \begin{bmatrix} \mathbf{x}_1^T\mathbf{u} \\ \mathbf{x}_2^T\mathbf{u} \\ \vdots \\ \mathbf{x}_n^T\mathbf{u} \end{bmatrix} = \mathbf{X}^T\mathbf{u},$$

where $\mathbf{x}_i^T\mathbf{u}$ is the projection of vector $\mathbf{x}_i$ onto the unity vector $\mathbf{u}$. The square sum of the total projection is a function of $\mathbf{u}$, denoted by

$$J(\mathbf{u}) = \|\mathbf{p}\|^2 = \mathbf{p}^T\mathbf{p} = (\mathbf{X}^T\mathbf{u})^T(\mathbf{X}^T\mathbf{u}) = \mathbf{u}^T\mathbf{X}\mathbf{X}^T\mathbf{u}$$

To maximize the total projection $J(\mathbf{u})$ under the constraint $\|\mathbf{u}\| = 1$, we can use the Lagrange Multiplier to form a new objective function:

$$\tilde{J}(\mathbf{u},\lambda) = \mathbf{u}^T\mathbf{X}\mathbf{X}^T\mathbf{u} + \lambda(1-\mathbf{u}^T\mathbf{u})$$

To maximize the new objective function, we can find its gradient and set it to zero, as follows:

$$\nabla_{\mathbf{u}} \tilde{J}(\mathbf{u},\lambda) = 0 \quad \Rightarrow \quad 2\mathbf{X}\mathbf{X}^T\mathbf{u} - 2\lambda\mathbf{u} = 0$$
$$\Rightarrow \qquad \mathbf{X}\mathbf{X}^T\mathbf{u} = \lambda\mathbf{u}$$

Now it is obvious to see that the maximum occurs when $\mathbf{u}$ is one of the eigenvectors of $\mathbf{X}\mathbf{X}^T$ and λ is the corresponding eigenvalues. Under this condition, the corresponding total projection is:

$$\begin{aligned} J(\mathbf{u}) = \|\mathbf{p}\|^2 &= \mathbf{u}^T\mathbf{X}\mathbf{X}^T\mathbf{u} \\ &= \mathbf{u}^T\lambda\mathbf{u} \\ &= \lambda \end{aligned}$$

We can arrange the eigenvalues of $\mathbf{X}\mathbf{X}^T$ into an descending order

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d ,$$

with the corresponding eigenvectors $\mathbf{u}_1, \mathbf{u}_2 \cdots \mathbf{u}_m$. Then the maximum value of $J(\mathbf{u})$ is λ_1 which occurs at $\mathbf{u} = \mathbf{u}_1$; while the minimum is λ_d which occurs at $\mathbf{u} = \mathbf{u}_d$.

Once we have found the first principal component $\mathbf{u}_1$ (as the unity eigenvector corresponding to the maximum eigenvague of $\mathbf{X}\mathbf{X}^T$), we can continue to find the second principal component that achieves the maximum total projection with the constraint that it is orthogoanl to $\mathbf{u}_1$. To this end, we can form the objective function:

$$\tilde{J}_2(\mathbf{u},\rho_1,\rho_2) = \mathbf{u}^T\mathbf{X}\mathbf{X}^T\mathbf{u} + \rho_1(1-\mathbf{u}^T\mathbf{u}) + \rho_2(\mathbf{u}^T\mathbf{u}_1)$$

Again, we can set its gradient to zero, as follows:

$$\nabla_{\mathbf{u}}\tilde{J}_2(\mathbf{u},\rho_1,\rho_2) = 0 \Rightarrow 2\mathbf{X}\mathbf{X}^T\mathbf{u} - 2\rho_1\mathbf{u} + \rho_2\mathbf{u}_1 = 0$$

If we pre-multiply the above equation by $\mathbf{u}_1$, we have

$$\mathbf{u}_1^T\mathbf{X}\mathbf{X}^T\mathbf{u} - 2\rho_1\mathbf{u}_1^T\mathbf{u} + \rho_2\mathbf{u}_1^T\mathbf{u}_1 = 0$$

Since $\mathbf{u}_1^T\mathbf{u} = 0$, the second term vanishes. The first term also vanishes since $\mathbf{u}_1^T\mathbf{X}\mathbf{X}^T\mathbf{u} = \lambda_1\mathbf{u}_1^T\mathbf{u} = 0$. Therefore $\rho_2 = 0$ and the original gradient equation becomes

$$2\mathbf{X}\mathbf{X}^T\mathbf{u} - 2\rho_1\mathbf{u} = 0,$$

whch indicates that the second principal component is still an eigenvectors of $\mathbf{X}\mathbf{X}^T$ and its total projection is the corresponding eigenvalue. As a result, the second principal component is $\mathbf{u}_2$ and the corresponding total projection is λ_2. By repeating this process, we can obtain the successive principal components as the eigenvectors $\mathbf{u}_1, \mathbf{u}_2 \cdots \mathbf{u}_m$ of $\mathbf{X}\mathbf{X}^T$. (Note that since $\mathbf{X}\mathbf{X}^T$ is symmetric, its eigenvectors $\{\mathbf{u}_1, \mathbf{u}_2, \cdots \mathbf{u}_m\}$ form an orthonomal basis with $\mathbf{u}_i^T\mathbf{u}_j = 0, \forall i \neq j$.)

The step-by-step guide for performing PCA on a dataset is as follows:

1. Find the sample mean of the dataset: $\mu = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$.
2. Compute the covariance matrix $\mathbf{C} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$.
3. Find the eigenvalues of $\mathbf{C}$ and arrange them into descending order $\{\lambda_1, \lambda_2, \cdots, \lambda_d\}$, with the corresponding eigenvectors $\{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_d\}$.

4. The transformation matrix is then $\mathbf{U}^T$, with $\mathbf{U} = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_d \\ | & | & & | \end{bmatrix}$. In other words, vector $\mathbf{x}$ after transformation is $\mathbf{y} = \mathbf{U}^T\mathbf{x}$. If we only want to keep the first 3 dimention, we can simply put only the first 3 eigenvectors ($\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$) into $\mathbf{U}$ directly.