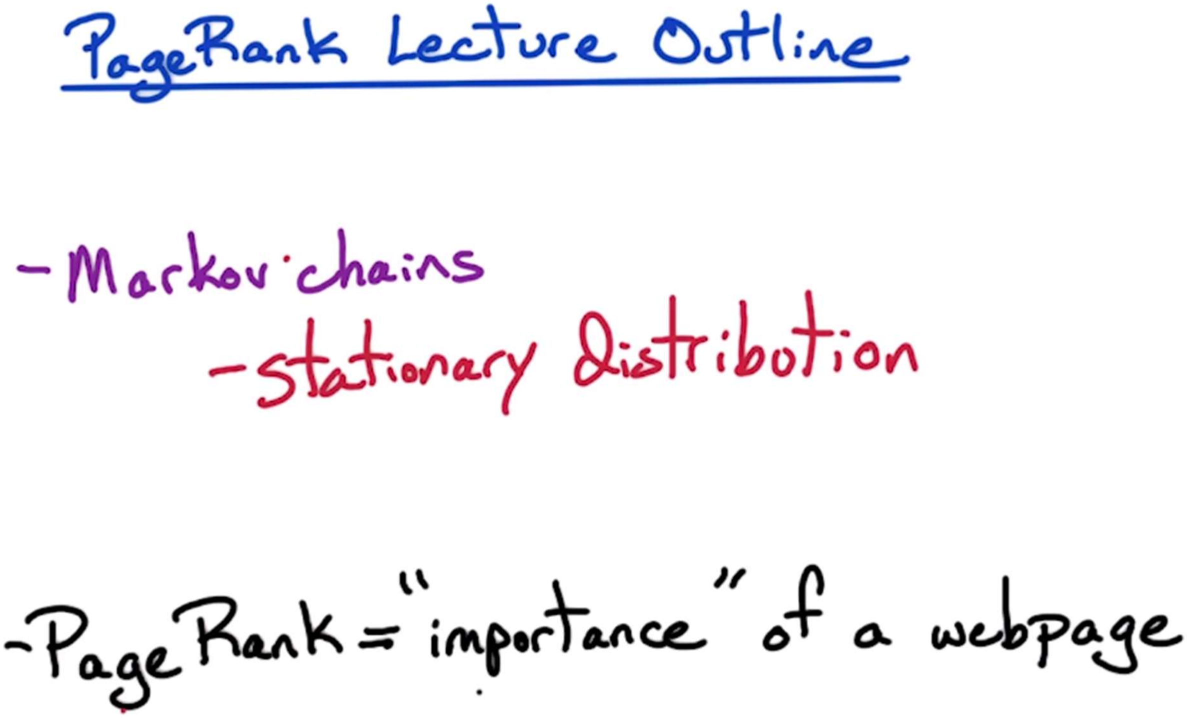
LESSON: GR4: Markov Chains and PageRank (Optional)

***

(Slide 1) PageRank Lecture Outline



PageRank Lecture Outline

- Markov chains
  - stationary Distribution

- PageRank = "importance" of a webpage

In this lecture, we're going to look at the PageRank algorithm. This is an algorithm for assigning the importance of a webpage.

Now, what exactly do we mean by importance? Well, this is a subjective term, but we're going to give a precise quantitative interpretation of importance.

Now, the PageRank algorithm is the algorithm devised by Brin and Page which is at the heart of Google search engine. Now, the PageRank algorithm itself is fairly simple, but to fully understand the algorithm, we have to first understand some basic mathematical tools known as Markov chains.

Now, I'm going to give you a quick primer on Markov chains.

- What exactly are Markov chains? And what are the key properties of Markov chain?

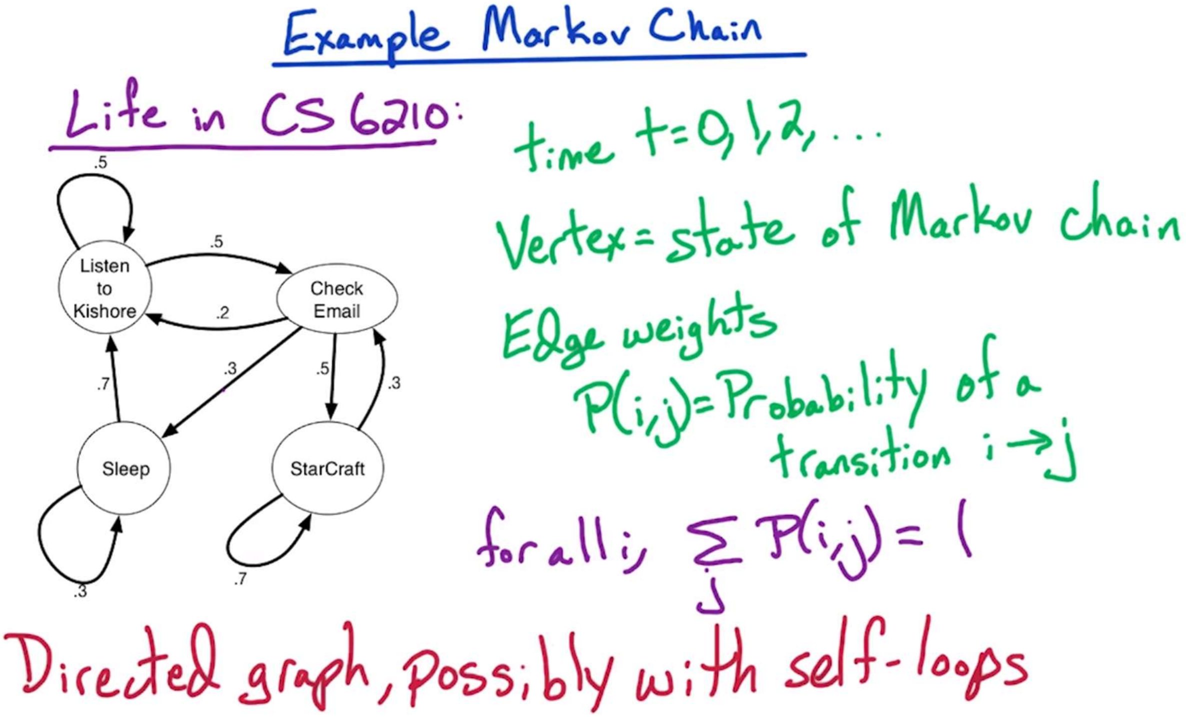
  Now, the PageRank algorithm is itself a Markov chain but Markov chains come up quite often. For example, you may have heard of the term MCMC, That stands for Markov chain Monte Carlo. Also, you may have heard of simulated annealing, which is another example of a Markov chain. Now, we're not going to look at these two examples in detail, but if you understand Markov chain, the basics of Markov chains, then that'll help you understand these more sophisticated concepts if you decide to delve into them in more detail.

- Now, the key concept for Markov chain is the notion of a stationary distribution. So, I'm going to spend some time explaining to you what exactly does a stationary distribution mean. How do we determine what the stationary distribution are and what are the important properties of a Markov chain which connect to the notion of a stationary distribution?

- Finally, once we understand Markov chains and stationary distributions for Markov chains, then we will be able to fully appreciate the PageRank algorithm and the design choices in the design of the PageRank algorithm. Now, let's go ahead and dive into a Markov chain and look at a specific example of a Markov chain.

***

(Slide 2) Example Markov Chain



Example Markov Chain

Life in CS 6210:

time $t=0,1,2,\ldots$

Vertex = state of Markov chain

Edge weights

$P(i,j)$ = Probability of a transition $i \to j$

for all i, $\sum_j P(i,j) = 1$

Directed graph, possibly with self-loops

Now, this directed graph is an example of a Markov Chain. Now, this example is meant to illustrate your state of being at various times while sitting in CS 6210. It certainly doesn't illustrate 6505 since we have this sleep here.

Now we can think of discretizing time (having a discrete value) so that the time is a parameter t which goes from zero, one, two, … It has integer values. You can think of the time as being like the time in seconds or the time in minutes.

Now for this particular example, there are four possible states at each time. (1) You can be listening to Kishore (the professor for 6210), (2) you can be sleeping, (3) you can be checking your email or (4) you can be playing this video game StarCraft.

So each vertex in this directed graph corresponds to a state of the Markov Chain. So there's a Markov Chain has four possible states. Now the edges of this directed graph have weights. The weights correspond to the probability of a transition. So the weight of the edge is between say, checking email and StarCraft, is the probability of changing from checking email at time t to playing Starcraft at time t+1.

So let's say I'm checking email at time 0. Then at time 1, with probability .3, I'll be sleeping; with probability .5, I'll be playing Starcraft; and with probability .2, I'll be listening to Kishore.

Similarly, if I'm listening to Kishore at some time t, then at time t+1 with probability .5, I'll be listening to Kishore again and with probability .5, I'll be checking email.

So in general, a Markov Chain is defined by a directed graph and one key thing is that this directed graph might have self-loops. For instance, if I'm listening to Kishore at time t, then with probability .5, I'm listening to Kishore at time t+1.

Now notice that the out edges out of each vertex define the probability distribution for the next state. So, if I'm checking email at time t, then I'm listening to Kishore with probability .2, sleeping with probability .3, and playing Starcraft with probability .5 in the next time-step.

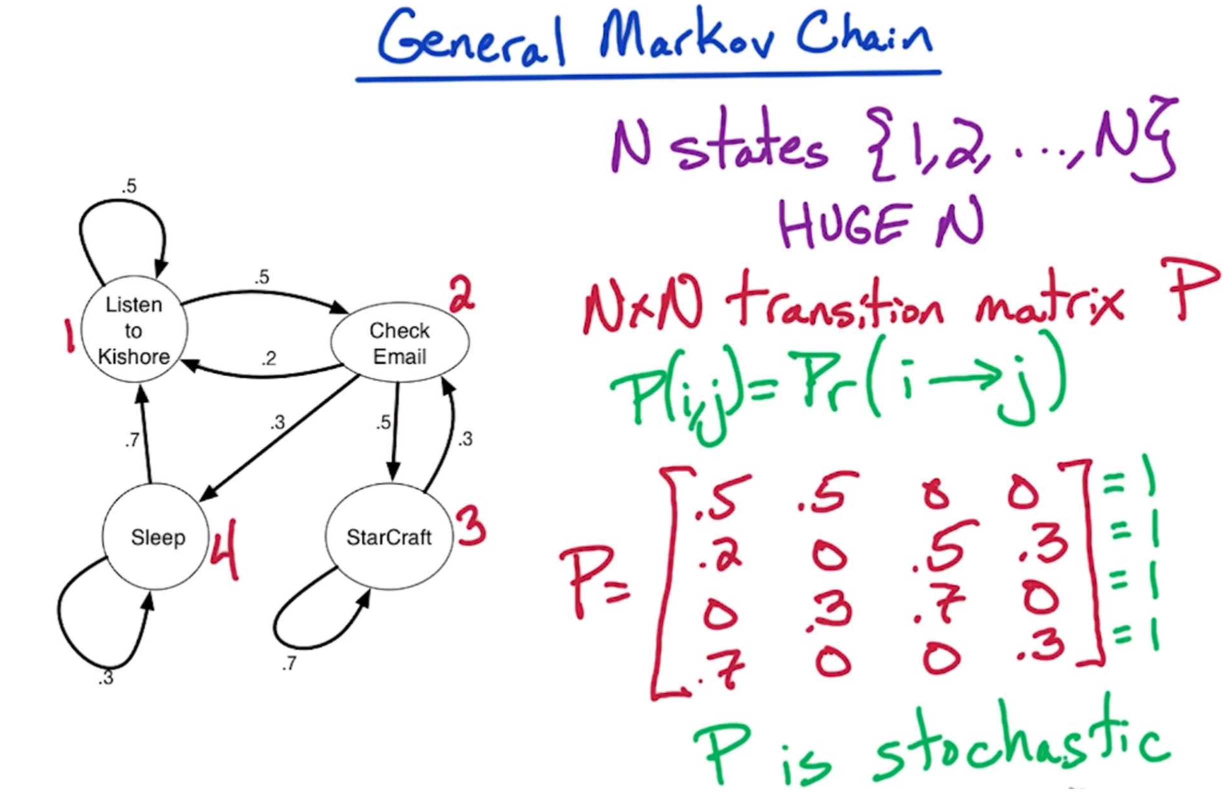
Notice that these edge weights .2 plus .3 plus .5 have to sum up to 1 because this is a probability distribution for the next state.

So for every state I - in this case, i equals check e-mail - if I sum over the out edges - so I sum over the weights of these out edges - this is P(i,j). Then, what do these some to? They have to sum up to 1 because this is a probability distribution for the next state given I'm in state i at time t then I'm going to be in state j at time t+1.

Also, what are valid edge weights? Well, these correspond to probabilities. Probabilities have to be between 0 and 1. So, all the edge weights are between 0 and 1. You give me any directed graph with edge weights between 0 and 1 - that defines a Markov Chain. And similarly, any Markov Chain can be viewed as a directed graph with these edge weights.

***

(Slide 3) General Markov Chain



General Markov Chain

N states $\{1,2,\dots,N\}$

HUGE N

$N \times N$ transition matrix P

$P(i,j) = Pr(i \rightarrow j)$

$$P = \left[\begin{array}{cccc} .5 & .5 & 0 & 0 \\ .2 & 0 & .5 & .3 \\ 0 & .3 & .7 & 0 \\ .7 & 0 & 0 & .3 \end{array}\right] \begin{array}{l} =1 \\ =1 \\ =1 \\ =1 \end{array}$$

P is stochastic

So how do I define a Markov chain in general?

In general, I'm going to have capital N states, and we are going to label these states by 1 through N.

So in this example, capital N equals four. But, in general, think of in our applications, N is going to be huge. For instance, when we do PageRank, N is going to be the number of webpages on the internet - the TOTAL number of webpages on the internet.

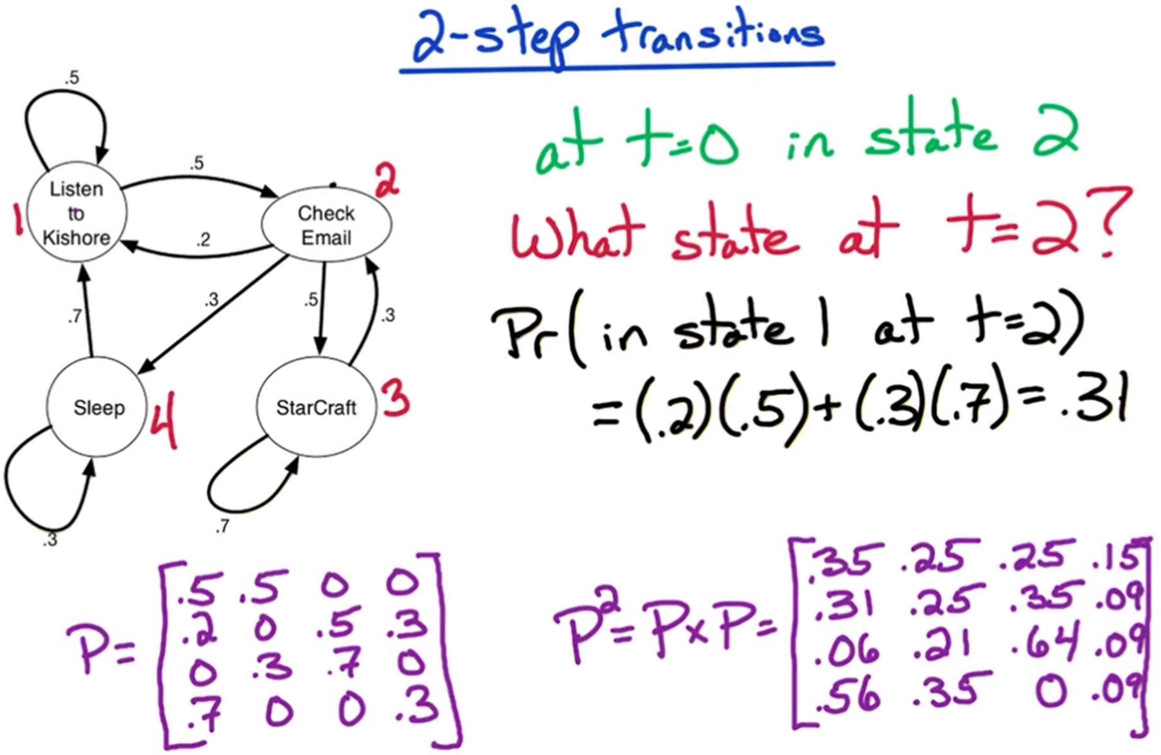
Now, this weighted directed graph is defined by its adjacency matrix. This is the adjacency matrix for this graph where we think of this vertex as this state - Listening to Kishor is state 1, this is state 2, playing Starcraft is state 3 and sleeping is state 4. So, this is the adjacency matrix for this graph - the weighted adjacency matrix.

Now, I denote this by P, and we refer to this as a transition matrix P. Why? - because the entry P(i,j) corresponds to the probability of transitioning from state i to state j. So, if I'm at state i at time t, the probability I'm in state j at time t+1 is exactly the entry P(i,j) - I look at row i and column j.   For instance, if I'm playing Starcraft at time t, the probability that I'm checking email at time t+1 is exactly 0.3.

Now, the property that you noticed before about P is that each row sums up to 1. The terminology for this is that P is a stochastic matrix. If the columns also sum to 1, then it's called doubly stochastic. But, for a Markov Chain, all we know is that the rows sum up to 1. It doesn't necessarily mean that the columns have to sum up to 1.

***

(Slide 4) 2-Step Transitions



2-step transitions

at t=0 in state 2

What state at t=2?

$$\Pr(\text{in state } 1 \text{ at } t=2) = (.2)(.5)+(.3)(.7)=.31$$

$$P=\begin{bmatrix} .5 & .5 & 0 & 0 \\ .2 & 0 & .5 & .3 \\ 0 & .3 & .7 & 0 \\ .7 & 0 & 0 & .3 \end{bmatrix} \qquad P^2=P\times P=\begin{bmatrix} .35 & .25 & .25 & .15 \\ .31 & .25 & .35 & .09 \\ .06 & .21 & .64 & .09 \\ .56 & .35 & 0 & .09 \end{bmatrix}$$

Here again, is our earlier example of a Markov chain with four states. This is a weighted graph. But, for a moment, let's ignore the edge weights and let's look at the unweighted version of this graph.

Here's the adjacency matrix, where this is the unweighted version of this graph. From State 1, there are edges to itself to State 2 and that's it. So, the first row looks like 1 1 0 0, and it’s straightforward to check the remainder of the adjacency matrix.

Now, I want to look at the matrix, $A^2 = A \times A$. This is still going to be a 4 x 4 matrix. Let's simply multiply it out, and what do we get?

I claim that this is the matrix A^2. What do these entries mean?

- Well, take a look at this first entry, (1,1). It has value 2, why? Well, because there are 2 paths going from State 1 to State 1 of length two. In particular, I can self-loop and then self-loop, or I can go to State 2 and then back to State 1.
- Take another State, (4,1). From State 4, look at the path of length two to State 1. I can self-loop and then follow this edge or I can follow this edge, and then self-loop.
- But if I look at (4,2), what can I do? Well, there's only one path of length 2. I can go to State 1 and then over to State 2.

- And, if I look at the paths of length two from (4,3) - where there are no paths of length 2 that go from State 4 to State 3 - There's no way to reach 3 from 4 by a path of length 2.

So, this matrix A^2 encodes the number of paths of length 2.

Now let's go back to the weighted version of this graph. This is the transition matrix for this Markov chain. Let's write the transition matrix once again for this Markov chain.

I got half probability of staying at State 1, half probability of going from State 1 to State 2, and then zero probability of going from State 1 to State 3 or State 4. So this is the first row. This is the remainder of the transition matrix.

We're going to look at this matrix, $P^2 = P \times P$ - well, let's simply multiply it out first and see what matrix we get – well, if we multiply out P x P, this is the matrix P^2 that we get.

What do these entries mean? Well, one thing we noticed first, is that we get a 0 entry here, and we also had a 0 entry in the matrix A^2. But why is that? - well, there's no paths of length 2 in this graph from State 4 to State 3.

So this entry (4,3) is 0. These other entries - there are no longer integer values, so they no longer correspond to the number of paths between this pair, (i, j). But they do correspond to the total weight of the paths from i to j.

Suppose we start the Markov Chain at time 0, in State 2 (checking email). And I ask, what is the state at time t=2 - two time steps away? Well, one time step away is defined by the matrix P.

Now let's look more precisely at the probability of going from State 2 at time 0, to State 1 at time 2. So what's this probability of going from State 2 to State 1, in two time steps?

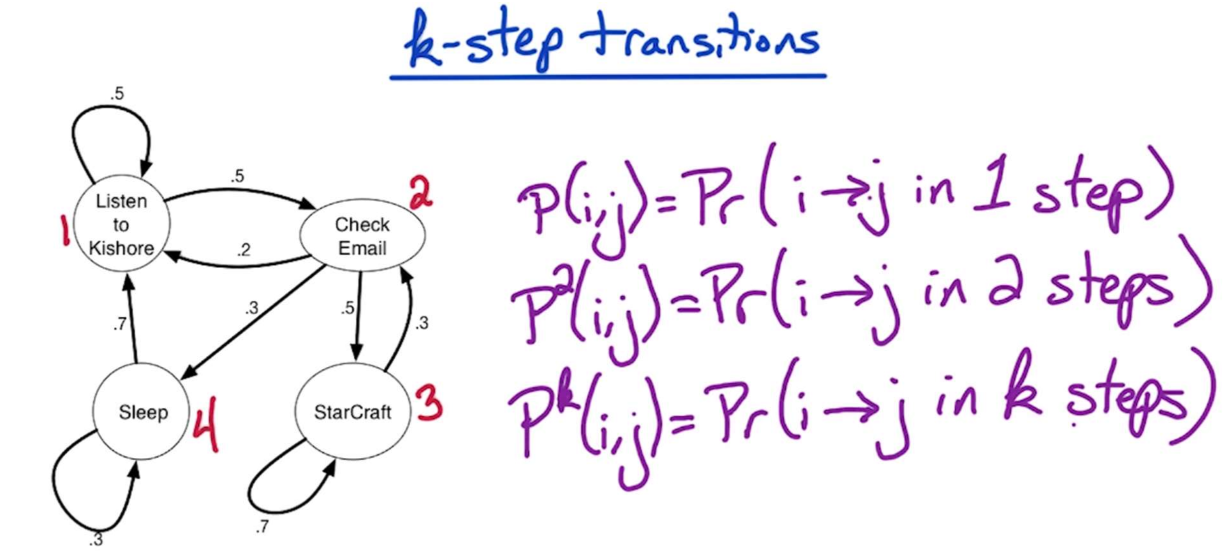
Well, there are two ways to do it. I can go from state 2 -> 1, and then self-loop, or I can go 2->4 and then to state 1. The probability of going from State 2 to State 1 is .2, and then from state 1 to state 1, self-looping, is .5 probability. The other path has probability .3 times .7.

Now if you work this out, what do you get? You get .31, which is exactly the entry (2,1) in this matrix P square.

So, in this matrix P^2 at the entry (i,j) tells us the probability of going from State i to state j in exactly two time steps.

***

(Slide 5) k-Step Transitions



k-step transitions

$$P(i,j) = Pr(i \to j \text{ in } 1 \text{ step})$$

$$P^2(i,j) = Pr(i \to j \text{ in } 2 \text{ steps})$$

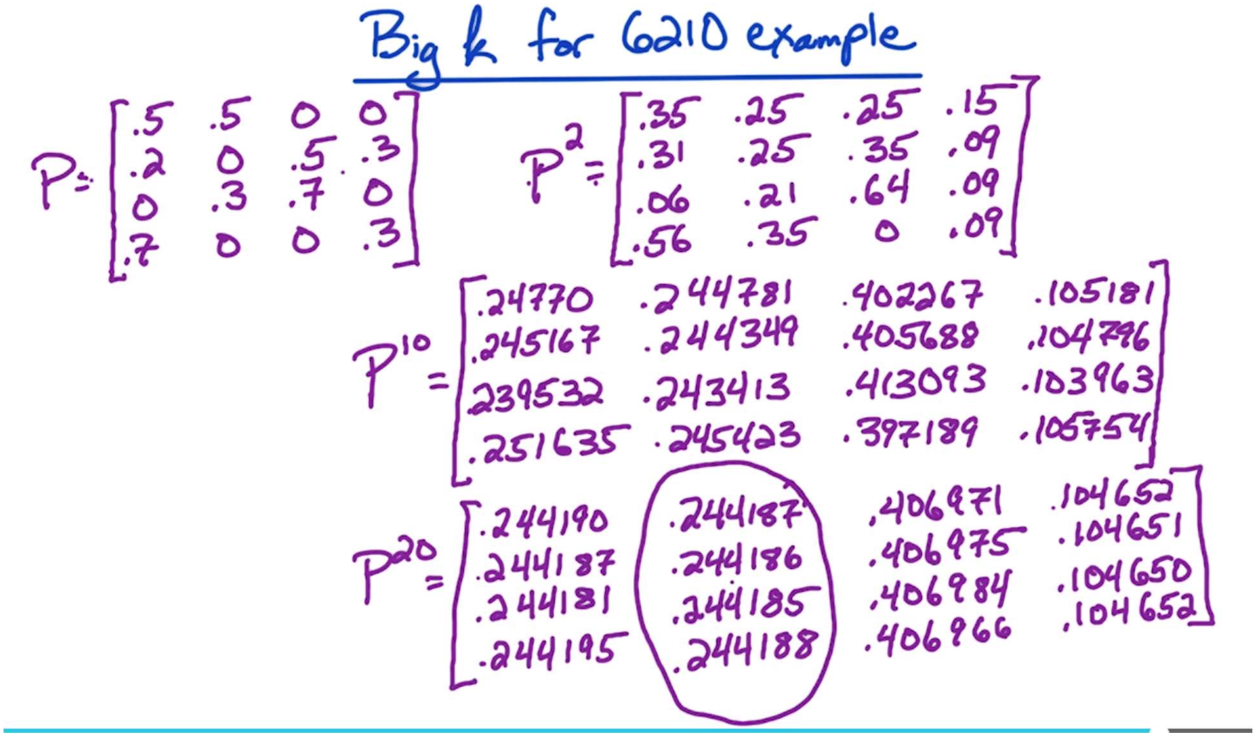
$$P^k(i,j) = Pr(i \to j \text{ in } k \text{ steps})$$

Now if we look at this entry (i,j) , in this matrix P, this transition matrix P, corresponding to the weights in the adjacency matrix. P(i,j) tells me the probability of making the transition from i to j in one step. So if I'm in state i at time t, then P(i,j) is the probability that I'm in State j at time t+1.

Now, what we just saw is that if we look at the square of the transition matrix (P^2), then the entry (i,j) tells me the probability of going from state i to state j in two steps. And, in general, for any non-negative integer k, P^k - the kth power of this transition matrix P - this tells me the probability of making a transition in exactly k steps.

***

(Slide 6) Big-k for 6210 example



Big k for 6210 example

$$P = \begin{bmatrix} .5 & .5 & 0 & 0 \\ .2 & 0 & .5 & .3 \\ 0 & .3 & .7 & 0 \\ .7 & 0 & 0 & .3 \end{bmatrix} \qquad P^2 = \begin{bmatrix} .35 & .25 & .25 & .15 \\ .31 & .25 & .35 & .09 \\ .06 & .21 & .64 & .09 \\ .56 & .35 & 0 & .09 \end{bmatrix}$$

$$P^{10} = \begin{bmatrix} .24770 & .244781 & .402267 & .105181 \\ .245167 & .244349 & .405688 & .104796 \\ 239532 & .243413 & .413093 & .103963 \\ .251635 & .245423 & .397189 & .105754 \end{bmatrix}$$

$$P^{20} = \begin{bmatrix} .244190 & .244187 & .406971 & .104652 \\ .244187 & .244186 & .406975 & .104651 \\ .244181 & .244185 & .406984 & .104650 \\ .244195 & .244188 & .406966 & .104652 \end{bmatrix}$$

Now let's stick with our 6210 example.

Here's the transition matrix again for that 6210 example. And now I want to look at powers of P. S o I want to look at P to the K for big K and then we're going to see some interesting properties about this matrix.

Now, once again, what we saw before the square of this matrix, so P^2 is this matrix.

Now, let's look at it for big K. So, let's look at it from P^{10} and P^{20}.

> Now, actually, if you like to code up, you just code it up yourself. Take this matrix and look at powers of it for big K, or take a different matrix. Make up a matrix and look at powers of it for big K. Just make sure that it's a stochastic matrix so each row sums to one, then it corresponds to a Markov chain.

So this is the first row for P^{10}. And here's the second row. Notice that it's quite similar to the first row. Is that just a fluke? Let's look at more rows to see whether that was just a fluke or if it has some important properties. Now, the exact numbers in this matrix aren't important, but what's important is this interesting property that seems to be coming up: All the roads seem to be converging to the same value. Now, there's still a little bit of variation look in this third column.

Let's see what happens for P^{20}. While looking at the first column of P^{20}, we noticed that it seems to be converging quite nicely. They all agree on the first four significant digits. If we look at the other columns, we see that those are converging quite nicely as well. So what's our conclusion?

Our conclusion is that there seems to be a row vector and all of the rows are converging to this row vector.

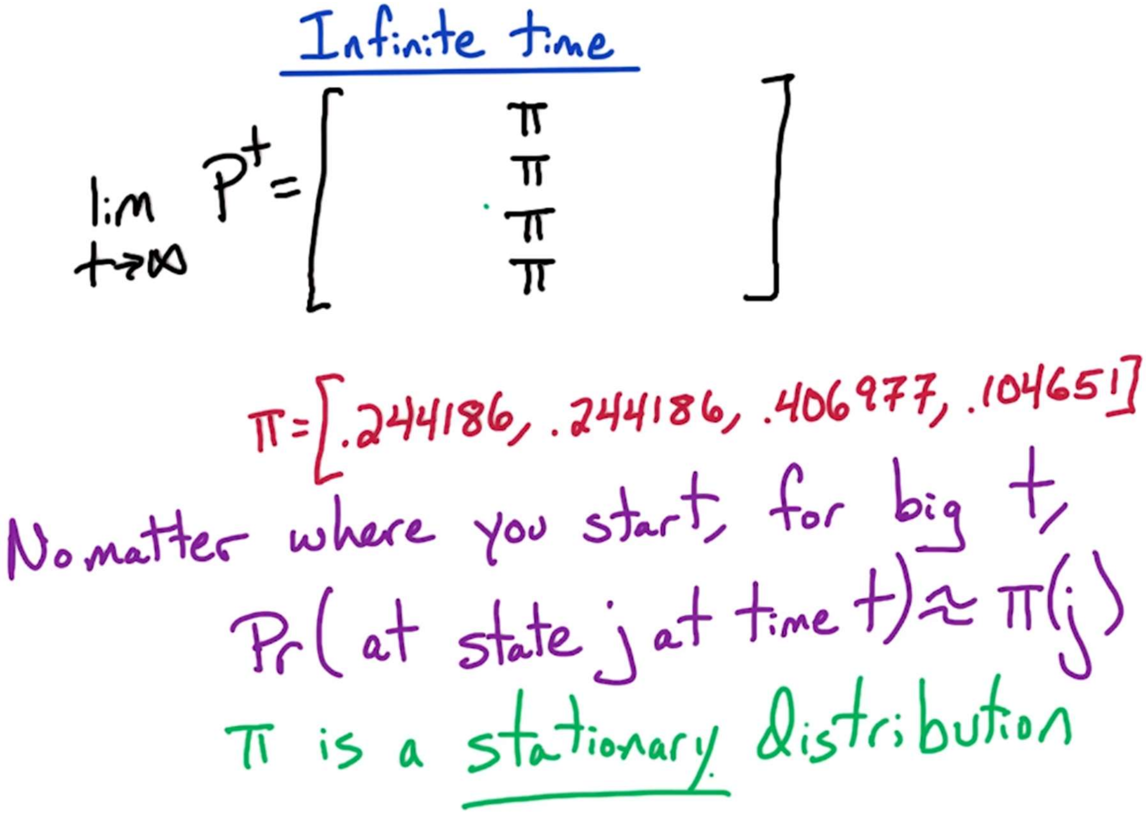
Now, let's look at this matrix see what it means. Take any particular column. Let's take column 2. Now let's look at this entry (1,2) in this matrix P^{20}. What does that mean? - well, $P^{20}(1,2)$ means if I start in stage 1 at time 0, what's the probability that I'm in stage 2 at time 20? Well that's exactly this entry.

Similarly, if I start in stage 2 or 3 or 4, what's the probability that I'm in stage 2 at time 20? Well regardless of where I start in - it seems like it's independent of where I start in. At time 20, it's going to be exactly this. And, if I look at a larger time then this is going to converge even more. So there's going to be a specific probability that I'm in stage 2 for big time regardless of where I start at time 0. So, that's the key property of this Markov chains - regardless of where you start, it doesn't matter. If you look for big time, I'm going to converge to some value.

So where I am at some big large time is independent of where I start. Let's say that again a little bit more precisely.

***

(Slide 7) Infinite Time



Infinite time

$$\lim_{t\to\infty} P^t = \begin{bmatrix} \pi \\ \pi \\ \pi \\ \pi \end{bmatrix}$$

$\pi = [.244186, .244186, .406977, .104651]$

No matter where you start, for big t,

$Pr(\text{at state } j \text{ at time } t) \approx \pi(j)$

π is a stationary distribution

Now, what we saw in the previous slide was we looked at P^t for big t. We did t=20, but let's take t->∞ and see what it looks like.

So let's take the limit P^t as t->∞ and look at P^t. It turns out there's this row vector π. Now these entries might look quite familiar. They look very similar to the row of P^{20}.

Now what is P^t going to look like as t->∞? Well, each row is going to converge to π. So every row is going to converge to this row vector π. What does this mean? - this means that no matter where you start - it doesn't matter where you start because that's the row here - independent of where you start, if you look for big enough t, the probability that you are at state j at time t is going to be exactly defined by this row vector π. So π(j) is going to be the probability that I'm in State j at time t, for big t.

So, for our 6210 example, what does this mean? This means no matter where you start at time 0, if the class is long enough, the probability that you're sleeping at time t is exactly 0.104. And similarly, the probably you playing Starcraft at time t for big t is exactly point 0.406 for big t - regardless of where you start at time 0.

This π is referred to as a **stationary distribution**. You can think of it like a fixed point of the process. For this particular example, regardless of where you start, you eventually reach this stationary distribution and once you're at the stationary distribution, you're going to stay at the stationary distribution. It's going to be invariant.

Now what we want to understand is - does every Markov chain have a stationary distribution? And, does every Markov chain have this property that regardless of where I start I eventually reach this stationary distribution? Moreover, is there a unique stationary distribution or are there multiple stationary distributions?

If you think of the analogy with fixed points are there multiple fixed points? - well, there's only one fixed point and regardless of where I start, the basic attraction is everywhere. So regardless of where I start, I always reach that one attractive fix point.

So, is there one stationary distribution which I reach regardless of where I start? Or can there be multiple stationary distributions? – certainly, there can be multiple, but we want to look at conditions where there are a unique stationary distribution and regardless of where I start, I always reach that stationary distribution.

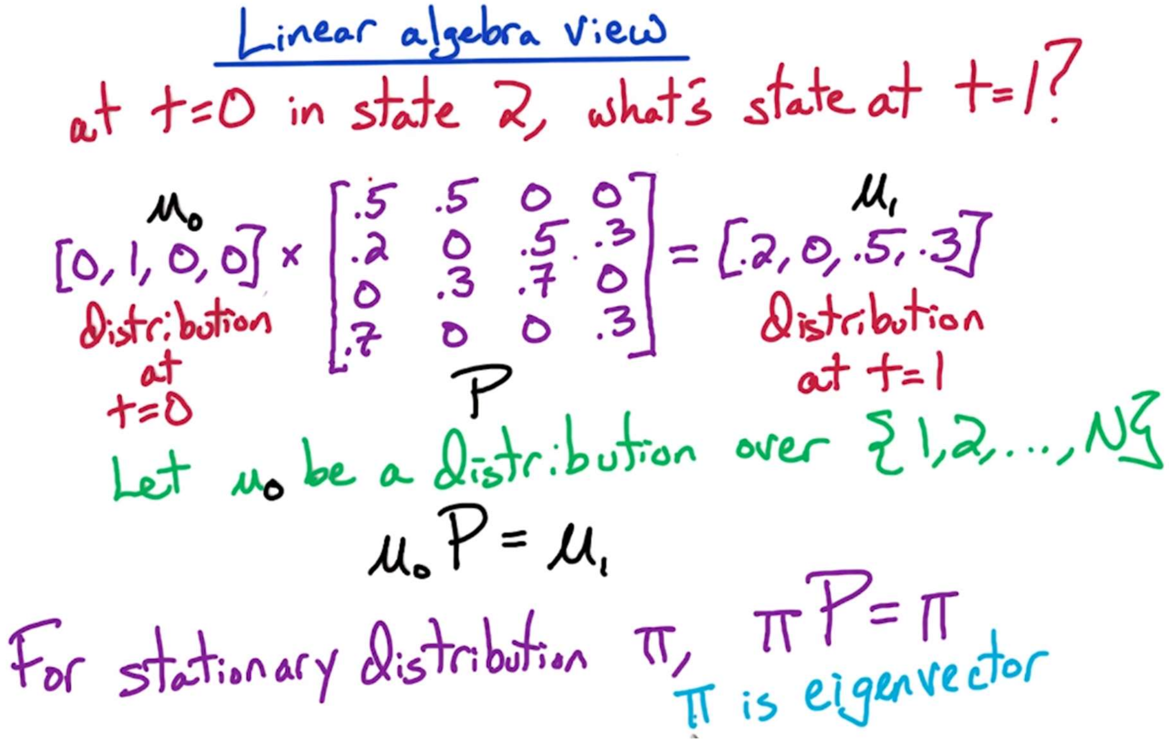
Then, we want to look at what is this π? What is the stationary distribution? Now this is quite important - the stationary distribution – why? - For PageRank, what its going to correspond to … is we're going to do a random walk – a Markov chain on the webpages - and then the page rank is going to correspond to the stationary distribution of that Markov chain.

So all this technology is going to be useful when we're trying to understand the PageRank algorithm. Now, before we move on and look at details about stationary distributions, I want to look at it from another perspective.

I want to look at it from a linear algebra perspective. What does a stationary distribution π mean from a linear algebra perspective?

***

(Slide 8) Linear Algebra View



Linear algebra view

at $t=0$ in state 2, what's state at $t=1$?

$$\underset{\substack{\text{Distribution}\\\text{at}\\t=0}}{\overset{\mu_0}{[0, 1, 0, 0]}} \times \underset{P}{\begin{bmatrix} .5 & .5 & 0 & 0 \\ .2 & 0 & .5 & .3 \\ 0 & .3 & .7 & 0 \\ .7 & 0 & 0 & .3 \end{bmatrix}} = \underset{\substack{\text{Distribution}\\\text{at } t=1}}{\overset{\mu_1}{[.2, 0, .5, .3]}}$$

Let μ_0 be a distribution over $\{1, 2, \ldots, N\}$

$$\mu_0 P = \mu_1$$

For stationary distribution π, $\pi P = \pi$

π is eigenvector

Now here's our running example once again.

Now let's suppose that at time 0, I'm in state 2 and I want to know the state at time t=1. What's the distribution for the state at time t=1? Well, how do I get it?

Why just look at this row. The second row of this transition matrix tells me the distribution for the state at time t=1. It's the one step transition matrix.

Now, what's another way to get row 2? - well, I can take this vector which has a 1 in entry two and zeros everywhere else and I can multiply this row vector by this matrix, and then what do I get? - I get row two of this matrix. So this is my distribution at time 0 and I multiply that by the transition matrix and I get the distribution at time t=1.   And, in general, at time 0, I don't have to be in a fixed state. I can be in a distribution over the states.

So let μ (mu) be an arbitrary distribution over the N states. So what exactly does that mean? - that means μ is this vector and this row vector of size N and it's a probability distribution - So, the sum of these entries is exactly 1 … and all these entries are between 0 and 1.

Now we're taking this vector ([0, 1, 0, 0]) by distribution μ. Then here we have P.   And, then we get a distribution at time t=1. So let's call this μ0 to denote the distribution at time 0. And, over here, we get a distribution for the time t=1 – so, let's call this μ1. In general, if you take μ0 - the distribution at time zero and multiply by this matrix P, the transition matrix, then, we get the distribution for the state at time t=1. So we take this row vector for the distribution of time 0, multiply by one step. So we do one step of our random walk and then we get the distribution - the row vector for the state at time t=1.

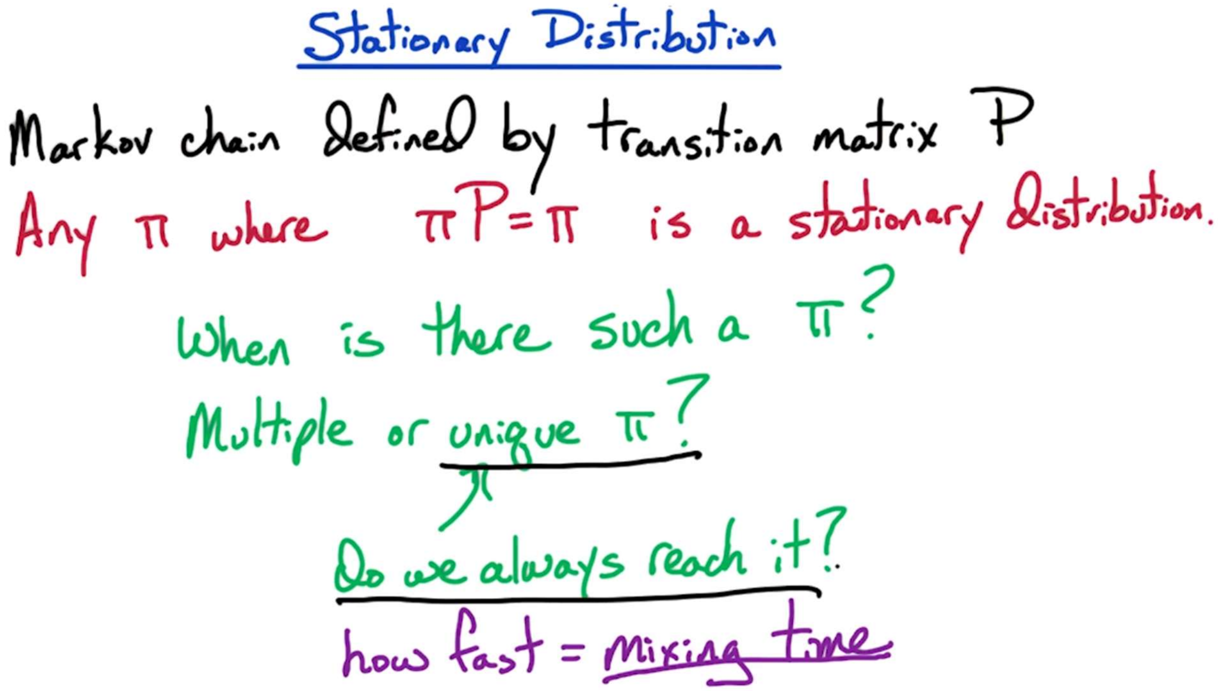
Now the key property is that for a stationary distribution π - for any stationary distribution π - so, if I am in the stationary distribution at time zero or at any time t and I look at the state at time t+1, then what is the distribution going to be? – well, once I reach this distribution, I stay in it. It's the limiting distribution. It's like a fixed point of the process. So once you reach a fixed point you stay in a fixed point and that's the same for a stationary distribution. So, if I'm in the stationary distribution at time t, and I do one step then I'm still in the stationary distribution π. So π P = π. What does that mean in terms of linear algebra?

Well this π is an **eigenvector** with eigenvalue 1. So π is an eigenvector for this matrix P and it's an eigenvector with eigenvalue 1. 1 turns out to be the largest eigenvalue for this matrix. So this is the principle eigenvector for this matrix.

Now, there can be multiple eigenvectors with eigenvalue 1. We're going to look at situations where we know that there is at most one eigenvector with eigenvalue 1. So there is at most one stationary distribution.

***

(Slide 9) Stationary Distribution



Stationary Distribution

Markov chain defined by transition matrix P

Any π where $\pi P=\pi$ is a stationary distribution.

When is there such a π?

Multiple or unique π?

Do we always reach it?

how fast = mixing time

Let's recap what we know so far about stationary distributions of a Markov Chain.

So let's consider a Markov Chain defined by the transition matrix P. Now, if our Markov Chain is defined on N states, then we consider any distribution π on those N states. so this is a row vector of size N. So, for any π which satisfies π P = π (so π once again) is a eigenvector with eigenvalue 1.

What this equation says is that, if we start in distribution π and we do one step of our random walk defined by P then we'll stay in the stationary distribution π, so π' isn't variant. Once we reach it we stay in it. Now this defines a stationary distribution.

Now, when is there such a π? What Markov Chains have a stationary distribution? And if there is one, is it unique or are there multiple ones? Under what properties do we have multiple or unique stationary distributions?

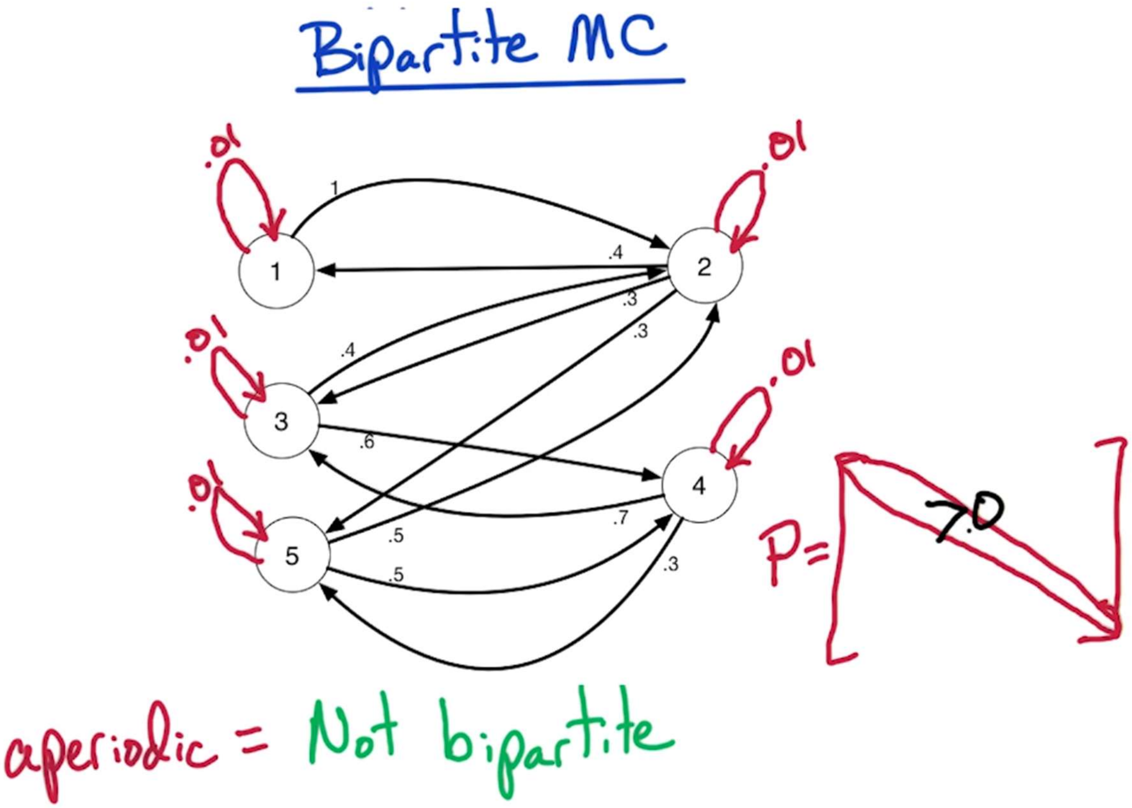
And for our simple 6210 example, we noticed that no matter where you started. You always reach the stationary distribution. So, in this case when there's a unique stationary distribution, regardless of where we start, do we always reach this stationary distribution? Finally, in this case where there is a unique stationary distribution and we always reach it, we could ask how fast we reach it. This is known as the **mixing time** of the Markov Chain - how fast the Markov

Chain reaches its stationary distribution. This is one of the things I study in my research, trying to prove bounds about the mixing time of Markov Chains. We're not going to look at the mixing time here but what we are going to look at the properties of the Markov Chain which ensure that we have a unique stationary distribution and that we always reach the stationary distribution.

So regardless of where we started from, we eventually, in the limit over time, will reach a stationary distribution. In order to see which properties ensure a unique stationary distribution, let's look at examples where we have multiple stationary distributions. That will give us some insight into the properties needed to guarantee a unique stationary distribution.

***

(Slide 10) Bipartite Markov Chain



Here's an example of a Markov chain whose graph corresponds to a bipartite graph. Let's look at some basic properties of this Markov chain.

Suppose we start our random walk from one of these vertices on the left side - either state 1, 3, or 5. What do we know? We know at time 1 we're on the right side, and actually at any odd time, we're on the right side. And, at any even time, we're on the left side.

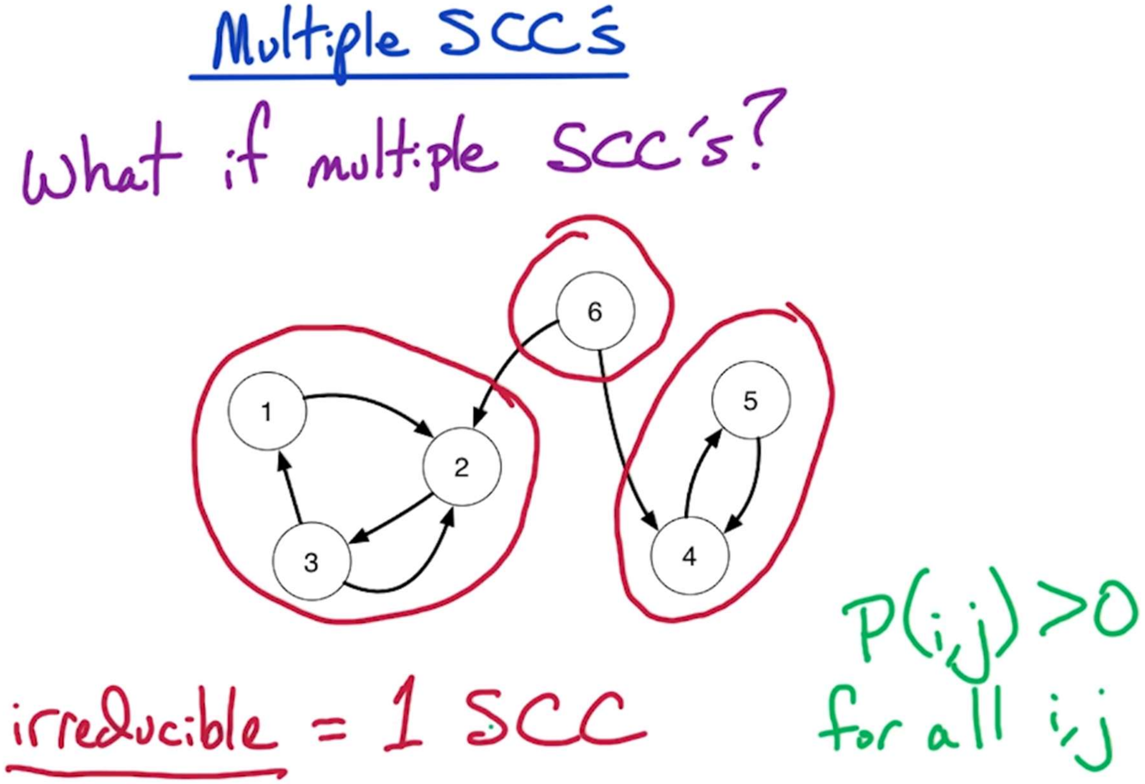
Now suppose we start on the right side, so at time 0, we're at either vertex 2 or vertex 5. Now, we have the opposite situation - at odd times, we’re on the left side; at even times, we're guaranteed to be on the right side. So, the punch line is that the starting state matters in this graph. Whenever we have a bipartite graph the starting state matters.

Now we want a simple way to ensure that our Markov chain is not bipartite, and even more so we want to ensure that the Markov chain has no periodic structure. Instead of being bipartite, having these cycles of period two, we could have cycles of period three. So in general, if we want to ensure that the graph has no periodic structure (that is aperiodic) – well, what's an easy way to ensure that? - an easy way is to have a self-loop on each vertex.

So, with some probability, we stay where we are. This can be a very small, miniscule probability. So let's say with probability .01 we stay where we are, and then we can rescale the other probabilities so that they sum up to one. In terms of their transition matrix, this means that our diagonals, the self-loops are diagonal entries in this transition matrix. We want all of these diagonal entries to be strictly greater than zero. So, we want the diagonal entries to be positive. If they're positive, that destroys any periodic structure. It can't be bipartite or any periodic structure. So for every state i, we'll make sure that the probability of going from state i and staying in state i is strictly greater than zero. Okay that gets around this pitfall.

***

(Slide 11) Multiple SCC’s



Multiple SCC's

What if multiple SCC's?

irreducible = 1 SCC

$P(i,j) > 0$
for all i,j

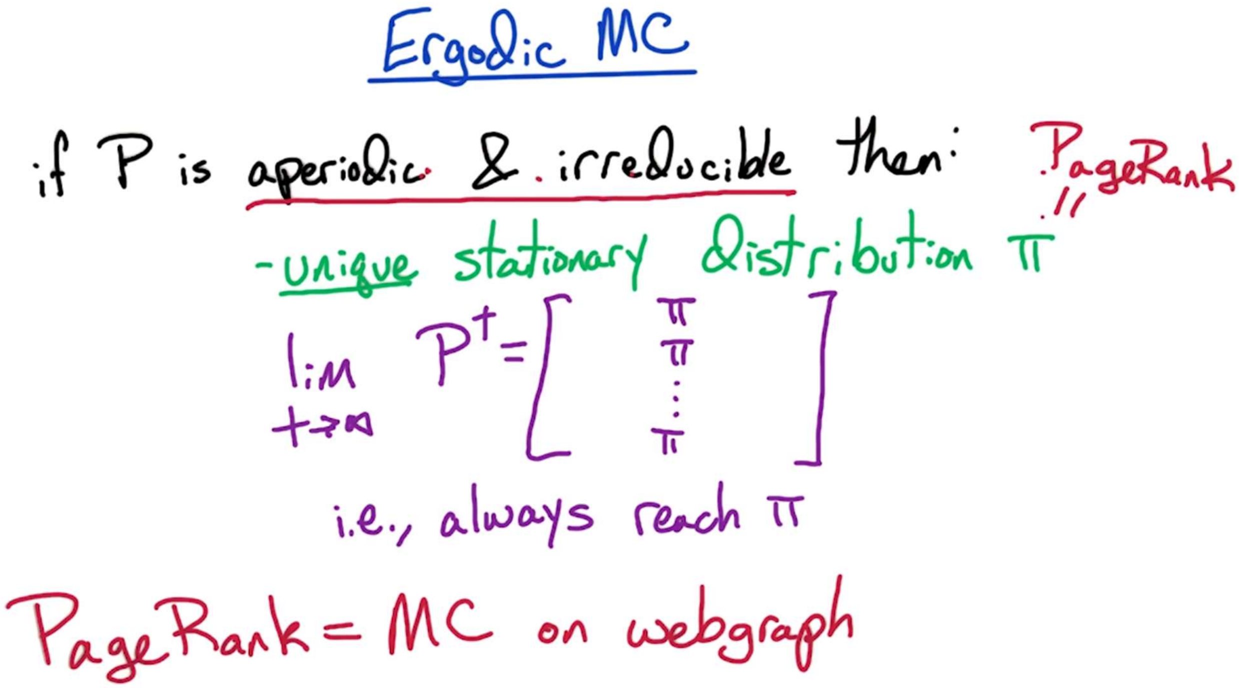
Now another pitfall that can happenis if our underlying graph has multiple strongly connected components. For example, in this graph, there are three strongly connected components. Now look, if we start at one of these three vertices we're only going to reach these three vertices. If we start at one of these two vertices, we only reach these two vertices. So the starting state definitely matters.

What we would like is that the graph has one strongly connected component. The terminology for this is that P is **irreducible**. If the graph has one strongly connected component - every pair of vertices are strongly connected with each other - then the transition matrix is irreducible.

Now what's an easy way to assure that the graph is one strongly connected component - well connect up every pair of vertices - make it the complete graph. For all pairs of states i and j, we make the entry P(i,j) be strictly greater than zero. So there's a positive probability of going between every pair of vertices. The matrix P is all positive, and therefore the graph is fully connected, so it's one strongly connected component. This is the easiest way to ensure that the graph is one strongly connected component and therefore irreducible.

***

(Slide 12) Ergodic MC



Ergodic MC

if P is aperiodic & irreducible then: PageRank ″

- unique stationary distribution π

$$\lim_{t \to \infty} P^t = \begin{bmatrix} \pi \\ \pi \\ \vdots \\ \pi \end{bmatrix}$$

i.e., always reach π

PageRank = MC on webgraph

These are the two key properties of a Markov chain: that it is aperiodic and irreducible.

Recall that aperiodic means that the underlying graph is not bipartite. How do we get around it? We add self-loops, and that ensures that the graph is not bipartite and has no aperiodic structure.

Irreducible means that it has one strongly connected component. How do we ensure that? By making sure that the graph is fully connected. All pairs of states can get between each other in one step, there's an edge between every pair of vertices.

Now, a Markov chain which is aperiodic and irreducible is called **Ergodic**, and that's the key property.

An Ergodic Markov chain has nice properties. A Markov chain with these two properties has a stationary distribution π and is unique. There is exactly one stationary distribution. Moreover, we have the following nice property for P^t: for some big enough t, then the matrix looks like π, π, π, every row is π.

What does that mean? - that means regardless of where we start (so, we are starting at one of these rows, and then we're doing t steps of our random walk), we're going to reach this distribution π. So, no matter the starting state, we eventually reach the unique stationary distribution π. This was the same scenario that happened for our simple example on four states for the 6210 example. So in other words, we always reach π no matter where we start.

How is page rank going to be defined? Well, page rank is going to be defined by looking at a Markov chain on the webgraph, so, we're going to do a random walk on the webgraph, the sites, the vertices are going to be webpages, the edges are going to correspond to hyperlinks.

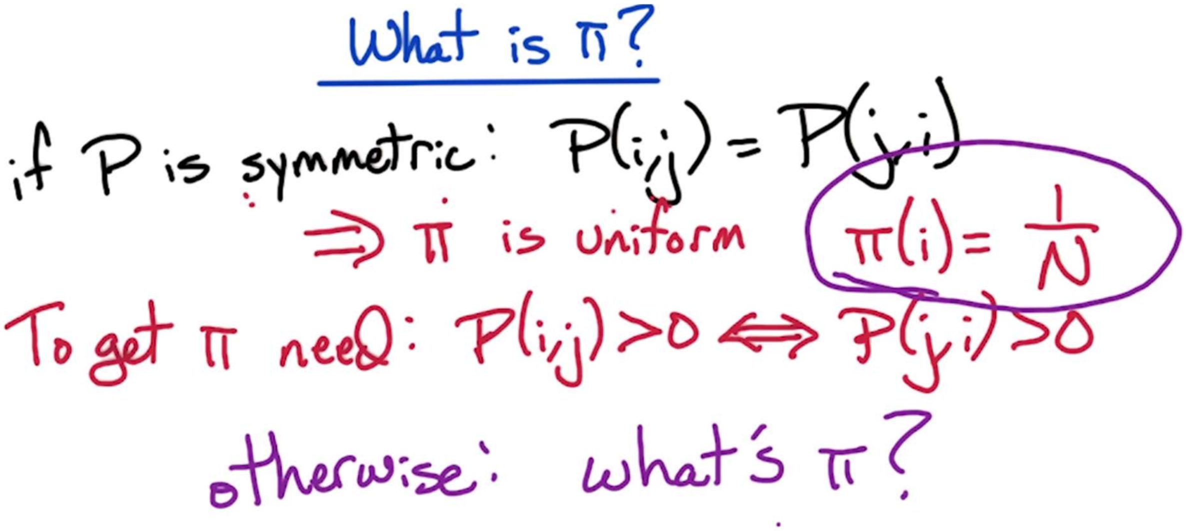
Now, we're going to have some technicalities to ensure that the underlying Markov chain is aperiodic and irreducible. How are we going to solve them? Exactly as we mentioned earlier. We're going to add self-loops and we're going to add to it to be fully connected and then the page rank is going to be defined as the stationary distribution of this Markov chain.

Since it's Ergodic, there's a unique stationary distribution and no matter where we start as random walk, we'll always reach this unique stationary distribution. So, it's well-defined, the stationary distribution will correspond to the page rank.

So, our measure of the importance of a webpage will be related to the probability of ending at that webpage. We start a random walk from any state, we run the random walk for many steps. What's the probability we end at a particular webpage j? That corresponds to the page rank - the importance of the webpage j.

***

(Slide 13) What is Pi?



What is π?

if P is symmetric: $P(i,j) = P(j,i)$

$\Rightarrow \pi$ is uniform $\pi(i) = \frac{1}{N}$

To get π need: $P(i,j) > 0 \Leftrightarrow P(j,i) > 0$

otherwise: what's π?

Now what is this Stationery Distribution π? Is there a nice formula for this π? In general, no; but, in some cases, yes, there is.

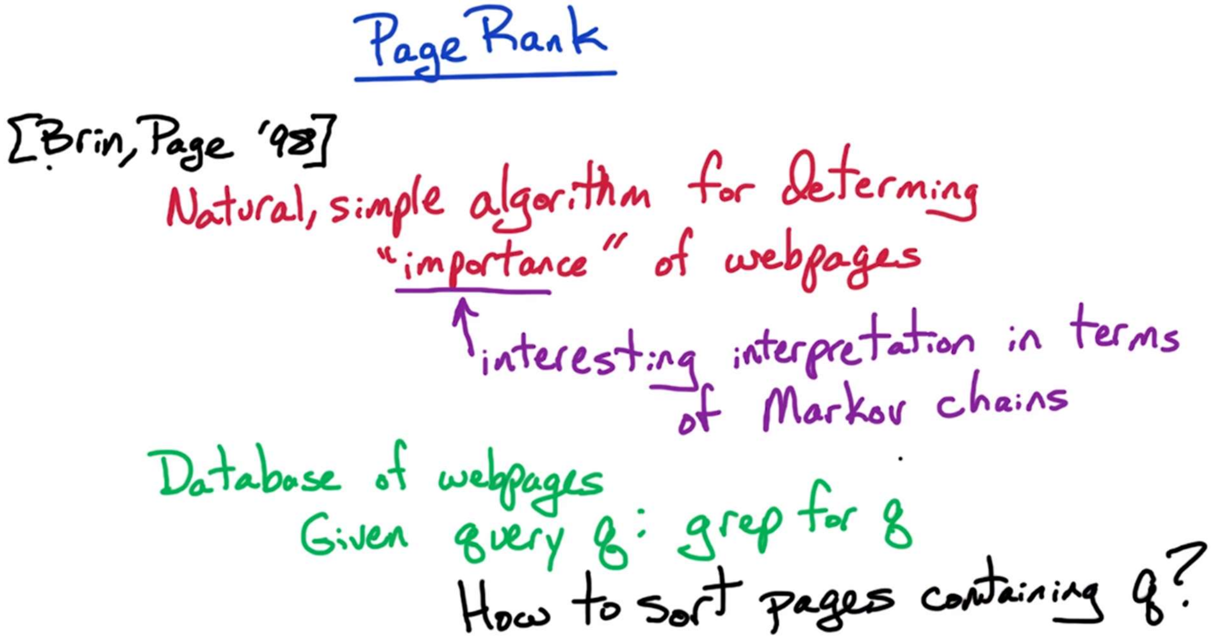
A simple case is when P is symmetric - what does this mean? - that means the entry P(i,j) = P(j,i). So if I look at row i and column j, the probability of going from i to j, is the same as the probability of going from j to i. If that's the case then this is a uniform distribution. π(i) = 1/N for all i. Now that's the simplest case.

A generalization is known as reversibility. It's kind of a weighted version of symmetry in which case π is not necessarily uniform but we can still figure out the π easily. But what reversibility requires is that, if there's an edge from i to j then there has to be edge from j to i. Now symmetry says that these probabilities are the same. We don't necessarily need them to be the same; but, we need that, if there's an edge from i to j, then there's an edge from j to i and vice versa. Now if this is the case, then we might be able to figure out what π is - we might have a nice formula for what π is. If this is not the case - the chain is not reversible … so, for instance, there might be an edge from i to j but there is no edge from j to i - then, in general, we have no idea what the stationary distribution - there is no way to figure it out with a closed formula. Now we can try to look at P to some high power and figure out what the stationary distribution is; but, we're not going to get some nice formulas such as this, for the stationary distribution. That completes our description of the introduction to Markov chains.

Now we can dive into the details of the PageRank algorithm.

***

(Slide 14) PageRank



PageRank

[Brin, Page '98]

Natural, simple algorithm for determing "importance" of webpages

interesting interpretation in terms of Markov chains

Database of webpages

Given query q: grep for q

How to sort pages containing q?

Now we're going to talk about the PageRank algorithm. This is the algorithm invented by Brin and Page. It was published in 1998. We're going to start by forgetting about Markov chains. You can understand the basic idea of the PageRank algorithm without even knowing what a Markov chain is.

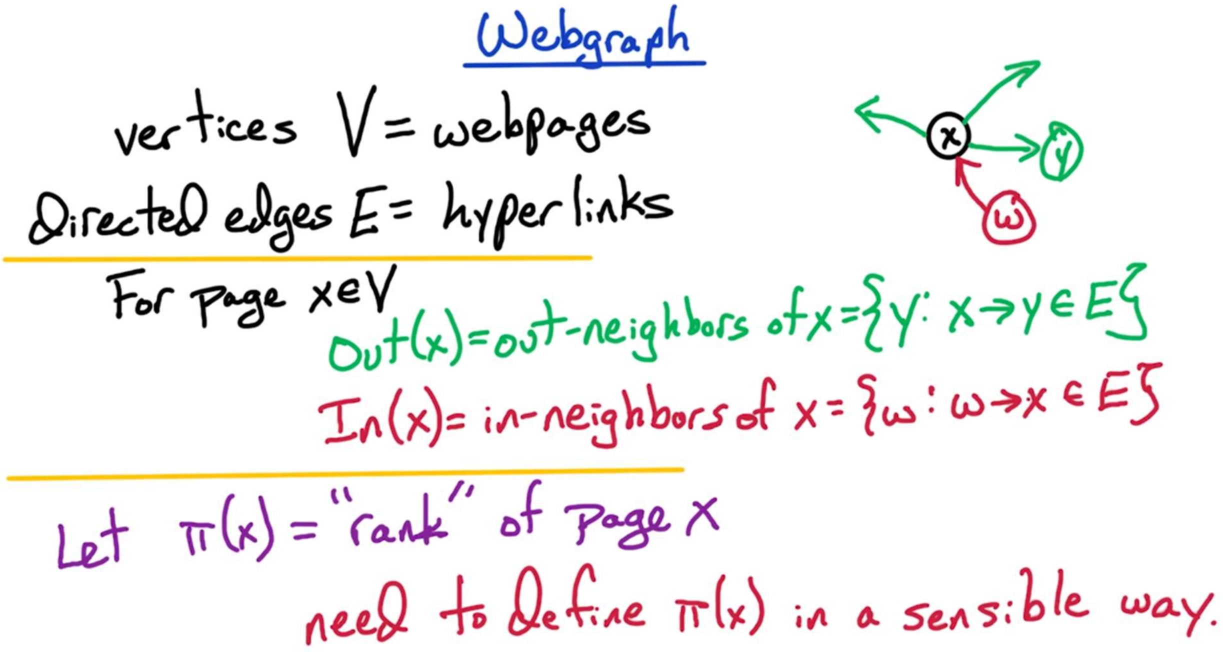
PageRank is a natural simple algorithm for determining the importance of webpages. Now importance is a subjective term and it's something that we're going to have to quantify ourselves. Now part of the appeal of PageRank is how they define importance has an interesting interpretation in terms of Markov chains. Now we're going to get to that at the end of the lecture. For now, we'll forget about Markov chains and we'll just look at the simple idea for defining the page rank.

To understand the appeal of PageRank, let's go back and think about how search engines worked in the mid-90s. Well, the search engines would maintain a database of webpages and then given a query term Q, what would they do? They would do a grep for Q. So they would search for all webpages containing that query term Q. Now it's easy to spam or trick those search engines so that many of the common query terms are embedded in your webpage. That's another issue, let's ignore that for now. The bigger issue is that you have many webpages that contain this query term and now how do you present it to the user? How do you sort the webpages? You want to put the most relevant webpage up front so there are many webpages, let's say, containing this query term Q. How do you sort them? Well, that's where the importance of the webpage comes in. We're going to put the most important webpages containing this query term at the beginning of our list. For example, if you search for Markov chains and CNN happens to have an article about Markov chains (I'm not sure why that might be the case but let's just say). CNN has an article about Markov chains and I have on my webpage I have several lecture notes about Markov chains. Now when we do this search, this grep, both webpages, mine and CNN are going to contain the query term for Markov chains.

Which one should be presented first? Well, presumably the user is more likely to be interested in the CNN article about Markov chains rather than my lecture notes about Markov chains. So let's dive into the algorithm description.

***

(Slide 15) Webgraph



Webgraph

vertices V = webpages

Directed edges E = hyperlinks

For page $x \in V$

$\mathrm{Out}(x)$ = out-neighbors of $x = \{y : x \rightarrow y \in E\}$

$\mathrm{In}(x)$ = in-neighbors of $x = \{\omega : \omega \rightarrow x \in E\}$

Let $\pi(x)$ = "rank" of page x

need to define $\pi(x)$ in a sensible way.

We're going to look at the so called Webgraph. The graph on webpages.

What exactly is our graph? – well, the vertices of our graph are going to be webpages. Now, this is humongous graph - because we have a vertex for every webpage and the edges of our graphs are the hyperlinks. Now, these are directed edges.

A webpage x might have a hyperlink to a webpage y or y doesn't necessarily have a hyperlink back to x. So, it's important we think of these edges as directed edges.

Now, let's introduce some notation, for a webpage x, let's define π(x) to be the rank of this page. The rank is our measure of the importance of the webpage. Rank or importance are subjective terms. We need to define π(x), the rank, in a sensible way. Of course, sensible is also a subjective term, but, in any case, we're going to define π(x) so that it has a very nice natural mathematical interpretation. If you just watched the bit about Markov chains, you might think “why is he using π once again? - well, it's not a coincidence, before we used π to correspond to the stationary distribution and that's going to come out later, but for now, once again, we're not going to talk about Markov chains.

Now, it would be useful to have a little bit on notation. So, let's consider a page x is the vertex in our graph, so it's denoted here. Now there are a bunch of hyperlinks at the webpage x.

Those are directed edges out of x. We want some notation for this set of neighbors which have edges from x to them. So we're going to define this set out(x) = the out-neighbors of X. These are the webpages which have a hyperlink from x to them. So these are the webpages y, where there's a hyperlink from x to y, there is a directed edge from x to y. So out of x are the set of all ys such that there's a link from x to y.

Similarly, we're going to have to look at the set of webpages or vertices which have a link to x. So we'll let in(x) = the set of in-neighbors of X. This is a set of webpages {w: w-> x є E} which have a hyperlink from w to x. So, there's a directed edge from w to x.

So just remember, out(x) are the out-neighbors of x and in(x) are the in-neighbors of x. This is the only totation that we'll need.

***

(Slide 16) First idea

1st idea

Like academic Papers: use citation counts

= # of links to x

Set $\pi(x) = |In(x)| = \#$ of links to x

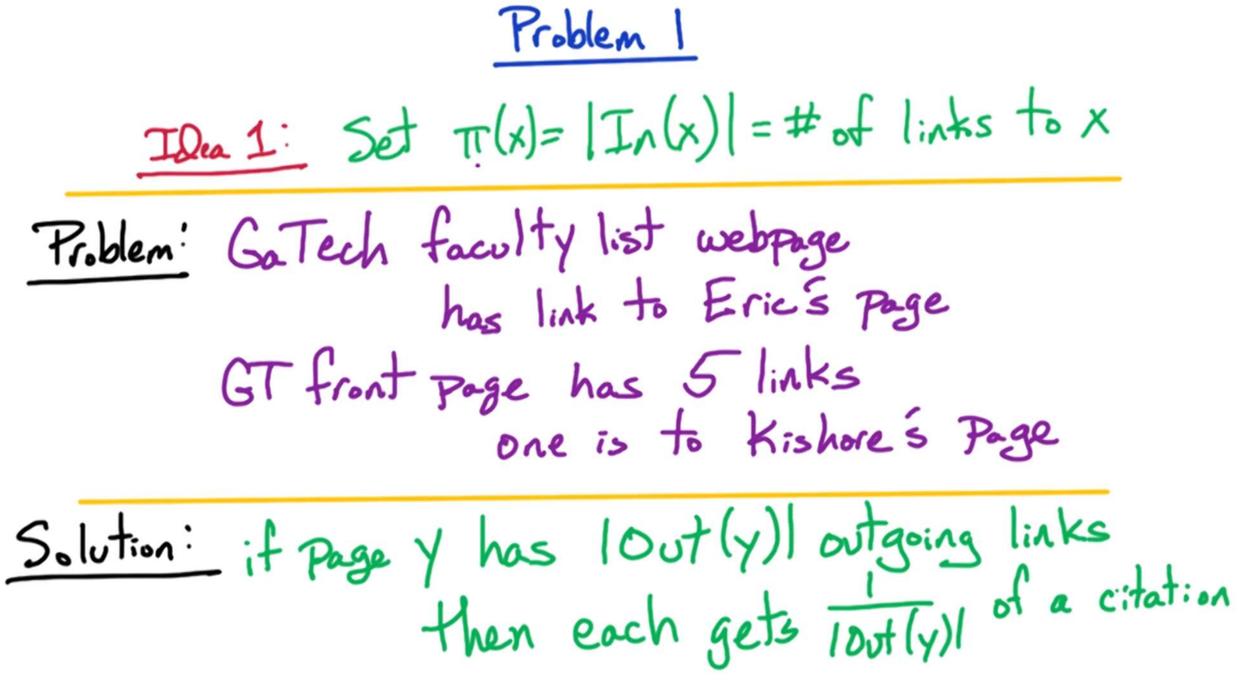
So imagine you're studying for your PhD, and you're trying to come up with some measure for the importance of webpages. So you might think of the analogy with academic papers. What's the importance of an academic paper? How do you measure the importance of an academic paper?

Well one way, that we still use, is using citation counts. How many other papers cite your paper? What does that mean in terms of webpages? – well, might be the number of links, the hyperlinks to your webpage.

So, our first idea for defining the rank of a page x, is to define π(x) to be the number of links to the page x, the number of hyperlinks to the webpage x. In terms of the graph, this means we're going to look at the number of in-neighbors to x - how many edges come into x.

***

(Slide 17) Problem 1



Problem 1

Idea 1: Set $\pi(x) = |In(x)| = \#$ of links to x

Problem: GaTech faculty list webpage
has link to Eric's page
GT front page has 5 links
one is to Kishore's Page

Solution: if page y has $|Out(y)|$ outgoing links
then each gets $\frac{1}{|Out(y)|}$ of a citation

So here was our first idea. We set the rank of a page x to be the number of links into the page x.

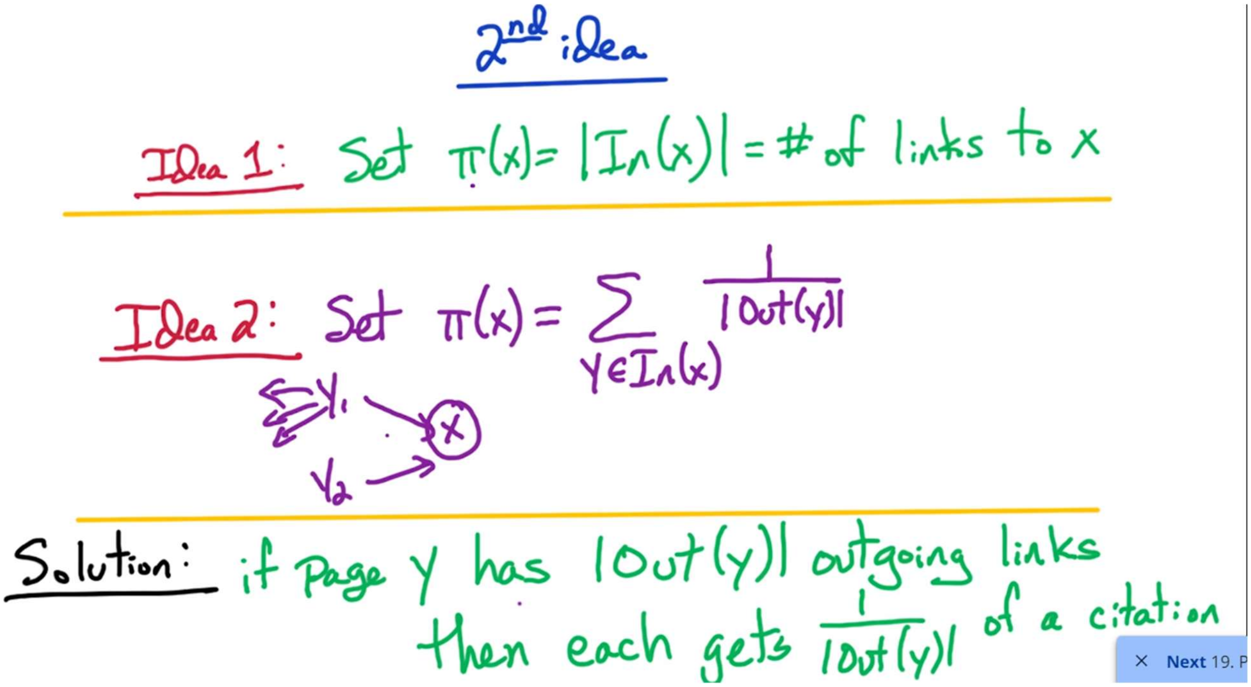
So π(x) is the in-degree of x. Now, what's the obvious problems with this? - well, here's a simple example. Let's suppose that Georgia Tech has a webpage which lists all the faculty … okay that's probably true …and the faculty list is probably a thousand long. One of those is going to link to my webpage. So, one out of a thousand of these links links to my webpage.

Now let's suppose that Georgia Tech's frontpage or the COCs frontpage has only five links on it and one of those happens to be the Kishore's webpage. Now, in the current measure of the rank of a page x, I get a +1 for this link and Kishore gets a +1 for this link. So both of those links count the same for us.

Now, that doesn't seem so fair. The former is one out of a thousand.   The latter is one out of five. So how do we get around it? Well, the obvious way is to scale it by the number of links. The natural solution to this problem is if George Tech's webpage has a thousand links and one of them goes to my webpage then I get 1/1000 and if Georgia Tech's front page has five links and one of those is to Kishore, then Kishore gets 1/5th of a citation. And, in general, if a page y has these many outgoing links, then each webpage is going to get one over the number of outgoing links of a citation. So Georgia Tech's faculty list webpage has a thousand links let's say, so I'm going to get 1/1000 of a citation. This webpage has five links, one to kishore's, so Kishore is going to get 1/5th of a citation.

***

(Slide 18) Second Idea



2^{nd} idea

Idea 1: Set $\pi(x) = |In(x)| = \#$ of links to x

Idea 2: Set $\pi(x) = \sum_{y \in In(x)} \frac{1}{|Out(y)|}$

Solution: if page y has $|Out(y)|$ outgoing links then each gets $\frac{1}{|Out(y)|}$ of a citation

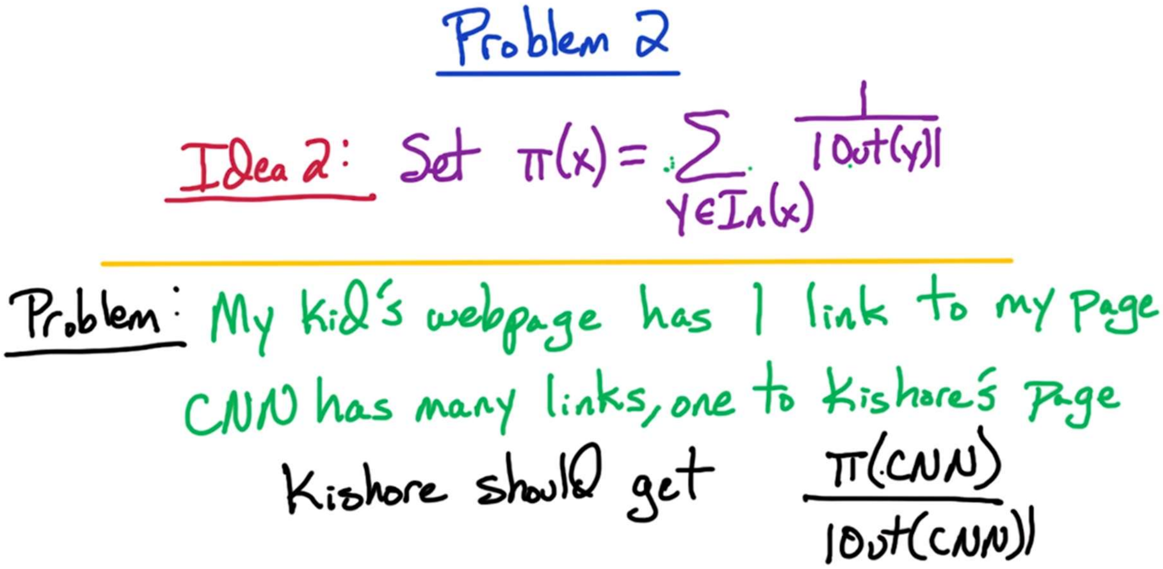
So we've got a webpage x. We're going to look at the in-neighbors.

So these are the webpages y which have a link to x. So we're going to sum over these. The old scheme just counted the number of these in-neighbors. So the old scheme you can view it as a sum over the in-neighbors and we get +1 for each in-neighbor.

In the new scheme, we want to scale it by number of outgoing links from each of these webpages. So, if y1 has a lot of outgoing links, let's say it has 1000, then this link is going to give us 1/1000. And, if this webpage has only one link to x, then this one gives us 1. And in general, from a webpage y, we're going to get 1/(the number of out-links from y). And we're going to sum this up over the y's which have a link to x.

***

(Slide 19) Problem 2



Problem 2

Idea 2: Set $\pi(x) = \sum_{y \in In(x)} \frac{1}{|Out(y)|}$

Problem: My kid's webpage has 1 link to my Page

CNN has many links, one to Kishore's Page

Kishore should get $\frac{\pi(CNN)}{|Out(CNN)|}$

So here's our current proposed solution. So, instead of setting the rank of the page x to be the number of in-neighbors - the number of links into x - we've scaled each of those links by the number of out-neighbors from that page y. Now, there are some obvious problems with this. Let's look at one example.

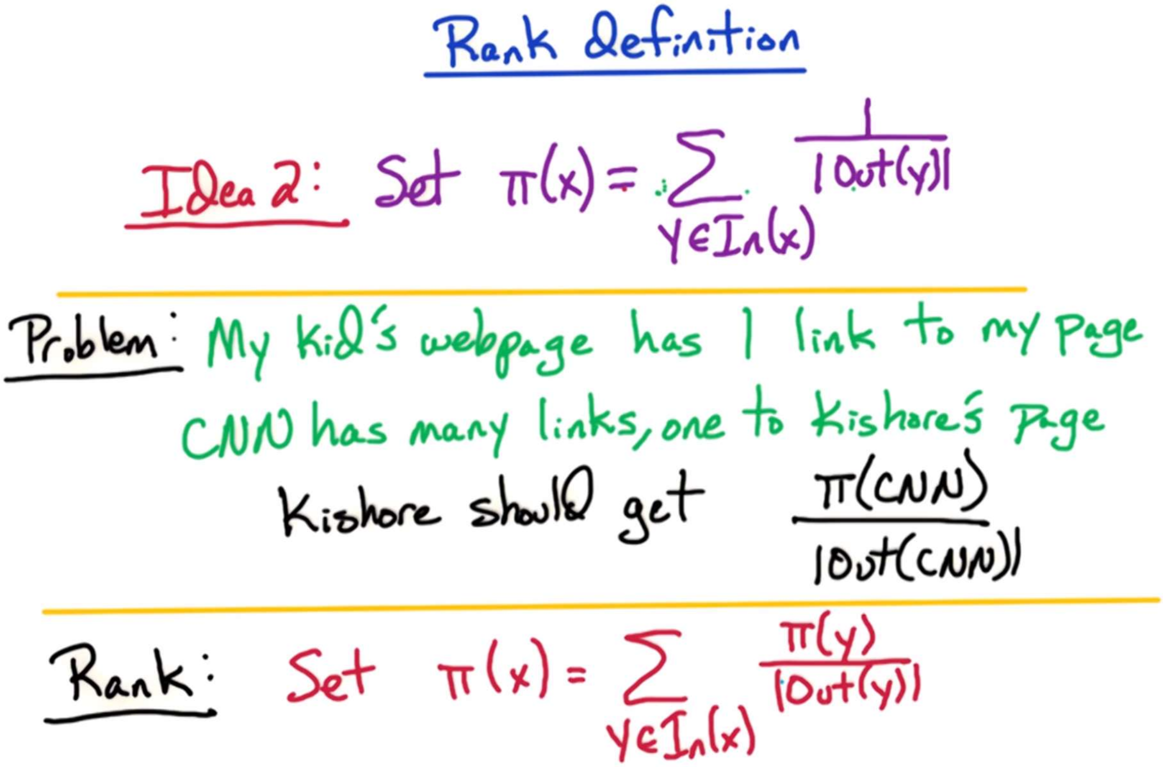
Let's suppose that my kids made a webpage and it has only one link on it, to my webpage. So under this count, I get one citation for it and now let's look at CNN webpage. They probably have many links on their webpage but perhaps they have an article about Kishore's awesome new research and they have a link to Kishore's webpage. So if they have 100 links, Kishore is going to get 1/100th of a citation here whereas I'm going to get 1 citation.

Now this doesn't seem very fair because CNN's webpage is very important. So, we should scale this citation from CNN based on the importance of CNN's webpage. So instead of a webpage having 1 citation to give out, the webpages should have π(x) - its rank - to give out. So CNN should have π(CNN) - its rank - that's how many citations it has to give out - that's the worth of its citations to give out.

Now, we can scale each of those citations as we did before based on how many links CNN has. So CNN has 1000 links in each of these webpages which has a link from it - is going to get 1/1000ths of CNN's worth. So CNN’s total citation value is π(CNN) - that's its rank - and then each webpage that has a link from it, is going to get 1/1000th of that worth - the recommendation value. So, a citation from a more important webpage, has more value because it's going to be proportional to the value of the rank of that webpage.

***

(Slide 20) Rank Definition



Rank Definition

Idea 2: Set $\pi(x) = \sum_{y \in In(x)} \frac{1}{|Out(y)|}$

Problem: My kid's webpage has 1 link to my Page

CNN has many links, one to Kishore's Page

Kishore should get $\frac{\pi(CNN)}{|Out(CNN)|}$

Rank: Set $\pi(x) = \sum_{y \in In(x)} \frac{\pi(y)}{|Out(y)|}$

So, let's go ahead and formalize this idea. So, for a citation from webpage y, its value is going to be proportional to the importance the rank of webpage y. So, we're going to scale this quantity by π(y) - formally, the rank of the webpage x. We're going to get it by looking at this sum over all webpages y, which have a link to x. The total value of the citations from y is π(y). And y has this many outgoing links. This is the number of out-neighbors.

So, this link from y to x has value π(y) / (the number of out-neighbors of y). And this is going to be the definition of the page rank of webpage x.

Well, now to be precise, this is not exactly the definition of the page rank of a webpage x. There's a technical glitch that we'll notice by looking at this from the perspective of Markov chains. Before we dive back into Markov chains, let's look closely at this proposed definition of the rank.

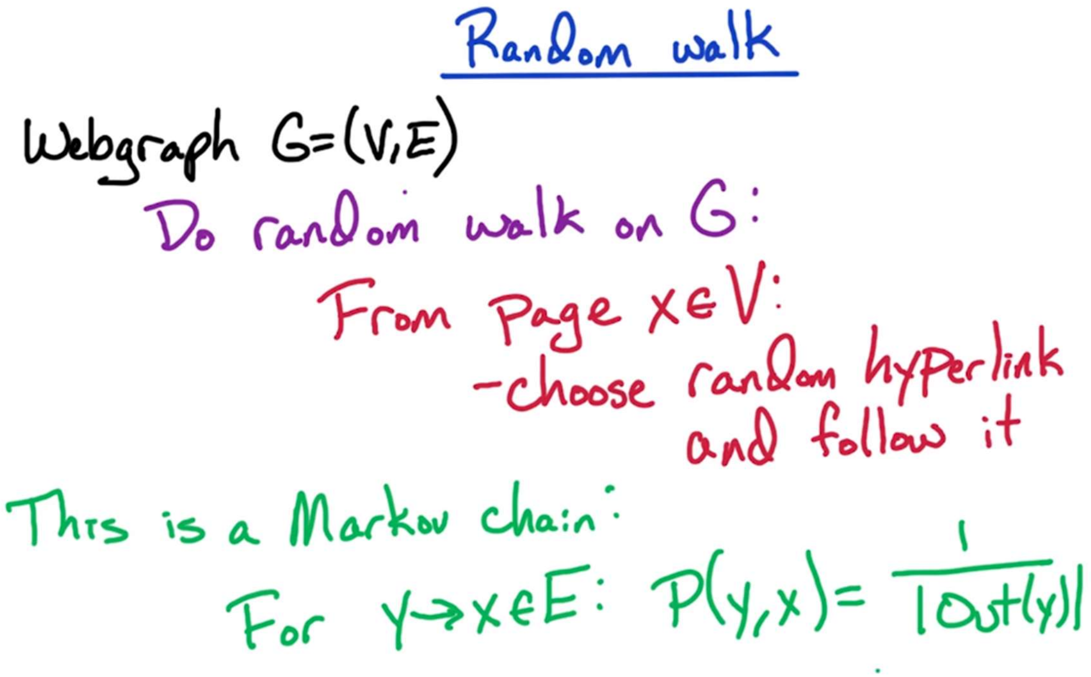
It's a recursive definition. So, first off, is it well defined? Is there a π satisfying it for every vertex x? This question of whether there exists a π satisfying it corresponds to whether there exists a stationary distribution for the corresponding Markov chain.

Now, if there does exist such a π, is there any unique such π … or are there multiple π's? This corresponds to the question of whether there is a unique stationary distribution or multiple stationary distributions. We'll address both of these questions using our intuition from Markov chains.

First, we'll derive this definition using Markov chains and then we'll see how to ensure that there is a unique stationary distribution. So, let's dive back into Markov chains now.

***

(Slide 21) Random Walk



Random walk

Webgraph $G=(V,E)$

Do random walk on G:

From Page $x \in V$:

-choose random hyperlink and follow it

This is a Markov chain:

For $y \to x \in E$: $P(y,x) = \frac{1}{|Out(y)|}$

Let's look at this problem from a completely different perspective.

So we have the webgraph G. The vertices are webpages and the edges are the hyperlinks. These are directed edges. Now let's do a random walk on this graph. What exactly does that mean?

It's just like you're surfing the web. So you write a webpage, you're going to follow a random hyperlink from that webpage, go to the next webpage, look at it for a second, hit a random hyperlink and so on.

So, we started at some webpage - say we're currently at webpage x. Then let's choose a random hyperlink - so uniformly, at random, from all hyperlinks on this webpage x. And, then we follow that hyperlink and we repeat the procedure from the new webpage.

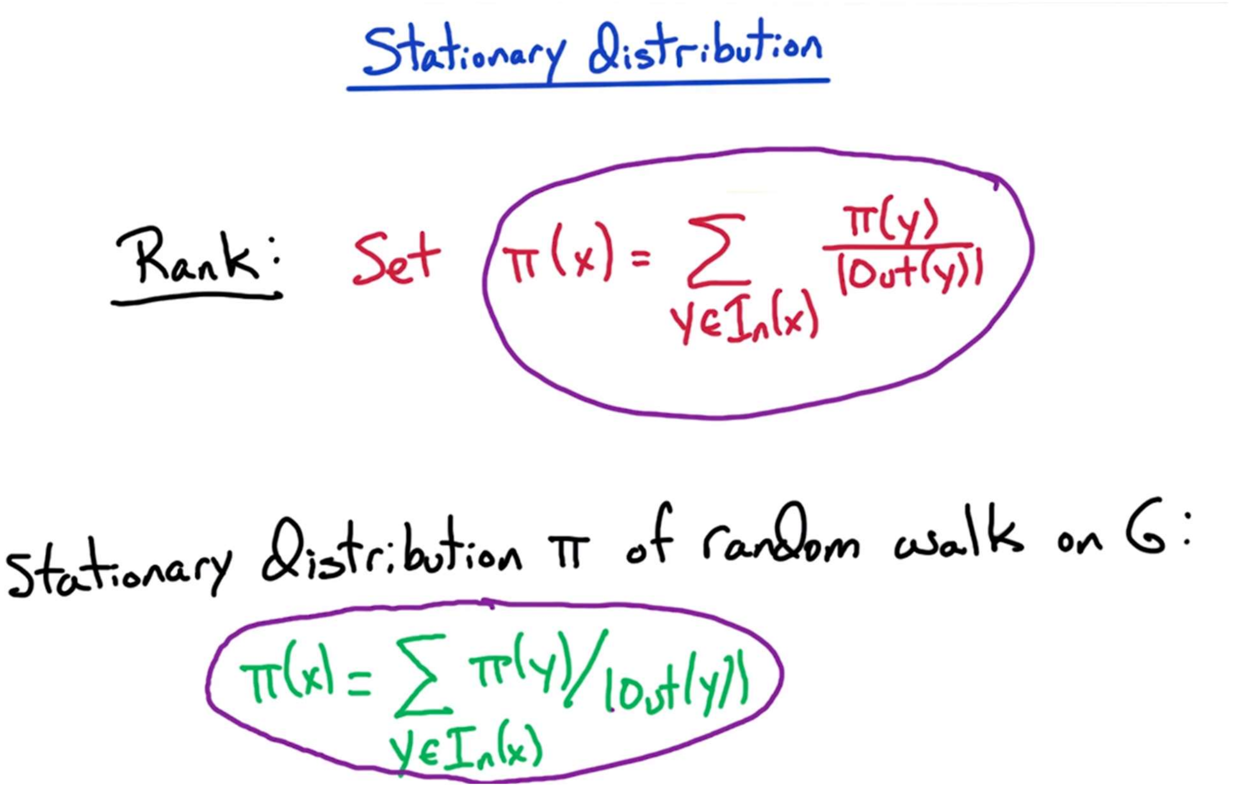
This is a random walk on the directed graph. From a vertex, we're choosing a random outgoing edge. So this is the Markov chain. What's the transition matrix for this Markov chain? Well, for webpage y which has a hyperlink to a webpage x – so, there's a direct edge from Y to X in this webgraph - the the weight of this edge in the transition matrix, this entry P(y,x) is 1/|out(y)| - the probability of following this particular link when we're at webpage y is exactly one over the number of links at webpage y.

So if y has a thousand links, and the probability of following this particular link is 1/1000. And if webpage y doesn't have a link to webpage x, so there's no edge and this transition matrix is 0 at that entry. So this defines the transition matrix for this Markov chain.

Now this is a Markov chain. So it has a stationary distribution. What does this stationary distribution look like?

***

(Slide 22) Stationary Distribution



Stationary Distribution

Rank: Set $\pi(x) = \sum_{y \in In(x)} \frac{\pi(y)}{|Out(y)|}$

Stationary Distribution π of random walk on G:

$\pi(x) = \sum_{y \in In(x)} \pi(y)/|Out(y)|$

So recall what is a stationary distribution? A stationary distribution is any vector distribution π which satisfies the following identity, $\pi P = \pi$ (π times the transition matrix equals π). So π is an eigenvector of P with eigenvalue 1 or in other words, if I'm in distribution π and I do one step of the random walk to find by P, then I'm still in distribution pi. So π is an invariant distribution. Once I reach it I stay in it. Let's expand this out to see what this means a little bit more precisely.

I have π here. If there are a million webpages then π is a row vector of size a million. Now, I have P here. If there are a million webpages, then P is of size a million by a million.

Now, let me just flip this. So let us look at $\pi = \pi P$. So I just flip the left and right hand side and here's the right hand side, $\pi = \pi P$. Let's look at an entry x here. So $\pi(x)$, this x entry here. If there are a million webpages and let's say x is maybe the 900th entry. How do I get this entry?

Well, this x is going to define the column I looked at over here. So this is the 900th entry here and it's going to be the 900th column here. And, then, what I do … I take this column and then multiply by this row. So y is going to vary over this row and then Y is going to vary over the rows of this matrix.

I'm going to multiply the first entries together plus the second entries multiplied together and so on. So I'm going to sum over y (the number of y's is the number of webpages). So y is varying over all vertices in the graph - all webpages in the graph - and I take the y's entry over here which is π(y) and I multiply by the y's entry in this column which is the (P, y) over x. So I do the dot product of this row vector with this column vector and that gives me the x entry in π.

Now let's look at this term. What do we know about P(y,x) (this term)? What did we say that the transition matrix was for this random walk (for the last time)? - well, if there's an edge from y to x, it's P(y,x) = 1/|out(y)| and there's no edge from y to x, then P(y,x) = 0 - there's no probability of this random walk going from y to x in one step. So, for any y which does not have an edge to x, then this term is zero. So, we can drop it. So, we only have to look at y's which have an edge to x. In other words, we only have to look at y's which are in the in-neighbor set of x.

So we can simplify this sum over all y's to only y's which are in the in-neighbors – so, now we only consider those y's which have an edge from y to x and we get π(y) again and now we can replace this term P(y,x) by this quantity (1/|out(y)|). So we're going to divide π(y) by the number of out-neighbors of y because we know that if there's an edge from y to x then P(y,x) is exactly this. So we can replace P(y,x) by this (1/|out(y)|). So we have that π(x) equals this quantity (the sum of π(y)/|out(y)| over all y ϵ in(x)).

So what have we shown? We've shown that the stationary distribution of this random walk on the webgraph … ff we do this simple random walk on the webgraph … it's stationary distribution satisfies the following identity - this is what we just saw because π = π P and we expanded our π times P and we got this slightly simpler expression.

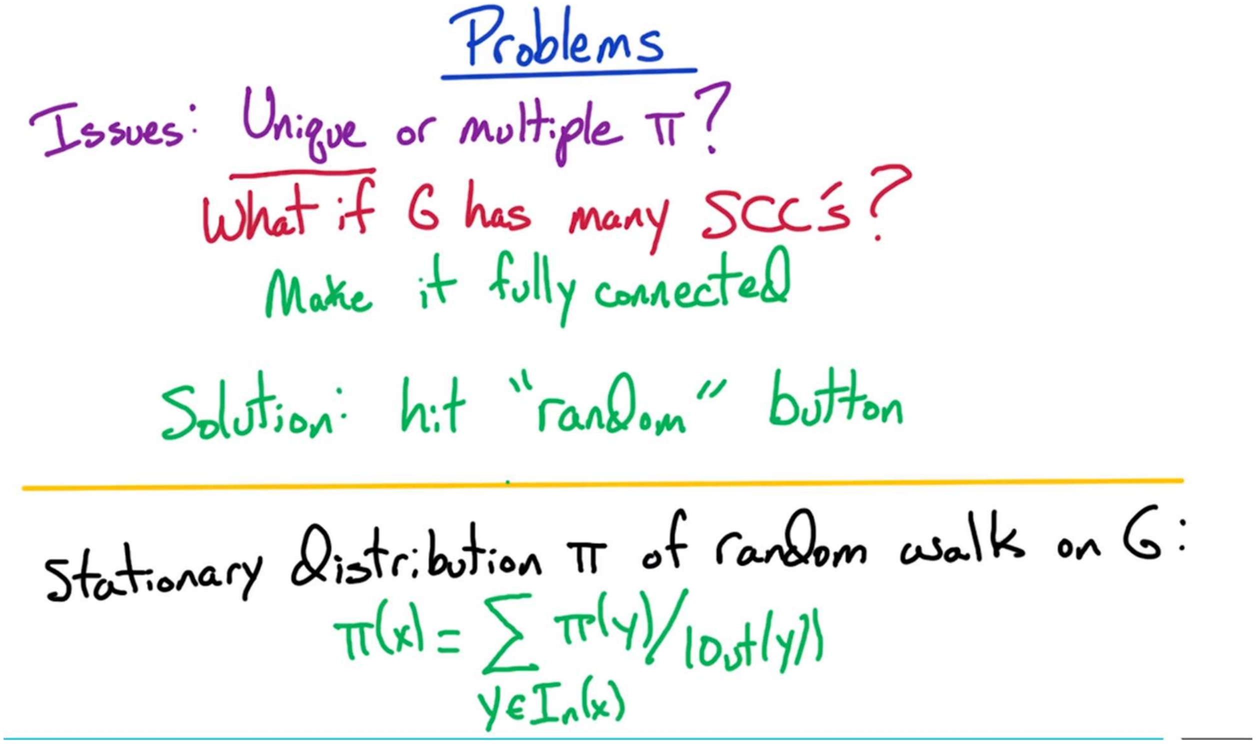
Now, where have we seen this before? Well, what we saw before when we ignored Markov chains and we defined the rank of a webpage in terms of citation count intuition. We got this definition. And look, they're identical. This definition and that definition are identical. So these two views are identical. This intuition from citation counts and this intuition from random walks - we get equivalent definitions.

Now this random walk interpretation is very appealing. Think about it. So you started at any webpage, you run the random walk. So, you just do random surfing. Now what's the chance you end up at a page x? Well, an important webpage like CNN or Google, I mean, there's probably a pretty good chance we're going to end up at that webpage when we do a random surfing. Whereas somebody is like my webpage … well, there's probably a very small chance that we're going to end up there.

So the stationary distribution of this random walk is a very natural nice appealing measure of the importance of a webpage. So, this is what we wanted to define the page rank of the webpage to be.

***

(Slide 23) Problems



Problems

Issues: Unique or multiple π?

What if G has many SCC's?

Make it fully connected

Solution: hit "random" button

Stationary distribution π of random walk on G:

$$\pi(x) = \sum_{y \in In(x)} \pi(y)/|Out(y)|$$

Now here's what we just saw. We saw that if we did the random walk on the webgraph, then a stationary distribution satisfies this identity. And, this is what we'd like to define the page rank for this webpage to be.

First off, is this well defined? Think back to our discussion about Markov chains and about stationary distributions of Markov chains. What are some of the key issues that can arise when we talk about stationary distributions of Markov chains? Was there a unique stationary distribution or are there multiple stationary distributions (i.e., well, if there are multiple stationary distributions and it's not clear which one we are referring to)?. Also, we want that no matter where we start this random walk, we're always going to reach the stationary distribution.

For instance, what if G has many strongly connected components? I mean, this is definitely the case. The webgraph is going to have many strongly connected components. So, there might be one strongly connected component, a very small strongly connected component containing my webpage. And, perhaps, if you start the random walk in that strongly connected component, you might have a high probability of ending at my particular webpage. Now, does that mean that I should have a high page rank? - probably not.

Now, there might be another strongly connected component, a very giant component and this giant component probably contains Google's webpage. And if you start the random walk in that giant component, then you probably have a good probability of ending at Google's webpage. So Google's webpage should have a high page rank because in that component it has a high rank.

So how can we ensure that the Markov chain has a unique stationary distribution? We want to ensure that there's only one strongly connected component and also we want to get rid of any periodicity. There might be some parts of the graph which are bipartite.

The easy way we discussed for making sure that there's a unique stationary distribution is to make the graph fully connected. So we're going to add edges, maybe with very small probability, but there will be edges between every pair of vertices. So, from any webpage, we'll have some positive probability of going to any other particular webpage.

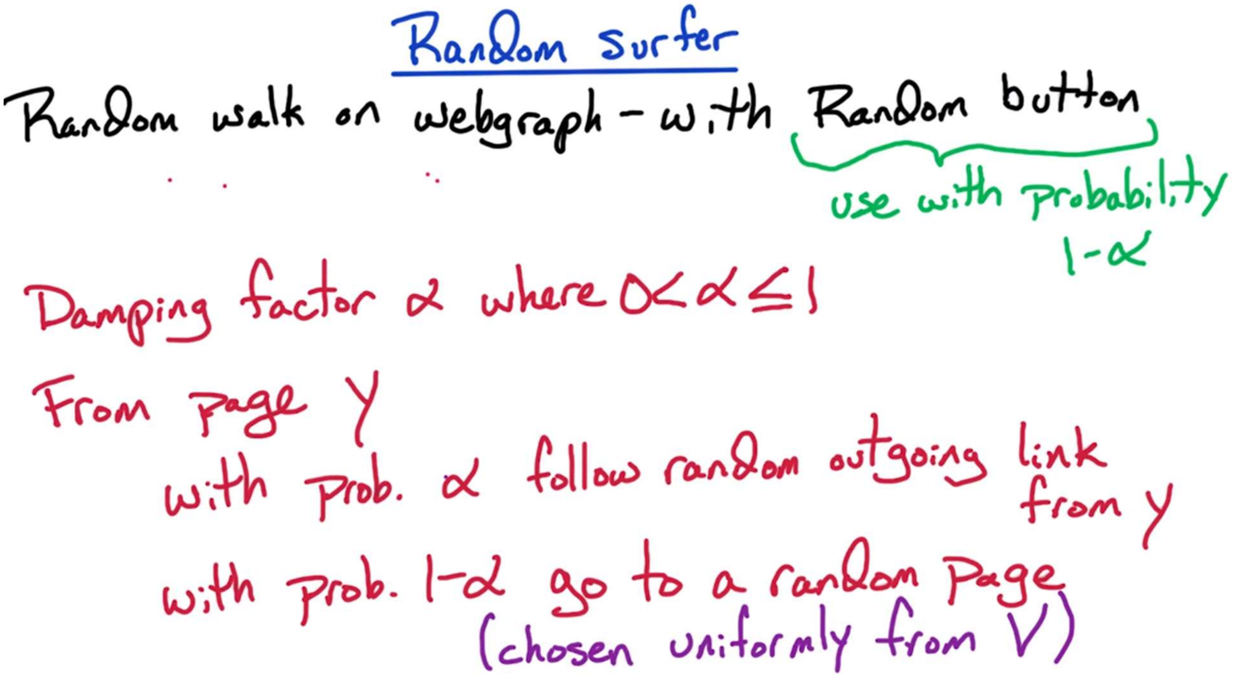
So, suppose your web browser had a random button. If you hit this random button, then it's going to take you to a random webpage. The webpage will be chosen uniformly at random from all webpages. So if there are a million webpages, you have probability 1/million of ending at any particular webpage.

Now, let's suppose you're surfing the web. What's the random walk going to look like? So, with some probability you're going to follow a random outgoing link from the current webpage and with some probability you're going to hit the random button. And that's going to take you to a random webpage in the whole graph. So, that random button is going to make the graph fully connected.

Let's formalize this random walk that we're talking about here.

***

(Slide 24) Random Surfer



Random surfer

Random walk on webgraph - with Random button

use with probability $1-\alpha$

Damping factor α where $0<\alpha\leq 1$

From page y

with prob. α follow random outgoing link from y

with prob. $1-\alpha$ go to a random page (chosen uniformly from V)

So let's formalize this notion of doing a random walk on the webgraph, where occasionally we hit the random button. And by hitting the random button, we're going to go to a random webpage.

So, we're going to have an additional parameter α and this α is going to be the probability that we hit the random button. Actually, to be consistent with the actual page rank definition, α's can be a complement of that event. So, with probability 1 - α, we're going to hit the random button and go to a random webpage in the graph. And with probably α, we're going to follow a random edge out of the current webpage.

So this parameter α is called the damping parameter. It's strictly greater than zero and it's at most one. Why is it called the damping parameter? Because what we're doing is we're scaling down this original webgraph by a factor α. So we're scaling down the webgraph by a factor α and then we're adding in a complete graph of weight 1 - α.

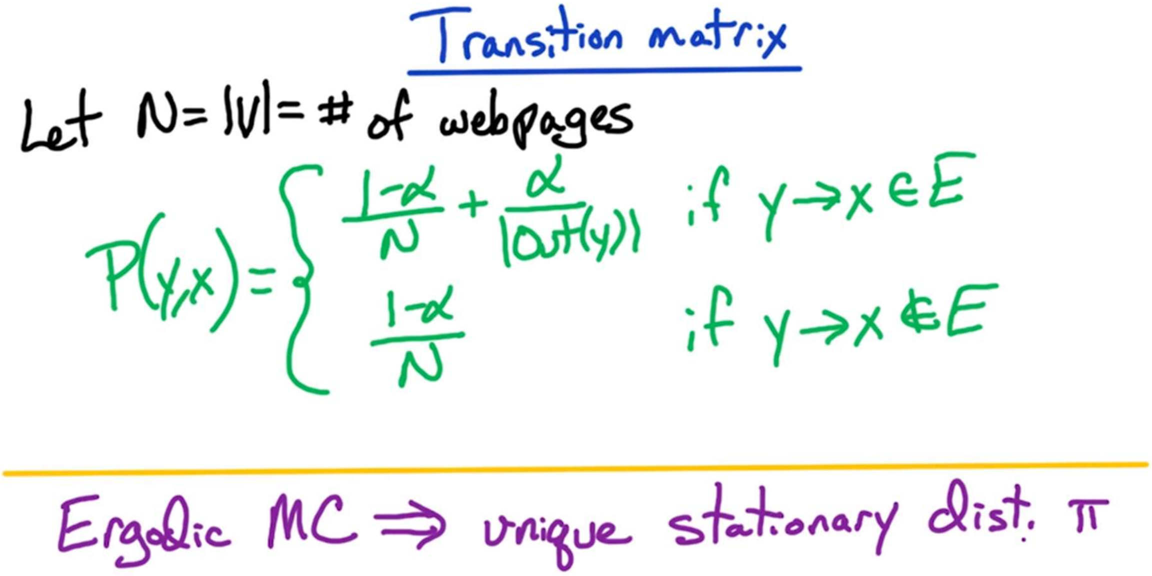
So now let's look at our random walk. Let's say we're currently at a webpage y. Then with probability α, we're going to follow a random outgoing link from y. So if $\alpha = 1$, this is exactly the same as our original random walk. So we're not using this random button at all, because $1 – \alpha = 0$. But when $\alpha < 1$, then we're going to use the random button sometimes.

So with probability 1 - α, we're going to go to a random page. This destination page is chosen uniformly at random from all webpages. So, this is our random surfer model. So you give me a parameter α and then my random walk looks like the following: I'm currently at a webpage y, with probability α. I'll look at all the outgoing links from y and I'll choose one of those uniformly at random. And with probability 1 – α, I'm going to go to a random webpage uniformly at random from all webpages in the graph. So I'm at this webpage y, I flip a coin or I choose a random number uniformly at random between 0 and 1. If this random number is at most α, then I choose a random outgoing link. If this random number is strictly greater than α , then I go to a random webpage uniformly at random from all webpages.

Now what α should we choose? - well, according to Wikipedia, Google apparently chooses an α which is at roughly 0.85.

***

(Slide 25) Transition Matrix



Transition matrix

Let $N = |V| = \#$ of webpages

$$P(y,x) = \begin{cases} \frac{1-\alpha}{N} + \frac{\alpha}{|Out(y)|} & \text{if } y \to x \in E \\ \frac{1-\alpha}{N} & \text{if } y \to x \notin E \end{cases}$$

Ergodic MC $\Rightarrow$ unique stationary dist. π

This random walk that we just defined is a Markov chain. Let's look at the precise definition of the transition matrix for this Markov chain.

Let's let N denote the number of webpages. So, this is the number of vertices in our directed graph. This is going to be humongous.

Now, we have this transition matrix of size N x N. Now, there are two cases, either y to x is an edge in the original graph or it's not. So let's look at these two cases separately. Either y to x is an edge in the original graph and there's a hyperlink from Y to X or there's no hyperlink from y to x.

If there's no hyperlink from y to x then the only way to go from y to x is to use the random button. What's the chance of that? Well, there's probability 1 - α that we hit the random button. Now, when we hit the random button, we go to a random webpage. So the probability of going to a particular webpage is one over the number of webpages – so, it's 1/N. So the chance of going from y to x using the random button is one over capital N. So, in this case, the probability of going from y to x if there's no hyperlink from y to x then it's 1 - α for hitting the random button. And then 1/N is the probability of going to this particular webpage x.

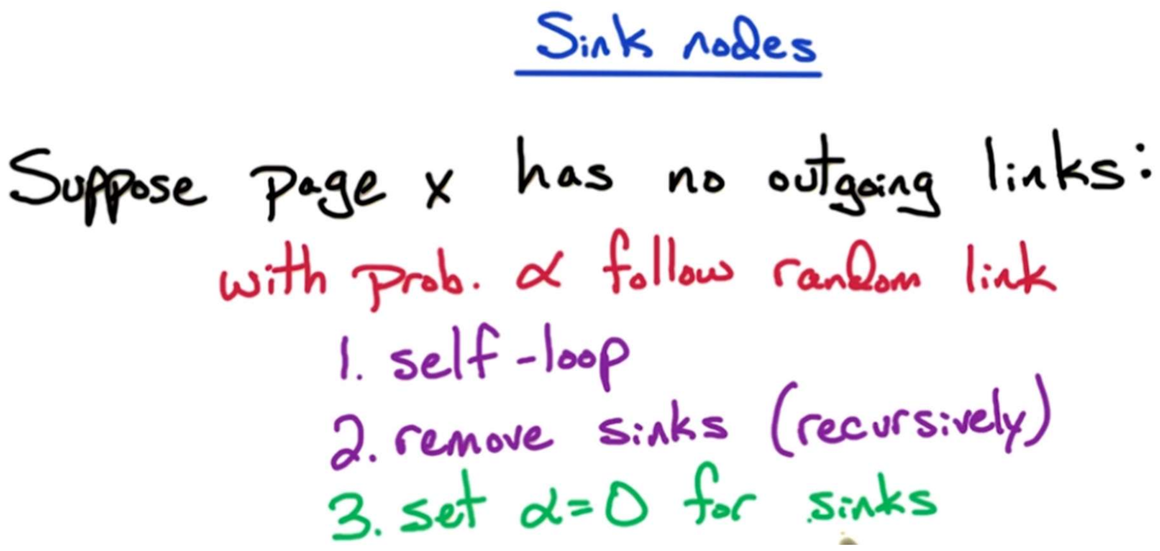
Now, if there is a hyperlink from y to x … well we can still get from y to x using the random button - probability of that is the same. But, we can also get from Y to X using the hyperlink. There's probability α of following a random hyperlink. And given where, in this case, the probability of following a particular hyperlink is one over the number of hyperlinks. The number of hyperlinks at y is the number of out-neighbors of Y.

Now, this transition matrix corresponds to a fully connected graph. Every pair of vertices has an edge. Now, the probability the weight of a particular edge might be very small but still there's an edge between every pair of vertices. So, in our terminology from Markov chains this corresponds to an Ergodic Markov chain which implies that there is a unique stationary distribution π. And regardless of where you start the random walk you always reach this stationary distribution in the limit over time.

So the definition of this π is independent of the starting state of the random walk. So this π is well defined. So we can set this π to a stationary probability of webpage x - to be the page rank, the importance of webpage x.

***

(Slide 26) Sink Nodes



Sink nodes

Suppose page x has no outgoing links:

with prob. α follow random link

1. self-loop
2. remove sinks (recursively)
3. set $\alpha = 0$ for sinks

Let's consider the random surfer model we just defined.

There's a problem in the current definition. In particular, suppose that a page x or webpage x has no outgoing links. In a random surfer model with probability 1- α, we go to a random webpage from the entire graph. And with probability α, we follow a random link from this current page x. Now what happens if this page x is a sink node and has no outgoing links? What do we do in this case with probability α?

Currently the model is not well-defined because of this case. And, there are several alternatives that we can consider which will make it well-defined. The simplest option is just to self-loop.

What exactly do we mean? We mean that if there's no outgoing links, if this page acts as a sink node, then with probability α, we just stay at the page x. We have a link from x to itself. The downside of this approach is that we're adding an incoming link into x. So this is going to artificially boost the page rank of page x.

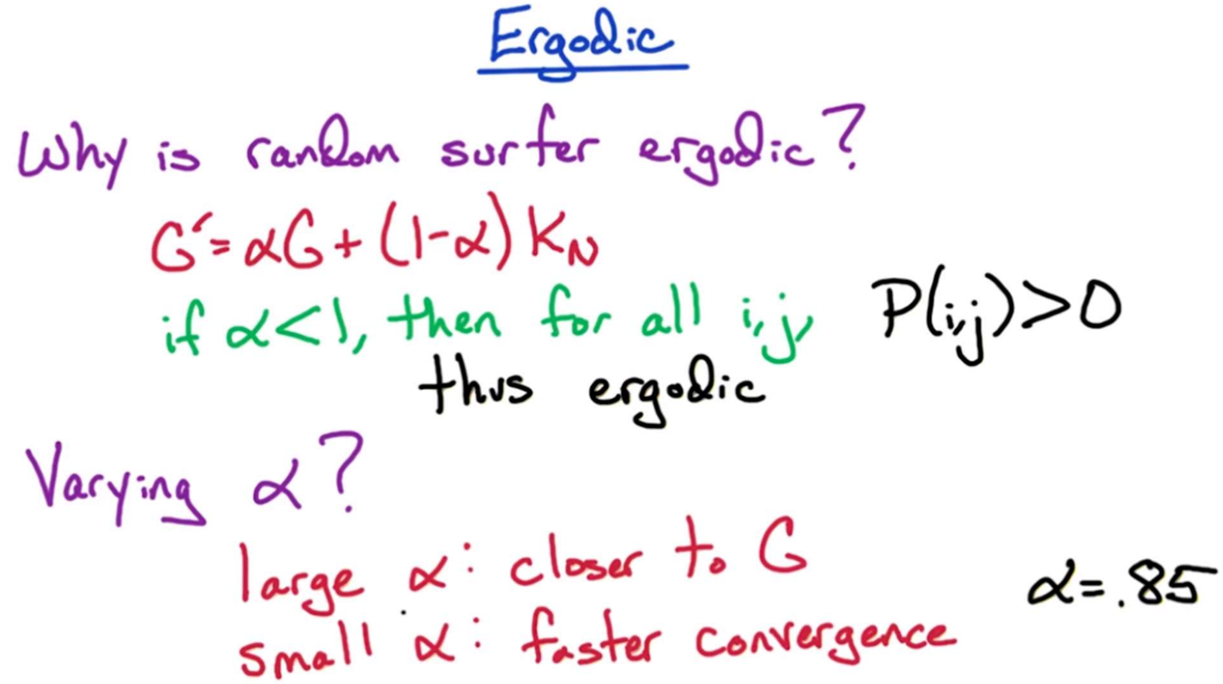
Another option is to simply remove these sink nodes. Now, once we remove some of these sink nodes, then there will be new sink nodes that might be created. So we have to recursively apply this approach. We have to keep removing sink nodes recursively until there are no sink nodes remaining in the graph. The problem with this approach is that we shrink the graph - there may be some nodes which will not get a page rank.

One more natural approach I want to consider is that, in this case of a sink node x, with probability α, we'll go to a random webpage chosen uniformly at random from the entire graph.

In other words, we're going to set $\alpha = 0$ for just these sink nodes. So, for a sink node with probability one, we'll choose a random webpage from the entire graph and go to that random webpage. This is a quite natural approach and this is apparently what PageRank actually does according to Wikipedia.

***

(Slide 27) Ergodic



Ergodic

Why is random surfer ergodic?

$G' = \alpha G + (1-\alpha) K_N$

if $\alpha < 1$, then for all i,j $P(i,j) > 0$

thus ergodic

Varying α?

large α: closer to G

small α: faster convergence

$\alpha = .85$

Let's look again at why the random surfer model is Ergodic.

We have our original webgraph, G. Now, what happens in the random surfer model? Well, with probably α, we follow a random outgoing link in G. And with probably 1 - α, we go to a random webpage in the entire graph. This corresponds to adding in the complete graph, where N is the number of vertices in this original graph. So, this defines our new graph, G′.

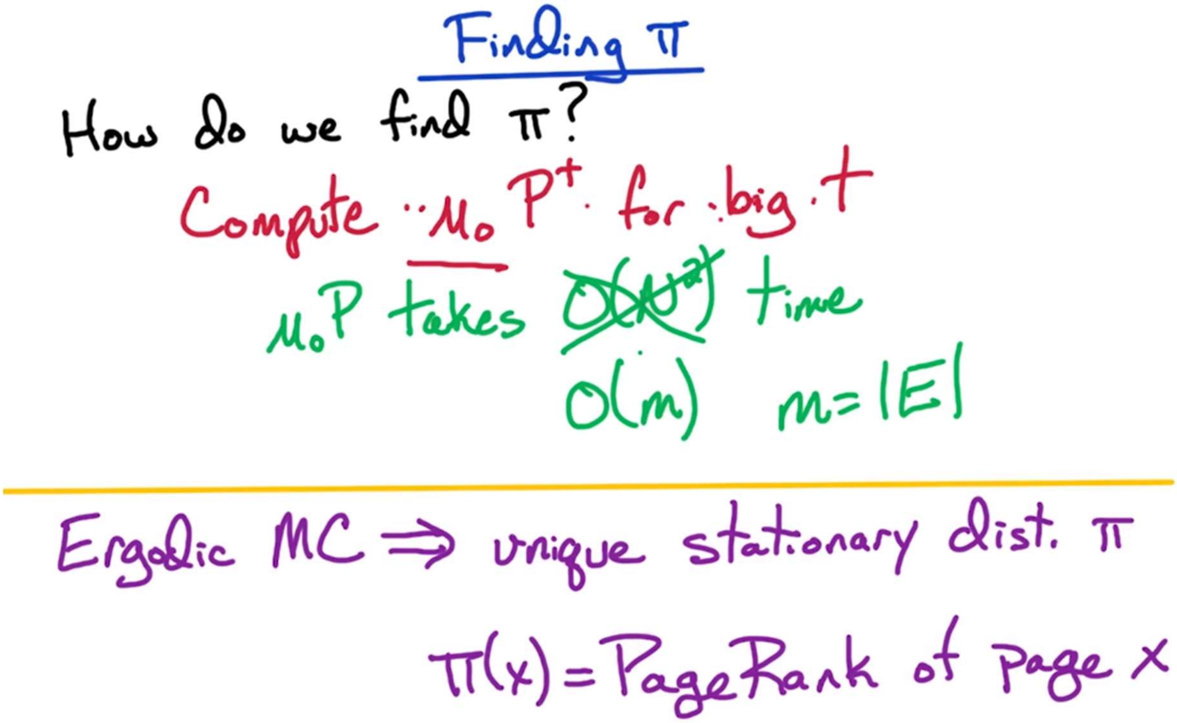
Now, suppose that α is strictly less than one. That means, with some positive probability, we use these edges from the complete graph. So, if we consider any pair of states, i and j. Let's look at the probability of going from state i to state j in one step. Well, if α is strictly less than one, then there's some chance, some probability, of going from state i to state j, using this last type of transition. That means, in the transition matrix, the entry P(i j) is positive and all the entries in this matrix are positive. There's no zero entries in this matrix. It's a fully connected transition matrix. Therefore, if $\alpha < 1$, then this random surfer model is Ergodic.

Now, what happens if α equals one? – then, we're not using this random button here. So, we're just using the original graph and there's no reason why the original graph is going to be Ergodic. The original graph that we're interested in is this graph, G. Instead, we're looking at this graph, G′. For the case $\alpha < 1$, we need this condition that α is less than one in order for it to be Ergodic. But how does this new graph, G′, compare to this original graph, G? In particular, how does the principle eigenvector for this graph G′, this Page rank vector for G′, compare to the properties of the original graph G? While this is a somewhat vague, very triggering, but a very vague question. I don't know how to address it. But what we can look at is “what is the effect of varying α”? How does the Page rank vector change as we vary α? Well, if α is large, if it's close to 1, then this graph G′ is close to the original graph, G. So we hope that the properties of G′ are close to the properties of G. As α gets smaller, then this complete graph is becoming bigger and we're becoming further away from the original graph, G. But there's a trade-off. As α decreases, our convergence rate to the page rank vector to the principle eigenvector is going to go faster. We're going to converge faster to this principle eigenvector, because of this complete graph - As this becomes bigger, it will mix faster.

Now, according to Wikipedia, Google uses α as 0.85. This presents a reasonable trade-off between these two scenarios. But an interesting question is “how does the Page rank vector change as we vary α”? But if we look at α big like 0.99 or 0.95, compared to α being 0.85 or 0.75, how does a page rank vector change? For example, if you look at the top sites, those sites with the largest Page rank vectors, how does a set of top sites change with α? And does their ordering change with α? So, if you implement the PageRank algorithm and you take a large dataset, then you can look at what is the effect of varying α on the ordering of the sites, according to page rank.

***

(Slide 28) Finding Pi



Finding π

How do we find π?

Compute $\mu_0 P^t$ for big t

$\mu_0 P$ takes $\cancel{O(N^2)}$ time

$O(m)$ $m = |E|$

Ergodic MC ⇒ unique stationary dist. π

$\pi(x)$ = PageRank of Page x

Now, how do I find this vector π? How do I find the stationary distribution of this Markov chain? If I want to use π as the measure of the importance of a webpage, I have to find π, or find something which is a close approximation to it.

Now, to find pi, what do I do? Well, I start at some initial distribution. I can take any initial distribution. Let's define that initial distribution by a row vector μ0 and then I'm going to run the Markov chain, the random walk for P^t for Big t. Computationally, that means I take μ0 multiply it by P^t.

Now how big of a t do we need to use? It turns out for the random walk that we're considering - just a very small t suffices. Why is that the case? - well, for this particular Markov chain we have with probability 1 - α, we choose a random webpage. Those links corresponding to the random button allow the Markov chain to mix rapidly.

In practice, to check whether you did a big enough t what do you do? Well, you empirically check whether this thing seemed to converge or not. Now, what do you use for this initial vector? Well, if you're running this on all the webpages, well that's a humongous quantity. So you want to use a reasonable approximation to the real π as your starting state.

Well, if you're updating the webgraph every week let's say, then I would use last week's π as my initial distribution and then I would run the random walk for a small number of steps hopefully and then that would give me my approximation for the new π so I use my last week's π as my initial distribution.

Now one important thing to consider, this matrix is huge. N, the number of webpages, is humongous. So $O(N^2)$ is too big. So how long does it take me to compute this vector times this matrix? Well, the naive way it's going to take me $O(N^2)$ time - that's too long. For the N's we're considering, there's no way you can run for $O(N^2)$ time, you won't even have $O(N^2)$ space.

You need to do it in O(m) - m is the number of edges in the original webgraph. Now of course some webpages might have many hyperlinks out of it but typically there's going to be a constant number of hyperlinks out of each webpage. So m is probably going to be O(N) and if you think about the transition matrix that we use for PageRank, then one can implement this in O(m) time with just a little bit of thought and this will typically be O(N) as opposed to $O(N^2)$.

So linear time is more reasonable - that's something we can implement for large N.

Well that completes the description of the PageRank algorithm.