Principal component analysis

Aapo Hyvärinen

Based on material from the book *Natural Image Statistics*
to be published by Springer-Verlag in 2009

March 19, 2009

1 Basic idea

Assume we have a random vector $\mathbf{x}$. How to characterize the distribution in simple terms? The first, obvious answer is the *mean.* To characterize the remaining distribution, you might think of computing the variances of the variables. However, this does not really characterize a multivariate distribution well in general. Look at Fig. 1 and you see that what we would (intuitively) like to describe is the how the data "cloud" is elongated or directed. Principal component analysis is one answer to this problem.

Let us actually subtract the mean from the random vector, so that each x_i has zero mean. Thus, we can better concentrate on the structure which is there in addition to the mean. In the following, it is assumed that all the variables have zero mean. Actually, we are dealing with linear combinations of the observed variables, so all these linear combinations also have zero mean.

Principal components are linear combinations $s = \sum_i w_i x_i$ that contain (or "explain") as much of the variance of the input data as possible. It turns out that the amount of variance explained is directly related to the variance of the component, as will be discussed in Section 2 below. Thus, the first principal component is intuitively defined as the linear combination of observed variables, which has the maximum variance. The idea is to find the "main axis" of the data cloud, which is illustrated in Fig. 1.

Some constraint on the weights w_i, which we call the *principal component weights*, must be imposed as well. If no constraint were imposed, the maximum of the variance would be attained when w_i becomes infinitely large (and the minimum would be attained when all the w_i are zero). In fact, just multiplying all the weights in w_i by a factor of two, we would get a variance that is four times as large, and by dividing all the coefficients by two the variance decreases to one quarter.

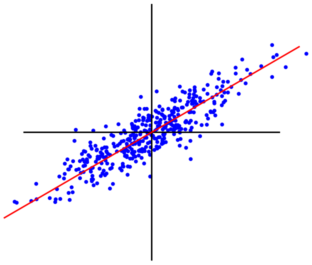

Figure 1: Illustration of PCA. The principal component of this (artificial) two-dimensional data is the oblique axis plotted. Projection on the principal axis explains more of the variance of the data than projection on any other axis. It describes the data in some way better than just looking at the variances.

A natural thing to do is to constrain the norm of the vector $\mathbf{w} = (w_1, \ldots, w_n)$:

$$\|\mathbf{w}\| = \sqrt{\sum_i w_i^2} = 1 \tag{1}$$

For simplicity, we constrain the norm to be equal to one, but any other value would give the same results.

This definition gives only one principal component. Typically, the way to obtain many principal components is by a “deflation” approach: After estimating the first principal component, we want to find the linear combination of maximum variance *under the constraint* that the new combination must be *orthogonal* to the first one (i.e. the dot-product is zero, as in Equation 4). This will then be called the second principal component. This procedure can be repeated to obtain as many components as there are dimensions in the data space. To put this formally, assume that we have estimated k principal components, given by the weight vectors $\mathbf{w}_1, \mathbf{w}_2 \ldots, \mathbf{w}_k$. Then the $k+1$-th principal component weight vector is defined by

$$\max_{\mathbf{w}} \operatorname{var}\left(\sum_i w_i x_i\right) \tag{2}$$

under the constraints

$$\|\mathbf{w}\| = \sqrt{\sum_i w_i^2} = 1 \tag{3}$$

$$\sum_i w_{ji} w_i = 0 \text{ for all } j = 1, \ldots, k \tag{4}$$

2 Dimension reduction by PCA

One task where PCA is very useful is in reducing the dimension of the data so that the maximum amount of the variance is preserved.

Consider the following general problem. We have a very large number, say m, of random variables $x_1, \ldots, x_m$. Computations that use all the variables would be too burdensome. We want to reduce the dimension of the data by linearly transforming the variables into a smaller number, say n, of variables that we denote by $z_1, \ldots, z_n$:

$$z_i = \sum_{j=1}^{m} w_{ij} x_j, \text{ for all } i = 1, \ldots, n \tag{5}$$

The number of new variables n might be only 10% or 1% of the original number m. We want to find the new variables so that they preserve as much information on the original data as possible. This "preservation of information" has to be exactly defined. The most wide-spread definition is to look at the squared error that we get when we try to reconstruct the original data using the z_i. That is, we reconstruct x_j as a linear transformation $\sum_i a_{ji} z_i$, minimizing the average error

$$E\left\{\sum_j \left(x_j - \sum_i a_{ji} z_i\right)^2\right\} = E\left\{\|\mathbf{x} - \sum_i \mathbf{a}_i z_i\|^2\right\} \tag{6}$$

where the a_{ji} are also determined so that they minimize this error. For simplicity, let us consider only transformations for which the transforming weights are orthogonal and have unit norm:

$$\sum_j w_{ij}^2 = 1, \text{ for all } i \tag{7}$$

$$\sum_j w_{ij} w_{kj} = 0, \text{ for all } i \neq k \tag{8}$$

What is the best way of doing this dimension reduction? The solution is to take as the z_i the n first principal components! Furthermore, the optimal reconstruction weight vectors $\mathbf{a}_i$ in Equation (6) are given by the very same principal components weights which compute the z_i. A very simple version of this result is shown in the exercises. The general result is not proven here because it is quite complicated.

The solution is not uniquely defined, though, because any orthogonal transformation of the z_i is just as good. This is understandable because any such transformation of the z_i contains just the same information: we can make the inverse transformation to get the z_i from the transformed ones.

3 Solution of PCA using eigenvalue decomposition

Here we present the fundamental proof of how PCA is related to eigenvalues of the covariance matrix.

3.1 Definitions

The variance of a random variable is defined as

$$\mathrm{var}(x_1) = E\{x_1^2\} - (E\{x_1\})^2 \tag{9}$$

This can also be written $\mathrm{var}(x_1) = E\{(x_1^2 - E\{x_1\})^2\}$, which more clearly shows how variance measures average deviation from the mean value.

When we have more than one random variable, it is useful to analyze the *covariance*:

$$\mathrm{cov}(x_1, x_2) = E\{x_1 x_2\} - E\{x_1\}E\{x_2\} \tag{10}$$

which measures how well we can predict the value of one of the variables using a simple linear predictor, as will be seen below.

If the covariance is zero, which is equivalent to saying that the correlation coefficient is zero, the variables are said to be *uncorrelated*.

3.2 Covariance matrix

The variances and covariances of the elements of a random vector $\mathbf{x}$ are often collected to a *covariance matrix* whose i, j-th element is simply the covariance of x_i and x_j:

$$\mathbf{C}(\mathbf{x}) = \begin{pmatrix} \mathrm{cov}(x_1, x_1) & \mathrm{cov}(x_1, x_2) & \dots & \mathrm{cov}(x_1, x_n) \\ \mathrm{cov}(x_2, x_1) & \mathrm{cov}(x_2, x_2) & \dots & \mathrm{cov}(x_2, x_n) \\ \vdots & & \ddots & \vdots \\ \mathrm{cov}(x_n, x_1) & \mathrm{cov}(x_n, x_2) & \dots & \mathrm{cov}(x_n, x_n) \end{pmatrix} \tag{11}$$

Note that the covariance of x_i with itself is the same as the variance of x_i. So, the diagonal of the covariance matrix gives the variances. The covariance matrix is basically a generalization of variance to random vectors: in many cases, when moving from a single random variable to random vectors, the covariance matrix takes the place of variance.

In matrix notation, the covariance matrix is simply obtained as a generalization of the one-dimensional definitions in Equations (10) and (9) as

$$\mathbf{C}(\mathbf{x}) = E\{\mathbf{x}\mathbf{x}^T\} - E\{\mathbf{x}\}E\{\mathbf{x}\}^T \tag{12}$$

where taking the transposes in the correct places is essential. In most of this book, we will be dealing with random variables whose means are zero, in which case the second term in Equation (12) is zero.

If the variables are uncorrelated, the covariance matrix is diagonal. If they are all further standardized to unit variance, the covariance matrix equals the identity matrix.

3.3 Eigenvalues of covariance matrix

The important point to note is that the variance of any linear combination can be computed using the *covariance matrix* of the data. That is, considering any linear combination $\mathbf{w}^T\mathbf{x} = \sum_i w_i x_i$ we can compute its variance simply by:

$$E\{(\mathbf{w}^T\mathbf{x})^2\} = E\{(\mathbf{w}^T\mathbf{x})(\mathbf{x}^T\mathbf{w})\} = E\{\mathbf{w}^T(\mathbf{x}\mathbf{x}^T)\mathbf{w}\} = \mathbf{w}^T E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w} \\ = \mathbf{w}^T\mathbf{C}\mathbf{w} \quad (13)$$

where we denote the covariance matrix by $\mathbf{C} = E\{\mathbf{x}\mathbf{x}^T\}$. So, the basic PCA problem can be formulated as

$$\max_{\mathbf{w}:\|\mathbf{w}\|=1} \mathbf{w}^T\mathbf{C}\mathbf{w} \quad (14)$$

(We assume here that the mean is zero: $E\{\mathbf{x}\} = \mathbf{0}$).

A basic concept in linear algebra is the eigenvalue decomposition. The starting point is that $\mathbf{C}$ is a symmetric matrix, because $\mathrm{cov}(x_i, x_j) = \mathrm{cov}(x_j, x_i)$. In linear algebra, it is shown that any symmetric matrix can be expressed as a product of the form:

$$\mathbf{C} = \mathbf{U}\mathbf{D}\mathbf{U}^T \quad (15)$$

where $\mathbf{U}$ is an orthogonal matrix, and $\mathbf{D} = \mathrm{diag}(\lambda_1, \ldots, \lambda_m)$ is diagonal. The columns of $\mathbf{U}$ are called the *eigenvectors*, and the λ_i are called the *eigenvalues*. Many efficient algorithms exist for computing the eigenvalue decomposition of a matrix.

Now, we can solve PCA easily. Lets us make the change of variables $\mathbf{v} = \mathbf{U}^T\mathbf{w}$. Then we have

$$\mathbf{w}^T\mathbf{C}\mathbf{w} = \mathbf{w}^T\mathbf{U}\mathbf{D}\mathbf{U}^T\mathbf{w} = \mathbf{v}^T\mathbf{D}\mathbf{v} = \sum_i v_i^2 \lambda_i \quad (16)$$

Because $\mathbf{U}$ is orthogonal, $\|\mathbf{v}\| = \|\mathbf{w}\|$, so the constraint is the same for $\mathbf{v}$ as it was for $\mathbf{w}$. Let us make the further change of variables to $m_i = v_i^2$. The constraint of unit norm of $\mathbf{v}$ is now equivalent to the constraints that the sum of the m_i must equal one (they must also be positive because they are squares). Then, the problem is transformed to

$$\max_{m_i \geq 0, \sum m_i = 1} \sum_i m_i \lambda_i \quad (17)$$

It is rather obvious that the maximum is found when the m_i corresponding to the largest λ_i is one and the others are zero. Let us denote by i^* the index of the maximum eigenvalue. Going back to the $\mathbf{w}$, this corresponds to $\mathbf{w}$ begin equal to the i^*-th eigenvector, that is, the i^*-th column of $\mathbf{U}$. Thus, we see how the first principal component is easily computed by the eigenvalue decomposition.

Since the eigenvectors of a symmetric matrix are orthogonal, finding the second principal component means maximizing the variance so that v_{i^*} is kept zero. This is actually equivalent to making the new $\mathbf{w}$ orthogonal to the first eigenvector. Thus, in terms of m_i, we have exactly the same optimization problem, but with the extra constraint that $m_{i^*} = 0$. Obviously, the optimum is obtained when $\mathbf{w}$ is equal to the eigenvector corresponding to the *second* largest eigenvalue. This logic applies to the k-th principal component.

Thus, all the principal components can be found by ordering the eigenvectors $\mathbf{u}_i, i = 1, \ldots, m$ in $\mathbf{U}$ so that the corresponding eigenvalues are in decreasing order. Let us assume that $\mathbf{U}$ is ordered so. Then the i-th principal component s_i is equal to

$$s_i = \mathbf{u}_i^T \mathbf{x} \tag{18}$$

Note that it can be proven that the λ_i are all non-negative for a covariance matrix.

Using the eigenvalue decomposition, we can prove some interesting properties of PCA. First, the principal components are *uncorrelated*, because for the vector of the principal components

$$\mathbf{s} = \mathbf{U}^T \mathbf{x} \tag{19}$$

we have

$$\begin{aligned} E\{\mathbf{s}\mathbf{s}^T\} = E\{\mathbf{U}^T\mathbf{x}\mathbf{x}^T\mathbf{U}\} = \mathbf{U}^T E\{\mathbf{x}\mathbf{x}^T\}\mathbf{U} = \mathbf{U}^T(\mathbf{U}\mathbf{D}\mathbf{U}^T)\mathbf{U} \\ = (\mathbf{U}^T\mathbf{U})\mathbf{D}(\mathbf{U}^T\mathbf{U}) = \mathbf{D} \end{aligned} \tag{20}$$

because of the orthogonality of $\mathbf{U}$. Thus, the covariance matrix is diagonal, which shows that the principal components are uncorrelated.

Moreover, we see that the variances of the principal components are equal to the λ_i.

3.4 Uniqueness of PCA

The fact that the variances of the principal components are given by λ_i has an important implication for the *uniqueness* of PCA. If two of the eigenvalues are equal, then the variance of those principal components are equal. Then, the principal components are not well-defined anymore, because we can make a *rotation* of those principal components without affecting their variances. This is because if z_i and z_{i+1} have the same variance, then linear combinations such as $\sqrt{1/2}z_i + \sqrt{1/2}z_{i+1}$ and $\sqrt{1/2}z_i - \sqrt{1/2}z_{i+1}$ have the same variance as well; all the constraints (unit variance and orthogonality) are still fulfilled, so these are equally valid principal components. In fact, in linear algebra, it is well-known that the eigenvalue decomposition is uniquely defined only when the eigenvalues are all distinct.

The lack of uniqueness is not very serious in the case of dimension reduction: What matters in the dimension reduction context is not so much the actual components themselves, but the *subspace* which they span. The *principal subspace* means the set of all possible linear combinations of the n first principal components. It corresponds to taking all possible linear combinations of the principal component weight vectors $\mathbf{w}_i$ associated with the n principal components. So, the n-dimensional principal subspace is usually uniquely defined even if some principal components have equal variances. Of course, it may happen that the n-th and the $(n+1)$-th principal components have equal variances, and that we cannot decide which one to include in the subspace. But the effect on the whole subspace is usually quite small and can be ignored in practice.

3.5 Proportion of variance explained by the components

Above, we saw that the eigenvalues of the covariance matrix give the variance of each component. This is typically interpreted as the amount of the total variance of the data which is "explained" by the component. If we take m first principal components, they together explain the variance of amount $\sum_{i=1}^{k} \lambda_i$. Typically, this is reported as a percentage of the total variance:

$$\text{proportion of variance explained} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \tag{21}$$

where n is the dimension of the data. In fact, the denominator is equal to $\sum_{i=1}^{n} \text{var}(x_i)$ (prove this), so it is really the total variance of the data.

[End of 3nd lecture here, the rest will be in the 4rd lecture]

4 Illustration

We illustrate PCA on an artificial data set, given by the matrix

$$\mathbf{X} = \begin{pmatrix} 5 & 3 & 0 & 1 & -1 & -3 & 5 & 0 & -4 & -4 \\ -2 & -1 & 0 & 0 & 1 & 4 & -3 & 1 & 5 & 3 \\ 0 & 1 & 4 & -1 & 0 & 5 & 5 & -5 & -3 & -3 \\ 0 & 2 & 3 & 0 & -1 & 3 & 3 & -7 & -2 & 0 \\ 3 & 4 & -2 & 1 & 3 & -3 & -3 & 2 & 0 & 0 \end{pmatrix} \tag{22}$$

where each row is one variable and each column is one observation. The eigenvalues of the covariance matrix are:

$$\begin{pmatrix} 25.6351 & 16.1255 & 3.0215 & 0.9756 & 0.3201 \end{pmatrix} \tag{23}$$

which shows that the first two principal components explain 90.6% of the variance. The corresponding two eigenvectors of the covariance matrix are

$$\mathbf{u}_1 = \begin{pmatrix} -0.4170 & 0.3237 & -0.6399 & -0.5184 & 0.2075 \end{pmatrix}^T \tag{24}$$

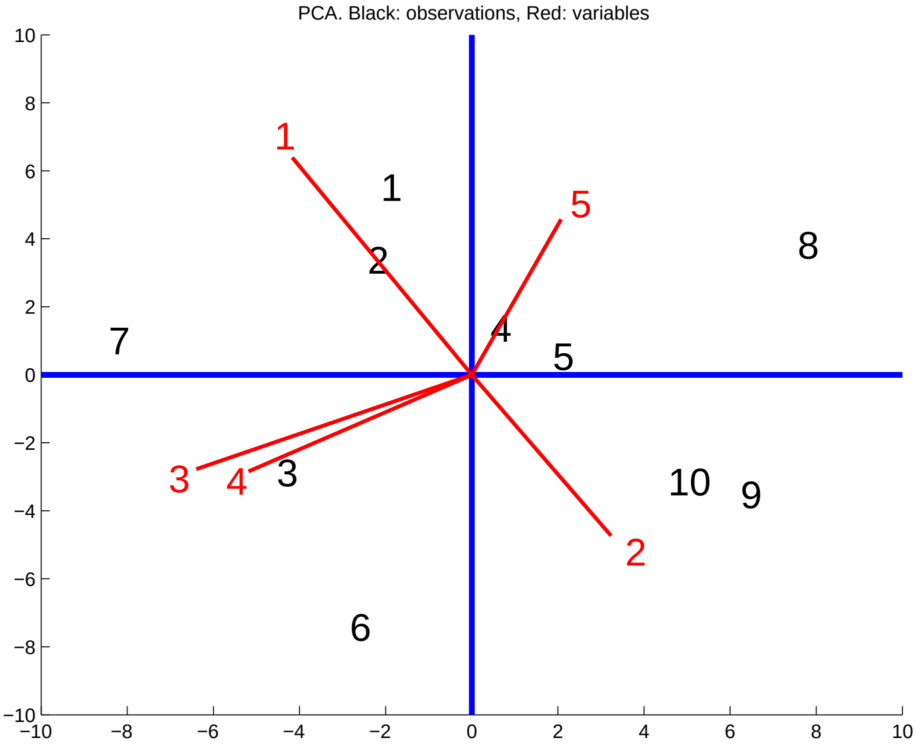

Figure 2: Illustration of application of PCA on an artificial data set.

and

$$\mathbf{u}_2 = \begin{pmatrix} 0.6393 & -0.4736 & -0.2777 & -0.2841 & 0.4574 \end{pmatrix}^T \tag{25}$$

We can plot each observation by projecting it on the coordinate axes given by $\mathbf{u}_1$ and $\mathbf{u}_2$. Moreover, we can plot the original variables, simply by using the values in $\mathbf{u}_1$ and $\mathbf{u}_2$. See Figure 2.

We can basically see that similar observations are close to each other, and similar variables are close to each other (and variables with opposite signs are pointing in opposite directions).

5 Whitening

5.1 Whitening as normalized decorrelation

Another task for which PCA is quite useful is whitening. Whitening is an important preprocessing method where the variables x_i are transformed to a set of new variables $s_1, \ldots, s_n$ so that the s_i are uncorrelated and have unit variance:

$$E\{s_i s_j\} = \begin{cases} 0 \text{ if } i \neq j \\ 1 \text{ if } i = j \end{cases} \tag{26}$$

(It is assumed that all the variables have zero mean.) It is also said that the resulting vector $(s_1, \ldots, s_n)$ is white.

A central property of whitened data is that the variance is the same in all directions:

$$\mathrm{var}(\mathbf{w}^T\mathbf{x}) = 1 \text{ for any } \mathbf{w} : \|\mathbf{w}\| = 1 \tag{27}$$

Related properties are considered below.

In addition to the principal components weights being orthogonal, the principal components themselves are uncorrelated, as will be shown in more detail in Section 3.3. So, after PCA, the only thing we need to do to get whitened data is to normalize the variances of the principal components by dividing them by their standard deviations. Denoting the principal components by y_i, this means we compute

$$s_i = \frac{y_i}{\sqrt{\mathrm{var}(y_i)}} \tag{28}$$

to get whitened components s_i. Whitening is a useful preprocessing method that is often used before other learning. Whitening by PCA is illustrated in Figure 3.

5.2 The family of whitening transformations

It must be noted that there are many whitening transformations. In fact, if the random variables $s_i, i = 1 \ldots, n$ are white, then any *orthogonal transformation* of those variables is also white (the proof is left as an exercise). Often, whitening is based on PCA because PCA is a well-known method that can be computed very fast, but it must be kept in mind that PCA is just one among the many whitening transformations.

Later, we will often use the fact that the connection between orthogonality and uncorrelatedness is even stronger for whitened data. In fact, if we compute two linear components $\sum_i v_i s_i$ and $\sum_i w_i s_i$ from white data, they are uncorrelated *only* if the two vectors $\mathbf{v}$ and $\mathbf{w}$ (which contain the entries v_i and w_i, respectively) are orthogonal.

In general, we have the following theoretical result. For white data, multiplication by a square matrix gives white components if *and only if* the matrix is orthogonal. Thus, when we have computed one particular whitening transformation, we also know that *only* orthogonal transformations of the transformed data can be white.

Note here the tricky point in terminology: a matrix if called orthogonal if its columns, or equivalently its rows, are orthogonal, *and* the norms of its columns are all equal to one. To emphasize this, some authors call an orthogonal matrix ortho*normal*. We stick to the word "orthogonal" here.

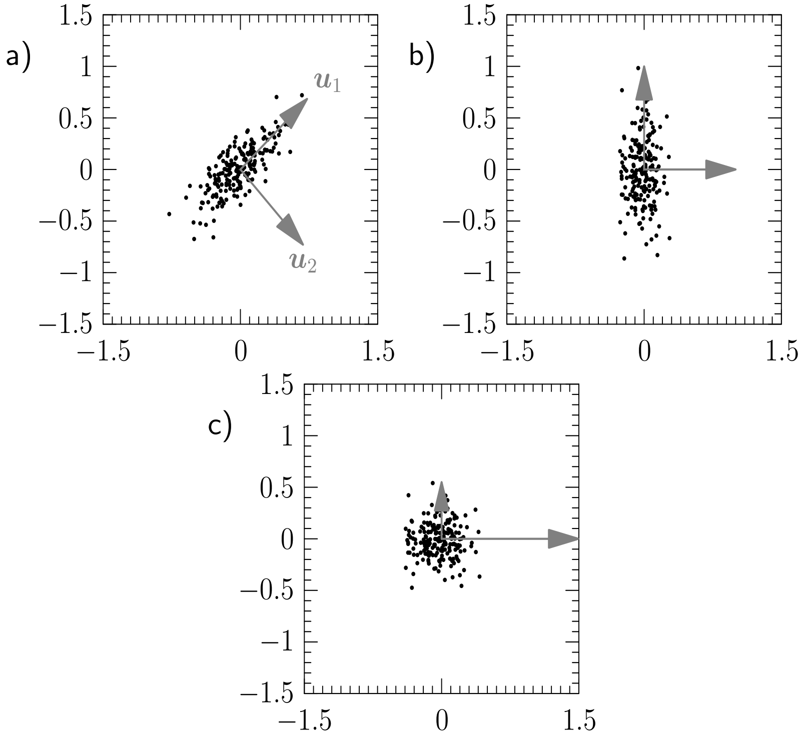

Figure 3: Illustration of PCA and whitening. a) The original data "cloud". The arrows show the principal components. The first one points in the direction of the largest variance in the data, and the second in the remaining orthogonal direction. b) When the data is transformed to the principal components, i.e. the principal components are taken as the new coordinates, the variation in the data is aligned with those new axes, which is because the principal components are uncorrelated. c) When the principal components are further normalized to unit variance, the data cloud has equal variance in all directions, which means it has been whitened. The change in the lengths of the arrows reflects this normalization; the larger the variance, the shorter the arrow.

5.3 Whitening exhausts second-order information

Information contained in the covariance matrix is called “second-order” information. It is often said that whitening exhausts second-order information. This is natural because it makes the covariance matrix “trivial”, i.e. equal to identity. Another viewpoint is to consider what happens if we do PCA on whitened data. Basically, there is no point in doing PCA because the variance is the same in all directions!

After whitening, the only information that is left in the data is of “higher order” (i.e. higher than second). Later, when talking about independent component analysis, we will see what such higher-order information is.