DIGITAL CULTURE
ITMO UNIVERSITY

Principal Component Analysis

High School of Digital Culture
ITMO University
dc@itmo.ru

Contents

1 Welcome Remarks

Hello everyone! We are excited to welcome you to Advanced Machine Learning. In this course, we will consider a powerful classification technique called support vector machine (or SVM) and review the elements of factor analysis in terms of principal components. You will learn how to calculate the information entropy and how to construct decision trees based on the obtained results. The course will also explain how to combine multiple models into one mega model using ensembles. And the cherry on top is reinforcement learning that we will study by the end the course. Well, let's get started.

No wonder that one may feel lost while working with massive amounts of data of unknown nature, the understanding of which is similar to the story of the blind men and the elephant [1]. Well, then how to reveal hidden patterns and relationships? How to group the objects? The answer to these questions (or at least a hint) is often based on the experience or eyeballing. In fact, numeric data can be visualized for analyzing (we will not bring up the subject of non-numeric data in this module). But what if the data dimensionality is large? What can we do? On the one side, we can disregard some features and do not consider them. It is one of possible approaches. However, we cannot be sure that we are not missing something important. On the other side, we can use the available features to synthesize new ones of a less number. Apparently, we will lose some of the information about the objects. Therefore, the new features should convey as much information as possible. But how to achieve this? This module will answer all of these questions.

2 Heuristic Arguments

The goal of the Principal Component Analysis (PCA) is to reduce the dimensions of input data while minimizing the information loss by way of a new coordinate system. In other words, we will try to obtain as much informative data as possible by introducing new predictors for the data. We will not disregard a portion of the data but compose the features making them less numerous. We'll select such a new coordinate system so that the data mostly differs along the first axis. Next, from the remaining coordinate axes, we'll choose the one, in which the data mostly differs along the second axis, and so on. To understand the idea, look at fig. 1.

The blue points that form an ellipse are data (each object has a pair of features). It seems that the change in data is the most significant along the straight line PC_1. Why do we think so? Well, the spread from the mean (from

[1] https://www.datasciencecentral.com/profiles/blogs/the-story-of-big-data-data-science-amp-data-mining

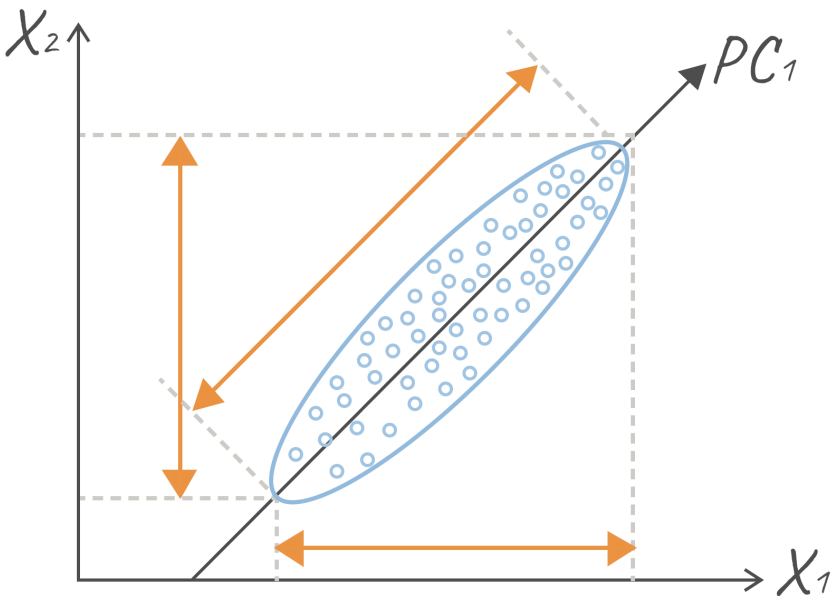

Figure 1: Changing the coordinate system.

the ellipse center) along the straight line is the largest. So, we assume that the more the sample variance along the axis, the better (or more) the data changes. The idea of constructing a new coordinate system isn't new. It echoes the idea and experience in analytic geometry. You choose a convenient coordinate system, and the problem is solved in no time. Let's consider another example. Let's consider another example.

Assume a company has five car sales managers. The head of the sales department wants to divide the bonus payments between employees based on the number of cars each of them sold. Two predictors to consider are the number of premium cars and the number of economy cars. The input data is given in the table:

Employee (x_i)	**Premium** (X_1), **pcs**	**Economy** (X_2), **pcs**
1	9	19
2	6	22
3	11	27
4	12	25
5	7	22

The problem of bonus payments is easier to solve when each employee is associated with the only one number, and all employees differ from each other. But how do we make this possible? For example, we can use some coefficients φ_1 and φ_2 to calculate the overall rating z_i of the employee under the number i using the relation

$$z_i = \varphi_1 x_{i1} + \varphi_2 x_{i2},\ i = \{1, 2, \ldots, 5\},$$

where x_{ij} is the number of cars sold by a particular employee, the index i is the employee's number (from 1 to 5), the index j shows the type of the car sold (1 is for premium, 2 is for economy). For example, $x_{52} = 22$ is the number of economy cars sold by the manager No. 5. After the transformation, each employee under the number i will be characterized only by the number z_i. Its value allows us to solve the problem of bonus payments based on the performance criteria. All we've got left is to choose the coefficients φ_1 and φ_2. Hence, we will need to introduce the terms and formulate the problem mathematically.

3 Principal Component Analysis

3.1 An Example of Method Application

3.1.1 Constructing the First Principal Component

Let's consider a set of objects $x_1, x_2, \ldots, x_n$ with two features. Thus, each object x_i can be equated with the vector having two coordinates (x_{i1}, x_{i2}), that is,

$$x_i = (x_{i1}, x_{i2}),\quad i = \{1, 2, \ldots, n\}.$$

Based on this, a set of objects can be geometrically interpreted as a set of points in the plane (fig. 2).

First, let's perform data centering that is the subtraction from each coordinate of each object of the mean value of this coordinate for all objects. It doesn't affect the ideology we are discussing (because we only moved the center of the coordinate system), but it will ease further transformations. The result is shown in fig. 3.

After the collapse of the original feature space into one dimension, one coordinate z_i will correspond to the each object x_i:

$$x_i = (x_{i1}, x_{i2}) \longrightarrow z_i,\quad i \in \{1, 2, ..., n\}.$$

Then, new coordinates of all objects $x_1, x_2, ..., x_n$ can be written as a column vector

$$Z_1 = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix}.$$

If the object is characterized by one feature, it can be represented as a point on some straight line. Moreover, since the objects are centered, it is reasonable to

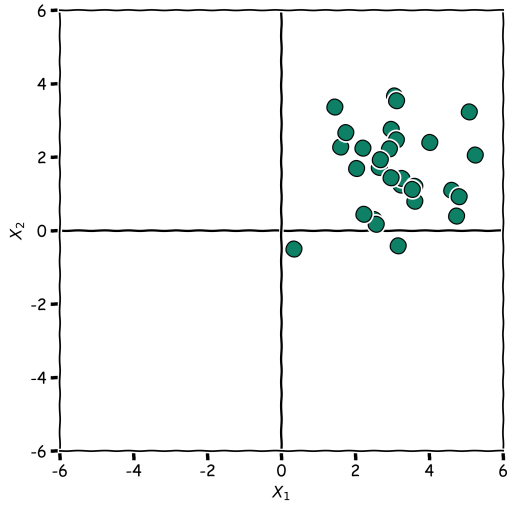

Figure 2: The input data before centering.

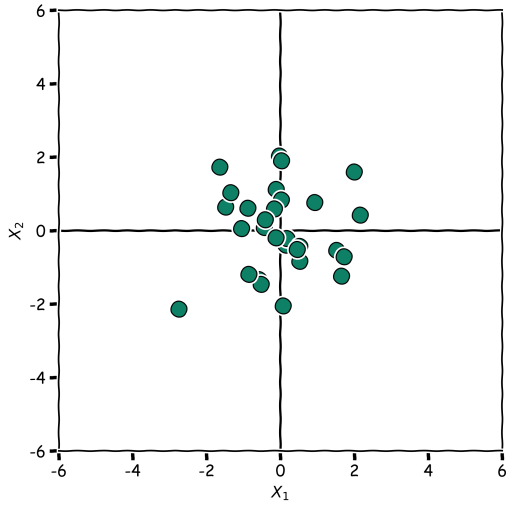

Figure 3: The input data after centering.

require the straight line to be passing through the origin, which ensures that new objects remain centered relative to the old coordinate system. We also want the straight line to be drawn in such a way that new coordinates differ from each other to the greatest extent. We will use sample variance as a dissimilarity measure:

$$S^2(Z_1) = \frac{1}{n}\sum_{i=1}^{n}\left(z_i - \overline{Z_1}\right)^2,$$

where $\overline{Z_1} = \frac{1}{n}\sum_{i=1}^{n} z_i$ is a sample mean.

It's reasonable to introduce the following definition.

Definition 3.1.1 *A straight line, which is passing through the origin and the projections of the centered coordinates of initial objects on which have the largest sample variance, is called the first principal component and designated by* PC_1.

Definition 3.1.2 *A unit directional vector* φ *of the first principal component (PC) is called the loading vector of the first PC.*

In the case of a two-dimensional space, the loading vector φ will have two coordinates φ_1 and φ_2.

Remark 3.1.1 *Note that each principal component always has exactly two loading vectors having different directions. In practice, it does not matter which one to choose.*

Let's construct the projection of one object (fig. 4); it is designated by the blue point. The straight line in the same figure is the first principal component PC_1

with the loading vector $\varphi = (\varphi_1, \varphi_2)$. The green point corresponds to the object x_i with the coordinates x_{i1} and x_{i2}. Therefore, it can be considered as a vector $x_i = (x_{i1}, x_{i2})$.

To find the new coordinate z_i on the straight line PC_1 with the basis vector φ (coinciding with the loading vector in this case), we can use the dot product defined as follows:

$$(x_i, \varphi) = |x_i| \cdot |\varphi| \cos \alpha,$$

where α is an angle between the vectors x_i and φ. Given that $|\varphi| = 1$, we obtain:

$$(x_i, \varphi) = |x_i| \cos \alpha = z_i,$$

The value of the dot product (x_i, φ) gives the value z_i of the projection x_i onto the direction φ, which will be the coordinate on the straight line PC_1 with the basis vector φ. On the other side, it can be easily proved that the dot product can be computed as the sum of the product of the corresponding coordinates in the rectangular (Cartesian) coordinate system.

$$z_i = (x_i, \varphi) = x_{i1}\varphi_1 + x_{i2}\varphi_2.$$

The same can be applied to all the objects of the considered dataset.

The results are shown in fig. 5. The initial object projections onto the straight line PC_1 are represented by blue points.

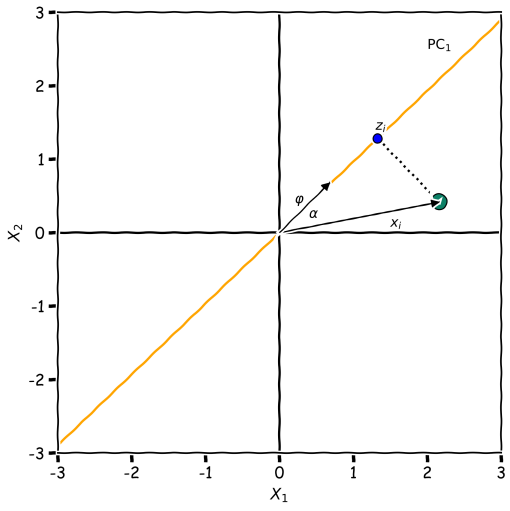

Figure 4: The object projection onto the first PC.

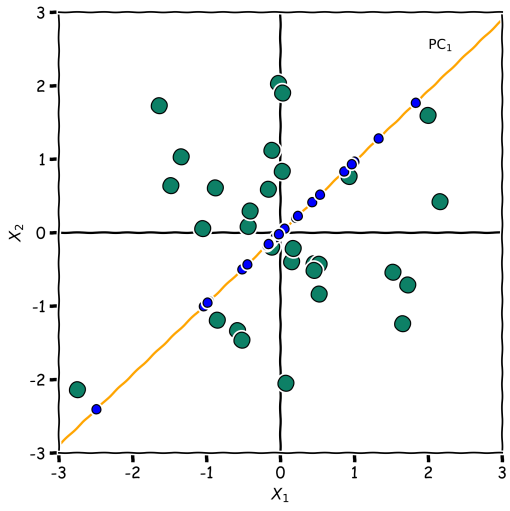

Figure 5: New object coordinates.

As you can see, constructing projections is easy. Thus, our task boils down to finding such loadings φ_1 and φ_2 (or finding such a way of drawing the straight line) so that the coordinates of the point projections on the straight line differ to the greatest extent (in other words, have the largest sample variance). We will discuss this matter a little bit later.

3.1.2 The Second and Subsequent Principal Components

Let's consider what happens from the geometrical standpoint when an initial feature space has the dimensionality $p \geq 2$, and it is necessary to construct not one, but k principal components, where $2 \leq k \leq p$. Note that the first principal component is always constructed like this. A straight line is drawn through the origin so that the coordinates of the point projections on the line have the largest sample variance.

The second and subsequent principal components are constructed so that they also pass through the origin (to preserve object centering) in the orthogonal direction to all the constructed principal components. The orthogonality condition of the principal components ensures that new object features are uncorrelated. Moreover, each subsequent PC is constructed so that the sample variance of the projection coordinates of the input data onto it is the largest. Fig. 6 and 7, respectively, show the input data, projection plane, and projection of initial objects onto the plane.

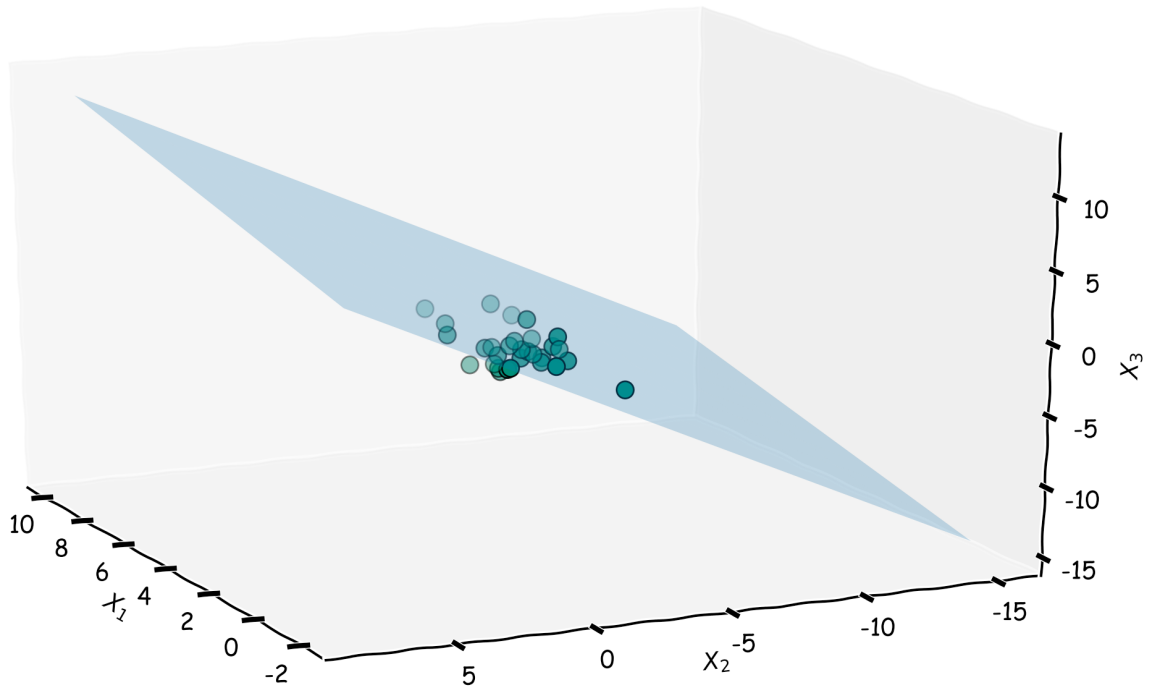

Figure 6: The input data and projection plane.

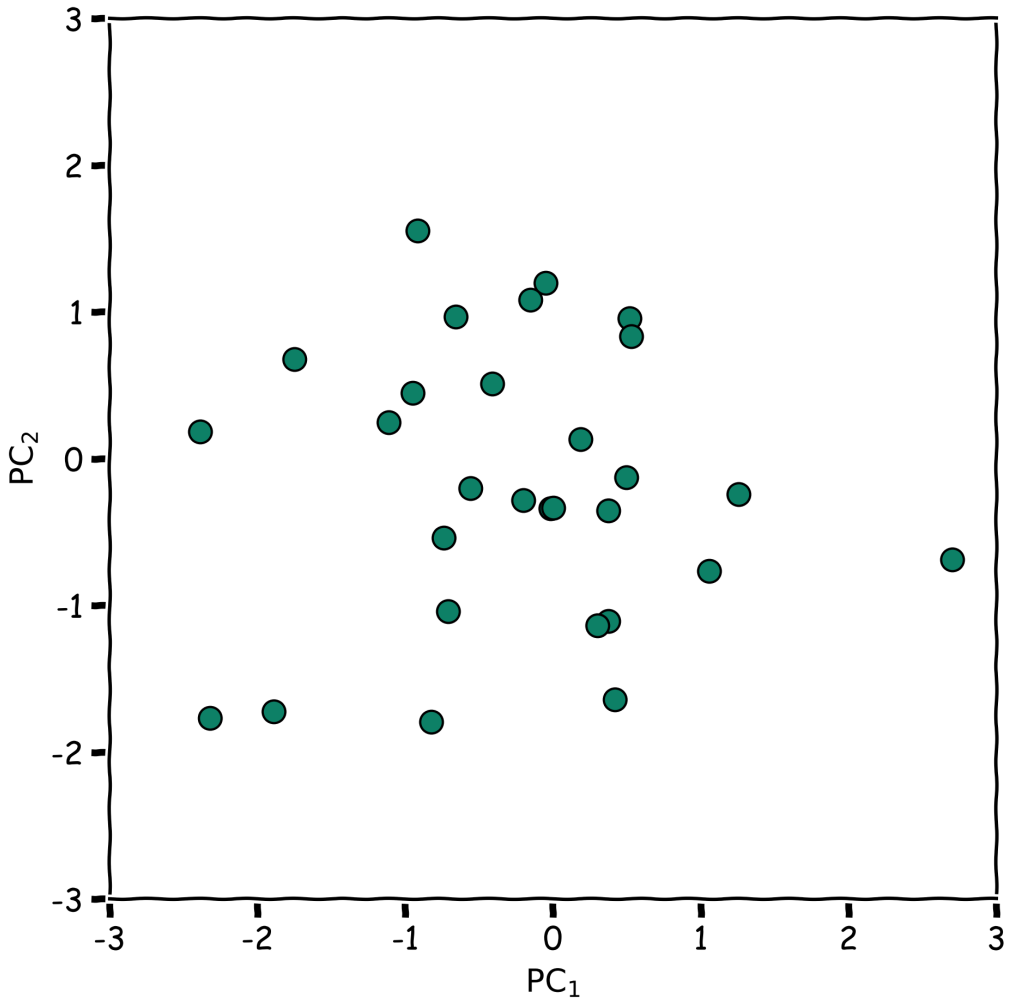

Figure 7: The projections of the input data onto the plane.

3.2 General Description of PCA

3.2.1 Formulation of the Problem and the First PC

Now we are ready to review the strict description of the principal component method in a general form. For this purpose, let's introduce several definitions. This time we will use matrix notations for convenience.

Definition 3.2.1 *Let each of n objects $x_1, x_2, \ldots, x_n$ have p features, that is,*

$$
\begin{aligned}
x_1 &= \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \end{pmatrix}, \\
x_2 &= \begin{pmatrix} x_{21} & x_{22} & \ldots & x_{2p} \end{pmatrix}, \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots, \\
x_n &= \begin{pmatrix} x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}.
\end{aligned}
$$

Then, we will call the input data matrix the matrix of size $[n \times p]$ *of the following form:*

$$F = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix},$$

where the i*th row of the matrix* F *contains the corresponding features of the* i*th object.*

From now on, we will assume that the matrix F is obtained by centering an initial object matrix F'.

$$F' = \begin{pmatrix} x'_{11} & x'_{12} & \dots & x'_{1p} \\ x'_{21} & x'_{22} & \dots & x'_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x'_{n1} & x'_{n2} & \dots & x'_{np} \end{pmatrix}$$

Centering ensures that each predictor has a zero mean. It's performed as follows. We subtract from each jth feature of each object x_i the arithmetic mean of the same jth feature, but taken from all objects $x_1, x_2, ..., x_n$, i.e.

$$x_{ij} = x'_{ij} - \overline{X'_j},\ i = \{1, 2, \dots, n\},\ j = \{1, 2, \dots, p\}.$$

where $\overline{X'_j}$ is the mean value of the jth feature, that is,

$$\overline{X'_j} = \frac{x'_{1j} + x'_{2j} + \ldots + x'_{nj}}{n} = \frac{1}{n}\sum_{i=1}^{n} x'_{ij}.$$

Note that the last expression is nothing more than an arithmetic mean of the elements of the jth column of the matrix F'. Let's show that the sample mean for all the features of the matrix F will be zero after the transformation. According to the definition, the sample mean for the jth feature is an arithmetic mean $\overline{X_j}$ of the jth column of the matrix F. Based on the centering, we obtain

$$\overline{X_j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij} = \frac{1}{n}\sum_{i=1}^{n}(x'_{ij} - \overline{X'_j}) = \frac{1}{n}\sum_{i=1}^{n} x'_{ij} - \frac{1}{n}\sum_{i=1}^{n}\overline{X'_j} =$$

$$= \overline{X'_j} - \frac{1}{n} \cdot n\overline{X'_j} = 0.$$

Recall the definition of the first PC and its loading vector.

Definition 3.2.2 *A straight line, which is passing through the origin and the projections of the centered coordinates of initial objects on which have the largest sample variance, is called the first principal component and designated by* PC_1.

Definition 3.2.3 *A unit directional vector* $\varphi_1 = (\varphi_{11}, \varphi_{21}, \ldots, \varphi_{p1})$ *of the first principal component is called the loading vector of the first principal component.*

Since the length of vector φ_1 is equal to one, that is,

$$|\varphi_1| = \sqrt{\varphi_{11}^2 + \varphi_{21}^2 + \ldots + \varphi_{p1}^2} = 1,$$

then

$$\varphi_{11}^2 + \varphi_{21}^2 + \ldots + \varphi_{p1}^2 = 1.$$

We will write the coordinates of the vector φ_1 in matrix notations as a corresponding column vector:

$$\varphi_1 = \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \\ \vdots \\ \varphi_{p1} \end{pmatrix}.$$

Remark 3.2.1 *It's worth reflecting why the constraint on the loading vector length is introduced. If the length of the loading vector equals unity, then, as has been shown before, the coordinate of the object projection onto the principal component with the basis vector (or the loading vector) can be found as the dot product of the vector corresponding to the object and loading vector. Thus, the constraint on the loading vector length simplifies the calculations.*

Definition 3.2.4 *A score vector of the first PC is a column vector* Z_1 *of the form*

$$Z_1 = \begin{pmatrix} z_{11} \\ z_{21} \\ \vdots \\ z_{n1} \end{pmatrix},$$

consisting of the projections of the centered input data coordinates onto the loading vector of the first PC.

To put it differently, the score vector of the first PC is the new coordinates of the objects with respect to the first principal component.

So, as we have already noted, if the loading vector φ_1 of the first principal component is found, we can calculate the score vector of the first principal component as follows.

Theorem 3.2.1 *Let*

$$F = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}$$

be the input data matrix consisting of centered input data, and φ_1 be the loading vector of the first principal component. Then, the score vector of the first PC can be found from the equality:

$$Z_1 = F\varphi_1.$$

Proof. As we have already noted, when the loading vector of the first principal component $\varphi_1 = (\varphi_{11}, \varphi_{21}, \dots, \varphi_{p1})$ has unit length, the projection coordinate z_{i1} of a random object $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$ can be found as

$$z_{i1} = x_{i1}\varphi_{11} + x_{i2}\varphi_{21} + \dots + x_{ip}\varphi_{p1},$$

if you carefully examine the matrix representation of the relation,

$$Z_1 = F\varphi_1 = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \\ \vdots \\ \varphi_{p1} \end{pmatrix},$$

you'll see that it is the ith coordinate of the vector Z_1. We obtain it according to the matrix multiplication rule. We multiply the ith row by the column. It can be written as follows:

$$Z_1 = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \\ \vdots \\ \varphi_{p1} \end{pmatrix} =$$

$$= \begin{pmatrix} x_{11}\varphi_{11} + x_{12}\varphi_{21} + \dots + x_{1p}\varphi_{p1} \\ x_{21}\varphi_{11} + x_{22}\varphi_{21} + \dots + x_{2p}\varphi_{p1} \\ \vdots \\ x_{n1}\varphi_{11} + x_{n2}\varphi_{21} + \dots + x_{np}\varphi_{p1} \end{pmatrix} = \begin{pmatrix} z_{11} \\ z_{21} \\ \vdots \\ z_{n1} \end{pmatrix}.$$

□

3.2.2 Finding the First PC

There's still an unanswered question. How to find the first principal component? In fact, we only need to find out how to obtain a loading vector. Let's see.

The loading vector Z_1 of the first principal component PC_1 is the product of the data matrix and the loading vector φ_1:

$$Z_1 = F\varphi_1,$$

that is,

$$\begin{pmatrix} z_{11} \\ z_{21} \\ \vdots \\ z_{n1} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \\ \vdots \\ \varphi_{p1} \end{pmatrix},$$

given that the sample variance of the obtained set is the largest.

Recall that the sample variance of the sample Z_1 can be found by the following formula:

$$S^2(Z_1) = \overline{Z_1^2} - \overline{Z_1}^2,$$

where $\overline{Z_1^2}$ is a sample mean of the squared values Z_1, and $\overline{Z_1}^2$ is a squared sample mean.

As to the coordinates, we obtain

$$S^2(Z_1) = \frac{1}{n}\sum_{i=1}^{n} z_{i1}^2 - \left(\frac{1}{n}\sum_{i=1}^{n} z_{i1}\right)^2.$$

Let's consider the right side of this difference, in particular, the expression given in parentheses. As we have already noted, according to the matrix multiplication rule,

$$z_{i1} = \varphi_{11}x_{i1} + \varphi_{21}x_{i2} + \dots + \varphi_{p1}x_{ip},\ i = \{1, 2, \dots, n\}.$$

Then,

$$\frac{1}{n}\sum_{i=1}^{n} z_{i1} = \frac{1}{n}\sum_{i=1}^{n} (\varphi_{11}x_{i1} + \varphi_{21}x_{i2} + \dots + \varphi_{p1}x_{ip}) =$$

$$= \varphi_{11}\frac{1}{n}\sum_{i=1}^{n} x_{i1} + \varphi_{21}\frac{1}{n}\sum_{i=1}^{n} x_{i2} + \dots + \varphi_{p1}\frac{1}{n}\sum_{i=1}^{n} x_{ip}.$$

Since the features x_{ij} are centered, which means their sample means $\overline{X_j}$ are zero,

$$\overline{X_j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij} = 0,\ j = \{1, 2, \dots, p\},$$

each term in the last sum equals zero. Hence, the sum is zero, and the squared sum is also equal to zero:

$$\frac{1}{n}\sum_{i=1}^{n} z_{i1} = 0 \Longrightarrow \left(\frac{1}{n}\sum_{i=1}^{n} z_{i1}\right)^2 = 0.$$

Going back to the sample variance $S^2(Z_1)$, we need to maximize the following expression:

$$S^2(Z_1) = \frac{1}{n}\sum_{i=1}^{n} z_{i1}^2 \longrightarrow \max_{\varphi_1},$$

given that $|\varphi_1| = 1$. Since we can change only the coordinates of φ_1 in the expression, the problem reduces to finding the loading vector φ_1 that maximizes $S^2(Z_1)$. A mathematical formulation of the problem can be written as follows:

$$\arg\max_{\varphi_1}\left(\frac{1}{n}\sum_{i=1}^{n} z_{i1}^2\right) = \arg\max_{\varphi_1}\left(\frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{p}\varphi_{j1}x_{ij}\right)^2\right),$$

given that $|\varphi_1| = 1$. Since the optimization problem does not depend on n, we need to maximize the squared length of the vector Z_1. In other words, we are looking for such a loading vector that maximizes the squared length $|Z_1|^2$ of the score vector Z_1. Let's formulate the problem in the language of math:

$$\arg\max_{\varphi_1}\left(\sum_{i=1}^{n} z_{i1}^2\right) = \arg\max_{\varphi_1}\left(\sum_{i=1}^{n}\left(\sum_{j=1}^{p}\varphi_{j1}x_{ij}\right)^2\right),$$

given that $|\varphi_1| = 1$

3.2.3 Finding an Arbitrary PC

By now, you've learned the concepts of the first principal component, loading vector, and score vector. But what if you want to match more than one coordinate to each object? Let's introduce the following definition.

Definition 3.2.5 *Let the first principal component* PC_1, $p \geq 2$, *be constructed. A straight line, which is passing through the origin in the orthogonal direction to the first principal component* PC_1 *and the projections of the centered coordinates of initial objects on which have the largest sample variance, is called the second principal component and designated by* PC_2.

Let there be $k-1$ *principal components* $\mathsf{PC}_1, \mathsf{PC}_2, ..., \mathsf{PC}_{k-1}$ *that are constructed. A straight line, which is passing through the origin in the orthogonal*

direction to each principal component $\mathsf{PC_i}$, $i \in \{1, 2, ..., (k-1)\}$, $k-1 < p$, *and the projections of the centered coordinates of initial objects on which have the largest sample variance, is called the kth principal component and designated by* $\mathsf{PC_k}$.

Hence, each subsequent principal component is a straight line, which is passing through the origin in the orthogonal direction to all the principal components constructed earlier in a way so that the projections of the coordinates of initial objects on it have the largest sample variance. As has been noted earlier, the orthogonality condition ensures that new coordinates are uncorrelated. Moreover, the number of principal components cannot be greater than the dimensionality of the original space, that is, cannot be greater than p. We invite you to think about it.

As has been done earlier, let's introduce the definition of loadings.

Definition 3.2.6 *A unit directional vector* $\varphi_k = (\varphi_{1k}, \varphi_{2k}, ..., \varphi_{pk})$ *of the kth principal component is called the loading vector of the kth principal component.*

We will write the coordinates of the vector φ_k in matrix notations as a corresponding column vector:

$$\varphi_k = \begin{pmatrix} \varphi_{1k} \\ \varphi_{2k} \\ \vdots \\ \varphi_{pk} \end{pmatrix}.$$

Remark 3.2.2 *To describe the algorithm of finding principal components conveniently, we would like to note that the orthogonality of the principal components is equivalent to the orthogonality of their directional vectors. In coordinates, the orthogonality of the vectors* φ_i *and* φ_j *given that* $i \neq j$ *can be written as follows:*

$$\varphi_{1i}\varphi_{1j} + \varphi_{2i}\varphi_{2j} + \ldots + \varphi_{pi}\varphi_{pj} = 0.$$

The last equality is the equality of the dot product of the vectors $\varphi_i = (\varphi_{1i}, \varphi_{2i}, ..., \varphi_{pi})$ *and* $\varphi_j = (\varphi_{1j}, \varphi_{2j}, ..., \varphi_{pj})$ *to zero.*

We are looking for scores, not the principal components, so let's introduce the last definition.

Definition 3.2.7 *A score vector of the kth principal component is a vector* Z_k *of the form*

$$Z_k = \begin{pmatrix} z_{1k} \\ z_{2k} \\ \vdots \\ z_{nk} \end{pmatrix},$$

consisting of the coordinates of the centered input data projections onto the loading vector of the kth PC.

As before, based on the loading vector, the score vector can be obtained in the following way.

Theorem 3.2.2 *Let*

$$F = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}$$

be the input data matrix consisting of centered input data, and φ_k be the loading vector of the kth principal component. Then, the score vector of the kth PC can be found from the equality:

$$Z_k = F\varphi_k.$$

Without going into details, we would like to note that, when finding the kth principal component, we need to solve the problem of maximizing the square of the length $|Z_k|^2$ of the score vector Z_k given that the loading vector φ_k is orthogonal to all the constructed loading vectors $\varphi_1, \varphi_2, ..., \varphi_{k-1}$. A mathematical formulation of the problem is written as follows:

$$\arg\max_{\varphi_k} \left(\sum_{i=1}^{n} z_{ik}^2 \right) = \arg\max_{\varphi_k} \left(\sum_{i=1}^{n} \left(\sum_{j=1}^{p} \varphi_{jk} x_{ij} \right)^2 \right),$$

$$\varphi_k \perp \varphi_i,\ i \in \{1, 2, ..., (k-1)\},\ |\varphi_k| = 1$$

To find the kth principal component, or (what's the same) the loading vector φ_k and, consecutively, the score vector Z_k, you can follow an algorithm that we will describe in a moment.

1. It's postulated that the vector φ_k has unit length,

$$\sum_{i=1}^{p} \varphi_{ik}^2 = \varphi_{1k}^2 + \varphi_{2k}^2 + ... + \varphi_{pk}^2 = 1.$$

2. It's postulated that the vector φ_k is orthogonal to each of the constructed loading vectors $\varphi_1, \varphi_2, ..., \varphi_{k-1}$. In coordinates, it's equivalent to the $(k-1)$ equality of the following form:

$$\varphi_{1k}\varphi_{1i} + \varphi_{2k}\varphi_{2i} + ... + \varphi_{pk}\varphi_{pi} = 0,\ i \in \{1, 2, ..., (k-1)\}$$

3. Given the setting, we are solving the following problem:

$$\arg\max_{\varphi_k}\left(|Z_k|^2\right) = \arg\max_{\varphi_k}\left(\sum_{i=1}^{n} z_{ik}^2\right) = \arg\max_{\varphi_k}\left(\sum_{i=1}^{n}\left(\sum_{j=1}^{p}\varphi_{jk}x_{ij}\right)^2\right).$$

4. To obtain the score vector of the kth principal component, we calculate

$$Z_k = F\varphi_k.$$

3.3 An Example of Finding the First Principal Component

Let's return to the example of sales bonuses. Please take a look at the input data given in the table.

Employee (x_i)	**Premium** (X_1')**, pcs**	**Economy** (X_2')**, pcs**
1	9	19
2	6	22
3	11	27
4	12	25
5	7	22

Let's find the mean values for each feature:

$$\overline{X_1'} = \frac{9+6+11+12+7}{5} = 9,$$

$$\overline{X_2'} = \frac{19+22+27+25+22}{5} = 23$$

then center the object x_1

$$x_{11} = x_{11}' - \overline{X_1'} = 9-9 = 0, \quad x_{12} = x_{12}' - \overline{X_2'} = 19-23 = -4,$$

and the object x_2

$$x_{21} = x_{21}' - \overline{X_1'} = 6-9 = -3, \quad x_{22} = x_{22}' - \overline{X_2'} = 22-23 = -1,$$

etc. Hence, we obtain the table of centered object coordinates.

Employee (x_i)	**Feature** X_1	**Feature** X_2
1	0	-4
2	-3	-1
3	2	4
4	3	2
5	-2	-1

Based on the table, let's write down a matrix of the centered input data:

$$F_{5\times 2} = \begin{pmatrix} 0 & -4 \\ -3 & -1 \\ 2 & 4 \\ 3 & 2 \\ -2 & -1 \end{pmatrix}.$$

Recall that Z_1 is found as follows:

$$Z_1 = F\varphi_1,$$

where $\varphi_1 = \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \end{pmatrix}$ is a loading vector, thus,

$$\varphi_{11}^2 + \varphi_{21}^2 = 1.$$

Based on the described algorithm, to calculate the scores of the first principal component, we only need to compute

$$Z_1 = \begin{pmatrix} 0 & -4 \\ -3 & -1 \\ 2 & 4 \\ 3 & 2 \\ -2 & -1 \end{pmatrix} \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \end{pmatrix} = \begin{pmatrix} -4\varphi_{21} \\ -3\varphi_{11} - \varphi_{21} \\ 2\varphi_{11} + 4\varphi_{21} \\ 3\varphi_{11} + 2\varphi_{21} \\ -2\varphi_{11} - \varphi_{21} \end{pmatrix}$$

It has been well noted that, to maximize the sample variance $S^2(Z_1)$, it's enough to maximize the square of Z_1 length considering that $|\varphi_1| = 1$, that is, to maximize the following expression:

$$|Z_1|^2 = (-4\varphi_{21})^2 + (-3\varphi_{11} - \varphi_{21})^2 + (2\varphi_{11} + 4\varphi_{21})^2 +$$

$$+ (3\varphi_{11} + 2\varphi_{21})^2 + (-2\varphi_{11} - \varphi_{21})^2.$$

We expand the parentheses and collect like terms. As a result, we obtain the expression for the square of Z_1 length of the following form:

$$|Z_1|^2 = 26\varphi_{11}^2 + 38\varphi_{11}\varphi_{21} + 38\varphi_{21}^2.$$

Here we have a function of two variables. To find its maximum value given that $|\varphi_1| = 1$, we can use the Lagrange method. However, there's another way to do it. Let's consider two cases.

1. Since

$$\varphi_{11}^2 + \varphi_{21}^2 = 1,$$

the expression for $|Z_1|^2$ can be rewritten as follows:

$$|Z_1|^2 = 26 + \frac{38\varphi_{11}\varphi_{21}}{\varphi_{11}^2 + \varphi_{21}^2} + \frac{12\varphi_{21}^2}{\varphi_{11}^2 + \varphi_{21}^2}.$$

Assume that $\varphi_{11} \neq 0$. We divide the numerator and denominator in each fraction by φ_{11}^2 and introduce the substitution $t = \frac{\varphi_{21}}{\varphi_{11}}$. t can take any values from the set of all real numbers. Let's consider the function of one variable $G(t)$

$$G(t) = 26 + \frac{12t^2}{1+t^2} + \frac{38t}{1+t^2},$$

which maximum value we are looking for. Recall that, to find critical points, we can find the first derivative and zeros of the numerator and denominator. Then,

$$G'(t) = \frac{-38t^2 + 24t + 38}{(1+t^2)^2}.$$

Since the denominator is not zero, let's equate only the numerator to zero and obtain the quadratic equation:

$$-38t^2 + 24t + 38 = 0,$$

with the square roots

$$t_{1,2} = \frac{6 \pm \sqrt{397}}{19}.$$

The function has a local maximum at the point where the derivative changes the sign from plus to minus. In our case, $t = \frac{6+\sqrt{397}}{19}$ is a local maximum. Moreover,

$$G\left(\frac{6+\sqrt{397}}{19}\right) = 32 + \sqrt{397} \approx 51.925.$$

On the other hand, the function $G(t)$ is decreasing on the interval $\left(-\infty, \frac{6-\sqrt{397}}{19}\right)$. Therefore, it tends to a possible maximum value given that $t \to -\infty$.

To find this value, let's find the limit

$$\lim_{t\to-\infty}\left(26 + \frac{12t^2}{1+t^2} + \frac{38t}{1+t^2}\right) = 38.$$

2. Now, let's consider the case $\varphi_{11} = 0$. Given that

$$|Z_1|^2 = 26\varphi_{11}^2 + 38\varphi_{11}\varphi_{21} + 38\varphi_{21}^2,$$

and

$$\varphi_{11}^2 + \varphi_{21}^2 = 1,$$

we obtain that $\varphi_{21} = \pm 1$ and

$$|Z_1|^2 = 38.$$

The comparison of the three obtained values reveals that the function attains its maximum at

$$\frac{\varphi_{21}}{\varphi_{11}} = \frac{6 + \sqrt{397}}{19}.$$

Thus, from the system of equations,

$$\begin{cases} \varphi_{11}^2 + \varphi_{21}^2 = 1 \\ \frac{\varphi_{21}}{\varphi_{11}} = \frac{6+\sqrt{397}}{19} \end{cases},$$

we get 2 pairs of solutions: $\varphi_{11} \approx 0.591$, $\varphi_{21} \approx 0.807$ and $\varphi_{11} \approx -0.591$, $\varphi_{21} \approx -0.807$. The choice of a pair does not matter because the only difference is the direction. In this case, the new coordinates (or scores) of objects with respect to the first PC can be written as follows:

$$Z_1 = \begin{pmatrix} 0 & -4 \\ -3 & -1 \\ 2 & 4 \\ 3 & 2 \\ -2 & -1 \end{pmatrix} \begin{pmatrix} 0.591 \\ 0.807 \end{pmatrix} = \begin{pmatrix} -3.226 \\ -2.580 \\ 4.409 \\ 3.387 \\ -1.989 \end{pmatrix}.$$

Since the direction of the first PC is defined by the loading vector φ_1, we can construct the straight line having the equation:

$$X_2 = \frac{\varphi_{21}}{\varphi_{11}} X_1 = \frac{0.807}{0.591} X_1 = 1.365 X_1.$$

The geometric interpretation is shown in fig. **??**. The green points are the centered initial features. The orange line is the first PC, that is, a straight line with the loading vector φ_1. The new coordinates (or scores) are the coordinates of initial object projections onto the obtained straight line (blue points). Moreover, by the description of the method, the chosen straight line maximizes the sample variance of the scores.

Each manager is now characterized by only one number. Thus, if we sort the results in descending order, the largest bonus should be given to the employee under the number $i = 3$, then the employee under the number $i = 4$, and so on. The negative results do not imply that the employees should pay a fine. The less the value, the less the bonus an employee should receive.

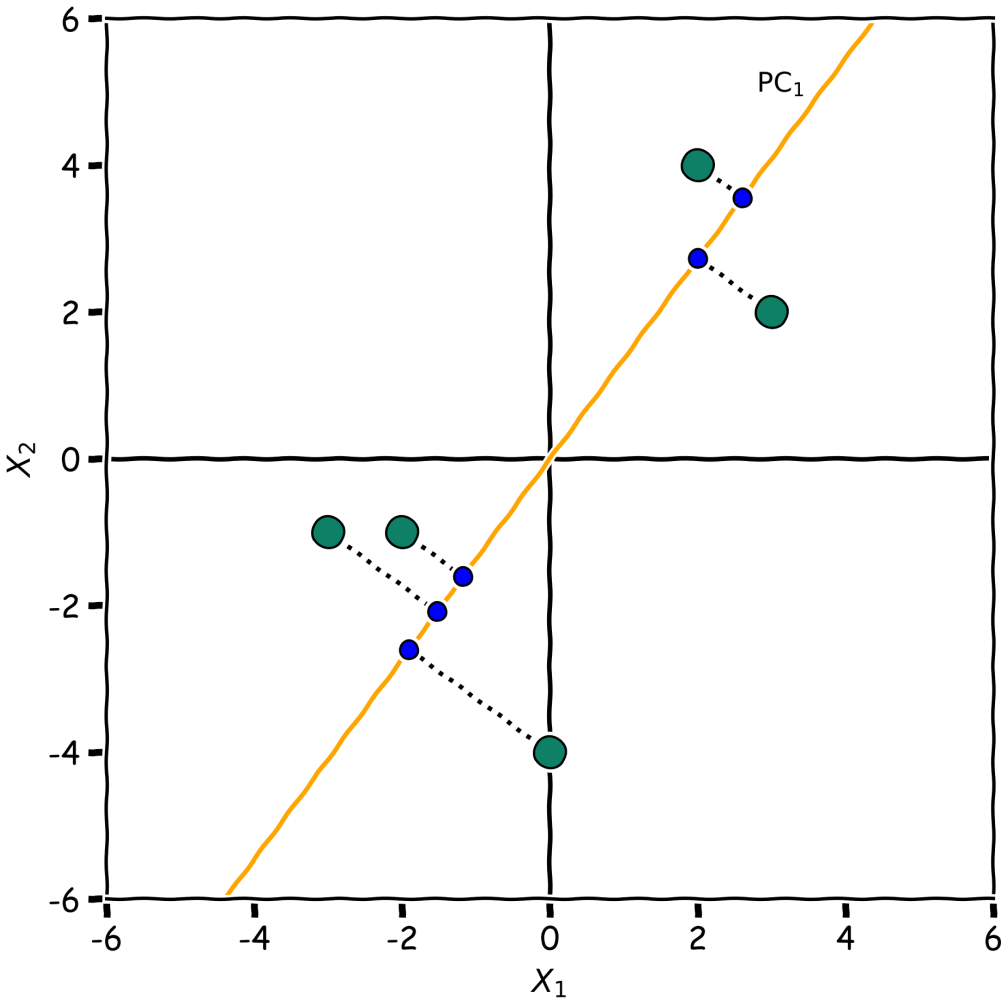

Figure 8: Constructing the first principal component.

4 One More Way of Constructing Principal Components

By now you know that finding principal components is a compelling difficulty particularly because the search of each subsequent principal component requires solving the optimization problem with an increasing number of constraints (the orthogonality of the loading vector of the subsequent PC to all the loading vectors of the previous PCs). But still we can ask a few sensible questions. Have we constructed enough PCs? Did we lose too much useful information about the objects of interest while reducing the dimensionality of the feature space? How well can we estimate the amount of lost information? Can we reconstruct the input data (even with losses)? Is there a more general method of finding the loading vectors of the principal components not iteratively, but all of them at once? We can answer all of these questions in the affirmative, but before that, let's do some preparations.

4.1 Additional Information from Linear Algebra

Earlier we've said that, while using the principal component method, we change a coordinate system, or, what's the same, the basis. How to change the basis in math? Let's see.

Let there be two bases. The first one consists of elements $e_1, e_2, \ldots, e_p$ (we will call it old), and the second consists of elements $e'_1, e'_2, \ldots, e'_p$ (the new basis). Since two bases are in the same space, the vectors of the new basis can be expressed in terms of the vectors of the old basis:

$$\begin{aligned} e'_1 &= \varphi_{11}e_1 + \varphi_{21}e_2 + \cdots + \varphi_{p1}e_p, \\ e'_2 &= \varphi_{12}e_1 + \varphi_{22}e_2 + \cdots + \varphi_{p2}e_p, \\ &\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ e'_p &= \varphi_{1p}e_1 + \varphi_{2p}e_2 + \cdots + \varphi_{pp}e_p, \end{aligned}$$

where φ_{ij} are numbers, $i, j \in \{1, 2, ..., p\}$. It turns out that, for coordinate transformations, it's convenient to write down a so-called transformation matrix.

Definition 4.1.1 *A matrix*

$$\Phi = \begin{pmatrix} \varphi_{11} & \varphi_{12} & \cdots & \varphi_{1p} \\ \varphi_{21} & \varphi_{22} & \cdots & \varphi_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{p1} & \varphi_{p2} & \cdots & \varphi_{pp} \end{pmatrix}$$

is called a transformation matrix from $e_1, e_2, ..., e_p$ *to* $e'_1, e'_2, ..., e'_p$.

Note that the expansion coefficients of the vector e'_1 in the old basis are in the first column of the transformation matrix Φ. The expansion coefficients of the vector e'_2 in the old basis are in the second column of the transformation matrix Φ, and so on.

This matrix Φ makes it easier to transform the coordinates of a vector from the new basis to the old one.

Lemma 4.1.1 *In the bases* $e_1, e_2, ..., e_p$ *and* $e'_1, e'_2, ..., e'_p$, *the vector* X *is given as*

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} \quad \textit{and} \quad X = \begin{pmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_p \end{pmatrix},$$

respectively, and Φ *is a transformation matrix from the old to new. Therefore, the following relation is true:*

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} = \Phi \begin{pmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_p \end{pmatrix}.$$

The coordinates of the vector X in the old basis are the product of the transformation matrix and the coordinates of the vector X in the new basis.

Proof. Let's write down the expansion of the vector X in the new and old bases. On the one side, we obtain

$$X = x_1 e_1 + x_2 e_2 + \ldots + x_p e_p,$$

and, on the other side,

$$X = x'_1 e'_1 + x'_2 e'_2 + \ldots + x'_p e'_p.$$

Since the new and old bases are related,

$$e'_1 = \varphi_{11} e_1 + \varphi_{21} e_2 + \cdots + \varphi_{p1} e_p,$$

$$e'_2 = \varphi_{12} e_1 + \varphi_{22} e_2 + \cdots + \varphi_{p2} e_p,$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$e'_p = \varphi_{1p} e_1 + \varphi_{2p} e_2 + \cdots + \varphi_{pp} e_p,$$

then, after the substitution and transposition of the terms in the right part of the equation, and after collecting the like terms before the basis elements e_i, we obtain

$$X = x'_1 e'_1 + x'_2 e'_2 + \ldots + x'_p e'_p = (x'_1 \varphi_{11} + x'_2 \varphi_{12} + \ldots + x'_p \varphi_{1p}) e_1 +$$

$$+ (x'_1 \varphi_{21} + x'_2 \varphi_{22} + \ldots + x'_p \varphi_{2p}) e_2 + \ldots + (x'_1 \varphi_{p1} + x'_2 \varphi_{p2} + \ldots + x'_p \varphi_{pp}) e_p.$$

On the other side, as has been noted,

$$X = x_1 e_1 + x_2 e_2 + \ldots + x_p e_p.$$

Since the expansion of a vector relative to a basis is unique, we have

$$x_1 = x'_1 \varphi_{11} + x'_2 \varphi_{12} + \ldots + x'_p \varphi_{1p},$$

$$x_2 = x'_1 \varphi_{21} + x'_2 \varphi_{22} + \ldots + x'_p \varphi_{2p},$$

$$\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$
$$x_p = x'_1\varphi_{p1} + x'_2\varphi_{p2} + \ldots + x'_p\varphi_{pp}.$$

This can be written in matrix notation as follows:

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} = \begin{pmatrix} \varphi_{11} & \varphi_{12} & \cdots & \varphi_{1p} \\ \varphi_{21} & \varphi_{22} & \cdots & \varphi_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{p1} & \varphi_{p2} & \cdots & \varphi_{pp} \end{pmatrix} \begin{pmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_p \end{pmatrix}.$$

□

In PCA, we don't change an arbitrary basis to an arbitrary basis, we change an orthonormal basis to orthonormal. Let's remind ourselves what it means.

Definition 4.1.2 *The basis $e_1, e_2, ..., e_p$ is orthonormal if*

$$|e_1| = |e_2| = \ldots = |e_p| = 1$$

and elements e_i and e_j are orthogonal when $i \neq j$, that is, $(e_i, e_j) = 0$, $i \neq j$.

In other words, the basis is orthonormal if the lengths of all its elements are equal to unity, and the elements are pairwise orthogonal. As you may have noticed, these are the loading vector requirements we are looking for while using PCA.

When we change an orthonormal basis to an orthonormal basis, the transformation matrix Φ reveals useful properties.

Theorem 4.1.1 *Let $e_1, e_2, ..., e_p$ and $e'_1, e'_2, ..., e'_p$ be two orthonormal bases and Φ be a transformation matrix. Then,*

I. *The columns of the matrix Φ are of unit length and pairwise orthogonal.*
II. *The rows of the matrix Φ are of unit length and pairwise orthogonal.*
III. *$\Phi^{-1} = \Phi^T$, that is, the inverse and transpose matrices are the same.*

Proof. Let's first prove the points I and III, and then II.
I. To do so, we are going to use the change-of-basis formulas:

$$e'_1 = \varphi_{11}e_1 + \varphi_{21}e_2 + \cdots + \varphi_{p1}e_p,$$
$$e'_2 = \varphi_{12}e_1 + \varphi_{22}e_2 + \cdots + \varphi_{p2}e_p,$$
$$\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$
$$e'_p = \varphi_{1p}e_1 + \varphi_{2p}e_2 + \cdots + \varphi_{pp}e_p.$$

Note that, since the bases $e_1, e_2, ..., e_p$ and $e'_1, e'_2, ..., e'_p$ are orthonormal,

$$(e'_k, e'_l) = (e_k, e_l) = \begin{cases} 1, & k = l \\ 0, & k \neq l \end{cases}, \quad k,\ l = \{1, 2, \ldots, p\}.$$

1. If $i = j$, then

$$1 = (e'_i, e'_i) = ((\varphi_{1i}e_1 + \varphi_{2i}e_2 + \cdots + \varphi_{pi}e_p), (\varphi_{1i}e_1 + \varphi_{2i}e_2 + \cdots + \varphi_{pi}e_p)) =$$

$$= \left(\sum_{j=1}^{p} \varphi_{ji}e_j,\ \sum_{k=1}^{p} \varphi_{ki}e_k\right) = \sum_{j,k=1}^{p} \varphi_{ji}\varphi_{ki}(e_j, e_k) = \varphi_{1i}^2 + \varphi_{2i}^2 + \ldots + \varphi_{pi}^2,$$

since the dot product (e_j, e_k) is not zero only when $j = k$. Therefore, the columns of the matrix Φ are of unit length.
2. If $i \neq j$, then

$$0 = (e'_i, e'_j) = ((\varphi_{1i}e_1 + \varphi_{2i}e_2 + \cdots + \varphi_{pi}e_p), (\varphi_{1j}e_1 + \varphi_{2j}e_2 + \cdots + \varphi_{pj}e_p)) =$$

$$= \left(\sum_{j=1}^{p} \varphi_{ji}e_j,\ \sum_{k=1}^{p} \varphi_{ki}e_k\right) = \sum_{j,k=1}^{p} \varphi_{ji}\varphi_{ki}(e_j, e_k) = \varphi_{1i}\varphi_{1j} + \varphi_{2i}\varphi_{2j} + \varphi_{pi}\varphi_{pj},$$

thus, the columns of the matrix Φ are pairwise orthogonal.
III. Let's consider the product $\Phi^T\Phi$. Since the columns of the matrix Φ (the rows of the matrix Φ^T) are pairwise orthogonal and of unit length, we obtain

$$\Phi^T\Phi = \begin{pmatrix} \varphi_{11} & \varphi_{21} & \cdots & \varphi_{p1} \\ \varphi_{12} & \varphi_{22} & \cdots & \varphi_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{1p} & \varphi_{2p} & \cdots & \varphi_{pp} \end{pmatrix} \begin{pmatrix} \varphi_{11} & \varphi_{12} & \cdots & \varphi_{1p} \\ \varphi_{21} & \varphi_{22} & \cdots & \varphi_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{p1} & \varphi_{p2} & \cdots & \varphi_{pp} \end{pmatrix} =$$

$$= \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{pmatrix} = E.$$

Hence,

$$\Phi^T\Phi = E,$$

thus, the matrix Φ^T is the left inverse of Φ. According to the square matrix properties, Φ^T is also the right inverse of Φ, thus,

$$\Phi\Phi^T = E,$$

therefore,

$$\Phi^{-1} = \Phi^T.$$

II. Since $\Phi^T = \Phi^{-1}$ and by using the properties of the inverse matrix, we obtain

$$\Phi^T\Phi = \Phi^{-1}\Phi = \Phi\Phi^{-1} = \Phi\Phi^T = E.$$

The last equality can be written in the following form

$$\begin{pmatrix} \varphi_{11} & \varphi_{12} & \dots & \varphi_{1p} \\ \varphi_{21} & \varphi_{22} & \dots & \varphi_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{p1} & \varphi_{p2} & \dots & \varphi_{pp} \end{pmatrix} \begin{pmatrix} \varphi_{11} & \varphi_{21} & \dots & \varphi_{p1} \\ \varphi_{12} & \varphi_{22} & \dots & \varphi_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{1p} & \varphi_{2p} & \dots & \varphi_{pp} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix},$$

from it follows (the same as in I) that rows of the matrix Φ are pairwise orthogonal and of unit length. □

To complete the picture, let's introduce a more general definition.

Definition 4.1.3 *The matrix Φ, for which $\Phi^{-1} = \Phi^T$ is true, is called orthogonal.*

It turns out that any orthogonal matrix transforms an orthonormal basis into an orthonormal one. Thus, for any orthogonal matrix, the points I and II of the previous theorem are true.

4.2 Back to PCA

Now we can apply the studied apparatus to PCA. Let F be an input data matrix of size $[n \times p]$ consisting of n objects, each having p features:

$$F = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}.$$

We will also think that the centering is completed. Our goal is to make a transition from one orthonormal basis (initial) to another orthonormal (provided by PCA). In this case, the transformation matrix Φ will be orthogonal. Thus, as we've noted, for the matrix

$$\Phi = \begin{pmatrix} \varphi_{11} & \varphi_{12} & \dots & \varphi_{1p} \\ \varphi_{21} & \varphi_{22} & \dots & \varphi_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{p1} & \varphi_{p2} & \dots & \varphi_{pp} \end{pmatrix}$$

the following statements are true:

1. $\Phi^T = \Phi^{-1}$.

2. The rows and columns of the matrix Φ are orthonormal vectors.

The rows of the matrix F are objects (or vectors) of the input dataset. In further transformations, it will be more convenient for us that the object is not a row, but a column, so let the column

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix}$$

be an object X (a row of the matrix F) with the coordinates relative to the original basis (we will consider it new), and the column

$$Z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{pmatrix}$$

be the same object with the coordinates relative to another basis (say, old), and let Φ be the transformation matrix.

From the equality $Z = \Phi X$, we express the vector X:

$$X = \Phi^{-1} Z = \Phi^T Z,$$

where

$$\Phi^T = \begin{pmatrix} \varphi_{11} & \varphi_{21} & \cdots & \varphi_{p1} \\ \varphi_{12} & \varphi_{22} & \cdots & \varphi_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{1p} & \varphi_{2p} & \cdots & \varphi_{pp} \end{pmatrix}.$$

The expression $X = \Phi^T Z$ can be written in matrix notations as follows:

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} = \begin{pmatrix} \varphi_{11} & \varphi_{21} & \cdots & \varphi_{p1} \\ \varphi_{12} & \varphi_{22} & \cdots & \varphi_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{1p} & \varphi_{2p} & \cdots & \varphi_{pp} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{pmatrix}$$

After multiplying matrices Φ^T and Z, we get:

$$x_1 = \varphi_{11} z_1 + \varphi_{21} z_2 + \cdots + \varphi_{p1} z_p$$

$$x_2 = \varphi_{12} z_1 + \varphi_{22} z_2 + \cdots + \varphi_{p2} z_p$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$x_p = \varphi_{1p} z_1 + \varphi_{2p} z_2 + \cdots + \varphi_{pp} z_p.$$

Since z_i is a number, the vector X takes the following form:

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} = \begin{pmatrix} \varphi_{11} \\ \varphi_{12} \\ \vdots \\ \varphi_{1p} \end{pmatrix} z_1 + \begin{pmatrix} \varphi_{21} \\ \varphi_{22} \\ \vdots \\ \varphi_{2p} \end{pmatrix} z_2 + \cdots + \begin{pmatrix} \varphi_{p1} \\ \varphi_{p2} \\ \vdots \\ \varphi_{pp} \end{pmatrix} z_p.$$

Therefore, we can introduce the following definition.

Definition 4.2.1 *The representation of the vector X in the form*

$$X = \begin{pmatrix} \varphi_{11} \\ \varphi_{12} \\ \vdots \\ \varphi_{1p} \end{pmatrix} z_1 + \begin{pmatrix} \varphi_{21} \\ \varphi_{22} \\ \vdots \\ \varphi_{2p} \end{pmatrix} z_2 + \cdots + \begin{pmatrix} \varphi_{p1} \\ \varphi_{p2} \\ \vdots \\ \varphi_{pp} \end{pmatrix} z_p,$$

where $(\varphi_{i1}\ \varphi_{i2}\ \ldots\ \varphi_{ip})$ are the rows of the transformation matrix, and z_i are the corresponding coordinates of the object in the old basis, is called the expansion of the vector X into its principal components.

We designate by

$$\varphi_1 = \begin{pmatrix} \varphi_{11} \\ \varphi_{12} \\ \vdots \\ \varphi_{1p} \end{pmatrix},\ \varphi_2 = \begin{pmatrix} \varphi_{21} \\ \varphi_{22} \\ \vdots \\ \varphi_{2p} \end{pmatrix},\ \ldots,\ \varphi_p = \begin{pmatrix} \varphi_{p1} \\ \varphi_{p2} \\ \vdots \\ \varphi_{pp} \end{pmatrix}$$

the columns of the matrix Φ^T (the rows of the matrix Φ). Then, the expansion of the vector X into the principal components takes the following form:

$$X = \sum_{i=1}^{p} \varphi_i z_i.$$

Assume that we want to keep $k \leq p$ first principal components:

$$\widehat{X} = \begin{pmatrix} \widehat{x}_1 \\ \widehat{x}_2 \\ \vdots \\ \widehat{x}_p \end{pmatrix} = \begin{pmatrix} \varphi_{11} \\ \varphi_{12} \\ \vdots \\ \varphi_{1p} \end{pmatrix} \cdot z_1 + \begin{pmatrix} \varphi_{21} \\ \varphi_{22} \\ \vdots \\ \varphi_{2p} \end{pmatrix} \cdot z_2 + \cdots + \begin{pmatrix} \varphi_{k1} \\ \varphi_{k2} \\ \vdots \\ \varphi_{kp} \end{pmatrix} \cdot z_k,$$

or (in a less cumbersome form):

$$\widehat{X} = \sum_{i=1}^{k} \varphi_i z_i.$$

We expect that the fewer principal components we take, the more information about the initial objects is lost. Since X is an object of the input data matrix and since it has p features, and each feature is a random variable, the object is also a random variable. Thus, we can consider the so-called MSE. It stands for a mean squared error that is equal to the expected value of the square of the norm of the difference between X and $\widehat{X}$, that is,

$$\mathsf{E}\left\|X-\widehat{X}\right\|^2 = \mathsf{E}\left\|\sum_{i=1}^{p}\varphi_i z_i - \sum_{i=1}^{k}\varphi_i z_i\right\|^2 =$$

$$= \mathsf{E}\left\|\sum_{i=k+1}^{p}\varphi_i z_i\right\|^2 = \mathsf{E}\left(\sum_{i=k+1}^{p}\varphi_i z_i, \sum_{i=k+1}^{p}\varphi_i z_i\right).$$

Considering that the vectors φ_i are pairwise orthogonal, and the length of each of them is equal to unity, that is,

$$(\varphi_i, \varphi_j) = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases},$$

where $i, j = \{k+1, \ldots, p\}$, and, using the linearity property of the dot product, we obtain that

$$\mathsf{E}\left\|X-\widehat{X}\right\|^2 = \mathsf{E}\left(\sum_{i=k+1}^{p}\varphi_i z_i, \sum_{i=k+1}^{p}\varphi_i z_i\right) = \mathsf{E}\left(\sum_{i=k+1}^{p} z_i^2\right) = \sum_{i=k+1}^{p}\mathsf{E}z_i^2.$$

Since the expression $Z = \Phi X$ is rewritten in matrix notations as follows:

$$\begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{pmatrix} = \begin{pmatrix} \varphi_{11} & \varphi_{12} & \ldots & \varphi_{1p} \\ \varphi_{21} & \varphi_{22} & \ldots & \varphi_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{p1} & \varphi_{p2} & \ldots & \varphi_{pp} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix},$$

each coordinate z_i is equal to the product of the ith row of the matrix Φ and the column X, and, given that φ_i is the ith column of the matrix Φ^T, we get:

$$z_i = \varphi_i^T X.$$

On the other hand, z_i can be represented as the product of the transposed vector X and the vector φ_i:

$$z_i = X^T \varphi_i.$$

Going back to the considered expression and using the expected value properties, we obtain

$$\mathsf{E}\left\|X-\widehat{X}\right\|^2=\sum_{i=k+1}^{p}\mathsf{E}z_i^2=\sum_{i=k+1}^{p}\mathsf{E}\left(z_i\cdot z_i\right)=$$

$$=\sum_{i=k+1}^{p}\mathsf{E}\left(\varphi_i^TX\cdot X^T\varphi_i\right)=\sum_{i=k+1}^{p}\varphi_i^T\mathsf{E}\left(X\cdot X^T\right)\varphi_i.$$

Let's denote $\Theta=\mathsf{E}\left(X\cdot X^T\right)$.

Remark 4.2.1 *As you can see, Θ is a covariance matrix for a centered random vector X.*

We obtain the expression to be minimized:

$$\mathsf{E}\left\|X-\widehat{X}\right\|^2=\sum_{i=k+1}^{p}\varphi_i^T\Theta\varphi_i\longrightarrow\min_{\varphi_i}.$$

Let's recall the definitions that we are going to use.

Definition 4.2.2 *A non-zero vector φ is called an eigenvector of the matrix Θ if, for some number λ, the following equality is true:*

$$\Theta\varphi=\lambda\varphi.$$

The number λ is called an eigenvalue of the matrix Θ.

Remark 4.2.2 *Note that the covariance matrix is symmetric, that is, $\Theta=\Theta^T$, and it has p non-negative eigenvalues according to its multiplicity. Moreover, we can use its eigenvectors to construct an orthonormal set of p elements.*

Theorem 4.2.1 *Let*

$$\lambda_1\geq\lambda_2\geq\ldots\geq\lambda_p\geq0$$

be eigenvalues of the covariance matrix. The minimum of the expression

$$\mathsf{E}\left\|X-\widehat{X}\right\|^2=\sum_{i=k+1}^{p}\varphi_i^T\Theta\varphi_i$$

is attained when φ_i are orthonormal eigenvectors corresponding to the smallest eigenvalues λ_i of the matrix Θ, and

$$\min_{\varphi_i}\mathsf{E}\left\|X-\widehat{X}\right\|^2=\min_{\varphi_i}\sum_{i=k+1}^{p}\varphi_i^T\Theta\varphi_i=\sum_{i=k+1}^{p}\lambda_i.$$

Proof. To prove the theorem, we need to find such vectors φ_i that minimize the value of this expression. Therefore, here we have the conditional extremum problem. To solve it, let's use the Lagrange multiplier, where $\mathsf{E}\left\|X-\widehat{X}\right\|^2$ is a minimized expression, and $|\varphi_i|=1, \varphi_i^T\varphi_i=1$ are the limitations. Recall that, in the Lagrange method, the problem of finding a conditional extremum is reduced to finding an extremum of the so-called Lagrange function. The Lagrange function has the following form:

$$L(\varphi,\lambda)=\sum_{i=k+1}^{p}\varphi_i^T\Theta\varphi_i-\sum_{i=k+1}^{p}\lambda_i\left(\varphi_i^T\varphi_i-1\right).$$

Let's rewrite this expression by using sigma notation for the terms on the right side:

$$L(\varphi,\lambda)=\sum_{i=k+1}^{p}\left(\varphi_i^T\Theta\varphi_i-\lambda_i\left(\varphi_i^T\varphi_i-1\right)\right).$$

We differentiate the Lagrange function with respect to each variable, equate the partial derivatives to zero, and solve the obtained system of equations. In our case, both vectors φ_i and numbers λ_i are unknown. We can show that

$$\frac{\partial\left(\varphi_i^T\Theta\varphi_i\right)}{\partial\varphi_i}=(\Theta+\Theta^T)\varphi_i.$$

In the case of the symmetric matrix Θ (and the covariance matrix is symmetric), we obtain

$$\frac{\partial\left(\varphi_i^T\Theta\varphi_i\right)}{\partial\varphi_i}=2\Theta\varphi_i,$$

hence,

$$\frac{\partial L(\varphi,\lambda)}{\partial\varphi_i}=2\Theta\varphi_i-2\lambda_i\varphi_i=0.$$

Otherwise, the last expression is rewritten as follows

$$\Theta\varphi_i=\lambda_i\varphi_i.$$

What you see is the definition of eigenvalues and eigenvectors of the matrix Θ. If we plug this expression in L, then, considering that $\sum_{i=k+1}^{p}\lambda_i\left(\varphi_i^T\varphi_i-1\right)=0$ (due to constraint equations), we obtain:

$$\mathsf{E}\left\|X-\widehat{X}\right\|^2=\sum_{i=k+1}^{p}\varphi_i^T\lambda_i\varphi_i=\sum_{i=k+1}^{p}\lambda_i.$$

□

To put it differently, the expression is minimized when λ_i are the smallest eigenvalues in the matrix Θ. Hence, to lose less information while reducing the dimensionality, it's reasonable to take the eigenvector corresponding to the largest eigenvalue of the matrix Θ as a loading vector of the first principal component and to take the eigenvector corresponding to the next largest eigenvalue of the matrix Θ as a loading vector of the second PC, and so on.

Remark 4.2.3 *We would like to note without details that the ith eigenvalue λ_i equals the variance of the scores of the ith PC.*

4.3 Defining the Number of Principal Components

Now the most important question is how many principal components should be left to avoid losing too much information about the initial objects. Let's introduce the following definition.

Definition 4.3.1 *Let*

$$\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p \geq 0,$$

be eigenvalues of the covariance matrix, and let's take the eigenvectors of the covariance matrix as the loading vectors of the first k PCs. These eigenvectors correspond to their largest eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_k$. The value

$$\delta_k = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i}$$

is called a fraction of variance explained.

Remark 4.3.1 *The value $1 - \delta_k$ is called a fraction of variance unexplained (residual).*

δ_k that takes values from zero to one shows what part of the variance is considered when we are using the first k PCs with respect to the entire variance. Thus, the closer δ_k to one, the less the amount of information about the initial objects we lose.

It makes sense to keep such a number of principal components, so that subsequent adding does not considerably change the fraction of variance explained. Figure 8 makes it clear.

The abscissa stands for the number of preserved principal components, and the ordinate stands for the fraction of variance explained. The fraction of variance explained does not considerably change from the 3rd PC, and three principal components describe about 95% of the variance. In this case, it's reasonable to keep only three principal components.

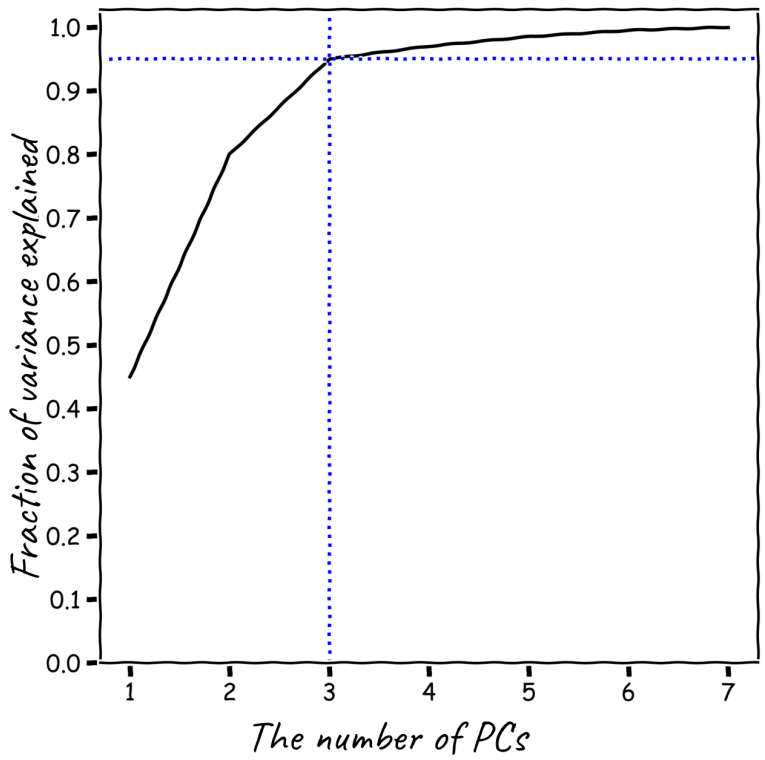

Figure 9: Defining the number of principal components.

4.4 Algorithm

Let's provide a step-by-step instruction for finding principal components. Let F be a centering matrix containing information about n objects with p features.

$$F = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}.$$

Since we are dealing with a sample (instead of a random variable) and because the objects are rows of the input data matrix, it's reasonable to perform the following steps:

1. Find the sample covariance matrix from the equality

$$\Theta = \frac{1}{n}\left(F^T \cdot F\right).$$

2. Find the eigenvalues λ_i of the matrix Θ, $i = \{1, 2, \dots, p\}$.

3. Find the orthonormal eigenvectors φ_i of the matrix Θ, which correspond to the eigenvalues λ_i.

4. Select the necessary number of principal components. For MSE to be the smallest, it's reasonable to take the eigenvector of the matrix Θ, which corresponds to the largest eigenvalue of this matrix, as a loading vector of the first principal component. Similarly, the eigenvector, which corresponds to the second largest eigenvalue, should be taken as a loading vector of the second principal component, and so on.

5. Find the new coordinates (or scores) of the objects in the selected basis by multiplying
$$Z = F\Phi,$$
considering that the coordinates of the selected eigenvectors are the columns of the matrix Φ, and that the first column Φ contains the coordinates of the loading vector corresponding to the largest eigenvalue of the matrix Θ, and that the second contains the coordinates of the loading vector corresponding to the second largest eigenvalue, and so on.

Note that the values of the eigenvalues λ_i for the selected PCs are equal to the sample variances of the ith scores.

4.5 Reconstructing Features Based on Principal Components

Principal component analysis is often used to reduce the number of features. However, we can also encounter the opposite problem, that is, the problem of reconstructing initial features of objects. Clearly, if the number of PCs is smaller than the dimensionality of the initial space of objects, then, due to an error, we will lose some information in the process. It's the case when we encounter an outlier after constructing the principal components. When the initial features are reconstructed, we can carefully examine the object and analyze the 'almost initial' features to reveal deviations from the tendency.

Well, how the reconstruction is performed? Let Φ be a matrix, which columns correspond to the coordinates of normalized eigenvectors (loading vectors). Then,

$$Z_{[n\times p]} = F_{[n\times p]}\Phi_{[p\times p]},$$

and the matrix of scores has the dimensionality $[n \times p]$. Then, the old centered coordinates are reconstructed without losses by multiplying the entire equality on the right by Φ^T. Thus,

$$Z\Phi^T = F\Phi\Phi^T = FE = F,$$

since, due to the orthogonality of Φ, $\Phi\Phi^T = E$ is a unit matrix. However, usually the number of PCs we keep is less than the dimensionality of the initial space.

By keeping k of them, we obtain the matrix Φ of size $[p \times k]$ and the score vector

$$Z_{[n\times k]} = F_{[n\times p]}\Phi_{[p\times k]}$$

of size $[n \times k]$. After we multiply the equality on the right by Φ^T, we see that, although the product $\Phi\Phi^T$ is of size $[p \times p]$, it is not a unit matrix. Therefore,

$$Z\Phi^T = F\Phi\Phi^T = \widetilde{F},$$

where $\widetilde{F}$ is a matrix with the coordinates of centered initial objects reconstructed approximately.

The reconstruction is approximate because we lose information when the dimensionality of space reduces. Fig. 9 makes it clear. When we match the object x_1 having two features (x_{11}, x_{12}) to just one, the reconstruction of the old features based on the knowledge about the first principal component leads to an error in features $(\widetilde{x}_{11}, \widetilde{x}_{12})$.

To reconstruct the initial features, we should add the calculated mean values $\overline{X'_j}$, $j \in \{1, 2, ..., p\}$ to those obtained after the reconstruction (because the input data are not centered). Let's introduce the matrix $\overline{X}$ of size $[n \times p]$ of the following form:

$$\overline{X} = \begin{pmatrix} \overline{X'_1} & \overline{X'_2} & \dots & \overline{X'_p} \\ \overline{X'_1} & \overline{X'_2} & \dots & \overline{X'_p} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{X'_1} & \overline{X'_2} & \dots & \overline{X'_p} \end{pmatrix}.$$

Then, the final reconstruction (or approximate reconstruction) for the scores Z can be written as follows:

$$\widetilde{F'} = Z\Phi^T + \overline{X},$$

and, if Φ is of size $[p \times p]$, then $\widetilde{F'} = F'$, where F' is a matrix of initial objects before centering.

4.6 Another Look at the Example

Let's use the sales bonus case as an example to show how to find the principal components using a covariance matrix. We will also find out what part of the information is lost when only one principal component is used and what the reconstructed data is alike. Recall the problem setting.

The input data is given in the table.

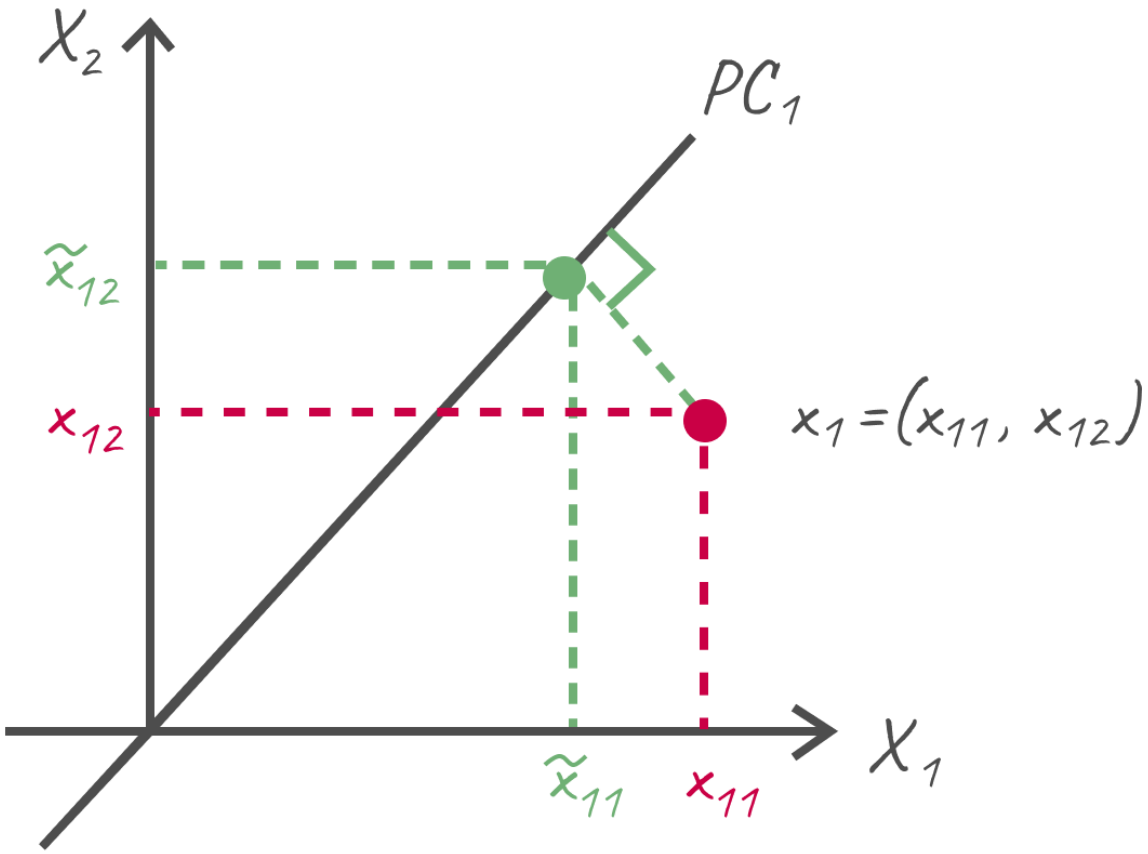

Figure 10: Reconstruction geometry.

Employee (x_i)	**Premium** (X_1), **pcs**	**Economy** (X_2), **pcs**
1	9	19
2	6	22
3	11	27
4	12	25
5	7	22

The feature mean values are as follows:

$$\overline{X_1'} = 9,$$

$$\overline{X_2'} = 23$$

The next table contains the centered input data.

Employee (x_i)	**Feature** X_1	**Feature** X_2
1	0	-4
2	-3	-1
3	2	4
4	3	2
5	-2	-1

Let's write down the matrix F:

$$F = \begin{pmatrix} 0 & -4 \\ -3 & -1 \\ 2 & 4 \\ 3 & 2 \\ -2 & -1 \end{pmatrix}$$

and a sample covariance matrix Θ:

$$\Theta = \frac{1}{n}\left(F^T \cdot F\right) = \frac{1}{5}\begin{pmatrix} 0 & -3 & 2 & 3 & -2 \\ -4 & -1 & 4 & 2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0 & -4 \\ -3 & -1 \\ 2 & 4 \\ 3 & 2 \\ -2 & -1 \end{pmatrix}.$$

After matrix multiplication, we get:

$$\Theta = \frac{1}{5}\begin{pmatrix} 26 & 19 \\ 19 & 38 \end{pmatrix}.$$

Let's find the eigenvalues of the matrix. Recall that we find eigenvalues given that the determinant of the difference between Θ and λE is zero:

$$\left|\Theta - \lambda E\right| = 0,$$

where E is a unit matrix. In this case, we get the equation

$$\begin{vmatrix} \frac{26}{5} - \lambda & \frac{19}{5} \\ \frac{19}{5} & \frac{38}{5} - \lambda \end{vmatrix} = 0.$$

Using the rule of finding the second-order determinant, we get:

$$\left(\frac{26}{5} - \lambda\right)\left(\frac{38}{5} - \lambda\right) - \left(\frac{19}{5}\right)^2 = 0.$$

Solving this equation as a quadratic, we find λ_1 and λ_2:

$$\lambda_1 = \frac{32 + \sqrt{397}}{5},$$

$$\lambda_2 = \frac{32 - \sqrt{397}}{5}.$$

Since $\lambda_1 = \max(\lambda_1, \lambda_2)$, the corresponding normalized eigenvector will be the loading vector of the first principal component. Let's find it. Using the definition:

$$\Theta\varphi_i = \lambda_i\varphi_i,$$

we obtain:

$$\begin{pmatrix} 26 & 19 \\ 19 & 38 \end{pmatrix} \cdot \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \end{pmatrix} = (32 + \sqrt{397}) \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \end{pmatrix}.$$

Then, to find φ_1, we'll solve the system of equations:

$$\begin{cases} (-6 - \sqrt{397})\varphi_{11} + 19\varphi_{21} = 0 \\ 19\varphi_{11} + (6 - \sqrt{397})\varphi_{21} = 0 \end{cases}.$$

The system has an infinite set of solutions. We take φ_{11} as a basic variable and φ_{21} as a free variable. Then, given that $\varphi_{21} = 1$, we get $\varphi_{11} \approx 0.733$. Recall that the length of the loading vector of the principal component must be equal to unity. Therefore, we will perform the normalization by dividing each coordinate by the vector length:

$$\varphi_{11} \approx \frac{0.733}{\sqrt{1^2 + 0.733^2}} \approx 0.591,$$
$$\varphi_{21} \approx \frac{1}{\sqrt{1^2 + 0.733^2}} \approx 0.807.$$

Then, if $Z_1 = F\varphi_1$, we get the following expression for the first PC:

$$Z_1 = \begin{pmatrix} 0 & -4 \\ -3 & -1 \\ 2 & 4 \\ 3 & 2 \\ -2 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0.591 \\ 0.807 \end{pmatrix} = \begin{pmatrix} -3.226 \\ -2.580 \\ 4.409 \\ 3.387 \\ -1.989 \end{pmatrix}.$$

Let's find the fraction of variance explained in the case when we keep only one principal component.

$$\delta_1 = \frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{32 + \sqrt{397}}{(32 - \sqrt{397}) + (32 + \sqrt{397})} \approx 0.811.$$

We are going to solve the opposite problem of reconstructing the initial features using the first principal component. Then,

$$\widetilde{F}' = Z\Phi^T + \overline{X} =$$

$$= \begin{pmatrix} -3.226 \\ -2.580 \\ 4.409 \\ 3.387 \\ -1.989 \end{pmatrix} \cdot \begin{pmatrix} 0.591 & 0.807 \end{pmatrix} + \begin{pmatrix} 9 & 23 \\ 9 & 23 \\ 9 & 23 \\ 9 & 23 \\ 9 & 23 \end{pmatrix} = \begin{pmatrix} 7.093 & 20.397 \\ 7.475 & 20.918 \\ 11.606 & 26.558 \\ 11.002 & 25.733 \\ 7.825 & 21.395 \end{pmatrix}.$$

If we round the data reconstructed using the first PC to integers, it will be similar to the input data:

$$F' = \begin{pmatrix} 9 & 19 \\ 6 & 22 \\ 11 & 27 \\ 12 & 25 \\ 7 & 22 \end{pmatrix}, \quad \widetilde{F}' = \begin{pmatrix} 7 & 20 \\ 7 & 21 \\ 12 & 27 \\ 11 & 26 \\ 8 & 25 \end{pmatrix},$$

where F' is the input data, $\widetilde{F}'$ is the reconstructed data.

5 PCA Application Examples

5.1 A Visualization Example

Let's consider an example of PCA application to identify the outliers and find ways of splitting the objects into the groups. The input data will be the dataset of Latin alphabet images[2]. The data consists of randomly distorted pixel images of 26 capital letters in 20 different fonts. The example is shown in fig. 10.

Figure 11: The example of distorted letters.

Each object has 16 features. At first, the features described the statistical characteristics of the pixel distribution (horizontal and vertical coordinates of the

[2]https://archive.ics.uci.edu/ml/datasets/Letter+Recognition

center of the smallest rectangle containing all the painted pixels, the width and height of the rectangle, the total number of painted pixels, and so on). Next, the features were scaled so that they could take integer values from 0 to 15.

To demonstrate the idea, we're going to choose from the entire dataset only those objects that correspond to the letters A, B and C and visualize the data using 2 principal components. The results are shown in fig. 11.

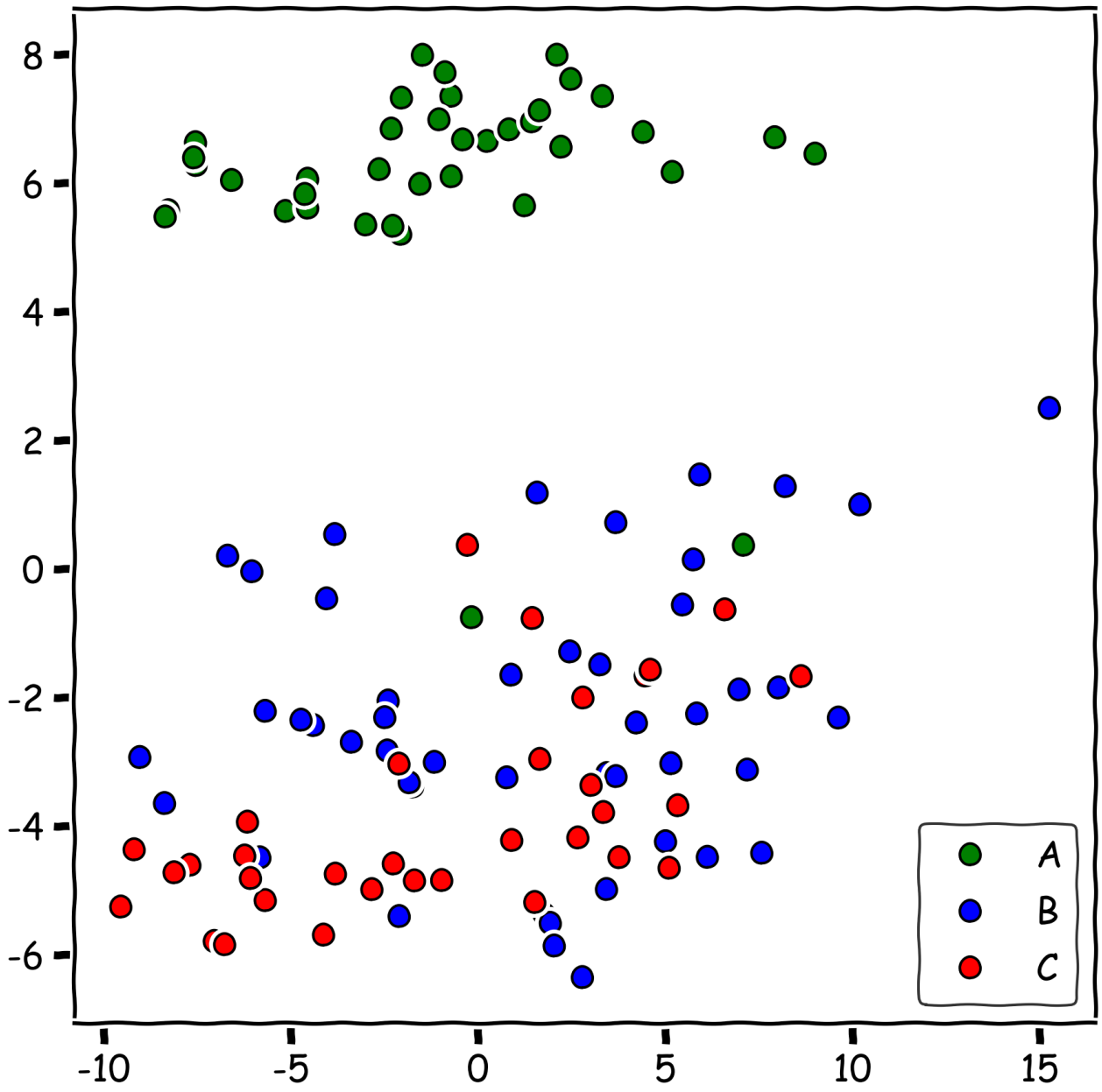

Figure 12: Visualization based on the first two PCs.

Look at the obtained image. As you can see, the objects are well split into groups, in particular, by the letter A. There are obvious outliers. We can clearly see two green points corresponding to the letters A and being found among the blue and red ones corresponding to B and C.

This leads us to a reasonable question. How much information will we have if we keep only two PCs? Let's create a graph reflecting the relationship between the fraction of variance explained and the number of principal components (fig. 12).

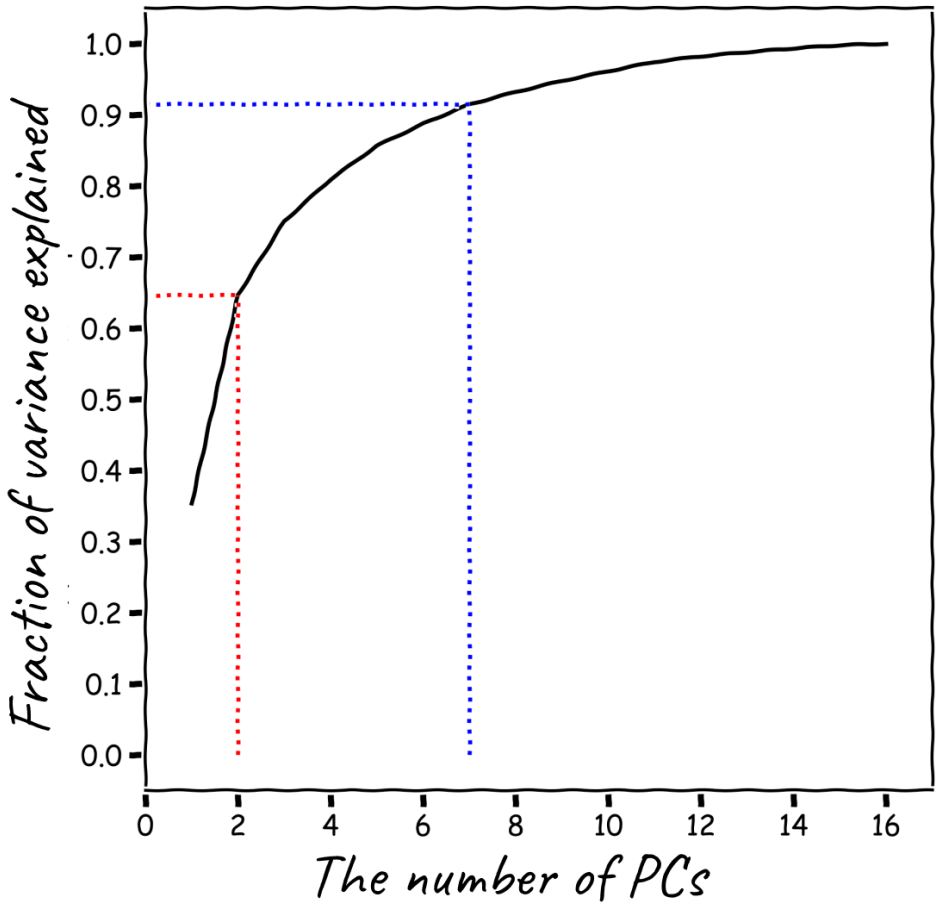

Figure 13: Defining the number of principal components.

If we keep only two PCs, we will preserve about 65% of the information. If we increase the number of PCs up to seven, the fraction of variance explained will be more than 0.9, and we will keep more than 90% of the information.

5.2 An Example of Image Compression

Another example of PCA application is image compression. To lower the amount of data, PCA and then image reconstruction are used. However, this comes with losses.

Let's apply the method to the well-known dataset containing the images of hand-written digits[3]. The images can be divided into 10 classes where each class is a number. Each image has the size of 8×8, which corresponds to 64 attributes.

[3]https://archive.ics.uci.edu/ml/datasets/Optical+Recognition+of+Handwritten+Digits

Each attribute takes integers from 0 to 16 (grayscale from white to black). The image example is shown in fig. 13.

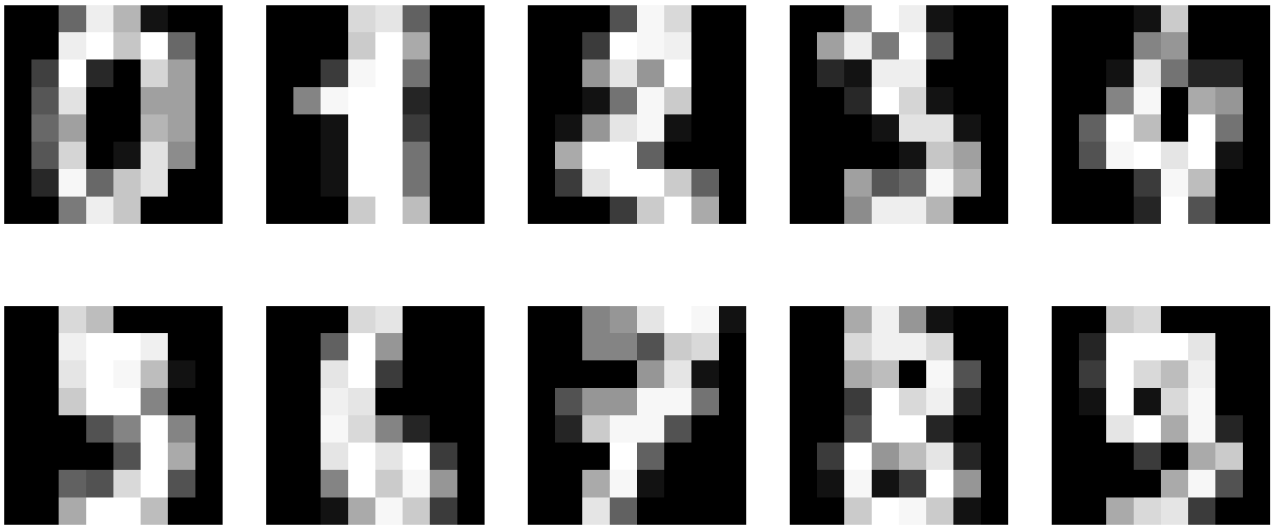

Figure 14: Initial images.

Note that the input data is not of high quality. Thus, the reconstruction will make the picture even worse. First, we need to determine the number of PCs we will consider while reconstructing the input data. Let's create the graph reflecting the relationship between the fraction of variance explained and the number of principal components. It is shown in fig. 14.

If we use less than ten principal components, we'll lose more than a quarter of the input information. At the same time, it makes no sense to keep more than forty PCs. Let's reconstruct the data for the cases of 2, 5, 10, 20, and 40 PCs. They are shown in fig. 15. On the left, you can see the number of PCs used (k) and the fraction of variance explained (δ). With 64 principal components, the images are reconstructed without any losses, and they match the originals. As we have assumed based on the graph reflecting the relationship between the fraction of variance explained and the number of principal components used, this case requires only 20 PCs to reconstruct the input data properly. When 40 PCs are used, the reconstructed data almost doesn't differ from the original.

6 PCA and Variance

Principal Component Analysis is sensitive to the units of measurement of input data due to the sample variance. To make it clear, let's consider an example. Let the objects be roads. They have two features including road length (in meters) and the average number of road traffic accidents per year.

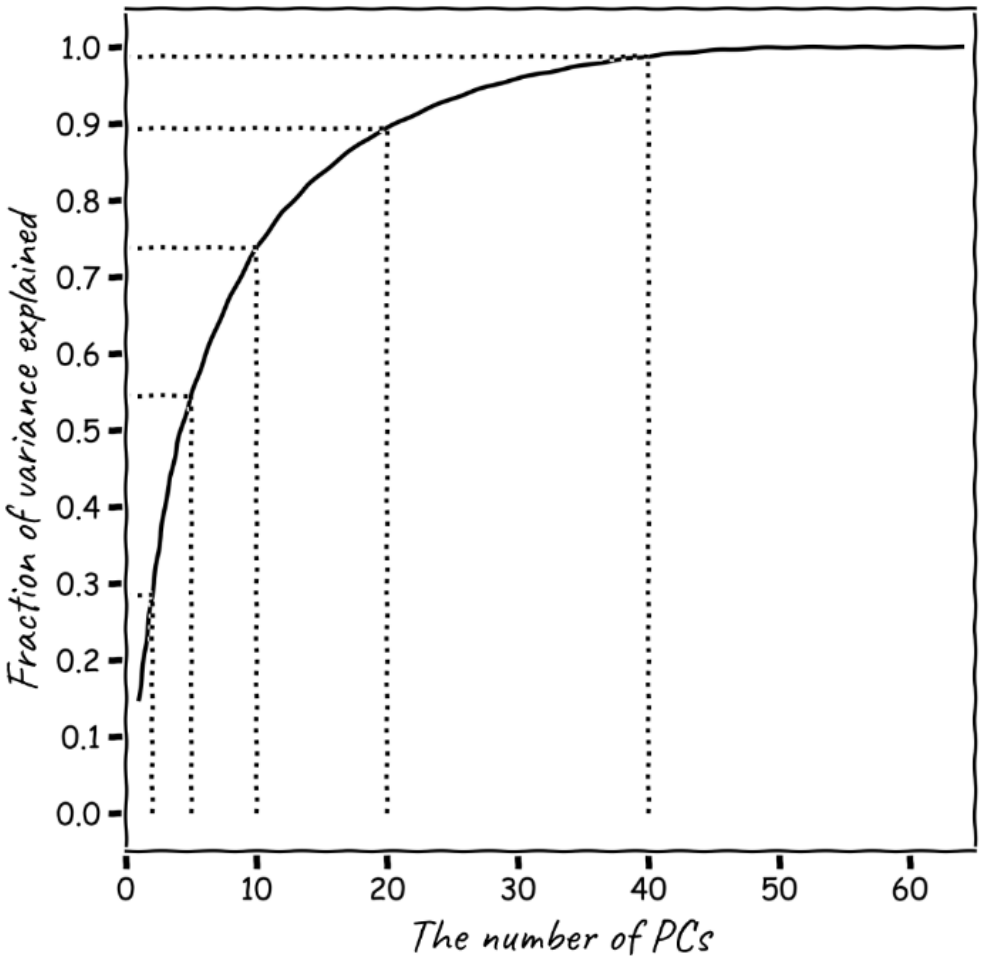

Figure 15: Defining the number of principal components.

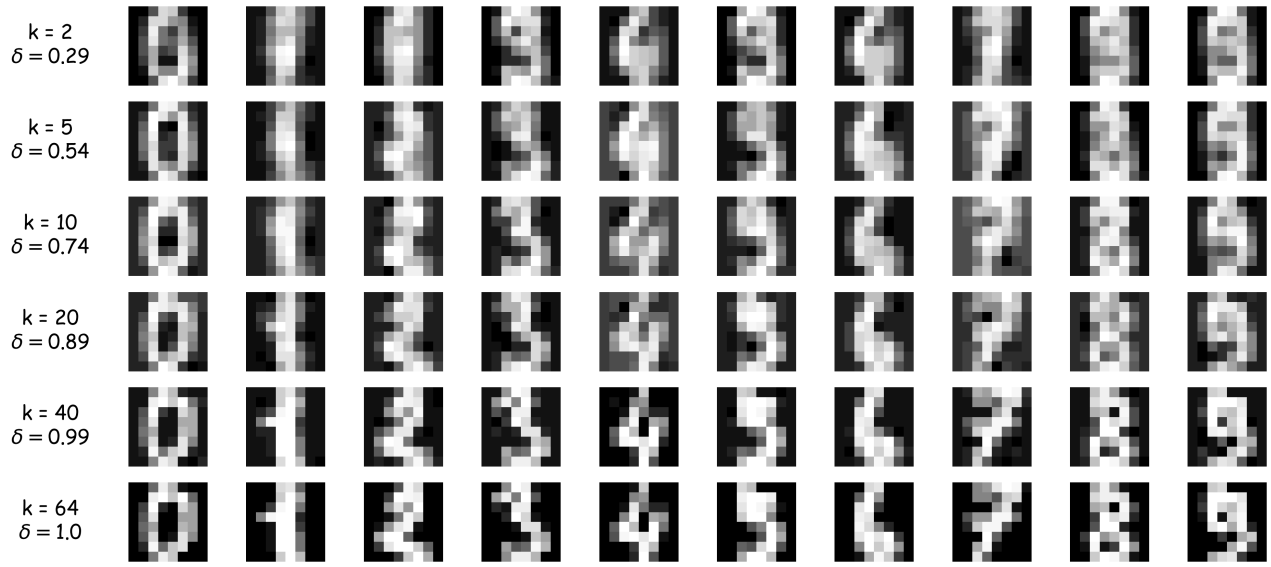

Figure 16: The reconstruction of the input data.

	E95	**M4**	**M10**
Length, m	37,700,000	1,517,000	697,000
Average number of accidents, pcs	345	84	51

If we are using the Euclidean distance, one-meter difference in the first coordinate will make the same contribution as one-accident difference in the second, which is not correct. Recall that, while finding principal components, we are also looking for the direction of the largest spread.

$$S^2_{\text{Length}} \approx 2.5 \cdot 10^{12},$$

$$S^2_{\text{Accidents}} = 25,941.$$

When finding the first principal component, the sample variance will make little difference to the accidents, and the direction of the PC will be just slightly different from the direction of the axis corresponding to the length. However, the average number of accidents is an important characteristic because it shows how the initial objects differ from each other as you can see from the table. To fix this, we can use some kind of normalization so that the features are comparable. For example, if we use linear normalization, we get the following values:

	E95	**M4**	**M10**
Length	1	0.27	0
Average number of road traffic accidents	1	0.11	0

Now the sample variance in each case is comparable:

$$S^2_{\text{Length}} \approx 0.27,$$

$$S^2_{\text{Accidents}} \approx 0.3.$$

7 Conclusion

To sum up, principal component analysis is an effective technique for dimensionality reduction of the initial feature space. It is often used to visualize the input dataset. In particular, the first two or three principal components are constructed to visualize the input data as the objects in the plane or space. Moreover, PCA has many applications in different domains, for example, image compression, image noise reduction, bioinformatics, and so on. It is important to consider that features can have different areas of change or correspond to the various characteristics of objects. Thus, proper PCA requires input data normalization.