Homework 4. Due Wednesday, Sept. 25.

Additional reading for problems 2 and 3: L. N. Trefethen and D. Bau III, "Numerical Linear Algebra"

1. (**5 pts**) Find an upper bound for the condition number for eigenvector r_j of a non-symmetric matrix A assuming that all its eigenvalues are distinct. In what case will this condition number be large?

2. (**4 pts**) Let A be an $n \times n$ matrix. The Rayleigh quotient $Q(x)$ is the following function defined on all $x \in \mathbb{R}^n$:

$$Q(x) := \frac{x^\top Ax}{x^\top x}.$$

   (a) Let A be symmetric. Prove that $\nabla Q(x) = 0$ if and only if x is an eigenvector of A.

   (b) Let A be asymmetric. What are the vectors x at which $\nabla Q = 0$?

3. Consider the Rayleigh Quotient Iteration, a very efficient algorithm for finding an eigenpair of a given matrix

   `Input:` $x_0 \neq 0$ `is the initial guess for an eigenvector`
   $v = x_0/\|x_0\|$
   `for k = 0, 1, 2, ...`
      $\mu_k = v^T Av$
      `Solve` $(A - \mu_k I)w = v$ `for` w
      $v = w/\|w\|$.
   `end for`

   Here is Matlab program implementing the Rayleigh Quotient Iteration for finding an eigenpair of a random $n \times n$ symmetric matrix starting from a random initial guess:

```
function RayleighQuotient()
n = 100;
A = rand(n);
A = A' + A;
v = rand(n,1);
v = v/norm(v);
k = 1;
mu(k) = v'*A*v;
tol = 1e-12;
I = eye(n);
res = abs(norm(A*v - mu(k)*v)/mu(k));
```

```
fprintf('k = %d: lam = %d\tres = %d\n',k,mu(k),res);
while res > tol
    w = (A - mu(k)*I)\v;
    k = k + 1;
    v = w/norm(w);
    mu(k) = v'*A*v;
    res = abs(norm(A*v - mu(k)*v)/mu(k));
    fprintf('k = %d: lam = %d\tres = %d\n',k,mu(k),res);
end
end
```

(a) **(2 pts)**Let A be a symmetric matrix with all distinct eigenvalues. Let μ be not an eigenvalue of A. Show that if (λ, v) is an eigenpair of A then $((\lambda-\mu)^{-1}, v)$ is an eigenpair of $(A-\mu I)^{-1}$.

(b) **(4 pts)**The Rayleigh Quotient iteration involves solving the system $(A-\mu_k I)w = v$ for w. The matrix $(A-\mu_k I)$ is closed to singular. Nevertheless, this problem is well-conditioned (in exact arithmetic). Explain this phenomenon. Proceed as follows. Without the loss of generality assume that v is an approximation for the eigenvector v_1 of A, and μ is an approximation to the corresponding eigenvalue λ_1. Let $\|v\| = 1$. Write v as

$$v = \left(1 - \sum_{i=2}^{n} \delta_i^2\right)^{1/2} v_1 + \sum_{i=2}^{n} \delta_i v_i,$$

where δ_i, $i = 2, \ldots, n$, are small. Show that the condition number $\kappa((A-\mu I)^{-1}, v)$ (see page 88 in Bindel & Goodman) is approximately $\left(1 - \sum_{i=2}^{n} \delta_i^2\right)^{-1/2}$ which is close to 1 provided that δ_i are small.

(c) **(4 pts)** It is known that the Rayleigh Quotient iteration converges cubically, which means that the error $e_k := |\lambda - \mu_k|$ decays with k so that the limit

$$\lim_{k\to\infty} \frac{e_{k+1}}{e_k^3} = C \in (0, \infty).$$

This means, that the number of correct digits in μ_k triples with each iteration. Try to check this fact experimentally and report your findings. Proceed as follows. Run the program. Treat the final μ_k as the exact eigenvalue. Define $e_j := |\mu_j - \mu_k|$ for $j = 1, \ldots, k-1$. Etc. Pick several values of n and make several runs for each n. Note that you might not observe the cubic rate of convergence due to too few iterations and floating point arithmetic.