Principal Component Analysis (PCA)

Need For Principal Component Analysis (PCA)

Machine Learning in general works wonders when the dataset provided for training the machine is large and concise. Usually having a good amount of data lets us build a better predictive model since we have more data to train the machine with. However, using a large data set has its own pitfalls. The biggest pitfall is the curse of dimensionality.

It turns out that in large dimensional datasets, there might be lots of inconsistencies in the features or lots of redundant features in the dataset, which will only increase the computation time and make data processing and EDA more convoluted.

To get rid of the curse of dimensionality, a process called dimensionality reduction was introduced. Dimensionality reduction techniques can be used to filter only a limited number of significant features needed for training and this is where PCA comes in.

What Is Principal Component Analysis (PCA)?

Principal components analysis (PCA) is a dimensionality reduction technique that enables you to identify correlations and patterns in a data set so that it can be transformed into a data set of significantly lower dimension without loss of any important information.

The main idea behind PCA is to figure out patterns and correlations among various features in the data set. On finding a strong correlation between different variables, a final decision is made about reducing the dimensions of the data in such a way that the significant data is still retained.

Such a process is very essential in solving complex data-driven problems that involve the use of high-dimensional data sets. PCA can be achieved via a series of steps. Let’s discuss the whole end-to-end process.

Step By Step Computation of PCA

The below steps need to be followed to perform dimensionality reduction using PCA:

1. Standardization of the data
2. Computing the covariance matrix
3. Calculating the eigenvectors and eigen values
4. Computing the Principal Components

5. Reducing the dimensions of the data set

Let’s discuss each of the steps in detail:

Step 1: Standardization of the data

If you’re familiar with data analysis and processing, you know that missing out on standardization will probably result in a biased outcome. Standardization is all about scaling your data in such a way that all the variables and their values lie within a similar range.

Consider an example, let’s say that we have 2 variables in our data set, one has values ranging between 10 to100 and the other has values between 1000-5000. In such a scenario, it is obvious that the output calculated by using these predictor variables is going to be biased since the variable with a larger range will have a more obvious impact on the outcome.

Therefore, standardizing the data into a comparable range is very important. Standardization is carried out by subtracting each value in the data from the mean and dividing it by the overall deviation in the data set.

It can be calculated like so:

$$Z = \frac{\textit{Variable value - mean}}{\textit{Standard deviation}}$$

Post this step, all the variables in the data are scaled across a standard and comparable scale.

Step 2: Computing the covariance matrix

As mentioned earlier, PCA helps to identify the correlation and dependencies among the features in a data set. A covariance matrix expresses the correlation between the different variables in the data set. It is essential to identify heavily dependent variables because they contain biased and redundant information which reduces the overall performance of the model.

Mathematically, a covariance matrix is a p × p matrix, where p represents the dimensions of the data set. Each entry in the matrix represents the covariance of the corresponding variables.

Consider a case where we have a 2-Dimensional data set with variables a and b, the covariance matrix is a 2×2 matrix as shown below:

$$\begin{bmatrix} Cov(a, a) & Cov(a, b) \\ Cov(b, a) & Cov(b, b) \end{bmatrix}$$

In the above matrix:

- Cov(a, a) represents the covariance of a variable with itself, which is nothing but the variance of the variable ‘a’
- Cov(a, b) represents the covariance of the variable ‘a’ with respect to the variable ‘b’. And since covariance is commutative, Cov(a, b) = Cov(b, a)

Here are the key takeaways from the covariance matrix:

- The covariance value denotes how co-dependent two variables are with respect to each other
- If the covariance value is negative, it denotes the respective variables are indirectly proportional to each other
- A positive covariance denotes that the respective variables are directly proportional to each other

Simple math, isn’t it? Now let’s move on and look at the next step in PCA.

Step 3: Calculating the Eigenvectors and Eigenvalues

Eigenvectors and eigenvalues are the mathematical constructs that must be computed from the covariance matrix in order to determine the principal components of the data set.

But first, let’s understand more about principal components

What are Principal Components?

Simply put, principal components are the new set of variables that are obtained from the initial set of variables. The principal components are computed in such a manner that newly obtained variables are highly significant and independent of each other. The principal components compress and possess most of the useful information that was scattered among the initial variables.

If your data set is of 5 dimensions, then 5 principal components are computed, such that, the first principal component stores the maximum possible information and the second one stores the remaining maximum info and so on, you get the idea.

Now, where do Eigenvectors fall into this whole process?

Assuming that you all have a basic understanding of Eigenvectors and eigenvalues, we know that these two algebraic formulations are always computed as a pair, i.e, for every eigenvector there is an eigenvalue. The dimensions in the data determine the number of eigenvectors that you need to calculate.

Consider a 2-Dimensional data set, for which 2 eigenvectors (and their respective eigenvalues) are computed. The idea behind eigenvectors is to use the Covariance matrix to understand where in the data there is the most amount of variance. Since more variance in the data denotes more information about the data, eigenvectors are used to identify and compute Principal Components.

Eigenvalues, on the other hand, simply denote the scalars of the respective eigenvectors. Therefore, eigenvectors and eigenvalues will compute the Principal Components of the data set.

Step 4: Computing the Principal Components

Once we have computed the Eigenvectors and eigenvalues, all we have to do is order them in the descending order, where the eigenvector with the highest eigenvalue is the most significant and thus forms the first principal component. The principal components of lesser significances can thus be removed in order to reduce the dimensions of the data.

The final step in computing the Principal Components is to form a matrix known as the feature matrix that contains all the significant data variables that possess maximum information about the data.

Step 5: Reducing the dimensions of the data set

The last step in performing PCA is to re-arrange the original data with the final principal components which represent the maximum and the most significant information of the data set. In order to replace the original data axis with the newly formed Principal Components, you simply multiply the transpose of the original data set by the transpose of the obtained feature vector.

So that was the theory behind the entire PCA process. It’s time to get your hands dirty and perform all these steps by using a real data set.