Notes on Principal Component Analysis (PCA)

Gabriele Tolomei

March 25, 2020

In many use cases, such as computer vision or natural language processing, data is naturally represented by vectors in a very high-dimensional space (i.e., a space made of thousands or even million of dimensions). High-dimensional data suffer from the well-known problem which is typically referred to as the *curse of dimensionality.* Very roughly, this refers to the inability to distinguish between data points that are close from those that are far away from each other, since data points in high-dimensional space tend in fact to be all distant (and sparser) from each other.
Principal Component Analysis (PCA) is an effective technique to reduce data dimensionality, which identifies a low-dimensional sub-space so as to preserve as much as possible the "structure" (i.e., variance) of the data represented in the original, high-dimensional space.

1 Preliminaries

We are given with a set of n data points $\mathbf{x}_1, \ldots, \mathbf{x}_n$, each one laying on a d-dimensional space, such that $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,d})$, for all $i = \{1, \ldots, n\}$. Moreover, we associate a random variable X_j to each dimension (i.e., $j = \{1, \ldots, d\}$). On top of that, we define the expected value of each X_j as the mean computed from the n observations (of that dimension j). In other words:

$$E[X_j] = \mu_j = \frac{1}{n}\sum_{i=1}^{n} x_{i,j}$$

Then, we first rewrite all the data points by "centering" them around the mean (i.e., we substract from each dimension its corresponding mean):

$$\mathbf{x}_i = (x_{i,1} - \mu_1, \ldots, x_{i,d} - \mu_d)$$

Once we do that, the values of each random variable X_j is changed accordingly, so that all the dimensions will now have 0 mean:

$$E[X_j] = \mu_j^{\text{new}} = \frac{1}{n}\sum_{i=1}^{n}(x_{i,j} - \mu_j) = \frac{1}{n}\left(\sum_{i=1}^{n} x_{i,j} - \sum_{i=1}^{n} \mu_j\right) =$$

$$\frac{1}{n}\left(\sum_{i=1}^{n} x_{i,j} - n\mu_j\right) = \frac{1}{n}\left(\sum_{i=1}^{n} x_{i,j} - n\frac{1}{n}\sum_{i=1}^{n} x_{i,j}\right) = \frac{1}{n}\left(\sum_{i=1}^{n} x_{i,j} - \sum_{i=1}^{n} x_{i,j}\right) = 0$$

Moreover, in a d-dimensional space, we define the *covariance matrix* K as the d-by-d square matrix as follows:

$$K = \begin{bmatrix} \text{Cov}(X_1, X_1) & \text{Cov}(X_1, X_2) & \dots & \text{Cov}(X_1, X_d) \\ \text{Cov}(X_2, X_1) & \text{Cov}(X_2, X_2) & \dots & \text{Cov}(X_2, X_d) \\ \vdots & \vdots & \vdots & \vdots \\ \text{Cov}(X_d, X_1) & \text{Cov}(X_d, X_2) & \dots & \text{Cov}(X_d, X_d) \end{bmatrix}$$

where:

$$\text{Cov}(X_j, X_k) = E[(X_j - E[X_j])(X_k - E[X_k])]$$

Under the assumption that $E[X_j] = 0$ for all j after properly subtracting the mean of each dimension from all the n observations, the equation above turns into:

$$\text{Cov}(X_j, X_k) = E[X_j X_k] = \frac{1}{n}\sum_{i=1}^{n} x_{i,j} x_{i,k}$$

Note that $\text{Cov}(X_j, X_k) = \text{Cov}(X_k, X_j)$, i.e., the covariance matrix is symmetric. Moreover, observe that on the main diagonal of the K we have the following:

$$\text{Cov}(X_j, X_j) = E[X_j X_j] = \frac{1}{n}\sum_{i=1}^{n} x_{i,j}^2 = (E[X_j - \underbrace{E[X_j]}_{=0}])^2 = \text{Var}(X_j)$$

Overall, we have:

$$K = \begin{bmatrix} \text{Var}(X_1) & \text{Cov}(X_1, X_2) & \dots & \text{Cov}(X_1, X_d) \\ \text{Cov}(X_2, X_1) & \text{Var}(X_2) & \dots & \text{Cov}(X_2, X_d) \\ \vdots & \vdots & \vdots & \vdots \\ \text{Cov}(X_d, X_1) & \text{Cov}(X_d, X_2) & \dots & \text{Var}(X_d) \end{bmatrix}$$

We have already “visually” convinced ourselves that in order to find the principal components of our d-dimensional space, we must find the *eigenvectors* (and their associated *eigenvalues*) of the covariance matrix K. In other words, we need to solve the following equation:

$$K\mathbf{e} = \lambda\mathbf{e}$$

where $\mathbf{e}$ is a d-dimensional eigenvector and λ is the corresponding eigenvalue.

The equation above can be rewritten as follows:

$$K\mathbf{e} - \lambda\mathbf{e} = \mathbf{0} \Rightarrow (K - \lambda I)\mathbf{e} = \mathbf{0}$$

where I is a d-by-d *identity matrix.*

We therefore need to solve the above *homogeneous* system of equations; any homogeneous system has always a *trivial* solution (i.e., in the case above the zero-vector $\mathbf{e} = \mathbf{0}$). The only way for a homogeneous system like the one above to have non-trivial solutions is for its matrix $(K - \lambda I)$ to be *non-invertible.* If $(K - \lambda I)$ is invertible then we can multiply by its inverse $(K - \lambda I)^{-1}$ both sides of the equation:

$$(K - \lambda I)(K - \lambda I)^{-1}\mathbf{e} = \mathbf{0}(K - \lambda I)^{-1}$$

Eventually, the only solution we obtain is still $\mathbf{e} = \mathbf{0}$.

From linear algebra theory we know that a square matrix like $(K - \lambda I)$ is invertible iff its determinant is **not** equal to 0; on the other hand, if the determinant of $(K - \lambda I)$ is equal to 0 then that matrix will not be invertible, and therefore the corresponding homogeneous system will have a non-trivial solution.

As a result, we must solve for λ the following equation:

$$\det(K - \lambda I) = 0$$

The equation above is called the *characteristic equation* or *characteristic polynomial* of K. It is a polynomial function in λ of degree d. So we know that this equation will not have more than d roots or solutions, therefore no more than d eigenvalues.

Assuming we find d eigenvalues $\lambda_1, \ldots, \lambda_d$ as the solutions of the characteristic equation above. Then, we can plug each of these eigenvalues to in turn figure out the corresponding eigenvector.

Finally, eigenvectors must be divided by L2-norm in order to normalize them as length-1 vectors.

In the next section, we formally prove the two statements below:

1. The eigenvectors maximize the variance among all possible data directions (i.e., data projections);

2. We pick the eigenvector with the largest eigenvalue $\lambda_{\max}$ as the first principal component because $\lambda_{\max}$ is exactly the variance of the data along that eigenvector (i.e., projecting data onto any other eigenvector will result in a smaller variance).

2 Eigenvectors: Greatest Variance Direction

Let's consider again our set of n d-dimensional input data points above $\mathbf{x}_1, \ldots, \mathbf{x}_n$, adequately centered around the mean, i.e., $\mathbf{x}_i = (x_{i,1} - \mu_1, \ldots, x_{i,d} - \mu_d)$.

Let's also assume $\mathbf{e} = (e_1, \ldots, e_d)$ is the d-dimensional vector pointing towards the direction of the greatest variance we aim to find. Suppose we want to project each vector $\mathbf{x}_i$ onto the vector $\mathbf{e}$. Generally speaking, the *vector projection* of a vector $\mathbf{a}$ on another (non-zero) vector $\mathbf{b}$ is a vector whose magnitude is the *scalar projection* of **a** on **b** with the same direction as **b**. In other words, it is defined as:

$$\mathbf{a}_{||} = (||\mathbf{a}|| \cos\theta)\hat{\mathbf{b}}$$

where $(||\mathbf{a}|| \cos\theta)$ is the *scalar projection* of the vector projection (i.e., the scaling factor), assuming θ is the angle between **a** and **b**, and $\hat{\mathbf{b}} = \frac{\mathbf{b}}{||\mathbf{b}||}$ is the unit vector having the same direction of **b**. When θ is known we can compute:

$$\cos\theta = \frac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{a}||||\mathbf{b}||}$$

As such, the scalar projection can be computed in terms of the dot product by noticing that:

$$||\mathbf{a}|| \cos\theta = ||\mathbf{a}|| \frac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{a}||||\mathbf{b}||} = \frac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{b}||}$$

Similarly, we can compute the vector projection as follows:

$$\mathbf{a}_{||} = \underbrace{\frac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{b}||}}_{||\mathbf{a}|| \cos\theta} \underbrace{\frac{\mathbf{b}}{||\mathbf{b}||}}_{\hat{\mathbf{b}}} = \frac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{b}||^2}\mathbf{b}$$

Going back to our setting, we aim at considering the scalar projection of each $\mathbf{x}_i$ onto **e**. By using the definition above, the scalar projection can be computed as follows:

$$\frac{\mathbf{x} \cdot \mathbf{e}}{||\mathbf{e}||}$$

By enforcing **e** to be normalized (i.e., a unit-length vector, such that $||\mathbf{e}|| = 1$), the scalar projection simply turns into computing the dot product between **x** and **e**:

$$\mathbf{x}_i \cdot \mathbf{e} = \sum_{j=1}^{d} x_{i,j} e_j$$

Let's now compute what is the variance of the scalar projections of *all* our n input data points, as measured along the direction of **e**:

$$V(\mathbf{e}) = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i \cdot \mathbf{e} - \mu)^2 = \frac{1}{n} \sum_{i=1}^{n} \Big(\sum_{j=1}^{d} x_{i,j} e_j - \mu \Big)^2$$

where μ is the mean computed among all the scalar projections.

First of all, we will show that such $\mu = 0$. Indeed, by definition we have:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} \underbrace{\Big(\sum_{j=1}^{d} x_{i,j} e_j \Big)}_{\mathbf{x}_i \cdot \mathbf{e}}$$

$$\mu = \sum_{j=1}^{d} e_j \left(\frac{1}{n} \sum_{i=1}^{n} x_{i,j} \right)$$

where e_j can be factored out from the internal summation, as it does not depend on i. By our initial assumption (i.e., each input data point being centered around the mean), we know that $\frac{1}{n}\sum_{i=1}^{n} x_{i,j} = 0$, for all $j \in \{1, \ldots, d\}$, and therefore $\mu = 0$.

Putting all together, the variance as measured when data points are projected onto $\mathbf{e}$ is:

$$V(\mathbf{e}) = \frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)^2$$

Eventually, we want to find that vector $\mathbf{e}^*$, such that it maximizes the above quantity. In other words:

$$\mathbf{e}^* = \text{argmax}_{\mathbf{e}}\{V(\mathbf{e})\} = \text{argmax}_{\mathbf{e}}\left\{\frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)^2\right\}$$

Now, if we haven't put the constraint on the length of $\mathbf{e}$ – such that $||\mathbf{e}|| = 1$ – the optimization problem above will not be solvable, as we can always be able to find a vector whose magnitude increases the variance. Therefore, the actual problem we want to solve is a *constrained* optimization problem defined as follows:

$$\mathbf{e}^* = \text{argmax}_{\mathbf{e}}\left\{\frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)^2\right\}$$

$$\text{s.t. } ||\mathbf{e}|| = 1 \Rightarrow ||\mathbf{e}|| - 1 = 0$$

Whenever we aim to maximize (or minimize) a function which is subject to an equality constraint like the one defined above, the method of Lagrange multipliers is used. As such, the optimization problem turns into the following:

$$\mathbf{e}^* = \text{argmax}_{\mathbf{e}}\left\{\frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)^2 - \lambda\Big[\underbrace{\Big(\sum_{j=1}^{d} e_j^2\Big) - 1}_{\text{constraint: } ||\mathbf{e}||-1=0}\Big]\right\}$$

To solve the optimization problem above, we have to take the gradient of the function, set it to 0, and solve it for $\mathbf{e}$:

$$\nabla V(\mathbf{e}) = \nabla\left\{\frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)^2 - \lambda\Big[\Big(\sum_{j=1}^{d} e_j^2\Big) - 1\Big]\right\}$$

The gradient is just the d-dimensional vector of all the partial derivatives of $V(\mathbf{e})$ w.r.t. each dimension of $\mathbf{e}$, i.e., $e_1, \ldots, e_d$:

$$\nabla V(\mathbf{e}) = \left(\frac{\partial V(\mathbf{e})}{\partial e_1}, \ldots, \frac{\partial V(\mathbf{e})}{\partial e_d}\right)$$

The generic k-th component of the gradient is computed as follows:

$$\frac{\partial V(\mathbf{e})}{\partial e_k} = \frac{2}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)x_{i,k} - 2\lambda e_k$$

In order to find $\mathbf{e}^*$, we have to set $\nabla V(\mathbf{e}) = \mathbf{0}$, which is equal to set *all* of its components $\frac{\partial V(\mathbf{e})}{\partial e_k} = 0$ simultaneously, and solve it for $\mathbf{e}$.

Let's work out how to solve the equation below for the k-th component, and generalize it to all the components:

$$\frac{2}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)x_{i,k} - 2\lambda e_k = 0 \Rightarrow \frac{2}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)x_{i,k} = 2\lambda e_k$$

We can again factor out e_j from the summation, as it does not depend on i:

$$2\sum_{j=1}^{d} e_j \underbrace{\frac{1}{n}\sum_{i=1}^{n}(x_{i,k}x_{i,j})}_{\text{Cov}(X_k,X_j)} = 2\lambda e_k$$

$$=$$

$$\sum_{j=1}^{d} e_j \underbrace{\frac{1}{n}\sum_{i=1}^{n}(x_{i,k}x_{i,j})}_{\text{Cov}(X_k,X_j)} = \lambda e_k$$

The equation above must hold contemporarily for all k, i.e., $k = 1 \dots d$. Therefore, we have to solve the following system of equations:

$$\begin{cases} \sum_{j=1}^{d} \text{Cov}(X_1, X_j)e_j = & \lambda e_1 \\ \dots & \dots \\ \sum_{j=1}^{d} \text{Cov}(X_d, X_j)e_j = & \lambda e_d \end{cases}$$

Each equation above is the dot product of a row of the covariance matrix K with the vector $\mathbf{e}$. In other words, we want to find a solution to:

$$K\mathbf{e} = \lambda\mathbf{e}$$

As we have already seen in the previous section, $\mathbf{e}$ is a non-trivial solution to the homogeneous system above if it is an eigenvector, and this proves what we wanted.

To summarize:

1. We start from computing the variance $V(\mathbf{e})$ of our input data points w.r.t. a generic projection vector $\mathbf{e}$;

2. We try to maximize $V(\mathbf{e})$ by computing its gradient (constrained with the proper Lagrange multiplier) and setting it to 0;

3. Eventually, we find that the (non-trivial) solution to this problem corresponds exactly to $\mathbf{e}$ being an eigenvector of the covariance matrix.

3 Largest Eigenvalue: Principal Component

In the section above, we have formally proved that eigenvector points towards the direction which maximizes the variance of data. In this section, we aim to prove why the principal component corresponds to the eigenvector with the largest eigenvalue[1].

Let's go back to our initial definition of variance along the (eigen)vector $\mathbf{e}$, which is defined as follows:

$$V(\mathbf{e}) = \frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)^2$$

We rewrite the equation above by unrolling the sum of squares as follows:

$$V(\mathbf{e}) = \frac{1}{n}\sum_{i=1}^{n}\Big(\sum_{j=1}^{d} x_{i,j}e_j\Big)\Big(\sum_{k=1}^{d} x_{i,k}e_k\Big)$$

The order of summations can be (carefully) moved outside:

$$V(\mathbf{e}) = \sum_{k=1}^{d}\sum_{j=1}^{d}\left(\frac{1}{n}\sum_{i=1}^{n} x_{i,k}x_{i,j}\right)e_je_k$$

Again, the most internal summation is just the covariance between random variables X_k and X_j (associated with the k-th and j-th dimension, respectively):

$$V(\mathbf{e}) = \sum_{k=1}^{d}\sum_{j=1}^{d}\left(\underbrace{\frac{1}{n}\sum_{i=1}^{n} x_{i,k}x_{i,j}}_{\text{Cov}(X_k,X_j)}\right)e_je_k$$

Therefore:

$$V(\mathbf{e}) = \sum_{k=1}^{d}\left(\sum_{j=1}^{d}\text{Cov}(X_k,X_j)e_j\right)e_k$$

Again, the most internal summation corresponds to the dot product between the k-th row of the covariance matrix K and the vector $\mathbf{e}$. Since we now know that $\mathbf{e}$ is an eigenvector, by definition of it we also know that for all $k = 1,\ldots,d$ it must hold the following:

$$\sum_{j=1}^{d}\text{Cov}(X_k,X_j)e_j = \lambda e_k$$

Therefore:

$$V(\mathbf{e}) = \sum_{k=1}^{d}\Big(\lambda e_k\Big)e_k = \lambda\sum_{k=1}^{d}e_k^2 = \lambda||\mathbf{e}||$$

[1] Remember that, in general, there are d distinct eigenvalues for a d-by-d covariance matrix.

Since $||\mathbf{e}|| = 1$, we obtain:

$$V(\mathbf{e}) = \lambda$$

In other words, the variance along the unit-length eigenvector $\mathbf{e}$ is exactly equal to its corresponding eigenvalue λ.

As such, to find the $k \ll d$ principal components we will just need to find up to d eigenvalues $\lambda_1, \ldots, \lambda_d$ and their associated eigenvectors $\mathbf{e}_1, \ldots, \mathbf{e}_d$. Then, we sort eigenvectors from the largest to the smallest eigenvalue and we pick the first k of them. Indeed, the eigenvector with the largest eigenvalue will correspond to the direction with the highest variance, the eigenvector with the second largest eigenvalue will correspond to the direction with the second highest variance, and so on and so forth.