PS 6

Due: Fri, Mar 27

Note: While it is not assigned, you may wish to look at problem 10 if you're still thinking about Project 2.

1: By the book Book section 8.4, problem 7, parts (a), (b), and (d). For part (d), you may also assume the initial vector v_0 is non-negative.

Answer:

1. $A(\alpha) = \alpha P + (1-\alpha)E$ where $E = ee^T/n$. Both P and E are independently column stochastic. A convex combination of non-negative things is non-negative, so all elements of $A(\alpha)$ are non-negative; similarly, each column sum of $A(\alpha)$ is $(1-\alpha)\cdot 1 + \alpha \cdot 1 = 1$. So $A(\alpha)$ is column stochastic.

2. The largest eigenvalue is equal to one (it is bounded by $\|A(\alpha)\|_1 = 1$). The corresponding left eigenvector is e^T, where e is the vector of all ones. The right eigenvector (normalized to unit length) is
$$x = (1-\alpha)\,(I - \alpha P)^{-1}\, e/n.$$

3. We'll do part (c) even though it wasn't assigned. Noting that e^T is the row eigenvector for the eigenvalue 1, let v be the column eigenvector for any other eigenvalue λ. This implies that $e^T v = e^T A v = \lambda e^T v$, which means $e^T v = 0$. Therefore $A(\alpha)v = \alpha P v$, so $|\lambda| \leq \|\alpha P\|_1 = \alpha$.

4. Observe that $e^T A(\alpha)^k v_0 = e^T v_0 = 1$ because e is a row eigenvector. Also, observe that $A(\alpha)$ is elementwise non-negative and v_0 is elementwise non-negative, so all the elements of $A(\alpha)^k v_0$ are non-negative. Therefore $\|A(\alpha)^k v_0\| = e^T A(\alpha)^k v_0 = 1$.

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?

Answer: For v, this is power iteration with $A^T A$. The eigenvalues of $A^T A$ are the singular values squared, and the eigenvectors are columns of V. Similarly, for u, this is power iteration with AA^T. Either way, convergence is like $(\sigma_2/\sigma_1)^2$.

2: 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);
```

Answer: Note that the residual is equivalent to $\|AV^{(k)} - AV^{(k-1)}\|_F$. Otherwise, this is a straightforward implementation task:

```
% [V,R] = p6subspace(A, m, maxiter, rtol)
%
% Run subspace iteration on an m-dimensional subspace until reaching
% either maxiter iterations or until the residual
%    A*V-V*R
% is less than rtol in the Frobenius norm.
%
function [V,R] = p6subspace(A, m, maxiter, rtol)

n = length(A);
[V,R] = qr(randn(n,m),0);
AV = A*V;
[V,R] = qr(AV,0);
for k = 2:maxiter
  AVprev = AV;
```

```
  AV = A*V;
  resid = norm(AV-AVprev, 'fro');
  if resid < rtol
    fprintf('Converged after %d steps\n', k);
    return
  end
  [V,R] = qr(AV,0);
end
fprintf('Stopped after maxiter steps, residual = %e\n', resid);
```

We test with two simple triangular matrices, one of which admits a well-behaved subspace and the other of which does not.

```
% Should converge fairly quickly
ltest = [1:8, 15, 21];
A = diag(ltest) + triu(randn(10),1);
[V,R] = p6subspace(A, 2, 1000, 1e-6);

% Never converges
A = eye(10) + triu(randn(10),1);
[V,R] = p6subspace(A, 2, 1000, 1e-6);
```