instability that affects both the modified and classical Gram–Schmidt algorithms. In floating point arithmetic, these algorithms may produce vectors q_j that are far from orthogonal. The loss of orthogonality occurs when A is close to rank-deficient, and, like most instabilities, it can appear even in low dimensions.

Starting on paper rather than in MATLAB, consider the case of a matrix

$$A = \begin{bmatrix} 0.70000 & 0.70711 \\ 0.70001 & 0.70711 \end{bmatrix} \tag{9.1}$$

on a computer that rounds all computed results to five digits of relative accuracy (Lecture 13). The classical and modified algorithms are identical in the 2×2 case. At step $j = 1$, the first column is normalized, yielding

$$r_{11} = 0.98996, \qquad q_1 = a_1/r_{11} = \begin{bmatrix} 0.70000/0.98996 \\ 0.70001/0.98996 \end{bmatrix} = \begin{bmatrix} 0.70710 \\ 0.70711 \end{bmatrix}$$

in five-digit arithmetic. At step $j = 2$, the component of a_2 in the direction of q_1 is computed and subtracted out:

$$r_{12} = q_1^* a_2 = 0.70710 \times 0.70711 + 0.70711 \times 0.70711 = 1.0000,$$

$$v_2 = a_2 - r_{12} q_1 = \begin{bmatrix} 0.70711 \\ 0.70711 \end{bmatrix} - \begin{bmatrix} 0.70710 \\ 0.70711 \end{bmatrix} = \begin{bmatrix} 0.00001 \\ 0.00000 \end{bmatrix},$$

again with rounding to five digits. This computed v_2 is dominated by errors. The final computed Q is

$$Q = \begin{bmatrix} 0.70710 & 1.0000 \\ 0.70711 & 0.0000 \end{bmatrix},$$

which is not close to any orthogonal matrix.

On a computer with sixteen-digit precision, we still lose about five digits of orthogonality if we apply modified Gram–Schmidt to the matrix (9.1). Here is the MATLAB evidence. The "eye" function generates the identity of the indicated dimension.

```
A = [.70000 .70711               Define A.
     .70001 .70711];
[Q,R] = qr(A);                   Compute factor Q by Householder.
   norm(Q'*Q-eye(2))                Test orthogonality of Q.
[Q,R] = mgs(A);                  Compute factor Q by modified G–S.
   norm(Q'*Q-eye(2))                Test orthogonality of Q.
```

The lines without semicolons produce the following printed output:

```
ans = 2.3515e-16,     ans = 2.3014e-11.
```