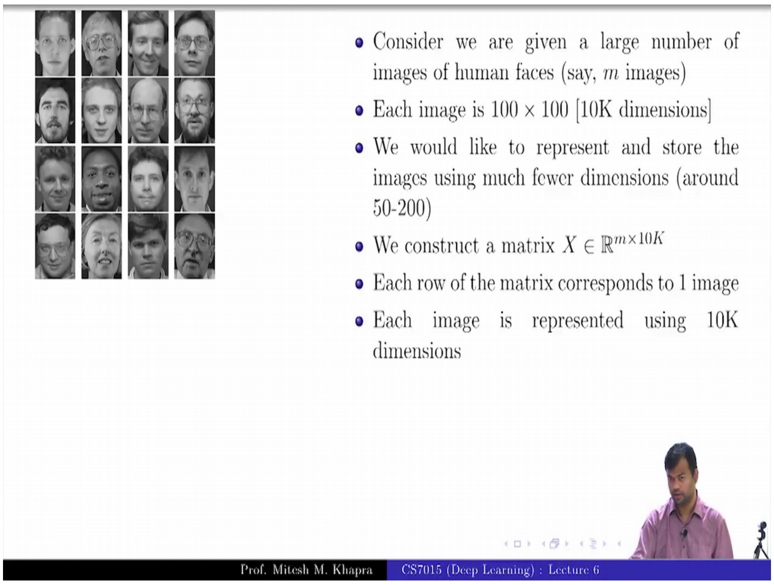
Deep Learning
Prof. Mithesh M. Khapra
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module - 6.7
PCA: Practical Example
Lecture – 06

So, we will in this module we will look at practical example where PCA is used and I just like to give you a flavor of why all this is important right; why do we need to throw away some dimensions and then how does it practically help ok?

(Refer Slide Time: 00:26)



So, consider that we are given a large number of human images right. So, this is like some faces data set; a database that says one of the intelligent agency someone is maintaining, one of the government agency or may be Aadhar data bases or something like that ok. Now each image here is 100 cross 100; that means, it is 10 K dimensions right, it is a very high dimensional data ok. And your job is to actually store this on to do some database for a large amount of the population right because you are collecting these images from various people.

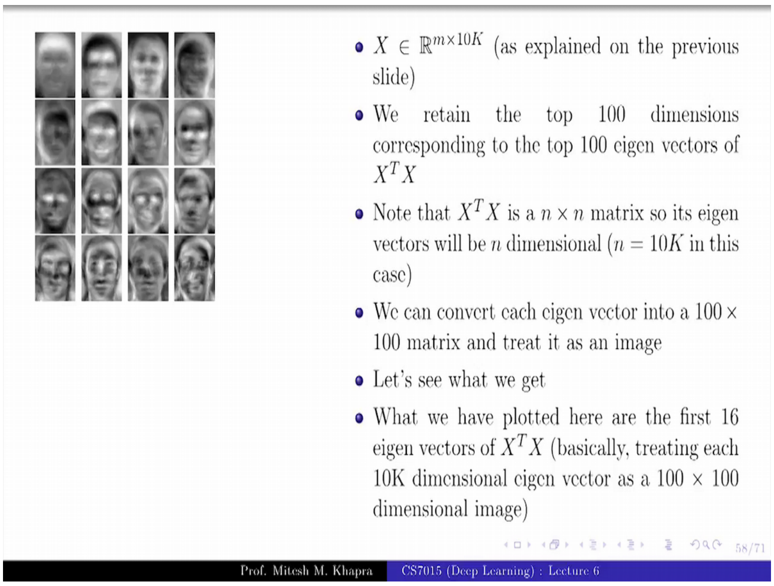
So, now we would like to represent and store this data using much fewer dimensions right. And you would be really ambitious that if you want to store that more than 50 to 200 dimensions right, so you see the compression that I am looking at. You have 10 K which is a big storage problem for me and I want to just bringing out to 50 to 200, but I have know that this is crucial data right. I do not want to store information which is not able to distinguish these faces, I was still be able to reconstruct the faces from this information right; do well I mean minimum error deconstruction from this and that is exactly what PCAs are allowing us to do right. So, now we construct a matrix of m cross 10 K; what is m?

Student: (Refer Time: 01:46).

The numbers of samples you have, the numbers of data points that we have and each of this is of dimensions 10 K. So, this is what matrix, what do we call this matrix? Oh it is already given right; it is the X matrix the data matrix that we always. Now each row of the matrix corresponds to one image and each image is represented using 10 K dimensions just to rehydrate ok

Now, let us see; so now what would you do? This is the original data, I want a dimensionally reduced data right, you want store this ah; is the mike working? you want this data to be represented by a fewer dimensions. So, what is your solution? Do PCA. So, what will you do? X transpose X right and I did not get my slide (Refer Time: 02:43) ok.

(Refer Slide Time: 02:44)



So, we retain the top 100 dimensions corresponding to the top 100 eigen vectors of X transpose X right. So, basically we do a PCA, find the 100 find all the eigen vectors of X transpose X and then just retain the top 100 of those ok. Now what is the dimension of each of these eigen vectors? Should be straight forward take your time, it is early morning.

Student: (Refer Time: 03:10).

10 K right ok. So, now can you think of a physical interpretation of this? So, what are you trying to do? You are trying to store faces and now you have come up with these dimensions ok; no sorry we have come up with these basis vector which is eigen vectors and each of them is also 10 K; which is as same as dimensions of your faces ok.

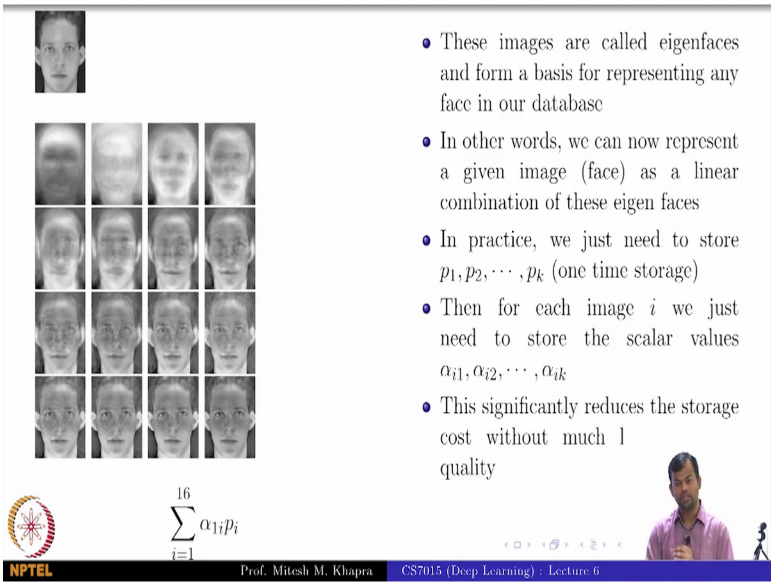
Can you think of a physical interpretation of what is happening here? None of went you through the slide except perhaps you or I do not know just think about it. So, what you are trying to do is you are trying to represent any possible face in your database, right using a linear combination of some vectors ok. Now these vector should have some interpretations right, it should be connected to faces and somewhere; otherwise how will you construct a face from taking a linear combination or some random vectors; do you get the point?

So, can you think of each of these 10 K dimensional vectors which is the same as the dimension of your original data; as a face and try to plot it? Can you try to do that at least it make sense 10 K dimensional ok, that is the same what you are image size was. I could just variance these 10 K dimension as 100 cross 100 and try to plot it ok. So, let us see if you do that what happens ok; we convert each eigen vector into 100 cross 100 matrix and treat it as an image and let us see what we get.

This is what we get. So, this is the top 16 eigen vectors that I have plotted; now can you tell me a physical interpretation of this? This is the basis for constructing any face in your data base right; that what you are trying to say all the faces that you have in your database or in the world you can combine them by looking at the these elementary face structures right, which are your basis.

And then you could scale them up by using these alphas, you will be multiply them with the certain alpha right. And when you combine them you will get the base any face that you had in your original database. Does a physical interpretation make sense, how many of you get this? Ok good.

(Refer Slide Time: 05:28)



So, that is what is happening here. So, we have constructed this basis now I will come to that later. So, these images are actually called eigen faces and the form of basis for representing any face in our database. In other words, we can now represent a given image as a linear combination of these eigen faces. So, this is my original image ok; I want to reconstruct it. Now I will use 16 or 25 of these eigen faces; what do you think would happen? We will get some face which has some error; there is some error in reconstructing this face ok

So, let us see what we get. So, I am using only one basis vector and I found out this alpha one I how would I found it out?

Student: (Refer Time: 06:11).

Dot product of the face vector with the basis vector ok, now I instead of 1; I take 2 ok. You see I have took this 2 basis vectors, scaled them with the corresponding alpha coordinates and added it them up right. And I am trying to get some face value it does not look it goes to the original face; that means, the dash is very high; the reconstruction error is very high.

That means, I have still dropped some of the important dimensions; I have still drops some of the important eigen vectors right. So, the value of K which is the top eigen vectors is something that I need to take care of it and should be in a way. I can always construct the reconstruct; I can always compute the reconstruction error here right, how will you compute the reconstruction error?

Student: (Refer Time: 07:00).

Take the square error between the original image and this second image that you see here right. So, you will take the square error between this and this and you will end up with the number which is not acceptable right. Now what I will do is I will go further I have taken 4 still not quite there, but I can see a shape emerging right.

I go to 8 things becomes better; since you are already know what the original face at least you can make sense of it. And by the time I reach 16 I am almost there right; at least I can recognize the face that is probably losing out some certain things in the face, but I can recognize it.

Now, how many of you appreciate what is happening here? Yeah of course, now what is happening here? So, think in terms of a practical application right; what have you done, what have you able to be achieve, how may basis vectors where you able to store or did you store?

Student: 16.

16; that means, 16 into 10 K values and suppose you had a million images in your database; how many would you require to store?

Student: (Refer Time: 08:11).

Wait let us we do it step wise forget about PCA, if you had a million image in your data base and each of them have 10 K dimensions how much storage do we need?

Student: (Refer Time: 08:24).

Million into 10 K floating point values ok. Now with if I say 16; 16 may be too ambushes may be later on I will say 50 or 100 is, but let us say 60 then how much data we need to store?

Student: 16 into 10 K.

16 into 10 K and you can reconstruct any face?

Student: (Refer Time: 08:49).

Yeah alpha's needs to be stored; right. So, for every image instead of storing 10 K dimensions, you will just store the 16 alphas right. And you can see that even if I go there to 100 and its still manageable, instead of 10 K I am going to just store 100 alphas right and as I go to 100 what would happen?

Student: (Refer Time: 09:09).

The reconstruction error would become even lesser ok. So, is that is the intuition clear ok? So, this eigen vector storage is a onetime storage; we are going to store this K eigen vectors, each of them are 10 K dimensional and K is 100 or 200; we do not really care because the original data was very large right. And for each image we just need to store these alpha values K of them right. So, for each of them instead of 10 K, we will store 100 to 200 alpha values and of course, it is significantly reduces right.

So, this is why we need to do all this right and what is the other advantage of doing this? Anything of; something else; so what is PCA actually allowing you to do? If you again think of it not; I mean subtract the math out, just think it in terms of physically interpretation and what is that it is allowing it to do? If you had to say it in English what is it allowing you to do? Compression is a loaded word can you just spell it out for me what is this compression mean actually?

Student: (Refer Time: 10:14).

Right. So, it is storing all the relevant information in the image and discarding all the error element information right. Now this also ensures that if you have multiple images of the person, then what would happen? You should take the image under lighting conditions or may be at person had applied some makeup or something like that right what would happen?

Student: (Refer Time: 10:38).

In the original space, the 10 K dimensional space what would happen to these images? They will be very far from each other right because the lighting conditions have changed, you see a dark person, so have a light person something like that right. And now because PCA has elliptically through with this dimensions right; may be the exact terminology which I am using may be the lighting condition do not do it. Because you can imagine that there would something right that suppose as some element which is calling the image to look slightly different, but that is not the important information right. So, that would get discarded off and only the relevant information would stay; right.

So, then multiple images of the same person which were dash in the original space would come dash in the new space.

Student: (Refer Time: 11:22).

Far in the original space would come closer in the new space right. So, this is what compression helps you to do; so, this is what you want to learn you want to learn the important characteristics of your original data and that is what PCA allows you to do fine.