Extrapolating Method for Accelerating PageRank Computation

Arturo Hernandez, Dennise Lopez
Department of Mathematics
California State University, Long Beach

Introduction:

The original PageRank algorithm used the Power method to compute successive iterates that converge to the principal eigenvector of Markov Matrix representing the web link graph. The algorithm, presented here, called **Quadratic Extrapolation** accelerates the convergence of Power method by periodically subtracting off estimates of the nonprincipal eigenvectors from the current iterate of the Power Method.

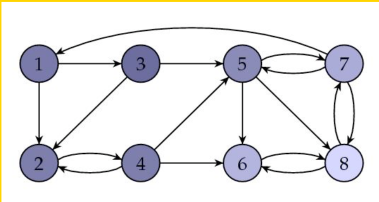

Figure 1: Shown is a representation of a small collection (eight) of web pages with links represented b y arrows. This shows that page 8 has the highest PageRank.

Figure 2 The corresponding Markov Matrix to web link graph in Figure 1. (Markov Matrix is a matrix whose sum of all row(or columns) elements equal 1.

$$H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 0 \\ 1/2 & 0 & 1/2 & 1/3 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 1 & 1/3 & 0 \end{bmatrix}$$

Methods:

Both methods computes the principal eigenvector of A corresponding to the dominant eigenvalue, λ=1.

Power Method:

The Power iteration starts with a vector, which may be an approximation to the dominant eigenvector or a random vector. The method is described by the iteration. So at every iteration, the vector is multiplied by the matrix A and normalized.

function $\vec{x}^{(n)} = \mathit{PowerMethod()}$ {
$\vec{x}^{(0)} = \vec{v}$;
$k = 1$;
repeat
$\quad \vec{x}^{(k)} = A\vec{x}^{(k-1)}$;
$\quad \delta = ||x^{(k)} - x^{(k-1)}||_1$;
$\quad k = k + 1$;
until $\delta < \epsilon$;
}

Algorithm 1: Power Method Algorithm

Methods (cont):

Quadratic Extrapolation:

The Quadratic Extrapolation iteration starts with a vector, which is a linear combination of 3 eigenvectors (could be randomly chosen). So at every iteration, it periodically subtracts off estimates of the nonprincipal eigenvector from the current iterate of the Power Method.

function $\vec{x}^* = \mathit{QuadraticExtrapolation}(\vec{x}^{(k-3)}, \ldots, \vec{x}^{(k)})$ {
for $j = k-2 : k$ **do**
$\quad \vec{y}^{(j)} = \vec{x}^{(j)} - \vec{x}^{(k-3)}$;
end
$Y = \begin{pmatrix} \vec{y}^{(k-2)} & \vec{y}^{(k-1)} \end{pmatrix}$;
$\gamma_3 = 1$;
$\begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = -Y^+ \vec{y}^{(k)}$;
$\gamma_0 = -(\gamma_1 + \gamma_2 + \gamma_3)$;
$\beta_0 = \gamma_1 + \gamma_2 + \gamma_3$;
$\beta_1 = \gamma_2 + \gamma_3$;
$\beta_2 = \gamma_3$;
$\vec{x}^* = \beta_0 \vec{x}^{(k-2)} + \beta_1 \vec{x}^{(k-1)} + \beta_2 \vec{x}^{(k)}$;
}

Algorithm 2: Quadratic Extrapolation Algorithm

Development of the Quadratic Extrapolation Algorithm

$$\vec{x}^{(k-3)} = \vec{u}_1 + \alpha_2 \vec{u}_2 + \alpha_3 \vec{u}_3$$

We then define the successive iterates

$$\begin{aligned} \vec{x}^{(k-2)} &= A\vec{x}^{(k-3)} \\ \vec{x}^{(k-1)} &= A\vec{x}^{(k-2)} \\ \vec{x}^{(k)} &= A\vec{x}^{(k-1)} \end{aligned}$$

Since we assume A has 3 eigenvectors, the characteristic polynomial $p_A(\lambda)$ is given by:

$$p_A(\lambda) = \gamma_0 + \gamma_1 \lambda + \gamma_2 \lambda^2 + \gamma_3 \lambda^3$$

A is a Markov matrix, so we know that the first eigenvalue $\lambda_1 = 1$. The eigenvalues of A are also the zeros of the characteristic polynomial $p_A(\lambda)$. Therefore,

$$p_A(1) = 0 \Rightarrow \gamma_0 + \gamma_1 + \gamma_2 + \gamma_3 = 0$$

We may rewrite this as,

$$(\vec{x}^{(k-2)} - \vec{x}^{(k-3)})\gamma_1 + (\vec{x}^{(k-1)} - \vec{x}^{(k-3)})\gamma_2 + (\vec{x}^{(k)} - \vec{x}^{(k-3)})\gamma_3 = 0$$

Let us make the following definitions:

$$\begin{aligned} \vec{y}^{(k-2)} &= \vec{x}^{(k-2)} - \vec{x}^{(k-3)} \\ \vec{y}^{(k-1)} &= \vec{x}^{(k-1)} - \vec{x}^{(k-3)} \\ \vec{y}^{(k)} &= \vec{x}^{(k)} - \vec{x}^{(k-3)} \end{aligned}$$

We can now write equation in matrix notation:

$$\begin{pmatrix} \vec{y}^{(k-2)} & \vec{y}^{(k-1)} & \vec{y}^{(k)} \end{pmatrix} \vec{\gamma} = 0$$

We now wish to solve for $\vec{\gamma}$. Since we're not interested in the trivial solution $\vec{\gamma} = 0$, we constrain the leading term of the characteristic polynomial γ_3:

$$\gamma_3 = 1$$

We may do this because constraining a single coefficient of the polynomial does not affect the zeros.[3]

$$\begin{pmatrix} \vec{y}^{(k-2)} & \vec{y}^{(k-1)} \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = -\vec{y}^{(k)}$$

Results:

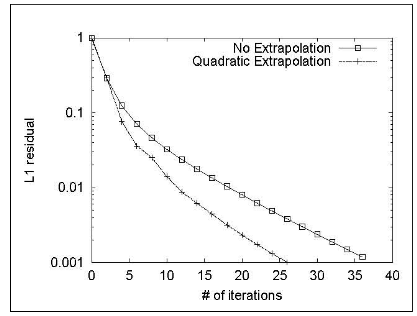

Figure 3: Comparison of convergence rates for Power Method and Quadratic Extrapolation. Note the Quadratic Extrapolation convergences faster than the Power Method.

Summary:

•Quadratic Extrapolation is a simple technique that requires little addition infrastructure to integrate into the standard power method.

•Additionally, the extrapolation step need only be applied periodically to enhance the convergence of the PageRank.

Conclusions:

For Web graphs containing a billion nodes, computing a PageRank vector can take several days. Computing PageRank quickly is necessary to reduce the lag time from when a new crawl is completed to when that crawl can be made available for searching.

Acknowledgements:

•Kamvar, Sepandar D. (2003) *Extrapolation Methods for Accelerating PageRank Computation.*

•Golub, Gene H. (2003) *Numerical Methods for Rapid Computation of PageRank.*

•Austin, David. (2006) *How Google Finds your Needle in the Web's Haystack.*

•Dr. Jen-Mei Chang