Published in Towards Data Science

Akash Dubey Follow

Dec 21, 2018 · 6 min read ·

Save

The Mathematics Behind Principal Component Analysis

From raw data to principal components

Photo by Tim Johnson on Unsplash

Introduction

The central idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of a large number of interrelated variables while retaining as much as possible of the variation present in the data set. This is achieved by transforming to a new set of variables, the *principal components (PCs)*, which are uncorrelated, and which are ordered so that the first few retain most of the variation present in all of the original variables.

Mathematics Behind PCA

PCA can be thought of as an unsupervised learning problem. The whole process of obtaining principle components from a raw dataset can be simplified in six parts :

- Take the whole dataset consisting of *d+1 dimensions* and ignore the labels such that our new dataset becomes *d dimensional.*
- Compute the *mean* for every dimension of the whole dataset.
- Compute the *covariance matrix* of the whole dataset.
- Compute *eigenvectors* and the corresponding *eigenvalues*.
- Sort the eigenvectors by decreasing eigenvalues and choose k eigenvectors with the largest eigenvalues to form a $d \times k$ *dimensional* matrix **W.**
- Use this $d \times k$ *eigenvector matrix* to transform the samples onto the new subspace.

So, let's unfurl the maths behind each of this one by one.

1. **Take the whole dataset consisting of *d+1 dimensions* and ignore the labels such that our new dataset becomes *d dimensional.***

Let's say we have a dataset which is *d+1 dimensional*. Where d could be thought as *X_train* and 1 could be thought as y_train *(labels)* in modern machine learning paradigm. So, *X_train + y_train* makes up our complete train dataset.

So, after we drop the labels we are left with *d dimensional* dataset and this would be the dataset we will use to find the principal components. Also, let's assume we are left with a three-dimensional dataset after ignoring the labels i.e d = 3.

we will assume that the samples stem from two different classes, where one-half samples of our dataset are labeled class 1 and the other half class 2.

Let our data matrix **X** be the score of three students :

Student	Math	English	Art
1	90	60	90
2	90	90	30
3	60	60	60
4	60	60	90
5	30	30	30

2. Compute the mean of every dimension of the whole dataset.

The data from the above table can be represented in matrix **A,** where each column in the matrix shows scores on a test and each row shows the score of a student.

$$\mathbf{A} = \begin{bmatrix} 90 & 60 & 90 \\ 90 & 90 & 30 \\ 60 & 60 & 60 \\ 60 & 60 & 90 \\ 30 & 30 & 30 \end{bmatrix}$$

Matrix A

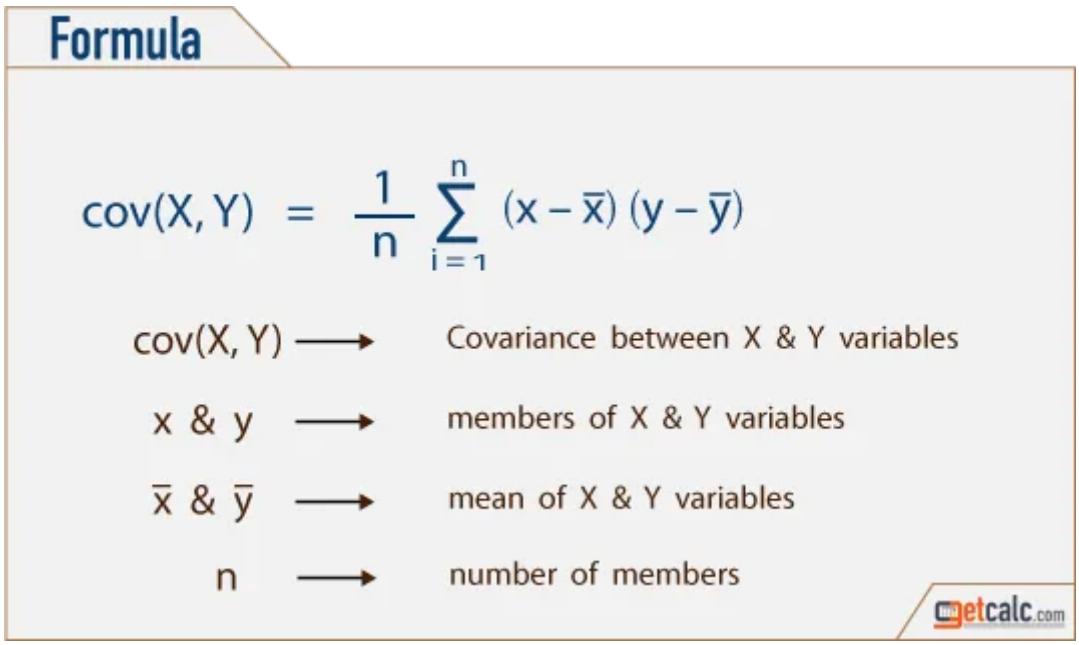
So, The mean of matrix **A** would be

$$\bar{\mathbf{A}} = [\, 66 \quad 60 \quad 60 \,]$$

Mean of Matrix A

3. Compute the *covariance matrix* of the whole dataset (sometimes also called as the variance-covariance matrix)

So, we can compute the covariance of two variables **X** and **Y** using the following formula :



https://getcalc.com/statistics-covariance-calculator.htm

Using the above formula, we can find the covariance matrix of **A.** Also, the result would be a *square matrix of d ×d dimensions.*

Let's rewrite our original matrix like this

	Math	*English*	*Arts*
1	90	60	90
2	90	90	30
3	60	60	60
4	60	60	90
5	30	30	30

Matrix A

Its c*ovariance matrix* would be

	Math	*English*	*Art*
Math	504	360	180
English	360	360	0
Art	180	0	720

Covariance Matrix of A

Few points that can be noted here is :

- Shown in *Blue* along the diagonal, we see the variance of scores for each test. The art test has the biggest variance (720); and the English test, the smallest (360). So we can say that art test scores have more variability than English test scores.
- The covariance is displayed in black in the off-diagonal elements of the matrix **A**

a) The covariance between math and English is positive (360), and the covariance between math and art is positive (180). This means the scores tend to covary in a positive way. As scores on math go up, scores on art and English also tend to go up; and vice versa.

b) The covariance between English and art, however, is zero. This means there tends to be no predictable relationship between the movement of English and art scores.

4. Compute Eigenvectors and corresponding Eigenvalues

> *Intuitively, an eigenvector is a vector whose direction remains unchanged when a linear transformation is applied to it.*

Now, we can easily compute eigenvalue and eigenvectors from the covariance matrix that we have above.

> *Let* **A** *be a square matrix,* ν *a vector and* λ *a scalar that satisfies* $\mathbf{A}\nu = \lambda\nu$*, then* λ *is called eigenvalue associated with eigenvector* ν *of* **A**.

The eigenvalues of A are roots of the characteristic equation

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

Calculating $det(A-\lambda I)$ first, I is an identity matrix :

$$\det\left(\begin{pmatrix} 504 & 360 & 180 \\ 360 & 360 & 0 \\ 180 & 0 & 720 \end{pmatrix} - \lambda\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\right)$$

Simplifying the matrix first, we can calculate the determinant later,

$$\begin{pmatrix} 504 & 360 & 180 \\ 360 & 360 & 0 \\ 180 & 0 & 720 \end{pmatrix} - \begin{pmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{pmatrix}$$

$$\begin{pmatrix} 504-\lambda & 360 & 180 \\ 360 & 360-\lambda & 0 \\ 180 & 0 & 720-\lambda \end{pmatrix}$$

Now that we have our simplified matrix, we can find the determinant of the same :

$$\det \begin{pmatrix} 504-\lambda & 360 & 180 \\ 360 & 360-\lambda & 0 \\ 180 & 0 & 720-\lambda \end{pmatrix}$$

$$-\lambda^3 + 1584\lambda^2 - 641520\lambda + 25660800$$

We now have the equation and we need to solve for λ, so as to get the *eigenvalue of the matrix*. So, equating the above equation to zero :

$$-\lambda^3 + 1584\lambda^2 - 641520\lambda + 25660800 = 0$$

After solving this equation for the value of λ, we get the following value

$$\lambda \approx 44.81966\ldots, \lambda \approx 629.11039\ldots, \lambda \approx 910.06995\ldots$$

Eigenvalues

Now, we can calculate the eigenvectors corresponding to the above eigenvalues. I would not show how to calculate eigenvector here, visit this link to understand how to calculate eigenvectors.

So, after solving for *eigenvectors* we would get the following solution for the corresponding *eigenvalues*

$$\begin{pmatrix} -3.75100\ldots \\ 4.28441\ldots \\ 1 \end{pmatrix}, \begin{pmatrix} -0.50494\ldots \\ -0.67548\ldots \\ 1 \end{pmatrix}, \begin{pmatrix} 1.05594\ldots \\ 0.69108\ldots \\ 1 \end{pmatrix}$$

5. Sort the eigenvectors by decreasing eigenvalues and choose k eigenvectors with the largest eigenvalues to form a *d* × *k dimensional* matrix W.

We started with the goal to reduce the dimensionality of our feature space, i.e., projecting the feature space via PCA onto a smaller subspace, where the eigenvectors will form the axes of this new feature subspace. However, the eigenvectors only define the directions of the new axis, since they have all the same unit length 1.

So, in order to decide which eigenvector(s) we want to drop for our lower-dimensional subspace, we have to take a look at the corresponding eigenvalues of the eigenvectors. Roughly speaking, the eigenvectors with the lowest eigenvalues bear the least information about the distribution of the data, and those are the ones we want to drop.

The common approach is to rank the eigenvectors from highest to lowest corresponding eigenvalue and choose the top *k* eigenvectors.

So, after sorting the eigenvalues in decreasing order, we have

$$\begin{pmatrix} 910.06995 \\ 629.11039 \\ 44.81966 \end{pmatrix}$$

For our simple example, where we are reducing a 3-dimensional feature space to a 2-dimensional feature subspace, we are combining the two eigenvectors with the highest eigenvalues to construct our *d*×*k* dimensional eigenvector matrix **W.**

So, *eigenvectors* corresponding to two maximum eigenvalues are :

$$\mathbf{W} = \begin{bmatrix} 1.05594 & -0.50494 \\ 0.69108 & -0.67548 \\ 1 & 1 \end{bmatrix}$$

6. Transform the samples onto the new subspace

In the last step, we use the 3x2 dimensional matrix W that we just computed to transform our samples onto the new subspace via the equation $y = W' \times x$ where W' is the *transpose* of the matrix W.

So lastly, we have computed our two principal components and projected the data points onto the new subspace.

Credits and Sources :

1. Sebastian Raschka Blog
2. Stattrek Matrix Algebra

Machine Learning | Data Science | Artificial Intelligence | Dimensionality Reduction | Mathematics