Hello guys. So we're going to start a new machine learning algorithm which is called as principal component analysis. And we also call it as PCA. And we also say it as dimensionality reduction. Now before we go ahead and understand what exactly is PCA, first of all we need to understand why should we use PCA or dimensionality reduction. So for this we really need to understand about a topic which is called as curse of dimensionality. Okay, so curse of dimensionality. And in order to make you understand this specific topic, let me take some examples over here. Let's say I have various models. Okay. So let's say this is my machine learning model M one. This is my machine learning model M two. This is my machine learning model M3. This is my machine learning model M4. And I still have some more machine learning models, so be patient. Okay. M5. And this is my machine learning model M6 okay, so let's consider I have this many machine learning models. Now let's say that I have a data set. And in this specific data set I have somewhere around 500 features. Okay. And whenever we say about dimensionality, this basically means we are talking about something called as dimensions. And whenever I say dimensions this is nothing. But this is features. Okay. So let's say um, this data set that I specifically have is to basically determine or find out the price of the house price of the house. So some of the features that I may have in this data set is like, uh, house size, right? Let's say number of bedrooms. Or it may be number of bathrooms, or it may be, uh, I may also have features like number of bathrooms. Right. I may have this many kind of lot of features. Okay. Now, obviously, you know, to basically find out the price of the house, what is the most important thing? You know, probably you may need to know the house size. You need to know the room size. You need to know the number of bedrooms. You need to know how many number of washrooms. Right. Let's say for the model M1, I have provided around the most three important features to train out of this entire 500 features. Okay, let's say after that for the M2 model, I provided somewhere around six features for training the model. Okay, similarly, for the model three, I have provided around nine features, let's say for training the model M3, right? And after that for the M4, let's say I have provided 50 features. Okay. Or let me just not make it as 50 right now. Let's say that okay. In this in this particular case I have provided probably 15 features. Okay. So let's say over here I have provided 15 features. And here now I have increased the number of features here. I have provided 50 features for training this particular model. And then probably here I have provided 100 features. And then here I have actually provided 500 complete features. Okay. Now what will happen when I'm providing this many number of features for every model? Okay. Now let's say this model M1, uh, I will be getting some accuracy. Okay. And uh, this accuracy is based on this three specific features. Right. So here let's say this is my accuracy one. So here I am specifically getting accuracy one right now after I train my model with this three features and probably train this particular model I'm going to get accuracy one. Now let's go to the next model where I have specifically used six features. And let's consider this six features. All are very, very important for determining the price of the house. Then what may happen is that my accuracy two will still increase, right? Uh, when compared to accuracy one. So here I can definitely say accuracy two will be greater than accuracy one. Then let's say M3. Uh, again when which whichever features we actually provided like this 15 features are still more important when compared to the previous six feature. So I may get an accuracy three, which may be more good at this point of time. Okay, now let's say when I probably went to model four here. Now you will be able to see that I'm providing 50 features now. Now in this features, there may be a scenario that some of them may not be at all so much important, right. And some may have more importance. Some may have very less importance. Some may have no importance at all. Right. So now when we will try to find out the accuracy here, you will be able to find out. The accuracy will keep on decreasing when compared to the previous accuracy. Now what will happen is that. Uh, in this scenario, you will be seeing that this accuracy three will be greater than this accuracy four. Okay. And I'll explain you why this specific thing will probably happen. So let's say this is my accuracy four. And here you can see that the accuracy four has decreased when compared to accuracy three. Then when I go to the next model, when I'm specifically providing 100 features here, the accuracy will still decrease. And then when we go to the next model where I've specifically provided 500 features, and this may include all the features that I've provided in the data set. Here, you'll be able to see the accuracy six that you will be getting. Still more decrease when compared to the previous accuracy. Now you may be thinking right why this thing is basically happening. This is the reason why we say curse of dimensionality. Now just imagine guys, it's just like over here in model M1, M2 and M3 when we were seeing that after training with, let's say an M3 model, we were training with 15 features, the accuracy was quite increasing over here, and we can consider that this all features were quite important for this particular model to perform well, you know, but when we went to accuracy four or when we went to probably this model four, you know here we have increased the number of features. Now this specific model need not require that many number of features to do the prediction. Over here you can just say that the model is over fitted right. The model is over fitted. So model should be given that many number of features through which the accuracy which should keep on increasing. But here you can see that the accuracy is decreasing. Similarly, after this particular step, when we go to the next step, you know, here also you could see that as soon as I provided 100 features. And obviously in this features there may be a situation that some of the features may not be at all useful, you know, but still it is model is basically getting trained on that. And lot of confusion actually happens within the, uh, model itself. Why? Because see, model is altogether, uh, inside. If you probably say, right, any machine learning algorithm, those are nothing but mathematical equations. Now with respect to mathematical equations, you know, as you keep on feeding new and new features, it will try to learn those features also, even though it is not that important. Right? So because of that, your model will get overfitted and it will definitely lead to a lot of confusion. And based on that, as the number of features gets increased, your model will start performing like it will stop performing well. You know? And from that point of time, your model performance will also degrade. Now here your second point is that your model performance will also degrade. Why? Because as the number of features are increasing, just imagine the number of dimensions are increasing. Your model performance will degrade because now the mathematical calculation will happen for that many number of dimensions. So because of that the model performance will also degrade. Okay. Let me give you a very simple example. Let's say you are a human being okay. Let's say uh, there is a person uh, I'll just consider there's a person over here. Let's, let's say and suppose, uh, I will just ask this person, okay, what is the cost of a house in this specific location? Okay. In location A. What is the cost of the house? Okay, then the person will probably understand some of the features. Okay. Which location this is okay. And let's say he does not have any other location, uh, any other features. So this location may be my feature one. Now here he may guess that, okay, you're probably your house may cost between 450 K to 500 K. Okay. Now you say that. No, I specifically want a three BHK apartment. Okay, now I have got a new feature. Okay, then he may say that. Okay, fine. Uh, this was for an average. He just thought that. Okay, this may be an average price for a two BHK apartment. Now, when he says three BHK apartment, then he may say that, okay, now your price may be rising between 600 to, uh, 500 to 600 K. Then he suddenly says that I need a house near a beach. Okay, now this particular parameter is added. Now you know that what will happen? The person will start thinking and this price will keep on increasing. Okay, then let's say after some point of time, the person demands increases. The other person who wants to buy the house, he says that I want a house near to a celebrity. Okay, near to a celebrity house. So let's say now this particular feature is also added. Now here again, the price will keep on increasing like this. It will keep on increasing. Now suddenly the person says, now I also want something like in uh, beside my house, there should be a lot of grocery shops. Now, this particular feature may not be that important to basically increase the price or not, but still, we are adding this particular feature for training our model right now. What will happen? You know, this may create an impact, but it will create a very less impact. Okay. Then suddenly you'll be saying that okay, how many number of schools are there surrounding my, uh, surrounding my house? Then suddenly one more feature will keep on getting added. Now, as soon as we keep on adding this many multiple things, this person who is about to say the price right. He will also get confused at one point of time, right? Because we are just over feeding him with multiple features. Okay, I want this, I want that, I want this. So now the person will still get confused and he will not be able to tell you the proper price. That basically means the performance of this particular person who is a domain expert in telling you the price range of the houses will decrease, and he may also not be able to tell you the accurate result. And this is the same thing that is happening when we train our model with many, many, many features. That is where your curse of dimensionality comes into picture. Now the question arises how do we prevent it? There are two different ways how to, uh, remove this curse of dimensionality. So two different ways. To remove. Curse of dimensionality. One is. Feature selection. Now, what does feature selection will be saying? That we will try to take the most important features, and then we'll try to train our model. That is what happens in the feature selection. The second technique, which we are going to discuss in this series is something called as PCA. And I'll not just write PCA, I'll just say dimensionality reduction. And in this series we are going to focus more on dimensionality reduction. And in dimensionality reduction. There are again many, many algorithms. We'll start with PCA. That is principal component analysis okay. So two ways to remove curse of dimensionality. One is through feature selection and feature selection. What we do is that we try to take the important features and then we train our model. Whereas in the case of dimensionality reduction, what we do, we do. And this process is basically called as feature extraction. Okay. Feature extraction. Now, feature extraction basically says that we'll try to derive a feature from a set of features. You know, where we'll be capturing much essence of the previous feature. Let's say that I have features like F1, F2, F3, and this is my output. What we'll do is that we'll try to derive. We will try to derive a feature like f one, and let's say we'll try to derive a new feature. This will be our D1 and D2 from out of this three features. And we'll use this features to basically find out our output okay. Then we'll try to see that. How do we derive this particular feature and all everything. In short we are extracting this new information. So that is the reason we say this as feature extraction. Super important topic. Okay, so I hope, uh, you got an idea about what is curse of dimensionality. In brief, if I tell you that as we keep on increasing the number of features, those features, uh, at one point of time when we are training those models, my model will get confused because I have so many features to learn. Uh, and because of that, the performance of the model will basically degrade. And I have also given you an example with respect to a specific person, right. If you keep on adding features and if you ask this domain expert what may be the price, then he may. He or she may also get confused with respect to this. Right? So how do we prevent, uh or remove curse of dimensionality. Uh, there is two techniques. One is feature selection and the other technique is something called as feature extraction and feature extraction. We are going to learn about the first technique which is called as dimensionality reduction. And in that we are going to learn about this algorithm which is called as PCA. Okay. In short we will be extracting some features from out of the previous features. So this was my original feature. And out of this we will be extracting some features like this. It may be in lesser dimensions. And obviously remember this will be in the lesser dimensions and we'll try to capture the essence out of all these original features, the the the same essence of variance. I can also say, uh, and then we'll try to derive this particular feature, which will be able to perform well when we are training our model. So yes, uh, in the upcoming video, we are going to now deep dive into what is principal component analysis. And we're going to understand the geometric intuition. So thank you I'll see you all in the next video.

In this section, we are going to discuss the differences between feature selection versus feature extraction. And this techniques, as you all know, is used in dimensionality reduction, which will actually helps us to reduce the number of features or to extract some important features from the older features that we already have. Now, before we go ahead and understand about this, we really need to understand why should we perform dimensionality reduction. So here I'm going to note down some important points. And this can also be an interview question. The foremost reason is to prevent the curse of dimensionality. Prevent curse of dimensionality. So that is the reason why we specifically use dimensionality reduction. Coming to the second point. Coming to the second point over here is that to improve the performance of the model, to improve the performance of the model. Now just imagine guys in your data set, if you have 100 features, that basically means 100 dimensions are there. Obviously to train the model, right. Because in every model, when the training happens, some mathematical equation is already present over there. Right? So those mathematical equation gets applied to all the 100 dimensions. Right. So because of that it may take more time for training of the model. So basically to improve the performance of the model we can also use dimensionality reduction. The last point that we are going to discuss is super super important to visualize the data. And at the end of the day, once we are able to visualize, we will also be able to understand the data. Now you may be thinking what exactly this is. Guys, we human being, you know we can at the max, we can see in a three dimension way. We can visualize anything in three dimension, right? That is in 3D. You can also visualize in 2D, right? We cannot visualize 4D or any dimension that are higher than 3D, right. Suppose if in my data set I have 100 dimensions, obviously I cannot visualize this. So in order to understand this data, what I can do, I can reduce the number of dimensions to 3D or 2D. And I can also see this particular information clearly in front of my eyes, so that we will be able to understand the data. Right. So if in any interview question, if someone asks you why do you you specifically use dimensionality reduction, you should definitely talk about this three points. One is curse of dimensionality, which I have already explained. Improve the performance of the model, right? Visualize the data. If you really want to visualize the data in 2D, 3D, or 1D mode, you can definitely use dimensionality reduction. And the major aim is basically to understand the data. Now let's go ahead and discuss about feature selection okay. So this feature selection is a process wherein it will help us to select the most important feature, which will actually help us to predict the output. Okay. Now in order to understand the feature selection process, let's consider that I have a feature which is called as X, and I have a feature that is called as y. Now let's say I have some data points over here. Something like this. Now. With respect to this. If I say when my x value is increasing, my y value is increasing, and if my x value is decreasing, my y value is decreasing. So here I can definitely say there is some kind of relationship right? In this particular scenario. That is, when the x is increasing, y is increasing, and when x is decreasing, y is decreasing. Similarly, I can have one more relationship wherein when my x is decreasing, my y is increasing, and when my x is increasing, my y is decreasing, right? So this is also another relationship that I can have between x and y. And in this particular case let's consider x is my input feature and y is my output feature. Right. With respect to x I need to predict what is y. Now whenever I have this kind of relationship, then here you will be able to see if I plot with respect to x and y axis. Here, you'll be able to see all the data points will be. In a linear relationship, right? When it is in a linear relationship, then it will probably follow this particular approach where x is increasing, y is increasing, and x is decreasing, y is increasing. And that is true in this particular case. When x is decreasing, increasing y is increasing right now. Similarly, in this particular case, if I probably want to draw a scatter plot, my scatter plot will look something like this with respect to x and y. Right now, if I probably try to see this, all the points will be in this inverse relationship. Now whenever we have in this inverse linear relationship, then you'll be able to see that it will follow this specific approach. In this particular case, it will follow this particular specific approach. Now obviously, you know that when you have this kind of linear relationship, then definitely x value will be very, very helpful in finding out the y output. And mathematically, we can also find out a way to quantify this relationship. And the technique that we specifically use in this is something called as covariance. Okay. So here if I really want to find out the formula between covariance of x comma y, the formula that is used is summation of I is equal to one to n x minus x bar multiplied by y minus y bar divided by. And here I'll write x of I y of I divided by n minus one. Since we are doing it for sample data, that is the reason we use n minus one. Now in this particular case, let's say if we are going to get a positive value, if it comes as a positive value, that basically means we are going to follow this specific relationship. If it comes as a negative value, then we are going to follow this specific relationship. So for this, whenever we get negative we are going to follow this. Whenever you get positive, we are going to follow this. That basically means the relationship between X and Y will have this kind of relationship. That is the linear relationship when my covariance is positive, when my when my covariance is negative, it will follow inverse relation linear relationship. And along with that let's say if my covariance is approximately equal to zero, that basically means there is no relationship between X and Y, no relationship, no relationship between x and Y, no relationship between x and y. So if I try to plot this in this kind of data scatter plot with respect to X and Y, I will be just getting a scatter plot, which may be in circular way, and here you can dot definitely find out, uh, the relationship between x and y. That basically means when x is increasing, whether y is increasing or not, or whether y is increasing or y is decreasing, whether x is increasing or not. Right. So whichever have this kind of highly positive relationship sorry covariance, highly positive relationship which we basically calculate through covariance. We can definitely say that those features are super important features. Right. So in this particular case I can definitely say that if I get a covariance is positive, I can definitely say x is a super important feature and it will definitely be helpful in predicting Y. But if I get the covariance between x and y is zero, I'll definitely say that there is no much relationship between x and Y, right? So I can definitely remove this specific feature. Also, right now let's understand. Along with covariance, there is also a technique which is called as correlation. And we basically say it as Pearson correlation. And Pearson. Correlation can be given as a formula of covariance of x comma y multiplied by standard deviation of x multiplied by standard deviation of y. Now, with the help of this, your correlation will be ranging between minus one to plus one. In this case there is no such range. It can be any positive value. It can be any negative value. But with the help of Pearson correlation coefficient you will be getting your value between minus one to plus one the more the value towards plus one. Now understand this is a super important point, the more towards the value of plus one. The more positive correlation it is. The more positive correlated it is. X and Y is okay, and the more the value towards minus one more negatively correlated it is right. Some same thing gets applied to minus one right? More negative correlated it is. And similarly, if my correlation between x and y is near to zero, that basically means no relationship. Okay. Now, this was an idea to talk about the relationship between X and Y. Now how this particular process is basically used in feature selection. This is just one of the technique guys. There are other techniques also in feature selection. But I just really want to give you a brief idea before we go ahead and understand PCA. Okay. Let's say I have a specific data set. And let's say this is my housing data set. In this data set. Let's say I have features like house size. The other feature that I have is something called as fountain size. Let's say that this house is are present within an apartment, and we are also going to take this as a property to predict the price of the house. Now, in this case, these are my. Independent features. And this is my. Output or dependent feature. So that basically means I'm going to use this two independent feature to predict the output feature. And obviously when this value changes this is also going to change. Now guys if you just apply common sense, you know that fountain size may not be a very important feature because obviously even though you have a bigger fountain or smaller fountain does not mean that your house house price will increase within an apartment, right within a number of apartments. Right? And obviously most of the bigger apartments, you know, a group of apartments will definitely have some kind of fountain. And all right, just to showcase, just to put some nice showcase. This fountain is basically used right now. If I go ahead and take each and every feature, let's say, let's say one of the feature, I have taken house size and I have plotted along with the price House. And let's say if I have found out this linear relationship, definitely this indicates that this house size is an important parameter to determine the price, because it has this linear relationship. And how do you calculate the linear relationship? Or how do you quantify the relationship between house size and price? You can definitely use something like covariance, and obviously you will either get a very high positive value or for some other use cases, you may also get some negative value. Some negative value okay. Or you can also apply correlation. That basically means you'll be getting a value between minus one to plus one. Now in this particular case, definitely you can say house size is an important property and obviously important attribute and just buy common sense. Also you can come to the conclusion. Now what about the feature? That is if I try to plot this feature that is fountain size. And probably price of the house. Now, in this particular case, let's say my plotting happens like this. And let's say that this is not at all that much important, right? Now here you can clearly see that there is no relationship between fountain size and price of the house. You can see that. Or here you can definitely see that even though the fountain size is increasing, the price of the house is stagnant almost in within this specific region. Right. So definitely we can conclude, come to a conclusion that the relationship between fountain size and price of the house will be very, very less or it will be approximately equal to zero. Or let's say it is ranging between 0 to 0.25. Okay. Not bad. Not that positively or negatively correlated. Right. So in this particular case, what you can do is that you can consider that fountain size is not that important feature. And you can drop this feature. Now this steps that we have performed, we have basically we are basically seeing this as a feature selection process. And here you can also make sure that you can apply covariance and find out how a correlation and find out that what is the what is the relationship between this fountain size and price of the house? So this is how you can come to a conclusion whether fountain size is important or not. And based on that you can basically drop it. Right. And this is what we usually do in the feature selection process. We use correlation and covariance to do it okay. Now let's go ahead and discuss I hope everybody got an idea about feature selection. Now what we are going to do is that we are going to discuss about a important topic, which is the next topic which is called as feature extraction. Now what does exactly happen in feature extraction? In feature extraction, let's say I have features like X1X2. And let's let's consider the same housing example. Now let's say I have something like house size. And let's say I have like number of rooms. And finally I have something called as price of the House. Now, in this particular case, you want to reduce your number of features from 2 to 1. That is what we are doing, right? We are in dimensionality reduction. Let's say that I want to reduce my two features into one feature. That is what we do. Now in this scenario you know that. Obviously, or let's say instead of house size, you know, I will just write. I'll just write room size. Okay. Room size. Now here I really want to perform some dimensionality reduction wherein from two features I want to convert this into one feature. So obviously I cannot use feature selection. Why? Because both these features are super important in predicting the output that is your price. Both these features are super, super important right? This these are my independent feature. And if I also try to find out the correlation or covariance between my input and the dependent feature here, you will be always be able to find out either very high positively correlated or high negatively correlated will be able to find that out. And it is a common sense question. Guys. This is obviously going to have a linear relationship with respect to this specific feature. On the top. We basically had fountain size, and I just try to show you an example by plotting it. It did not have a much a linear relationship. Now in this particular case, I obviously cannot drop one of the feature and just consider one feature because because here we can definitely see that there is some relationship between the independent and the dependent feature. So in this case, what do we do. In this case we perform something called as feature extraction. Now in feature extraction the core idea is that we take this two independent feature. We apply some transformation. We apply some transformation to extract new feature. And let's say this new feature is something called as house size. I'm just giving as an example. And my output feature is something called as price. Now instead of using this two feature that is room size and number of rooms, if I also get to know the information of the house size, then also I will be able to predict the price. So in short, from this two features, I am trying to derive a new feature or extract a new feature which is called as house size. Now just imagine for a domain expertise. If domain expertise is given with this two fields, room size and number of rooms, he or she will definitely be able to predict or tell you the price. Now, instead of giving this two features. If we just give one feature to the domain expert, then obviously he will. Also he or she will also be able to predict the price with some difference, right? With some difference, obviously some amount of information is lost over here. But again, domain expertise will also get some clear idea about like with the help of house size. Also he or she can definitely predict the price. So in feature extraction, what we are doing is that we are trying to extract a new feature from the feature that is already present. Okay. And this is how we basically reduce the number of features or dimensions. Okay. Usually here the example that I have taken only two independent feature. But guys in a real world scenario, you will be having 10 to 15 features and will try to reduce it to 2 to 3 features. Okay, now once I reduce this, I will obviously be able to visualize it in a proper way. I'll be able to see it and I'll be also able to understand the data. Now this is the idea behind feature selection versus feature extraction. So in this video I've covered so many things. Uh, first of all, why dimensionality reduction then? Why feature selection? What is feature selection? What is covariance. What is Pearson correlation? There is also another correlation which is called as Spearman rank correlation. Um, then we have taken an example with respect to the data set housing. Then I showed you that how we can go ahead with feature selection and how do we come up with feature extraction. Some transformation will happen. What kind of transformation will happen? What is the idea behind our geometric intuition behind PCA? Will discuss that in the next video. Okay, so yes, in the next section we'll discuss about this.

So guys, now let's go ahead and discuss about the geometric intuition behind the principal component analysis. Now obviously you know that principal component analysis is used for dimensionality reduction right. Dimensionality reduction. That basically means we are going to extract some features with respect to the older features that are present in the data set. Let's say I have features like size of the house and number of rooms. And this is basically the price of the house, which is my output feature. Right? This is my output feature. Now here let's go ahead and probably plot it. Plot a data points between the size of the house and the price of the house. So let's go ahead and plot some data points over here. And let's say this is my size of the house, and this is basically my number of rooms. Now let's say that I have some data points which looks like this, right? Let's say I have some data points which looks clearly like this. Okay. Now in this particular case, you can see that, uh, from this you are able to find out when the size of the house increases. The number of rooms are also basically increasing. And obviously it is a common sense question. It really needs to increase when the size of the house increases. You can assume that, yes, the number of rooms is also going to increase. Now let's consider that with the help of PCA. Okay I want to reduce this two dimension of features into one dimension. Okay into one dimension. Now, if I want to perform this, that basically means I just want to convert this two dimension into one dimension. Okay? This is what I'm actually looking with the help of PCA. That basically means instead of having two independent features, I just need to have one independent feature and try to basically create that independent feature, try to find out this one dimension or independent feature. Now, in this particular scenario, let's say that one way, obviously I've already taught you right, feature selection in feature selection. What you can do, you can take either one of the feature and you can ignore the other feature. And a simple way of converting a two dimension into one dimension is that what I will do over here is that I will make sure that I will plot all this point in my x axis. Let's say I'm just going to project all this point in my x axis. So here you will be able to see here I I'm getting the point here, I'm getting the point here I'm getting the point right. So once I do this you'll be able to see that I will be able to project all the points over here. Now when I project all the points, obviously I'm able to get all my data points in one dimension. So this data points that you'll be seeing, we are able to get it in one dimension. So here what I have done, I have basically converted this into from 2D to 1D. Right. And uh, one more thing with respect to this specific data points, which is super important to understand in PCA, is that if I probably see the first data point and the last data point, the area between this is basically my spread of the data points. Now, if my spread is huge, then what will happen? My variance will also increase. My variance will also increase. If my spread is increasing, my variance will also increase. Okay. So this is both are directly proportional to each other. Now what is the disadvantage by using this approach where I'm directly projecting in my x axis. So here obviously the size information is getting captured but the number of rooms information is getting lost. So this information that you had related to number of rooms why this information is getting lost. Because understand over here if I try to project from here to the first point also right here also you have some amount of spread. Now, as I said, if the spread is increasing, the variance is also increasing. And over here, because of this variance here also some information regarding number of rooms are there. Right. And you are directly skipping this information. You are neglecting this information when you use this specific approach, right, wherein you are directly projecting into one axis. So here with this approach, what is the major thing that is happening. There is loss of information, right. And in this case you are losing the information about the number of rooms. Obviously you created, you created from a two dimensional, you created a one dimension. And this is this is our or all your one dimension points. But from this, when you convert it from this 2D to 1D, you are losing much information from the number of rooms right about one specific feature. Now because of this, what will happen you once you lose all this information, you know your model may not perform well with respect to this specific, uh, predictions or uh, so here. In short, you are doing feature extraction, but you are doing feature extraction with lot of information that are lost. Right? So this in this feature extraction process you are losing lot amount of information. Right. So I hope you have got this specific idea now. You also have to take care when you do this kind of feature extraction. This information need not be lost that much. So how do you prevent it? So in PCA, to give you a brief idea, let me draw this particular diagram again. Let's say if this is my size and this is my number of rooms. And let's say these are my data points over here. Let's say these are my data points over here. Now in PCA, what you do is that you do some kind of transformation on this axis, on this x axis and y axis. What kind of transformation? You apply some mathematical equation. Uh, I'll again not talk about that mathematical equation right now, but uh, I'll just name it. You apply something called as eigen decomposition. Eigendecomposition on some matrix. On some matrix. Okay. We'll discuss about this. What exactly this is. But in short, you are going to apply some transformation and this is the specific transformation. And with the help of this specific transformation, you will be getting a new axis which will look like this. Let's say this will be a new axis which will look like this. Okay. And once you get this specific axis, what will happen is that then you will try to project this information over here. You will try to project all this information over here. Now when you try to project this specific information. Okay. Now suppose since you have two features, you will try to apply some transformation which will change the size axis to some other axis like size dash and one more axis will get created. Let's say uh, the other axis will be exactly perpendicular to this axis okay. And this will be your number of rooms. Dash. Number of room dash, and then you will be projecting all these points over here. So when you project all these points over here, here are your data points. Now in this projection and in this projection, what are the differences in the previous projection and in this projection. What are the differences. Now see in the x axis. If I consider this as my new x axis here you will be able to see that the spread is properly captured with respect to this specific points. And similarly, if I try to plot all these points on this new axis here, you will be able to see your spread that is going to get lost once we project over here will be very, very less. Will be very, very less. So here, what you are doing is that you are applying some transformation and you're creating a new axis like this size dash and number of room dash. And once you create this new axis, all you do is that you now just project all this point to this particular axis. And after projecting this point, you can see that maximum variance is getting captured for all the points over here. So here maximum variance is getting captured. Maximum variance is getting captured. And why this is better than this? Because in the y axis, you're not losing much information. You're just losing this amount of information. Right? This amount of variance, you're basically losing right when you project it on the x axis. Now with this transformation, you're able to convert your two dimension into one dimension. And you are making sure that much information is not lost. Much information is not lost. And how do you how do you come up with this specific point? Because see, in the y axis, because of this new dimension here the variance is less right. So even though you project this point over here, most of the variance is basically getting captured by your x axis. So much information is not lost if you compare the previous example here. Once you project it into the x axis, huge amount of information is getting lost because here the variance is quite high or spread is quite high. Here it is not that high right? So I hope you have understood. Our main aim in PCA is that to find out this kind of lines. Right. So after transformation, which is the new axis that is basically getting created now since I just had two dimensions. So after I do the transformation I basically call this as principal component analysis one. So PCA one principal component one. And we basically say this as principal component two. Suppose if you have three dimension, that basically means you will be having three principal components. Always remember PC1 will be capturing the maximum amount of variance and then PC2 will be capturing. The next maximum amount of variance. Similarly, if you have three dimension, then PC3 will be capturing the maximum amount of variation after PC1. Sorry, maximum amount of variance after PC1 and PC2. Right? So over here in geometric intuition behind PCA, our main aim is to basically find out principal components. Right. Like how we found out this. And at the end of the day, we really need to find out one line that should be able to capture maximum amount of variance. And once we take this PC1 and then we can convert this into a from a two dimension, we will just be taking up this specific point and will finally get converted into a one dimension point. Right. So I hope you got an idea about. What is the geometric intuition behind PCA? PCA that is principal component analysis. We'll talk more about what is eigen decomposition on matrix and what is this specific transformation. Uh, with examples I'll try to show you, but just understand that your main aim is to find out the principal component which has, after projecting all the points, it should have captured the maximum variance or spread. Okay. And why variance is important? Because variance basically talks about the data, how the data is basically spread. And with the help of variance you will be able to capture more and more information. So guys, the final goal of a PCA algorithm will be that to find out the principal components, uh, in such a way that maximum variance will be captured. So suppose if I have two dimension points then I will basically be having PC one, then PC two. Right. And we know that the variance with respect to PC one, after we, uh, you know, project all the points in that particular component will be greater than the variance with respect to the principal component. Two once we project it, suppose if we have specifically three dimensions, then here you will be able to see that we will be getting three principal components. Okay. That basically means like we will be having three different axis. One is PC1, PC2, PC3. And obviously the variance with respect to PC1 will be greater than the variance with respect to PC2. After we project the points and the variance with greater than PC three. Okay. So this variance will be following this specific approach. So let's say that if I have a task wherein I have two dimension points which is like this, and let's say I want to convert this 2D to 1D. So at the end of the day, what I will be doing is that the PC algorithm, uh, will be finding out the best principal component line. It can be this line, it can be this line, it can be this line, it can be this line, you know. And it will try to select the two best principal component lines. Principal component lines. And how we say that this is the best principal components line. Because once we project, let's say once we project this specific points on this line, right. We will be able to call this line as the best line only when the variance that is getting captured is maximum. Okay, so let's say this line has actually captured the maximum variance. So I will say this as pc1. Suppose if this line has captured the second maximum variance. Then I will basically say it as PC2. So. So this specific line will be my PC2. Right. And at the end of the day, if I'm converting this into 2D to 1D, then what will happen? I will basically be taking up all this specific points that I have actually plotted over here. Okay. So guys, the final goal will basically be to get. The best principal component. And how do we say that it is the best principal component which captures maximum variance, right. Which captures maximum variance? Okay, now let's say that if I want to convert a three dimensional point to one dimension. Now in the case of three dimension points, you know we will be getting three components. One is PC1, PC2 and PC3, right? And, uh, obviously it can be all these three lines, any of the best three lines. And you know that the variance of PC1 will be greater than variance of PC2 and this will be greater than variance of PC3. So since I really want to convert this into one dimension, what I will do, I will project all the points in PC1. And I will be taking all these points as my one dimension point, right? One dimension point. Let's say if I want to convert this three dimension into two dimension, then what I'll do, I will probably take PC1, and I'll also make sure that I'll project all the points in PC2, and I'll take this two information and combine it together to basically get my two dimension. Right. So this is what is the approach that we are going to follow. Again, let me repeat it. The major and the final aim of the principal component analysis is to find the best line after the transformation, which will be able to capture maximum variance. And the line can be existing anywhere in this space, right. And with with respect to that, based on the number of dimension that you want to convert, you can take that many number of principal components. Okay. So yes, uh, in the next video, we are going to see the mathematical intuition behind, uh, how this transformation basically happens. And that is where we are going to learn about eigen decomposition on a specific matrix. We'll talk about all what are the steps and how do you project it. We will also see that. How do you project the specific point on the specific line, and how do we make sure that the variance will be maximum. Uh, we'll be discussing about that. Okay. So yes, I will see you all in the next video.

So guys now let's go ahead and understand the maths intuition behind the PC algorithm. Now over here let's consider that I have this two axes x and y. And I've just plotted this point. And let's consider that my aim is to convert or reduce from two dimensions to one dimension. And as you know, the final goal of the PC algorithm is to find out the best principal component line after the transformation from this axis in such a way that maximum variance needs to be get captured. Now let's say for this example, I'm just going to consider this is that specific line which is my principal component one, because as soon as I plot all this point over here, you can see that maximum variance will get captured. Right. So that is the reason why we are selecting this as the best principal component line. Okay. Now what PCA is actually going to do in order to come to this conclusion, like whether this is the best or not, there are two important things. One is projections and the second thing is the cost function. That is related to the variance. Okay. So let's go ahead and discuss about both of them okay. So out of all these points let's do one thing. Let's take up one point. And for now I can just consider this as my point P one okay. Which is denoted by XNX1Y1 okay. This is obviously denoted by X1Y1 because y this is my x axis. This is my y axis. So let's say this is denoted by x one comma y one okay. So I can consider I can consider also this p as a vector okay. So let's say that this is a specific vector. And this p one is a vector. And let's consider that we also have a unit vector which is specifically given as u okay. So this is my unit vector. Now the first point let's go to the first point. The first point is all about projection. So let's say that if we really want to project this p one into this particular point okay, into this particular vector, then we will get something called as p one dash. Now we really need to find out what is this new projection. Because see over here our main aim is to capture the maximum variance. The maximum variance will only get captured when all the other points that you will be seeing in this. Right. Once we do the projection over here, once we do the specific projection over here, over here, over here, over here, then only we will be able to take up all this point and calculate the maximum variance, or we will be able to calculate the variance. Right. So that is the reason why we really need to project this particular point over here. So once we project let's say if you are trying to project P1 to P1 or to this particular unit vector that is UP1 dash, then this projection is given by an equation. Let's say if I'm saying projection of RP1 on you, right? If we write this specific equation, the equation will be given something like by. We can write it as p one. That is magnitude of p one. Sorry. Uh p one vector multiplied by u vector divided by magnitude of u. Okay. So this is the equation that is basically used with respect to the projection of a specific vector on a unit vector. Okay. And since this magnitude of U. Is equal to one, since this is a unit vector, and whenever we have a unit vector, the magnitude of that particular vector will be one. We can write the projection of. P1 on U is equal to nothing but dot product of p1 multiplied by u. So this is the entire thing that we basically get. So as soon as I do a dot product of P1, which is given by this coordinate x one and y one. And let's say this unit vector is given by the coordinate X2Y2. When we do the dot product with respect to p one and u, I am actually going to get p one dash. Okay, so this is nothing, but this is p one dash. Now this is with respect to the projection which everybody needs to understand okay. Now uh. Let's go towards understanding like this. When we do all the projection, let's say once we project P one, we will be getting p one dash. Let's say there will be another point. Let's say this is p two, then we'll be getting p two dash. Then we'll get p three dash. Then we'll get p four dash. Right. All the points. We will finally project it. Let's say PP0 dash. Like this. We have more all the points and we get finally p and dash. Now once we have all these points and remember whatever value we get from this particular dot product, it is a scalar value, right? And when we take up all the scalar values. So this all values that we are going to get after projecting it, this all will be a scalar value itself, right. Scalar value itself. Now once we have all these values this values are basically talking about the distance. Let's say what is p one dash. Basically talking about it is just talking about the distance from here to this particular axis. Right. So let's say if I'm probably calculating the distance from origin to this right. It is basically talking about this specific distance. Right p one dash. Similarly p two dash. Suppose if this is p two dash I'm basically talking about this distance right from here to p two dash. Now once I have the scalar values then it becomes easy for me to compute the variance. Right. Let's say uh these are all my points. So I'm just going to select all these things okay. And uh these are my points p0 p1 p2 p3 dash. So let's say I'm talking about this specific points. Uh, let me give some different notations so that it becomes easy for you to understand. So let's say this is X0-X1X2-X3-X4-. Like this I have x n dash. Okay. Now in order to compute the variance, I can simply use an equation that is, summation of I is equal to one to n, and here I will specifically be using. How do you calculate variance? You will be able to take x of I minus x of I or sorry. This will be x bar uh which will be the mean of x whole square divided by n. So if you use this equation you will be able to calculate the variance and understand here your main aim will be to. Your. Your main aim will be to find out the unit vector. The best unit vector which has the maximum variance. Your goal is basically to find the best unit vector. Best unit vector. And how do we say best. Because it captures which captures maximum variance. Maximum variance. So this is the example with respect to this. What we have actually discussed here. At the end of the day, we are trying to capture or we are trying to find out the vector which captures the maximum variance. Right. So this basically becomes our cost function. This becomes our cost function. So here I'm just going to write max of variance. We have to do we have to find the max of variance. So in short we have to find the best unit vector which captures the maximum variance. Right. So this is in short an understanding where you are taking care of two things. One is projection and one is the cost function wherein you are focusing more on getting the maximum variance. Right now we will try to understand. Now see, obviously we cannot keep on going ahead and selecting uh, different different units vectors and probably trying to find out the best vector in that way. So what happens is that there is a technique which is called as eigen decomposition, which we specifically say as eigenvectors and eigenvalues. Suppose. Okay this eigenvectors and eigenvalues. Because our main aim is, as said right. We really need to find out the unit vector right. Whether the unit vector should be this, this or this or this or this. Right. So how do I find out which unit vector we should basically select to capture the maximum variance. So for that we will be using something called as eigenvectors and eigenvalues. And there are some steps right that are steps which are involved in this. First of all the first step will be that I really need to find out the covariance matrix. Between features. Between features. And in order to find out eigenvectors and eigenvalues, what I need to do is that, uh, we this eigenvectors and eigenvalues will be. Eigenvectors and eigenvalues will be computed or will be found out. From. This covariance matrix. How that will discuss about it. Don't worry. And whichever eigenvector will be the largest one. When we say the eigenvector which is largest one, that basically means, uh, for the eigenvector where the eigenvalue is high. This eigenvalue basically talks about the magnitude of the eigenvector. Okay. So this eigenvalue is nothing. But this is basically the magnitude of the. Eigenvector. This will basically capture the maximum variance. Capture the maximum variance. And this is mathematically proved. Guys, I will not be proving you this thing, but, uh, here, according to linear algebra and with some of the mathematical equation, we can definitely prove this. So in order to why do we compute eigenvectors and eigenvalues? Because see, at the end of the day, I really need to find out the best unit vector such that maximum variance is captured. And we cannot keep on doing hit and trial. Right. So what we do is that in order to find out, we really need to go ahead and find out. Eigenvalues and eigenvectors sorry eigenvectors and eigenvalues. And the step is that first of all, we will go ahead and find out the covariance matrix between the features okay. Between the features. We'll go ahead and find out the covariance matrix. And then we'll try to find out the eigenvectors and eigenvalues. Uh the eigenvectors and eigenvalues can be found out by the simple equation. That is a lambda is equal to uh sorry a vector v is equal to lambda vector v. We'll try to understand this specific equation. What exactly this equation is. This is nothing but linear transformation. Of matrix. Of metrics. So we'll try to understand what exactly is this linear transformation of metrics. Uh, and then once we do this, you'll be able to get some eigen values. Whichever eigen vector has the highest eigen value, which is nothing but magnitude of the eigen vector, that specific vector will get selected if. Suppose this is the vector that has the highest magnitude, then this will get selected because mathematically it is proven that whenever we project on this specific line, maximum variance will get captured. Okay, so this is what we are going to do. In the next step we'll understand about eigenvalues eigenvector eigen decomposition from a specific matrix. Okay. So I hope you have understood till here. Now let's go ahead in the part two of the maths in depth intuition behind PCA.

In this video we are going to discuss about eigenvectors and eigenvalues. Now, already in my previous video I've spoken about it. What eigenvectors and eigenvalues can actually do? See, at the end of the day, we have to find the best principal component line which will be able to capture maximum variance. And in order to find this we will be using this eigenvectors and eigenvalues. Because in short whichever eigenvector like if. Suppose let's let's consider that we have uh we have some data points. Let's say I have some data points over here. And this data points. Let's say I want to find out the best principal component line in such a way that once we make sure we project this right in this right in this particular line, we should be able to capture the maximum variance. Now what is so amazing about this eigenvectors and eigenvalues. And we'll also be talking about linear transformation okay. And we also see it as eigen decomposition of covariance matrix. That basically means for this particular matrix we will try to find out the eigenvectors and eigenvalues okay. Now. Now, the most important thing about these vectors is that suppose let's say I have a specific matrix. Okay. I can have any specific metrics on this matrix. Or if I have a vector, let's say the vector is v okay. And this vector will have a different coordinates. Let's say if this vector on this particular vector let's say if I apply some linear transformation okay. What does this linear transformation basically mean? I'll give you an example that basically means on this particular vector, let's say if I apply a linear transformation with this specific matrix, then I will be getting one lambda value. And this lambda is nothing but my eigen value multiplied by the same vector. So here you can see that this basically means that. Suppose I have a vector with this particular dot. On top of this. If I apply linear transformation what is linear transformation basically mean let's say. I have my coordinate system and right now it is in this grid manner. Okay, let's say this is entirely all my data is in this grid manner, right? When I say linear transformation, you know, this entire data can be moved or this entire plane can be moved in different, different directions. Okay, I'll just show you one, one example directly. So always understand in order to compute the eigen values and vectors. So what we do is that let's say if A is my matrix okay. On top of it, if I apply this matrix, uh, on on the top of this specific vector, like if I apply the linear transformation on top of this specific vector, then I will be getting lambda which is my eigen value along with that same vector. So this is the equation that I will specifically be using in order to find out eigen values and eigen vectors. Okay. And from this equation, whichever eigen vector. Eigenvector has the. Biggest magnitude or has the maximum magnitude. Because for a space for a matrix, I can definitely have multiple vectors. Okay, so from this, whichever has the maximum magnitude that will be used as a plane or principal component because it is practically proved through maths. Again, I'm not going to go towards that particular proof, but this principal component will be able to capture maximum variance. Okay, so let me go ahead and show you one example. Okay. So here I will just show you one example. So this is a linear transformation. Let's say I have one vector over here okay I have one vector. So this vector is nothing but one comma one right. So here I hope everybody is able to see this okay one comma one. So this is my vector. Now on this particular vector if I apply a linear transformation with this matrix okay. With this matrix now what will happen. You see because once we do the linear transformation, this particular line will now get changed and it will display something like this. See, after I apply this transformation, my new points that I am going to get is something like 442. Right now four two is nothing. But here you can see the lambda value will be four comma two because 110101.011.0 is my vector. Right. So here also you'll be able to see I can multiply lambda multiplied by this particular number. So here you can see four comma two. That basically means for a vector. If I apply a linear transformation with this particular matrix I'm going to get a lambda value with four and two as its lambda uh, as its eigen value. Now let me just show you the eigenvectors. Also, let's say here I've checked on eigenvectors. And let me also show the snap grid okay. So here you will be able to see what is basically happening. This line that you see. Right. This is basically my vector okay. Now here you can see for this particular point when I apply this transformation, the vector is in another direction and this is in another direction. Right now let's say my initial point was somewhere here. Let's say my initial point was somewhere here. And if I probably apply the same operation now it is minus one zero, right? Then what will happen is that you can see the new transformation, right? I'm actually getting minus two comma two. And here you can see this is in the direction of the vector. So what I can do is that I can project this particular point into this plane. Right. So here this will be my longest vector with this longest magnitude okay. Just for an example I'm considering like that. If I keep on doing another another points right. If I keep on doing, let's say I'm putting it over here. Right. And if I go back here you will be able to see this was my initial vector. Now after applying linear transformation it has gone outside the plane. Similarly I can and this time vector did not match. Suppose if I have this point over here and if I try to apply it here, you can see again, uh, this will get transformed to different different values. So your lambda value will keep on getting changed based on the different different vectors. Okay. So from this I can we can definitely see that what is happening is that from this specific equation we will be able to get the eigen vectors. And this eigen vector which has the maximum magnitude. When I say maximum magnitude, that basically means the max eigen values. That will be selected as the best principal component line. Okay. In this particular case it is PC1 because it captures maximum variance. Okay. So at the end of the day you'll be able to see with the help of this eigenvectors and eigenvalues you'll be able to see this okay. And the same thing is happening. Now let's go ahead and understand the steps to calculate eigenvalues. Steps to calculate. Eigenvalues and eigenvectors. Value and eigen vector because as I said, that you require a matrix, right? In this specific case, we have to calculate the covariance of the features. Right. Now let's say if your features were x comma y your independent feature, and this is your Z as your output feature. So for this I want to extract another feature which will can be x dash. So what I will do is that I will try to find out the covariance of x comma y. So let me write down the equation of covariance of x comma y. So here you can see that I will be able to write summation of I is equal to one to n x of I minus x bar multiplied by y of I minus y bar divided by n minus one. Okay. Now here you can definitely see that, uh, if I really want to find out the covariance of x comma y, usually we get two cross two matrix okay. Now this specific matrix looks something like this. Okay. Let's say I'm just going to divide like this. So this will be a two cross two two matrix. This will be x. This will be y. This will be x. This will be y. So if I try to find out the covariance of x comma x then what it will become. So if I try to find out the covariance of x comma x here you will be able to see I will. I can also write this as variance of x y because x comma x basically means x of I minus x bar, right. So x of I minus x bar will be nothing but x of I minus x bar whole square, which is nothing but variance of x. Right. So here you will be able to see that I will write variance of x here this will be covariance of x comma y. Here it will be covariance of y comma x. And here it will be variance of y okay. Because covariance of y comma y is nothing but variance of y. Right. Now this is the first step. We have to compute this specific matrix. And this is for a two cross two matrix. If suppose if I have three independent feature x y z then I have to probably create three input three cross three matrix. Right. I have to basically create like this three cross three matrix. Now here you'll be able to see that I will be creating this kind of lines. And here will be one, here will be another. Right now again here. All in the in the diagonal element. It will be the variance of x, variance of y, variance of z and remaining all. You'll be able to see this. And one more thing. That variance of covariance of x comma y is equal to covariance of y comma x. This both will be equal. Right now what we are going to do in order to find out the eigen values we will take up this covariance matrix. This let's consider it is a and we will try to take a vector v and this a will be taken which will be my covariance matrix. And I will try to apply transformation of the vector. And finally we will be getting lambda lambda over here. And this vector will be given back. Now since I have two features over here f1 and f2, let's say then I will be getting two lambda one values like lambda one and lambda two. Lambda one will basically be giving my principal component line one magnitude, and whereas my lambda two will be giving me the principal component two magnitude. Right. So this will be PC1 right. So like this. When you have three features you are going to get all Lambda one, Lambda two, Lambda three which will specifically talk about the eigen values. So just by using this equation, at the end of the day you will be getting an eigen vector. And that eigen vector will be giving you 2 or 3 based on the number of features that you have, that many lambda values. Let's say if it is giving lambda one and lambda two, that basically means you have two eigen values and whichever eigen value is the highest over here, that will be and it will already give that in that specific value in like, uh, the lambda one will be the highest and lambda two will be the highest. So this way you will be able to get this, ah, Lambda one and lambda two, which are specifically the eigen values. And then this can be considered as PC1. And this can be considered as PC2. So guys just to revise all the steps now uh, what we are going to do is that let's consider the first step over here. So I'm going to write it down. Uh, the first step uh that again we are going to do, let's say that I have my data points over here. So let's say this is my x and y. And uh, let's say I want to probably first of all convert this into 2D from one from 1D to sorry, from 2D to 1d. Then the first step that I am actually going to do is basically standardize this data. Now, once I standardize this data, what will happen is that all this data points that you will be seeing, it will be zero centered, you know, so here you will be able to see all the data points like this. Okay, let's say these are my data points that have got zero centered. After this. Uh, the second step is that we go ahead and find out the covariance matrix. Of this x and y. Okay. Let's say this is my x and y. So we basically find out the covariance matrix. And again in the case of two independent features uh, two independent features I'll be getting a two cross two matrix. Right. So this will basically be my two cross two matrix with respect to x and y x and y. Here I'm going to get the variance of x. And here I'm basically going to get uh covariance of x comma y. And here I'll be getting covariance of y comma x. And finally I'll be getting variance of y. Right now after I get this covariance matrix all I have to do is that find out. Find out. Eigenvectors and values. Vectors and values. And how do I find it out? By just applying. Let's say this is my matrix A so this a when we apply a transformation on vector B, I'm going to get this eigen value of v right now. Since this is a two cross two matrix here I'm going to get lambda one and lambda two okay. So this two values are my eigen values and eigen values okay. And this uh. This lambda one will basically be indicating me by PC1. That is principal component one line, and this will be my PC two line. So in short, after this particular points, what we are going to do is that let's say that we will be creating two different line. One line may be this, one line may be this, or let me just draw it in different color. So let's say this may be my PC one. Okay, so this may be my PC1 and this may be my PC2. Okay. But as we know that the first lambda one value which I'm getting, which is nothing but the magnitude of the eigen vector magnitude of the eigen vector. Here it will capture the maximum variance. So this variance that is getting captured will be maximum. So we go ahead and select this. Now let's say if I want to convert from a three dimension to one dimension then what will happen I will get three lambda values. That is three eigen values. Now what I can do I can combine this two, lambda one and lambda two and probably take. Okay. Or, uh, since I have already explained you, this lambda one is nothing but PC one. This is PC two and this is PC three, right? So I can combine pc1 and PC2, uh and convert this two dimension into 1D. And then this can also be taken as another 1D. So when I combine this two right it becomes a 2D right. So if I probably want to convert from 3D to 2D I'll be going in this way. So let's say if I want to convert from 3D to 1D. So once I find out my eigen value and vectors, I'll be getting lambda one, lambda two, lambda three. I have to combine all these three and probably get one dimension in this case. Similarly in the case of 2D to 1d. So let's say if I want to convert from 2D to 1D here, I'll be getting lambda one and lambda two. This is nothing but my PC one. This is nothing but my PC two. So we can combine this two both and we can get a one dimension point right when I say combining, that basically means I'm talking about projection. Once I project all the point, I will be able to get that specific dimension. So, uh, I hope you have got an idea, but these are the steps that we basically follow in order to find out the projection. But at the end of the day, you know, you are trying to find out. You are trying to find out with respect to any data points, you're trying to find out the best principal component line that can fit on this data such that it captures maximum variance. Now, like in this case, this can be or in this case this can be right. So this can be my PC one. This can be my PC two. Why I'm saying this because maximum variance is captured. See maximum variance is getting captured. Right. So this was the idea behind principal component analysis I hope you have understood the maths behind it. If not, I would suggest please revise it. I've written around nine pages just to make you explain about principal component analysis. So yes, uh, I will see you all in the next video. Thank.