Rayleigh Quotient Iteration

- Describe inverse iteration.
  \[ (A^{-1})^k x_0 \]
- Describe Rayleigh Quotient Iteration.
  \[ \frac{\dot{x}^T A x}{\dot{x}^T x} \]
Demo: Power Iteration and its Variants
Computing Multiple Eigenvalues

- All Power Iteration Methods compute one eigenvalue at a time. What if I want all eigenvalues?

  - Suppose I know $\lambda$ so that $Ax = \lambda x$

    $A = Q \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix} Q^T$

    $\Rightarrow$ Deflation

  - Power it w/ multiple vectors
Simultaneous Iteration

- What happens if we carry out power iteration on multiple vectors simultaneously?

\[ X_0 \in \mathbb{R}^{n \times p} \]

\[ X_{k+1} = AX_k \]

- Drawbacks:
  - \( p \) times same answer (eigvec for largest ev.)
  - Expensive
  - Ill-conditioned
Orthogonal Iteration

\[ X_0 \in \mathbb{R}^{n \times p} \]

\[ Q_k \mathbf{e}_k = X_k \]

\[ X_{\text{new}} = A Q_k \]

\[ Q_0 \mathbf{e}_0 = X_0 \]

\[ x_1 = A Q_0 \Rightarrow A = Q_1 \mathbf{e}_1 \mathbf{e}_1^T \]

\[ Q_1 \mathbf{e}_1 = X_1 \]

\[ x_2 = A Q_1 \]

\[ Q_n \mathbf{e}_n = X_n \Rightarrow A = Q_n \mathbf{e}_n \mathbf{e}_n^T \]

\[ X_n = A Q_n \]

\[ A = Q \begin{pmatrix} \sqrt{n} \end{pmatrix} Q^T \]

\[ A \approx Q_n \mathbf{e}_n \mathbf{e}_n^T \]

\[ x_2 = Q_n^T A Q_n \approx \mathbf{e}_n \]

\( \rho < n \) starting vectors

- expensive

- slow/linear convergence
Demo: Orthogonal Iteration
In-class activity: Eigenvalue Iterations
QR Iteration/QR Algorithm

Orthogonal iteration:
\[ X_0 = A \]
\[ Q_k R_k = X_k \]
\[ X_{k+1} = A Q_k \]

Tracing through reveals:
- \( \hat{X}_k = \bar{X}_{k+1} \)
- \( Q_0 = \bar{Q}_0 \)
  \[ Q_1 = \bar{Q}_0 \bar{Q}_1 \]
  \[ Q_k = \bar{Q}_0 \bar{Q}_1 \ldots \bar{Q}_k \]

Orthogonal iteration showed: \( \hat{X}_k = \bar{X}_{k+1} \) converge. Also:

\[ \bar{X}_{k+1} = \bar{R}_k \bar{Q}_k = \bar{Q}_k^T \bar{X}_k \bar{Q}_k, \]

so the \( \bar{X}_k \) are all similar → all have the same eigenvalues.

→ QR iteration produces Schur form.
QR Iteration: Incorporating a Shift

- How can we accelerate convergence of QR iteration using shifts?

\[
\tilde{X}_0 = A \\
\tilde{Q}_k \tilde{R}_k = \tilde{X}_n - \sigma_k I \Rightarrow \tilde{R}_n = \tilde{Q}_n^\dagger [\tilde{X}_n - \sigma_k I] \\
\tilde{X}_{n+1} = \tilde{R}_n \tilde{Q}_n + \sigma_n I \\
\tilde{X}_{n+1} = \tilde{R}_n \tilde{Q}_n + \sigma_n I = \tilde{Q}_n^\dagger [\tilde{X}_n - \sigma_k I] \tilde{Q}_n + \sigma_n I \\
= \tilde{Q}_n^\dagger \tilde{X}_n \tilde{Q}_n - \sigma_k I \tilde{Q}_n^\dagger + \sigma_n I \\
= \tilde{Q}_n^\dagger \tilde{X}_n \tilde{Q}_n - \sigma_k I + \sigma_n I
\]

1. Pickings \( \sigma_k \approx (\tilde{X}_k)^{mn} \)

2. Pick two eigenvalues \( \Box \) in the BR of \( \tilde{X}_k \)
\[ A x = \lambda x \]

\[ (A - \lambda I) x = 0 \]
QR Iteration: Computational Expense

- A full QR factorization at each iteration costs $O(n^3)$—can we make that cheaper?
4.4 Krylov Space Methods