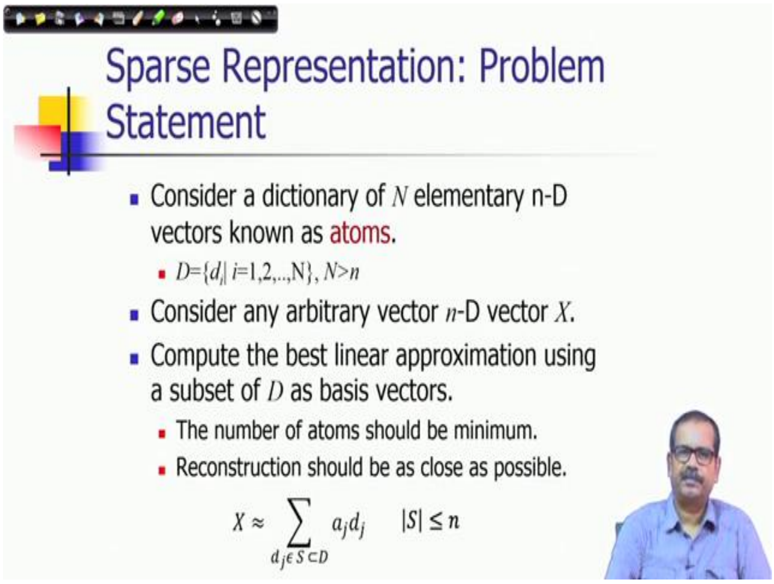
Computer Vision
Prof. Jayanta Mukhopadhyay
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 53
Dimensions Reduction and Sparse Representation Part – III

In previous lectures we discussed about dimension reduction of data points.

(Refer Slide Time: 00:23)



Now, I will be discussing another issues related to data representation which is called sparse representation. So, let us understand the problem statement involved in this particular issue. Consider we have a dictionary of N elementary n D vectors we call it dictionary it is something like set of basis vectors.

But in the set of basis vectors we would like to have only those basis vectors which will be sufficient to represent data. But here we consider a redundant set of basis vectors; that means, we have more number of basis vectors and if you would be only required to represent a data.

And the term what we will be using instead of set of basis vectors we will be calling this particular collection as a dictionary. As if you have lot of redundant vectors and conveniently you would like to use them for representing your data or signal.

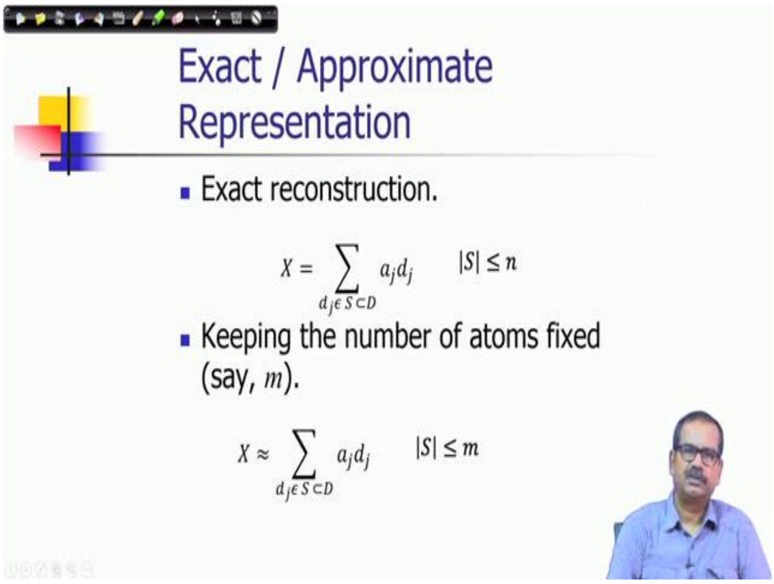
Now, the elements of this dictionary they are called atoms that is a terminology we will be using and you consider any arbitrary vector n dimensional vector. So, the problem here you have to compute the best linear approximation using a subset of D as basis vectors. As I was mentioning that D is an over complete representation of basis vectors which means there are many redundant vectors and it is not required all the vectors should be used for representing data.

So, the linear combination of the subset of D can be expressed mathematically in this form as you can see. That X is your input vector it should be very close to a linear combination which is given as a multiplication of each dictionary elements with the scalar constraint aj and some of them. And dj is are the elements of that set which has been considered for representing X and which has to be subset of this dictionary.

And in an n dimensional space it is sufficient if you have n independent basis vector, n linearly independent basis vectors and then it is sufficient you can use into represent any arbitrary vector X. So, here you have an over complete representations and you deserve that this number should be less and that is the sparsity we are considering.

That you required a few basis vectors only from D which should be sufficient to represent X we do not require N basis vector. So, it should be much smaller to n and that is what we are trying to achieve using these kinds of representation, so the idea is that this number of atom should be minimum. So, for the particular problem statement you would like to have that S subset whose cardinality should be minimum and also reconstruction should be as close as possible. So, it is an approximation.

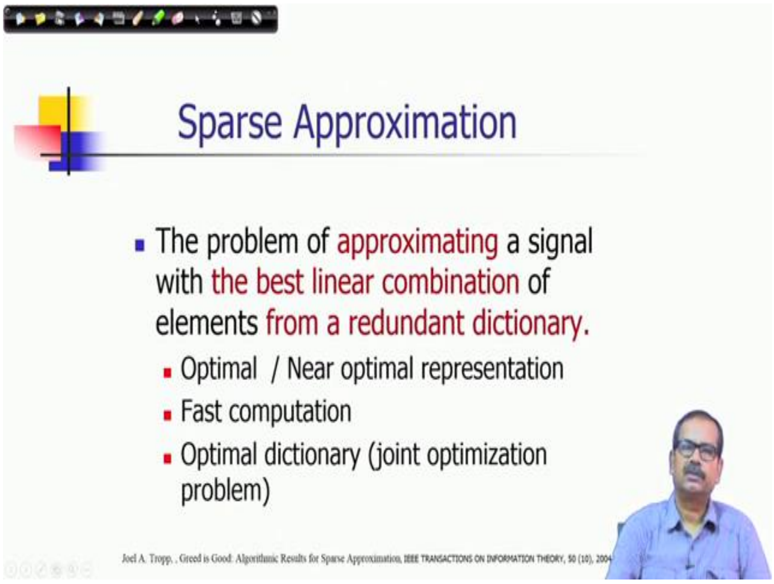
(Refer Slide Time: 03:34)



So, to make it more precise in that case how close it is first thing it could be exactly construction. So, you find out the minimums which will give you the minimum cardinality of S, a minimum number of atoms which will give you the exact reconstruction. So, this could be a problem and it is a very precise problem statement.

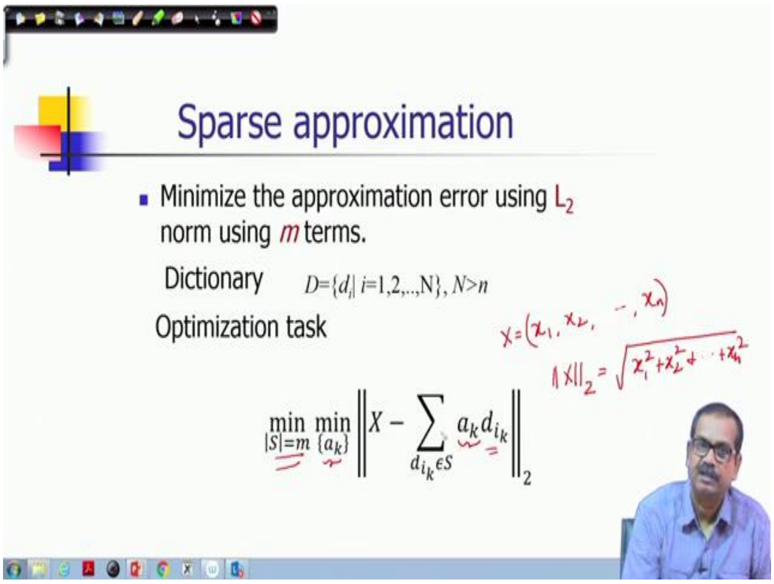
Or what you can do we can keep the number of atoms fixed say what is the best representation; that means, how close that should be the error between the representation approximation of X and X should be the minimum original value, so it should be very close to that.

(Refer Slide Time: 04:11)



So, to summarize this problem it is the problem of approximating a signal with the best linear combination of elements from a redundant dictionary. It should give some optimal or near optimal representation, the competition should be fast. These are the desirable properties and the dictionary also should be optimal and sometimes there are joint optimization problem that you would like to have also a dictionary of with an optimal size, so that gives also the optimal sparse representation of data.

(Refer Slide Time: 04:46)



So, to elaborate this optimization task we can consider this mathematical formulation that we have to minimize the approximation error using L2 norm. You know L2 norm is basically it is the Euclidean distance or in the n dimensional space and this it is we are using this and also using m terms. So, you have a dictionary D which is a set of d i's each one is an atom and this considers L 2 norm. So, just to show you what means by L2 norm, suppose you have a vector is represented as say x1, x2 to xn.

So, L2 norm is

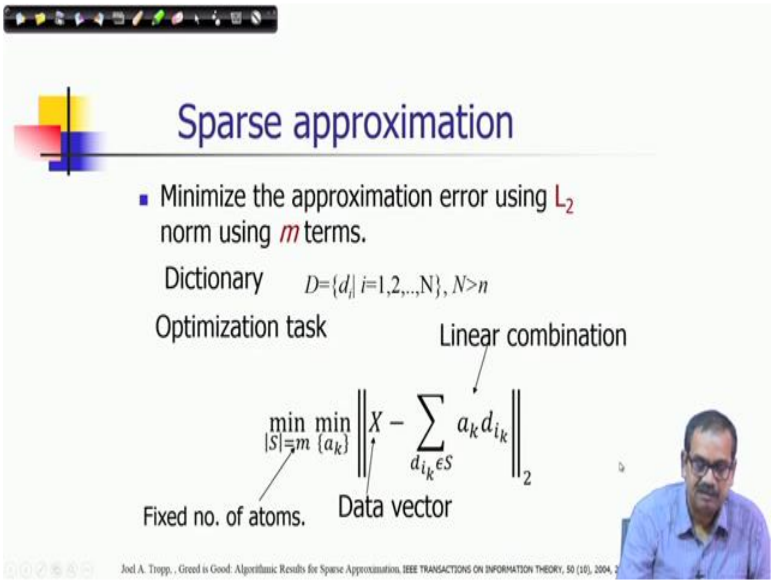
$$L_2 = \sqrt{(x_1^2 + x_2^2 + .. + x_n^2)}$$

So, you consider this is an error vector

$$\min \min | \left| X - \sum_{d_{ik} \in S} a_k d_{ik} \right| |_2$$

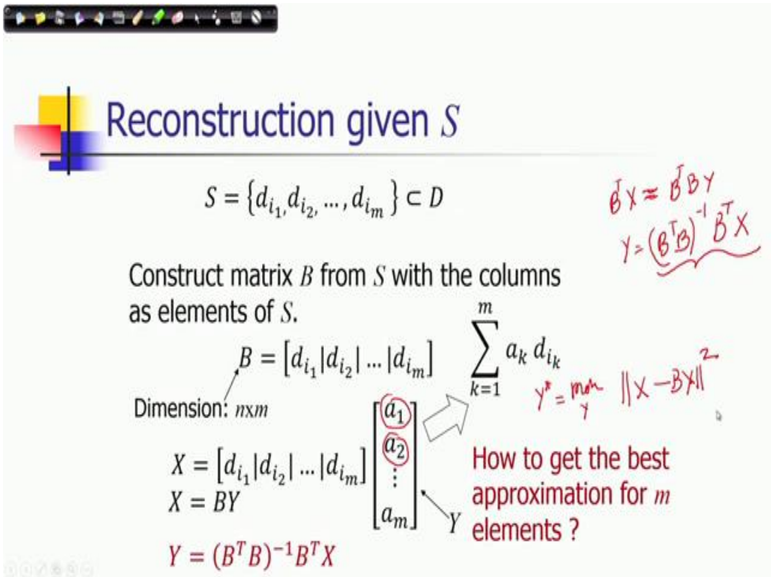
So, the problem here is that you should get that set of coefficients and the total number of atom should be exactly equal to m, so that you get the minimum error minimum L2 norm of these values.

(Refer Slide Time: 06:24)



So, these are these explaining the corresponding conference of this expression. This is the data vector and this is the linear combination and it gives you the fixed number of atoms that constraint.

(Refer Slide Time: 06:38)



Suppose I have given you a set of atoms which is a subset of the dictionary D. Now, how can I get the best reconstruction we in terms of linear combination of these atoms for in a arbitrary X, for in a arbitrary vector X. So, we can see that we can probably it easily the least square error automation problem. We construct a matrix B, so once again be relates to that setup basis vectors and each element of S is considered to be a vector this is vector.

So, this B is found in this weight, so each column vector of B is a corresponding atom of S. So, the dimension of B should be n X m as that dimension of each vector each base is for each element of d is n. So, each column victor is of n that is the dimension, so number of rows would be n and since there are m elements m atoms, so we will have m columns.

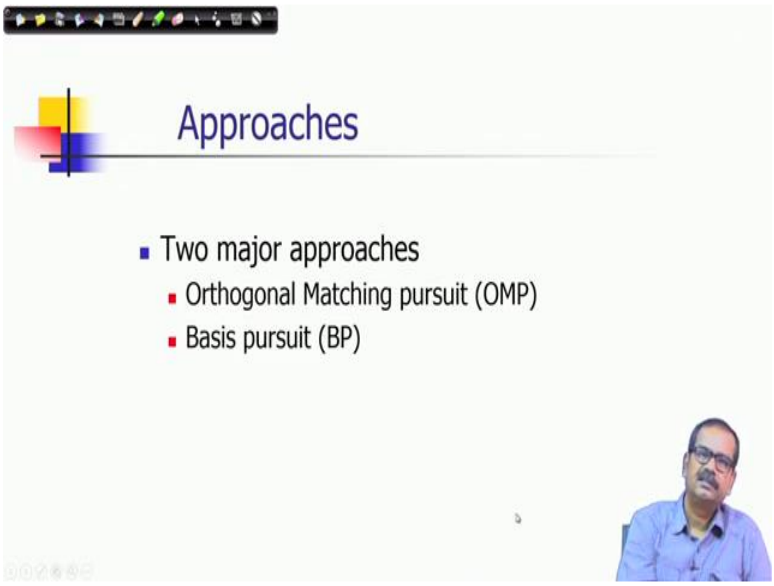
So, the dimension is n X m and if you consider the linear combination of these atoms which can be conveniently represented as in this form. So, here a1, a2 these are all scalar quantities these are all scalar. And you are multiplying $a_1 d_{i1} + a_2 d_{i2} + .. + a_k d_{ik}$ like this you proceed and you can represent in this form that is well you get the linear combination.

So, this linear combination can be represented and this is this set of coefficients or this column victor let me consider. Let me denote as Y and this linear combination can be represented as the multiplication of B and Y, B is the set of basis vectors as I mentioned from the dictionary where each basis vector is the column of B. Or dictionary elements of is the column of B and Y is the corresponding set of coefficients, early stock coefficients with respect to each atom, so Y is the representation of X given this dictionary B.

So, Y is the transform of X and this is this should be sparse we would like to have Y as sparse as possible, so how to get the best approximations for m elements. As you can see this is nothing but you can convert it to a list squared error estimation because, what you can do you can minimize here B is given X is given what you have to find out? You have to find out Y.

So, minimize this norm X minus B Y squared and that Y will give you the optimal representation of this particular element representation of X with respect to these constraints. So, we know this solution a number of times you know we have we did it, because here once again just let me put it I would like to get the close approximation in this form. So, what we can do? We can multiply B transpose and then Y can be written as B transpose B inverse B transpose X. So, this is the representation this is the solution of this particular problem, so this is how you can get Y out of this process.

(Refer Slide Time: 10:56)

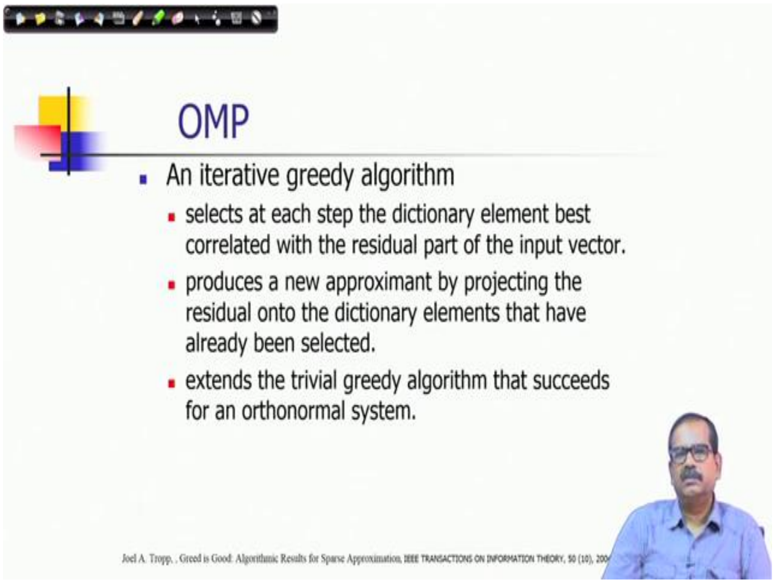


So, the problem here is that if I give you is then use the least square error estimation method to get a representation Y for particular X. But which subset you should consider, which list to m atoms you should consider from the dictionary that is what the basic problem is.

If I can if I know that this set of atoms will give me the minimum reconstruction, then our sparse representation I can easily get the solution.

So, there are various approaches to select those atoms which will derive this kind of sparse representation. Now, these approaches are called pursuit approaches, so two major approaches; one is orthogonal matching pursuit, another one is basis pursuit.

(Refer Slide Time: 11:50)



In orthogonal matching pursuit or in short we call it OMP here the algorithm is in iterative greedy algorithm, what it does it selects at each step the dictionary element which is based correlated with the residual part of the input vector. We will see what it means; that means, what it is doing it is successively refining the refining the representations it is like successive approximations.

You take the original vector X or input vector X and find out the basis vector where the component is maximum; that means, that vector is representing the maximum part of the signal. So, you choose it in your dictionary element and that is your representation itself, so that vector and the component.

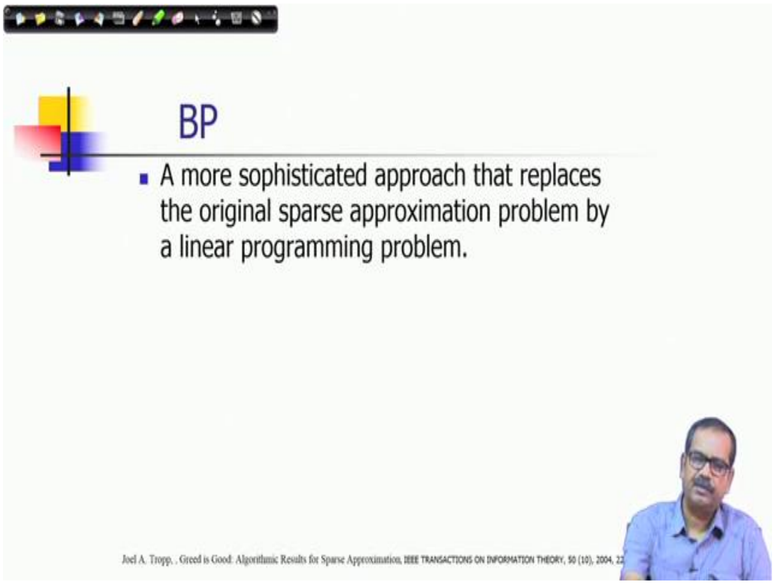
There next time what can you subtract the represented part already what is being done by that vector that is called residual. The next time you are going to carry you in the same operations with residual, so with residual again you find out along which directions you are getting maximum component.

Then you add it to your dictionary use it for least squared error estimation, use it you as I mentioned that if I give you a set of basis vectors I can easily derive what is the best representation best linear combinations which will represent the corresponding vector. So, use it use that technique to get derive the best representation and continue doing it for m atoms m number of times than or as long as you want to represent the data.

So, it produces a new approximate by projecting the residual onto the dictionary elements that have already been selected, what I described that is mentioned. Here that every time you are considering residual part of the input vector you project it on a particular vector a dictionary element which has not has been used already.

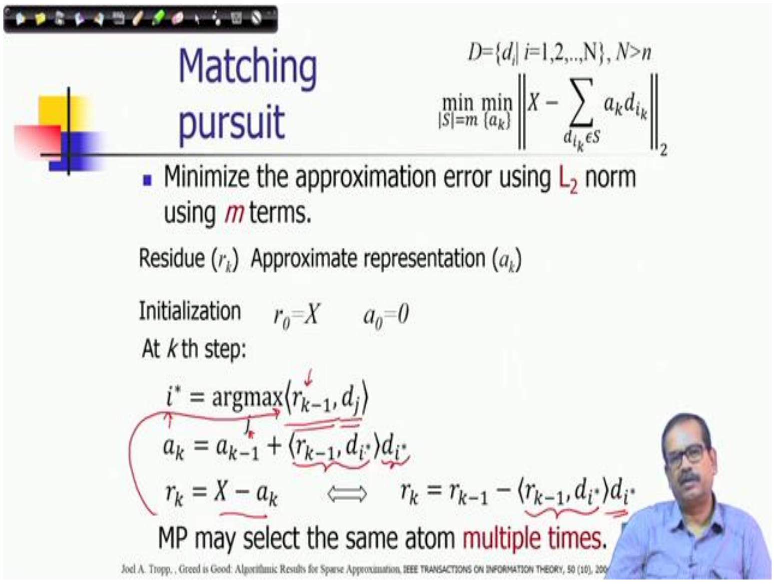
And the dictionary element which is giving you the maximum value of projection that should be included into your set S. And use set S to represent the efficient representation it extends the trivial greedy algorithm that success for an orthonormal system that is a characteristics.

(Refer Slide Time: 14:14)



The other approach which is called basis for suit it is a more sophisticated approach. And what it does it replaces the sparse approximation problem as a linear programming problem I will explain elaborate that problem statement.

(Refer Slide Time: 14:29)



So, before describing the orthogonal matching pursuit algorithm. Let me describe a similar algorithm which was say it before orthogonal matching pursuit it is simple matching pursuit algorithm. So, the idea is here that at every time you just look at only that those you know elements of the dictionary where you get the maximum residual component; and use that you know element itself to represent the vector.

So, it is minimize the approximation error using L 2 norm using m terms that is objective. And you are considering that at k th iteration since it is iterative process, you have residue representation as $r_k = X - a_k$.

So, the initialization is that your whole signal or whole vector input vector itself as X like a residue r naught is equal to X and so far you do not have any approximation which is 0 which you need to do. Then at every step every k th step what we are doing, as I mentioned you are finding out the direction finding out that element of dictionary where you get the maximum you know projected component.

So, this is the residual at this stage, so what is the projected residue along this particular you know dictionary element and then consider only that dictionary element where this value is maximum. So, you choose that dictionary elements say i star is that index, so now use it for representing this signal. So, this would be your magnitude and this would be your direction of the vector.

So, now you are, so the signal which is represented by a k that is the approximation approximately representation at the kth iteration, it should be incrementally it is representing that approximation. So, with previous approximate representation you are adding this vector as I mentioned. So, this is the magnitude part of that vector which is the component what is there in the residue. And this is the direction of the vector, so d_i* gives you the vector itself.

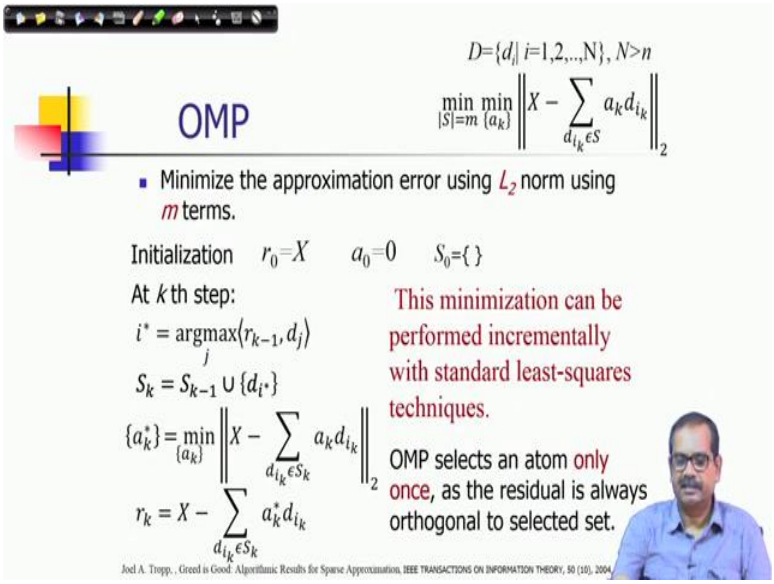
So, you add it with a k - 1 that is previous approximation and then you get the corresponding approximately representation. So, now, the residue rk, ak th rate step would be this and this residue will be for the next iteration use this residue once again for finding out the next of it is representation approximations. So, you privet this process till you get a good approximation and you should note that this is equivalent to this competition.

So, I can subtract the approximations what is the approximate representative signal with from the total signal then also I get the residue at this stage or I can incrementally build up the residue. So, earlier I had residues still k - 1 stage and out of that already this part is accounted. That means, the rk-1 this residue the projected component of the residue as in di* it has been already taken care of it has been transferred to the approximation part.

So, the residual part gets reduced, so slowly residue is reducing and it is trending to 0. So, when it becomes 0 it becomes exact representation otherwise, the whatever part is there that is the approximation still that error is error representation will remain as a residues. Now, problem of matching pursuit as you can see in this process it may select the same dictionary element again and again.

So, we have not put any restriction we are concerting of the throughout the whole dictionary element itself. So, if you are a residue it is still, so at some stage once again you can get the same dictionary element and if I am trying to restrict per m terms. So, I have to also keep a separate count that how many elements are presenting taken care of in my representation of this signal.

(Refer Slide Time: 19:02)



In orthogonal matching pursuit algorithm we take care of that factor, so that the elements of the dictionary they do not appear repeatedly. So, what because we are we will be performing here least square estimation, so your residue will be always orthogonal to the selected set of dictionary atoms. So, that is the property we can ensure by doing least square error estimation. So, here the process goes like this first you have to initialize it, so you have this two atoms r_0 =X and a_0=0.

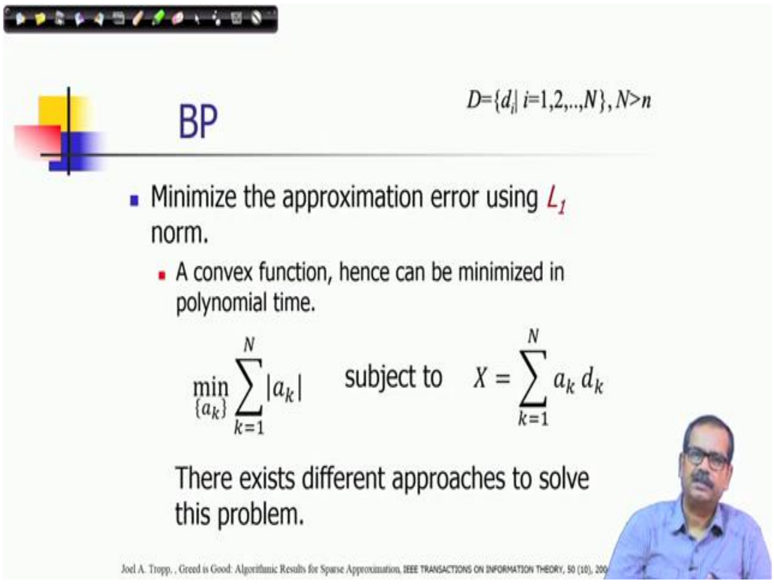
So, this is the same initializations and at kth step and also you are considering the set which will be including the set of dictionary elements which will be reconstructing the signal. So, at kth step first we find out that what is the direction along which you get the maximum component of the residue. And in the same way what we did for the matching pursuit and once you have chosen that then you put it into the set of atoms what would we use for representing the signal X or the input vector X.

And we would like to find out the best linear approximation there we will be using the least square error estimate method. So, that is the difference from matching pursuit, but by doing it what you are ensuring that all the atoms selected in the set S. That means, Sk at k th iteration all these atoms they are orthogonal to the residue rk at that stage, k th stage residue. So, the rescue is computed like this, so I mentioned that this minimization can be performed with standard least square techniques.

So, this is a residue and this residue it we have ensured through the step by enforcing least square error estimation that rk is orthogonal to all the elements of Sk. And that is why in the next iteration you do not require to bother about finding out the elements.

You are only choosing those elements which are remaining elements of dictionary which are not within the Sk. And you go on doing these things till m th terms for best m terms approximation this is what is orthogonal matching pursuit. So, as I mentioned that OMP selects and atom only once as a residual is always orthogonal to the selected set.

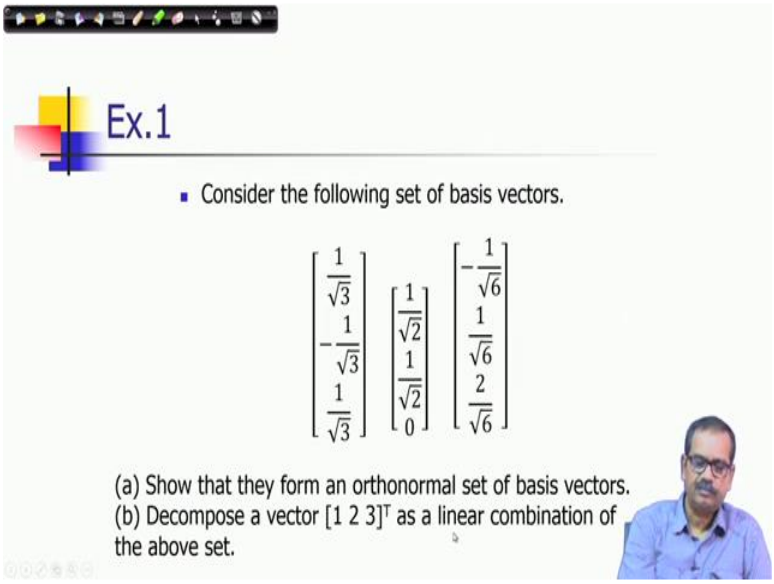
(Refer Slide Time: 22:03)



Basis pursuit problem is different form orthogonal matching pursuit problem. It minimizes approximation error using L1 norm and that too it is L1 norm on the coefficients of representation; that means, coefficients of the linear combination. And it gives you a convex function and hence it can be minimized in polynomial time and you can see that this is the problem statement as I mentioned that you are trying to minimize the sum of coefficients represent the signal.

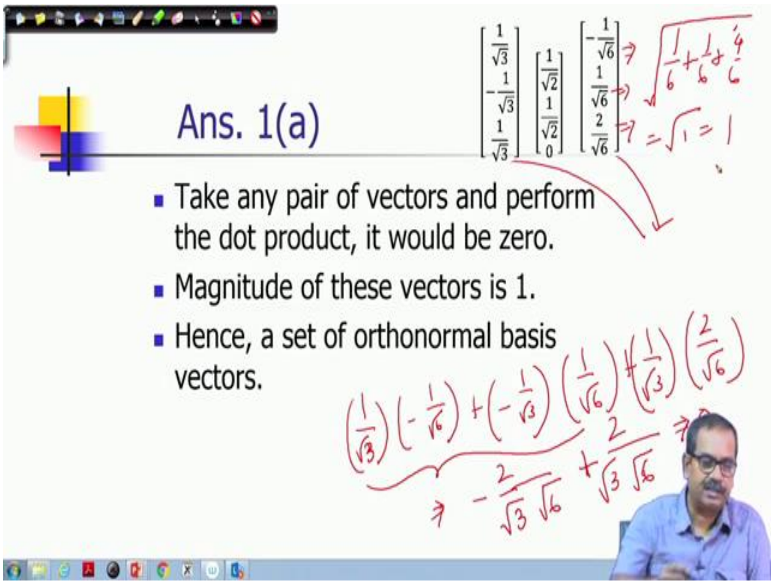
And by that you will be achieving as sparse representation and there exists no different approaches. To solve this problem we are not discussing going to discuss any such solution it involves also mathematical complex mathematics to argue on those solutions. So, in this particular case we will be considering only orthogonal matching pursuit for representing for getting a sparse approximate representation of a signal using a dictionary.

(Refer Slide Time: 23:13)



So, let me solve some exercises to elaborate this competitions, let us consider these problem that you consider the set of basis vectors there are three set of basis vectors. And you have to show that they form an orthonormal set of basis vectors and then if I given you these orthonormal set of basis vectors you represent a vector 1, 2, 3 as a linear combination of the above set.

(Refer Slide Time: 23:46)

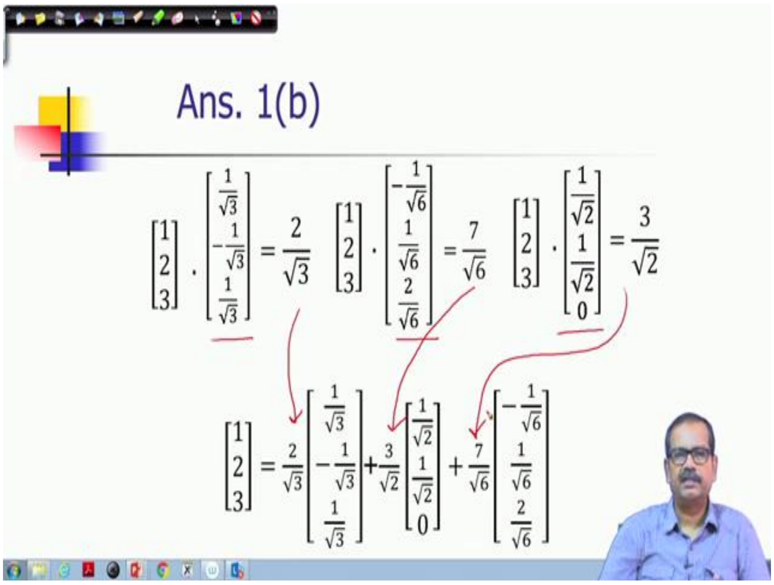


Now, for checking orthonormal setup basis vectors what you can do, you take any pair of vectors and if you perform dot product then it should be 0. So, any pair of vectors different vectors and magnitudes of these vectors all the magnitudes of these vectors should be equal to 1. So, and that is how we can test and you can see here also just typically let me give you one example, suppose I have taken this vector and say this vector.

So, the dot product of this vector is 1 by root 3 into minus 1 by root 6 plus minus 1 by root 3 into 1 by root 6 plus 1 by root 3 into 2 by root 6. So, you will see that these are the two negative terms it should be minus 2 by root 3 into root 6 plus this is 2 by root 3 into root 6, so they should they should give 0.

And in this way you can choose any pair of vectors and you can find out they are dot product is 0 and if I take the magnitude, so take the magnitude of any of these vector say magnitude of this. So, magnitude means the square of minus 1 by root 6 which is 1 by 6 plus square of this 1 by 6 plus square of this which is 4 by 6. And you take the root of that and which is root of one that is it, so this is how you know you can show that this is orthonormal this is vectors.

(Refer Slide Time: 25:52)

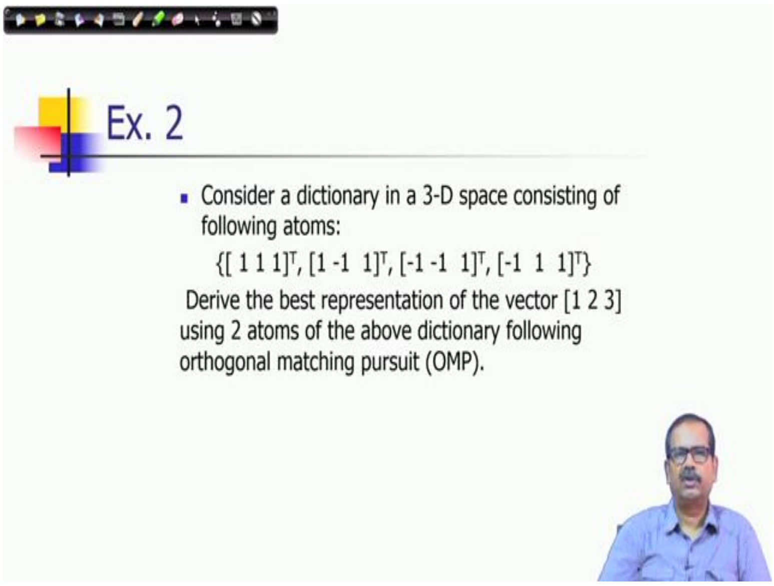


Considered the other example other exercise ; that means, how to represent a any arbitrary vector a vector 1, 2, 3 as a component. So, what we are doing you are taking the component of this vector 1, 2, 3 alone all these orthonormal basis vectors another basis vectors.

So, this is the component 2 by root 3 this is the magnitude 2 by root 3 along these vectors and 7 by root 6 along these vectors, 3 by root 2 along these vectors. So, finally, linear combination can be expressed as you can see that this provides you the coefficients of linear combinations. All these dot products they are providing the coefficients of this linear combinations, we have discussed this theory in my lecture in image transform, so you can revisit that lecture once again.

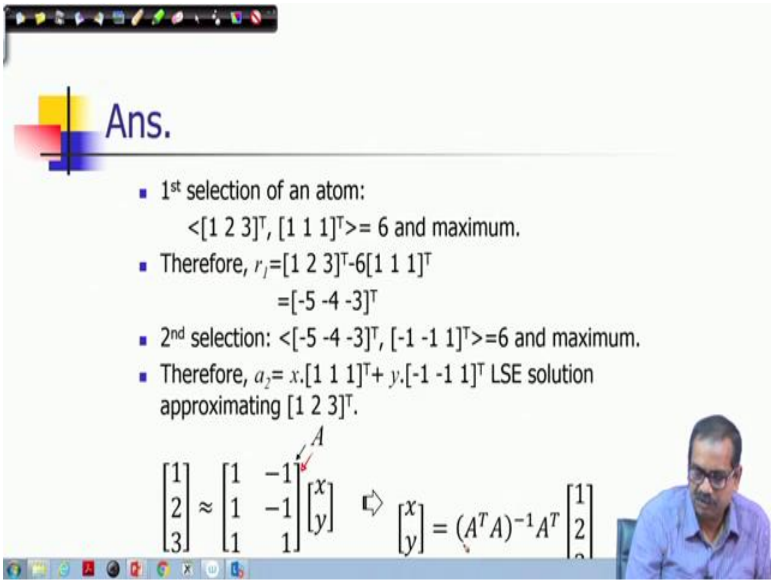
So, here the point is that if I give you an orthonormal set of basis vectors, then you do not require to do any complex operations what we discussed for matching pursuits. Simply you take the dot products with respect to any arbitrary vector and that value itself will give you the coefficients of the linear combination.

(Refer Slide Time: 27:11)



Let me take the other example, if I consider a dictionary in a 3 D space consisting of this atoms, so you have seen there are four atoms there in the dictionary of 3 d space and we can show that they are all linearly independent atoms. So, which means it is, so over complete set of dictionary as you know in 3 D space, 3 linearly independent vectors are sufficient to represent any arbitrary vector. So, derive the best representation of the vector 1, 2, 3 using two atoms of the above dictionary following orthogonal matching pursuit that is what our no problem statement is.

(Refer Slide Time: 27:49)



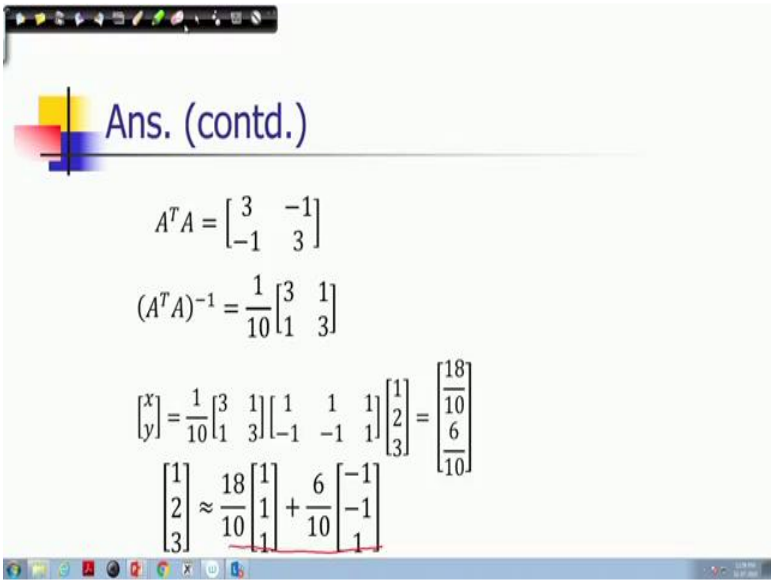
So, we will apply the orthogonal matching pursuit algorithm, first selection of an atom. So, find out the atom along which you get the maximum you know dot product value or component value. So, you take this dot product of these 2 and you get 6 and that is the maximum, you can check over other 3 atoms also it is a maximum. So, you select this atom for the representation which means this is the magnitude and the direction is 1, 1, 1 there is a vector.

So, the residue would be 1, 2, 3 minus 6 1, 1, 1 transpose I am for the convenience of representing in one slide I am using row vectors representation of this you know particular vectors. So, your residue at this stage becomes this vector, now the second selection. So, the second selection means now which atom from the remaining set of atoms one second you perform the dot products or with each of them.

And you can observe that minus 1 minus 1 1 that is giving you the maximum value of these dot products and that value is also 6 here. So, now, you have this in your dictionary you have to atoms 1, 1, 1 and minus 1 minus 1 1. So, use this a dictionary to obtain the best linear combination representation of the vector and you can use as you can what you can do that you from that best set of this is vector is B as we have done.

So, this is what is B and this is what is A in our representation and this is input this is what was X in our representation. And then oh here actually there is a some confusion of this, we are we are considering these values A these matrix is A here instead of N.

(Refer Slide Time: 30:06)



So, now, you get this is a solution, so to compute that A transpose A inverse you can compute. And you can compute the linear combination those coefficients X and Y using that this pseudo inverse by using pseudo inverse you can compute it. So, finally, your linear combination of the dictionary elements the best linear combination using two atoms for 1, 2, 3 is given this, so this is your required solution these the linear combinations that you are getting. So, with this let me stop at this point and we will continue this discussion for sparse representation in the next lecture.

Thank you very much for listening.

Keywords: Sparse approximation, orthogonal matching pursuit, basic pursuit.