Numerical Analysis II: Homework 2

Peter Shaffery

February 8, 2017

1 Problems

Problem 1a. Show that the Hilbert matrix is positive definite.

Proof. This follows almost immediately from the hint given. Each entry of the Hilbert matrix is given $H_{ij} = \frac{1}{i+j-1} = \int_0^1 x^{i+j-2}dx$, which is an inner product of monomials x^{i-1} and x^{j-1} (it's pretty trivial to show that the integral satisfies the linearity and symmetry requirements of inner products; and when both arguments are the same then the integrand is raised to an even power, hence must be non-negative so the integral is non-negative). Therefore H is a Gram matrix and as such is positive semi-definite.

Now assume that there exists a vector u such that $u^THu = 0$. Let the vector $w = [1, ..., x^n]^T$, hence $H = \int_0^1 ww^T dx$. Therefore $u^T \int_0^1 ww^T dx\ u = \int_0^1 u^T ww^T u\ dx = \int_0^1 (w^Tu)(w^Tu)dx = \int_0^1 (w^Tu)^2 dx = 0$. Since the integrand is squared and hence non-negative, if the integral equals zero than the integrand must be zero for all x, ie. $w^Tu = 0$. Clearly w is not zero for all values of x, so $u = 0$. Therefore for all nonzero u, $u^THu > 0$. $\square$

Problem 1b. Implement the power method to estimate the largest eigenvalue and its corresponding eigenvector for the Hilbert matrix with unspecified dimension (I chose $n = 4$ because it's simpler to typset the eigenvector, instead of some 16-entry thing).

The largest eigenvalue is $\lambda_1 \approx 1.5002$ and it's eigenvector is $v_1 = [0.3377+0.7171i, 0.1926+0.4088i, 0.1374+0.2917i, 0.1075+0.2281i]^T$ (see Appendix for code).

Problem 1c. Modify the power method to find the smallest eigenvalue of the Hilbert matrix when $n = 16$. Is this consistent with the estimate $\min_{\lambda\in\sigma(H)}|\lambda - \mu| \leq \|E\|_2$, where μ is the eigenvalue of the perturbed matrix $H + E$?

I peturbed the Hilbert matrix H by $E = -\lambda_1 I$, hence it's smallest eigenvalue λ_{16} goes to $\lambda_16 - \lambda_1$. Since H is positive definite all of its eigenvalues are positive, hence $|\lambda_{16} - \lambda_1|$ is the largest absolute value of the eigenvalues of the perturbed matrix $H + E$. I then used the power method to calculate the largest eigenvalue of $H + E$ and added λ_1 to this value.

The smallest eigenvalue of H is $\lambda_{16} \approx 1.036\text{e}-06$. This estimate is consistent with the given bound, $|\lambda_1 - \lambda_{16}| \leq \|E\|_2 = |\lambda_1|$, since all $\lambda_i > 0$.

Problem 1d. Assume that A is a real, symmetric matrix with eigenvalues $\lambda_1 = -\lambda_2$, where $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq ... \geq |\lambda_n|$. Suggest a modification to the power method to find the eigenvectors corresponding to λ_1 and λ_2.

First let $B = A + I$, so the eigenvalues of B are just $\lambda_i + 1$. Since $|\lambda_1 + 1| \neq |\lambda_2 + 1|$, the matrix B has a dominant eigenvalue λ_b which can be found with the power method. Subtract 1 from λ_b to find either λ_1 or λ_2 and note that the eigenvector for λ_b corresponds to either λ_1 or λ_2. WLOG assume it belongs to λ_1 (it is trivial to confirm this in algorithm). Now set $B' = A - \lambda_1 I$, so that the dominant eigenvalue of B' is $2\lambda_2$. Find this using the power method and note that its eigenvector is the same as the eigenvector for λ_2. We have now found both x_1 and x_2

Problem 1e. A real, symmetric matrix A has an eigenvalue 1 with multiplicity 8; the rest of its eigenvalues are ≤ 0.1 in absolute value. Describe an algorithm, based on the power method, to calculate a basis for the 8-dimensional subspace spanned by the eigenvectors of the dominant eigenvalue. Estimate the number of iterations necessary to achieve double precision accuracy.

Randomly select (and normalize) 8 vectors $\vec{q}_{i=1,...8}$. Note that $\vec{q}_i = \sum_{i=1}^{8} b_i x_i + \sum_{i=1}^{9} b_i \lambda_i x_i$ where $|\lambda_{i \geq 9}| \leq 0.1$. Then $A^k \vec{q}_i \approx \sum_{i=1}^{8} b_i x_i$ where the $x_{i=1,...,8}$ are the eigenvectors we want. The error in this approximation goes as 0.1^k, so choose $k \geq 3$ to achieve double precision for each of the $\vec{q}_i$ (this puts us at 24 iterations). Since the $A^k \vec{q}_i$ (approximately) belong to the 8-dimensional eigenspace corresponding to the dominant eigenvalue, we can use Gram-Schmidt to orthogonalize the $A^k \vec{q}_i$ to achieve the desired basis.

2 Appendix

```
import cmath as c
import numpy as np
import scipy as sp
import numpy.linalg as npla
import scipy.linalg as scla
#############
#Question 1b#
#############
n = 16
H = scla.hilbert(n)

q = np.array([complex(i,np.random.rand()) for i in np.random.rand(n)])
q = q/npla.norm(q,2)

steps = 5000
for i in range(0,steps):
    q = np.dot(H,q)
    q = q/npla.norm(q,2)
    l = np.dot(np.conj(q),np.dot(H,q))

print('The largest eigenvalue value is: %s' %str(l.real))
print('It\'s eigenvector is:\n%s' %np.array2string(q))

#############
#Question 1c#
#############
n=16
l1 = l.real #The eigenvalues are all real, so ditch any rounding errors
mu = l1
H = scla.hilbert(n)
Hp = H - mu*np.eye(n)

q = np.array([complex(i,np.random.rand()) for i in np.random.rand(n)])
q = q/npla.norm(q,2)

steps = 50000
for i in range(0,steps):
    q = np.dot(Hp,q)
    q = q/npla.norm(q,2)
    l = np.dot(np.conj(q),np.dot(Hp,q))
l2 = l.real + mu
print('The smallest eigenvalue value is: %s' %str(l2))
print('It\'s eigenvector is:\n%s' %np.array2string(q))
```