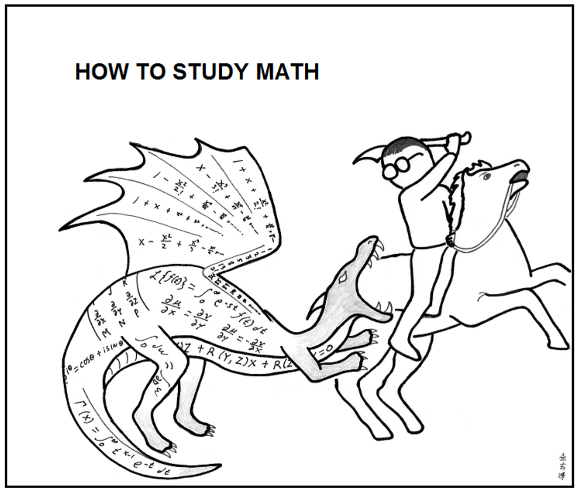
Notes: Pagerank III (the last one!)



Don't just read it; fight it!

--- Paul R. Halmos

1 Alternative Motivations

Pagerank can be used in many situations besides web ranking. In particular, if you can formalize your problem as somehow finding the "most important" nodes in a graph where something important is "flowing" between those nodes, then you can use pagerank.

Problem 1. Describe how pagerank can be used for the following tasks.

1. You are in charge of advertising the new Apple iGadget. You want to deliver free samples to Twitter influencers, with the hope that they will write good reviews that cause many people to purchase the iGadget. How do you select which people to give these free samples to?

2. You are the director of the CIA and you want to kill terrorists. How do you identify likely terrorists?

NOTE: In 2014, former CIA and NSA director Michael Hayden famously said "We kill people based on metadata." This is exactly what he was referring to. See:

`https://www.justsecurity.org/10318/video-clip-director-nsa-cia-we-kill-people-based-metadata/`

Problem 2. Create an example problem statement similar to the problem statements from Problem 1 above, and describe how the problem can be solved using pagerank.

2 A Faster Algorithm?

Problem 3. There are many algorithms for computing top eigenvectors, and any of these algorithms can be used to compute the pagerank vector. In this problem, we will investigate the *exponentially accelerated power method* (EAPM). The EAPM a divide and conquer algorithm that can achieve the same accuracy ϵ as the power method with only a logarithmic number of iterations.

The EAPM does not have a standard name, and so it is not something you can easily google to learn more about. Furthermore, it is not covered in the reference paper. Therefore, you will have to solve all of the problems below on your own without aide from outside sources. The surface-level purpose of this problem is to get you to understand the EAPM trick, but the deeper purpose of this problem is to get you comfortable analyzing algorithms for which there is no reference analysis available anywhere in the world.

Recall that in the standard power method, at each iteration we perform the computation in Eq (5.1) from the text. It is reproduced below:

$$\mathbf{x}^{(K)T} = \mathbf{x}^{(K-1)T}\bar{\bar{\mathbf{P}}}. \tag{1}$$

Unraveling the recursion, we have that

$$\mathbf{x}^{(K)T} = \mathbf{x}^{(0)T}\bar{\bar{\mathbf{P}}}^K. \tag{2}$$

In the standard power method, we solve Eq (2) with a procedure involving K matrix-vector operations. The key idea of the EAPM is to "reparenthesize" these operations to instead solve $\log K$ matrix-matrix operations. We will see that in some cases this reparenthesization results in much (i.e. exponentially) faster runtimes.

The output of the EAPM after K iterations is the vector $\mathbf{y}^{(K)}$, which is defined as

$$\mathbf{y}^{(K)} = \frac{\mathbf{x}^{(0)}\mathbf{Q}_K}{\|\mathbf{x}^{(0)}\mathbf{Q}_K\|_2}, \tag{3}$$

where

$$\mathbf{Q}_K = \begin{cases} \bar{\bar{\mathbf{P}}} & \text{if } K = 0 \\ \mathbf{Q}_{K-1}\mathbf{Q}_{K-1} & \text{otherwise} \end{cases}. \tag{4}$$

In the standard power method, the matrix $\bar{\bar{\mathbf{P}}}$ is not stored explicitly; we substitute the definition of $\bar{\bar{\mathbf{P}}}$ into Eq (1) above and perform all calculations on the sparse $\mathbf{P}$ matrix. In the EAPM, the $\bar{\bar{\mathbf{P}}}$ matrix is stored explicitly as a dense matrix, and each $\mathbf{Q}_K$ is also stored as a dense matrix.

1. What is the runtime of calculating $\mathbf{Q}_K$ given $\mathbf{Q}_{K-1}$?

2. What is the runtime of computing $\mathbf{y}^{(K)}$ in terms of K?

3. As with the standard power method, we do not know the total number of iterations of the exponential power method in advance. Instead, we iterate until the answer is "good enough" as measured by the residual. That is, we iterate until

$$\|\mathbf{y}^{(K)} - \mathbf{y}^{(K-1)}\|_2 \leq \epsilon, \tag{5}$$

where ϵ is a predetermined small constant value. The *Deeper Inside Pagerank* paper does not discuss the EAPM algorithm, so I have provided below a simple theorem (without proof) about its convergence rate.

Theorem 1. For all K, the EAPM satisfies

$$\|\mathbf{y}^{(K)} - \mathbf{y}^{(K-1)}\|_2 \leq \alpha^{2^K}. \tag{6}$$

Use Theorem 1 above to compute the number of iterations K required to achieve a target ϵ value.

4. What is the runtime of computing $\mathbf{y}^{(K)}$ in terms of ϵ?

5. Under what conditions is the exponentially accelerated power method faster or slower than the standard power method?

6. Why does it not make sense to use the exponentially accelerated power method to compute the pagerank vector? In particular, what bad thing(s) would happen if $\mathbf{P}$ was stored as a sparse matrix, and we substitute the defintion of $\bar{\bar{\mathbf{P}}}$ in terms of $\mathbf{P}$ into Eq 4 above?