The QR Algorithm

Part 1: "Pure" QR for Symmetric Matrices

In this lecture, $A = A^T$ (real symmetric)

$$\Rightarrow \quad \underset{V}{A \begin{bmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_n \\ | & | & & | \end{bmatrix}} = \underset{V}{\begin{bmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_n \\ | & | & & | \end{bmatrix}} \underset{\Lambda}{\begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}}$$

orthogonal matrix $V^{-1} = V^*$ real eigenvalues

In contrast to $A = LU$ and $A = QR$, there is no finite-length algorithm to compute $A = V\Lambda V^*$.

Instead, we will approximate V and Λ **iteratively**.

"Pure" QR-algorithm (not to be confused with QR factorization)

$A^{(0)} = A$
for $k = 1, 2, 3, \ldots$
 $Q^{(k)} R^{(k)} = A^{(k-1)}$ [QR factorization of $A^{(k-1)}$]
 $A^{(k)} = R^{(k)} Q^{(k)}$ [Combine factors "backward"]

Remarkably, the iteration converges as follows:

[Strictly, this is true when $\lambda_i \neq \lambda_j$ for $i \neq j$]

$$A^{(k)} \to \Lambda \quad \text{as} \quad k \to \infty$$

$$Q^{(1)}Q^{(2)}\cdots Q^{(k)} \to V \quad \text{as} \quad k \to \infty$$

↑ strictly speaking, signs of columns can change - more later.

Note, convergence occurs in the limit $k \to \infty$. In practice, we need a stopping criteria so that the for loop terminates in finitely many operations. The outputs $A^{(k_*)}$ and $Q^{(k_*)}$ will be approximations to Λ and V.

Power iterations

To understand the accuracy of the approx. $A^{(k)}$ and $Q^{(k)}$, we can relate them to simpler iterates whose convergence is easier to study.

Given $\hat{x}^{(0)} \in \mathbb{R}^m$, $\|\hat{x}^{(0)}\| = 1$

for $k = 1, 2, 3, \ldots$

$x^{(k)} = A\hat{x}^{(k-1)}$ Power iteration

$\hat{x}^{(k)} = x^{(k)} / \|x^{(k)}\|$ Normalize vector

$\lambda^{(k)} = \hat{x}^{(k)T} A \hat{x}^{(k)}$ "Rayleigh-Quotient"

Convergence of Power Iterations

Assumptions:

1) $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$ (Spectral gap)

2) $c_1 := v_1^T \hat{x}^{(0)} \neq 0$ (non-deficient initial $\hat{x}^{(0)}$)

$$\Rightarrow \hat{x}^{(k)} = \frac{A\hat{x}^{(k-1)}}{\|x^{(k)}\|} = \cdots = \frac{A^k \hat{x}^{(0)}}{\|x^{(k)}\| \cdots \|x^{(1)}\|} \leftarrow$$

just a scalar factor $= \alpha_k$

$$\Rightarrow \hat{x}^{(k)} = \alpha_k^{-1} A^k [\underbrace{c_1 v_1 + \cdots + c_n v_n}_{\hat{x}^{(0)}}]$$

$$= \alpha_k^{-1} \lambda_1^k [c_1 v_1 + \underbrace{c_2 \frac{\lambda_2^k}{\lambda_1^k} v_2 + \cdots + c_n \frac{\lambda_n^k}{\lambda_1^k} v_n}_{= O(|\frac{\lambda_2}{\lambda_1}|^k)}]$$

$$\sim \alpha_k^{-1} \lambda_1^k c_1 v_1 \quad \text{as } k \to \infty.$$

Since $\|\hat{x}^{(k)}\| = 1$ for every $k = 0, 1, 2, \ldots$, we must have

$$\alpha_k^{-1} \lambda_1^k c_1 \to \pm 1 \quad \text{as } k \to \infty \quad \text{b/c } \|v_1\| = 1$$

(orthonormal eigvecs)

so that $\hat{x}^{(k)} \to \pm v_1$ as $k \to \infty$.

Then $\hat{x}^{(k)T} A \hat{x}^{(k)} \to v_1^T A v_1 = \lambda_1 v_1^T v_1 = \lambda_1$.

Simultaneous Power Iterations

What about $v_2, v_3, \ldots$ and $\lambda_2, \lambda_3, \ldots$?

<u>Idea</u>: run power iteration on multiple vectors simultaneously.

$$\underbrace{\begin{bmatrix} | & & | \\ x_1^{(k)} & \cdots & x_n^{(k)} \\ | & & | \end{bmatrix}}_{\hat{X}^{(k)}} = A \underbrace{\begin{bmatrix} | & & | \\ q_1^{(k-1)} & \cdots & q_n^{(k-1)} \\ | & & | \end{bmatrix}}_{\hat{Q}^{(k-1)}}$$

Power iteration

$$\underbrace{\begin{bmatrix} | & & | \\ q_1^{(k)} & \cdots & q_n^{(k)} \\ | & & | \end{bmatrix}}_{\hat{Q}^{(k)}} \underbrace{\begin{bmatrix} \; & \; & \; \\ \; & \ddots & \; \\ \; & \; & \; \end{bmatrix}}_{\hat{R}^{(k)}} = \underbrace{\begin{bmatrix} | & & | \\ x_1^{(k)} & \cdots & x_n^{(k)} \\ | & & | \end{bmatrix}}_{\hat{X}^{(k)}}$$

orthonormalize vectors

$$\hat{A}^{(k)} = \hat{Q}^{(k)T} A \hat{Q}^{(k)}$$

"generalized Rayleigh Quotient"

<u>Intuition:</u> The first column is just power iteration so $q_1^{(k)} \to \pm v_1$ as $k \to \infty$. As $q_1^{(k)} \to \pm v_1$, then $v_1^T q_2^{(k)} \approx q_1^{(k)T} q_2^{(k)} = 0$, so $q_2^{(k)} \to \pm v_2$ the second largest eigenvalue's eigvec. Similarly, $q_3^{(k)} \to \pm v_3$, $q_4^{(k)} \to \pm v_4$, and so on. In other words $Q^{(k)} \xrightarrow{\pm} V$ as $k \to \infty$.

where $\overset{\pm}{\to}$ means convergence up to column signs.
Moreover, $\hat{A}^{(k)} = \hat{Q}^{(k)T} A \hat{Q}^{k} \to V^T A V = \Lambda$.

Thm 1 Suppose that A has distinct eigenvalues with $|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|$, and let $\hat{Q}^{(0)}$ have orthonormal columns with $V^T \hat{Q}^{(0)}$ of full (column) rank. Then

$$\hat{Q}^{(k)} \overset{\pm}{\to} V \quad \text{and} \quad \hat{A}^{(k)} \to \Lambda$$

"Pure" QR-algorithm as Simultaneous Iteration

Analysis of QR-algorithm rests on its connection to Simultaneous Iteration with initial vectors $\hat{Q}^{(0)} = I$ (identity matrix).

Simultaneous Iter.

$\hat{Q}^{(0)} = I$

$\hat{X}^{(k)} = A \hat{Q}^{(k-1)}$

$\hat{Q}^{(k)} \hat{R}^{(k)} = \hat{X}^{(k)}$

$\hat{A}^{(k)} = \hat{Q}^{(k)T} A \hat{Q}^{(k)}$

QR-algorithm

$A^{(0)} = A$

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

$A^{(k)} = R^{(k)} Q^{(k)}$

Let $\underline{R}^{(k)} = R^{(k)} R^{(k-1)} \cdots R^{(1)}$

$\underline{Q}^{(k)} = Q^{(1)} Q^{(2)} \cdots Q^{(k)}$

Thm 2 Simultaneous Iteration generate identical sequences of matrices

$$\hat{A}^{(k)} = A^{(k)}$$
$$\tilde{Q}^{(k)} = \underline{Q}^{(k)}$$

and $$\underline{R}^{(k)} = \hat{R}^{(k)} \hat{R}^{(k-1)} \cdots \hat{R}^{(1)}.$$

Moreover, $A^k = \underline{Q}^{(k)} \underline{R}^{(k)}$ and $A^{(k)} = \underline{Q}^{(k)T} A \underline{Q}^{(k)}$.

Pf Look at the first iteration and proceed by induction.

$\underbrace{SI}$ $\underbrace{QR}$

$\underline{k=1}$

$\hat{Q}^{(1)} \hat{R}^{(1)} = \hat{X}^{(1)} = AI = A$

$A^{(1)} = \hat{Q}^{(1)T} A \hat{Q}^{(1)}$

$Q^{(1)} R^{(1)} = A^{(0)} = A$

$A^{(1)} = R^{(1)} Q^{(1)}$

$= \underbrace{Q^{(1)T} A}_{R^{(1)}} Q^{(1)}$

$\underline{k \geq 1}$ Suppose thm holds for first $k-1$ iterations, show that it then must hold for k^{th} iteration.

<u>SI</u>		<u>QR</u>
$A^k = A\underbrace{\underline{Q}^{(k-1)}}\underline{R}^{(k-1)}$	inductive hypothesis $A^{(k-1)} = \underline{Q}^{(k-1)}\underline{R}^{(k-1)}$	$A^k = A\underbrace{\underline{Q}^{(k-1)}}_{=\underline{Q}^{(k-1)}A^{(k-1)}}\underline{R}^{(k-1)}$

SI:

$$A^k = A\underbrace{\underline{Q}^{(k-1)}}\underline{R}^{(k-1)}$$

$$= (\hat{Q}^{(k)}R^{(k)})\underline{R}^{(k-1)}$$

$$= \hat{Q}^{(k)}\underline{R}^{(k)} \quad (\text{def. of } \underline{R}^{(k)})$$

Since QR is unique,

$$\hat{Q}^{(k)} = \underline{Q}^{(k)}$$

$$\therefore \quad A^k = \underline{Q}^{(k)}\underline{R}^{(k)} \quad ✓$$

Then,

$$A^{(k)} = \hat{Q}^{(k)T}A\hat{Q}^{(k)}$$

$$= \underline{Q}^{(k)T}A\underline{Q}^{(k)} \quad ✓$$

QR:

$$A^k = A\underbrace{\underline{Q}^{(k-1)}}_{=\underline{Q}^{(k-1)}A^{(k-1)}}\underline{R}^{(k-1)}$$

by ind. hyp. $A^{(k-1)} = \underline{Q}^{(k-1)T}A\underline{Q}^{(k-1)}$

$$= \underline{Q}^{(k-1)}\underbrace{A^{(k-1)}}_{=Q^{(k)}R^{(k)}}\underline{R}^{(k-1)}$$

$$= \underbrace{\underline{Q}^{(k-1)}Q^{(k)}}_{\underline{Q}^{(k)}}\underbrace{R^{(k)}\underline{R}^{(k-1)}}_{\underline{R}^{(k)}}$$

$$\therefore \quad A^k = \underline{Q}^{(k)}\underline{R}^{(k)} \quad ✓$$

Now, $A^{(k)} = \underbrace{R^{(k)}}Q^{(k)}$

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

b/c $Q^{(k)}R^{(k)} = A^{(k-1)}$.

Then, $A^{(k)} = Q^{(k)T}\underbrace{A^{(k-1)}}_{\underline{Q}^{(k-1)T}A\underline{Q}^{(k-1)}}Q^{(k)}$

$$= \underline{Q}^{(k)T}A\underline{Q}^{(k)} \quad ✓$$

$$\boxed{\text{The connection between } \underline{\text{SI}} \text{ and } \underline{\text{QR}} \text{ immediately establishes the convergence of QR-algorithm under hypotheses of Thm 1.}}$$

To make QR-algorithm computationally efficient, we need to make sure that

(1) each iteration is not too expensive

(2) The iterates converge quickly

Next time, we'll address both (1)-(2) and describe how to adapt QR to compute the SVD of general non-symmetric matrices.