

NAG Toolbox for MATLAB

Chapter Introduction

F05 – Orthogonalisation

Contents

1	Scope of the Chapter	2
2	Background to the Problems	2
2.1	Gram–Schmidt Orthogonalisation	2
2.2	Householder Orthogonalisation	2
3	Recommendations on Choice and Use of Available Functions	2
4	References	3

1 Scope of the Chapter

This chapter is concerned with the orthogonalisation of vectors in a finite dimensional space.

2 Background to the Problems

Let a_1, a_2, \dots, a_n be a set of n linearly independent vectors in m -dimensional space; $m \geq n$.

We wish to construct a set of n vectors q_1, q_2, \dots, q_n such that:

- the vectors $\{q_i\}$ form an orthonormal set; that is, $q_i^T q_j = 0$ for $i \neq j$, and $\|q_i\|_2 = 1$;
- each a_i is linearly dependent on the set $\{q_i\}$.

2.1 Gram–Schmidt Orthogonalisation

The classical Gram–Schmidt orthogonalisation process is described in many textbooks; see for example Chapter 5 of Golub and Van Loan 1996.

It constructs the orthonormal set progressively. Suppose it has computed orthonormal vectors q_1, q_2, \dots, q_k which orthogonalise the first k vectors a_1, a_2, \dots, a_k . It then uses a_{k+1} to compute q_{k+1} as follows:

$$\begin{aligned} z_{k+1} &= a_{k+1} - \sum_{i=1}^k (q_i^T a_{k+1}) q_i \\ q_{k+1} &= z_{k+1} / \|z_{k+1}\|_2. \end{aligned}$$

In finite precision computation, this process can result in a set of vectors $\{q_i\}$ which are far from being orthogonal. This is caused by $|z_{k+1}|$ being small compared with $|a_{k+1}|$. If this situation is detected, it can be remedied by reorthogonalising the computed q_{k+1} against q_1, q_2, \dots, q_k , that is, repeating the process with the computed q_{k+1} instead of a_{k+1} . See Danial *et al.* 1976.

2.2 Householder Orthogonalisation

An alternative approach to orthogonalising a set of vectors is based on the QR factorization (see the F08 Chapter Introduction), which is usually performed by Householder's method. See Chapter 5 of Golub and Van Loan 1996.

Let A be the m by n matrix whose columns are the n vectors to be orthogonalised. The QR factorization gives

$$A = QR$$

where R is an n by n upper triangular matrix and Q is an m by n matrix, whose columns are the required orthonormal set.

Moreover, for any k such that $1 \leq k \leq n$, the first k columns of Q are an orthonormal basis for the first k columns of A .

Householder's method requires twice as much work as the Gram–Schmidt method, provided that no reorthogonalisation is required in the latter. However, it has satisfactory numerical properties and yields vectors which are close to orthogonality even when the original vectors a_i are close to being linearly dependent.

3 Recommendations on Choice and Use of Available Functions

The single function in this chapter, f05aa, uses the Gram–Schmidt method, with reorthogonalisation to ensure that the computed vectors are close to being exactly orthogonal. This method is only available for real vectors.

To apply Householder's method, you must use functions in Chapter F08:

for real vectors: f08ae, followed by f08af

for complex vectors: f08as, followed by f08at

The example programs for f08ae or f08as illustrate the necessary calls to these functions.

4 References

Daniel J W, Gragg W B, Kaufman L and Stewart G W 1976 Reorthogonalization and stable algorithms for updating the Gram–Schmidt *QR* factorization *Math. Comput.* **30** 772–795

Golub G H and Van Loan C F 1996 *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore
