QR Algorithm

Contents

Householder Transformation

Motivation

Problem

Given $\mathbf{z} \in \mathbb{R}^m$, find an orthogonal matrix $H \in \mathbb{R}^{m \times m}$ such that $H\mathbf{z}$ is nonzero only in the first element.

- Since orthogonal matrices preserve the 2-norm, H must satisfy

$$H\mathbf{z} = \begin{bmatrix} \pm \|\mathbf{z}\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \pm \|\mathbf{z}\|_2 \, \mathbf{e}_1 .$$

- The **Householder transformation matrix** H defined by

$$H = I - 2\frac{\mathbf{v}\mathbf{v}^{\mathrm{T}}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}}, \quad \text{where } \mathbf{v} = \pm \|\mathbf{z}\|_2 \, \mathbf{e}_1 - \mathbf{z},$$

solves the problem. See Theorem 1 on the next slide.

Properties of Householder Transformation

Theorem 1

Let $\mathbf{v} = \|\mathbf{z}\|_2 \mathbf{e}_1 - \mathbf{z}$ *and let* H *be the Householder transformation defined by*

$$H = I - 2\frac{\mathbf{v}\mathbf{v}^{\mathrm{T}}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}}.$$

Then

1. H *is symmetric;*
2. H *is orthogonal;*
3. $H\mathbf{z} = \|\mathbf{z}\|_2 \mathbf{e}_1$.

- H is invariant under scaling of $\mathbf{v}$.
- If $\|\mathbf{v}\|_2 = 1$, then $H = I - \mathbf{v}\mathbf{v}^{\mathrm{T}}$.

Geometry Behind Householder Transformation

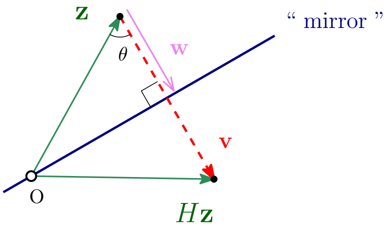
The Householder transformation matrix H is the reflector across $\langle \mathbf{v} \rangle^{\perp}$.

From any $\mathbf{z}$ to the "mirror":

$$\mathbf{w} = -\frac{\mathbf{z}^{\mathrm{T}}\mathbf{v}}{\sqrt{\mathbf{v}^{\mathrm{T}}\mathbf{v}}} \cdot \frac{\mathbf{v}}{\sqrt{\mathbf{v}^{\mathrm{T}}\mathbf{v}}} = -\mathbf{v}\frac{\mathbf{z}^{\mathrm{T}}\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}} .$$

From any $\mathbf{z}$ to its reflection:

$$H\mathbf{z} - \mathbf{z} = -2\mathbf{v}\frac{\mathbf{z}^{\mathrm{T}}\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}} .$$



Thus, for any $\mathbf{z}$,

$$H\mathbf{z} = \mathbf{z} - 2\mathbf{v}\frac{\mathbf{z}^{\mathrm{T}}\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}} = \left(I - 2\frac{\mathbf{v}\mathbf{v}^{\mathrm{T}}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}}\right)\mathbf{z} \quad \Longrightarrow \quad H = I - 2\frac{\mathbf{v}\mathbf{v}^{\mathrm{T}}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}} .$$

QR Factorization Algorithm

QR Factorization Algorithm via Triangularization

- The Gram-Schmidt orthogonalization (thin QR factorization) is unstable in floating-point calculations.
- **Stable alternative:** Find orthogonal matrices $H_1, H_2, \ldots, H_n$ so that

$$\underbrace{H_n H_{n-1} \cdots H_2 H_1}_{=:Q^{\mathrm{T}}} A = R\,.$$

  introducing zeros one column at a time below diagonal terms.
  Householder matrices will do.
- As a product of orthogonal matrices, Q^{T} is also orthogonal and so $(Q^{\mathrm{T}})^{-1} = Q$. Therefore,

$$A = QR\,.$$

MATLAB Implementation: MYQR

```
function [Q, R] = myqr(A)
  [m, n] = size(A);
  A0 = A;
  Q = eye(m);
  for j = 1:min(m,n)
      Aj = A(j:m, j:n);
      z = Aj(:, 1);
      v = z + sign0(z(1))*norm(z)*eye(length(z), 1);
      Hj = eye(length(v)) - 2/(v'*v) * v*v';
      Aj = Hj*Aj;
      H = eye(m);
      H(j:m, j:m) = Hj;
      Q = Q*H;
      A(j:m, j:n) = Aj;
  end
  R = A;
end
```

MATLAB Implementation: MYQR (cont')

(continued from the previous page)

```
% local function
function sign0(x)
  y = ones(size(x));
  y(x < 0) = -1;
end
```

- The MATLAB command `qr` works similar to, but more efficiently than, this.
- The function finds the factorization in $\sim (2mn^2 - n^3/3)$ flops asymptotically.

Which Reflector Is Better?

Recall:

$$H = I - 2\frac{\mathbf{v}\mathbf{v}^{\mathrm{T}}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}}, \quad \text{where } \mathbf{v} = \pm \|\mathbf{z}\|_2 \, \mathbf{e}_1 - \mathbf{z},$$

In `myqr.m`, the statement

```
v = z + sign0(z(1))*norm(z)*eye(length(z), 1);
```

defines $\mathbf{v}$ slightly differently, namely,

$$\mathbf{v} = \mathbf{z} \pm \|\mathbf{z}\|_2 \, \mathbf{e}_1.$$

This does not cause any difference since H is invariant under scaling of $\mathbf{v}$; see p. 5.

Which Reflector Is Better? (cont')

The sign of $\pm \|\mathbf{z}\|_2$ is chosen so as to avoid possible catastrophic cancellation in forming $\mathbf{v}$:

$$\mathbf{v} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} + \begin{bmatrix} \pm \|\mathbf{z}\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} z_1 \pm \|\mathbf{z}\|_2 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}$$

Subtractive cancellation may arise when $z_1 \approx \pm \|\mathbf{z}\|_2$.

- if $z_1 > 0$, use $z_1 + \|\mathbf{z}\|_2$;
- if $z_1 < 0$, use $z_1 - \|\mathbf{z}\|_2$;
- if $z_1 = 0$, either works.

For numerical stability, it is desirable to reflect $\mathbf{z}$ to the vector $s \|\mathbf{z}\|_2 \mathbf{e}_1$ that is not too close to $\mathbf{z}$ itself. (Trefethen & Bau)

Appendix: Gram-Schmidt Orthogonalization

The Gram–Schmidt Procedure

Problem: Orthogonalization

Given $\mathbf{a}_1, \ldots, \mathbf{a}_n \in \mathbb{R}^m$, construct orthonormal vectors $\mathbf{q}_1, \ldots, \mathbf{q}_n \in \mathbb{R}^m$ such that

$$\operatorname{span}\{\mathbf{a}_1, \ldots, \mathbf{a}_k\} = \operatorname{span}\{\mathbf{q}_1, \ldots, \mathbf{q}_k\}, \quad \text{for any } k \in \mathbb{N}[1, n].$$

- **Strategy.** At the jth step, find a unit vector $\mathbf{q}_j \in \operatorname{span}\{\mathbf{a}_1, \ldots, \mathbf{a}_j\}$ that is orthogonal to $\mathbf{q}_1, \ldots, \mathbf{q}_{j-1}$.
- **Key Observation.** The vector $\mathbf{v}_j$ defined by

$$\mathbf{v}_j = \mathbf{a}_j - \left(\mathbf{q}_1^{\mathrm{T}} \mathbf{a}_j\right) \mathbf{q}_1 - \left(\mathbf{q}_2^{\mathrm{T}} \mathbf{a}_j\right) \mathbf{q}_2 - \cdots - \left(\mathbf{q}_{j-1}^{\mathrm{T}} \mathbf{a}_j\right) \mathbf{q}_{j-1}$$

  satisfies the required properties.

GS Algorithm

The Gram–Schmidt iteration is outlined below:

$$
\begin{aligned}
\mathbf{q}_1 &= \frac{\mathbf{a}_1}{r_{11}}, \\
\mathbf{q}_2 &= \frac{\mathbf{a}_2 - r_{12}\mathbf{q}_1}{r_{22}}, \\
\mathbf{q}_3 &= \frac{\mathbf{a}_3 - r_{13}\mathbf{q}_1 - r_{23}\mathbf{q}_2}{r_{33}}, \\
&\vdots \\
\mathbf{q}_n &= \frac{\mathbf{a}_n - \sum_{i=1}^{n-1} r_{in}\mathbf{q}_i}{r_{nn}},
\end{aligned}
$$

where

$$
r_{ij} = \begin{cases} \mathbf{q}_i^{\mathrm{T}}\mathbf{a}_j, & \text{if } i \neq j \\ \pm\left\|\mathbf{a}_j - \displaystyle\sum_{k=1}^{j-1} r_{kj}\mathbf{q}_k\right\|_2, & \text{if } i = j \end{cases}.
$$

GS Procedure and Thin QR Factorization

- The GS algorithm, written as a matrix equation, yields a **thin QR factorization**:

$$A = \underbrace{\begin{bmatrix} & & \\ \mathbf{a}_1 & \cdots & \mathbf{a}_n \\ & & \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} & & \\ \mathbf{q}_1 & \cdots & \mathbf{q}_n \\ & & \end{bmatrix}}_{\hat{Q}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & r_{nn} \end{bmatrix}}_{\hat{R}} = \widehat{Q}\widehat{R}$$

- While it is an important tool for theoretical work, the GS algorithm is numerically unstable.