Implementing and Exploring the QR Algorithm

Arleth Salinas

December 5, 2021

THE UNIVERSITY OF CHICAGO

Overview

1. Recall Gram-Schmidt and QR Algorithm
2. Implementation
3. Light Experimenting

Gram-Schmidt

Let $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ be a linearly independent system.
Gram-Schmidt constructs an orthonormal basis $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ such that

$$\text{span}\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} = \text{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$$

Gram-Schmidt Process

$\mathbf{v}_1 = \mathbf{x}_1$, $\mathbf{e}_1 = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}$.

$\mathbf{v}_2 = \mathbf{x}_2 - \text{proj}_{\mathbf{v}_1}(\mathbf{x}_2)$, $\mathbf{e}_2 = \frac{\mathbf{v}_2}{\|\mathbf{v}_2\|}$.

$\mathbf{v}_3 = \mathbf{x}_3 - \text{proj}_{\mathbf{v}_1}(\mathbf{x}_3) - \text{proj}_{\mathbf{v}_2}(\mathbf{x}_3)$, $\mathbf{e}_2 = \frac{\mathbf{v}_3}{\|\mathbf{v}_3\|}$.

$\vdots$

$\mathbf{v}_n = \mathbf{x}_n - \text{proj}_{\mathbf{v}_1}(\mathbf{x}_n) - \text{proj}_{\mathbf{v}_2}(\mathbf{x}_n) - \ldots - \text{proj}_{\mathbf{v}_{n-1}}(\mathbf{x}_n)$,$\mathbf{e}_n = \frac{\mathbf{v}_n}{\|\mathbf{v}_n\|}$.

So to find $\mathbf{v}_k$:

$$\mathbf{v}_k = \mathbf{x}_k - \sum_{j=1}^{n-1} \text{proj}_{\mathbf{v}_j}(\mathbf{x}_k)$$

QR Algorithm

The QR algorithm, developed my John Francis uses the QR decomposition of a matrix to find the eigenvalues of a matrix.

QR Decomposition via Gram-Schmidt

$\mathbf{v}_1 = \mathbf{x}_1$, $\mathbf{e}_1 = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}$.

$\mathbf{v}_2 = \mathbf{x}_2 - \mathrm{proj}_{\mathbf{v}_1}(\mathbf{x}_2)$, $\mathbf{e}_2 = \frac{\mathbf{v}_2}{\|\mathbf{v}_2\|}$.

$\mathbf{v}_3 = \mathbf{x}_3 - \mathrm{proj}_{\mathbf{v}_1}(\mathbf{x}_3) - \mathrm{proj}_{\mathbf{v}_2}(\mathbf{x}_3)$, $\mathbf{e}_3 = \frac{\mathbf{v}_3}{\|\mathbf{v}_3\|}$.

$\vdots$

$\mathbf{v}_n = \mathbf{x}_n - \mathrm{proj}_{\mathbf{v}_1}(\mathbf{x}_n) - \mathrm{proj}_{\mathbf{v}_2}(\mathbf{x}_n) - \ldots - \mathrm{proj}_{\mathbf{v}_{n-1}}(\mathbf{x}_n)$,$\mathbf{e}_n = \frac{\mathbf{v}_n}{\|\mathbf{v}_n\|}$.

Let $\mathbf{Q} = [\mathbf{e}_1|\mathbf{e}_2|\ldots|\mathbf{e}_n]$, $\mathbf{R} = \begin{bmatrix} \mathbf{e}_1 \cdot \mathbf{x}_1 & \mathbf{e}_1 \cdot \mathbf{x}_2 & \mathbf{e}_1 \cdot \mathbf{x}_3 & \cdots \\ 0 & \mathbf{e}_2 \cdot \mathbf{x}_2 & \mathbf{e}_2 \cdot \mathbf{x}_3 & \cdots \\ 0 & 0 & \mathbf{e}_3 \cdot \mathbf{x}_3 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$

QR Algorithm

Let $\mathbf{X} = \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$.
Initialize $\mathbf{X}_1 = \mathbf{X}$.

$$\mathbf{X}_1 = \mathbf{Q}_1\mathbf{R}_1$$

$$\mathbf{R}_1\mathbf{Q}_1 = \mathbf{X}_2 = \mathbf{Q}_2\mathbf{R}_2$$

$$\mathbf{R}_2\mathbf{Q}_2 = \mathbf{X}_3 = \mathbf{Q}_3\mathbf{R}_3$$

$$\vdots$$

Under the right conditions, the matrices $\mathbf{X}_k$ converge to a triangle matrix, the Schur form of $\mathbf{X}$ with the eigenvalues of $\mathbf{X}$ on the diagonal.

Implementation

There were two main steps to implementing the QR Algorithm.

Implementing QR Decomposition via Gram-Schmidt

- Limited myself to accepting only square matrices.

Implementing the QR Algorithm

- I had to incorporate a tolerance level and maximum iteration limit.
- I decided to consider Julia's eigenvalue function from their LinearAlgebra library as the "true" eigenvalues of the matrix.
- To calculate the error in the eigenvalues QR calculated versus the eigenvalues Julia determined, I used the L2 norm.

Light Experimenting

Recall, a complex number can be represented as

$$a + bi$$

where a is the real part and b is the imaginary part.

Matrices with Imaginary Eigenvalues

- I ran matrices that had only complex eigenvalues through the QR algorithm.
- I noticed, QR managed to find the real part of the eigenvalue, but did not manage to get the imaginary part.

$$\begin{bmatrix} cos(90) & -sin(90) \\ sin(90) & cos(90) \end{bmatrix}$$

- Julia says:
  -0.4480736161291702 - 0.8939966636005579im
  -0.4480736161291702 + 0.8939966636005579im
- QR says:
  -0.4480736161291702
  -0.4480736161291702

$$\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

- Julia says:
  1.0 - 1.0im
  1.0 + 1.0im
- QR says:
  1
  1

Thank you!