Concept:
zero all subdiagonal entries of each active column with one direct rotation.

Ⓐ

Input an $m\times n$ matrix A_1 and find an $m\times m$ matrix U_1 such that

- $U_1^{T} = U_1^{-1}$ with $\det\left(U_1\right) = 1$
- $A_2 = U_1 A_1$
- $\mathrm{col1}\left(A_2\right) = r_{1,1}\, \vec{e}_1$ where $r_{1,1}$ is a scalar
- $| r_{1,1} | = \ell_1 \quad \left(\text{length}\quad \left(\mathrm{col1}\left(A_1\right)\right)\right)$

In other words, U rotates A_1 so that its first column aligns with $\vec{e}_1$.

If $\mathrm{col1}\left(A_1\right)$ is already a multiple of $\vec{e}_1$,

$U_1 = I$

In this case, the diagonal entry $r_{1,1}$

of the resultant matrix A_2 may be positive, negative or 0.

Otherwise,

①

We define the rotation plane spanned by $\mathrm{col1}\left(A_1\right)$ and $\vec{e}_1$.

②

Construct the second unit vector $\hat{u}_2$ inside the plane such that

- $\hat{u}_2 \cdot \vec{e}_1 = 0$
- $| \hat{u}_2 | = 1$

$\hat{u}_2$ can be obtained by subtracting the projection of col1(A_1) onto $\vec{e}_1$ and normalizing the resultant vector

③

Define an orthonormal basis for the plane:

$$B_1 = [\, \vec{e}_1 \mid \hat{u}_2 \,]$$

④

Compute the angle θ between the direction of col1(A_1) and $\vec{e}_1$ using their normalized dot product

⑤

Extend the plane rotation to a rotation U_1 of $\mathbb{R}^m$

Denote J_1 as the 2×2 rotation by θ: $\begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}$

in the coordinate system described by B_1.

U_1 performs the following geometric actions:

ⓐ leaves every vector perpendicular to the plane described by B_1 unchanged

ⓑ rotates every vector inside the plane by θ using $B_1 J_1 B_1^T$

Any vector $\vec{v}$ can be decomposed as

$$\vec{v} = \vec{v}_{\parallel} + \vec{v}_{\perp}.$$

Since the columns of B_1 are orthonormal,

$$B_1^T B_1 = I_2.$$

The two components are

- $\vec{v}_{\parallel}$ is the projection onto the plane described by B_1:

$$\vec{v}_{\parallel} = B_1 B_1^T \vec{v}$$

- $\vec{v}_{\perp}$ is perpendicular to that plane:

$$\vec{v}_{\perp} = \vec{v} - \vec{v}_{\parallel}$$

$$= \left(I_m - B_1 B_1^T\right)\vec{v}$$

U_1 transforms these two components as follows:

- the component inside the plane is rotated:

$$\vec{v}'_{\parallel} = B_1 J_1 B_1^T \vec{v}$$

- the component perpendicular to the plane is unchanged:

$$\vec{v}'_{\perp} = \vec{v}_{\perp} = \left(I_m - B_1 B_1^T\right) \vec{v}$$

$$\downarrow$$

$$\vec{v}' = \vec{v}'_{\parallel} + \vec{v}'_{\perp} =$$

$$B_1 J_1 B_1^T \vec{v} + \left(I_m - B_1 B_1^T\right) \vec{v}$$

↓

$$U_1 = B_1 J_1 B_1^{\mathsf{T}} + \left(I_m - B_1 B_1^{\mathsf{T}}\right)$$

⑥

Simplify $U = B\,J\,B^{\mathsf{T}} + (I_m - B\,B^{\mathsf{T}})$:

$$U = I_m + B\,J\,B^{\mathsf{T}} - B\,B^{\mathsf{T}} =$$

$$= I_m + B\,(J - I_2)\,B^{\mathsf{T}}$$

Ⓑ

Apply U_1 to A_1:

$$A_2 = U_1 A_1.$$

The first column of A_2 is now complete:

$$\mathrm{col1}\left(A_2\right) = r_{1,1}\,\vec{e}_1.$$

↓

$$A_2 = \left[\begin{array}{c|c} r_{1,1} & \vec{*}^{\mathsf{T}} \\ \hline \vec{0} & \tilde{A}_2 \end{array}\right]$$

The first column has the required upper-triangular form.
It does not need to be changed again.

Deflate A_2 by setting aside its first row and first column.

The remaining active matrix is the trailing submatrix $\tilde{A}_2$, which has dimensions $(m - 1)\times(n - 1)$.

Repeat construction Ⓐ on $\tilde{A}_2$.

This produces an $(m - 1)\times(m - 1)$ rotation $\tilde{U}_2$

that transforms the first column of $\tilde{A}_2$ into a multiple of its first coordinate vector

Embed $\tilde{U}_2$ into an $m\times m$ matrix $U_2 = \left[\begin{array}{c|c} 1 & \vec{0}^T \\ \hline \vec{0} & \tilde{U}_2 \end{array}\right]$

Thus U_2 leaves the first row of A_2 unchanged and acts only on the deflated rows.

Set

$$A_3 = U_2 A_2.$$

Using the block forms of U_2 and A_2:

↓

$$A_3 = \left[\begin{array}{c|c} 1 & \vec{0}^T \\ \hline \vec{0} & \tilde{U}_2 \end{array}\right] \left[\begin{array}{c|c} r_{1,1} & \vec{*}^T \\ \hline \vec{0} & \tilde{A}_2 \end{array}\right] = \left[\begin{array}{c|c} r_{1,1} & \vec{*}^T \\ \hline \vec{0} & \tilde{U}_2 \tilde{A}_2 \end{array}\right]$$

By construction,

the first column of $\tilde{U}_2 \tilde{A}_2$ is a multiple of its first coordinate vector.

↓

Therefore, the first two columns of A_3 have upper-triangular form

Repeat this deflation and rotation process for each remaining active column

- The product of the transposed rotations forms an orthogonal matrix Q
- The final upper-triangular or upper-trapezoidal matrix R satisfies $A_1 = Q\,R$

Optimization, page 1

On this page, m denotes the number of active rows, and we omit the iteration subscripts.

For each column, the rotation matrix was derived as

$$U = I_m + B\,(J - I_2)\,B^{T}.$$

The following quantities have already been defined (not yet computed):

- $\ell = \text{length}\big(\text{col1}(A)\big)$
- $c = \cos(\theta)$
- $s = \sin(\theta)$
- $B = [\ \vec{e}_1 \mid \hat{u}_2\ ]$

Let

$$d = c - 1$$

and

$$D = J - I_2.$$

$$\downarrow$$

$$D = J - I_2 = \left[\begin{array}{c|c} d & s \\ \hline -s & d \end{array}\right]$$

$$\downarrow$$

$$U = I_m + B\,D\,B^T$$

$$B\,D = \left[\begin{array}{c|c} \vec{e}_1 & \hat{u}_2 \end{array}\right]\left[\begin{array}{c|c} d & s \\ \hline -s & d \end{array}\right] = \left[\begin{array}{c|c} d\,\vec{e}_1 - s\,\hat{u}_2 & s\,\vec{e}_1 + d\,\hat{u}_2 \end{array}\right]$$

$$B\,D\,B^T = \left[\begin{array}{c|c} d\,\vec{e}_1 - s\,\hat{u}_2 & s\,\vec{e}_1 + d\,\hat{u}_2 \end{array}\right]\left[\begin{array}{c} \vec{e}_1^{\,T} \\ \hat{u}_2^T \end{array}\right]$$

$$= d\,\vec{e}_1\,\vec{e}_1^{\,T} - s\,\hat{u}_2\,\vec{e}_1^{\,T} + s\,\vec{e}_1\,\hat{u}_2^T + d\,\hat{u}_2\,\hat{u}_2^T.$$

$$\downarrow$$

$$U = I_m + d\,\vec{e}_1\,\vec{e}_1^{\,T} - s\,\hat{u}_2\,\vec{e}_1^{\,T} + s\,\vec{e}_1\,\hat{u}_2^T + d\,\hat{u}_2\,\hat{u}_2^T.$$

Since the first entry of $\hat{u}_2$ is zero, $\hat{u}_2 = \begin{bmatrix} 0 \\ \bar{u}_2 \end{bmatrix}$

where $\bar{u}_2$ contains entries 2 through m of $\hat{u}_2$.

Using the block forms of $\vec{e}_1$ and $\hat{u}_2$, the four outer-product terms affect U as follows:

$$d\,\vec{e}_1\,\vec{e}_1^{\,T} = \left[\begin{array}{c|c} d & \vec{0}^T \\ \hline \vec{0} & 0_{m-1} \end{array}\right]$$

$$s\,\vec{e}_1\,\hat{u}_2^{\,T} = \left[\begin{array}{c|c} 0 & s\,\bar{u}_2^{\,T} \\ \hline \vec{0} & 0_{m-1} \end{array}\right]$$

$$-s\,\hat{u}_2\,\vec{e}_1^{\,T} = \left[\begin{array}{c|c} 0 & \vec{0}^T \\ \hline -s\,\bar{u}_2 & 0_{m-1} \end{array}\right]$$

$$d\,\hat{u}_2\,\hat{u}_2^{\,T} = \left[\begin{array}{c|c} 0 & \vec{0}^T \\ \hline \vec{0} & d\,\bar{u}_2\,\bar{u}_2^{\,T} \end{array}\right]$$

$$1 + d = c$$

$$\downarrow$$

$$U = I_m + d\,\vec{e}_1\,\vec{e}_1^{\,T} - s\,\hat{u}_2\,\vec{e}_1^{\,T} + s\,\vec{e}_1\,\hat{u}_2^{\,T} + d\,\hat{u}_2\,\hat{u}_2^{\,T} = \left[\begin{array}{c|c} c & s\,\bar{u}_2^{\,T} \\ \hline -s\,\bar{u}_2 & I_{m-1} + d\,\bar{u}_2\,\bar{u}_2^{\,T} \end{array}\right]$$

We can now summarize the quantities that must actually be computed.

Let $a = \big(\mathrm{col1}(A)\big)_1$, and let $\bar{p}$ contain entries 2 through m of col1(A). No projection computation is required.

- $\rho^2 = \bar{p}^T \bar{p}$
- $\ell = \sqrt{a^2 + \rho^2}$
- $\ell^{-1} = \dfrac{1}{\ell}$
- $c = a\ \ell^{-1}$
- $\bar{g} = \bar{p}\ \ell^{-1}$
- $h = \dfrac{c - 1}{\rho^2}$

For m active rows, computing these quantities requires

- 2m multiplications
- m additions and subtractions
- 2 divisions
- 1 square root.

Therefore, the block form becomes $U = \left[\begin{array}{c|c} c & \bar{g}^T \\ \hline -\bar{g} & I_{m-1} + h\ \bar{p}\ \bar{p}^T \end{array}\right]$

The quantities c, $\bar{g}$, and h determine the block matrix U.

However, instead of multiplying the quantities to compute U, we will derive the algebraic product U A

The block form derived on the preceding page contains two quantities that can be simplified before they are computed:

- $s\,\bar{u}_2$
- $d\,\bar{u}_2\,\bar{u}_2^T$.

Let $\bar{p}$ contain entries 2 through m of col1(A).
Since subtracting the projection onto $\vec{e}_1$ only removes the first entry of col1(A),

$$\vec{p} = \text{col1}(A) - \big(\text{col1}(A)\big)_1\,\vec{e}_1 = \begin{bmatrix} 0 \\ \bar{p} \end{bmatrix}$$

Let
$\rho = \text{length}(\vec{p})$.

Then

$$\bar{u}_2 = \frac{\bar{p}}{\rho}$$

and

$$s = \frac{\rho}{\ell}.$$

Simplify $s\,\bar{u}_2$:

$$s\,\bar{u}_2 = \left(\frac{\rho}{\ell}\right)\left(\frac{\bar{p}}{\rho}\right)$$

$$= \frac{\bar{p}}{\ell}.$$

Therefore, define

$$\bar{g} = \frac{\bar{p}}{\ell}.$$

Then
$s\,\bar{u}_2 = \bar{g}$,
so neither s nor ρ needs to be computed
for the off-diagonal blocks of U.

②
Simplify $d\,\bar{u}_2\,\bar{u}_2^{\mathsf{T}}$:

$$d\,\bar{u}_2\,\bar{u}_2^{\mathsf{T}}$$

$$= d\left(\frac{\bar{p}}{\rho}\right)\left(\frac{\bar{p}^{\mathsf{T}}}{\rho}\right)$$

$$= \left(\frac{d}{\rho^2}\right)\bar{p}\,\bar{p}^{\mathsf{T}}.$$

Only ρ^2 is required in this expression.
The square root $\rho = \text{length}(\vec{p})$ is not required.

Since

$\rho^2 = \bar{p}^T \bar{p}$,

the same sum of squares used in

$$\ell^2 = \big(\text{col1}(A)\big)_1^{\,2} + \rho^2$$

can be reused directly.

Thus, let

$$h = \frac{d}{\rho^2}.$$

The two simplified quantities are

$s\,\bar{u}_2 = \bar{g}$

and

$d\,\bar{u}_2\,\bar{u}_2^T = h\,\bar{p}\,\bar{p}^T$.

Therefore, U may be written as $\left[\begin{array}{c|c} c & \bar{g}^T \\ \hline -\bar{g} & I_{m-1} + h\,\bar{p}\,\bar{p}^T \end{array}\right]$

where

$$\bar{g} = \frac{\bar{p}}{\ell}$$

and

$$h = \frac{c-1}{\rho^2}.$$

The rotation can therefore be computed using only one square root:

$$\ell = \text{length}\big(\text{col1}(A)\big).$$

Optimization, page 3

Algebraic derivation of U A

This page concerns the general case $\rho^2 > 0$.
In the zero and parallel cases, $U = I_m$ and $U A = A$.

Separate the first column of A from its remaining columns:

$$A = \left[\ \text{col1}(A) \mid C\ \right]$$

The first column of U A is already known from the construction of U:
$$U\,\text{col1}(A) = \ell\,\vec{e}_1$$
↓
Only the remaining columns C must be transformed.

Partition C into its first row and lower block,
and write the first column of A using a and $\bar{p}$:
↓

$$C = \begin{bmatrix} \vec{b}^T \\ C_b \end{bmatrix} \qquad A = \left[\begin{array}{c|c} a & \vec{b}^T \\ \hline \bar{p} & C_b \end{array}\right]$$

$$\downarrow$$

$$U\,C = \left[\begin{array}{c|c} c & \bar{g}^T \\ \hline -\bar{g} & I_{m-1} + h\,\bar{p}\,\bar{p}^T \end{array}\right] \begin{bmatrix} \vec{b}^T \\ C_b \end{bmatrix} = \begin{bmatrix} c\,\vec{b}^T + \bar{g}^T\,C_b \\ -\bar{g}\,\vec{b}^T + C_b + h\,\bar{p}\,\bar{p}^T\,C_b \end{bmatrix}$$

The row vector $\bar{p}^T\,C_b$ is used in both blocks.

Compute it once:

$$\vec{z}^T = \bar{p}^T\,C_b$$

Since $\bar{g} = \bar{p}\,\ell^{-1}$,

$$\bar{g}^T\,C_b = \ell^{-1}\,\vec{z}^T.$$

$$\downarrow$$

$$U\,C = \begin{bmatrix} c\,\vec{b}^T + \ell^{-1}\,\vec{z}^T \\ -\bar{g}\,\vec{b}^T + C_b + h\,\bar{p}\,\vec{z}^T \end{bmatrix} = \begin{bmatrix} c\,\vec{b}^T + \ell^{-1}\,\vec{z}^T \\ C_b + (-\bar{g}\,\vec{b}^T + h\,\bar{p}\,\vec{z}^T) \end{bmatrix}$$

Combining the known first column with U C gives

$$U\,A = \left[\begin{array}{c|c} \ell & c\,\vec{b}^T + \ell^{-1}\,\vec{z}^T \\ \hline \vec{0} & C_b - \bar{g}\,\vec{b}^T + h\,\bar{p}\,\vec{z}^T \end{array}\right]$$

Thus U is never constructed.
The active matrix is updated using one row-vector product

and scaled outer-product terms.

Since $\bar{g} = \bar{p}\,\ell^{-1}$, define
$\vec{w}^{\mathsf{T}} = -\ell^{-1}\,\vec{b}^{\mathsf{T}} + h\,\vec{z}^{\mathsf{T}}$.

Then
$-\bar{g}\,\vec{b}^{\mathsf{T}} + h\,\bar{p}\,\vec{z}^{\mathsf{T}} = \bar{p}\,\vec{w}^{\mathsf{T}}$.
↓

$$U\,C = \begin{bmatrix} c\,\vec{b}^{\mathsf{T}} + \ell^{-1}\,\vec{z}^{\mathsf{T}} \\ C_b + \bar{p}\,\vec{w}^{\mathsf{T}} \end{bmatrix}$$

For an active m×n matrix, computing U C requires
- $(2m + 2)(n - 1)$ multiplications
- $(2m - 1)(n - 1)$ additions and subtractions

Including the quantities computed on the preceding page,
one complete active step requires
- $2m + (2m + 2)(n - 1)$ multiplications
- $m + (2m - 1)(n - 1)$ additions and subtractions
- 2 divisions
- 1 square root.

The first column of U A is now complete.
Deflate U A by setting aside its first row and first column,
and repeat the same construction on the trailing
$(m - 1)\times(n - 1)$ active matrix.

When no active column has entries below its diagonal position,
return the resulting upper-triangular or upper-trapezoidal matrix R
and the accumulated orthogonal matrix Q.

Computation cost: comparison with Givens and Householder methods
(Full explicit Q and R for an n×n matrix)

Method	Givens rotations	Direct plane rotations	Householder reflections
Multiplications	$\approx \left(\frac{10}{3}\right) n^3$	$\left(\frac{5}{3}\right) n^3 + \left(\frac{9}{2}\right) n^2 - \left(\frac{25}{6}\right) n - 2$	$\approx \left(\frac{4}{3}\right) n^3$
Additions and subtractions	$\approx \left(\frac{5}{3}\right) n^3$	$\left(\frac{5}{3}\right) n^3 - \left(\frac{2}{3}\right) n - 1$	$\approx \left(\frac{4}{3}\right) n^3$
Flops (multiplications + additions and subtractions)	$\approx 5n^3$	$\left(\frac{10}{3}\right) n^3 + \left(\frac{9}{2}\right) n^2 - \left(\frac{29}{6}\right) n - 3$	$\approx \left(\frac{8}{3}\right) n^3$
Divisions	$\approx n(n - 1)$	$2(n - 1)$	Implementation-dependent lower-order terms
Square roots	$\frac{n(n - 1)}{2}$	$n - 1$	$n - 1$
Weighted cost	$\approx 5n^3 + 12n(n - 1)$	$\left(\frac{10}{3}\right) n^3 + \left(\frac{9}{2}\right) n^2 + \left(\frac{115}{6}\right) n - 27$	$\geq \left(\frac{8}{3}\right) n^3 + 8(n - 1)$

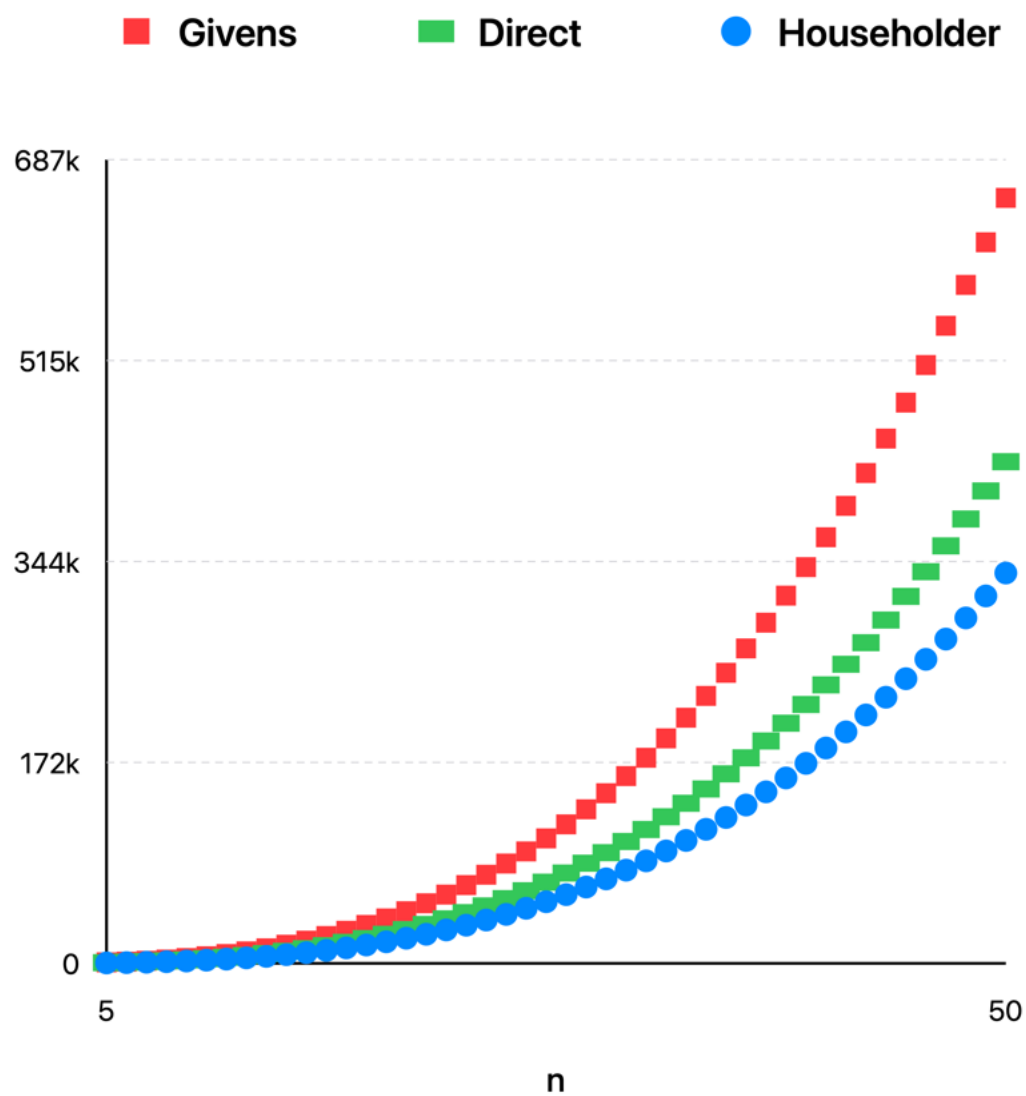
Blue: best cost among the three methods in that operation category.
Green: direct plane rotations improve on Givens rotations, but not on Householder.

Weighted-cost convention:

- additions, subtractions, and multiplications are assigned weight 1
- divisions and square roots are assigned weight 8

These weights are approximate and hardware-dependent.
The plot should be interpreted as a comparative estimate,
not as measured execution time.
The Householder value is a lower bound because its
implementation-dependent division count is not included.

The plot shows estimated weighted arithmetic cost for dense square matrices where
$5 \le n \le 50$



There may be significant opportunities to improve efficiency
by computing some of the quantities simultaneously.
We have not yet implemented these optimizations.

Open questions:
① How does numerical stability compare with Householder
when the target column is close to $\vec{e}_1$?
If we are lucky and the comparison is favorable,
a hybrid algorithm may be useful.

② How does efficiency compare with Givens rotations
for sparse matrices?
If we are lucky and the comparison is favorable,
a hybrid algorithm may also be useful.

③ Is there an advantage to a hybrid algorithm selecting
among Householder, Givens, and direct plane rotations
on a column-by-column basis?

④ Does this simplify computations for rank-1 QR updates?

⑤ Are there any other algebraic shortcuts that we have not implemented?

AI assistance:
AI tools were used extensively for information gathering,
checking references, and assisting with operation counts.
All mathematical claims, formulas, visualizations, and final
editorial decisions were reviewed and accepted by the authors.

LA
