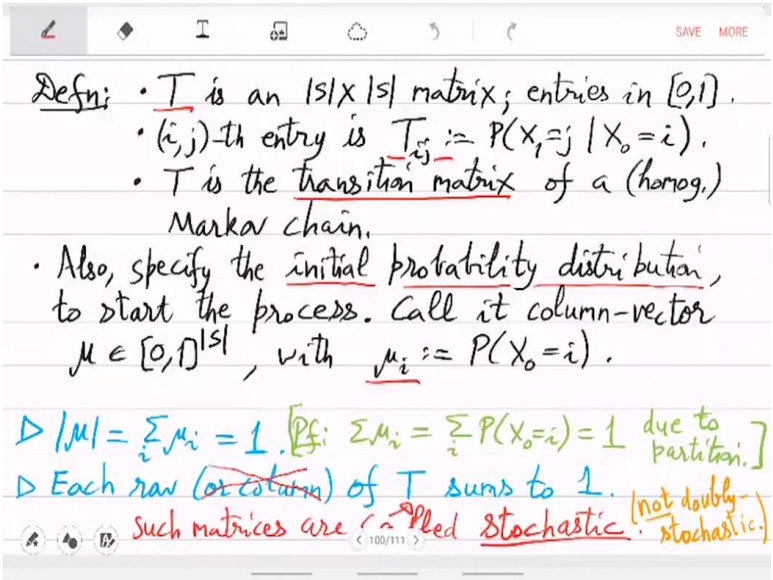
**Probability for Computer Science**
**Prof. Nitin Saxena**
**Department of Computer Science and Engineering**
**Indian Institute of Technology - Kanpur**

**Module - 6**
**Lecture - 21**
**Perron-Frobenius Theorem, Page Rank Algorithm**

Last week we started stochastic process and we defined Markov chain as a sequence of random variables where the current variable depends only on the previous one and is independent of everything that came prior to that. And then we called it Markov chain. And we are especially interested in this course in homogeneous Markov chains, which basically means that it does not depend on time; the probability does not depend on time.

So, X K over X K - 1, probability is the same as it was at any given point of time; so, in particular, X 1 over X 0. We saw some examples. We saw a way to represent Markov chain, homogenous Markov chain. And then we came to transition matrix.

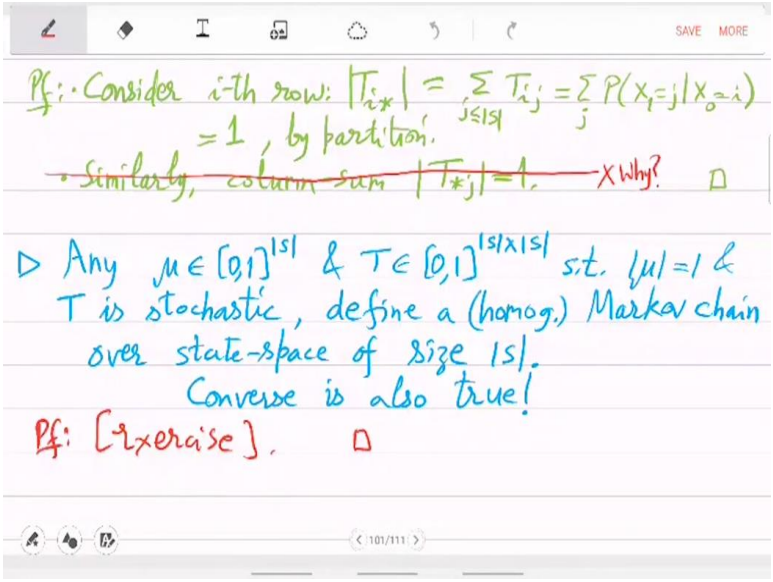
**(Refer Slide Time: 01:05)**




So, here, I have to repeat some things. So, Markov chain is equivalent to; whenever I say Markov chain, I mean homogeneity implicitly. So, it is equivalent to defining or giving a transition matrix; T, right? So, T ij is the probability that j appears given i appeared before. And initial probability distribution; so, what was the probability of being in a state out of n states or out of these s states, in the very beginning.

So, I said that the initial probability distribution mu; sum of mu i is 1, which is correct. Then the rows, each row sum of T is 1; that is also correct; but column sum may not be 1. We do not know anything about column sum. So, this matrix T is called stochastic just by the row sum being 1. If column sum was also 1, then we call it doubly-stochastic, but this may not be doubly-stochastic. So, that was a mistake. And this is important to note.

**(Refer Slide Time: 02:14)**




So, for the row sum, we said that you just look at Sigma T ij. And that is, where did you go from i? So, you have to go to one of the j's. So, that probability sum is 1 by partition; but you cannot do the same thing for column sum, because, there you will be asking how did you come to j? But this Sigma T ij as you vary i, you cannot say anything about this probability. That was the mistake. So, this column sum may not be 1.

It may be less; it may be more than 1. Then we saw this example of drunkards walk in 1 dimension. There are many such examples where any physical process which seems memoryless is actually Markov chain. Then we went to evolution of a Markov chain. So, that happens basically by right multiplication of matrix, transition matrix. And so, the Markov chain keeps evolving till n equal to infinity, and then what happens?

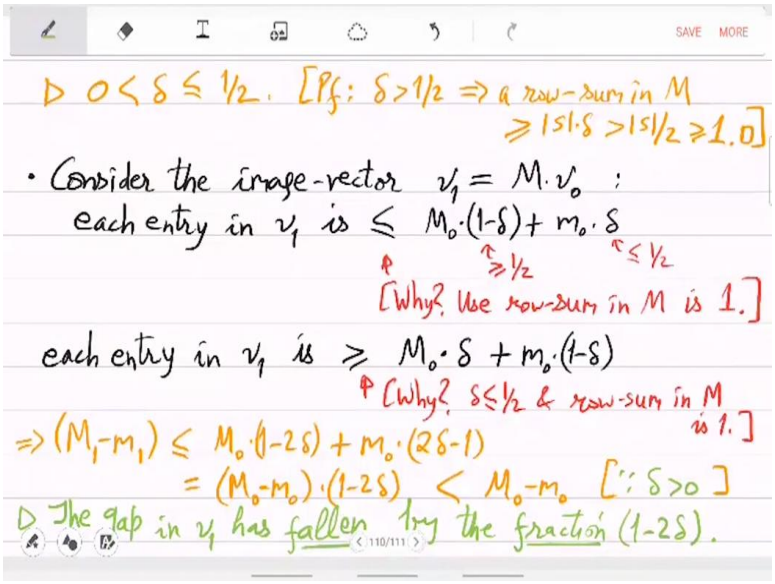
So, this P n, probability distribution on the states as n tends to infinity, what is this? So, this we started studying as stationary distribution and for that we needed regularity. So, at some point of time; so, M to the t should be all positive entries, strictly positive. When that happens, we started this Perron-Frobenius theorem, which says that if M is the transition

matrix of a regular homogeneous Markov chain, then M to the limit exists, limit of M to the n exists. In fact, this is just a rank 1 matrix; column vector 1 times a row vector w.

This is what we will show now. This also tells you that; again it reminds you of memorylessness, because this w in the end, the stationary distribution we call it, is independent of what you started from, what mu you started from. It only depends on transition probabilities, not on initial distribution. So, we said that we will work with matrix action m. So, M acting on a vector v 0, let us say, gives you v 1.

We will show that the entries of v 1 are getting closer with each matrix action. So, ultimately what will happen is, M to the n times v 0 will become a scalar. That is the viewpoint we take. So, let us say in v 0, the minimum is small m 0 and the maximum entry is big M 0, and similarly for v 1. And let us also assume that matrix M has been entry delta, which we can assume to be positive, because of regularity. If any entry is 0, then you basically go to M to the t, work with that.

**(Refer Slide Time: 05:43)**



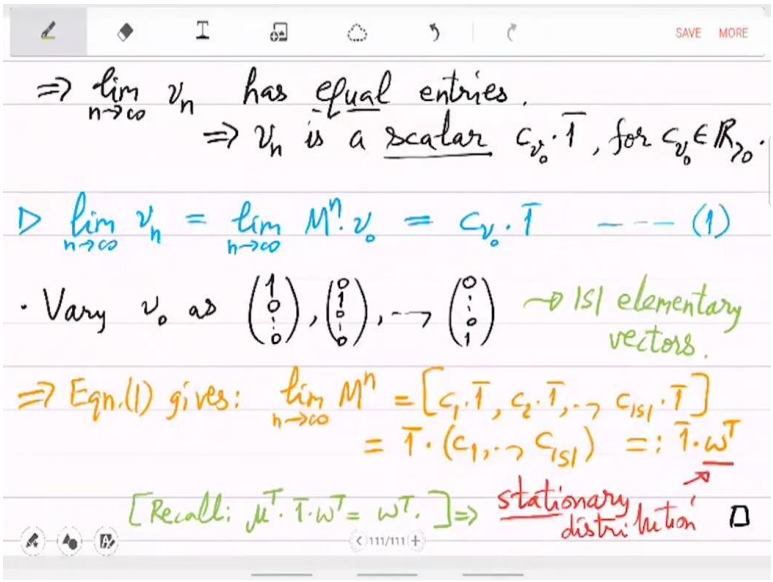

Now, delta cannot be more than half, because if the minimum entry is more than half, then the row sum will exceed 1; that cannot happen. So, now let us look at the action M times v 0 given v 1. So, you can very easily estimate that each entry in v 1 upper bound is, if you give this big M 0, the biggest, the largest entry in v 0, if you give it more weight, which is 1 minus delta, this is at least half, so, this is the larger weight.

And the smaller m 0 part, you give smaller weight delta. So, that is an upper bound. You can show that this actually works using the row sum property of M. And symmetrically, the lower bound on entries in v 1 is, if you do the opposite. So, you give more weight, 1 minus delta to smaller entries like m 0, smaller m 0; and less weight delta to the larger entries like big M 0. So, that is the thing we now continue with. So, what does this mean?

This means that the difference of maximum minimum entries in the vector v 1 is less than equal to; so, maximum is smaller than the first bound and minimum is at least the second bound. So, you take the difference. And what do you get? So, big M 0 times 1 minus 2 delta plus small m 0 times 2 delta minus 1, which is M 0 minus small m 0 times 1 minus 2 delta, which is less than big M 0 minus small m 0. Since, delta is positive.

If delta was 0, then there would have been no change in the deviation; but since delta is positive, the deviation is actually falling. So, in v 1, elements have become a bit closer, and you also know how close. So, it is at least by this fraction 1 minus 2 delta. So, let us remember this. So, the gap falls, has fallen by the fraction 1 minus 2 delta. So, since this is a constant fraction, you repeat this many times. And in fact, in the limit, the deviation becomes 0. That is the conclusion.

**(Refer Slide Time: 09:28)**




So, this implies that limit as n tends to infinity of v n has equal entries, which means v n is a scalar. So, scalar depending on this thing that you started with, right? v 0 times 1 bar for c v 0, a positive constant. So, we have shown that, if you keep doing this action of matrix

multiplication on the left by M, then ultimately, the column vector that you will get will be just all-1 vector multiplied by some constant.

So, that is quite nice, because this tells you that limit as n tends to infinity of v n is limit M to the n times v 0 = c 0 times 1 bar. Let us call this equation 1. So, the limit exists and it is actually very nice; it is essentially the all-1 vector. So, this was true for any vector v 0. So, you can as well vary the, you can look at different actions. So, you can take v 0 to be an elementary vector.

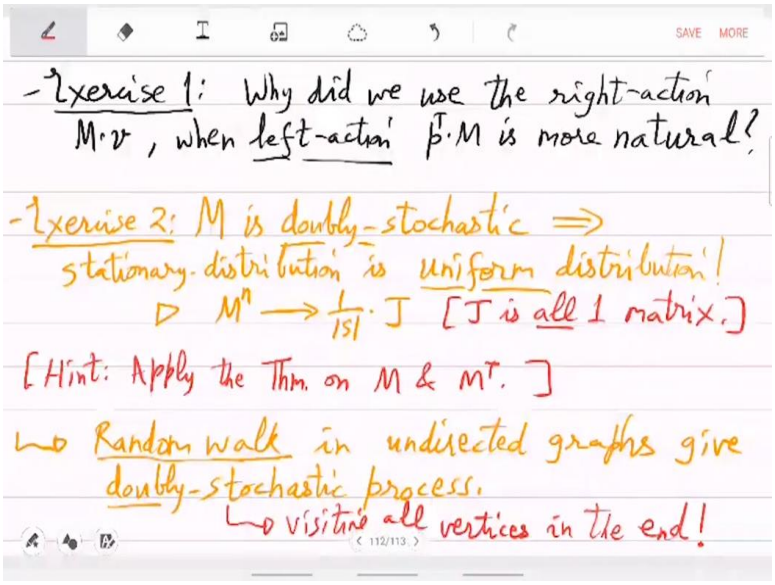
So, vary v 0 as a vector with 1 only in the first position, 1 in the second position and 1 in the last position. So, these are your s elementary vectors. Then what do you get? So, then, equation 1 gives actually an information about limit of M to the n. So, instead of the action of M to the n, you can actually now talk about M to the n matrix itself. So, what is this matrix? This is essentially M to the n times the first choice of v 0, that gives you the first column.

So, the first column is some constant times 1 bar; the second is c 2 times 1 bar; and finally, c s times 1 bar. So, essentially, the rows are equal; that is what has happened. So, this is, in other words, multiplication of column 1 bar with row these constants, these scalars, right? This you can see just by matrix multiplication definition. And that is what we can call w. This is the definition of w.

So, you have complete information about the matrix power, as the power keeps on increasing. So, this is Perron-Frobenius theorem, one version. You get the stationary distribution. And why is this stationary? Because; did I define stationary before? Yes; because this action on the left gives you exactly this. So, recall that, if you now look at initial distribution, multiplying on the left then you get the value 1 times w transpose.

So, this w is the stationary distribution. These c 1 to c s, there is a very clear meaning. c i is essentially the probability of being in that state, ith state, after infinite applications, infinite steps in this Markov process. So, that is what we have shown.

**(Refer Slide Time: 15:12)**



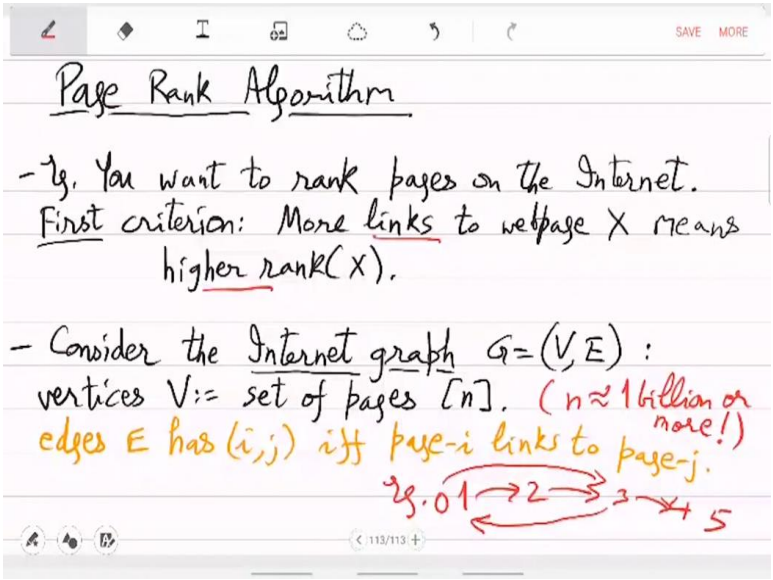

So, couple of exercises here, because it is a tricky proof, probably something very new for you. So, first exercise is, why did we use the right action M dot v, when the left action is more natural? So, Markov chain evolution works with left action, multiplying on the left by the previous probability distribution on the states. So, that is more natural. So, why did we actually use the right action in this proof?

Why cannot we just work with the left action and get the same statement? So, think about this. Second is, so, if M is doubly-stochastic, what happens? So, in the previous proof, M was only stochastic, which means that the row sums, every row sum was 1. What happens if you also know that the column sums are 1? Then you can actually show that stationary distribution is uniform.

So, essentially, you can show that M to the n as n tends to infinity, this is 1 over s times J, where J is the all-1 matrix. So, for doubly-stochastic Markov chains or transition matrix, if you keep following the process, then, in the end, actually all the probabilities become equal. So, this is a simple proof, follows the; I mean you can just invoke this Perron-Frobenius theorem twice; one on M as we did and the other on M transpose.

So, I will leave this as an exercise. So, apply the theorem on M and M transpose; that will do it. So, this is just scratching the surface of Markov chains. Let us now see some examples of this. And there are many good practical examples, physical phenomena which is modelled by Markov chains. So, let me give you some stunning examples, depending on how much time we have.

**(Refer Slide Time: 18:57)**




So, first is the Page Rank algorithm. And let me remark here that random walks in undirected graphs give doubly-stochastic. So, if you are doing a random walk in an undirected graph, then you actually get symmetric matrix M, which is a doubly-stochastic process. And so, basically, if you walk in the graph for a long time, then, if the graph is connected, then you will get to the stationary distribution which is uniform.

So, you will be able to visit every vertex; so, visiting all vertices in the end. This is what, this is one easy way to interpret what we just did, that a random walk in a graph will ultimately result in visiting all the vertices with equal chance. That is an amazing process itself. So, now, let us build on that and use it to search webpages in the internet. So, say you want to rank pages on the internet.

So, first criteria that you can use is, the more a page is linked by others, the more important it is. So, more links to a webpage X means higher rank. So, this is link based criteria. So, more pages link out to this webpage X, then it somehow is indicative of the importance of X. But this is only the first attempt, this may not be a very good criterion, because maybe people just, maybe a group forms in the internet that is linking this maliciously and trying to increase the rank.

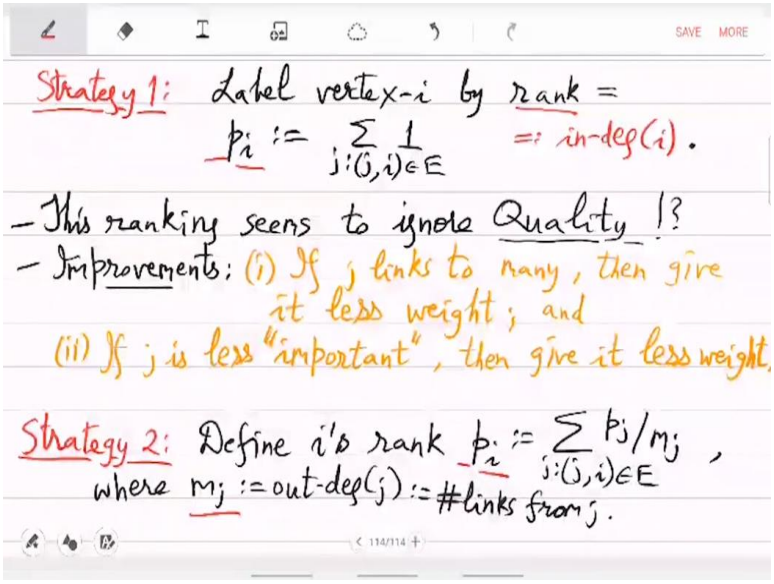
So, you have to make this criteria more refined. But anyways, you can; let us start with this, with something. So, consider the internet graph which has vertices V, edges E. What are the edges? So, well, first of all, vertices V are; this is just the set of webpages; and let us call this,

without loss of generality, label them by 1 to n. And you should think of n as very large, many billion. So, it is something like 10 to the 9 or 10 to the 10, 1 billion, 10 billion.

This obviously keeps on increasing with time. So, there is a huge number of pages in the internet. And what are the edges? Edges, E has i comma j if and only if page-i links to page-j. That is all. That is the internet graph. So, you have, 1 may be linking to 2; 2 linking to 3; 3 linking to 4; and then, maybe 3 linking to 1. And there might be a vertex 5 which may not have any links. Correct?

And similarly, maybe there is 0 who is not linked by anybody, but 0 links 3; but nobody links 0, right? So, in this, somehow, so, page 3 has 2 links to it; so, page 3 seems important. 0 and 5 seem not important; nobody is interested in those webpages. 1 has only 1 link; 4 has only 1 link; 2 has only 1 link. So, by this ranking method, you would give 3 highest rank and 0, 5 lowest and everything in the middle. That could be the first criterion. But how will you compute this rank? What is the algorithm for it? So, this one is obviously simple; it follows from the definition.

**(Refer Slide Time: 26:03)**




You just have to compute the n degree of a vertex. So, label vertex-i by rank of, rank p i which is just counter 1 for every j such that j to i is an edge. So, vertex-i's label is just the in-degree. So, that is simple, but the issue is this ranking business seems to ignore quality. By quality, we mean that, intuitively, if a vertex that is more important is linking to something, that should be given more weight.

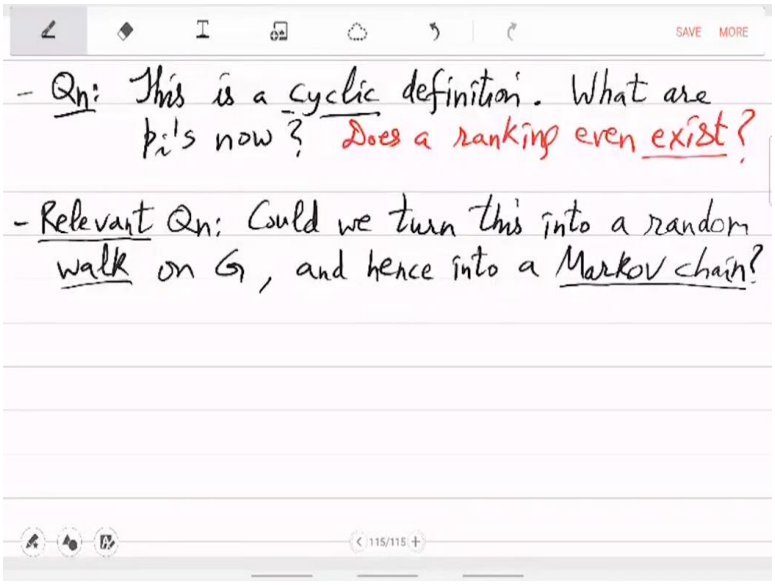
And a vertex that has a very little importance; so, in this case, in the previous example, 0 linking to 3 should have less weight than 1 linking to or 2 linking to 3, because 0 has, nobody is interested in 0. So, why should you count its link to 3? While people are interested in page 2 and page 2 is linking to 3, so, that is kind of more important. And this is a small example, but in a big example, this can be very important, this additional feature has to be added to get closer to the correct ranking.

So, who likes you is important, the quality of that webpage is important. Who links to you? It cannot be ignored. So, let us do an improvement. So, if j links to many, then give it less weight. If j is less important, then give it less weight. So, both the things seem reasonable. If j links too much, then its opinion should be discounted that much; by that much, right? And also if j's rank is very less, then its link should be given less weightage.

So, what does this mean mathematically? Let us see that. So, define i's rank p i to be p j divided by m j and j such that j links to i, where; so, now it is not simply in-degree, it is not actually property of the graph; this is something else; this is property of importance. I mean, here we are trying to capture j, j is important. So, if j's rank is high, which is p j, then this contribution is higher, the linking to i gets more weight.

And if m j is the number of links, so, m j is out-degree of j, which is number of links from j. So, we are defining 2 things. So, p i is sum of the links to it, but links are being weighted by how important j is or, I mean, what is the rank of j itself; and it will be reduced or normalised by the out-degree of j; whether j has a habit of linking too much; then we discount by this. But, now it has become cyclic; it is a cyclic definition, right? p i depends on p j.

**(Refer Slide Time: 31:56)**




So, this is a cyclic definition. So, what are p i's now? Do they even exist? Does the ranking even exist? That is the question. So, it is possible that, maybe these equations are not enough to get the ranking. So, the question that will actually solve our problem is to turn this into random walk. So, could we turn this into a random walk on G, and hence a Markov chain? So, we are actually trying to get this rehashed into a Markov chain and then solve it.