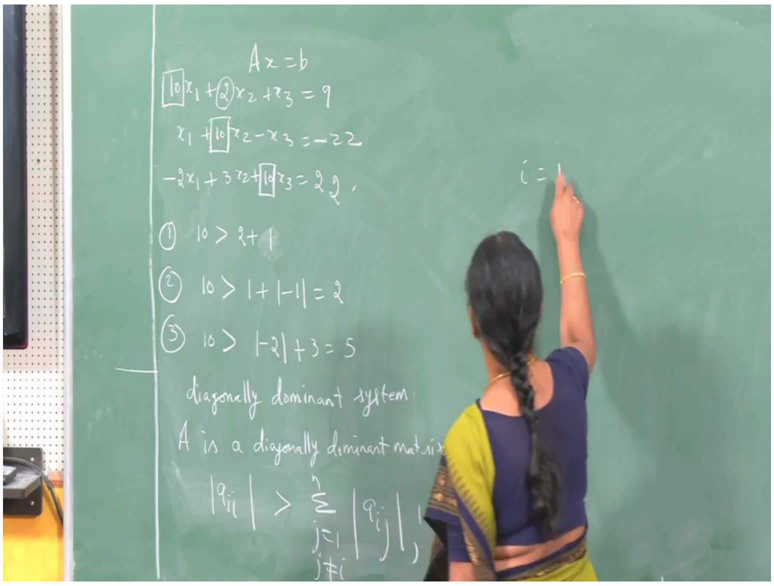
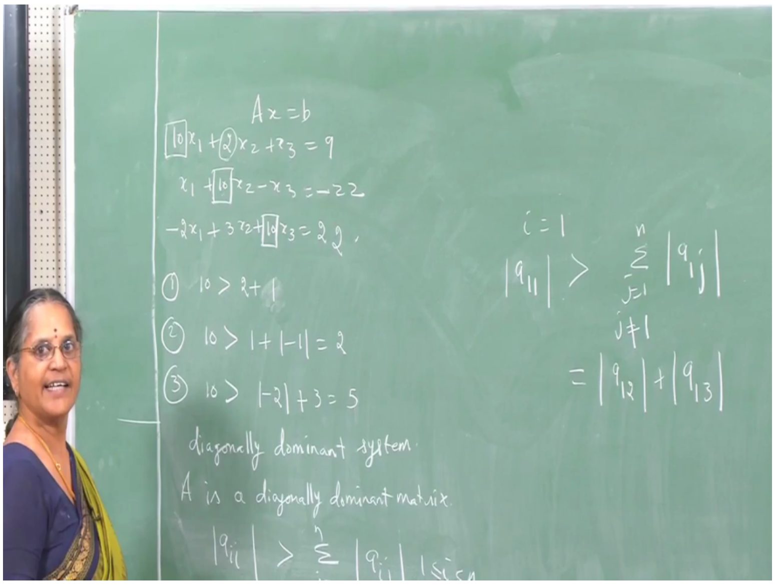
Numerical Analysis.
Professor R. Usha.
Department of Mathematics.
Indian Institute of Technology, Madras.
Lecture-46.
Solution of Linear Systems of Equations-8.
Iterative Methods-2.
Matrix Eigenvalue Problems-1.
Power Method-1.

(Refer Slide Time: 0:23)



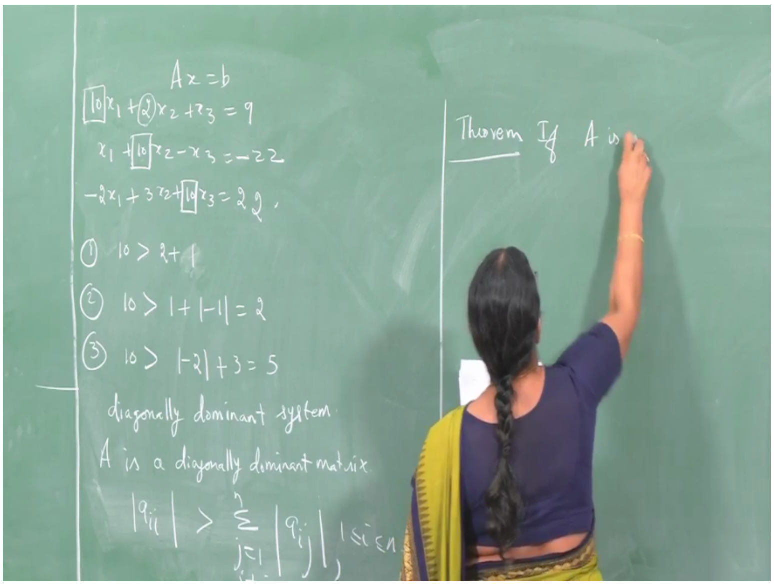


So we are given a system of equations, namely 3 equations in 3 unknown. We observe the following from this system, A is a diagonally dominant Matrix. The matrix A is the diagonally dominant Matrix and it has the property that modulus of A II is greater than Sigma J is equal to 1 to N, J not equal to I modulus of A IJ, for all values of I which run between 1 and N. Say for example if I take I as 1, then mod a11 is greater than Sigma J is equal to 1 to N, modulus of a1 J, so J should not take the value 1. So what is it, it is modulus of a12 + modulus of a13. So namely the coefficient of X1 which is a11 in absolute value is bigger than the sum of the absolute values of the coefficients which appear in the same equation.

So that happens for I equal to1, that should happen for I equal to 2, I equal to 3 and so on. If this property is satisfied for a matrix A, then we say that the matrix A is a diagonal dominant Matrix. If you are given a system of equations with the coefficient matrix as a diagonally dominant Matrix, then we have a result which says that if A is a diagonally dominant Matrix in the system AX is equal to B, then Gauss Jacobi method and Gauss Seidel method converge to the solution for any arbitrary starting vector for the unknowns X.

And here there given a system such that the coefficient matrix is the diagonally dominant Matrix. So we can start with any arbitrary initial vector X0 which is X1 0, X2 0, X30 and apply either Gauss Jacobi method or Gauss Seidel method, then our successive iterates are guaranteed to converge to the solution of the system. So we 1st write down the result and then work out the solution for this problem.

(Refer Slide Time: 3:17)



$Ax = b$

$\boxed{10}x_1 + 2x_2 + x_3 = 9$

$x_1 + \boxed{10}x_2 - x_3 = -22$

$-2x_1 + 3x_2 + \boxed{10}x_3 = 22.$

(1) $10 > 2 + 1$

(2) $10 > 1 + |-1| = 2$

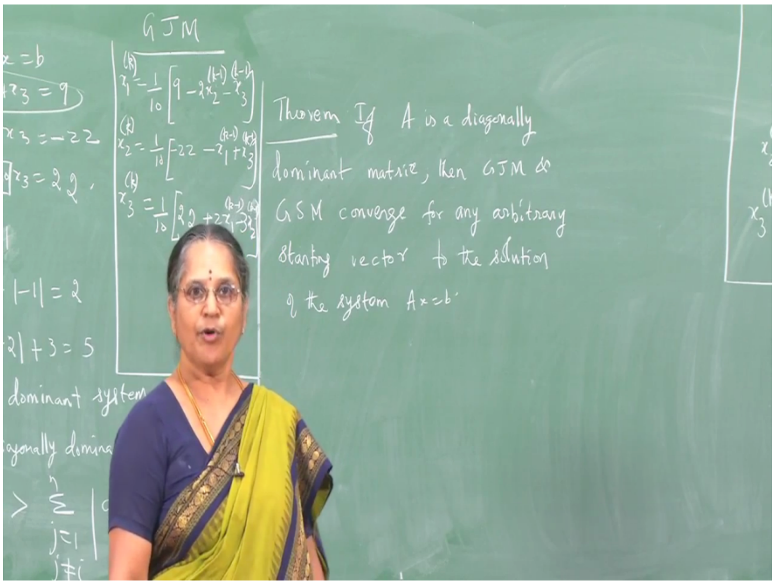
(3) $10 > |-2| + 3 = 5$

diagonally dominant system

A is a diagonally dominant matrix

$|a_{ii}| > \sum_{j=1}^{n} |a_{ij}|, \ 1 \leq i \leq n$

Theorem If A is

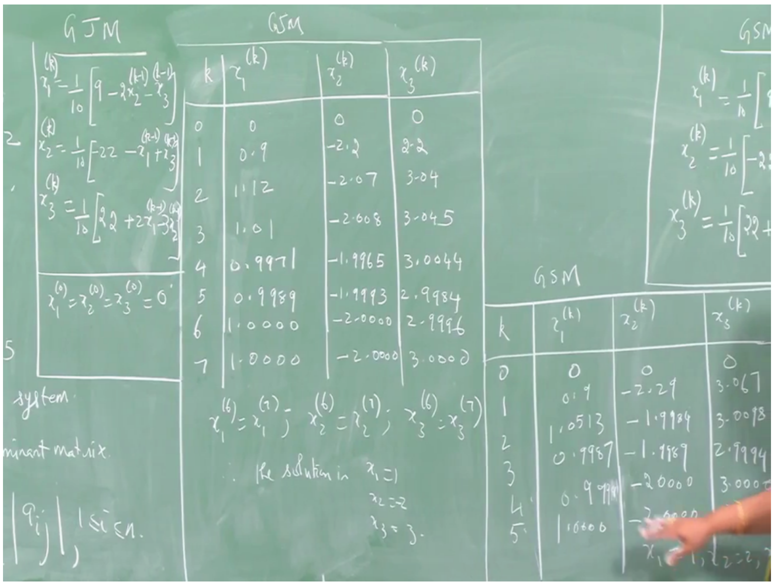

GJM
Theorem If A is a diagonally dominant matrix, then GJM & GSM converge for any arbitrary starting vector to the solution of the system Ax=b.
dominant system

So the results as and it is given by the following theorem. And it states that if A is a diagonally dominant Matrix, then Gauss Jacobi method and Gauss Seidel method converge for any arbitrary starting vector to the solution of the system A X is equal to B. So the theorem guarantees that the sequence of iterates generated by Gauss Jacobi method and Gauss Seidel method for this problem will converge to the solution of the system AX is equal to B. So let us 1st write down these methods, what should we do, I must associate the 1st equation with the 1st unknown.

And therefore X1 should be solved from the 1st equation and that gives you 1 by 10 into 9 - 2X2 - X3. X2 is 1 by 10, again, -22, - X1, + X3. Thirdly the X3 is 1 by 10 into 22+2 X1 -3 X2. So having written out the steps, if I want to apply the Gauss Jacobi method, my solution at the Kth step for X1 will be obtained by using the solution for K -1th step for X2 and the solution for 3 at the K -1th step. The solution for X1 at the Kth step will be obtained by using the values that the K -1th step for X2 and X 3. Similarly X2 K will be given by 1 by 10 - 22 - X1 at the K -1th step + X3 at the K -1th step.

And X 3 K is 1 by 10, mind 1 by 10 into 22+2 times X1 at the K -1th step + X2 at the K -1th step. And that this for the Gauss Jacobi method. What is it for Gauss Seidel method, so let us write down the details. This will be the iterative procedure that I will use it when I make use of Gauss Seidel method to generate the successive iterates. So we shall present the solutions for both these methods and see how the solutions look like.

(Refer Slide Time: 6:57)



GJM

$x_1^{(k)} = \frac{1}{10}\left[9 - 2x_2^{(k-1)} - x_3^{(k-1)}\right]$

$x_2^{(k)} = -\frac{1}{10}\left[-22 - x_1^{(k-1)} + x_3^{(k-1)}\right]$

$x_3^{(k)} = \frac{1}{10}\left[22 + 2x_1^{(k-1)} - 3x_2^{(k-1)}\right]$

$x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0$

system

minant matrix

$|a_{ij}|, 1 \le i \le n.$

GJM

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0	0	0
1	0.9	-2.2	2.2
2	1.12	-2.07	3.04
3	1.01	-2.008	3.045
4	0.9971	-1.9965	3.0044
5	0.9989	-1.9993	2.9984
6	1.0000	-2.0000	2.9996
7	1.0000	-2.0000	3.0000

$x_1^{(6)} = x_1^{(7)};\ x_2^{(6)} = x_2^{(7)};\ x_3^{(6)} = x_3^{(7)}$

∴ the solution in $x_1 = 1$
$x_2 = -2$
$x_3 = 3.$

GSM

$x_1^{(k)} = \frac{1}{10}\left[\right.$

$x_2^{(k)} = \frac{1}{10}\left[-2\right.$

$x_3^{(k)} = \frac{1}{10}\left[22+\right.$

GSM

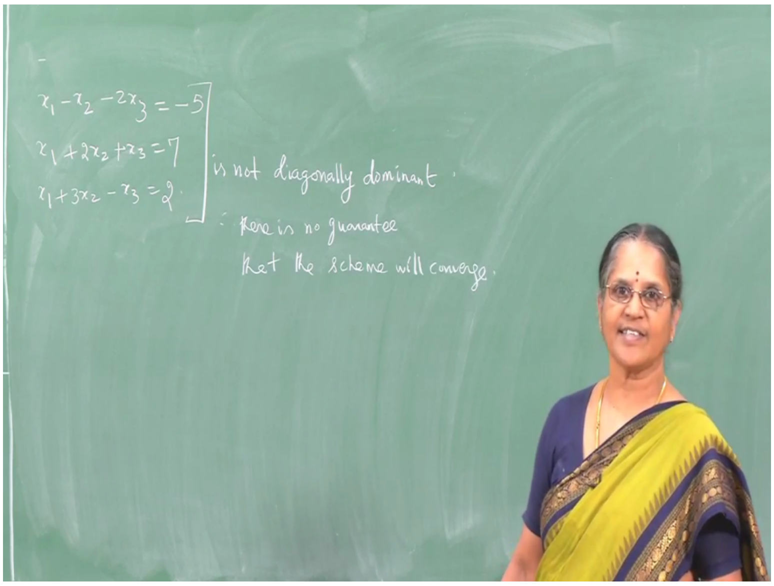
k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0	0	0
1	0.9	-2.29	3.067
2	1.0513	-1.9984	3.0098
3	0.9987	-1.9989	2.9994
4	0.9999	-2.0000	3.000
5	1.0000	-2.0000	[illegible]

$x_1 = 1, x_2 = -2,$

So the solutions using Gauss Jacobi method and Gauss Seidel method are presented here. We start with an initial approximation for X1, X2, X3 as 0. And apply Gauss Jacobi method everytime using the previously available values for the variables I have used and the values at this step are computed. When we perform our computations, we see that at the end of 7th iteration, we notice that the solution obtained at the 6th iteration and 7th iteration are the same for all the 3 variables. So I stop my computations and take the solution as X1 equal to 1, X2 equal to -2 and X 3 the solution using equal to 3. Now I perform the solution using Gauss Seidel method.

In this case the solution at any step is computed using the currently available values for the variables at that particular step. So when I do that, the solutions are given in this table. I again start with the initial vector 0 for all the 3 unknowns X1, X2, X3 and I perform 5 iterations and I observe that the solution turns out to be 1, -2 and 3 at the end of 5 iterations for X1, X2, X3 respectively. So the sequence of its rates generated by Gauss Seidel method converge faster than the sequence of iterates generated by Gauss Jacobi method.

(Refer Slide Time: 9:14)



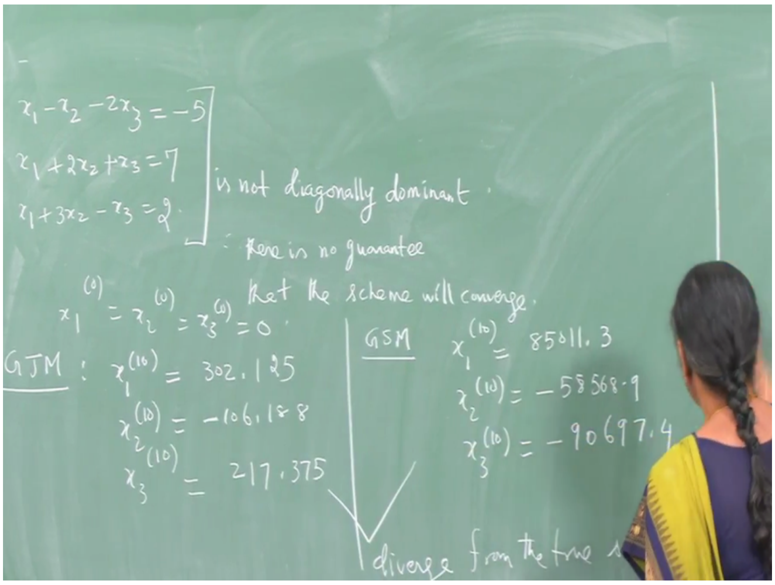
$x_1 - x_2 - 2x_3 = -5$

$x_1 + 2x_2 + x_3 = 7$

$x_1 + 3x_2 - x_3 = 2$

is not diagonally dominant.

∴ there is no guarantee that the scheme will converge.



$x_1 - x_2 - 2x_3 = -5$

$x_1 + 2x_2 + x_3 = 7$

$x_1 + 3x_2 - x_3 = 2$

is not diagonally dominant.

∴ there is no guarantee that the scheme will converge.

$x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0$

GJM : $x_1^{(10)} = 302.125$

$x_2^{(10)} = -106.188$

$x_3^{(10)} = 217.375$

GSM $x_1^{(10)} = 85011.3$

$x_2^{(10)} = -58568.9$

$x_3^{(10)} = -90697.4$

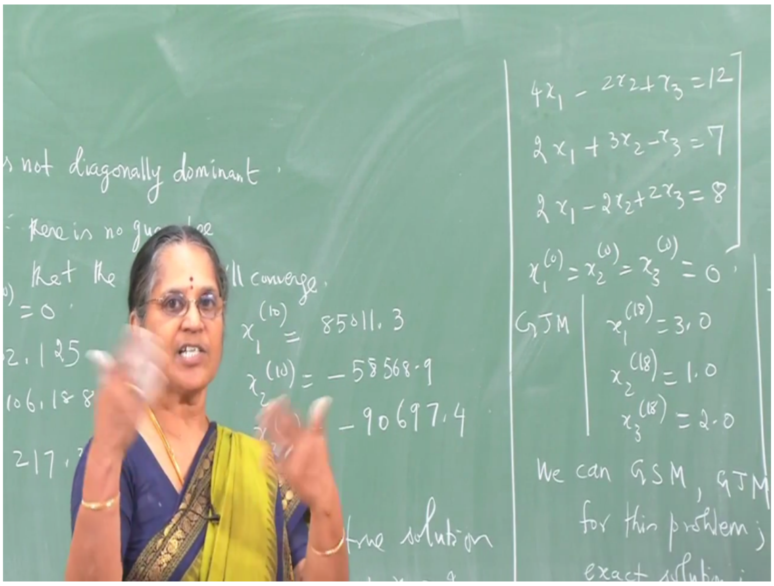
diverge from the true s

The question now is will the sequence of iterates generated by both these methods will always converge to the root of the equation. So let us look at some examples and given an answer this question. Let us consider the following example. The system is given by X1 - X2 - 2X3 equal to -5, X1 +2 X2 +X3 equal to 7 and X1 +3 X2 - X3 equal to 2. We see that the system is not diagonally dominant. And therefore there is no guarantee that the scheme will converge. The theorem that we stated said, if A the diagonally dominant Matrix, then the sequence of iterates generated by Gauss Seidel method and Gauss Jacobi method will converge to the solution of the system AX is equal to B for any starting initial vector.

So the conditions are sufficient and not necessary. And therefore in this case the matrix A which is the coefficient matrix is not diagonally dominant and therefore there is no guarantee that the scheme will converge. At the same time we cannot say that both the schemes will diverge, so let us work out the details and what we get starting with some initial vector. So if I start with an initial vector X10, X2 0, X30 equal to 0, then if I apply Gauss Jacobi method, that gives me at the end of 10 iterations the solution for X1 as 302.125 and X2 is -106.188 and X 3 at the 10th iteration turns out to be 217.375, X3 at the 10th iteration is 217.375, these are the values of X1, X2 and X3 at the end of 10 iterations.

Let us work out the same problem using Gauss Seidel method. And in this case at the end of 10 iterations we see that the solution for X1 is 85,011.3 and that for X2 is -58,568.9 and that for X3 is -90,697.4. So both these methods diverge from the true solution. Let us now consider another example.

(Refer Slide Time: 12:32)



Suppose say the system is 4 X1 - 2 X2 +3 is 12, 2 X1 + 3 X2 – X3 is 7 and 2 X1 -2 X2 +2 X3 is 8. So let us look at the system, we observe, so the system is not strictly diagonally dominant. So let us now try to work out starting from some initial vector and using Gauss Jacobi method and Seidel method. So again I start with the initial vector X1 0, X2 0, X30 to be 0. Then Gauss Jacobi method gives me at the end of 18 iterations, the solution to be X1 18 A3, X2 18 is 1 and X 3 at the end of 18 iterations is 2. And Gauss Seidel method also gives me the solution at the end of 15 iterations for X1, X2, X3 as 3, 1 and 2.

So we say that the Gauss Seidel method and Gauss Jacobi method converge for this problem, because in this case the exact solution for the problem is, X1 is equal to 3, X2 is equal to 1 and X3 is equal to 2. So just see Gauss Jacobi method and Gauss Seidel method, both have converged to the exact solution. So they generate sequence of iterates which converge to the exact solution of the problem and we observed that the matrix A is not a strictly diagonally dominant Matrix. So the conditions given in the theorem are sufficient and not necessary. So what is it that we infer from the result of the theorem and from the examples that we have taken.

Given a system AX is equal to B. If the coefficient matrix A is strictly likely dominant, then the sequence of iterates generated by both the methods will converge to the solution of the system A X is equal to B starting with any initial vector. If on the other hand given a matrix A, such that it need not be a diagonally dominant Matrix, the sequence of iterates generated by this matrix may either converge or diverge. So the conditions given in the theorem are sufficient but not necessary. So when we are asked to solve a system of equations, we check whether the given system is such that it has its coefficient matrix to be a diagonally dominant matrix.

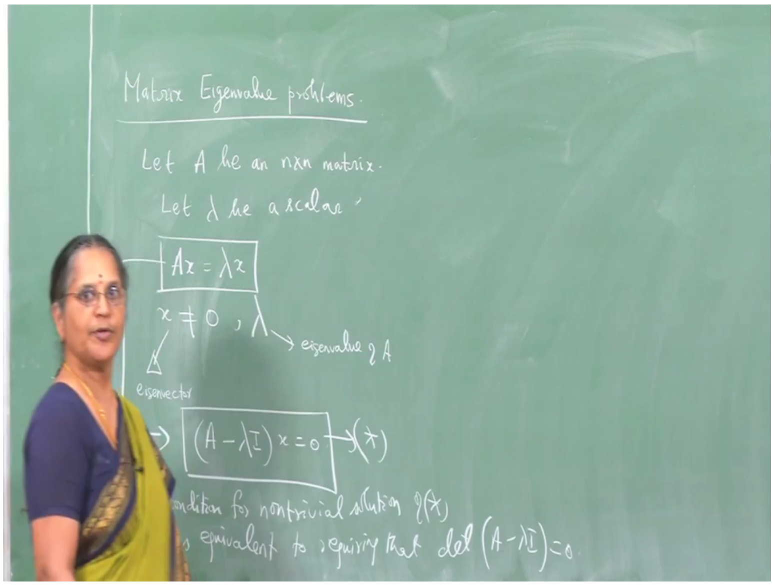
If suppose some exchange of equations can be made such that the coefficient matrix can be changed into a diagonally dominant Matrix, then you can apply both the methods and generate a sequence of iterates which will converge to the solution of the system of equations. If the matrix A that you are given is not strictly diagonally dominant, then nothing can be said when you apply your methods such as Gauss Jacobi method or Gauss Seidel method because the sequence of iterates may either converge or diverged for any initial starting vector. So this completes our discussion on the iterative methods for solving the system of equations.

So let us quickly summarise what we have done in this section of algebraic system of equations. So we said that using elementary operations we can convert a given matrix A to an upper triangular matrix and when we convert it into an upper triangular matrix, then back substitution method will enable us to solve the system immediately. So this was used when we described Gauss elimination method. Then we said that we may have to divide a particular equation by a very small quantity when we apply Gauss elimination method in which case the round off errors will become very large. In order to avoid that, as well as to avoid the situation where we may have to divide by a quantity which is the pivot and the pivot turns out to be 0 at that particular step.

Then in such cases the partial pivoting techniques must be used in Gauss elimination method so that we can solve, we can avoid division by 0 or division by a small quantity and the solution can be obtained. So summarising the methods for solving a system of equations, we said that we have 2 classes of methods, namely the direct methods and the iterative methods. And the direct methods comprise of decomposition methods, Gauss elimination method and Gauss Jordan method, whereas the iterative methods that we have learnt in this course are Gauss Jacobi method and Gauss Seidel method.

So given a system of equations, we can apply any one of these techniques depending upon and obtain the solution of the equation. And in all the cases, in both the cases we have a way to improve our solution, namely we can perform the iterative improvement method in the case of direct methods to get our solution to be as accurate as is required. And in the case of iterative methods, we can perform the iterations as many times as it is required, the solution is obtained correct to the desired degree of accuracy. So we close this section on the solution of system of algebraic equations and move on to the computation of eigenvalues and eigenvectors of a given matrix A.

(Refer Slide Time: 20:06)



Matrix Eigenvalue problems.

Let A be an n×n matrix.

Let λ be a scalar.

$\boxed{Ax = \lambda x}$

$x \neq 0$, $\lambda \rightarrow$ eigenvalue of A

eigenvector

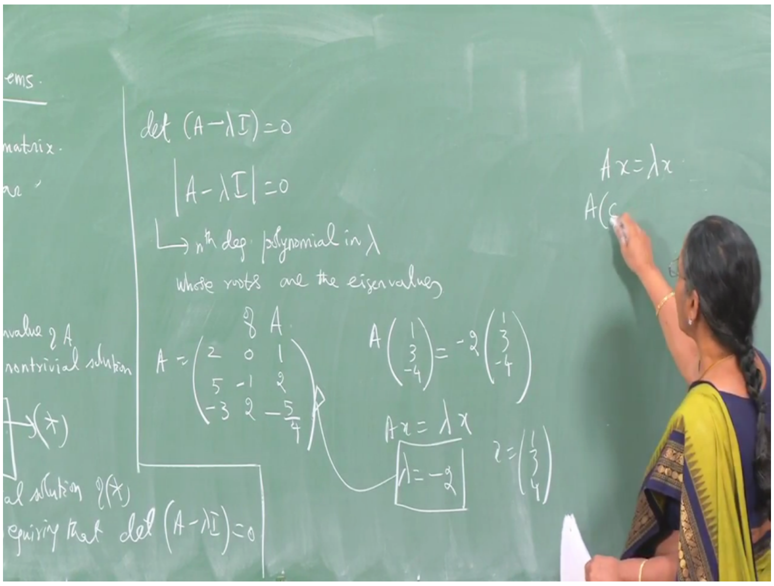
$\rightarrow \boxed{(A - \lambda I)x = 0} \rightarrow (*)$

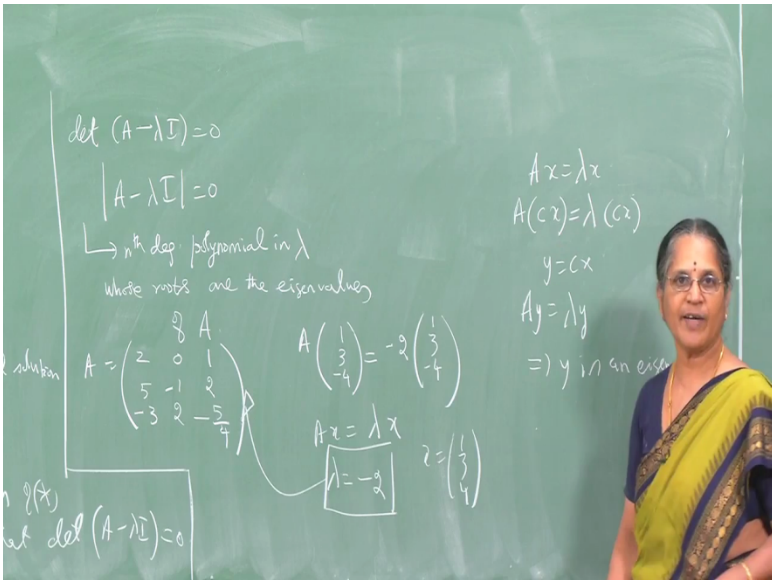
ondition for nontrivial solution of $(*)$

s equivalent to requiring that $\det(A - \lambda I) = 0$.

We consider matrix eigenvalue problems. So we are given a matrix A which is an N cross N matrix and let lambda be a scalar. Our goal is to solve an equation of the form AX is equal to lambda X. We seek a nonzero vector X and a scalar lambda which satisfies this equation A X is equal to lambda X. The scalar lambda is called an eigenvalue of the matrix A and the nonzero vector X which is associated with this eigenvalue lambda is called an Eigen vector of the matrix associated with the eigenvalue lambda. So solving this equation is equivalent to solving an equation of the form A - lambda I into X is equal to 0, which is a system of homogeneous algebraic equations.

(Refer Slide Time: 22:21)


ems
matrix
value of A
nontrivial solution
(*)
al solution of (*)
equiring that det (A-λI)=0
det (A−λI)=0
|A−λI|=0
↳ nth deg. polynomial in λ
whose roots are the eigenvalues
of A
A = (2 0 1 / 5 -1 2 / -3 2 -5/4)
A(1, 3, -4) = -2(1, 3, -4)
Ax = λx
λ = -2
x = (1, 3, 4)
Ax = λx
A(c


det (A−λI)=0
|A−λI|=0
↳ nth deg. polynomial in λ
whose roots are the eigenvalues
of A
A = (2 0 1 / 5 -1 2 / -3 2 -5/4)
A(1, 3, -4) = -2(1, 3, -4)
Ax = λx
λ = -2
x = (1, 3, 4)
Ax = λx
A(cx) = λ(cx)
y = cx
Ay = λy
=) y is an eigen

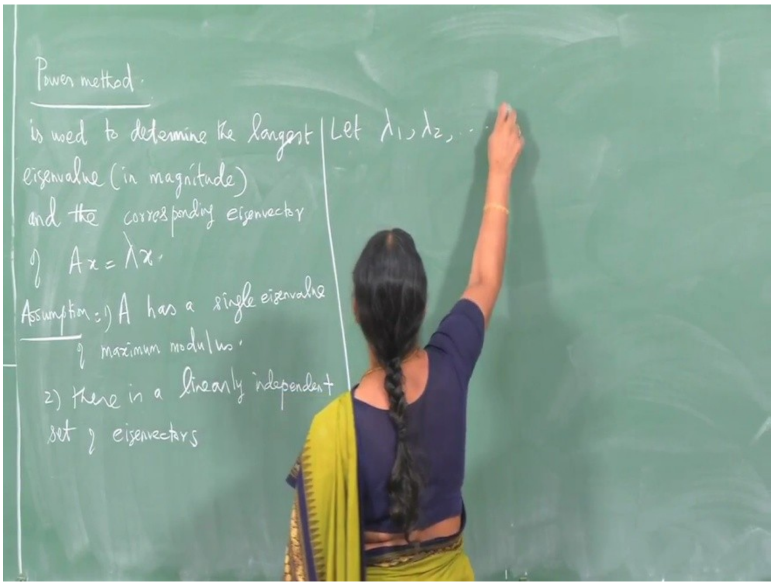
So the condition for a non-trivial solution of this system, let me call this as star, of star is equivalent to requiring that the determinant of the coefficient matrix, namely A - lambda I is equal to 0. So if you find determinant of A - lambda I and equate it to 0, you will observe that you have an Nth degree polynomial in lambda whose roots are the eigenvalues of the given matrix A. And once you determine these eigenvalues, you substitute each of those eigenvalues here and then solve the system A - lambda I into X is equal to 0 and you look for a nontrivial solution of this system corresponding to each of the eigenvalues which are roots of the equation determinant of A - lambda I equal to 0.

When you have done that, then you have solved the matrix eigenvalue problem. So let us consider a simple example by taking A to be in 3 by 3 matrix. Suppose say A is 2, 0, 1, 5, -1, 2, -3, 2, -5 by 4, I observe that A into X which is 1, 3, -4 is -2 times 1, 3, -4, so that this vector which has components 1, 3, -4 satisfies the equation AX is equal to lambda X. So lambda equal to -2 is an eigenvalue of this matrix A. If X is an Eigen vector associated with an eigenvalue lambda, so that AX is equal to lambda X, then if I multiply this vector by say a constant C, then I have A into CX is equal to lambda into CX.

So if I call Y to be CX, then I observe that AY equal to lambda Y and therefore Y is an Eigen vector. Because it satisfies the equation, AY is equal to lambda Y. So a constant multiple of an eigenvalue of a matrix A associated with the Eigen vector, eigenvalue lambda is also an Eigen vector of the matrix A. Now one can compute the eigenvalues and the corresponding eigenvectors by requiring that determinant of A - lambda I equal to 0, but the computations will be very tedious when the order of the matrix is very large. And therefore we require some numerical methods for computing the eigenvalues and eigenvectors of a given matrix A.

So we develop in this course a method known as power method which computes numerically the most dominant eigenvalue of a given matrix A. So you may ask me why only the most dominant eigenvalue and why not all the eigenvalues of a matrix. Of course numerical methods are available for computing all the eigenvalues of a given matrix numerically for special classes of matrices. In this course we focus only on the method where we are able to compute numerically the most dominant eigenvalue. The other methods which will help us to compute for special classes of matrices all the eigenvalues are beyond the scope of this course and so we focus only on the power method which helps us to compute the most dominant eigenvalue.

(Refer Slide Time: 26:49)



Power method.

is used to determine the longest eigenvalue (in magnitude) and the corresponding eigenvector of $Ax = \lambda x$.

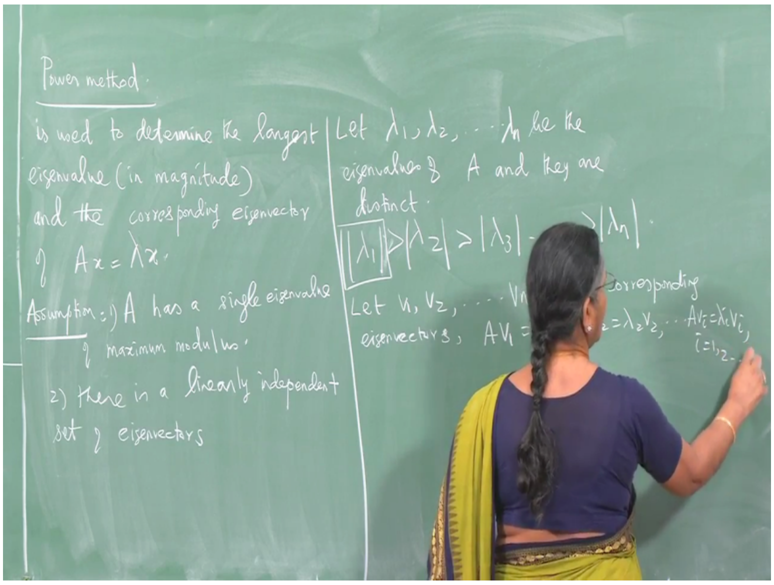
Assumption: 1) A has a single eigenvalue of maximum modulus.

2) there is a linearly independent set of eigenvectors

Let $\lambda_1, \lambda_2, \dots$

So we describe power method now. Power method is used to compute numerically the most dominant eigenvalue of a given matrix. And it also helps us to obtain the corresponding Eigen vector. So the method is based on the following assumption, namely the matrix A has a single eigenvalue of maximum modulus and that there is a linearly independent set of eigenvectors associated with the eigenvalues of a given matrix A. So under this assumption we compute the most dominant eigenvalue of the given matrix.

(Refer Slide Time: 28:37)



Power method.

is used to determine the longest eigenvalue (in magnitude) and the corresponding eigenvector of $Ax = \lambda x$.

Assumption: 1) A has a single eigenvalue of maximum modulus.

2) there is a linearly independent set of eigenvectors

Let $\lambda_1, \lambda_2, \dots \lambda_n$ be the eigenvalues of A and they are distinct.

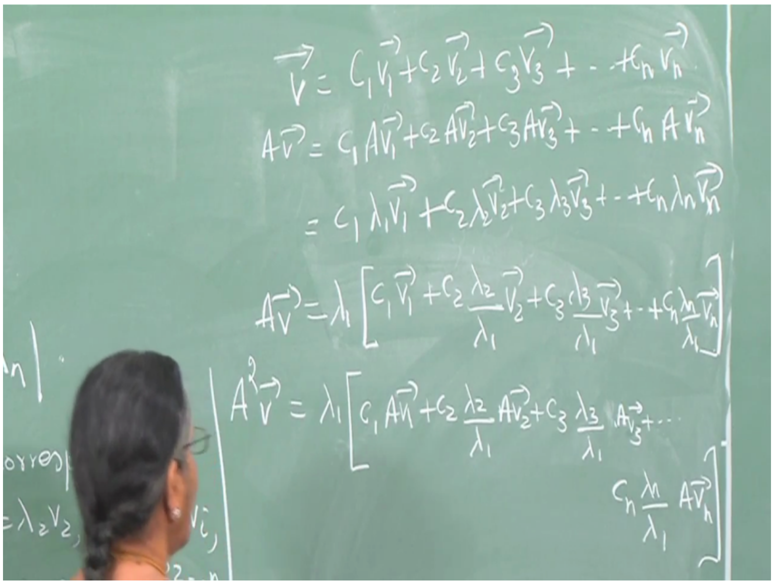
$\boxed{|\lambda_1|} > |\lambda_2| > |\lambda_3| \dots > |\lambda_n|$.

Let $v_1, v_2, \dots v_n$ … corresponding eigenvectors, $AV_1 =$ … $= \lambda_2 V_2, \dots AV_i = \lambda_i V_i$, $i = 1, 2 \dots$

So let us denote by lambda 1, lambda 2, etc, lambda N, the eigenvalues of the given matrix A and that they are distinct. And lambda 1 is greater than lambda 2 in absolute value is greater than mod lambda 3 and so on greater than mod lambda N. So lambda 1 is the most dominate eigenvalue in magnitude. So power method helps us to obtain this dominant eigenvalue numerically. So if suppose we denote by V1, V2, etc., VN the corresponding eigenvectors, what does that mean, it means A V1 is lambda 1 V1, A V2 is lambda 2 V2 are in general A V I is lambda I VI 4I is equal to 1, 2, 3 up to N.

Write down V if V is any vector, then it is, I use an arrow above V to indicate that it is a vector because they are solving a system AX is equal to lambda X, so X is a vector. If A is an N cross N matrix, X is an N cross 1 vector. So associated with lambda equal to lambda I, if we determine X I, that XI is denoted by V I which is in eigenvectors associated with the eigenvalue lambda I, so it is a vector. So any vector V can be expressed as a linear combination of the Eigen vectors associated with the eigenvalues lambda 1, lambda 2, etc. Any vector V will be of the form C1 V1 + C2 V2 + C3 V3 and so on + CN into VN.

(Refer Slide Time: 31:10)



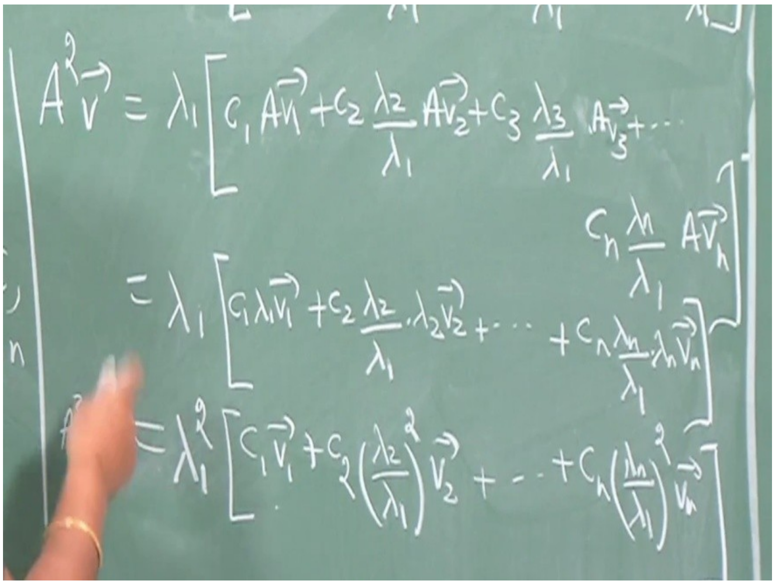
$$\vec{V} = c_1\vec{v_1} + c_2\vec{v_2} + c_3\vec{v_3} + \cdots + c_n\vec{v_n}$$

$$A\vec{v} = c_1A\vec{v_1} + c_2A\vec{v_2} + c_3A\vec{v_3} + \cdots + c_nA\vec{v_n}$$

$$= c_1\lambda_1\vec{v_1} + c_2\lambda_2\vec{v_2} + c_3\lambda_3\vec{v_3} + \cdots + c_n\lambda_n\vec{v_n}$$

$$A\vec{v} = \lambda_1\left[c_1\vec{v_1} + c_2\frac{\lambda_2}{\lambda_1}\vec{v_2} + c_3\frac{\lambda_3}{\lambda_1}\vec{v_3} + \cdots + c_n\frac{\lambda_n}{\lambda_1}\vec{v_n}\right]$$

$$A^2\vec{v} = \lambda_1\left[c_1A\vec{v_1} + c_2\frac{\lambda_2}{\lambda_1}A\vec{v_2} + c_3\frac{\lambda_3}{\lambda_1}A\vec{v_3} + \cdots \; c_n\frac{\lambda_n}{\lambda_1}A\vec{v_n}\right]$$

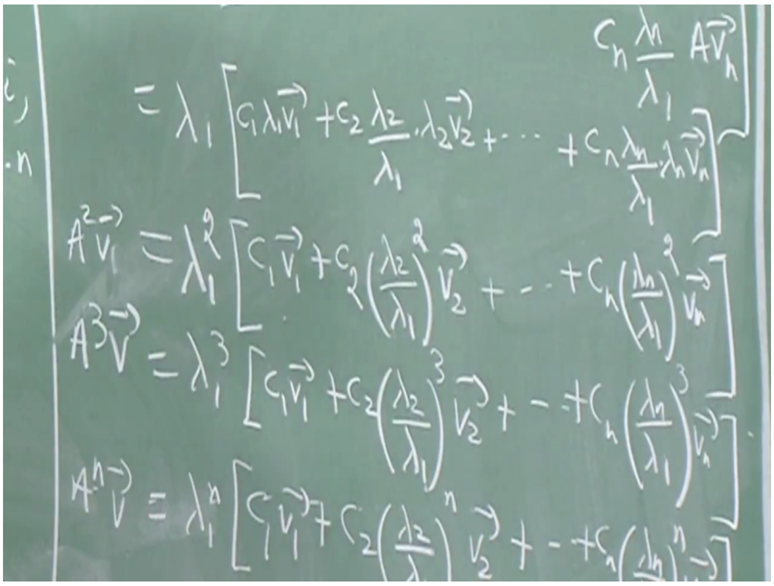

$A^2\vec{v} = \lambda_1\Big[c_1 A\vec{v_1} + c_2 \frac{\lambda_2}{\lambda_1} A\vec{v_2} + c_3 \frac{\lambda_3}{\lambda_1} A\vec{v_3} + \cdots + c_n \frac{\lambda_n}{\lambda_1} A\vec{v_n}\Big]$
$= \lambda_1\Big[c_1\lambda_1\vec{v_1} + c_2 \frac{\lambda_2}{\lambda_1}\lambda_2\vec{v_2} + \cdots + c_n \frac{\lambda_n}{\lambda_1}\lambda_n\vec{v_n}\Big]$
$= \lambda_1^2\Big[c_1\vec{v_1} + c_2\left(\frac{\lambda_2}{\lambda_1}\right)^2\vec{v_2} + \cdots + c_n\left(\frac{\lambda_n}{\lambda_1}\right)^2\vec{v_n}\Big]$

And so I compute A times vector V. So C1 is a constant, so it is A, C1 into A V1 + C2 into A V2 + C3 into A V3 + etc + CN + AVN. But what do we know about A V1, A V1 will be lambda 1 into V1, why, V1 is an Eigen vector are so stated with the eigenvalue lambda 1. So I substitute for A V2 as lambda 2 V2, A V3 as lambda 3 V3 and so on, A VN as lambda N VN. And therefore this will be equal to lambda 1 times C1 into V1 + C2 into lambda 2 by lambda 1 into V2, C3 into lambda 3 by lambda 1 into V3 and so on + CN into lambda 1 by lambda 1 into VN.

And so this will give me AV. Now I consider A on AV, that is I premultiply both sides of this matrix equation by A. So that will give me A square V, so that will be lambda 1 into C1 into A V1 + C2 into lambda 2 by lambda 1 into A V2 + C3 lambda 3 by lambda 1 into A V3, A V3 + etc. + CN into lambda N by lambda 1 into AVN. So I again make use of the fact that V1, V2, V3 are eigenvectors of matrix A associated with lambda 1, lambda 2, etc., lambda N. So A V1 will be lambda 1 V1 then + C2 into lambda 2 by lambda 1 into A V2 is lambda 2 V2 and so on + C N into lambda N by lambda 1 into A V1 is lambda N into VN.

And now I remove the factor lambda 1, so I get lambda 1 square into C1 V1 + C2 into lambda 2 by lambda 1 the whole squared into V2. Because I have removed a factor lambda 1 from it, so this will be lambda 2 by lambda 1, I already have that factor, so it is lambda 2 by lambda 1 the whole square. So I remove lambda 1 from this, that will give me lambda N by lambda 1 the whole square into vector VN. Now I see some pattern here, namely if I have A into V, that involves lambda 1 times C1 V1 + C2 into lambda 2 by lambda 1 into V 2 and so on, CN into lambda and by lambda 1 into VN.

(Refer Slide Time: 35:13)



$$C_n \frac{\lambda_n}{\lambda_1} A\vec{V}_n$$

$$= \lambda_1 \left[c_1 \lambda_1 \vec{v}_1 + c_2 \frac{\lambda_2}{\lambda_1} \lambda_2 \vec{v}_2 + \cdots + c_n \frac{\lambda_n}{\lambda_1} \lambda_n \vec{v}_n \right]$$

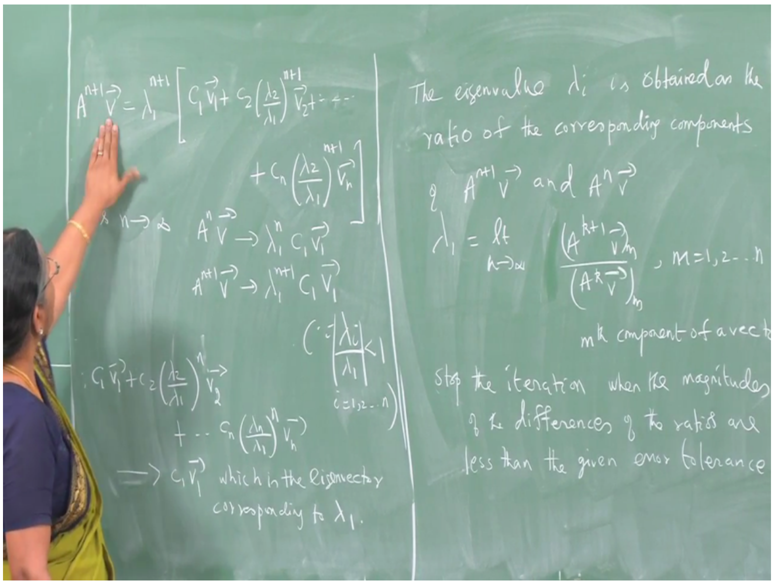
$$A^2 \vec{V}_1 = \lambda_1^2 \left[c_1 \vec{v}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^2 \vec{v}_2 + \cdots + c_n \left(\frac{\lambda_n}{\lambda_1}\right)^2 \vec{v}_n \right]$$

$$A^3 \vec{V} = \lambda_1^3 \left[c_1 \vec{v}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^3 \vec{v}_2 + \cdots + c_n \left(\frac{\lambda_n}{\lambda_1}\right)^3 \vec{v}_n \right]$$

$$A^n \vec{V} = \lambda_1^n \left[c_1 \vec{v}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^n \vec{v}_2 + \cdots + c_n \left(\frac{\lambda_n}{\lambda_1}\right)^n \vec{v}_n \right.$$

If I have H square times vector V, then that is lambda 1 square into C1 V1 + C2 into this + etc + CN into lambda N by lambda 1 the whole square into VN. So what do you think you will have when you compute A cube V, that is going to be lambda 1 cube into C1 V1 + C2 into lambda 2 by lambda 1 the whole cube into V2 + etc + CN into lambda N by lambda 1 the whole cube into VN. So you can continue to obtain every time premultiplying the equation at this step by A and obtaining say at the Nth step A to the power of N into V. So suppose we do that, then A power NV is equal to lambda 1 power N into C1 into V1 + C2 into lambda 2 by lambda 1 power N into V 2 + etc + CN into lambda N by lambda 1 to the power of N into VN.

(Refer Slide Time: 36:15)



$$A^{n+1}\vec{V} = \lambda_1^{n+1}\left[c_1\vec{v}_1 + c_2\left(\frac{\lambda_2}{\lambda_1}\right)^{n+1}\vec{v}_2 + \cdots + c_n\left(\frac{\lambda_2}{\lambda_1}\right)^{n+1}\vec{v}_n \right]$$

& $n \to \infty$ $\quad A^n\vec{V} \to \lambda_1^n c_1\vec{v}_1$

$$A^{n+1}\vec{V} \to \lambda_1^{n+1} c_1 \vec{v}_1$$

$$\left(\because \left|\frac{\lambda_i}{\lambda_1}\right| < 1 \quad i = 1, 2 \ldots n \right)$$

$$\therefore c_1\vec{v}_1 + c_2\left(\frac{\lambda_2}{\lambda_1}\right)^n \vec{v}_2 + \cdots c_n\left(\frac{\lambda_n}{\lambda_1}\right)^n \vec{v}_n$$

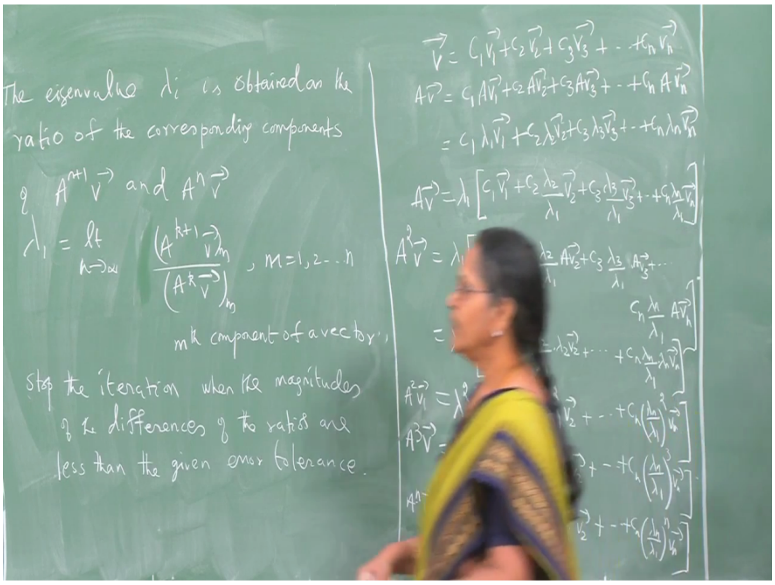
$\longrightarrow c_1\vec{v}_1$ which in the eigenvector corresponding to λ_1.

The eigenvalue λ_i is obtained as the ratio of the corresponding components of $A^{n+1}\vec{V}$ and $A^n\vec{V}$

$$\lambda_1 = \lim_{k\to\infty} \frac{(A^{k+1}\vec{V})_m}{(A^k\vec{V})_m}, \quad m = 1, 2 \ldots n$$

mth component of a vecto

Stop the iteration when the magnitudes of the differences of the ratios are less than the given error tolerance



The eigenvalue λ_i is obtained as the ratio of the corresponding components of $A^{n+1}\vec{v}$ and $A^n\vec{v}$

$$\lambda_1 = \underset{k\to\infty}{lt} \frac{(A^{k+1}\vec{v})_m}{(A^k\vec{v})_m}, \quad m = 1, 2 \ldots n$$

mth component of a vector

Stop the iteration when the magnitudes of the differences of the ratios are less than the given error tolerance.

$$\vec{v} = c_1\vec{v_1} + c_2\vec{v_2} + c_3\vec{v_3} + \ldots + c_n\vec{v_n}$$

$$A\vec{v} = c_1A\vec{v_1} + c_2A\vec{v_2} + c_3A\vec{v_3} + \ldots + c_nA\vec{v_n}$$

$$= c_1\lambda_1\vec{v_1} + c_2\lambda_2\vec{v_2} + c_3\lambda_3\vec{v_3} + \ldots + c_n\lambda_n\vec{v_n}$$

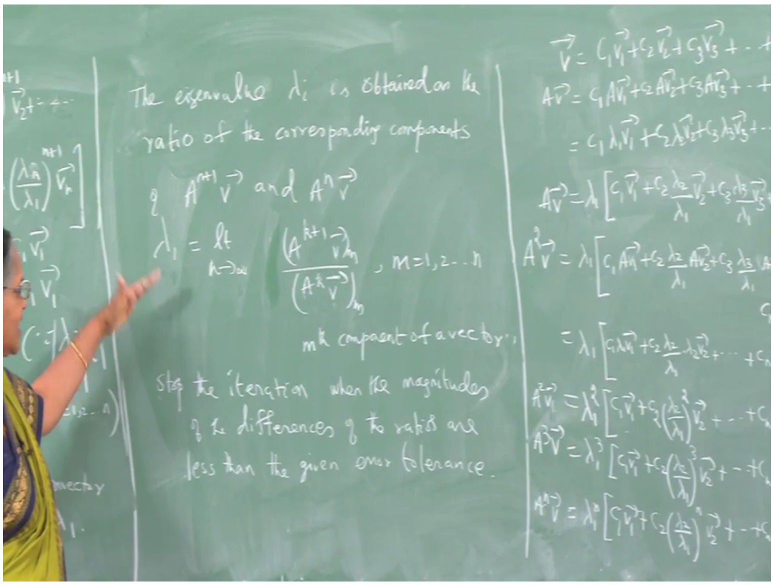
$$A\vec{v} = \lambda_1\left[c_1\vec{v_1} + c_2\frac{\lambda_2}{\lambda_1}\vec{v_2} + c_3\frac{\lambda_3}{\lambda_1}\vec{v_3} + \ldots + c_n\frac{\lambda_n}{\lambda_1}\vec{v_n}\right]$$

So at the end of N steps we have A power NV to be given by lambda 1 power N into this sum of vectors. Let us apply it once again. What do we do, we premultiply this by A, so we get A power N +1 into V and we pull out a factor lambda 1, so that will be lambda 1 power N +1 into the sum of these vectors within the bracket. Now let us take the limit as N tending to infinity. Then as N tends to infinity, we observe that the right-hand side vector tends to lambda 1 power N, C1 into V1. What happens to the rest of the terms lambda 2 by lambda 1 in absolute value is less than 1.

So as N tends to infinity of a quantity which is less than 1 is 0. So all these terms tends to 0 as N tends to infinity and the limit of A power N into V as N tends to infinity is lambda 1 power N into C1 into V1. Similarly when I take the limit as N tends to infinity of A power N +1 into V, then I will have lambda 1 power N +1 into C1 into V1 and rest of the terms which appear within the bracket will all tend to 0 because we have lambda 2 by lambda 1 is less than 1 and lambda N by lambda 1 is less than one in absolute value.

Now let us look at this vector, what did we get, C1 V1 + C2 into this + etc + CN into this, this vector tends to the vector C1 V1. Again the reason being in absolute value, lambda I by lambda 1 is less than 1 for I is equal to 1, 2, 3, etc., N. So as N tends to infinity, the vectors here are such that these terms will tend to 0 and we will end up with the vector C1 V1. So this is the Eigen vectors associated with the eigenvalue lambda 1. Then how do you compute the eigenvalue lambda 1 which is the most dominant eigenvalue? And that is obtained as a ratio of the corresponding components of these 2 vectors, namely A power N +1 V by A to the power of NV.

(Refer Slide Time: 39:27)



The eigenvalue λ_i is obtained as the ratio of the corresponding components of $A^{n+1}\vec{v}$ and $A^n\vec{v}$

$$\lambda_1 = \lim_{k\to\infty} \frac{(A^{k+1}\vec{v})_m}{(A^k\vec{v})_m}, \quad m=1,2\ldots n$$

mth component of a vector.

Stop the iteration when the magnitudes of the differences of the ratios are less than the given error tolerance.

$$\vec{v} = c_1\vec{v_1}+c_2\vec{v_2}+c_3\vec{v_3}+\ldots$$

$$A\vec{v} = c_1A\vec{v_1}+c_2A\vec{v_2}+c_3A\vec{v_3}+\ldots$$

$$= c_1\lambda_1\vec{v_1}+c_2\lambda_2\vec{v_2}+c_3\lambda_3\vec{v_3}+\ldots$$

$$A\vec{v} = \lambda_1\left[c_1\vec{v_1}+c_2\frac{\lambda_2}{\lambda_1}\vec{v_2}+c_3\frac{\lambda_3}{\lambda_1}\vec{v_3}\right.$$

$$A^2\vec{v} = \lambda_1\left[c_1A\vec{v_1}+c_2\frac{\lambda_2}{\lambda_1}A\vec{v_2}+c_3\frac{\lambda_3}{\lambda_1}\right.$$

$$= \lambda_1\left[c_1\lambda_1\vec{v_1}+c_2\frac{\lambda_2}{\lambda_1}\lambda_2\vec{v_2}+\ldots\right.$$

$$A^2\vec{v} = \lambda_1^2\left[c_1\vec{v_1}+c_2\left(\frac{\lambda_2}{\lambda_1}\right)^2\vec{v_2}+\ldots\right.$$

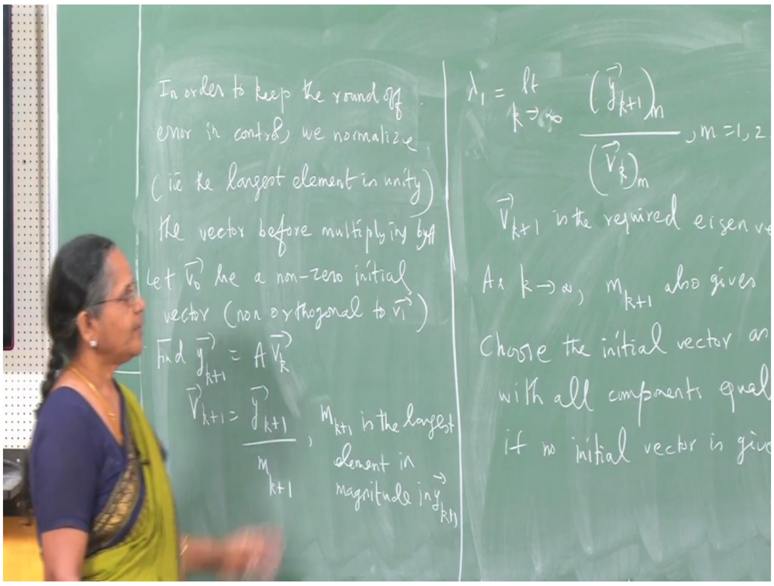
$$A^3\vec{v} = \lambda_1^3\left[c_1\vec{v_1}+c_2\left(\frac{\lambda_2}{\lambda_1}\right)^3\vec{v_2}+\ldots\right.$$

$$A^n\vec{v} = \lambda_1^n\left[c_1\vec{v_1}+c_2\left(\frac{\lambda_2}{\lambda_1}\right)^n\vec{v_2}+\ldots\right.$$

So that is lambda 1 power N +1 by lambda 1 power N, so that will give you lambda 1 and so lambda 1 will be equal to limit as N tending to infinity of A power K +1 into vector V by A power K into vector V but take the Mth component of each of these vectors where M takes values 1, 2, 3, etc. M. What does it mean, take the 1st component of this vector and take the 1st component of this vector and obtain the ratio of the 1st components of each of these vectors. And then similarly work out the details by computing the ratios of the Nth component of each of these vectors.

Lambda 1 is given by limit as N tending to infinity of the Mth comfort of the vector A power K + 1 V by the Mth component of the vector A power KV for N is equal to1, 2, 3 up to N. When do we stop the iterations, we stop the iterations when the magnitudes of the differences between the ratios that we obtain turn out to be less than the given error tolerance, so that we obtain lambda 1 correct to the desired degree of accuracy and the vector C1 V1 to which the sum of these vectors converge to as N tends to infinity will give us the corresponding to Eigen vector.

(Refer Slide Time: 41:24)



In order to keep the round off error in control, we normalize (ie the largest element in unity) the vector before multiplying by A

Let $\vec{v}_0$ be a non-zero initial vector (non orthogonal to $\vec{v}_1$)

Find $\vec{y}_{k+1} = A\vec{v}_k$

$\vec{v}_{k+1} = \dfrac{\vec{y}_{k+1}}{m_{k+1}}$, m_{k+1} is the largest element in magnitude in $\vec{y}_{k+1}$

$\lambda_1 = \underset{k\to\infty}{lt} \dfrac{(\vec{y}_{k+1})_m}{(\vec{v}_k)_m}$, m =1, 2

$\vec{v}_{k+1}$ is the required eigen ve

As $k\to\infty$, m_{k+1} also gives

Choose the initial vector as

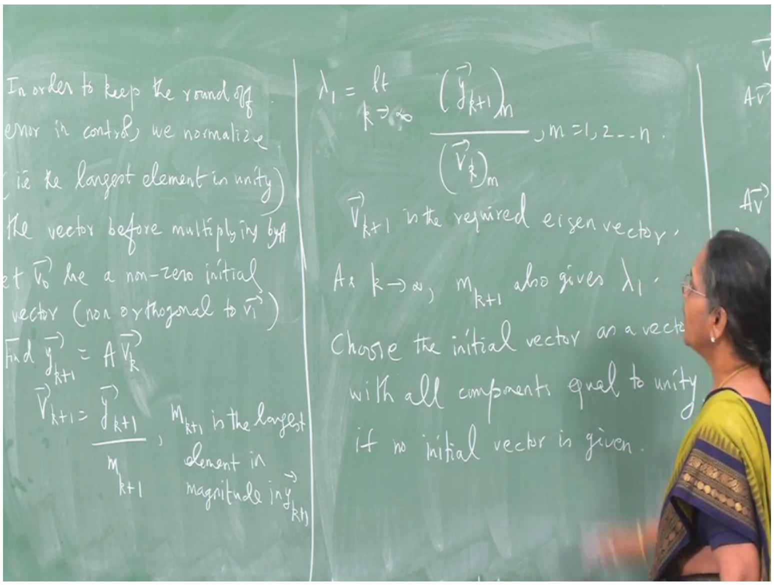
with all components equal

if no initial vector is give

So let us take an example and understand power method which helps us to determine numerically the largest eigenvalue in magnitude of a given matrix A. So let us try to illustrate power method which helps us to compute numerically largest in magnitude eigenvalue of a given matrix A. So we need to control the round of errors in our computation. So we take care of this by doing the following. Namely we normalise the vector before multiplying by A. You just look back and see what we did, we, at each steps we premultiplied the vector V by A, here we multiplied AV by A and so on.

In order that we have control of round off errors, before multiplying by A, we normalise the vector V, that is we see to it that the largest element in that vector is unity. So let V0 be a nonzero initial vector, we have to start with an initial vector V0 and generate a sequence of iterates for V and which tends to the Eigen vector associated with a given eigenvalue. So we do this as follows. So we take AVK and call it YK +1, we start with the vector V, so AV is obtained and that is what is mentioned here. Find at each what YK +1 is, so YK +1 will be AVK.

(Refer Slide Time: 44:15)



In order to keep the round off
error in control, we normalize
(i.e the largest element is unity)
the vector before multiplying by A
et $\vec{V_0}$ be a non-zero initial
vector (non orthogonal to $\vec{v_1}$)
Find $\vec{y}_{k+1} = A\vec{V}_k$
$\vec{V}_{k+1} = \dfrac{\vec{y}_{k+1}}{m_{k+1}}$, m_{k+1} is the largest element in magnitude in $\vec{y}_{k+1}$

$\lambda_1 = \lim_{k\to\infty} \dfrac{(\vec{y}_{k+1})_m}{(\vec{V}_k)_m}$, $m = 1, 2 \ldots n$.

$\vec{V}_{k+1}$ is the required eigen vector.

As $k \to \infty$, m_{k+1} also gives λ_1.

Choose the initial vector as a vect
with all components equal to unity
if no initial vector is given.

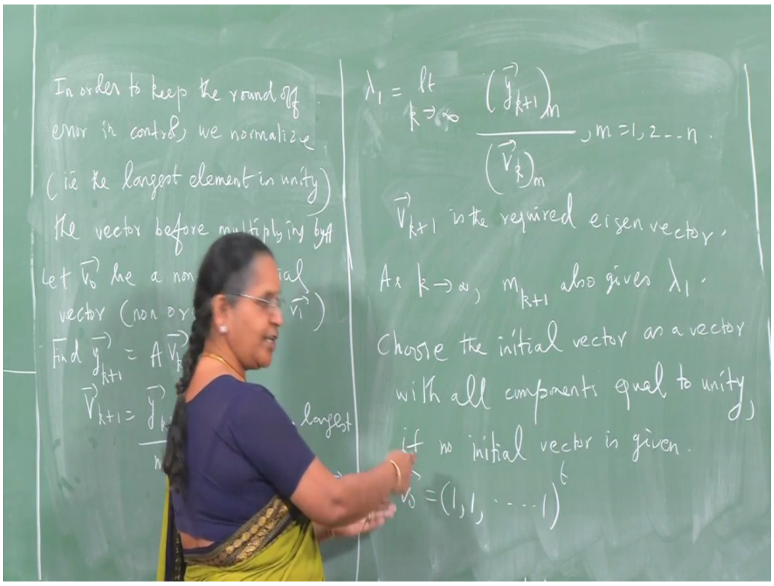
$A\vec{v}$

$A\vec{v}$

And now what is VK +1, V K +1 is YK +1 by MK +1, what is MK +1, it is the largest element in magnitude in the vector YK +1. So compute A VK, call it as YK +1, remove the largest element in magnitude in that vector and divide YK +1 vector by MK +1 and call that is VK +1. So if you premultiply now by A, you have a vector A VK +1, that will be of YK +2, from that vector you remove numerically the largest element, call it MK +2. So divide your vector YK +2 by MK +2 and the resulting vector is VK +3. So you continue to perform your computations this way, this will help you to control the round off errors.

So finally what is lambda 1, lambda 1 will be the ratio of the components the mth component of vector YK +1 to the Mth companies of vector VK as K tends to infinity. And what is the Eigen vector, Eigen vector will be VK +1 which is associated with the eigenvalue lambda was that you computed. In fact you can see that as K tends to infinity, MK +1 gives you the eigenvalue lambda 1. So choose your initial vector V0 with all its components to be equal to unity, provided you are not given any initial vector to start with in your computations.

(Refer Slide Time: 44:58)



In order to keep the round off error in control, we normalize (i.e the largest element is unity) the vector before multiplying by A

Let $\vec{V_0}$ be a non ... ial vector (non zer ... $\vec{V_1}$)

Find $\vec{y}_{k+1} = A\vec{V}_k$

$\vec{V}_{k+1} = \frac{\vec{y}_{k}}{m}$ ... largest

$$\lambda_1 = \underset{k\to\infty}{lt} \frac{(\vec{y}_{k+1})_m}{(\vec{V}_k)_m}, m = 1, 2 .. n.$$

$\vec{V}_{k+1}$ is the required eigen vector.

As $k\to\infty$, m_{k+1} also gives λ_1.

Choose the initial vector as a vector with all components equal to unity, if no initial vector is given.

$\vec{V}_0 = (1, 1, \cdots 1)^t$

So start with an initial vector V0 which has components 1, 1, 1 etc. and then compute A V0, A square V0 and so on and take the limit as N tends to infinity to obtain the ratio of the Mth component of these vectors in your computations and that will give you the eigenvalue which is the most prominent eigenvalue of the given matrix A. And the corresponding Eigen vector is given by vector VK +1 as K tends to infinity and you have the required result at the end of your computation, namely the largest eigenvalue in magnitude and the corresponding Eigen vector. So we shall illustrate this method in the next class by taking some examples.