CHAPTER 3

Using Principal Components

OBJECTIVE

After learning how to compute principal components in Chapter 2, this chapter will guide you on how to utilize them effectively. You will discover how to determine the minimum number of principal components needed to capture the key dimensions for analyzing your data. Additionally, you will gain insights into what these principal components represent in relation to the original variables and how they can be applied in further analyses of your dataset.

Contents

3.1 Introduction

After performing a Principal Component Analysis (PCA), the next step is to interpret the components in relation to your original data before deciding how to use them. This chapter addresses two key objectives: first, interpreting the principal components, and second, applying them to a specific problem.

The main questions when interpreting principal components are:

- How many principal components should be retained for further analysis?
- Which original variables are the key contributors to each composite score variable?

A key goal of PCA is dimensionality reduction. This chapter provides guidelines for determining whether your data should be represented in two, three, or more dimensions.

PCA has many practical applications. In this chapter, you will see how it can be used to identify the most relevant variables and detect outliers in multivariate datasets. In Part II of this book, we will explore how PCA can improve the efficiency of cluster analysis procedures.

3.2 Interpreting Principal Components

This section demonstrates how to determine the optimal number of principal components and how to identify key contributing variables using the *Iris20* dataset, previously discussed in Chapter 2.

3.2.1 *Optimal Number of Principal Components*

To determine the number of principal components to retain, the eigenvalues (ranked from highest to lowest) must be carefully examined. For the *Iris20* dataset, the eigenvalues are presented in Table 3.1, which will be central to determining the optimal number of principal components.

Table 3.1: Percent of variance explained by the principal components

Component#	Eigenvalue	Variance(%)	Cumul. Var.(%)
1	2.8800	72.0%	72.0%
2	0.9379	23.5%	95.5%
3	0.1707	4.3%	99.7%
4	0.0114	0.3%	100.0%

Each eigenvalue represents the variance of the associated composite score variable. To minimize the number of dimensions, retain the smallest number of components that explains an acceptable proportion of the total variance. The challenge is balancing the number of dimensions while capturing most of the dataset's variability.

Table 3.1 shows the eigenvalues[1] and their associated variance percentages. The first eigenvalue ($\lambda_1 = 2.88$) explains 72% of the total variance, and the second ($\lambda_2 = 0.9379$) explains 23.5%, meaning that together, the first two principal components account for 95.5% of the total variance.

The Eigenvalues are depicted in Figure 3.1, where you can see the percent variance explained going down as the principal component number goes up. This specific figure is known as the *Scree Graph* or the *Scree Plot*.

Several methods exist for selecting the number of principal components, but many are heuristic or subjective. Ferré (1995) acknowledges the absence of an ideal solution, while Jolliffe (2002) suggests that statistically-based procedures offer no clear advantage over simpler ones. Readers interested in comparing methods can refer to Zwick and Velicer (1986) or Jackson (1993) . In this section, we review three widely used methods.

Kaiser's rule, proposed by Kaiser (1960) recommends retaining principal components with eigenvalues greater than 1, which is the baseline variance of the original standardized variables. Based on this criterion, the first two components would be retained, accounting for almost 95.5% of the total variance.

Jolliffe (2002) proposes a modification, suggesting that components with eigenvalues larger than 0.7 should be retained, which would include all four components in this case. Reducing Kaiser's

[1] In linear algebra literature, an Eigenvalue value is often labeled as λ, which is the Greek character "lambda."

threshold for component retention to 0.7 supposedly accounts for statistical errors that may understate the actual value of those eigenvalues that exceed 1.

Cattell's scree plot criterion Cattell (1966) suggests looking for the point in the scree plot where the curve flattens, indicating diminishing returns from additional components. The "elbow" in Figure 3.1 occurs after the second component, supporting the retention of two components.

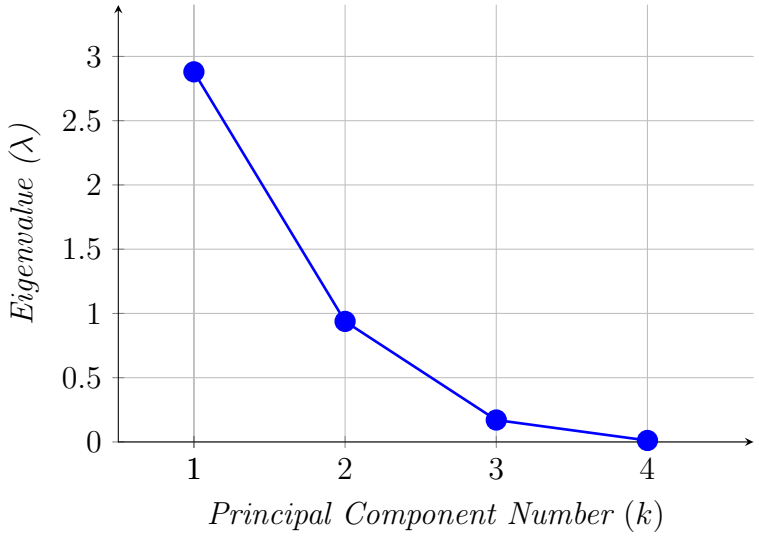

Figure 3.1: Scree graph of the Principal Component Analysis performed on the *Iris20* dataset of chapter 2

As Jolliffe (2002) stated, "The first point on the straight line is then taken to be the last factor/component to be retained." In other words, "... The specific PC number defining an 'elbow' in the graph is taken to be the number of components to be retained." The data point associated with PC#3 in Figure 3.1 could be seen

as the elbow. However, there is no clear-cut elbow in this scree plot, with the last line segment, linking the two lowest data points, being almost horizontal. One may argue that the elbow starts from PC#2, which would lead to retaining the first two components.

In my opinion, there is no need for an objective, general-purpose method to determine the optimal number of principal components. Ultimately, the number of components to retain depends on the goal of the analysis. Some researchers may focus solely on the first principal component to extract a single composite score that summarizes the entire dataset. Others may be interested in detecting outliers, in which case retaining two principal components may be more appropriate (see Section 3.3.2). If principal component scores are to be used as independent variables in a regression model, one might retain a sufficiently large number of principal components to ensure that all initial variables are adequately represented. Further discussion on these topics can be found in Section 3.2.2.

In section 3.2.2, you will see how to interpret principal components with respect to the original variables using their loadings.

3.2.2 *Principal Component Loadings*

Comparing composite score variables to the original attributes is good practice as it helps develop a strong understanding of their meaning. By doing so, you can identify the original variables that contribute most to each composite score variable. The first step towards this goal is to compute the principal component loadings. *A component loading is the correlation coefficient*

between an original attribute and a composite score variable.

Table 3.2 shows the component loadings for all four principal components. You will notice that three out of four original variables have high positive coefficients with the first principal component. Only *Sepal.Width* has a moderate negative correlation with it. However, *Sepal.Width* has a high positive correlation with the second principal component. The correlations of the original attributes with the third and fourth components are negligible. Therefore, you can use only the first two components in subsequent analyses while accounting for the contributions of all four initial attributes. Moreover, the first two components explain 95.5% of the total variance, as shown in Table 3.1. The remaining 4.5% of unexplained variance may well be statistical noise that can be safely ignored.

Table 3.2: Principal component loadings for the *Iris20* dataset

	Principal Component #			
Variable	1	2	3	4
Sepal.Length	0.8489	0.4388	0.2937	0.0211
Sepal.Width	-0.4977	0.8579	-0.1263	-0.0203
Petal.Length	0.9934	-0.0333	-0.0704	-0.0847
Petal.Width	0.9617	0.0910	-0.2520	0.0584

How are Table 3.2 loadings calculated?

1. The general approach for calculating PC loadings is by using

the following expression:

$$\text{Loadings} = \text{Eigenvectors} \times \frac{\sqrt{\text{Eigenvalues}}}{\text{Stdev}}, \qquad (3.2.1)$$

where "Stdev" is the standard deviation of the original variable. For the *Iris20* dataset, there are 4 eigenvectors of 4 elements each, 4 eigenvalues, and a standard deviation for each of the 4 variables.

2. If your input data is centered and standardized[2], then component loadings can be calculated by multiplying each eigenvector by the square root of the associated eigenvalue. That is expression 3.2.1 where Stdev = 1.

3. If your data is non standardized, then you need to first obtain the standard deviation of each input variable before apply expression 3.2.1.

4. Alternatively, you may proceed with the direct computation of these correlations, 2 original attribute and 1 composite score variable at a time.

To illustrate how the values in Table 3.2 are calculated, consider the loadings for Principal Component `#1`, found in column 2 of the table. The PC`#1` loadings are 0.8489, -0.4977, 0.9934, and 0.9617. These values are obtained by multiplying each element of PC`#1` in Table 3.3 by the square root of 2.8800, the first eigenvalue in Table 3.1. For example, $0.8489 = 0.5002 \times \sqrt{2.8800}$.

[2]The "Multivariate.xlsm" Excel template gives you this option, as well as 2 other options.

The loadings for PC#2 are obtained by multiplying each element of the PC#2 eigenvector in Table 3.3 by the square root of 0.9379, the second eigenvalue in Table 3.1. You can follow a similar process for the remaining loadings.

Table 3.3: Principal components / eigenvectors produced by the PCA of the *Iris20* dataset (see Figure 2.11 of chapter 2

	Principal Component #			
Variable	1	2	3	4
Sepal.Length	0.5002	0.4531	0.7110	0.1971
Sepal.Width	-0.2933	0.8858	-0.3056	-0.1895
Petal.Length	0.5853	-0.0344	-0.1703	-0.7919
Petal.Width	0.5667	0.0940	-0.6099	0.5459

In PCA literature, the terms eigenvectors and loadings are often used interchangeably. Should they be? No, they shouldn't. Loadings are correlation coefficients and are more easily interpretable. However, since loadings are proportional to eigenvectors, using either one to identify the most important contributing attributes to the composite scores is correct.

3.3 Applications of Principal Components

To illustrate a few applications of Principal Component Analysis (PCA), I will use a dataset containing 1979 employment data of 26 European countries. This dataset, referred to as *Euro1979*, contains the percentage of the labor force in different industries by