# Corrosion, Environmental Degradation and Surface Engineering

**Prof. Harish Hirani**
**Department of Mechanical Engineering**
**Indian Institute of Technology, Delhi**

## Lecture – 26
## Principle component analysis (PCA)

Hello and welcome to the twenty-third lecture of our course on Corrosion and Environmental Degradation in Surface Engineering. In this session, we'll be focusing on Principal Component Analysis (PCA). Although we've touched on this topic in two previous lectures, I felt it necessary to dedicate an entire lecture to PCA to provide a more comprehensive understanding. This is particularly important because, with the increasing processing power of modern computers, we can now analyze vast amounts of data in a very short period of time.

Earlier, we did not have this kind of facility. Therefore, we were limited to relying solely on data graphs. We assert that the decision-making process requires accessible data. Now, we want to really analyse much data and take a quick decision based on some sort of mathematics, or may be statistical methods, and PCA is one of the statistical methods that we are going to describe. So, in increasing processing power now, it has made possible to utilise a vast amount of data, which is what I mentioned.

In the past, we lacked the computational power needed to analyze large datasets effectively, which often led to data being stored away in archives due to limited processing capabilities. As a result, much of this valuable information was underutilized. However, with advancements in computational power, we now can thoroughly analyze these datasets. It's important to emphasize that by making better use of existing and historical data, we can significantly improve future decision-making processes.

How is it possible to buy an exploration of hidden connections? Whenever you want to go for more and more innovation, we actually need to learn what mistakes were made earlier, or may be there are a number of data analyses, and those data find out the relation. So, we can enhance that knowledge instead of repeating it, or we may only be delta-incrementing it; we want a really quantum jump. So, that is why these methods, or these kinds of methods, are important. Now, another thing comes: whether we really understand each data from a physics point of view may not be easy to understand, but at least it will give something different. So, that is why I say that even if a physical meaning is opaque or not very clear to us. What data we are getting is a combination of observed factors; this is what we call practical knowledge or observed data that are available. If we can combine or utilise this data, it will provide a more effective foundation for discoveries.

So, we know very well the physics comes first and the math after that; however, in this case we are saying we have data, we develop the mathematics first, and then we try to figure out the physics behind that. So, it will be allowing us to discover newer and newer inventions, and then it will be very useful for society. Another thing the PCA can be carried out utilising the covariance, and here basically we are trying to differentiate the covariance and correlation in the sense we are trying to figure out what the relation is between the variables, which we selected sometime right. Sometimes we selected time; sometimes we selected time in hours, days, minutes, or seconds; what were they, and was there any relation with a scale factor also? Those things can be analysed with a PCA, and in fact, another Indian thing that we have been emphasising here is that we will be able to find new variables that can be adopted.

Typically, when conducting experiments, we predefine variables like RPM, load, or stress, assuming they are the key factors. However, PCA offers a different approach by analyzing data to identify which variables are actually important. This means PCA can reveal new variables that were not initially considered—these could be non-dimensional numbers or a combination of multiple factors into a single variable. This ability to identify adaptive variables is particularly valuable, as the relevant variables may change as the dataset evolves. The term 'principal component' refers to the main component that has the most significant impact on the overall process. This adaptive approach is a departure from traditional methods, allowing us to extract key variables directly from the data.

In the past, we relied on predefined basis functions and variables. However, this type of analysis, like PCA, can effectively reduce noise in data. While frequency analysis and time-domain methods are common, PCA offers a powerful alternative. With PCA, we can consider many factors simultaneously without worrying about complexity. To do this, we begin by standardizing all variables, ensuring they have a mean value of zero and a unit standard deviation or variance. These steps are crucial when starting a Principal Component Analysis. We'll also discuss some limitations of PCA later in this lecture.

PCA is not without its challenges; it comes with several limitations, which we'll address in this lecture. When applying PCA, there are specific steps to follow. First, as mentioned earlier, it's crucial to standardize the range of variables. This process assumes that we are dealing with continuous variables, even if we're actually working with discrete ones—treating them as continuous allows us to use standard or normal distribution. Once the variables are standardized, the next step is to create a covariance matrix, which helps us understand the relationships between the variables.

If we have four variables, our goal is to understand the relationships among them—for example, the relationships between factor 1 and factor 2, factor 2 and factor 3, and so on. This analysis requires the creation of a covariance matrix. As explained earlier in this course, all students will have access to MATLAB, provided either by the university or the relevant company. In MATLAB, you can easily calculate the covariance matrix with a simple command. Once we have the covariance matrix, the next step is to determine the eigenvectors and eigenvalues, which are crucial for identifying the principal components. We'll go over the formulas, but MATLAB also allows you to compute these values directly with a single command. Using these results, we'll identify the key principal components, determine which variables and factors are most significant, and decide which ones can be discarded. After eliminating certain factors, we'll need to adjust and recast the data accordingly.

We will be rejecting a couple of factors, and we will be recasting the data, and it will turn out to be a new set of the data that will be available to us. So, this is a complete process. You see, in this case, standardisation is required because we are going to deal with different scales, and then what is our main aim that makes the mean equal to 0 and variance equal to 1 (mean = 0 and variance = 1). So, if I am dealing with a 1 nanometre scale, 1 mm scale, or 1 meter scale, all three can be analyzed simultaneously. So, it will not really affect the scale and will not affect the analysis for us because we are going to deal with a mean 0 and variance 1.

So, in the situation, whether I am dealing with a nanoscale, micromicron scale, or meter scale, the results will not affect that kind of scale. So, now and after this, what do we say? In the covariance matrix, we try to figure out what the association is between the pairs, whatever the variables we are choosing for pairs of the variables, and then we need to really interact, whether the correlation is a positive, correlation a negative, correlation a very strong, or a weak one. We need to really judge. So, this way we think creatively; we have so much data, but we are looking from different angles. Can I make it mean 0. Will I be able to make a variance of 1, or will I be able to find out the relation: is it a strong relation, is it a weak relation, is it a positive relation, is it a negative relation? Again, the theme here is only linear relations; if we want to cross a linear relation, we want to go for other relations. PCA will not be suitable.

The basic assumption in PCA is a linear relationship; all of the variables can be used to develop a new variable, and part of the summation will be there. So, the new variable will be basically a kind of linear summation of existing variables. Now another one, as I mentioned, that we need to figure out and find out the Eigen vector. So, we will be using Eigen decomposition, which is already in the methods available in MATLAB. Now coming to how do we select a principal component? That is why we say that we try to figure out the Eigen values and then try to arrange the Eigen values in a decreasing order.

We start by calculating the eigenvalues, ranking them from the highest to the lowest. Typically, the last few eigenvalues will be nearly negligible, so we only keep a subset of the principal components with the highest eigenvalues. For example, if we have five eigenvalues—let's call them λ1, λ2, λ3, λ4, and λ5—and λ1 is the highest, followed by λ2 and λ3, with λ4 and λ5 being very low, we can disregard the lower values. This means we'll focus on just three eigenvalues, effectively reducing the dimensionality of our data. What began as a five-dimensional problem can now be simplified to three dimensions, creating three new variables that are combinations of the original ones.

So, in this case, we are retaining a few principal components. This is what we say: this option reduces the dimensionality, which is a main aim, and moreover, it is keeping the most of the relevant information. So, it will try to minimise the noise while keeping a good signal with us, right? This will result in a reduction in dimension and an increase in the signal-to-noise ratio. Now, as I mentioned, there are some sort of limitations.

So, what we are saying here, PCA, is suppose that there is a linear relationship between the variables. If there is a non-linear relation, PCA cannot be utilised that way, and that is why the non-linear relation may not work well with a data set that exhibits a non-linear pattern. Therefore, if the data set exhibits strong non-linearity, PCA will not be effective for that purpose. However, the principal component analysis really gives a good understanding, great brainstorming, and a good way to think about whatever the data we have, and then if we are going at the new set of data or may be the new experiment, which we really require how to start those experiments, which variable will be really important. So, this data and this type of information can be obtained from a PCA. Now, I will just try to give a simple explanation of PCA, and this has already been done; the literature indicates the Duke University also on their website; there is a lecture on 15 principal component analysis; basically, they have tried to teach this topic in a very simple manner.

What they say is that we often have scattered data that is plotted on X and Y graphs, allowing you to see the dots, or perhaps empty or hollow circles, representing the data. Now, this data is a kind of a random data we are not getting very good information what kind of relation is there between X and Y. So, what they did was they applied PCA, and then after PCA they figured out instead of the U, may be say, U bar, some new axis, and then where that is, most of the data are aligned. So, and then orthogonal to that V, V will be having almost no data as such, or we saw it right. So, the orthogonal component kind of can be rejected, and then X Y can be plotted only along the Y U, of course. For plotting purposes, we can use an X Y, but at least now we are going to get a linear relation with much lesser variation.

The goal is to determine which axis, U or U', provides the best results based on the data we've collected. To do this, we calculate the covariance matrix from the standardized data, which reveals the relationship between the X and Y variables. The covariance matrix helps us understand how these variables are connected. To identify the principal component, or the best U' axis, we calculate the eigenvalues using the Eigen decomposition method. These eigenvalues are then sorted in decreasing order to determine which eigenvectors, or principal components, are most important.

To determine which eigenvalues are most significant, we compare them. For example, if we have two eigenvalues, $\lambda_1$ and $\lambda_2$, and $\lambda_2$ is significantly larger than $\lambda_1$, we would choose $\lambda_2$ and reject $\lambda_1$. This is the approach we'll take. Now, to illustrate this process, we'll go through a complete example and revisit these concepts with the help of some figures.

To start the PCA process, the first step is standardization. This is a common procedure taught in most statistical methods, and you may have encountered it in earlier classes. We begin by standardizing the data, which involves taking each value, x, calculating the mean of the data column, and then dividing the difference between the value and the mean by the standard deviation. The formula for this is:

$$Z = \frac{Value\ (x) - mean\ (M)}{Standard\ Deviation\ (Std.)}$$

So, this gives us the normalized Z value. Now, how do we calculate the mean? The mean value is found by taking the sum of all the terms and dividing it by the total number of terms. The formula is: $M = \frac{Sum\ of\ terms}{Total\ no.of\ terms}$ . How do we calculate the standard deviation? We start by taking each value, x, subtracting the mean, and then squaring the result. This ensures that both positive and negative deviations contribute equally. We then sum these squared differences and divide them by n - 1, where n is the total number of terms. The formula for standard deviation is:

$Std. = \sqrt{\frac{\sum(x-M)^2}{n-1}}$ ; $where\ M = mean$. However, in a MATLAB we do not have to calculate.

So, after normalizing the data, we can proceed to get the results. Let's consider an example. We've taken a table from the literature or an online source that illustrates this process. In this example, there are four factors, or variables, labeled F1, F2, F3, and F4, and five observations labeled 1, 2, 3, 4, and 5. It's important to note that the number of observations can vary; you could have, for instance, 2 variables and 50 observations. Here, we're constructing a matrix where each variable corresponds to a column, and the observations (such as 1, 4, 1, 4, 5) represent the values obtained for each variable.

Now, as I already explained, the standardisation stage is very important; it has to be done, and I also mention in MATLAB that it has n-value algorithms in this manner. So, we do not have to do too much, but we need to understand. We ensure that the variables with a larger variance do not outweigh those with a smaller range. What is the meaning of that? If you compare the F3 to the F4, the F3 has a higher value of 3, 6, 3, 1, 2 than the F4.

F4 has values of 1, 3, 2, 1, and 3. It's important to ensure that one variable, like F3, doesn't dominate just because of its values. To prevent this, standardization is necessary. As mentioned earlier, we could be working with scales ranging from nanometers to meters. Standardizing the variables allows PCA to focus less on the absolute values and more on the relative patterns and connections between them. This is why standardization is crucial in PCA—it ensures that the analysis isn't skewed by the differences in scale between the variables.

Now, each variable will have a 0 mean and unit variance. As we have already described, the standardisation is really required for the PCA. Let me again explain in a slightly different manner. We say standardisation stage: make sure that variables with the greater range or variable do not outweigh those with a lesser range and variable. So, for that purpose, we can take examples of the F3 and F4. You can see the F3; the values are on a higher side: 3, 6, 3, 1, 2. The value of F4 is less than 1, 3, 2, 1, 3.

So, with only half the values, if we skip normalization, F3 could end up having more weight than F4. That's why normalization is essential. PCA, when applied correctly, focuses less on the absolute values of the variables and more on the relative patterns and connections between them. Additionally, as mentioned earlier, it's important to ensure that each variable has a mean of 0 and a unit variance. This step leads to more accurate results. For illustration purposes, let's consider plotting F1 versus F2.

I don't see any clear pattern here. For instance, when F1 is 1, the corresponding F2 values are 5 or 4. Similarly, when F1 is 4, the F2 values are 2 or 4. These data points don't follow any discernible pattern. In situations like this, conducting a PCA analysis is essential to uncover any underlying relationships.

So that we achieve better results, a similar pattern (or lack thereof) can be observed in F3 and F4. For example, when F3 is 3, F4 is 1, and when F3 is 1, F4 is also 1. There's no clear pattern in these values. Additionally, when F3 is 6, F4 equals 3. Since these variables don't follow any specific pattern, PCA is particularly well-suited for analyzing this type of data.

To proceed, as mentioned earlier, we first need to normalize the variables. This involves calculating the mean and standard deviation from the data. When writing a MATLAB code, we can easily obtain these values directly. However, for completeness, I've included a small code snippet to demonstrate how to find these values manually.

The matrix is entered as follows: 1 5 3 4 1 5 3 4; then 4 2 6 3 4 2 6 3 ; then 1 4 3 2 1 4 3 2 again ; then comes 4 4 1 1 4 4 1 1 5 5 2 3 5 5 2 3. So, these are the matrix has been given here. Now coming to the mean value what is the mean value? We need to find out the mean value of F 1 we do not want a value mean value of whole matrix, here the mean value of variable F1. We are not talking about whole vary on a complete matrix.

To find the mean value of F1, we look at the first column of the matrix, which contains all the values for F1. Similarly, we can calculate the mean for columns 2, 3, and 4, corresponding to F2, F3, and F4, respectively. Once we have the means, we can use the standard deviation formula, which is built into MATLAB. This allows us to calculate the standard deviation for each column—first for F1, then for F2, F3, and finally, F4.

Now that we've completed this, we can move on to the standardization process. We are familiar with this formula, which we can then apply. We can find out this one by writing the column 1 minus the mean value of this column 1 and then dividing by SDM, which is a standard deviation. Next, we will focus on the second column, then the third column, and finally, the fourth column. Here the new term is coming at 3, and then what is the meaning of 3. However, first, when we may be, as we have not given a semicolumn.

When we input the matrix, it immediately displays a 5x4 matrix. Each column corresponds to a variable, and the mean value for each column is calculated, resulting in a 1x4 mean value matrix. For instance, F1 has a mean value of 3, F2 has a mean of 4, F3 has a mean of 3, and F4 has a mean of 2. Moving on to the standard deviation, each column (or variable) also has its own standard deviation.

For example, the standard deviation for F1 is 1.8708, while for F2, it's 1.2247. Interestingly, F1 and F3 share the same standard deviation, suggesting that F1 and F3 may have a similar impact on the overall process due to their identical mean values and standard deviations. This observation warrants further analysis.

After this, we calculate the Z-scores and create a normalized matrix (let's call it Matrix 2). You can see significant variations in the values, such as -0.1, 0.067, 0.535, -1.067, and so on. If you calculate the mean and standard deviation for each column in this new matrix, you'll find that the mean should be 0 and the standard deviation should be 1. For instance, the sum of values like -1, +1, -1, +1 equals 0, confirming that the mean is indeed 0.

Additionally, we typically round these values to three decimal places, which is why the values in Matrix 2 are displayed with three decimals. The format used can be adjusted using MATLAB’s formatting options, such as 'short g' or others, depending on the default settings or specific needs.

In MATLAB, we can use a short fixed decimal format or scientific notation, depending on the specific requirements of the program. Now that we’ve normalized the variables, each one will have a mean value of 0 and a standard deviation and variance equal to 1.

The next step is to calculate the covariance matrix. In a 2D space with variables x and y, the first element of the matrix will be the covariance of x with itself, which is equivalent to the variance of x. The matrix will also include covariances between x and y, y and x, and y with itself. We know that the covariance of x with y is the same as the covariance of y with x.

To calculate covariance, we take the deviation of x from its mean and the deviation of y from its mean, multiply these deviations for each data point, sum them up, and then divide by the number of data points minus 1.

The formula for covariance is: $Covariance = \frac{sum((x-(mean\ of\ x))(y-(mean\ of\ y)}{number\ of\ data\ points-1}$

This process will yield the covariance for the point matrix. If we replace x with y, the first column will represent the variance of y, meaning that self-variation will be calculated. For the normalized matrix z, the variance for column 1 is calculated by taking the square of z and dividing it by 4, represented as $\frac{z^2}{4}$ . As I mentioned, since there are a total of 5 data points, we need to divide by 4. The variances for variables 1 through 4 are calculated accordingly. The correlations between F1 and F2, F1 and F3, F1 and F4, as well as F2 and F3, F2 and F4, and F3 and F4, are also determined. These correlations are symmetric, meaning F3 and F1 will be the same as F1 and F3, F4 and F1 will be the same as F1 and F4, and so on. After calculating the complete matrix, it can be rounded to three decimal places if desired.

So, what we want is the word we can use to calculate a covariance matrix FC. However, if you want to utilise only the MATLAB code or MATLAB inbuilt algorithm, we will write down the new matrix name, but for the time being we are saying that FC, which has been calculated for MATLAB, is what the F C M round and covariance matrix are simple. I can write directly z, but we want to compare up to the 3 decimals. So, that is the 3; otherwise, you write only whatever the normalised matrix you have; you write down a COV of that matrix whose covariance matrix will be available, and that matrix has come something like this. So, this is what we wrote a program, and this is the word MATLAB has generated program or generated a matrix. You can see there is a complete matching, and then all diagonal elements need to be 1 because the 1 to 1 relation will be the same and that F1 will have the same relation with F1, and F2 will have the same relation with F2, which means being equal to 1 will not be having other things.

While in other cases it can be negative, it can be positive, it can be very low value, it can be very high value also. You look at here this is the point 0.218, which is the negative, that means reciprocal relation, and 0.535 is a positive relation in the both in the parameters will vary together. Now, in this case, in the second column, all 3 correlations are negative, except the diagonal element 1, and in this third column, the first 2 are negative, the third last 1 is a positive, while in this case there are 2 positive and 1 is a negative.

So, we are talking only about the off diagonal elements, and then what is the meaning of that? We say the type and intensity of the link, whatever the value that we are getting, or the link or value of the variables can be inferred from a signature, what is the negative sign or positive sign, and what is the magnitude of the covariance. Higher the value of the magnitude of covariance means it will have a greater impact; a low value may well have a lesser impact, right? And another mention is that positive covariance denotes that there is a tendency for both variables to rise or decrease, whatever value cov higher is okay. So, then both variables will rise when one rises, or maybe the second variable will rise when the one variable is rising. Another one is that we may also get a covariance value equal to 0, which indicates there is no relationship between two variables, which will be very useful. However, because the whole PCA is a development on the linear relationship, we cannot say that they do not have any relation; they may have a linear relation other than the linear relation.

We are 100 percent confident that there will not be any linear relation between those variables if the covariance is equal to 0. If the covariance is negative, one variable will naturally increase while the other will decrease. Conversely, if the value is positive, both variables will increase or decrease simultaneously, working in tandem with each other. Therefore, these conclusions can be drawn from a covariance matrix. I believe that by examining old data, whether it's archived in a data book or stored in a library, we can critically analyze these data and develop increasingly robust relationships.

The next step involves identifying the eigenvalues and eigenvectors. So, this is again a MATLAB code we can find out eigenvalue and eigenvector and has been mentioned shown results here eigenvector and eigenvalue. Now what is more important for us to figure out eigenvalues or maybe give a descending order or ascending order? However, we will prefer the descending order. You can look at the first eigenvalue, which is 2.3169, which is a very high value; the second eigenvalue is 1.164; the third eigenvalue and fourth eigenvalue are at the same 0.24247. So, if I try to figure out what the percentage contribution of these eigenvalues is, I can figure it out. Now the first variable, or the first F1, has the highest contribution of 58.9; the second variable has a slightly lesser contribution compared to the first, almost 50 percent, that is a 29.02. However, third and fourth have almost the same kind of contribution, which means if I reject this 12 percent 12% contribution, which is happening because of F3 and F4, at least I can get good 88 percent 88% results. ILet's focus solely on F1 and F2. ow I can reject one variable, I can say reject only F4, but values are more or less the same. If the value and then the F3 would have been like in a 10, 15, then I would have rejected F4, but here the both the values are almost the same and it is not very difficult to segregate which variable should be removed.

For the time being, we can effectively eliminate both F3 and F4. We say they almost have a 12 percent 12% contribution when first F1 and F2, which is a major contribution of 88 percent 88%, and that is what the way principle components are decided. So, in this case, the principle component for F, as is the F1 principle component, is F2. Therefore, principle components are the variables created by a linear combination of initial variables. Principle components are uncorrelated; if you try to plot F1 and F2, there will not be any relation among each other. We assert that we can compress or recover most of the information from the initial variables and re-plot it in these variables.

This implies that instead of focusing on x and y coordinates, I could consider a transformation, such as a 5 degree or 10 degree rotation. I have the option to alter the coordinate system, potentially resulting in a more favorable relationship and improved outcomes. That is what the meaning of this principle component analysis is: we try to figure out alternative axes that have a larger number of data points, and then we get more meaningful results. Now, we mention the word recast the data, and that is what the recasting data can be done. Now, if I assume that there are two principle functions, or may be factors or variables, and we find F1 and F2, that is also possible, we can do that. However, for the recasting initially only the whatever we calculated F1, F2, F3, F4, that means these all four, if I assume initially all four Eigen values are important, 100 percent (100%) variable are important, and then we still can recast the variables along those axes.

If we proceed with recasting the data along the principal components, we assume that all four values are important—for instance, one might contribute 58%, another 29%, and the third and fourth about 6% each. Since we consider all these contributions significant, we can recast the data by multiplying the normalized matrix with the Eigenvectors, resulting in a new recast matrix.

When examining this recast matrix, we see that the top three elements are negligible, while the fourth and fifth elements have some importance, though they are similar. If we decide to focus only on the two most significant components, accepting 88% of the total contribution, we can simplify our analysis by retaining only the first two columns of the matrix. This approach allows us to effectively recast the entire matrix without needing to consider all the Eigenvalues, focusing instead on the most impactful ones.

So, based on the percentage contribution, we should select the most significant components without losing the essence of the data. If we focus on F1 and F2, the first and second columns, and recast the data using only these variables, we can still obtain accurate results without significant information loss. This approach demonstrates that large datasets can be effectively reduced without substantial loss.

For example, rejecting F3 and F4 (or their corresponding Eigenvectors and Eigenvalues) might result in around a 12% data loss, but this is not significant. This highlights the importance of Principal Component Analysis (PCA), which allows us to reduce dimensionality while retaining most of the data's variance.

When we compare the new matrix, which has been transformed onto the plane of principal components, with the original normalized matrix, there's a noticeable difference. The values for the third and fourth components, which initially had high values, have decreased significantly—down to 0.09 and 0.14—compared to their previously high values.

Now, let's revisit an earlier example where plotting the data was challenging, particularly during the digitization and analysis phases. This example was covered in my previous lecture. Initially, we observed a scattered plot with about 34 data points (in red), but no discernible pattern emerged. We aim to apply PCA here to improve the results.

Previously, we attempted linear, polynomial, and exponential regressions. The linear regression yielded a correlation coefficient of 0.86, and I mentioned that we should not rely on any regression coefficient below 0.9. The polynomial regression was close to 0.9 but still not ideal. Therefore, it became necessary to explore whether PCA could improve the correlation and provide better results.

In this analysis, we consider only two factors: Factor 1 (possibly representing the electric corrosion period) and Factor 2 (possibly representing energy). We have 34 data points to work with in this case.

So, period has been mentioned very clearly up to the 98 days, of course. This red colour sign has been shown because we could not accommodate in one column. So, we say that 54.23 after that 51.78, and of course, these data are not available to us.

So, we digitized the data by extracting values directly from the graph. This approach, as I mentioned earlier, allows us to obtain data from existing literature, which we can then analyze using PCA to improve the system. In this example, we worked with 34 data points, with the last value being 98.09, representing the number of days. F2 represents the energy level.

In this case, we have only two variables and 34 observations, compared to the previous example, where we had 4 variables and 5 observations. Now, by applying the same methods, we can calculate the mean value, standard deviation, and normalized matrix for this dataset.

Again, here we use the word 3 and say that the variable will be up to the 3, or maybe say the decimal will be up to 3. So, this is what the matrix has come to, maybe say that the 2 by 3 by 4, and there are only two columns, and there are 34 rows in this case, and these are the values that have been given to us or that we could calculate from this point. Now if we try to go ahead with the covariance matrix again, we will follow the same procedure. In this case, covariance F1 versus F1 will be the variance, and F1 and F2 and F2 variance will be F2, while covariance F1 and covariance F2 and F1 will be the same. So, these two values will be the same, which is why we are saying that the variance 1, variance 2, F1 and F2 and Fc can be calculated using this one. Again, we have also calculated using MATLAB to get the same results, and then in this case, overall, this covariance matrix, which will be the kind of square matrix, depends on the variables. If the variables are 2, naturally, this matrix has to be 2 by 2.

So, the covariance matrix is a square matrix. For four variables, it would be a 4x4 matrix. We calculated this matrix using our MATLAB algorithm, and these are the values obtained. Now, we move on to calculate the Eigenvalues. As you can see, the first Eigenvalue is 0.069, while the second is 1.931, which is significantly higher.

Looking at the percentage contribution, the first Eigenvalue contributes only 3.45%, while the second Eigenvalue contributes a substantial 96.55%. This high percentage indicates that the first Eigenvalue can be neglected in this case, as shown in the bar chart where the 96.55% contribution stands out clearly, and the other value is negligible.

Next, we proceed to recast the data, and the results are displayed here. The first 17 data points are shown, followed by the next 17. A pattern is now visible in the recasted data. If we plot this, the initial plot appears on the left, while the new plot with the recasted data is on the right.

As you can see, the regression coefficient is now approximately 0.965. The earlier regression coefficient was something like 0.86. The beauty of this PCA lies in its ability to recast or find a better linear relation using this pattern. Now if instead of going ahead with a standard deviation, I mean the values to be plotted on X and Y, we find suppose the number of points may be this and slightly different variables in this case or maybe other valuable.

So, in this case, we plotted against the number of data points and found that this value has further improved. The value now stands at 0.97. So, even though the thinking is slightly different, maybe the different variables can be judged, or maybe we are trying to analyse 10 different resources, and then we can choose which one is giving the best results. This is where the power of PCA lies. So, we can say variables, or in this case, are negatively correlated, which is why the negative sign is coming.

These variables are negatively correlated, meaning that as one variable increases, the other tends to decrease. This is illustrated in the analysis. Additionally, the new line generated contains more data points clustered around it. Previously, the data points were more scattered, leading to a lower correlation coefficient. However, with the PCA, the new axis provides a clearer structure, making it easier to evaluate the data and track how observations vary. This improved relation is why the PCA yields better results. Our lecture began with this example to demonstrate how scattered data can be better understood through PCA. Can we really change the line, or maybe a curve-fit line in this case with the more number of points that can be accumulated or will come around that? So, this is what we have mentioned earlier you and you find here, while in this case you can see the old data the blue colour line, but the orange colour line is different, and when we come with another variable instead of going with normalised values instead of that we go number of points we are finding something different. So, this is a beauty of the PCA, and it should be utilised even though we have several pieces of available literature that can be, you know, gathered and utilised using the PCA. Thank you for attending this lecture. Thank you.