A Fast Algorithm for Matrix Balancing

Philip A. Knight* Daniel Ruiz†

10th October 2007

*University of Strathclyde
†ENSEEIHT

Outline

- Matrix balancing and applications
- The Sinkhorn-Knopp algorithm
- Newton's method
- Implementation
- Results

Matrix Balancing

- Given $A \in \mathbb{R}^{n \times n}$, $A \geq \mathbf{0}$ find diagonal matrices D_1 and D_2 so that $P = D_1 A D_2$ is doubly stochastic.

Applications

- Interpreting economic data (Bacharach, 1970)
- Preconditioning sparse matrices (Livne and Golub, 2004).
- Understanding traffic circulation (Lamond and Stewart, 1981).
- Ordering nodes in a graph (K, 2006).

Gravity Model

$$F = \frac{Gm_1 m_2}{d^2}.$$

- Flow in a network is proportional to a repulsion factor from the source and attraction of the destination.
- Suppose population area i has attractiveness d_i and repulsiveness f_i.
- The number of visits a_{ij} is proportional to $d_i f_j p_{ij}$.
- Measure A.
- We look for d, f, P where $A = \mathbf{diag}(d) P \mathbf{diag}(f)$.

Web Ranking

- Let A be matrix of graph of web links.
- Find diagonal matrices D_1 and D_2 so that $P = D_1 A D_2$ is doubly stochastic.
- Clearly, stationary distribution of P tells us nothing.
- Let $r = \mathbf{diag}(D_1)$, $c = \mathbf{diag}(D_2)$.
- Claim: authoritativeness of node i is inversely proportional to c_i, "hubness" is inversely proportional to r_i.
- As with PageRank, we add a small quantity, p, to every entry in the matrix.

Example: Wikipedia Ratings

United States	2000	2000	Political parties
Race (US Census)	Pop. density	Marriage	Environment topics
United Kingdom	km^2	US	State leaders
France	Census	2003, 2004, 2005	Airlines
2005,2004,2000	Square mile	UK/England	2 letter combinations
Canada	Marriage	Canada	Masts
England	Per capita income	Japan	Mathematicians
Cat. by country	US Census	Australia	Peerage of the UK
2003	Poverty line	2001, 2002	Record labels
Cat.:Culture	Race (US Census)	Germany	Biblical names

Existence and Uniqueness of Solutions

- Assume that $A \in \mathbb{R}^{n \times n}$, $A \geq \mathbf{0}$ and $p = \mathbf{1}$.
- DAF doubly stochastic, $\mathbf{diag}(D), \mathbf{diag}(F) > \mathbf{0}$.
- Solution exists if A contains sufficient nonzero entries.
- A fully indecomposable.
- Solution is unique (up to scaling).
- Sinkhorn, 1964; Brualdi et al., 1966; Marshall and Olkin, 1967.

The Sinkhorn-Knopp algorithm

- Let $\mathcal{D}(x) = \mathbf{diag}(x)$.
- SK generates a sequence of matrices whose rows and columns are normalised alternately.
- We want to satisfy $DAFe = e$ and $FA^TDe = e$.
- $D = \mathcal{D}(r)$, $F = \mathcal{D}(c)$.
- A solution satisfies $Ac = 1./r$, $A^Tr = 1./c$.
- Suggests the following algorithm.

$$c_{k+1} = 1./(A^T r_k), \quad r_{k+1} = 1./(Ac_{k+1}).$$

Convergence Results

- The SK algorithm converges if A has total support (SK67, Brualdi et al. 1966) and rate is linear (Soules 1991).

- If $A > \mathbf{0}$, rate of convergence is bounded above by $\kappa(A)^2$ where

$$\kappa(A) = \frac{\theta(A)^{1/2} - 1}{\theta(A)^{1/2} + 1}, \ \theta(A) = \max_{i,j,k,l} \frac{a_{ik} a_{jl}}{a_{jk} a_{il}}$$

  (Franklin and Lorenz 1989).

- If A is fully indecomposable and SK converges to P, rate of convergence is $\sigma_2(P)^2$ (K, 2006).

Alternative Algorithms

- Iterate with a view to reducing standard deviation between row sums iterates (Parlett and Landis, 1982).
- Similar approach by Linial et al. (2000). Upper bound on iteration counts $O(n^7)$.
- View matrix balancing as an optimisation problem (Marshall and Olkhin, 1968; Schneider, 1990; Balakrishnan et al., 2004).
- Newton's method (Khachiyan and Kalantari, 1992; Livne and Golub, 2004).

Newton's method

- Suppose A is symmetric.
- Looking to solve $f(x) = \mathcal{D}(x)Ax - e = \mathbf{0}$.
- $\dfrac{\partial f}{\partial x} = \mathcal{D}(x)A + \mathcal{D}(Ax)$.

$$x_{k+1} = x_k - (\mathcal{D}(x_k)A + \mathcal{D}(Ax_k))^{-1}(\mathcal{D}(x_k)Ax_k - e).$$

- Rearrange:
  $(A + \mathcal{D}((Ax_k)/(x_k)))x_{k+1} = Ax_k + 1/x_k$.
- If A is nonsymmetric we can work with
  $$\begin{bmatrix} \mathbf{0} & A \\ A^T & \mathbf{0} \end{bmatrix}.$$

Convergence

- $A \geq 0$, $y > 0$, $D = \mathcal{D}(Ay)\mathcal{D}(y)^{-1}$.
- If A has support, $A + D$ is S(S)PD.
- Proof: Note that $A + D$ is similar to the diagonally dominant matrix $B + \mathcal{D}(Be)$ where $B = \mathcal{D}(y)^{-1}A\mathcal{D}(y)$.
- If A is fully indecomposable, A is SPD.
- If A has support, the system

$$(A + \mathcal{D}(Ay)\mathcal{D}(y)^{-1})z = Ay + \mathcal{D}(y)^{-1}e$$

is consistent.

Splitting Methods

- $M - N = A + \mathcal{D}(x_k)^{-1}\mathcal{D}(Ax_k)$.
- $M = \mathcal{D}(x_k)^{-1}\mathcal{D}(Ax_k)$, $N = -A$ initial guess x_k.
- $x_{k+1} = \mathcal{D}(Ax_k)^{-1}e$.
- $x_{k+1} = x_k + M^{-1}(\mathcal{D}(x_k)^{-1}e - Ax_k)$.
- Suppose $DAD = P$ is doubly stochastic.
- Asymptotic rate of convergence given by spectral radius of splitting matrices of $P + I$.
- If A is symmetric, significant improvement over SK is possible with an appropriate choice of M.

Conjugate Gradient Method

- Solve $(A + \mathcal{D}((Ax_k)/(x_k)))x_{k+1} = Ax_k + 1/x_k$.
- PCG: Let $D_k = \mathcal{D}(x_k)$.

$$D_k(A + \mathcal{D}((Ax_k)/(x_k)))D_k D_k^{-1} x_{k+1} = D_k A x_k + e.$$

- Let $A_k = D_k A D_k$.
- Solve $(A_k + \mathcal{D}(A_k e))(x_{k+1}/x_k) = A_k e + e$.
- System is S(S)PD and diagonally dominant.
- Easily adaptable to nonsymmetric case.

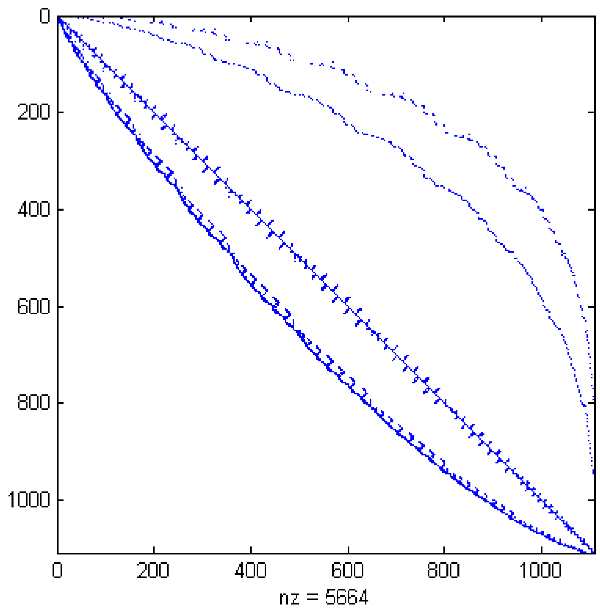
Implementation Details

- Principal cost is

$$(A_k + \mathcal{D}(A_k e))p = x_k \circ (A(x_k \circ p)) + v_k \circ p.$$

- Stopping criterion: $\|e - x_k \circ A x_k\|_2$.
- We need to avoid $-$ve components in iterates.
- Apply an Armijo-type rule inside the inner iteration.
- How much closer to the edge of the positive cone are we willing to move?
- All coordinate directions are treated equally.
- Theoretical problem: in nonsymmetric case, $A_k + \mathcal{D}(A_k e)$ is singular.

Results



H_n upper Hessenberg, $\mathbf{100}$ on diagonal, ones and zeros everywhere else.

R random sparse symmetric $\mathbf{10000} \times \mathbf{10000}$ matrix. $\mathbf{60000}$ nonzeros.

Results

	SK	CG	Error
A	2400	330	10^{-6}
A	4550	490	10^{-11}
H_{10}	3070	120	10^{-6}
H_{50}	61500	660	10^{-6}
R	5200	50	10^{-6}

Results