Motivation of PCA

Given a dataset $D=\left\{(\mathbf{x}_i \in \Re^n, y_i \in \Re)\right\}_{i=1,2,\cdots,\mathrm{N}}$, which is assumed to be of zero mean (otherwise the centering preprocess is first performed), PCA determines a subspace of dimension $m \leq n$, such that after projection on this subspace, the statistical variation of the data is optimally retained. [1]

The above subspace is defined with *m* mutually orthogonal vectors, known as *principle directions*. To optimally retain the variation of the data, we need to maximize the variance of the data after projection onto the subspace.

Determine the first principle axis

PCA is usually solved in a step-wise fashion. First assume $m=1$ and let $u \in \Re^n$ denote the single principle axis. The variance of the data after projection (having assumed centered data) is

$$\begin{aligned} J(\mathbf{u}) &= \frac{1}{N}\sum_{i=1}^{N}(\mathbf{u}^T\mathbf{x}_i)^2 = \frac{1}{N}\sum_{i=1}^{N}\mathbf{u}^T\mathbf{x}_i\mathbf{x}_i^T\mathbf{u} \\ &= \frac{1}{N}\mathbf{u}^T\left[\mathbf{x}_1 \quad \mathbf{x}_1 \quad \cdots \quad \mathbf{x}_N\right]\underset{\triangleq X}{\begin{bmatrix}\mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_N^T\end{bmatrix}}\mathbf{u} \\ &= \frac{1}{N}\mathbf{u}^T XX^T\mathbf{u} \\ &= \mathbf{u}^T S\mathbf{u} \end{aligned} \tag{1.1}$$

where $S=\frac{1}{N}XX^T \in \Re^{N\times N}$ is the sample covariance matrix (because the sample mean is zero).

Now the optimization task is formulated as

$$\begin{aligned} &\max\ \mathbf{u}^T S\mathbf{u} \\ &s.t. \left\|\mathbf{u}\right\| = 1 \end{aligned} \tag{1.2}$$

where the constrain stems from the fact that we are only interested in the directions.

The optimization problem (1.2) is a constrained one and by introducing the Lagrange multiplier λ its corresponding Lagrangian is given by

$$L(\mathbf{u},\lambda) = \mathbf{u}^T S\mathbf{u} - \lambda(\mathbf{u}^T\mathbf{u} - 1) \tag{1.3}$$

whose gradient with respect to **u** is

$$\nabla_{\mathbf{u}} L = 2S\mathbf{u} - 2\lambda\mathbf{u}. \tag{1.4}$$

Setting the above gradient equal to zero, we get

$$S\mathbf{u} = \lambda\mathbf{u}. \tag{1.5}$$

Therefore, the principle direction is an eigenvector of the sample covariance matrix S. Plugging (1.5) into (1.2), we obtain

$$J(\mathbf{u}) = \lambda. \tag{1.6}$$

Therefore, to maximum J, we just choose the eigenvector $\mathbf{u}_1$ corresponding to the maximum eigenvalue as the first principle axis.

The other principle directions

The second principle component is chosen such that

a) It is orthogonal to $\mathbf{u}_1$ and
b) It maximizes the variance after the data projection onto this direction.

If we ignore the first constraint for the time being, then this leads to the same problem as the first principle direction, i.e., we still get $S\mathbf{u} = \lambda\mathbf{u}.$ Since the second principle is also an eigenvector of S, it must be orthogonal to $\mathbf{u_1}$ because S is a symmetric matrix. Thus, the first constraint is automatically satisfied. Now it is obvious that the second principle axis is the eigenvector corresponding to the second largest eigenvalue of S.

Similarly, the k^{th} principle axis is the eigenvector corresponding to the k^{th} largest eigenvalue of S. We continue this process until we find all the m principle directions as required.

PCA and SVD

Now the question lies in how to find the eigenvectors of *S* corresponding to its *m* largest eigenvalues. A naive method is to compute *S* first and then solve its eigenvalues and eigenvectors. However, in practical datasets, the number of training examples *N* is very large, which leads to a huge matrix $S = \frac{1}{N}XX^T \in \Re^{N\times N}$. As a result, to find the eigenvectors of S is both space and time expensive. This is where singular value decomposition can play a role.

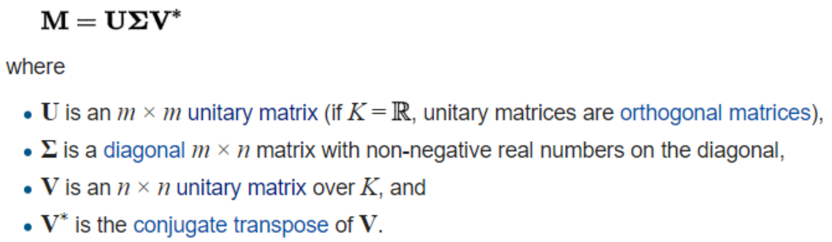
Assume the data matrix X is decomposed through SVD as follows

$$X = U\Lambda V^T. \tag{1.7}$$

Then, we know that the column vectors of U are eigenvectors of XX^T, which is equal to *NS*. Therefore, U contains the eigenvectors of S as columns corresponding to the non-zero eigenvalues.

In summary, assume the rank of X is *r*, then SVD tells that

1) U's columns are eigenvectors of XX^T corresponding to non-zero eigenvalues, which are also eigenvectors of S since there exists $S = \frac{1}{N}XX^T$.
2) Λ stores the singular values of XX^T (equivalently, X^TX).
3) In a full SVD, i.e., for $X \in \mathcal{R}^{m\times n}$, we have $U \in \mathcal{R}^{m\times m}$ and $V \in \mathcal{R}^{m\times n}$ and $\Lambda \in \Re^{m\times n}$.



$$\mathbf{M} = \mathbf{U\Sigma V^*}$$

where

- $\mathbf{U}$ is an $m \times m$ unitary matrix (if $K = \mathbb{R}$, unitary matrices are orthogonal matrices),
- $\mathbf{\Sigma}$ is a diagonal $m \times n$ matrix with non-negative real numbers on the diagonal,
- $\mathbf{V}$ is an $n \times n$ unitary matrix over K, and
- $\mathbf{V}^*$ is the conjugate transpose of $\mathbf{V}$.

U contains the left-singular vectors of M, i.e. the orthonormal eigenvectors of MM^{T}.
V contains the left-singular vectors of M, i.e. the orthonormal eigenvectors of $\mathrm{M}^{\mathrm{T}}M$.

4) In practice, we usually adopt a reduced form of SVD. For example, just keep the columns corresponding to the non-zero singular values. Number of non-zero singular values is also the rank of X.
5) Suppose

$$U = [u_1, \quad u_2, \quad \cdots, \quad u_m] \qquad \Sigma = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_p \end{bmatrix} \tag{1.8}$$
$$V = [v_1, \quad v_2, \quad \cdots, \quad v_n]$$

(Note generally $\Sigma \in \Re^{m \times n}$ is not square.)

Then we have

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T + \sum_{i=r+1}^{\min(n,m)} 0 \cdot u_i v_i^T$$

which means we can safely discard the columns of U and V corresponding to the zero singular values with no loss. Furthermore, we can ignore more terms if needed to get a so-called low-rank approximation.

Since for any matrix A, there exists $rank(A) = rank(AA^T) = rank(A^TA)$. We can say that the number of non-zero (positive) singular values is just the rank of A.

In other words, we can find the eigenvectors of S, the principle axes, easily in U.

Uncentered data

If the original data matrix X is uncentered, that is, its mean is not zero, then what shall we do?

In this case, the optimization objective (the variance after projection) is

$$\begin{aligned} J(\mathbf{u}) &= \frac{1}{N}\sum_{i=1}^{N} \mathbf{u}^T(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T\mathbf{u} \\ &= \mathbf{u}^T S \mathbf{u} \\ &= \frac{1}{N}\sum_{i=1}^{N} \mathbf{u}^T \overline{\mathbf{x}}_i \overline{\mathbf{x}}_i{}^T \mathbf{u} \\ &= \mathbf{u}^T \overline{S} \mathbf{u} \end{aligned} \tag{1.9}$$

where $\boldsymbol{\mu}$ is mean vector of all data samples and $\bar{\mathbf{x}}_i = \mathbf{x}_i - \boldsymbol{\mu}$ (centering). Again, we need to find the eigenvectors of the sample covariance matrix *S*.

However, in this scenario, $S = XX^T$ does NOT hold, which means we cannot simply apply SVD on X to get the eigenvectors of *S*.

Instead, if we define $\bar{\boldsymbol{x}}_i = x_i - \boldsymbol{\mu}$**,** then there exists $S = \bar{X}\bar{X}^T$. Subsequently, we apply SVD on $\bar{X}$.

In summary, if the original data is not centered, then we **should first center the data and then apply SVD on the centered data to get principle directions.**

From another perspective, since translation doesn't change the variance, therefore the principle axes we find for the centered data are exactly also the principle axes for the original data.

In fact, after centering the data, when we simply compute the eigenvector of $\bar{X}\bar{X}^T$, we are actually solving the eigenvectors of the original sample covariance matrix, as shown in (1.9).

The main benefit we obtain from centering data is that the eigenvector of $\bar{S} = \frac{1}{N}\bar{X}\bar{X}^T$ can be easily solved through the SVD of $\bar{X} = U\Lambda V^T$, which are the columns of U. SVD makes it possible to handle very large matrix.