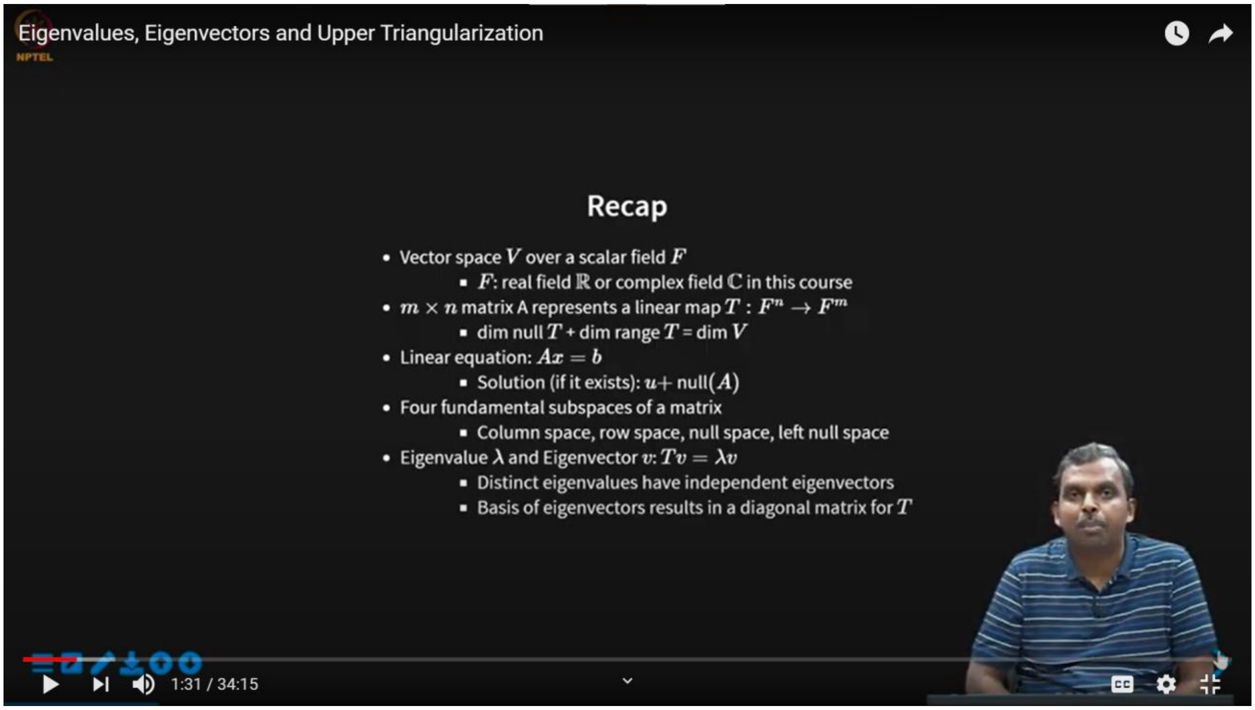
Applied Linear Algebra
Prof. Andrew Thangaraj
Department of Electrical Engineering
Indian Institute of Technology, Madras

Week 05
Eigenvalues, Eigenvectors and Upper Triangularization

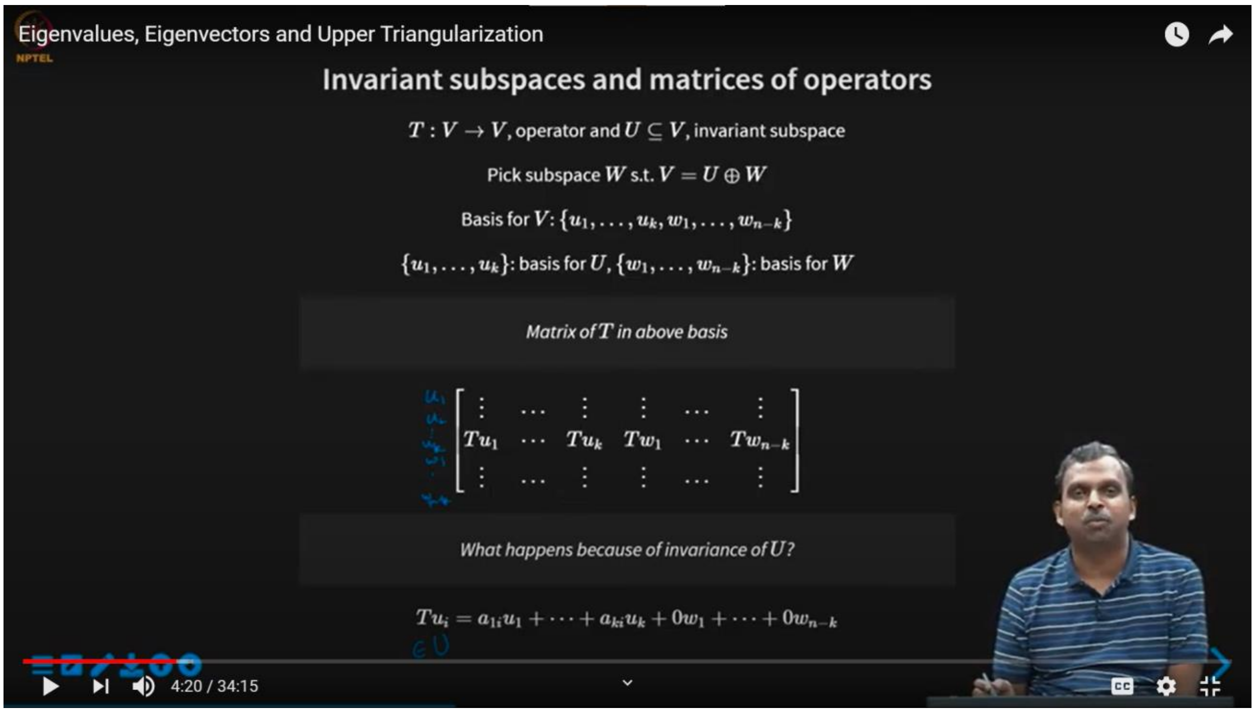
Hello and welcome to this lecture. We are now going to see this very interesting idea of eigenvalues, eigenvectors and upper triangularization. Towards the end of last lecture we saw that we may not always be able to get a diagonal matrix for a linear operator, there may be some limitations because you may not have enough eigenvectors. So, but it turns out upper triangularization is always possible. For any linear operator there is an upper triangular matrix. So that is the sort of the main result we will see. Along the way, I will also point you out other computational tools, numerical computational tools. In particular I will talk about SageMath. And notebook that I sometimes use with SageMath and then Matlab also little bit towards the end. So that is the agenda for this lecture. Let us get started.

(Refer Slide Time: 01:31)



So here is a quick overview. The last bullet probably is most important. Basis of eigenvectors results in a diagonal matrix for the operator and one sufficient condition for having a basis of eigenvectors is distinct eigenvalues. We will see that that is not the only, we saw already that that is not the only way, there are other matrices for which may be diagonalizable, which may have a basis of eigenvectors in spite of having repeated eigenvalues. We may not be able to completely characterize those things in this lecture. We will study more as we go along. But let us look at upper triangularization. This lecture is mostly about upper triangularization and that is possible for any operator, okay? So let us see that. So the crucial idea is the connection between invariant subspaces and matrices of operators. How do invariant subspaces affect the matrix of operators? We saw that when there is an entire basis of eigenvectors, the matrix becomes diagonal. But supposing you do not have that. You have just some invariant subspace, then what happens, okay?

(Refer Slide Time: 04:22)

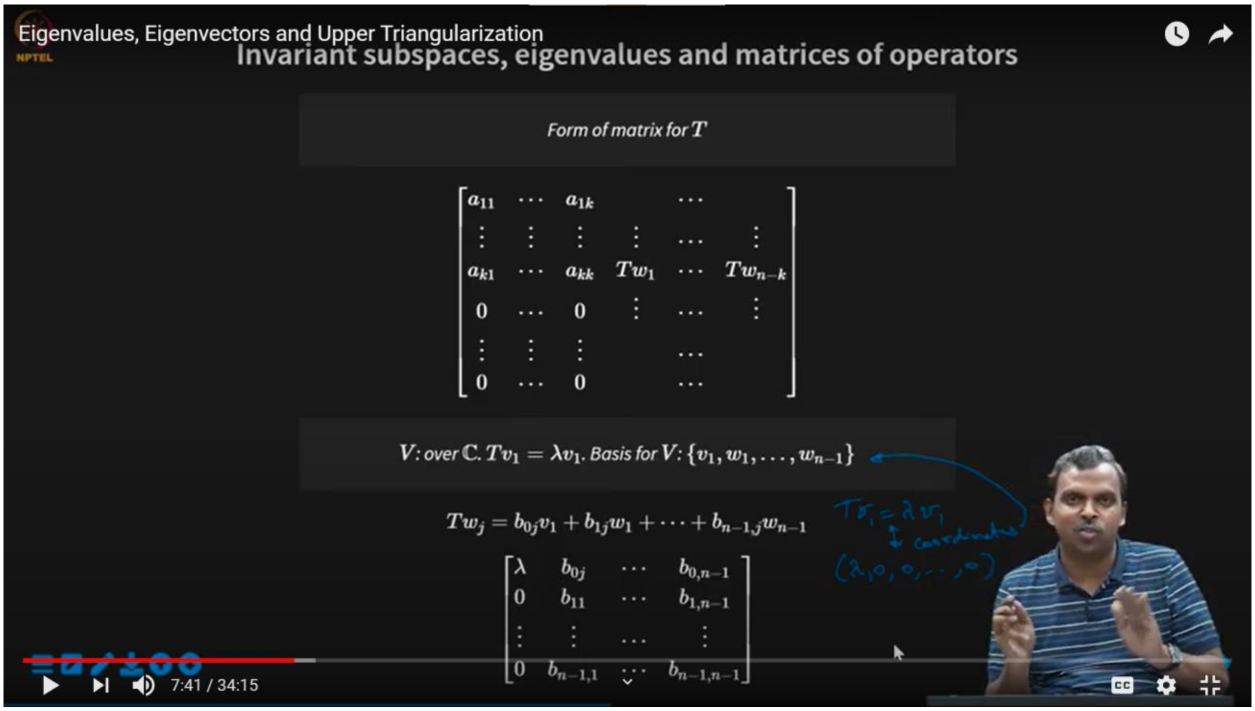


So here is an example I have taken. You have an operator $T : V \to V$ and then there is an invariant subspace for V, $U(V)$ is invariant under T of course. When I say invariant subspace, I mean invariant with respect to T. T invariant subspace, okay? We know we can pick a subspace W so that V is $U \oplus W$. We know that that's possible. We come up with the basis for V which will first have a basis for U and then the basis for W, right? So that is possible. We can do that also, okay? Because it is a direct sum. So now if you think of the matrix of the operator in this basis, you will get something slightly interesting. You will start seeing some zeros. Why is that? Because you know in this basis $u_1, u_2, \ldots, u_k, w_1, w_2, \ldots, w_{n-k}$, if you want to find the matrix, the first column is Tu_1 expressed in terms of this basis, right? So Tu_2 expressed in terms of this basis, so on you go up to u_k. And then w_1 to w_{n-k}. I am not saying anything new here. So now let us closely look at what happens because of the invariance of U. If you look at Tu_i, it will have coefficients from u_1 through u_k, there will be some coefficients a_{1i} multiplying u_1, a_{ki} multiplying u_k, okay? You could have. But because it is invariant, Tu_i belongs to U, right? Tu_i belongs to U. Since it belongs to U, it cannot have anything from W, okay? Because of this direct sum property. So it is, all these guys should be zero, okay? So these zeros have to show up for every u_i. Is that okay? So remember once again. See I am going to write coordinates for Tu_1 in terms of this basis. So in this matrix, this first guy will be u_1, second guy will be u_2, u_k. And then w_1 will come, w_{n-k} will be the last row, okay? Imagine, right? So every row you can associate with one of these basis elements. So that is the coordinate corresponding to u_1 in Tu_1, right? So that is how we write this matrix out. So all these bottom $n-k$ coordinates have to be zero in the first k columns, okay? That is because of the invariance of T. That is very nice, okay? So in invariant subspaces, if you pick the bases, they help you pick a basis in which at least quite a few zeros are guaranteed, okay? And this is the crucial idea and this was also used in eigenvectors and all that.

Here is the connection. I have written it down cleanly for you, right? So you have these as, which give you, you know, the invariant part of the, you know, operator. And then you have a bunch of zeros. After that you will have the $Tw_1, \dots, Tw_{n-k}$. I do not know what is going to happen there. When you do Tw_n, it may have something in U, it may have something in W also, right? Tw_1, okay, after I have transformed the W, basis of W with T, it could have both components in U and W also, I do not know. This could be a full matrix. But here I will definitely have zeros, okay? So that is the form for the matrix. Okay. If you assume that this vector space V is over the complexes, then I know that there is an eigenvalue, okay? So invariant subspace, this is for general invariant subspace. Let's focus on eigenvalues, right? Eigenvalues we know they exist definitely if you have complex fields. So let us say we have the complex field and there is an eigenvalue λ. Now this λv_1, this v_1 is a one dimensional, I mean… Span of v_1 is a one dimensional invariant subspace of T, right? So that's this relationship. We know that eigenvalues, eigenvectors correspond to one dimensional invariant subspaces, okay? So let us use this idea here, right? So you take the invariant subspace's basis v_1 and then pick w_1 through w_{n-1} to extend it, right? So you know pick a W which will, you know, give you a direct sum of V and you get a basis there, okay? So now let us say Tw_j, it will have something in v_1 and it will have w_1 through w_{n-1} also. But what about Tv_1? It will only have v_1, right? So Tv_1 will only have λv_1 so the coordinates in this basis, right, coordinates in this basis will simply be, you know, $(\lambda, 0, 0, \dots)$, okay? Same idea. So you have λ followed by a bunch of zeros. But here I will have lots of non-zeros, right? So nothing one can do about it. You will have lots of non-zeros, okay? So this is. I mean already is telling you how to do zeros.

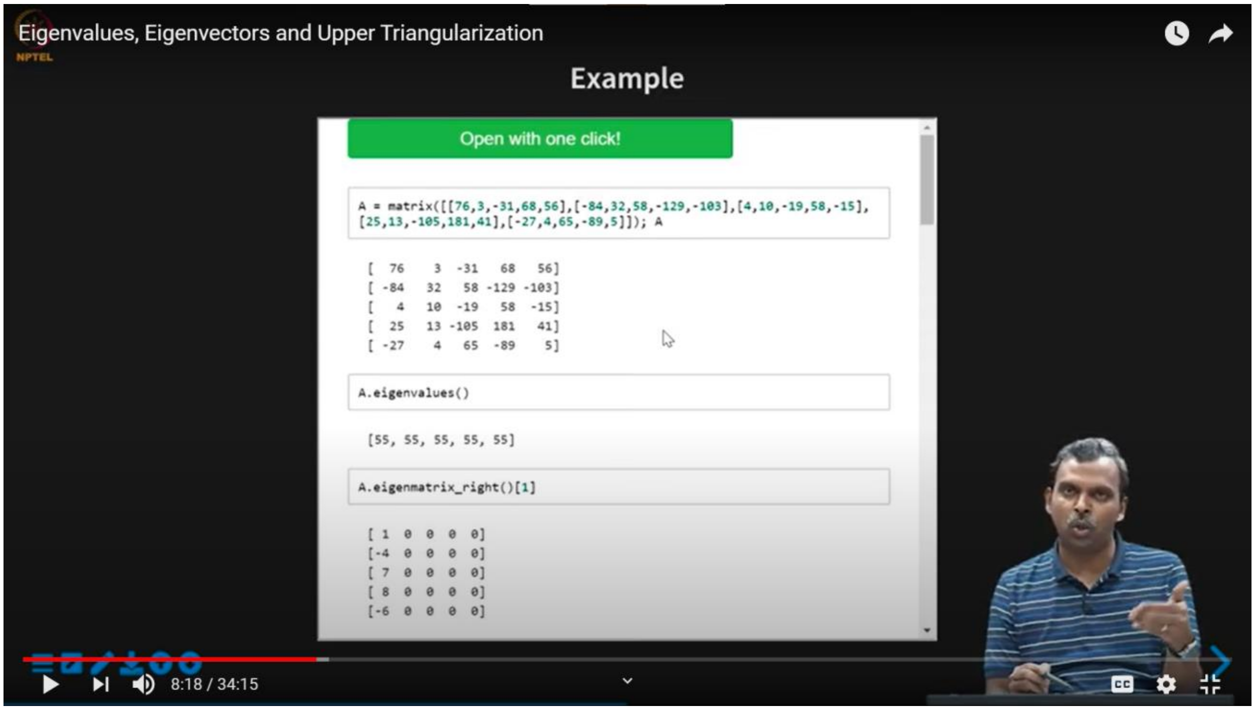
And let's attack this part separately, right? So we'll attack one by one. We have at least got a bunch of zeros in the first column. The way we want, remember, I am going towards upper triangularization. So this is a good step for me, okay? So let's see if we can do any further, okay? So as I describe this procedure, I want you to remember another upper triangularization that we did, okay? What is that? That is using elementary row operations, right? We were able to quickly and very easily using elementary row operations do upper triangularization. So here we cannot quite do that. When you do that then the basis changes, no? Input, output basis ends up changing, I don't want to do that. So I have to only do this eigenvalue business, okay? And only do change of basis. Every time I need to do a change of basis, I need to do a similarity transform, right? $S^{-1}AS$ that's the only thing I'm allowed. And whenever I do this, find a basis, find an eigenvalue and then do a change of basis, and... That's okay, change of basis is okay. Because I'm not changing, I'm changing basis on both sides, okay? If I do elementary row operations, then I'm changing bases only on one side and that is not, it's not what we are working with currently, okay? So this is interesting. But this process will be sort of like that, you know? You will go one after the other, sort of like pivoting and all that. But we are using similarity transforms, finding eigenvalues and doing things like that, okay?

(Refer Slide Time: 07:41)



Okay. So I promised I will show you a few other tools. And this is a tool which is a, it's a SageMath notebook called CoCalc. It's, I mean it also offers free accounts. You can also get paid accounts if you like. I use that sometimes when I want to use SageMath, okay? And this is syntax in SageMath. SageMath will have its own syntax. See, the thing about all these tools is every one of them has its own syntax, its own functions, they are all sort of similar, they will have the same sort of answers at the end of course. But you should get to know the tool more and more. And once you know one tool, usually, you know, other tools are also easy, okay? It's just the principles are the same, right? So it's just the names end up changing, okay? So SageMath for instance, you can declare a matrix like this. Now this is a very, very, very carefully curated matrix. It may not look like that to you, it might look like I have numbers all over the place. But this is a matrix I have very specially chosen, okay? So it is a very curated matrix. But, you know, you can also work with general matrices, you know? Even in SageMath there is a random matrix function, you can generate random matrices and work with that etc. and you can see how it works, right? So you can define a matrix, put a semicolon. I have to put A for it to print, okay? SageMath works like that. So that it prints. Otherwise you have to print it explicitly. You call that.

(Refer Slide Time: 08:18)

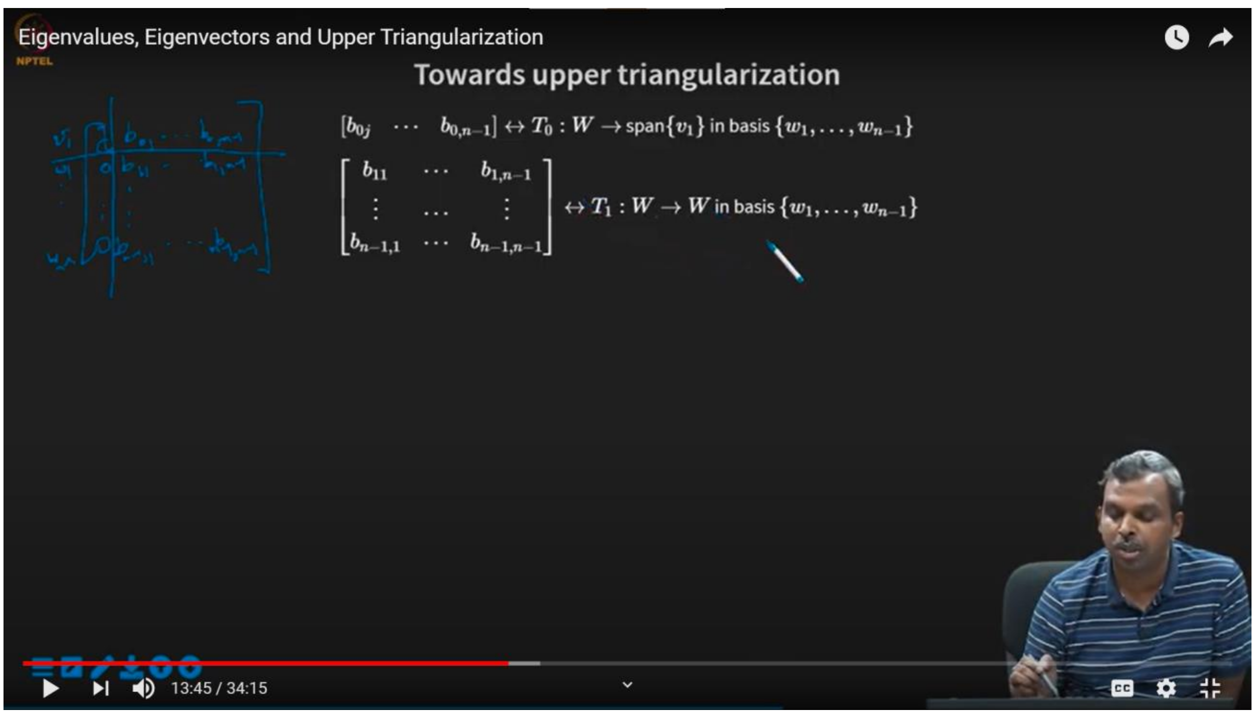


So this is my matrix. 5×5 matrix. In SageMath, A.eigenvalues() prints out all the eigenvalues. And you see eigenvalues are 55, 55, 55, 55, 55. This wouldn't have happened by magic, right? If I were to put random numbers here, it is very unlikely you will get 55, 55, …, okay? So this is again just an example to show you, right? If I put 55 on the diagonals and it's a diagonal matrix you will immediately believe me, that all the eigenvalues are 55. But here is a matrix which looks so different from the diagonal 55, right? All over the place there are numbers. But its eigenvalues are just 55, 55, …, okay? That's interesting. And then there is this, right, business that you have to do in SageMath, right? So if you... This command A dot, notice the structure, right? It thinks of A as some object and it puts methods on the objects. For those of you who have a background in computer science you'll see what I'm talking about. So it's sort of an object oriented sort of thing, okay? So it's a bit different. So this method, eigenmatrix_right, and then, you know, [1]. I mean, why [1]? Because eigenmatrix_right actually returns what python calls a tuple. I am interested in the second thing in the tuple. So that's [1]. And that's this matrix. So notice what has happened. There's only one eigenvector, okay? There are five eigenvalues, five repeated eigenvalues but there is only one eigenvector. Remember zero and all is not eigenvector, right? So there is only one eigenvector and it looks very simple, right? $(1\ -4\ 7\ 8\ -6)$, okay? So there's one eigenvector.

But I know there will be one eigenvector, right? Any matrix, I mean complex, okay? These are numbers. I know eventually I can go to complex numbers. So I will have one eigenvector, that is guaranteed, okay?

Once I have this eigenvector, what is it that I can do? I can make a change of basis. How do I choose, do a change of basis? I have to come up with this matrix S where every column is my basis, the new basis I want to go to. I have my first vector which is the eigenvector, okay? $(1\ -4\ 7\ 8\ -6)$. How do I pick the other ones? I simply extend, okay? I know how to extend, right? I just pick the standard basis $(0\ 1\ 0\ 0\ 0)$, $(0\ 0\ 1\ \ldots)$ I extend like this, okay? So this column_matrix is a construction in SageMath. It just lets you define one column at a time and put it all in one line, it's a simple way to construct a matrix. Again you see this is my S. I've prepared my basis transformation matrix. Once I do that, I have to implement my basis transformation and that is what I have done here. Again, there is this method called inverse. So if you do S.inverse(), you can see. I mean, those of you who have experience calling these methods you will see that open bracket close bracket clearly tells me there is a function here. So S.inverse() computes the inverse, multiplied by A, multiplied by S, you will get this matrix. And lo and behold, you know what's going to happen, right? So the first one was an eigenvector. So I got the first column 55, bunch followed by zeros. Remaining I do not know what it is, right? So it is all over the place. I get all sorts of entries, okay? So this shows you how this worked, okay?

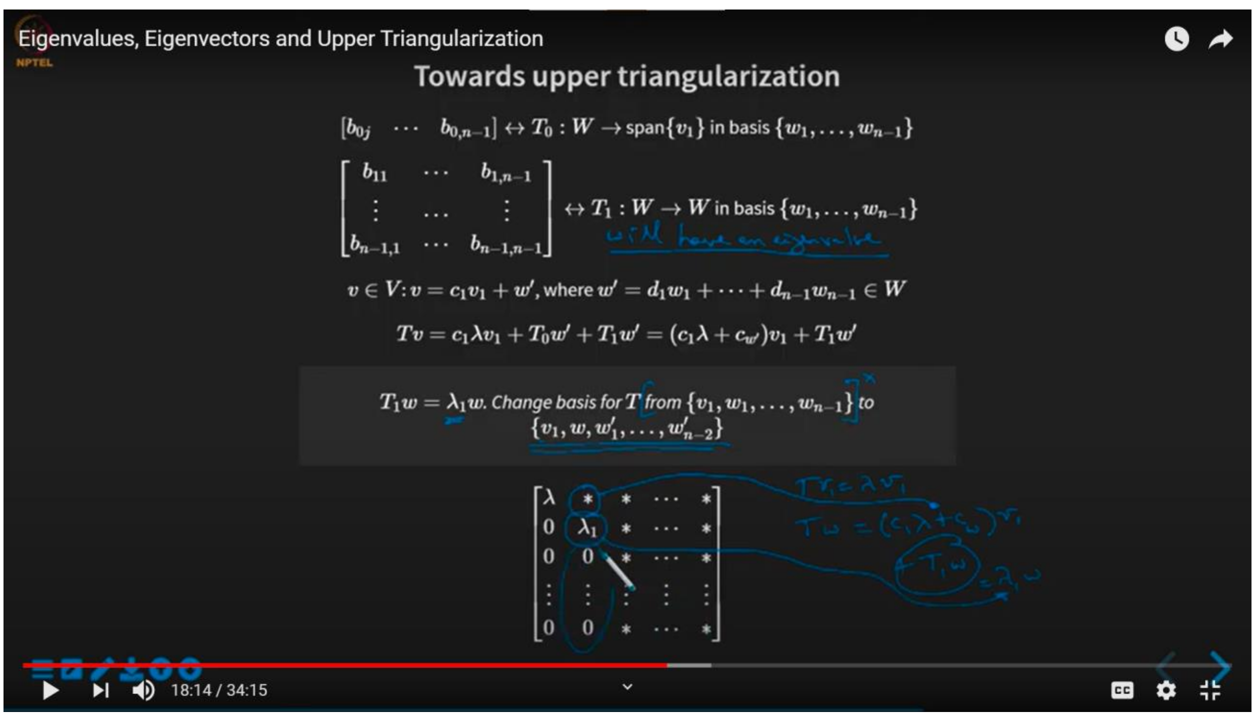
(Refer Slide Time: 13:50)



All right. So it turns out what we did for the first column we can continue, okay? I am going to come back to this sheet and show you and I have actually continued that. But let me first write down or show you, describe to you the principle behind how to continue this, you know, transformation beyond the first column, go to the second column, get a bunch of zeros there etc. Let's look at that next. Okay? So this is how you proceed towards upper triangularization. So we saw before that there was this first column which is λ... See remember, I have come to this state, right? I have come to my matrix being, you know, $(\lambda\ 0\ 0\ ..)$ and then there was this $(b_{01}, \ldots, b_{0,n-1})$ and then $(b_{11}, \ldots, b_{1,n-1})$, …, $(b_{n-1,1}, \ldots\ b_{n-1,n-1})$. So this is my matrix, right? Okay. So I am going to sort of split this like this, okay? So this corresponds to v_1 and these guys are all w_1 to w_{n-1}. So I am thinking of this matrix like this. I am taking the top part, okay, b_{0j} to $b_{0,n-1}$, this is a matrix, no? $1 \times (n-1)$ matrix. I am going to think of this matrix as representing some operator T_0 let us say from W to $span(v_1)$, okay? In the basis w_1 through w_{n-1}. I can do that, right? So it is not too bad. It is just a matrix. So in a subspace of dimension, it can take a subspace of dimension $n-1$ to a subspace of dimension 1. It's perfectly valid. Same thing I will do with this other square part also, okay. It's $(n-1) \times (n-1)$. So I will think of it as representing a map $T_1 : W \rightarrow W$ in that same basis, okay? So this is just a comfortable thing for me to do. Now what it lets me do is the following, okay? Notice what has happened. If I take any vector v in V, that v can be written as $v_1 + w'$ where this $w' \in W$, okay? So this we know because, you know, anyway these two are a direct sum so I should be able to do this, okay? So I can write like this. Now Tv is Tv_1, that is λv_1 plus Tw'. Now this Tw' one can write as $T_0 w'$ plus $T_1 w'$. You can see that that is what, that's what really happens here when I write Tv. I have $T_0 v$ happening on top taking me to the $span\{v_1\}$ and then this T_1 happening below which takes me to W. And these things, these two will add finally, right? So coordinates are going to add. So you have $T_0 w'$ plus $T_1 w'$, okay? Now what is $T_0 w'$? It took me to the $span\{v_1\}$. I don't know what it is, it will be some coefficient dependent on w'. But it will combine with the first term, it will give me the v_1, okay? And then I will be just left with $T_1 w'$. So this $T(v)$ can be written as something into v_1 plus $T_1 w'$. So that is the crucial part. So you take v, you split it into something in v_1 and w'. Tv also. This c_1 will transform into something in v_1 plus $T_1 w'$, okay? These are all linear transforms. You can see what I have done here. It's not something very fancy, just basic arithmetic and carefully written down, observing what is going on, okay? So this is sort of like quotienting an operator. Your book talks about it, I am not using that language, I am just briefly mentioning it for those who are interested. But this is just easy enough to describe in basic terms.

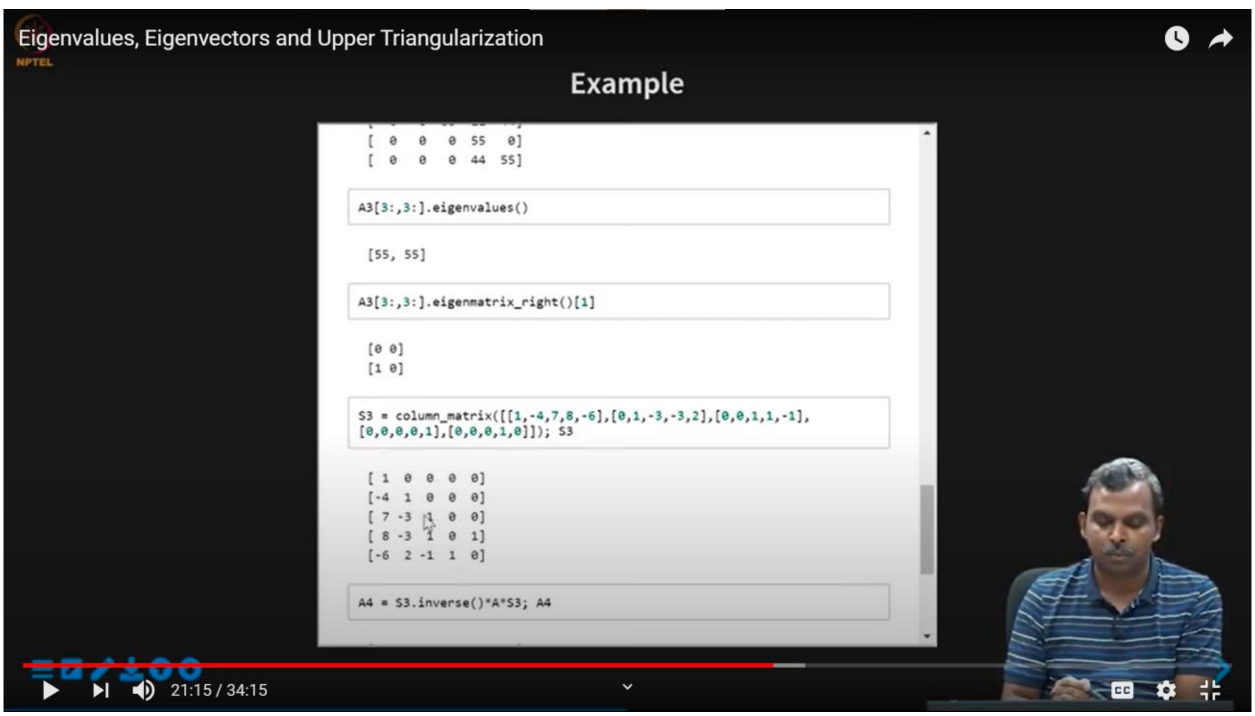
So what I can do with T_1 is: since I am in the complex field, T_1 will also have an eigenvalue, okay? That's what's very important. So T_1 will have an eigenvalue, okay, at least one eigenvalue is guaranteed, okay? Of course if you use determinants etc. you can do more things, but let us say we have one eigenvalue and that eigenvalue I am letting it to be λ_1 , okay? λ_1 is that eigenvalue and there will be an eigenvector w corresponding to it, okay? So there will be a w such that $T_1 w$ becomes $\lambda_1 w$, okay? So this is always true for any operator. So notice what is the change of basis that I am going to do now. See, previously I was in standard basis, some basis. I went to this basis $\{v_1, w_1, \ldots, w_{n-1})$, okay? So this is, yeah, so maybe from, okay this is okay, but I mean I do not need to go from here to here. Basically I need to change bases to this guy, okay? Forget about this from, it’s not so crucial, okay? So you can think of it like that. But it's not so crucial actually. In my opinion you can forget about it. You change the basis for T to this guy, okay? $\{v_1, w, w_1', \ldots, w_{n-2}'\}$. What are these things? See, v_1 is the eigenvector for T itself, the one eigenvector I found, okay? What is w? w is the eigenvector for T_1, okay? I found an eigenvalue for T_1 and its eigenvector is w. I take v_1, I take w and then I expand to the basis for the whole vector space V, okay? What will happen if I do that? You can go back and look at what has happened, okay? If this w' were to be an eigenvector, $T_1 w'$ will simply be $\lambda_1 w'$, okay? So you will get a form like this. So that is what is very important. So Tv_1, so in this basis Tv_1 becomes λv_1. What will be Tw? Remember Tw, w is an eigenvector, Tw will be something into, something into v_1, I do not know what that is, plus I will have a $T_1 w$. And what is $T_1 w$? $\lambda_1 w$, okay? So this guy simply becomes $\lambda_1 w$. So I will have something here corresponding to this term, okay? I do not know what that is, but the second term has to be λ and all these guys have to be 0 because it's an eigenvalue. So I have got my bunch of 0s here. So notice what I have done. I did it for T. I took care of the first column and then I shifted down, pivoted down into the bottom $(n-1) \times (n-1)$ part. And then applied the same idea again. I know I will get something on the top, I do not care, forget about it, but on the bottom I will get λ_1 followed by zeros, okay? And what you did here, you can continue over and over and over again. And you will end up in an upper triangular form, okay? So that is the crux of the upper triangularization idea, okay?

(Refer Slide Time: 18:16)



So hopefully that was clean enough. think about what I have done. So just exactly like you did elementary row operations, went from pivot to pivot, we are also doing that here except that in every stage I am finding the eigenvector, okay? So let me go back here to the sheet that I had. And we had done it for the first column, right? We got 55, bunch of zeros. Now let us move down to the next one, okay? So now in SageMath this notation A1[1:, 1:] will give me the bottom 4×4 matrix. One onwards, one onwards. The coordinates start from zero in SageMath, so one onwards for the x coordinate, one onwards for the y coordinate. So for the rows and columns. So you get this bottom 4×4. Its eigenvalues you see remarkably it's 55, 55, 55, 55 okay? So it's not very surprising because you know why it should work out like that, okay? And then if you look at the eigenvector matrix again, I have one eigenvector. $(1\ -3\ -3\ 2)$, okay? So now I will take this and make a change of basis for my A, okay? So notice what I have done here. I am creating an S1 which is the new basis I go to. The first column $(1\ -4\ 7\ 8\ -6)$, that is the original eigenvector that I had for the whole thing. The next one you see what I have done. I put $(1\ -3\ -3\ 2)$. Here I put zero. You can actually put what you like, but anyway let us say zero is good because it ensures linear independence and all that. And then I can extend, okay? I'm extending it to the basis for the whole vector space here, okay? And then I do just $S_1^{-1}AS_1$ and notice what has happened. I have got my 55 here and below that is all zeros, okay? Is that clear?

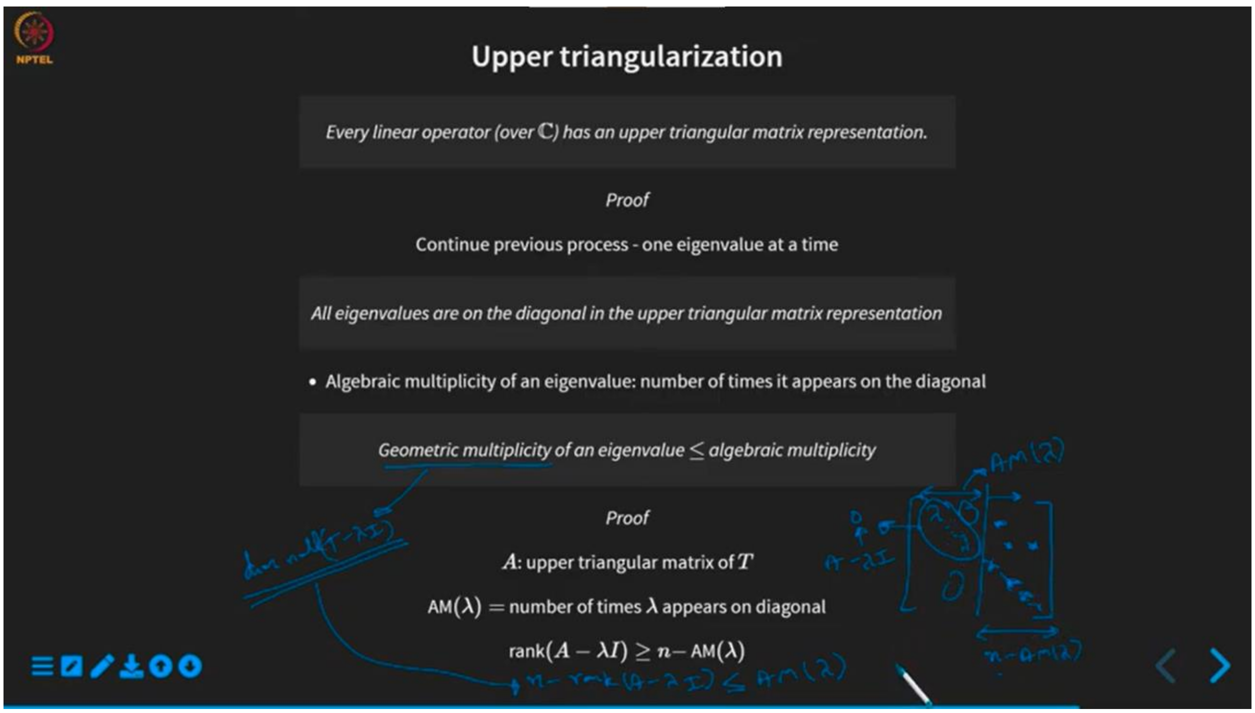
(Refer Slide Time: 21:16)



Now I repeat, okay, so I have come pivoted down to here, there is no problem with the pivot being 0 here, right? So because this is not like an elementary row operation. I have to take this matrix which is A2[2:, 2:]. That is that matrix. You can see the eigenvalues 55, 55, 55. And then you can find its eigenvector. It's $(1, 1, -1)$. And then I form my change of bases which is $(1\ -4\ 7\ 8\ -6)$, $(0, 1, -3, -3, 2)$ and then $(0, 0, 1, 1, -1)$ and then I have the usual extension. And then I get

S2. I get this. Again, I keep repeating, okay? Just the extension here you have to pay attention to. So here you got eigenvector 0, 1, so you got the 0, 1. The next extension cannot be 0, 0 it should be (0 0 0 1 0), okay? So just make sure the extension you pay attention to when you do this, when you construct this. S3, you will get the S3. Lo and behold, finally got an upper triangular matrix, okay? So if you keep repeating this operation over and over again, you will end up in an upper triangular matrix. So I found one similarity transform S3 which when acting on A, you know, $S_3^{-1}AS_3$ gave me an upper triangular matrix. So this is the upper triangular matrix which is equivalent to that other matrix, okay? So hopefully this SageMath... Once again you can click on this link, okay? It will give you a way to download the CoCalc notebook and you can open it and see it.

(Refer Slide Time: 26:21)



All right. So we saw the upper triangularization. So let's state it formally, right? Every linear operator over complex, okay, complex is needed because we need eigenvalues. It is not complex, you do not know if eigenvalues exist. Has an upper triangular matrix representation, okay? So the proof is to simply continue the previous process one eigenvalue at a time. I showed you how the diagonal occurs, how the zeros occur below the diagonal and you get an upper triangular matrix, okay? All eigenvalues are on the diagonal in the upper triangular matrix representation, that is very clear, okay? Algebraic multiplicity of eigenvalues, number of times it appears on the diagonal, okay? That we know. We know the geometric multiplicity now has to be less than or equal to algebraic multiplicity. Why is that? You take an upper triangular matrix of T, okay? And then if you look at algebraic multiplicity of eigenvalue λ, number of times λ appears on the diagonal, right? So you take... Okay, so just to proceed here, we finished with upper triangularization, we know what upper triangularization is. Now we are going to proceed and look at diagonalization. When will that upper triangularization become diagonalization, that is the sort of thing that we are going towards, okay? So for that, this algebraic multiplicity, geometric multiplicity and all of them are very important, okay? So once you get an upper triangular form, you will see proving all these things will become very easy, okay?

So how am I defining algebraic multiplicity? We defined it before also. How many times the eigenvalue repeats. Once you get an upper triangular matrix, the values on the diagonal are the eigenvalues, so you can easily find the multiplicity, you can define them, okay? Now what is geometric multiplicity? You might remember it is the, yeah, so it is basically... Let me just remind you what geometric multiplicity is, okay? So this geometric multiplicity is this number of linearly dependent eigenvectors, right? So in other words, it's dimension of $null(T - \lambda_i I)$, okay? So this is what it is. Geometric multiplicity, okay? So it turns out once you come up with the upper triangular matrix representation, it's easy to show geometric multiplicity has to be less than or equal to algebraic multiplicity, okay? Why is that? See, what is algebraic multiplicity? For simplicity, I will take the λs to be on top and below you have zeros and then you have a bunch of stars, right? So I know, I do not know what is going to happen here. I do not know what is going to happen here. But I have λs here, okay? How many λs do I have here? That is equal to the algebraic multiplicity of λ. How many ever times it appears, it appears. Now when I do $(A - \lambda I)$, what is going to happen? All these guys will become 0, okay? They will become zero when you do $(A - \lambda I)$, okay? So when they become zero, you can clearly see there will be a linear dependence on this side, right? So once they become zero, you have a linear dependence showing up, okay? So you can see rank of $(A - \lambda I)$, the rank becomes greater than or equal to... Did I get that right? Yeah. So greater than or equal to n minus AM of λ, okay? See when you have this λs subtracting, you will get a bunch of zeros here. So you have here n minus AM of λ, okay? That many rows and the rank has to be at least that, okay? Even from here, depending on these elements you may get some rank, additional rank may happen. But the zeros have been knocked out so you do not know if they will contribute to the rank. But the rank of this $(A - \lambda I)$ has to be at least n minus whatever you knocked out here, okay? Now that tells you that, you know, this is true, right? So why is that? So because n minus rank of $(A - \lambda I)$, okay, maybe I should write that down here. So n minus rank of $(A - \lambda I)$ is less than or equal to algebraic multiplicity of λ. And what is n minus rank? That is this guy, right? Dimension of $null(T - \lambda I)$, okay? So geometric multiplicity becomes less than or equal to algebraic multiplicity, okay? So notice how the, you know, the triangular matrix really simplified this for you, okay? So if this were not a triangular matrix, then just because you knocked out something on the diagonal, you do not know what happens to the rank. But below that zero is there and here all these guys are non-zero. So you know that the rank is at least this much, okay? So that's what's very useful in this result. So you can prove some nice results on geometric and algebraic multiplicity, okay? So this is useful. You will see this will play a nice role in the diagonalization.

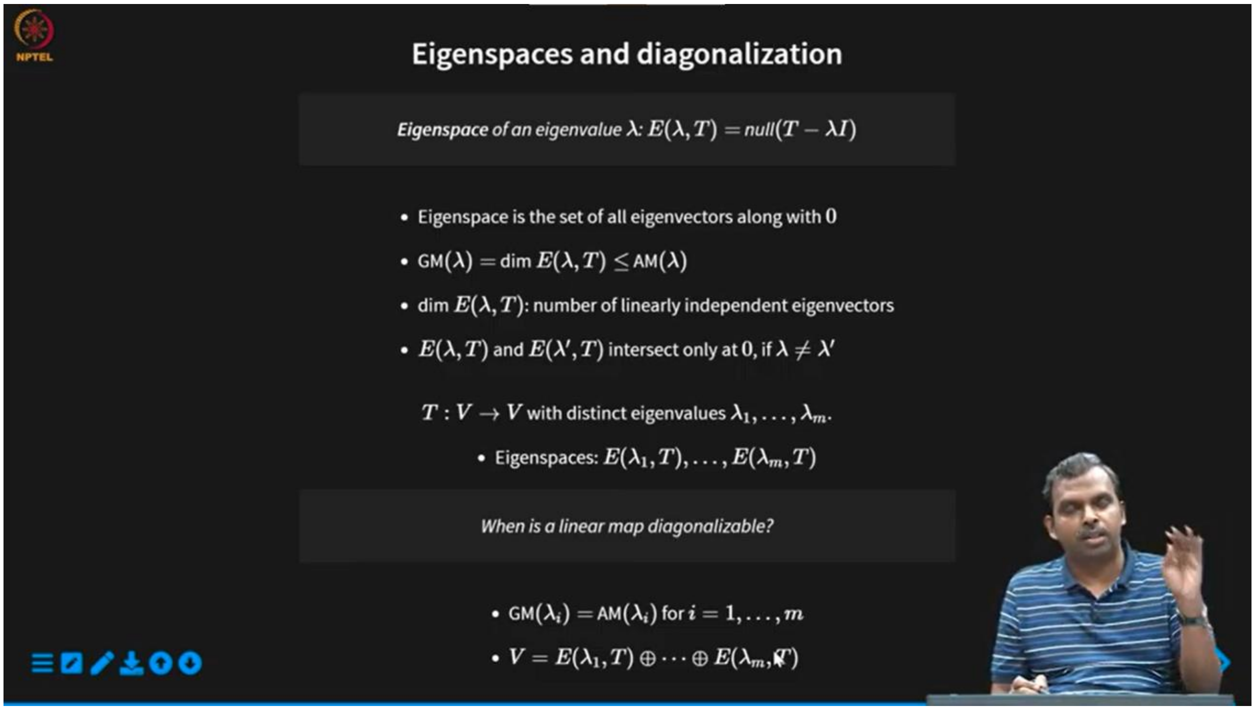
So now let us define something called an eigenspace and then connect it up with diagonalization. So we've been talking about this $null\ (T\ - \lambda I)$, $(T\ - \lambda I)$. Every time you have a λ, eigenvalue λ, this $null(T\ - \lambda I)$ has all the eigenvectors, right? So it's very natural to define it as the eigenspace. So this eigenspace is like the invariant subspace corresponding to the eigenvalue λ. And that's $null(T\ - \lambda I)$, okay? So that has a special notation we will call it $E(\lambda, T)$. So basically eigenspace is the set of all eigenvectors along with the zero guy also. So we see the geometric multiplicity is nothing but the dimension of the eigenspace. We have shown that it is less than or equal to the algebraic multiplicity, okay? So also notice dimension of this guy, this is geometric multiplicity, is the number of the linearly independent eigenvectors. Here is a very interesting result. Any two eigenspaces can only trivially intersect. Why is that? Because you know, if they have distinct eigenvalues, right, when they have distinct eigenvalues, then they can only intersect trivially because eigenvectors corresponding to distinct eigenvalues are linearly independent, okay? So you cannot have a non-trivial intersection there. And that would violate the linear independence of eigenvectors, okay? So that’s nice to see, okay? So let us say you have a linear operator T and it has m distinct eigenvalues λ_1 to λ_m, okay? The corresponding eigenspaces let us say are $E(\lambda_1)$ to $E(\lambda_m)$, okay? So I am looking at an operator, it has a bunch of eigenvalues, some of them may be repeated. Maybe there are m of them, each eigenvalue will correspond to an eigenspace.

Now we can give a partial answer to sort of, I mean sort of an interesting answer to, not actually partial answer, so good answer to: when is a linear map diagonalizable. It turns out a linear map is diagonalizable if and only if these conditions are true. The geometric multiplicity for every eigenvalue should be equal to its algebraic multiplicity, okay? If an eigenvalue is repeated so many times, there should be that many linearly independent eigenvectors. This is sort of like if and only if. This is not going to happen if any eigenvector is missing, right? If you have, it occurs many more times algebraically or on the diagonal of the upper triangular matrix but it does not give you enough eigenvectors, then you cannot diagonalize that operator, okay? There is no basis on which that operator will become diagonal. We haven't quite proved that fully in some sense. In this class I am not intending to. The textbook has a full proof, you can go take a look. But you can sort of intuitively see why this should be true, okay? So λ appears so many times and the number of times, number of linearly independent eigenvectors is not enough, then you are not going to be really be able to, you know, find all these one dimensional invariant subspaces, right? Which will all add up to the whole V, okay? You can notice here that when a linear map is diagonalizable, then the vector space becomes the direct sum of all the eigenspaces, okay? So that's what’s, that's central to making this happen, okay? So you have to have the geometric multiplicity being equal to algebraic multiplicity. And the vector space should be the direct sum of all these things.

So notice what will happen. The algebraic multiplicities will add up to n, right? They are on the diagonal. All the eigenvalues are on the, down the diagonal, so you add up the algebraic multiplicities you will get n. Now geometric multiplicity, if it is equal to that, some of the geometric multiplicities will also be n. And these eigenspaces do not intersect non-trivially, right?

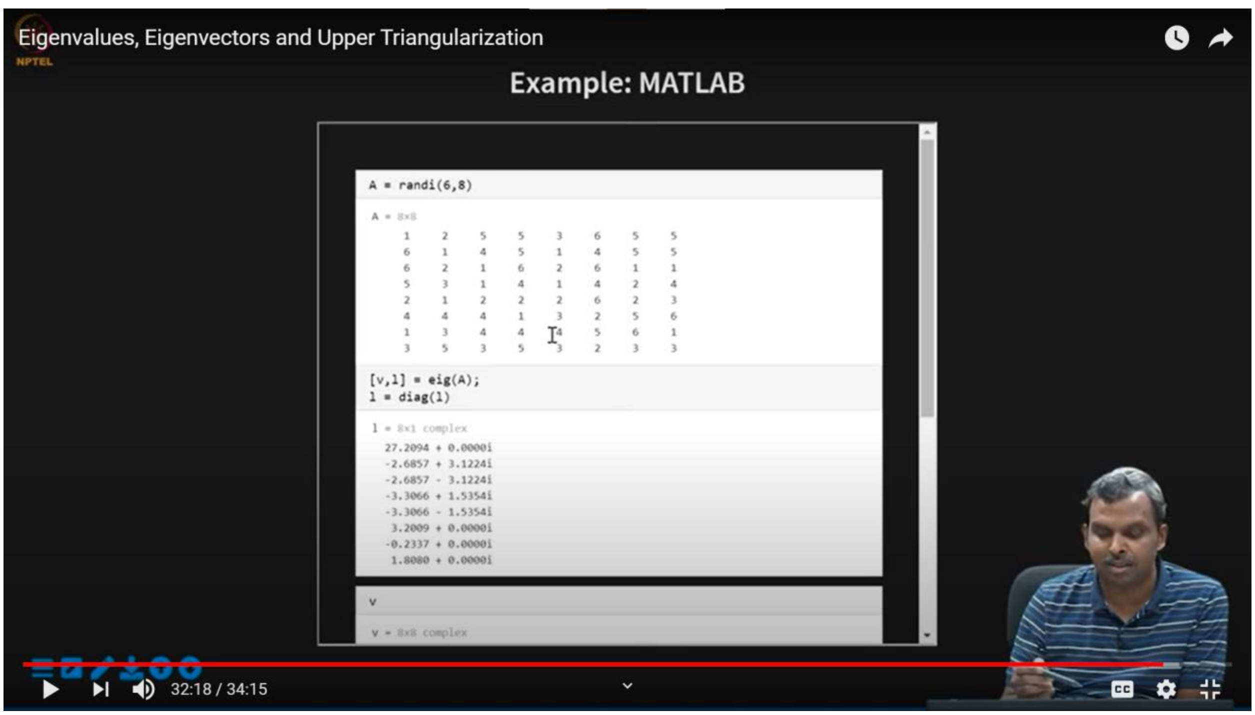
So they trivially intersect. So the direct sum of eigenspaces gives you V, okay? So this is a very nice picture to remember, okay? So when is a linear operator diagonalizable? There should be eigenspaces whose direct sum becomes equal to the entire vector space. In that case the operator is diagonalizable, otherwise it is not, okay? Another way of putting it is: geometric multiplicity should be equal to algebraic multiplicity for every eigenvalue, then it is diagonalizable, okay? So this sort of completes the picture of diagonalizable. You might naturally ask: what if it is not diagonalizable? What is the simplest form we can get? Maybe we will get to it later in the class if there is time, okay? But for now we will stop here. This is a good place to stop. So in general any operator you can upper triangularize there is no problem, okay? You will have an upper triangular representation. When will it be diagonalizable? If its geometric multiplicity is equal to the algebraic multiplicity, okay? So that sort of concludes this whole study of eigenspaces, eigenvalues from a diagonalization and upper triangularization point of view. We will study it further using more interesting properties etc. But for now we'll stop here.

(Refer Slide Time: 31:06)



But before I finish this lecture, let me show you the last computational tool that I promised to show you which is Matlab. A very basic few commands. Matlab is also very easy to use. You have this randi() command which will generate a random matrix for you. I put a 8x8 matrix. And there is this eig() command which will give you, you know, the eigenvalues along the matrix, diagonals of a matrix, that is why I put a diagonal. And then the eigenvectors in v. So you can see this matrix is a random sort of matrix and it has complex eigenvalues. Notice it has distinct eigenvalues. How many distinct eigenvalues? There are 1, 2, 3, 4, 5, 6, 7, 8. 8 distinct eigenvalues are there. So it is diagonalizable. I know that for sure. But it also has some complex eigenvalues. Notice complex conjugate eigenvalues are there, right? So it will be complex conjugate, no? You can also show that it will be complex conjugate eigenvalues. So that is easy to see because this is real, right? So $det(A - \lambda I)$ will be a real matrix, real polynomial. So it will have complex conjugate eigenvalues. So you can see complex eigenvalues showing up, okay? So in practice complex eigenvalues are very common, okay? So also notice another interesting thing. This one eigenvalue which seems to be relatively very large and all the other guys are sort of, you know, slightly small, that's also something interesting, okay? Now here are the eigenvectors. So you see the complex eigenvalues correspond to complex eigenvectors. The first one is a real eigenvalue, 27.20 something. And that corresponds to a real eigenvector, okay? So you can see that here. And the other eigenvalues, the complex eigenvalues correspond to a complex eigenvector, okay? And then likewise you will have other things. And if I do, you know, v inverse*A*v I get a diagonal matrix with the eigenvalues on the diagonal. So this is Matlab for you. I have not been able to share it quite so nicely. You can look at the commands here, okay, from the slides and then try it in Matlab also if you like. This is just a simple few lines, so it's not worth sharing in great detail anyway, okay?

(Refer Slide Time: 32:18)



All right. So let me once again remind you of what we looked at in this week's lectures. We looked at eigenvalues, eigenvectors, linear operators very closely, you know? How to compute them, how to, you know, work with them, how to think of diagonalizing an operator, how to get a diagonal matrix to represent an operator. The power of it we saw. We saw that upper triangularization is always possible but diagonalization in some cases, may be most cases, would happen. And I showed you also three computational tools that you can use as part of this course to try out, you know, matrices by hand. I would urge you to try it out just pick up some random matrices, look at eigenvalues, see what they tell you, look at row spaces, look at column spaces, look at rank, just get a feel for what they are really also, okay? Thank you very much. That's the end of the lectures for this week. We'll meet again next week. Thank you.