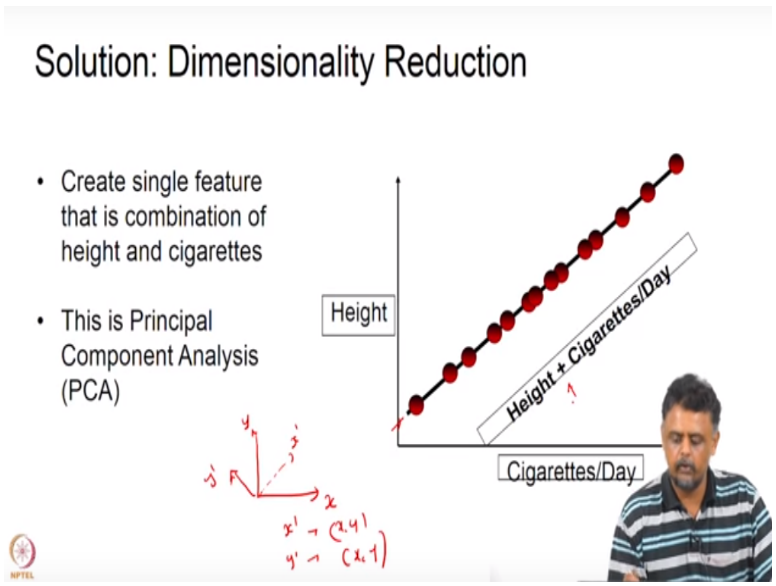
Machine Learning for Engineering and Science Applications
Professor Dr. Ganapathy Krishnamurthi
Department of Engineering Design
Indian Institute of Technology, Madras
PCA – Part 2

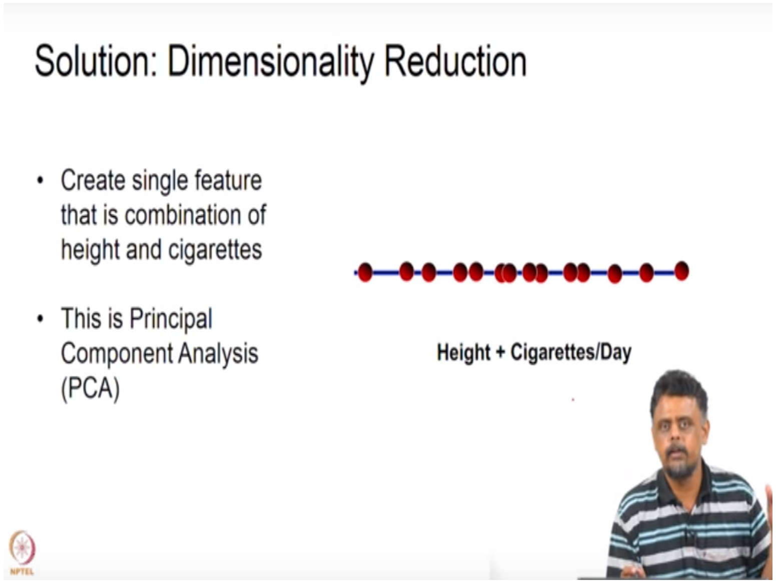
(Refer Slide Time: 00:14)



So that is what we have done the manage to project our data which is of two column which contains heights and cigarettes per day into a single axis, so you can think of it as some combination of height and cigarettes, so as I mentioned earlier you can also think of it as rotation of your axis, so I am sure if you have at some point in your college or school you must seen this when you have axis X Y and then you rotated by an angle.

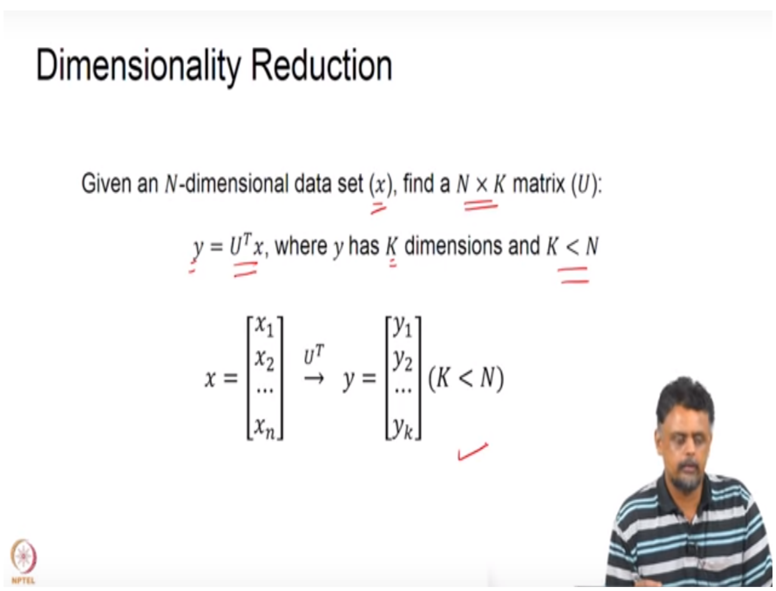
Let us say theta to get a new X prime and Y prime it would done be possible to express X prime and Y prime each of them as combination of X and Y, so this you must have seen at some point in your high school algebra and it is, so this is accomplished of something similar it is not exactly attributing but it is more complicated transformation involving both this features heights and cigarettes per day.

(Refer Slide Time: 01:10)



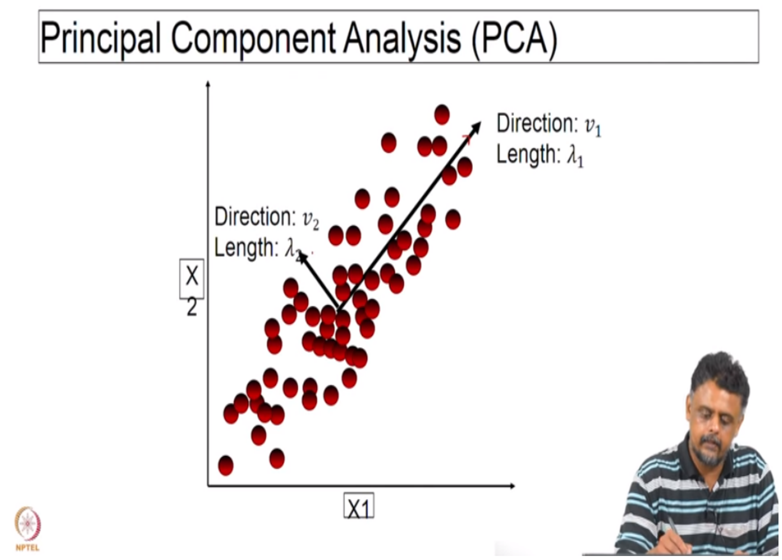
So finally that is what we have we create a single feature which is a combination of height and cigarettes and this process of reducing the dimensionality of the data is what we called principal component analysis.

(Refer Slide Time: 01:23)



So mathematically is what we can state this at have given an N-dimensional data set X or idea is to find and N by K matrix U, so that when we apply U transpose X, we do this operation U transpose X and we get this new data Y which has reduce dimension it has dimension K which is less then N, so that is what this given here in this part following operation, that is precisely what we going to do if you want to put in terms of linear algebra matrix operation

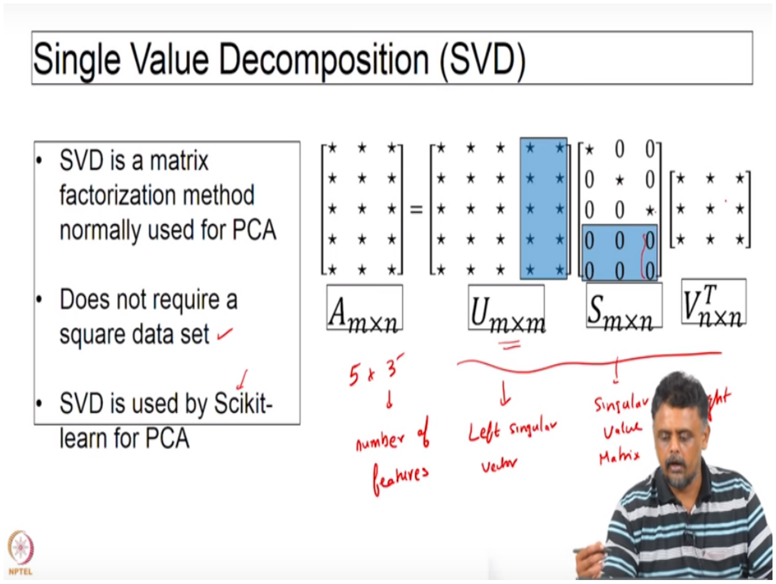
(Refer Slide Time: 01:57)



Now let us consider this that I said this has two features X1 and X2, there is X1 and X2 visually we can see that you know that lot of the information is along one axis, it is along the axis here very black arrow, so which mean that this axis at the maximum variances if you can construct this axis maximum variances so we have an another axis which is in this case it is orthogonal to this as 90 degree, so orthogonal to this current axis I have drawn an orthogonal to that axis variation is very small or the variances the other axis is very small so if project your data along this axis one of the first true the axis that we first true then we could be able capture most of the information because the variances is height in direction.

And the other direction that look we consider is the direction orthogonal to the axis that we are presented and that the variation along the direction is much lower so this the idea behind doing principle component analysis, so what we need for that we need two things we need the direction of this axis in this direction we need this vector this vector we need to know and we need to know the length of the vector because the length of the vector helps us to determine whether the variances is high in this direction or not the larger the length the more the variances along the direction and similarly we need to know the length of the direction of the other vector, so that is what principal component analysis help this equation.

(Refer Slide Time: 03:25)

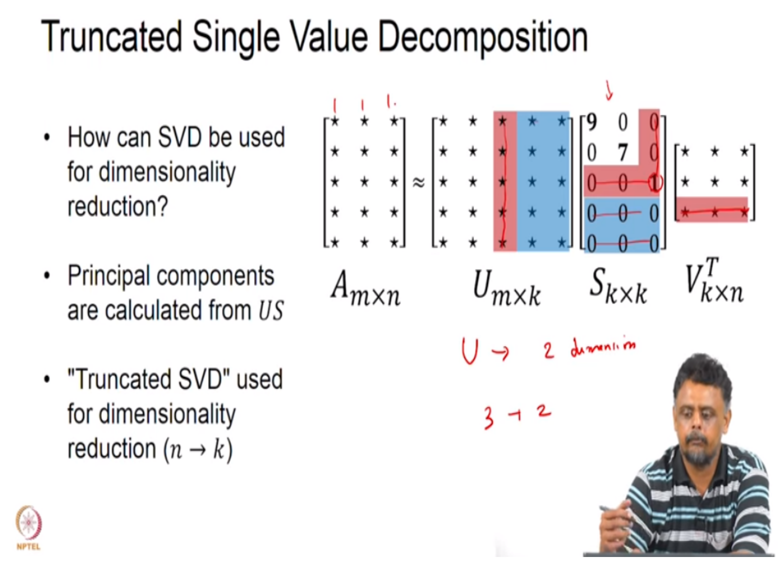


So how do we do this accomplished it we would not go through the actual algorithm for determining this but what we can do show the plot is process thus so this principal component analysis accomplished using what is known as single value decomposition there is stage there is usually called single value decomposition not single value it is a matrix factorization method that normally use for principle component analysis this is not required a square data set, so your matrix not be a square matrix and it is used in this python package Scikit-learn for PCA even MATLAB is a command SPD which help you to similar identity decomposition for mount square matrix.

So these has a five in this case M by N matrix M is five, M rows three column, so this is the number of features, so what singular value decomposition thus is to factories into a product of three metrics this is called the left singular vector so a right singular vector and this is called the singular value matrix, so if you do this on a square matrix, so what will get or the singular value matrix is nothing but the eigenvalues is the diagonal matrix of eigenvalue and the left and right singular vector nothing but the eigen matrix of eigenvectors, so if you have N in this case five cross three of matrix we have in this case five data points we have three feature that what this three features we want to reduce it.

So we get this U matrix which is a left singular vector which is of size M cross M five cross five the singular value matrix the singular value matrix which is of actually diagonal matrix so it is only case the this two dimensions have zero, so we these three singular values and the V transpose has dimensions N cross N which is three cross three, so this is the output of the S video algorithm and what so how do we decide you know how do we actually reduce a dimension right.

(Refer Slide Time: 05:57)



Again the idea is the visually we saw in that data set there are one direction along which the variances of maximum however when the number of dimension increases hard to be visualize so then we the visualization is done using this particular matrix which is the singular value matrix if you see that then these rows are already relevant because they do not correspond to any useful similar values so the least singular value corresponds to this, so we can drop the row and column corresponding to this singular value which corresponds to this particular column here and this row here in the V matrix.

So again we can drop these also because they do not correspond to any useful direction, so that way you get a V matrix which can be we can project to two dimension, so we have reduce the number of feature from three to two by throwing out one direction corresponding to the least singular value, so this is the truncated SVD that we used for dimensionality reduction, so we can so in this what we have to do is we can throughout this rows and columns and then actually do the multiplication to get a correct form of your data matrix so remember the each column correspond to features.

So that is how the data point should be arranged one more information before you do PCA is that they all have to be zero centered so each column has to be subtracted mean subtracted, so that the mean of this is zero the one of the required from doing a PCA, so this is one of the most often use technique for dimensionality reduction, it says that this is the pre-processing step for any machine learning algorithm classification regression you now name it and even if you want to do deep learning with images you can actually do this, except that now you have to rash up the images that two column and rows depending and how you arrange the data.

So that is you know this is like the work cost technique and it has proven to shown to improve performance in many algorithms, because what it does it is removes the unwanted feature, so by re-projecting your data into a new axis it remove unwanted features and only keep those feature which are relevant my themselves having a maximum various, so some key points remember I would like to retreat here, the idea is that we have this data set here all this red back data points represent the two dimensional data X1 and X2 in which is 2D the idea is to find a new axis to represent this data but the condition for the new axis then still for orthogonal axis is that we are still in on orthogonal axis that to they are orthogonal axis.

So that is enforced by the algorithm idea is we find this the axis corresponding to the most variation in the data and then find an another axis corresponding to perpendicular to it and then look at the variation and direction so and on so forth, so the principal axis or orthogonal very important and the main the axis we want to keep have maximum variant along that direction this are the key point that the you have to remember and the way this is accomplished by doing this SPD and since there are in the case of more than two and three dimensions the best way to figure out which axis is the most variant is look at the singular value matrix.

The singular value matrix the you can keep the first case significant terms so that you can project such the K less than N, where N is the original dimensional T of your data, here we conclude with principal component analysis again there are lots of resources on the web regarding the actual algorithm itself we will post some on the discussion for an as soon as only open announce for up.