Lecture 15
The QR Algorithm I

MIT 18.335J / 6.337J
Introduction to Numerical Methods
Per-Olof Persson
October 31, 2006

Real Symmetric Matrices

- We will only consider eigenvalue problems for real symmetric matrices
- Then $A = A^T ∈ \mathbb{R}^{m × m}$, $x ∈ \mathbb{R}^m$, $x^* = x^T$, and $||x|| = \sqrt{x^T x}$
- $A$ then also has
  real eigenvalues: $λ_1, \ldots, λ_m$
  orthonormal eigenvectors: $q_1, \ldots, q_m$
- Eigenvectors are normalized $||q_j|| = 1$, and sometimes the eigenvalues are ordered in a particular way
- Initial reduction to tridiagonal form assumed
  - Brings cost for typical steps down from $O(m^3)$ to $O(m)$

Rayleigh Quotient

- The Rayleigh quotient of $x ∈ \mathbb{R}^m$:
  $r(x) = \frac{x^T Ax}{x^T x}$
- For an eigenvector $x$, the corresponding eigenvalue is $r(x) = λ$
- For general $x$, $r(x) = α$ that minimizes $||Ax - αx||_2$
- $x$ eigenvector of $A$ if $\nabla r(x) = 0$ with $x ≠ 0$
- $r(x)$ is smooth and $\nabla r(q_j) = 0$, therefore quadratically accurate:
  $r(x) - r(q_j) = O(||x - q_j||^2)$ as $x → q_j$

Power Iteration

- Simple power iteration for largest eigenvalue:

  **Algorithm: Power Iteration**

  $v^{(0)}$ is some vector with $||v^{(0)}|| = 1$
  for $k = 1, 2, \ldots$
  $w = Av^{(k−1)}$ apply $A$
  $v^{(k)} = w/||w||$ normalize
  $λ^{(k)} = (v^{(k)})^T Av^{(k)}$ Rayleigh quotient

  - Termination conditions usually omitted

Convergence of Power Iteration

- Expand initial $v^{(0)}$ in orthonormal eigenvectors $q_j$, and apply $A^k$:
  $v^{(0)} = a_1 q_1 + a_2 q_2 + \cdots + a_m q_m$
  $v^{(k)} = c_k A^k v^{(0)}$
  $= c_k (a_1 λ_1^k q_1 + a_2 λ_2^k q_2 + \cdots + a_m λ_m^k q_m)$
  $= c_k λ_1^k (a_1 q_1 + a_2 λ_1/λ_2) q_2 + \cdots + a_m (λ_m/λ_1) q_m$,

  - If $|λ_1| > |λ_2| ≥ \cdots ≥ |λ_m| ≥ 0$ and $q_1^T v^{(0)} ≠ 0$, this gives:
  $||v^{(k)} - (± q_1)|| = O \left( \frac{|λ_2|}{|λ_1|} \right)^k$, $|λ^{(k)} - λ_1| = O \left( \frac{|λ_2|}{|λ_1|} \right)^{2k}$

  - Finds the largest eigenvalue (unless eigenvector orthogonal to $v^{(0)}$)
  - Linear convergence, factor $≈ λ_2/λ_1$ at each iteration

Inverse Iteration

- Apply power iteration on $(A - μI)^{-1}$, with eigenvalues $(λ_j - μ)^{-1}$

  **Algorithm: Inverse Iteration**

  $v^{(0)}$ is some vector with $||v^{(0)}|| = 1$
  for $k = 1, 2, \ldots$
  Solve $(A - μI)w = v^{(k−1)}$ for $w$, apply $(A - μI)^{-1}$
  $v^{(k)} = w/||w||$ normalize
  $λ^{(k)} = (v^{(k)})^T Av^{(k)}$ Rayleigh quotient

  - Converges to eigenvector $q_j$ if the parameter $μ$ is close to $λ_j$:
  $||v^{(k)} - (± q_j)|| = O \left( \frac{μ - λ_j}{μ - λ_j^2} \right)^k$, $|λ^{(k)} - λ_j| = O \left( \frac{|λ_j^2 - λ_j|}{μ - λ_j} \right)^{2k}$
Rayleigh Quotient Iteration

- Parameter $\mu$ is constant in inverse iteration, but convergence is better for $\mu$ close to the eigenvalue.
- Improvement: At each iteration, set $\mu$ to last computed Rayleigh quotient.

Algorithm: Rayleigh Quotient Iteration

$\mathbf{v}^{(0)}$ is some vector with $\|\mathbf{v}^{(0)}\| = 1$

$\lambda^{(0)} = (\mathbf{v}^{(0)})^T \mathbf{A} \mathbf{v}^{(0)}$ = corresponding Rayleigh quotient

for $k = 1, 2, \ldots$

- Solve $(\mathbf{A} - \lambda^{(k-1)} I) \mathbf{w} = \mathbf{v}^{(k-1)}$ for $\mathbf{w}$
- Apply matrix
- Normalize

Rayleigh quotient

Convergence of Rayleigh Quotient Iteration

- Cubic convergence in Rayleigh quotient iteration:

$$\|\mathbf{v}^{(k+1)} - (\pm q_j)\| = O(\|\mathbf{v}^{(k)} - (\pm q_j)\|^3)$$

and

$$|\lambda^{(k+1)} - \lambda_j| = O(|\lambda^{(k)} - \lambda_j|^3)$$

- Proof idea: If $\mathbf{v}^{(k)}$ is close to an eigenvector, $\|\mathbf{v}^{(k)} - q_j\| \leq \epsilon$, then the accurate of the Rayleigh quotient estimate $\lambda^{(k)}$ is $|\lambda^{(k)} - \lambda_j| = O(\epsilon^2)$. One step of inverse iteration then gives

$$\|\mathbf{v}^{(k+1)} - q_j\| = O(|\lambda^{(k)} - \lambda_j| \|\mathbf{v}^{(k)} - q_j\|) = O(\epsilon^3)$$

The QR Algorithm

- Remarkably simple algorithm: QR factorize and multiply in reverse order.

Algorithm: “Pure” QR Algorithm

$A^{(0)} = \mathbf{A}$

for $k = 1, 2, \ldots$

$Q^{(k)} R^{(k)} = A^{(k-1)}$ QR factorization of $A^{(k-1)}$

$A^{(k)} = R^{(k)} Q^{(k)}$ Recombine factors in reverse order

- With some assumptions, $A^{(k)}$ converge to a Schur form for $\tilde{A}$ (diagonal if $\tilde{A}$ symmetric).
- Similarity transformations of $\tilde{A}$:

$$A^{(k)} = R^{(k)} Q^{(k)} = (Q^{(k)})^T A^{(k-1)} Q^{(k)}$$

Unnormalized Simultaneous Iteration

- Define well-behaved basis for column space of $V^{(k)}$ by $\tilde{Q}^{(k)} \tilde{R}^{(k)} = V^{(k)}$.
- Make the assumptions:
  - The leading $n + 1$ eigenvalues are distinct.
  - All principal leading principal submatrices of $\tilde{Q}^T V^{(0)}$ are nonsingular, where columns of $\tilde{Q}$ are $q_1, \ldots, q_n$.

We then have that the columns of $\tilde{Q}^{(k)}$ converge to eigenvectors of $\mathbf{A}$:

$$\|q_j^{(k)} - \pm q_j\| = O(C^k)$$

where $C = \max_{1 \leq j \leq n} |\lambda_{k+1}^j/|\lambda_k|$.

- Proof. Textbook / Blackboard.

Unnormalized Simultaneous Iteration

- To understand the QR algorithm, first consider a simpler algorithm.
- Simultaneous Iteration is power iteration applied to several vectors.
- Start with linearly independent $\mathbf{v}_1^{(0)}, \ldots, \mathbf{v}_n^{(0)}$.
- We know from power iteration that $A^k \mathbf{v}_1^{(0)}$ converges to $q_1$.
- With some assumptions, the space $\langle A^k \mathbf{v}_1^{(0)}, \ldots, A^k \mathbf{v}_n^{(0)} \rangle$ should converge to $q_1, \ldots, q_n$.
- Notation: Define initial matrix $V^{(0)}$ and matrix $V^{(k)}$ at step $k$:

$$V^{(0)} = \begin{bmatrix} \mathbf{v}_1^{(0)} & \cdots & \mathbf{v}_n^{(0)} \end{bmatrix}, \quad V^{(k)} = A^k V^{(0)} = \begin{bmatrix} \mathbf{v}_1^{(k)} & \cdots & \mathbf{v}_n^{(k)} \end{bmatrix}$$

Simultaneous Iteration

- The matrices $V^{(k)} = A^k V^{(0)}$ are highly ill-conditioned.
- Orthonormalize at each step rather than at the end.

Algorithm: Simultaneous Iteration

Pick $\tilde{Q}^{(0)} \in \mathbb{R}^{n \times n}$

for $k = 1, 2, \ldots$

$$Z = A^{k-1} \tilde{Q}^{(k-1)}$$

$$\tilde{Q}^{(k)} \tilde{R}^{(k)} = Z$$ Reduced QR factorization of $Z$

- The column spaces of $\tilde{Q}^{(k)}$ and $Z^{(k)}$ are both equal to the column space of $A^k \tilde{Q}^{(0)}$, therefore same convergence as before.
Simultaneous Iteration $\Leftrightarrow$ QR Algorithm

- The QR algorithm is equivalent to simultaneous iteration with $\hat{Q}(0) = I$
- Notation: Replace $\hat{R}^{(k)}$ by $R^{(k)}$, and $\hat{Q}^{(k)}$ by $Q^{(k)}$

Simultaneous Iteration:

$Q^{(0)} = I$
$Z = AQ^{(k-1)}$
$Z = \hat{Q}^{(k)}R^{(k)}$
$A^{(k)} = (Q^{(k)})^T AQ^{(k)}$

Unshifted QR Algorithm:

$A^{(0)} = A$
$A^{(k-1)} = Q^{(k)}R^{(k)}$
$A^{(k)} = R^{(k)}Q^{(k)}$
$Q^{(k)} = Q^{(1)}Q^{(2)} \cdots Q^{(k)}$

- Also define $R^{(k)} = R^{(k)}R^{(k-1)} \cdots R^{(1)}$
- Now show that the two processes generate same sequences of matrices

Simultaneous Iteration $\Leftrightarrow$ QR Algorithm

- Both schemes generate the QR factorization $A^k = Q^{(k)}R^{(k)}$ and the projection $A^{(k)} = (Q^{(k)})^T AQ^{(k)}$
- Proof. $k = 0$ trivial for both algorithms.
  For $k \geq 1$ with simultaneous iteration, $A^{(k)}$ is given by definition, and
  $A^k = A Q^{(k-1)} R^{(k-1)} = Q^{(k)} R^{(k)} R^{(k-1)} = Q^{(k)} R^{(k)}$
  For $k \geq 1$ with unshifted QR, we have
  $A^k = A Q^{(k-1)} R^{(k-1)} = Q^{(k-1)} A^{(k-1)} R^{(k-1)} = Q^{(k)} R^{(k)}$
  and
  $A^{(k)} = (Q^{(k)})^T A^{(k-1)} Q^{(k)} = (Q^{(k)})^T A Q^{(k)}$