CS4220 PS6

Due: March 27, 2015

1. (Ascher and Greif, Section 8.4 Problem 7) A column-stochastic matrix P is a matrix whose entries are nonnegative and whose column sums are all equal to 1. In practice such matrices are often large and sparse.

Let E be a matrix of the same size as P, say, $n \times n$, all of whose entries are equal to $1/n$, and let α be a scalar, $0 < \alpha < 1$.

(a) Show that $A(\alpha) = \alpha P + (1 - \alpha)E$ is also a column-stochastic matrix.

(b) What are the largest eigenvalue and corresponding eigenvector of $A(\alpha)$?

(c) Show that the second largest eigenvalue of $A(\alpha)$ is bounded in absolute value by α.

(d) Suppose the dominant eigenvector of $A(\alpha)$ is to be computed using the power method. This vector, if normalized so that its ℓ_1-norm is equal to 1, is called the stationary distribution vector.

i. Show how matrix-vector products with $A(\alpha)$ can be performed in an efficient manner in terms of storage. (Assume n is very large, and recall that E is dense.)

ii. Show that if the power method is applied and if $\|v_0\|_1 = 1$, then all subsequent iterates v_k also have a unit ℓ_1-norm, hence no need to normalize throughout the iteration. (v_0 can also be assumed to be non-negative.)

Solution:

(a) $A(\alpha) = \alpha P + (1 - \alpha)E$. We know $p_{ij} \geq 0$ and $\sum_i p_{ij} = 1$, and $0 < \alpha < 1$, so each element a_{ij} of matrix A can be expressed as

$$a_{ij} = \alpha p_{ij} + \frac{1 - \alpha}{n} \geq 0. \tag{1}$$

The sum over all elements of any column k of A is

$$\sum_{i=1}^{n} a_{ik} = \alpha \sum_{i=1}^{n} p_{ik} + \frac{1 - \alpha}{n}(n) = \alpha + 1 - \alpha = 1. \tag{2}$$

This is true for all columns. By Eqs.(1) and (2), we know A is also a column-stochastic matrix.

(b) • **The Largest Eigenvalue λ_1 of A:**
Since A is column-stochastic, it is obvious the following equation is true:

$$A^T \begin{pmatrix} 1 \\ . \\ . \\ . \\ 1 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n a_{i1} \\ . \\ . \\ . \\ \sum_{i=1}^n a_{in} \end{pmatrix} = \begin{pmatrix} 1 \\ . \\ . \\ . \\ 1 \end{pmatrix}. \tag{3}$$

Let $e = (1, 1, \cdots, 1)^T$, the $n \times 1$ column vector with all elements equal to 1. Eq.(3) can be expressed as

$$A^T e = e, \tag{4}$$

so this equation shows 1 and column vector e are the eigenvalue and eigenvector pair of A^T, respectively. Because A and A^T have the same eigenvalues, 1 is also an eigenvalue of A. Next, we need to use the Gershgorin Disk Theorem in the subsequent argument for the bound of the eigenvalues, so we state the Gershgorin theorem here:

Theorem. *(Gershgorin Disk Theorem) Let A be any $n \times n$ matrix and λ be any eigenvalue of A. Then for some i, $1 \le i \le n$,*

$$|\lambda - a_{ii}| \le \sum_{j=1, j\neq i}^{n} |a_{ij}|.$$

Again because A and A^T have exactly the same eigenvalues, we use the column sums when applying the Gershgorin Disk theorem: any eigenvalue λ of A $(n \times n)$ must satisfy

$$|\lambda - a_{jj}| \le \sum_{i=1, i\neq j}^{n} |a_{ij}|, \quad \text{for some } j,\ 1 \le j \le n. \tag{5}$$

Because A is column-stochastic, and $a_{ij} \ge 0$,

$$\sum_{i=1, i\neq j}^{n} |a_{ij}| = 1 - |a_{jj}| = 1 - a_{jj},$$

we have

$$|\lambda - a_{jj}| \le 1 - a_{jj},$$
$$-1 + a_{jj} \le \lambda - a_{jj} \le 1 - a_{jj},$$

so we have obtained the bounds for any eigenvalues λ of A:

$$-1 \le -1 + 2a_{ij} \le \lambda \le 1, \quad \text{or} \quad |\lambda| \le 1. \tag{6}$$

By Eqs. (3) and (6) we conclude that the largest eigenvalue of A and A^T is 1, i.e., $\lambda_1 = 1$.

- **The Eigenvector of A Associated with $\lambda_1 = 1$:**
  We have shown by Eqs.(3) and (6) that, for A^T, $\lambda_1 = 1$ and the corresponding eigenvector is $e = (1, 1, \cdots, 1)^T$. Although the largest eigenvalue for A is the same ($\lambda_1 = 1$), the eigenvector for matrix A corresponding to $\lambda_1 = 1$ is *not* the same. The eigenvector v of A corresponding to the largest eigenvalue $\lambda_1 = 1$ is the solution to the following linear system:

  $$(A - I)v = 0. \tag{7}$$

  Numerically, we can use the inverse iteration method to find v by specifying a shift very close to 1 (e.g., $\sigma = 0.9999$), then apply the power method to $(A - \sigma I)^{-1}$, instead of A:

  $\sigma = 0.9999$
  repeat
  $v = (A - \sigma I)^{-1} v$
  $v = v / \|v\|_2$
  until convergence

(c) Show that the second largest eigenvalue of $A(\alpha)$ is bounded in absolute value by α.

Solution: We need the following lemma for a basic property of the eigenvectors of the transposed matrix:

Lemma. *If x_i is an eigenvector of A corresponding to the eigenvalue λ_i, and y_j is an eigenvector of A^T, then $x_i^T y_j = 0$, if $\lambda_i \neq \lambda_j$.*

Proof. The eigen-equations for A and A^T are

$$Ax_i = \lambda_i x_i, \tag{8}$$
$$A^T y_j = \lambda_j y_j. \tag{9}$$

We transpose Eq.(8) to get

$$x_i^T A^T = \lambda_i x_i^T. \tag{10}$$

Left multiplying Eq.(9) by x_i^T, and right multiplying Eq.(10) by y_j, then subtracting these two equations, now we get

$$x_i^T A^T y_j - x_i^T A^T y_j = 0 = (\lambda_i - \lambda_j) x_i^T y_j,$$

so we have the proof. □

From Eq.(3) we know e is the eigenvector of A^T corresponding to $\lambda_1 = 1$. By the lemma we have just proved we know the eigenvector v_2 corresponding to the second largest eigenvalue λ_2 of A is orthogonal to eigenvector e of A^T corresponding to λ_1:

$$e^T v_2 = 0, \quad \text{for } \lambda_1 \neq \lambda_2. \tag{11}$$

It can easily shown that $E = (1/n)ee^T$, so it leads to the following:

$$Ev_2 = \left(\frac{1}{n}ee^T\right) v_2 = \frac{1}{n}e(e^T v_2) = 0. \tag{12}$$

Consequently, substituting $Ev_2 = 0$ into the eigenequation for $Av_2 = \lambda_2 v_2$ we get

$$Av_2 = [\alpha P + (1-\alpha)E]\, v_2 = \alpha P v_2 + (1-\alpha)Ev_2 = \alpha P v_2 = \lambda_2 v_2, \tag{13}$$

therefore

$$Pv_2 = \left(\frac{\lambda_2}{\alpha}\right) v_2. \tag{14}$$

This is an eigen-equation for matrix P: v_2 is an eigenvector of P corresponding to the eigenvalue of (λ_2/α). Since P is a column-stochastic matrix, by the Gershgorin Disk theorem, we have shown earlier in part (b) that the absolute eigenvalue μ of P is bounded by 1 (as we have done in Eq.(6) for A):

$$|\mu| \leq 1. \tag{15}$$

so we get

$$\left|\frac{\lambda_2}{\alpha}\right| \leq 1,$$

and because $0 < \alpha < 1$, we have the the proof that

$$|\lambda_2| \leq \alpha.$$

(d) Suppose the dominant eigenvector of $A(\alpha)$ is to be computed using the power method. This vector, if normalized so that its ℓ_1-norm is equal to 1, is called the stationary distribution vector.

i. The matrix-vector product with $A(\alpha)$ can be performed in an efficient manner in terms of storage. (Assume n is very large, and E is dense.)

Answer: For $A(\alpha) = \alpha P + (1-\alpha)E$, when n is large (say, in the order of billions), the storage of a dense matrix E is a problem. But we can show that we do not have to store E when performing matrix vector products with $A(\alpha)$.

Recall that E is $n \times n$ with all entries equal to $1/n$, so the rank-1 matrix E can be written as

$$E = \frac{1}{n} ee^T, \quad \text{and} \quad A = \alpha P + \frac{1-\alpha}{n} ee^T.$$

Now for computing the matrix vector product $y = Az$, we write

$$y = Az = \alpha Pz + \frac{1-\alpha}{n} ee^T z. \tag{16}$$

In the second term on the right-hand-side of Eq.(16),

$$e^T z = \|z\|_1,$$

so

$$y = \alpha Pz + \left(\frac{1-\alpha}{n} \|z\|_1 \right) e.$$

In other words, in computing $y = Az$, the only matrix-vector product involved is Pz (P is usually sparse). There is *no* need to form or store the dense $n \times n$ matrix E at all (hence it is a big saving when n is huge). Therefore in performing matrix-vector product with A, it can be done very efficiently in terms of storage.

ii. Show that if the power method is applied and if $\|v_0\|_1 = 1$, then all subsequent iterates v_k also have a unit ℓ_1-norm, hence no need to normalize throughout the iteration.

Answer: If the power method is used, we need to compute

$$v^{(k+1)} = Av^{(k)} = A(Av^{(k-1)}) = \ldots = A^k v_0. \tag{17}$$

Assume v_0 is non-negative, and we have $\|v_0\|_1 = 1$, and note that the ℓ_1 norm of v_0 can be calculated as follows:

$$\|v_0\|_1 = \sum_i |(v_0)_i| = \sum_i (v_0)_i = e^T v_0,$$

The next iterate $v^{(1)}$ according to the power method is

$$v^{(1)} = Av_0 = \left[\alpha P + (1-\alpha)E\right] v_0,$$

and its ℓ_1-norm is

$$\|v^{(1)}\|_1 = e^T (Av_0) = e^T \left[\alpha P + (1-\alpha)E\right] v_0. \tag{18}$$

Because P is a column-stochastic matrix, we have $e^T P = e^T$. In addition,

$$E = \frac{1}{n} ee^T,$$

then we have

$$e^T E = \frac{1}{n}(e^T e)e^T = \frac{1}{n} n e^T = e^T. \tag{19}$$

Finally, now we can evaluate Eq.(18) as follows:

$$\|v^{(1)}\|_1 = \left[\alpha e^T + (1-\alpha)e^T\right] v_0 = e^T v_0 = \|v_0\|_1 = 1.$$

By following almost the same argument, we can show it is true that all subsequent iterates have a unit ℓ_1-norm, i.e., we do not have to normalize iterates $v^{(k)}$ in the power method for finding the eigenvector of A.

2: Simply SVD Consider the iteration

```
for k=1:kmax
  u = A*v; s = norm(u); u = u/s;
  v = A'*u; s = norm(v); v = v/s;
end
```

Argue that `u`, `v`, and `s` correspond to the first left and right singular vectors u_1 and v_1 and the dominant singular value σ_1, assuming $\sigma_1 > \sigma_2$. What is the rate of convergence?

Solution: Let A be an $m \times n$ matrix with rank r. The SVD of A is

$$A = U\Sigma V^T. \tag{20}$$

Eq.(20) can also be expressed as

$$A = \sum_{i=1}^{r} \sigma_i \, u_i \, v_i^T, \tag{21}$$

where u_i is the i-th column vector (left singular vector) of U, v_i the i-th column vector (right singular vector) of V and σ_i the i-th singular value of A, respectively. Taking the transpose of Eq.(21), we have

$$A^T = \sum_{i=1}^{r} \sigma_i \, v_i \, u_i^T, \tag{22}$$

Note that $\sigma_j^2 \equiv \lambda_j$, together with vector v_j, are the "eigenpair" of the symmetric matrix A^TA. Hence

$$A^TAv_i = \sigma_i^2 v_i,$$

and the ℓ_2-norm of Av_i is the corresponding singular value:

$$\|Av_i\|_2 = \sqrt{(Av_i)^T(Av_i)} = \sqrt{v_i^T A^T A v_i} = \sigma_i. \tag{23}$$

Right multiply Eq.(21) by v_i, and keep in mind that V is an orthogonal matrix, hence $v_i^T v_j = 0, i \neq j$, so we get the eigen-equation for A:

$$Av_i = \sigma_i u_i, \quad \text{therefore,} \quad u_i = \frac{Av_i}{\sigma_i} = \frac{Av_i}{\|Av_i\|_2}. \tag{24}$$

Similarly, right multiply Eq.(22) by u_i, we get the eigen-equation for A^T:

$$A^Tu_i = \sigma_i v_i, \quad \text{therefore,} \quad v_i = \frac{A^Tu_i}{\sigma_i} = \frac{A^Tu_i}{\|A^Tu_i\|_2}. \tag{25}$$

Note that A^TA and AA^T have exactly the same eigenvalues and the same singular values $\sigma_1, \sigma_2, \ldots, \sigma_r$.

Now the iteration of the given code can be understood in the light of Eqs.(24) and (25),

```
1   for k=1:kmax
2     u = A*v; s = norm(u); u = u/s;
3     v = A'*u; s = norm(v); v = v/s;
4   end
```

Starting from an initialization of $v = v^{(0)}$, line 2 in the code carries out Eq.(24), while line 3 carries out Eq.(25) in a loop. Note that normalizations of the iterates are included in Eqs.(24)) and (25). Combining the two steps together, we have the following relation per iteration:

$$v^{(k)} = \frac{A^T A\, v^{(k-1)}}{\|Av^{(k-1)}\|_2 \cdot \|A^T A v^{(k-1)}\|_2}. \tag{26}$$

We also have found that each of the two steps in the SVD iteration process is very similar to the power method: lines 2 and 3 will respectively converge to "$\rightarrow$" the dominant eigenvectors (left and right) and the dominant eignevalue/singular value according to Eqs.(24), (25) and (23):

- $u \rightarrow u_1$, the first left eigenvector;
- $v \rightarrow v_1$, the first right eigenvector; and
- $s \rightarrow \sigma_1$ the largest singular value of A.

For studying the convergence rate, we can express the k-th SVD iteration in Eq.(26) as

$$v^{(k)} = \frac{1}{\sigma_1^{2k}}(A^T A)^k v^{(0)}. \tag{27}$$

Since U and V are orthogonal matrices, by direct multiplication we have

$$A^T A = \left(\sum_i \sigma_i v_i u_i^T\right)\left(\sum_j \sigma_j u_j v_j^T\right) = \sum_{i=1}^{r} \sigma_i^2 v_i v_i^T, \tag{28}$$

and we can easily show

$$(AA^T)^k = \sum_{i=1}^{r} \sigma_i^{2k} v_i v_i^T. \tag{29}$$

Let the initial vector $v^{(0)}$ be spanned in terms of the orthonormal basis $\{v_1, \ldots, v_r\}$

$$v^{(0)} = c_1 v_1 + c_2 v_2 + \ldots + c_r v_r, \tag{30}$$

we then substitute Eqs.(29), (30) into Eq.(27). Most of the terms in the product are zero because, again, $\{v_1, \ldots, v_r\}$ is an orthonormal basis, and finally we get

$$\begin{aligned} v^{(k)} &= \left[v_1 v_1^T + \left(\frac{\sigma_2}{\sigma_1}\right)^{2k} v_2 v_2^T + \ldots + \left(\frac{\sigma_r}{\sigma_1}\right)^{2k} v_r v_r^T \right] (c_1 v_1 + c_2 v_2 + \ldots + c_r v_r) \\ &= c_1 v_1 + c_2 \left(\frac{\sigma_2}{\sigma_1}\right)^{2k} v_2 + \ldots + c_r \left(\frac{\sigma_r}{\sigma_1}\right)^{2k} v_r \,. \end{aligned} \tag{31}$$

Provided $\sigma_1 > \sigma_2$, Eq.(31) reveals that the convergence rate of the SVD iteration is proportional to $(\sigma_2/\sigma_1)^{2k}$.

3: Subspace iteration Implement orthogonal iteration on a m-dimensional space (see the book, page 239). Your function should have the interface

function [V,R] = p6subspace(A, m, maxiter, rtol)

and should iterate until either `maxiter` iterations have been reached or until the approximation $V^{(k)}$ satisfies the tolerance

$$\|AV^{(k)} - V^{(k)}R^{(k)}\|_F < \text{rtol}.$$

You should start your iteration with a random orthogonal basis, which you can compute with the line

[V,R] = **qr**(**randn**(n,m), 0);

Solution: Let $A \in \mathbb{R}^{n\times n}$, starting initially with an arbitrary orthogonal matrix $(n \times m)$

$$V^{(0)} = \begin{pmatrix} v_1^{(0)} & v_2^{(0)} & \ldots & v_m^{(0)} \end{pmatrix} \in \mathbb{R}^{n\times m},$$

subspace iteration is a basic algorithm to compute m eigenvectors simultaneously by the following iteration:

$$\begin{aligned} &X^{(k)} = AV^{(k-1)}, \\ &V^{(k)}R^{(k)} = X^{(k)}, \quad \% \text{ QR decomp of } X^{(k)} \end{aligned}$$

During the iteration, we have to keep columns of the iterate $V^{(k)}$ linearly independent to prevent they all converge to the dominant eigenvector (corresponding to the largest eigenvalue). So the common approach is to run a QR decomposition of $X^{(k)}$ in the iteration before moving to the next iteration level $(k+1)$. The iteration continues until either the `maxit` iterations have been reached or until the approximation matrix $V^{(k)}$ satisfies

$$\|AV^{(k)} - V^{(k)}R^{(k)}\|_F < \text{rtol}.$$

The code is listed as follows:

```
function [V, R] = p6subspace(A, m, maxiter, rtol)
% [V, R] = p6subspace(A, m, maxiter, rtol)
% Perform m-dimensional subspace iteration for matrix A,
% returning V (eigen matrix) and the upper triangular matrix R.
%
% The iterations continue till either the tolerance (rtol) has
% been satisfied or the maxiter iterations have been reached.
%
% Convergence criterion (using the Frobenius norm):
% | AV_k - V_k R_k |_F < rtol
%
n=length(A);
% pre-allocate V and X matrices (n by m):
V = zeros(n,m);
X = zeros(n,m);
% initialization : V_0:
[V, R]=qr(randn(n,m),0);

k=1;
while (k < maxiter)
    X = A*V;
    % To get V at the next level k+1 by QR decomp
    [V, R]=qr(X,0);
    % Schur's projection step
    %Aproj=V'*(A*V);
    %[U,T]=schur(Aproj);
    %V=V*U;
    DIFF=A*V - X;
    residual=norm(DIFF,'fro');
    if rem(k,10)==0
        disp(sprintf('\nresidual=%g, k=%d', residual, k))
    end
    if ( residual < rtol )
        return
    end
    k=k+1;
end
end
```