## CHAPTER 26

# Principal Component Analysis (PCA)

Principal component analysis (PCA) is an essential algorithm in machine learning. It is a mathematical method for evaluating the principal components of a dataset. The principal components are a set of vectors in high-dimensional space that capture the variance (i.e., spread) or variability of the feature space.

The goal of computing principal components is to find a low-dimensional feature sub-space that captures as much information as possible from the original higher-dimensional features of the dataset.

PCA is particularly useful for simplifying data visualization of high-dimensional features by reducing the dimensions of the dataset to a lower sub-space. For example, since we can easily visualize relationships on a 2-D plane using scatter diagrams, it will be useful to condense an n-dimensional space into two dimensions that retain as much information as possible in the n-dimensional dataset. This technique is popularly called dimensionality reduction.

## How Are Principal Components Computed

The mathematical details for computing principal components are somewhat involved. This section will instead provide a conceptual but solid overview of this process.

The first step is to find the covariance matrix of the dataset. The covariance matrix captures the linear relationship between variables or features in the dataset. In a covariance matrix, an increasingly positive number represents a growing relationship, while the converse is represented by an increasingly negative number. Numbers around zero indicate a non-linear relationship between the variables. The covariance matrix is a square matrix (that means it has the same rows and columns). Hence, given a dataset with $m$ rows and $p$ columns, the covariance matrix will be a $m \times p$ matrix.

E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*,
https://doi.org/10.1007/978-1-4842-4470-8_26

The next step is to find the eigenvectors of the covariance matrix dataset. In linear algebra theory, eigenvectors are non-zero vectors that merely stretch by a scalar factor, but do not change direction when acted upon by a linear transformation. We find the eigenvectors using a linear algebra technique called the singular value decomposition or SVD for short (see Figure 26-1). This advanced mathematical concept is beyond the scope of this book.

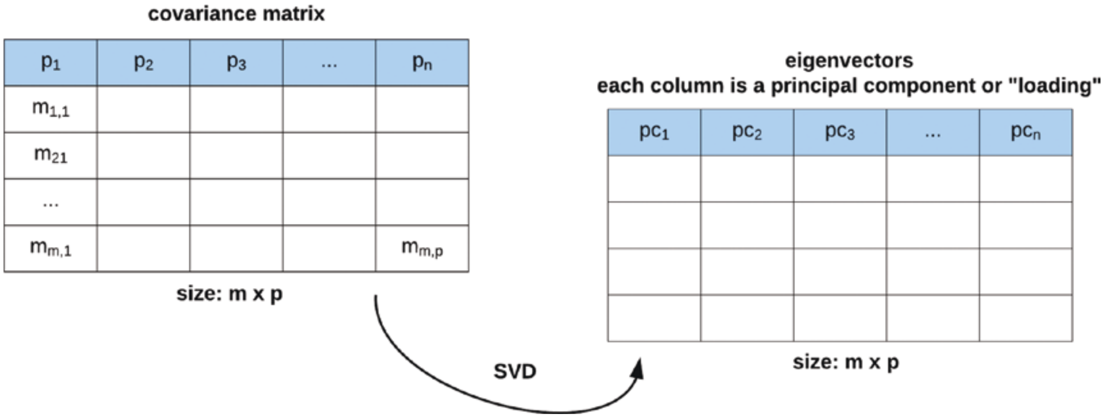

***Figure 26-1.*** *Decompose the covariance matrix using SVD to get the eigenvector matrix*

The critical point to note at this junction is that the SVD also outputs a square matrix ($p \times p$), and each column of the matrix is an eigenvector of the original dataset. This output is the same across different software packages that compute the eigenvectors because the covariance matrix satisfies a mathematical property of being symmetric and positive semi-definite (the non-math inclined can conveniently ignore this point). We have as many eigenvectors as they are attributes or features in the dataset.

Without delving into mathematical theory, we can conclude that the eigenvectors are the principal components or loadings of the feature space. Again remember that the principal components capture the most significant variance in the dataset by projecting the data onto a vector called the first principal component. Other principal components are perpendicular to each other and capture the variance not explained by the first principal component. The principal components are arranged in order of importance in the eigenvector matrix, with the first principal component in the first column, the second principal component in the second column, and so on.

## Dimensionality Reduction with PCA

To reduce the dimensions of the original dataset using PCA, we multiply the desired number of components or loadings from the eigenvector matrix, $A$, by the design matrix $X$. Suppose the design matrix (or the original dataset) has $m$ rows (or observations) and $p$ columns (or features), if we want to reduce the dimensions of the original dataset to two dimensions, we will multiply the original dataset $X$ by the first two columns of the eigenvector matrix, $A_{reduced}$. The result will be a reduced matrix of $m$ rows and 2 columns.

If $X$ is a $m \times p$ matrix and $A_{reduced}$ is a $p \times 2$ matrix,

$$T_{reduced} = X_{m \times p} \times A_{p \times 2}$$

Observe that the result $T_{reduced}$ is a $m \times 2$ matrix. Hence, $T$ is a 2-D representation of the original dataset $X$ as shown in Figure 26-2.

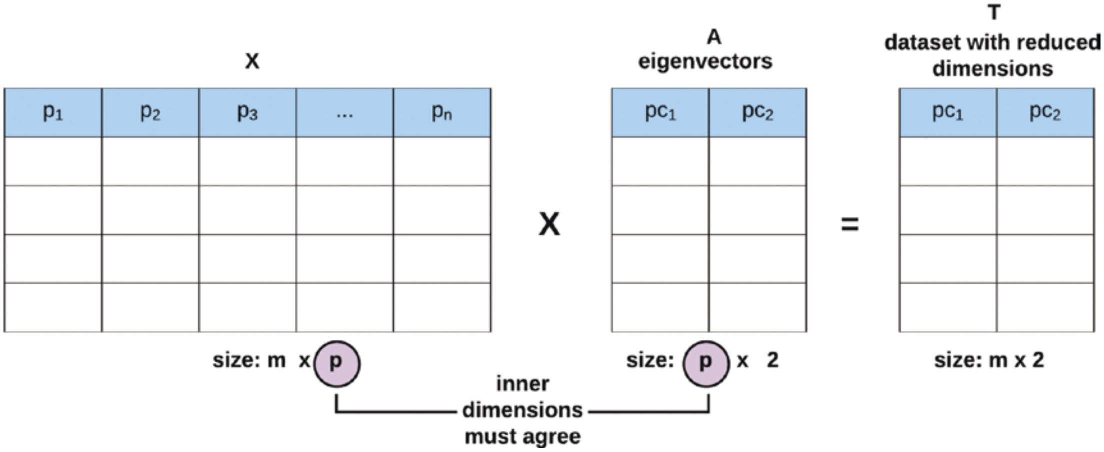

*Figure 26-2. Reducing the dimension of the original dataset*

In plotting the reduced dataset, the principal components are ranked in order of importance with the first principal component more prominent than the second and so on. Figure 26-3 illustrates a plot of the first two principal components.