Special Topics in Applied Mathematics I
Solution 1

XIA JIAHAN

January 21, 2026

Question 1

Iterative eigenvalue methods: an intuitive story

Think of a matrix A as a machine that acts on vectors: it can rotate, shear, and stretch them. Most directions are changed in complicated ways, but some special directions come back pointing the same way after applying A. Those directions are *eigenvectors* $v \neq 0$ satisfying

$$Av = \lambda v,$$

where λ is the corresponding *eigenvalue* (the factor by which A stretches v). When matrices are large, computing all eigenvalues exactly (e.g., from a characteristic polynomial) is impractical and numerically fragile. Iterative methods instead exploit one operation we can often do efficiently: repeatedly multiply by A (or solve linear systems involving A).

Power iteration: repeated multiplication reveals the dominant direction

The simplest method is *power iteration*. Choose any nonzero starting vector b_0 and repeat:

$$\begin{aligned} c_{k+1} &= Ab_k, \\ b_{k+1} &= \frac{c_{k+1}}{\|c_{k+1}\|}. \end{aligned}$$

Normalization "resets the length" so we can watch the direction. The method tends to align b_k with the eigenvector associated with the eigenvalue of largest magnitude.

Why it converges (qualitative). Assume A is diagonalizable with eigenpairs (λ_i, v_i) and $|\lambda_1| > |\lambda_2| \geq \cdots$. Write the starting vector as a combination of eigenvectors:

$$b_0 = \sum_{i=1}^{n} \alpha_i v_i \quad (\alpha_1 \neq 0).$$

Then

$$A^k b_0 = \sum_{i=1}^{n} \alpha_i \lambda_i^k v_i = \lambda_1^k \left(\alpha_1 v_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1} \right)^k v_i \right).$$

As k grows, the ratios $\left(\frac{\lambda_i}{\lambda_1}\right)^k$ shrink toward 0, so the direction of $A^k b_0$ becomes dominated by v_1. The convergence speed is governed by the spectral gap; a common rule of thumb is that the directional error shrinks like

$$\left| \frac{\lambda_2}{\lambda_1} \right|^k.$$

Estimating the eigenvalue: Rayleigh quotient

Once we have an approximate eigenvector x, a natural estimate of its eigenvalue is the *Rayleigh quotient*

$$R(x) = \frac{x^\top A x}{x^\top x}.$$

If x is exactly an eigenvector v, then $R(v) = \lambda$ because $v^\top A v = v^\top(\lambda v) = \lambda\, v^\top v$. When x is close to an eigenvector, $R(x)$ typically gives a good scalar approximation of the associated eigenvalue, so one can monitor the sequence $R(b_k)$ during power iteration.

Question 2

Appendix

AI conversation for Q1: `https://www.perplexity.ai/search/i-am-going-to-introduce-how-it-YmfU`
`0`