Foundations of Scientific Computing

Brad Baxter
Department of Economics, Mathematics and Statistics
Birkbeck College, University of London
Malet Street, London W1CE 7HX
b.baxter@bbk.ac.uk

CONTENTS

Introduction

While a lecturer at Imperial College (1995–2001), I taught the yearly course in introductory theoretical numerical analysis (M2N1), the class being mostly composed of second and third year mathematics undergraduates. These students had passed a very basic course in linear algebra, but still needed lots of introductory matrix algebra. Further, their only programming experience was some very basic Maple; a fine package, but poorly suited to the algorithms discussed here. The ideal language would be Matlab, but since this was unavailable at the time, I have left algorithms in pseudo-code form.

A more extensive version of these notes is being turned into a book, but there have been many requests for these lecture notes. I shall, of course, correct errors, but further development on this version has now ceased. You are free to distribute these notes provided they are unchanged in any way;

1. Applied Linear Algebra

1.1. Fundamentals

By definition, a $p \times q$ matrix has p rows and q columns, and we let $\mathbb{R}^{p\times q}$ denote the set of all $p \times q$ matrices with real elements. If $A \in \mathbb{R}^{p\times q}$, then we let A_{jk} denote the element of A in the jth row and kth column, and sometimes call this the (j,k)th element, or component, of A. Thus

$$A = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1q} \\ A_{21} & A_{22} & \dots & A_{2q} \\ \\ \vdots & \vdots & \vdots & \vdots \\ \\ A_{p1} & A_{p2} & \dots & A_{pq} \end{pmatrix}. \tag{1.1}$$

The *transpose* A^T of a matrix $A \in \mathbb{R}^{p\times q}$ is the $q \times p$ matrix defined by

$$(A^T)_{jk} = A_{kj}, \qquad \text{for } 1 \le k \le p, \quad 1 \le j \le q. \tag{1.2}$$

and, if $M \in \mathbb{R}^{n\times n}$ satisfies $M = M^T$, then we say that M is a *symmetric* matrix; if $M^T = -M$, then we call M *skew-symmetric*. A *column vector* is simply a $n \times 1$ matrix, whilst a *row vector* is a $1 \times n$ matrix.

Example 1.1. $(A + B)^T = A^T + B^T$, because

$$(A + B)^T_{jk} = (A + B)_{kj} = A_{kj} + B_{kj} = A^T_{jk} + B^T_{jk}.$$

Exercise 1.1. Let $C \in \mathbb{R}^{n\times n}$ be any matrix. Show that $(C + C^T)/2$ is a symmetric matrix, whilst $(C - C^T)/2$ is skew-symmetric. Prove that every square matrix A can be written as $A = S + T$, where S is skew-symmetric and T is symmetric. Is this decomposition unique?

Vector Notation: For this section, we shall usually write vectors (row and column) in boldface (in these notes) or underlined (in script). However, in later sections, we shall follow the modern convention of advanced work and *not* use boldface for vectors.

Thus we write

$$\mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \in \mathbb{R}^n \equiv \mathbb{R}^{n\times 1}, \tag{1.3}$$

and

$$\boldsymbol{v}^T = \begin{pmatrix} v_1 & v_2 & \dots & v_n \end{pmatrix} \in \mathbb{R}^{1\times n}. \tag{1.4}$$

We shall say that a vector $\boldsymbol{v}$ is *nonzero*, written $\boldsymbol{v} \neq 0$, if $\boldsymbol{v}$ is not the zero vector.

If $A \in \mathbb{R}^{p\times q}$ and $B \in \mathbb{R}^{q\times r}$, then their *matrix product* $AB \in \mathbb{R}^{p\times r}$ is defined by the equation

$$(AB)_{jk} = \sum_{\ell=1}^{q} A_{j\ell} B_{\ell k}, \qquad 1 \le j \le p, \quad 1 \le k \le r. \tag{1.5}$$

The $n \times n$ identity matrix I_n is defined in the usual way

$$I_n = \begin{pmatrix} 1 & 0 & 0 & \dots & & 0 \\ 0 & 1 & 0 & \dots & & \\ 0 & 0 & 1 & \dots & & \\ & & & \ddots & & \vdots \\ & & & & 1 & 0 \\ 0 & & & \dots & 0 & 1 \end{pmatrix} \in \mathbb{R}^{n\times n}. \tag{1.6}$$

If the size n of the identity matrix is clear from context, then we shall sometimes omit the subscript, writing I for I_n. It's also very useful to reserve a notation for the columns of the identity matrix. Unless stated otherwise, we shall always use the notations

$$\boldsymbol{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \boldsymbol{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \dots, \boldsymbol{e}_n = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}. \tag{1.7}$$

Further, if $A \in \mathbb{R}^{n\times n}$ and there exists a matrix $A^{-1} \in \mathbb{R}^{n\times n}$ such that $AA^{-1} = A^{-1}A = I_n$, then we say that A is *invertible*; another name is *nonsingular*.

Example 1.2. Let $A \in \mathbb{R}^{p\times q}$ and $B \in \mathbb{R}^{q\times r}$. Then $(AB)^T = B^T A^T \in \mathbb{R}^{r\times p}$. Indeed, we have

$$(AB)^T_{jk} = (AB)_{kj} = \sum_{\ell=1}^{q} A_{k\ell} B_{\ell j} = \sum_{\ell=1}^{q} B^T_{j\ell} A^T_{\ell k} = (B^T A^T)_{jk}. \tag{1.8}$$

Now show that $(P\boldsymbol{x})^T AP\boldsymbol{x} = \boldsymbol{x}^T P^T AP\boldsymbol{x}$, where P is $n \times n$ and $\boldsymbol{x} \in \mathbb{R}^n$.

Exercise 1.2. Let $\boldsymbol{a} \in \mathbb{R}^n$. Show that

$$\boldsymbol{a}^T \boldsymbol{a} = \sum_{k=1}^{n} a_k^2,$$

but $\boldsymbol{a}\boldsymbol{a}^T$ is the $n \times n$ matrix whose elements are given by the formula

$$(\boldsymbol{a}\boldsymbol{a}^T)_{jk} = a_j a_k, \qquad 1 \le j, k \le n. \tag{1.9}$$

More generally, given any two vectors $\boldsymbol{u} \in \mathbb{R}^p$ and $\boldsymbol{v} \in \mathbb{R}^q$, the matrix $\boldsymbol{u}\boldsymbol{v}^T \in \mathbb{R}^{p \times q}$ is called their *outer product* and has elements

$$(\boldsymbol{u}\boldsymbol{v}^T)_{jk} = u_j v_k, \qquad 1 \leq j \leq p, 1 \leq k \leq q.$$

Furthermore, notice that $(\boldsymbol{u}\boldsymbol{v}^T)\boldsymbol{x} = \boldsymbol{u}(\boldsymbol{v}^T\boldsymbol{x})$, so that such a linear map takes every vector to a multiple of the single vector $\boldsymbol{u}$. In other words, the *rank* of $\boldsymbol{u}\boldsymbol{v}^T$ is one, which just means that the image of this matrix is one-dimensional.

Exercise 1.3. Find the eigenvectors and eigenvalues of $A = \boldsymbol{u}\boldsymbol{v}^T$, where $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$. In other words, find n vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ and n numbers $\lambda_1, \ldots, \lambda_n$ for which $A\boldsymbol{v}_j = \lambda_j \boldsymbol{v}_j$.

Exercise 1.4. Prove that $A(BC) = (AB)C$.

Example 1.3. This example will be rather useful later. Let $A \in \mathbb{R}^{n \times n}$ and let $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$. Then

$$\boldsymbol{x}^T A \boldsymbol{y} = \sum_{j=1}^{n} \sum_{k=1}^{n} A_{jk} x_j y_k.$$

To show this, we just need to notice that

$$\begin{aligned} \boldsymbol{x}^T A \boldsymbol{y} &= \boldsymbol{x}^T (A\boldsymbol{y}) \\ &= \sum_{j=1}^{n} x_j (A\boldsymbol{y})_j \\ &= \sum_{j=1}^{n} x_j \sum_{k=1}^{n} A_{jk} y_k \\ &= \sum_{j=1}^{n} \sum_{k=1}^{n} A_{jk} x_j y_k, \end{aligned}$$

and we can swap the order of summation because we're only dealing with finite sums.

Exercise 1.5. Let $A \in \mathbb{R}^{n \times n}$. Show that $A_{jk} = \boldsymbol{e}_j^T A \boldsymbol{e}_k$.

Exercise 1.6. Let $\boldsymbol{e}_1 = (\,1 \quad 0\,)^T$, $\boldsymbol{e}_2 = (\,0 \quad 1\,)^T$ and $\boldsymbol{a} = (\,a_1 \quad a_2\,)^T$ be vectors in the plane $\mathbb{R}^2$. Calculate the matrices $P_1 = I - \boldsymbol{e}_1 \boldsymbol{e}_1^T$, $P_2 = I - \boldsymbol{e}_2 \boldsymbol{e}_2^T$ and the vectors $P_1 \boldsymbol{a}$, $P_2 \boldsymbol{a}$. It should come as no surprise that P_1, P_2 are called *projection matrices*.

Exercise 1.7. Let $\boldsymbol{w} \in \mathbb{R}^n$ be a nonzero vector and define $P \in \mathbb{R}^{n \times n}$ by

$$P = I_n - \frac{\boldsymbol{w}\boldsymbol{w}^T}{\boldsymbol{w}^T \boldsymbol{w}}.$$

Show that $P^T = P$ and prove that $P\boldsymbol{v}$ is orthogonal to $\boldsymbol{w}$ for every $\boldsymbol{v} \in \mathbb{R}^n$.

Exercise 1.8. For any $n \times n$ matrix M, we define trace $M = M_{11} + M_{22} + \cdots + M_{nn}$. Prove that trace $AB =$ trace BA, even when $AB \neq BA$.

Often it is useful to have a brief notation with which to manipulate the rows and columns of a matrix. Accordingly, we write

$$A = \begin{pmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 & \dots & \boldsymbol{a}_q \end{pmatrix}, \tag{1.10}$$

to indicate that the $p \times q$ matrix A has columns $\boldsymbol{a}_1, \dots, \boldsymbol{a}_q \in \mathbb{R}^p$. Similarly, we write

$$A = \begin{pmatrix} \alpha_1^T \\ \alpha_2^T \\ \vdots \\ \alpha_p^T \end{pmatrix}, \tag{1.11}$$

to indicate that the rows of A are precisely the row vectors $\alpha_1^T, \dots, \alpha_p^T$ in $\mathbb{R}^{1\times q}$.

Example 1.4. We can write $I_n = \begin{pmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \cdots & \boldsymbol{e}_n \end{pmatrix}$.

Exercise 1.9. Let $A = \begin{pmatrix} \boldsymbol{a}_1 & \cdots & \boldsymbol{a}_n \end{pmatrix} \in \mathbb{R}^{m\times n}$. Show that $\boldsymbol{a}_j = A\boldsymbol{e}_j$, for $1 \leq j \leq n$.

Exercise 1.10. Let $A \in \mathbb{R}^{p\times q}$ and let $B = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \cdots & \boldsymbol{b}_r \end{pmatrix} \in \mathbb{R}^{q\times r}$. Show that

$$AB = \begin{pmatrix} A\boldsymbol{b}_1 & A\boldsymbol{b}_2 & \cdots & A\boldsymbol{b}_r \end{pmatrix}.$$

Exercise 1.11. If $A = \begin{pmatrix} \boldsymbol{a}_1 & \cdots & \boldsymbol{a}_n \end{pmatrix} \in \mathbb{R}^{m\times n}$, show that

$$A^T = \begin{pmatrix} \boldsymbol{a}_1^T \\ \vdots \\ \boldsymbol{a}_n^T \end{pmatrix} \in \mathbb{R}^{n\times m}.$$

Exercise 1.12. Let

$$A = \begin{pmatrix} \boldsymbol{a}_1^T \\ \boldsymbol{a}_2^T \\ \vdots \\ \boldsymbol{a}_p^T \end{pmatrix} \in \mathbb{R}^{p\times q}$$

and $B = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \cdots & \boldsymbol{b}_r \end{pmatrix} \in \mathbb{R}^{q\times r}$. Show

$$(AB)_{jk} = \boldsymbol{a}_j^T \boldsymbol{b}_k, \qquad \text{for } 1 \leq j \leq p, 1 \leq k \leq r.$$

Exercise 1.13. Let $A = \begin{pmatrix} \boldsymbol{a}_1 & \cdots & \boldsymbol{a}_n \end{pmatrix} \in \mathbb{R}^{n\times n}$. Show that

$$(A^T A)_{jk} = \boldsymbol{a}_j^T \boldsymbol{a}_k, \qquad 1 \leq j, k \leq n.$$

Exercise 1.14. Let $P = (\boldsymbol{p}_1 \quad \boldsymbol{p}_2 \quad \cdots \quad \boldsymbol{p}_m) \in \mathbb{R}^{\ell \times m}$,

$$Q = \begin{pmatrix} \boldsymbol{q}_1^T \\ \boldsymbol{q}_2^T \\ \vdots \\ \boldsymbol{q}_m^T \end{pmatrix} \in \mathbb{R}^{m \times n}.$$

Show that

$$PQ = \sum_{k=1}^{m} \boldsymbol{p}_k \boldsymbol{q}_k^T.$$

As a special case of this exercise, now let

$$L = (\boldsymbol{\ell}_1 \quad \boldsymbol{\ell}_2 \quad \cdots \quad \boldsymbol{\ell}_n),$$

so that

$$L^T = \begin{pmatrix} \boldsymbol{\ell}_1^T \\ \boldsymbol{\ell}_2^T \\ \vdots \\ \boldsymbol{\ell}_n^T \end{pmatrix},$$

and show that

$$LL^T = \sum_{k=1}^{n} \boldsymbol{\ell}_k \boldsymbol{\ell}_k^T.$$

This is also a good time to review the notion of invertibility. The matrix

$$A = (\boldsymbol{a}_1 \quad \boldsymbol{a}_2 \quad \cdots \quad \boldsymbol{a}_n) \in \mathbb{R}^{n \times n}$$

is invertible if and only if its columns are linearly independent. In other words, the square matrix A is invertible if and only if $A\boldsymbol{x} = 0$ implies $\boldsymbol{x} = 0$. Further, and to jog your memory, the vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m \in \mathbb{R}^n$ are said to be *linearly independent* if the equation $\sum_{k=1}^m c_k \boldsymbol{v}_k = 0$ implies $c_1 = c_2 = \cdots = c_m = 0$; they're *linearly dependent* if we can find $c_1, \ldots, c_m$, not all zero, such that $\sum_{k=1}^m c_k \boldsymbol{v}_k = 0$. In other words, no vector belonging to a linearly independent set of vectors can be written as a linear combination of other vectors in the set. Alternatively, setting $V = (\boldsymbol{v}_1 \quad \cdots \quad \boldsymbol{v}_m)$, the vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$ are linearly independent if and only if there's no nonzero vector $\boldsymbol{c} \in \mathbb{R}^m$ for which $V\boldsymbol{c} = 0$.

We shall also be learning how to solve sets of linear equations. Let's begin with some rather simple, but fundamental, cases involving *triangular matrices*.

Definition 1.1. A matrix $M \in \mathbb{R}^{p \times q}$ is *upper triangular* if $M_{jk} = 0$ when $j > k$. It's *lower triangular* if $M_{jk} = 0$ for $k > j$.

Example 1.5.

$$U = \begin{pmatrix} 1 & 2 \\ 0 & -3 \\ 0 & 0 \end{pmatrix}$$

is a 3×2 upper triangular matrix, whilst

$$L = \begin{pmatrix} -4 & 0 & 0 \\ 0 & 7 & 0 \\ 2 & -1 & 5 \end{pmatrix}$$

is lower triangular.

It's simple to solve triangular linear systems. Here's the *back substitution* algorithm for upper triangular systems:

Algorithm 1.1. Given any upper triangular matrix $U \in \mathbb{R}^{n \times n}$ such that $U_{jj} \neq 0$, for $1 \leq j \leq n$, and given any vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, we solve $U\boldsymbol{x} = \boldsymbol{y}$ by first setting $x_n = y_n / U_{nn}$. Then, for $k = n-1, n-2, \ldots, 2, 1$, we let

$$x_k = \Big(y_k - \sum_{\ell=k+1}^{n} U_{k\ell} x_\ell \Big) / U_{kk}.$$

Exercise 1.15. Solve the upper triangular linear system

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$$

using back substitution.

Exercise 1.16. Calculate the number of arithmetic operations required to perform back substitution when applied to an $n \times n$ matrix.

Exercise 1.17. Prove that an upper triangular matrix is invertible if and only if its diagonal elements $\{U_{jj} : 1 \leq j \leq n\}$ are all nonzero.

Exercise 1.18. There's a very similar *forward substitution* algorithm for lower triangular systems. Describe it.

Exercise 1.19. Prove that the inverse of an invertible upper (or lower) triangular $n \times n$ matrix is also upper (or lower) triangular.

It's often useful to give only the *order* of arithmetic operations required for an algorithm. The notation is best shown by some examples: we write $\mathcal{O}(n)$ for an operation count of $p_0 + p_1 n$, $\mathcal{O}(n^2)$ for a count of $p_0 + p_1 n + p_2 n^2$, and so forth. Thus back and forward substitution require $\mathcal{O}(n^2)$ operations or *flops* (floating point operations).

Exercise 1.20. Show that a matrix–vector multiplication requires, in general, $\mathcal{O}(n^2)$ operations, whilst a matrix–matrix multiplication requires $\mathcal{O}(n^3)$. Optional: If Gaussian elimination was covered in last year's courses, show that it requires $\mathcal{O}(n^3)$ operations to solve n linear equations in n unknowns.

Most computation is done using the subprocessor dedicated to mathematical functions incorporated into almost all modern processors, such as the Pentium family. This performs the basic operations of arithmetic (addition, subtraction, multiplication and division) and provides some extra functions (at least squareroot, exponential, logarithm and trigonometric functions). All calculations are then performed to an accuracy of about 16 decimal places (computers use binary arithmetic, so the precise decimal precision achieved varies slightly). For example, a Pentium 400 MHz machine, costing about £500, will provide about 8×10^7 basic arithmetic operations per second; to use one common jargon, it achieves 80 Mflops.

Exercise 1.21. Assume a computer achieves 100 Mflops and that we're using an algorithm that requires $n^3/6$ operations when applied to an $n \times n$ matrix. Estimate the time required for $n = 10^k$, $1 \leq k \leq 7$. You may find it useful to know that one day contains 86400 seconds, whilst one year contains about 3×10^7 seconds. One or two significant figures will be sufficient.

The field of *computational complexity* is a cousin of numerical analysis and is devoted to studying the costs of algorithms of all kinds. The algorithms we have considered are usually described as having *polynomial complexity*, in the sense that, when the size of the problem is n, then the cost is $\mathcal{O}(n^m)$ for some fixed positive real number m. Unfortunately, there are some algorithms in graph theory and combinatorics that seem to have an operation count of $\mathcal{O}(a^n)$ operations, for some number $a > 1$. In other words, they seem to extort a cost that grows exponentially with n. This is far worse than polynomial complexity, because even a modest increase in n can take far too long. Repeat the previous exercise assuming $a = 2$ to see this. (I may discuss complexity in my `M2OD` course next term.)

It's important to understand that a little thought can save lots of time. For example, suppose we need to form matrix–vector products using the matrix

$$A = I - \boldsymbol{w}\boldsymbol{w}^T \in \mathbb{R}^{n \times n}. \tag{1.12}$$

Thus

$$A\boldsymbol{v} = \boldsymbol{v} - (\boldsymbol{w}^T\boldsymbol{v})\boldsymbol{w}. \tag{1.13}$$

Forming A using (1.12) and then calculating the components $\{\sum_{k=1}^n A_{jk}v_k : 1 \leq j \leq\}$ requires $\mathcal{O}(n^2)$ operations, but using (1.13) requires only $\mathcal{O}(n)$ operations. Consider the saving achieved here for $n = 10^6$, for instance. To ram this point home, for a 100 Mflop machine, this is the difference between

0.01s and roughly 2.8 hours. Since such matrices occur in many graphics programs, this is important.

Exercise 1.22. Let $A, B \in \mathbb{R}^{n\times n}$ and $\boldsymbol{x} \in \mathbb{R}^n$. Show that calculating $(AB)\boldsymbol{x}$ requires $\mathcal{O}(n^3)$ operations, whereas $A(B\boldsymbol{x})$ takes $\mathcal{O}(n^2)$.

It's easy to see that multiplying two $n \times n$ matrices in the obvious way requires $\mathcal{O}(n^3)$ operations. However, there exist some unusual algorithms that can improve on this figure, allowing matrix multiplication in $\mathcal{O}(n^{2+\delta})$ operations, where the smallest known value of δ is 0.376, due to Coppersmith and Winograd (1990). Unfortunately, their algorithm requires n to be astronomically large before the operation count beats $\mathcal{O}(n^3)$, and is not a practical alternative in its current form. However, there's no known reason why faster algorithms could not exist, $\mathcal{O}(n^2 \log n)$ say, and such a breakthrough would be dramatic.

1.2. The Gram-Schmidt algorithm

You have already met the *dot product*, or *scalar product*, of two vectors $\boldsymbol{u}, \boldsymbol{v}$ in $\mathbb{R}^n$:

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \boldsymbol{u}^T \boldsymbol{v} = \sum_{k=1}^{n} u_k v_k. \tag{1.14}$$

We shall call this an *inner product* in this course, and the concept will be greatly generalized.

Exercise 1.23. Prove that $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \langle \boldsymbol{v}, \boldsymbol{u} \rangle$,

$$\langle \alpha_1 \boldsymbol{u}_1 + \alpha_2 \boldsymbol{u}_2, \boldsymbol{v} \rangle = \alpha_1 \langle \boldsymbol{u}_1, \boldsymbol{v} \rangle + \alpha_2 \langle \boldsymbol{u}_2, \boldsymbol{v} \rangle$$

and

$$\langle \boldsymbol{u}, \beta_1 \boldsymbol{v}_1 + \beta_2 \boldsymbol{v}_2 \rangle = \beta_1 \langle \boldsymbol{u}, \boldsymbol{v}_1 \rangle + \beta_2 \langle \boldsymbol{u}, \boldsymbol{v}_2 \rangle,$$

where $\alpha_1, \alpha_2, \beta_1, \beta_2$ are real numbers.

Example 1.6. If $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0$, then

$$\begin{aligned} \|\boldsymbol{u} + \boldsymbol{v}\|^2 &= \langle \boldsymbol{u} + \boldsymbol{v}, \boldsymbol{u} + \boldsymbol{v} \rangle \\ &= \langle \boldsymbol{u}, \boldsymbol{u} \rangle + \langle \boldsymbol{u}, \boldsymbol{v} \rangle + \langle \boldsymbol{v}, \boldsymbol{u} \rangle + \langle \boldsymbol{v}, \boldsymbol{v} \rangle \\ &= \|\boldsymbol{u}\|^2 + \|\boldsymbol{v}\|^2, \end{aligned}$$

which is, of course, Pythagoras' theorem.

Exercise 1.24. Show that $\|\boldsymbol{a} + \boldsymbol{b}\|^2 = \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 + 2\langle \boldsymbol{a}, \boldsymbol{b} \rangle$.

You should also recall that the vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_m \in \mathbb{R}^n$ are called *orthogonal* if $\langle \boldsymbol{a}_j, \boldsymbol{a}_k \rangle = 0$ when $j \neq k$. Further, the vectors are said to be *orthonormal* if they're orthogonal and every vector has unit length, that is, $\langle \boldsymbol{a}_k, \boldsymbol{a}_k \rangle = 1$, for $1 \leq k \leq m$.

Definition 1.2. Kronecker delta notation There is a highly useful shorthand for describing a set of orthonormal vectors. We introduce a new symbol δ_{jk}, called the *Kronecker delta*, defined by

$$\delta_{jk} = \begin{cases} 1 & j = k, \\ 0 & j \neq k. \end{cases} \tag{1.15}$$

The Kronecker delta will be used throughout this course, and will be useful to you in many other courses also. For example, the vectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$ are orthonormal if and only if

$$\langle \boldsymbol{a}_j, \boldsymbol{a}_k \rangle = \delta_{jk}, \qquad \text{for } 1 \leq j, k \leq m.$$

As a second example, the identity matrix I can be defined by

$$I_{jk} = \delta_{jk}.$$

[As a frivolous aside, L. Kronecker is probably best known today for this notation and the quote: "God created the natural numbers; all the rest is the work of Man".]

Exercise 1.25. Later in the course, we'll see inner products applied to vector spaces of *functions*, and this example will be relevant then. Let $e_j(t) = \exp(ijt)$, for any integer j, where $i = \sqrt{-1}$. Show that

$$(2\pi)^{-1} \int_{-\pi}^{\pi} e_j(t) \overline{e_k(t)}\, dt = \delta_{jk}, \qquad j, k \in \mathbb{Z},$$

where $\overline{e_k(t)}$ is the complex conjugate of $e_k(t)$, that is, $\overline{e_k(t)} = \exp(-ikt)$.

Exercise 1.26. Let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m \in \mathbb{R}^n$ be nonzero orthogonal vectors. Why can we assume the inequality $m \leq n$? Prove that orthogonal vectors are linearly independent: if $\sum_{k=1}^{m} c_k \boldsymbol{a}_k = 0$, then $c_1 = c_2 = \cdots = c_m = 0$.

The Gram-Schmidt algorithm is a simple, but extremely important, technique for constructing orthonomal vectors from any set of linearly independent vectors. We shall use a notation for the *length* of a vector that may be slightly different from that introduced last year: the length, or *norm* of a vector $\boldsymbol{a} \in \mathbb{R}^n$ is defined by the equation

$$\|\boldsymbol{a}\| = \sqrt{\langle \boldsymbol{a}, \boldsymbol{a} \rangle}. \tag{1.16}$$

Of course, this is just another way of saying that

$$\|\boldsymbol{a}\| = \left(a_1^2 + a_2^2 + \cdots + a_n^2 \right)^{1/2},$$

but definition (1.16) allows us to generalize the notion of length when we generalize the notion of inner product, as we shall do in this course.

Now for the algorithm:

Algorithm 1.2. (Gram-Schmidt) Let $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_m \in \mathbb{R}^n$ be any set of

linearly independent vectors. We begin by setting

$$\boldsymbol{q}_1 = \boldsymbol{a}_1/\|\boldsymbol{a}_1\|. \tag{1.17}$$

Then, for $k = 2, 3, \ldots, m$, we let

$$\boldsymbol{v}_k = \boldsymbol{a}_k - \sum_{\ell=1}^{k-1} \langle \boldsymbol{a}_k, \boldsymbol{q}_\ell \rangle \boldsymbol{q}_\ell. \tag{1.18}$$

This vector need not have unit norm, so we normalize its length by defining

$$\boldsymbol{q}_k = \boldsymbol{v}_k/\|\boldsymbol{v}_k\|.$$

Then $\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_m$ is a set of orthonormal vectors for the subspace spanned by the vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_m$.

Exercise 1.27. Prove that $\boldsymbol{v}_k$ is orthogonal to $\boldsymbol{v}_j$ *and* $\boldsymbol{a}_j$ for $1 \le j < k$. Apply the algorithm to the vectors

$$\boldsymbol{a}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_3 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}.$$

Example 1.7. Warning: This simple form of Gram-Schmidt can generate enormous errors in practical computation. For instance, the Hilbert matrix $H^{(n)} \in \mathbb{R}^{n \times n}$ has elements

$$H^{(n)}_{jk} = \frac{1}{j + k - 1}, \qquad 1 \le j, k \le n.$$

If we apply the Gram-Schmidt algorithm above, generating vectors $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_n$ that *should* be orthonormal, then, setting $n = 8$ and $Q = (\boldsymbol{q}_1 \quad \cdots \quad \boldsymbol{q}_n)$, we obtain

$$Q^T Q = \begin{pmatrix} 1 & 2.2_{-17} & -5.8_{-15} & 1.1_{-13} & -2.6_{-12} & 3.6_{-11} & 1.4_{-09} & 1_{-09} \\ 2.2_{-17} & 1 & -1_{-15} & 1.3_{-14} & -6.8_{-13} & 3.7_{-11} & -1.2_{-09} & -1.2_{-09} \\ -5.8_{-15} & -1_{-15} & 1 & 3.6_{-12} & -9.2_{-11} & 1.8_{-09} & -3.6_{-08} & -3.5_{-08} \\ 1.1_{-13} & 1.3_{-14} & 3.6_{-12} & 1 & -3.4_{-09} & 1.4_{-07} & -4.4_{-06} & -4.1_{-06} \\ -2.6_{-12} & -6.8_{-13} & -9.2_{-11} & -3.4_{-09} & 1 & 7.6_{-06} & -0.00049 & -0.00045 \\ 3.6_{-11} & 3.7_{-11} & 1.8_{-09} & 1.4_{-07} & 7.6_{-06} & 1 & -0.04 & -0.032 \\ 1.4_{-09} & -1.2_{-09} & -3.6_{-08} & -4.4_{-06} & -0.00049 & -0.04 & 1 & 1 \\ 1_{-09} & -1.2_{-09} & -3.5_{-08} & -4.1_{-06} & -0.00045 & -0.032 & 1 & 1 \end{pmatrix},$$

which matrix was computed using 16 decimal places of accuracy, of which only 2 significant figures were displayed for clarity. (I've used the notation 2.2_{-17} for 2.2×10^{-17} to save space.) Since $(Q^T Q)_{jk} = \boldsymbol{q}_j^T \boldsymbol{q}_k$, the above matrix should be the identity! Matters worsen for even slightly larger values of n.

In fairness to Gram-Schmidt, Hilbert matrices, although seemingly innocuous, are horrid creatures for many methods. However, my point in this example is that Gram-Schmidt, in the basic form given here, is really not trustworthy in finite precision arithmetic.

You will see the Gram-Schmidt algorithm many times in this course.

1.3. The QR factorization

There is another way to describe the Gram-Schmidt algorithm that turns out to be highly useful. Let's begin with an exercise.

Exercise 1.28. Let $Q = (\, \boldsymbol{q}_1 \quad \boldsymbol{q}_2 \quad \cdots \quad \boldsymbol{q}_m \,) \in \mathbb{R}^{m \times m}$ be any matrix, and let $R \in \mathbb{R}^{m \times n}$ be an upper triangular matrix. Show that

$$(QR)\boldsymbol{e}_k = \sum_{\ell=1}^{k} r_{\ell k} \boldsymbol{q}_\ell, \qquad 1 \leq k \leq n.$$

Suppose we apply the Gram-Schmidt algorithm to n vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n \in \mathbb{R}^m$, where $m \geq n$. We obtain the equations

$$\boldsymbol{a}_1 = \|\boldsymbol{a}_1\| \boldsymbol{q}_1 \quad \text{and} \quad \boldsymbol{a}_k = \sum_{\ell=1}^{k-1} \langle \boldsymbol{a}_k, \boldsymbol{q}_\ell \rangle \boldsymbol{q}_\ell + \|\boldsymbol{v}_k\| \boldsymbol{q}_k, \qquad 2 \leq k \leq m.$$

In other words, we have

$$\boldsymbol{a}_k = \sum_{\ell=1}^{k} r_{\ell k} \boldsymbol{q}_\ell,$$

where the vectors $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_n$ are orthonormal. Hence the last exercise yields the *matrix factorization*

$$A = QR, \tag{1.19}$$

where $Q \in \mathbb{R}^{m \times m}$ has orthonormal columns, its first n columns being $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_n$, and $R \in \mathbb{R}^{m \times n}$ is upper triangular. The importance of this restatement is that, once the factorization is the recognized aim, we can use other methods for its calculation.

Now, since the matrix $Q = (\, \boldsymbol{q}_1 \quad \boldsymbol{q}_2 \quad \cdots \quad \boldsymbol{q}_m \,) \in \mathbb{R}^{m \times m}$ has orthonormal columns, we have

$$(Q^T Q)_{jk} = \boldsymbol{q}_j^T \boldsymbol{q}_k = \delta_{jk}, \qquad 1 \leq j, k \leq m.$$

In other words, $Q^T Q = I_m$, or $Q^{-1} = Q^T$, for any matrix whose columns are orthonormal. The class of such matrices is extremely important and is the subject of our next definition.

Definition 1.3. The matrix $Q \in \mathbb{R}^{m \times m}$ is called *orthogonal* if $Q^T Q = I_m$.

Proposition 1.1. Orthogonal matrices preserve length and angle: If $\boldsymbol{v}, \boldsymbol{w} \in \mathbb{R}^m$ and $Q \in \mathbb{R}^{m\times m}$ is orthogonal, then

$$\langle Q\boldsymbol{v}, Q\boldsymbol{w}\rangle = \langle \boldsymbol{v}, \boldsymbol{w}\rangle \quad \text{and} \quad \|Q\boldsymbol{v}\| = \|\boldsymbol{v}\|.$$

Proof. The equation $Q^TQ = I_m$ implies

$$\langle Q\boldsymbol{v}, Q\boldsymbol{w}\rangle = (Q\boldsymbol{v})^T(Q\boldsymbol{w}) = \boldsymbol{v}^TQ^TQ\boldsymbol{w} = \boldsymbol{v}^T\boldsymbol{w} = \langle \boldsymbol{v}, \boldsymbol{w}\rangle,$$

and we obtain the second relation by setting $\boldsymbol{w} = \boldsymbol{v}$. □

Definition 1.4. If $A \in \mathbb{R}^{m\times n}$ and $A = QR$, where $Q \in \mathbb{R}^{m\times m}$ is an orthogonal matrix and $R \in \mathbb{R}^{m\times n}$ is upper triangular, then we say that we have a *QR factorization* of A.

Exercise 1.29. Let $\theta \in \mathbb{R}$. Prove that the matrix

$$Q = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

is orthogonal.

Exercise 1.30. Let $U_1, U_2 \in \mathbb{R}^{n\times n}$ be orthogonal matrices. Prove that U_1U_2 is also orthogonal.

Exercise 1.31. Let $\boldsymbol{v} \in \mathbb{R}^n$ be any nonzero vector. Prove that the matrix

$$\rho = I - 2\frac{\boldsymbol{v}\boldsymbol{v}^T}{\boldsymbol{v}^T\boldsymbol{v}}$$

is orthogonal. Interpret ρ geometrically when $n = 2$ and $\boldsymbol{v} = (\,1 \quad 0\,)^T$, and hence give its general geometric interpretation.

Exercise 1.32. Prove that the square matrix

$$Q = \begin{pmatrix} 1 & & & & & & & & & \\ & \ddots & & & & & & & & \\ & & 1 & & & & & & & \\ & & & \cos\theta & & & & -\sin\theta & & \\ & & & & 1 & & & & & \\ & & & & & \ddots & & & & \\ & & & & & & 1 & & & \\ & & & \sin\theta & & & & \cos\theta & & \\ & & & & & & & & 1 & \\ & & & & & & & & & \ddots \\ & & & & & & & & & & 1 \end{pmatrix} \tag{1.20}$$

is orthogonal, where $\theta \in \mathbb{R}$, and the trigonometric functions occur in rows and columns p and q.

The rotation appearing in the previous exercise only affects the subspace spanned by $\{\boldsymbol{e}_p, \boldsymbol{e}_q\}$, and otherwise acts as the identity. In other words, if we form the matrix product QA, then Q only affects rows p and q of A, the other rows remaining unchanged. Such a matrix is extremely useful and is usually called a *Givens rotation* (Givens pioneered the use of these orthogonal matrices in the 1950s). We shall write $G_{pq}(\theta)$ to denote the Givens rotation of (1.20).

Specifically, let A be $n \times n$ and let

$$\tilde{Q} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \in \mathbb{R}^{2\times 2}.$$

Then rows p and q of the matrix A, that is

$$\begin{pmatrix} a_{p1} & a_{p2} & \cdots & a_{pn} \\ a_{q1} & a_{q2} & \cdots & a_{qn} \end{pmatrix},$$

which we shall temporarily regard as n vectors

$$\begin{pmatrix} a_{p1} \\ a_{q1} \end{pmatrix}, \begin{pmatrix} a_{p2} \\ a_{q2} \end{pmatrix}, \ldots, \begin{pmatrix} a_{pn} \\ a_{qn} \end{pmatrix}$$

in the plane $\mathbb{R}^2$, are mapped to the vectors

$$\tilde{Q}\begin{pmatrix} a_{p1} \\ a_{q1} \end{pmatrix}, \tilde{Q}\begin{pmatrix} a_{p2} \\ a_{q2} \end{pmatrix}, \ldots, \tilde{Q}\begin{pmatrix} a_{pn} \\ a_{qn} \end{pmatrix},$$

the other rows of A being unaffected.

Example 1.8. Check the following calculations: We shall calculate a QR factorization of the 3×2 matrix

$$A = \begin{pmatrix} 3 & 65 \\ 4 & 0 \\ 12 & 13 \end{pmatrix}. \tag{1.21}$$

If we choose $\cos\theta = 3/5$, $\sin\theta = -4/5$, then

$$G_{12}(\theta) = \begin{pmatrix} 3/5 & 4/5 & 0 \\ -4/5 & 3/5 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

and

$$A^{(1)} = G_{12}(\theta)A = \begin{pmatrix} 5 & 39 \\ 0 & -52 \\ 12 & 13 \end{pmatrix}.$$

Further, setting $\cos\phi = 5/13$, $\sin\phi = -12/13$ and

$$G_{13}(\phi) = \begin{pmatrix} 5/13 & 0 & 12/13 \\ 0 & 1 & 0 \\ -12/13 & 0 & 5/13 \end{pmatrix},$$

we obtain

$$A^{(2)} = G_{13}(\phi)A^{(1)} = \begin{pmatrix} 13 & 27 \\ 0 & -52 \\ 0 & -31 \end{pmatrix}.$$

Finally, we use

$$G_{23}(\psi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -52/\sqrt{3665} & -31/\sqrt{3665} \\ 0 & 31/\sqrt{3665} & -52/\sqrt{3665} \end{pmatrix},$$

whence

$$R = G_{23}(\psi)A^{(2)} = \begin{pmatrix} 13 & 27 \\ 0 & \sqrt{3665} \\ 0 & 0 \end{pmatrix}.$$

Exercise 1.33. Calculate the QR factorization of the 2×2 matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$.

What's the point of the QR factorization? One application is to solve linear systems. Indeed, let $A \in \mathbb{R}^{n\times n}$ be any square matrix and suppose we want to solve the linear system $A\boldsymbol{x} = \boldsymbol{y}$. One method is to apply the Givens rotations in the order specified above to both the matrix A and the vector $\boldsymbol{y}$, obtaining

$$R\boldsymbol{x} = \boldsymbol{z},$$

and then solve the resulting system by back substitution.

However, we can use the factorization to achieve far more. Consider the linear system

$$A\boldsymbol{x} = \boldsymbol{y},$$

where $A \in \mathbb{R}^{m\times n}$ and $m > n$. In other words, we have more equations than unknowns and, in general, there will be *no* solutions to the linear system. However, if we apply Givens rotations to both sides, then we obtain

$$R\boldsymbol{x} = \boldsymbol{z},$$

where $R \in \mathbb{R}^{m\times n}$ is upper triangular and $\boldsymbol{z} \in \mathbb{R}^m$. We can then easily solve for $x_1, \ldots, x_n$ using the first n equations only, using back substitution. But, what is the significance of this vector, given that we know the solution does not, in general, exist? We shall deal with this problem in our section on "Least Squares".

Exercise 1.34. Use the technique suggested in the last paragraph to "solve" the inconsistent linear equations.

$$A\boldsymbol{x} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

where A is given by equation (1.21). (You will soon see that this is a *least squares* solution of the linear system.)

1.4. The Cauchy-Schwarz inequality

You already know that, for any vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$, we have

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = \|\boldsymbol{a}\| \|\boldsymbol{b}\| \cos\theta,$$

where θ is the angle between the two vectors. This implies the inequality

$$|\langle \boldsymbol{a}, \boldsymbol{b} \rangle| \leq \|\boldsymbol{a}\| \|\boldsymbol{b}\|,$$

with equality if and only if the vectors are linearly dependent (in other words, $\cos\theta = \pm 1$). This inequality is true in much greater generality.

Theorem 1.2. Let $\boldsymbol{a}, \boldsymbol{b}$ be any vectors in $\mathbb{R}^n$. Then

$$|\langle \boldsymbol{a}, \boldsymbol{b} \rangle| \leq \|\boldsymbol{a}\| \|\boldsymbol{b}\|,$$

with equality if and only if the vectors are linearly dependent.

Proof. The inequality is obvious if one of the vectors is the zero vector, so we shall assume that $\boldsymbol{a}$ is not the zero vector. Following the first step of the Gram-Schmidt algorithm, we set $\boldsymbol{q} = \boldsymbol{a}/\|\boldsymbol{a}\|$ and define

$$\boldsymbol{c} = \boldsymbol{b} - \langle \boldsymbol{q}, \boldsymbol{b} \rangle \boldsymbol{q}.$$

Then $\|\boldsymbol{c}\| \geq 0$, with equality if and only if $\boldsymbol{c}$ is the zero vector, which can occur if and only if the vectors $\boldsymbol{a}, \boldsymbol{b}$ are linearly dependent. But then

$$\begin{aligned} 0 &\leq \|\boldsymbol{c}\|^2 \qquad (1.22) \\ &= \Big\langle \boldsymbol{b} - \langle \boldsymbol{q}, \boldsymbol{b} \rangle \boldsymbol{q}, \boldsymbol{b} - \langle \boldsymbol{q}, \boldsymbol{b} \rangle \boldsymbol{q} \Big\rangle \\ &= \|\boldsymbol{b}\|^2 - (\langle \boldsymbol{q}, \boldsymbol{b} \rangle)^2 \\ &= \|\boldsymbol{b}\|^2 - (\langle \boldsymbol{a}, \boldsymbol{b} \rangle)^2 / \|\boldsymbol{a}\|^2, \qquad (1.23) \end{aligned}$$

which completes the proof. □

Exercise 1.35. Use the Cauchy-Schwarz inequality to prove that, for any real numbers $x_1, x_2, \ldots, x_n$, we have the inequality

$$\Big(x_1 + \cdots + x_n\Big)^2 \leq n\Big(x_1^2 + \cdots + x_n^2\Big).$$

Exercise 1.36. Use Cauchy-Schwarz to prove that, for any vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^n$

$$(\|\boldsymbol{a}\| - \|\boldsymbol{b}\|)^2 \leq \|\boldsymbol{a} + \boldsymbol{b}\|^2 \leq (\|\boldsymbol{a}\| + \|\boldsymbol{b}\|)^2 .$$

These inequalities are usually called the triangle inequalities. Interpret them geometrically when $n = 2$.

1.5. The Gradient

If $f(x)$ is a function of one variable, then you should be familiar with the Taylor expansion

$$f(a + h) = f(a) + f'(a)h + \frac{1}{2}f''(a)h^2 + \mathcal{O}(h^3). \tag{1.24}$$

Thus we can approximate $f(x)$ by the quadratic polynomial

$$p(x) = f(a) + f'(a)(x - a) + \frac{1}{2}f''(a)(x - a)^2$$

near the point $x = a$. If we're rather close to $x = a$, then we might only need the linear approximation

$$\ell(x) = f(a) + f'(a)(x - a).$$

Exercise 1.37. Use these linear and quadratic polynomials to calculate two approximations to $\sqrt{65}$ and $\sqrt{220}$.

But what happens if f is a function defined on $\mathbb{R}^n$? For example, $f(\boldsymbol{x})$ might be the temperature at the point $\boldsymbol{x} \in \mathbb{R}^3$. As a second example, to emphasize that applications do not require $n \leq 3$, $f(\boldsymbol{x})$ might be the value of a financial contract depending on financial variables $x_1, x_2, \ldots, x_{100}$, such as interest rates and exchange rates for various currencies.

In fact, the n-dimensional form of Taylor's theorem is rather nice: if $f(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^n$, is a function of n variables, then

$$f(\boldsymbol{a} + \boldsymbol{h}) = f(\boldsymbol{a}) + \nabla f(\boldsymbol{a})^T \boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^T D^2 f(\boldsymbol{a})\boldsymbol{h} + \mathcal{O}(\|\boldsymbol{h}\|^3), \tag{1.25}$$

where the *gradient vector* is given by

$$(\nabla f(\boldsymbol{a}))_j = \frac{\partial f}{\partial x_j}(\boldsymbol{a}), \qquad 1 \leq j \leq n, \tag{1.26}$$

and the *second derivative matrix*, or *Hessian* matrix, is defined by

$$D^2 f(\boldsymbol{a})_{jk} = \frac{\partial^2 f}{\partial x_j \partial x_k}(\boldsymbol{a}), \qquad 1 \leq j, k \leq n. \tag{1.27}$$

Assumption: In this course, we shall always assume that

$$\frac{\partial^2 f}{\partial x_j \partial x_k} = \frac{\partial^2 f}{\partial x_k \partial x_j},$$

which implies that our Hessian matrices are always symmetric.

WARNING: A common error is to confuse the symbols ∂ and δ, so that a large minority of students write, say, $\delta f/\delta x$, believing this to mean $\partial f/\partial x$. *This is not so!* Any student making this error will be heavily penalized in M2N1.

Exercise 1.38. If $f(\boldsymbol{x}) = x_p$, for some $p \in \{1, 2, \ldots, n\}$, show that $\nabla f(\boldsymbol{x}) = \boldsymbol{e}_p$.

Example 1.9. If $f(x, y) = \sin x \sin y$, then

$$f(x+h, y+k) = f(x, y)+h\cos x \sin y+k \sin x \cos y-\frac{1}{2}(h^2+k^2)\sin x \sin y+hk \cos x \cos y+\cdots.$$

Exercise 1.39. Repeat this example when $f(x_1, x_2, \ldots, x_n) = \sin x_1 \sin x_2 \cdots \sin x_n$ and $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^2$.

You can use (1.25) without knowing its proof, which is not examined in this course. However, whilst I don't have time to present a rigorous treatment, the following sketch should provide some food for thought.

The multivariate form of Taylor's expansion
This material is not examinable.

For a univariate function $f(x)$, $x \in \mathbb{R}$, we have the Taylor expansion

$$f(a + h) = \sum_{m=0}^{\infty} \frac{h^m}{m!} f^{(m)}(a).$$

If we write $D \equiv \frac{d}{dx}$, so that $D^m f(z) \equiv f^{(m)}(z)$, then the Taylor expansion takes the suggestive form

$$f(a + h) = \sum_{m=0}^{\infty} \frac{(hD)^m}{m!} f(a),$$

which should immediately remind you of the series $\exp z = \sum_{m=0}^{\infty} z^m/m!$. Although a rigorous treatment of this resemblance is far beyond the scope of this course, we shall briefly forget the pedantry of the Twentieth Century and emulate the more relaxed manipulations of our Eighteenth Century ancestors. Thus we shall write our Taylor expansion in the form

$$f(a + h) = e^{hD} f(a).$$

If we now consider a function $f(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^n$, of n variables, then (why?) we find that

$$f(\boldsymbol{a} + \boldsymbol{h}) = e^{h_1\partial_1} \cdots e^{h_n\partial_n} f(\boldsymbol{a}),$$

where $\partial_j \equiv \partial/\partial x_j$. Thus we obtain

$$f(\boldsymbol{a} + \boldsymbol{h}) \quad = \quad e^{h_1\partial_1} \cdots e^{h_n\partial_n} f(\boldsymbol{a})$$

$$\begin{aligned} &= e^{h_1\partial_1+\cdots+h_n\partial_n} f(\boldsymbol{a}) \\ &= \sum_{m=0}^{\infty} \frac{(h_1\partial_1+\cdots+h_n\partial_n)^m}{m!} f(\boldsymbol{a}) \\ &= f(\boldsymbol{a}) + \Big(h_1\frac{\partial f}{\partial x_1}(\boldsymbol{a}) + \cdots + h_n\frac{\partial f}{\partial x_n}(\boldsymbol{a})\Big) + \frac{1}{2}\Big(h_1\partial_1+\cdots+h_n\partial_n\Big)^2 f(\boldsymbol{a}) + \cdots. \end{aligned}$$

Now for the third displayed term, using Example 1.3,

$$\begin{aligned} \Big(h_1\partial_1+\cdots+h_n\partial_n\Big)^2 f(\boldsymbol{a}) &= \sum_{j=1}^{n}\sum_{k=1}^{n} h_j h_k \partial_j\partial_k f(\boldsymbol{a}) \\ &= \sum_{j=1}^{n}\sum_{k=1}^{n} h_j h_k \frac{\partial^2 f}{\partial x_j \partial x_k}(\boldsymbol{a}) \\ &= \boldsymbol{h}^T D^2 f(\boldsymbol{a}) \boldsymbol{h}, \end{aligned}$$

as required.

By analogy with the univariate case, we call the function

$$\ell(\boldsymbol{x}) = f(\boldsymbol{a}) + \nabla f(\boldsymbol{a})^T(\boldsymbol{x}-\boldsymbol{a})$$

a linear polynomial in n variables, or simply a linear function, whilst

$$p(\boldsymbol{x}) = f(\boldsymbol{a}) + \nabla f(\boldsymbol{a})^T(\boldsymbol{x}-\boldsymbol{a}) + \frac{1}{2}(\boldsymbol{x}-\boldsymbol{a})^T D^2 f(\boldsymbol{a})(\boldsymbol{x}-\boldsymbol{a})$$

is termed a quadratic in n variables, or simply a quadratic.

Exercise 1.40. Let

$$f(\boldsymbol{x}) = \boldsymbol{a}^T\boldsymbol{x}, \qquad \boldsymbol{x}\in\mathbb{R}^n.$$

Show that $\nabla f(\boldsymbol{x}) = \boldsymbol{a}$.

Hessian matrices can be a little harder to compute.

Example 1.10. Let

$$f(\boldsymbol{x}) = \boldsymbol{x}^T A\boldsymbol{x}, \qquad \boldsymbol{x}\in\mathbb{R}^n,$$

where $A\in\mathbb{R}^{n\times n}$ is a symmetric matrix. Then, using Example 1.3,

$$f(\boldsymbol{x}) = \sum_{j=1}^{n}\sum_{k=1}^{n} A_{jk}x_j x_k,$$

which implies

$$\frac{\partial f}{\partial x_p} = \sum_{j=1}^{n}\sum_{k=1}^{n} A_{jk}\frac{\partial}{\partial x_p}(x_j x_k).$$

Now

$$\frac{\partial x_\ell}{\partial x_p} = \delta_{\ell p},$$

which yields

$$\begin{aligned}\sum_{j=1}^{n}\sum_{k=1}^{n} A_{jk}\frac{\partial}{\partial x_p}(x_j x_k) &= \sum_{j=1}^{n}\sum_{k=1}^{n} A_{jk}\left(\delta_{jp}x_k + x_j\delta_{kp}\right) \\ &= \sum_{k=1}^{n} A_{pk}x_k + \sum_{j=1}^{n} A_{jp}x_j \\ &= (A\boldsymbol{x})_p + (A^T\boldsymbol{x})_p.\end{aligned}$$

Since A is symmetric, we obtain

$$\nabla f(\boldsymbol{x}) = 2A\boldsymbol{x}.$$

As for the second derivative matrix, or Hessian matrix $D^2 f(\boldsymbol{x})$, we note that

$$\begin{aligned}\frac{\partial}{\partial x_q}\left(\frac{\partial f}{\partial x_p}\right) &= \frac{\partial}{\partial x_q}\left(2\sum_{k=1}^{n} A_{pk}x_k\right) \\ &= 2\sum_{k=1}^{n} A_{pk}\delta_{kq} \\ &= 2A_{pq}.\end{aligned}$$

Thus $D^2 f(\boldsymbol{x}) = 2A$, for all $\boldsymbol{x} \in \mathbb{R}^n$.

Exercise 1.41. Let A be a symmetric $n \times n$ matrix and define

$$g(\boldsymbol{x}) = a + \boldsymbol{b}^T\boldsymbol{x} + \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x}, \qquad \boldsymbol{x} \in \mathbb{R}^n.$$

Show that

$$\nabla g(\boldsymbol{x}) = \boldsymbol{b} + A\boldsymbol{x}.$$

Exercise 1.42. Let $h(\boldsymbol{x}) = \exp(\boldsymbol{x}^T A\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^n$, where $A \in \mathbb{R}^{n\times n}$. Compute $\nabla h(\boldsymbol{x})$.

One extremely important example of the above mathematical ideas occurs when our task is to minimize a function of n variables. We say that the function $f(\boldsymbol{x})$ has a *local minimum* at $\boldsymbol{x} = \boldsymbol{a}$ if

$$f(\boldsymbol{a} + h\boldsymbol{u}) \geq f(\boldsymbol{a}), \quad \text{for every unit vector } \boldsymbol{u}, \tag{1.28}$$

when $h > 0$ is sufficiently small. More rigorously, there exists $\epsilon > 0$ such that (1.28) holds for $0 \leq h \leq \epsilon$.

Exercise 1.43. Define local maximum for a function of n variables.

Of course, when $n = 1$, you should already be aware that $f(x)$ has a local minimum at $x = a$ if $f'(a) = 0$ and $f''(a) > 0$. We now consider the corresponding conditions for $n > 1$.

Example 1.11. The condition $f''(a) > 0$ is sufficient, but *not* necessary. For example, $f(x) = x^4$ has a local minimum at $x = 0$, but $f''(0) = 0$.

Proposition 1.3. If $\nabla f(\boldsymbol{a}) \neq 0$, then $f(\boldsymbol{x})$ does *not* have a local minimum, or maximum, at $\boldsymbol{x} = \boldsymbol{a}$.

Proof. Consider the linear approximation

$$\ell(\boldsymbol{x}) = f(\boldsymbol{a}) + \boldsymbol{g}^T(\boldsymbol{x} - \boldsymbol{a})$$

to $f(\boldsymbol{x})$, where $\boldsymbol{g} = \nabla f(\boldsymbol{a})$. Thus, for any $h > 0$ and any unit vector $\boldsymbol{u} \in \mathbb{R}^n$, we have

$$\ell(\boldsymbol{a} + h\boldsymbol{u}) = f(\boldsymbol{a}) + h\boldsymbol{g}^T\boldsymbol{u}.$$

Setting $\boldsymbol{u} = -\boldsymbol{g}/\|\boldsymbol{g}\|$, we obtain

$$\ell(\boldsymbol{a} + h\boldsymbol{u}) = f(\boldsymbol{a}) - h\|\boldsymbol{g}\| < f(\boldsymbol{a}).$$

Since $f(\boldsymbol{a}+h\boldsymbol{u}) = \ell(\boldsymbol{a}+h\boldsymbol{u})+\mathcal{O}(h^2)$, we deduce the inequality $f(\boldsymbol{a}+h\boldsymbol{u}) < f(\boldsymbol{a})$, for all sufficiently small positive h. Hence $f(\boldsymbol{x})$ does not possess a local minimum at $\boldsymbol{x} = \boldsymbol{a}$. If we choose $\boldsymbol{u} = \boldsymbol{g}$, then a similar argument shows that no local maximum occurs. □

Thus $\nabla f(\boldsymbol{a}) = 0$ is a necessary condition that $f(\boldsymbol{x})$ possess a local minimum or maximum at $\boldsymbol{x} = \boldsymbol{a}$. As in the univariate case, the necessary condition for a local minimum involves the second derivative.

Proposition 1.4. If $\nabla f(\boldsymbol{a}) = 0$ and the Hessian matrix $D^2 f(\boldsymbol{a})$ satisfies the inequality

$$\boldsymbol{w}^T D^2 f(\boldsymbol{a})\boldsymbol{w} > 0,$$

for every vector $\boldsymbol{w} \in \mathbb{R}^n$ that's not the zero vector, then $f(\boldsymbol{x})$ has a local minimum at $\boldsymbol{x} = \boldsymbol{a}$.

Proof. Using the quadratic approximation

$$p(\boldsymbol{x}) = f(\boldsymbol{a}) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{a})^T D^2 f(\boldsymbol{a})(\boldsymbol{x} - \boldsymbol{a})$$

we obtain, for $h > 0$ and any unit vector $\boldsymbol{u} \in \mathbb{R}^n$,

$$p(\boldsymbol{a} + h\boldsymbol{u}) = f(\boldsymbol{a}) + \frac{1}{2}h^2\boldsymbol{u}^T D^2 f(\boldsymbol{a})\boldsymbol{u}.$$

Hence

$$p(\boldsymbol{a} + h\boldsymbol{u}) > f(\boldsymbol{a}),$$

which implies that

$$f(\boldsymbol{a} + h\boldsymbol{u}) > f(\boldsymbol{a}),$$

for all sufficiently small h. □

Example 1.12. Let $f(\boldsymbol{x}) = x_1^2 - 2x_1 + 1 + x_2^2 - 2x_2 + 1$, where $\boldsymbol{x} = (\, x_1 \quad x_2 \,)^T$, so the only stationary point is at $\boldsymbol{x} = \boldsymbol{a} = (\, 1 \quad 1 \,)^T$. Now the Hessian matrix $D^2 f(\boldsymbol{x}) = 2I$ (check this!), so that the last proposition implies that $f(\boldsymbol{x})$ has a local minimum at $\boldsymbol{x} = \boldsymbol{a}$.

Exercise 1.44. Construct a function $f(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^n$, $n > 1$, for which $\nabla f(\boldsymbol{a}) = 0$, $D^2 f(\boldsymbol{a}) = 0$, and $f(\boldsymbol{x})$ has a local minimum at $\boldsymbol{x} = \boldsymbol{a}$.

Lots of applications generate Hessian matrices at local minima, so the condition required above has its own name and is studied in the next section.

Definition 1.5. The matrix $A \in \mathbb{R}^{n \times n}$ is *non-negative definite* if

$$\boldsymbol{x}^T A \boldsymbol{x} \geq 0 \tag{1.29}$$

for every vector $\boldsymbol{x} \in \mathbb{R}^n$. We say that A is *positive definite* if inequality (1.29) is strict when $\boldsymbol{x}$ is not the zero vector, that is

$$\boldsymbol{x}^T A \boldsymbol{x} > 0 \qquad \text{when } \boldsymbol{x} \neq 0.$$

In particular, the last proposition can be restated: If $\nabla f(\boldsymbol{a}) = 0$ and the Hessian matrix $D^2 f(\boldsymbol{a})$ is symmetric positive definite, then $f(\boldsymbol{x})$ has a local minimum at $\boldsymbol{x} = \boldsymbol{a}$.

Example 1.13. The matrix

$$A = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

is non-negative definite, because

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}^T A \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1^2 + x_2^2 - 2x_1 x_2 = (x_1 - x_2)^2 \geq 0.$$

It's not positive definite, since $\boldsymbol{e}^T A \boldsymbol{e} = 0$, where $\boldsymbol{e} = (\, 1 \quad 1 \,)^T$.

Exercise 1.45. Show that the matrix

$$A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$$

is positive definite.

Exercise 1.46. Show that the symmetric matrix

$$A = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$$

is positive definite if and only if $a > 0$ and $ac - b^2 > 0$. (Hint: first prove that

$$\begin{pmatrix} x \\ y \end{pmatrix}^T A \begin{pmatrix} x \\ y \end{pmatrix} = cy^2 + a\Big[(x + by/a)^2 - b^2y^2/a^2\Big].$$

and expand.)

1.6. Inner products revisited and positive definite matrices

Let $A \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. We can use A to generate a new definition of the length of a vector:

$$\|\boldsymbol{x}\|_A = \sqrt{\boldsymbol{x}^T A \boldsymbol{x}}. \tag{1.30}$$

We can also use A to generalize the idea of inner product, by defining

$$\langle \boldsymbol{u}, \boldsymbol{v}\rangle_A = \boldsymbol{u}^T A \boldsymbol{v}, \tag{1.31}$$

and we shall call this the *inner product induced by A*.

Exercise 1.47. Let $A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$, $\boldsymbol{e}_1 = (\,1 \quad 0\,)^T$, $\boldsymbol{e}_2 = (\,0 \quad 1\,)^T$. Calculate $\langle \boldsymbol{e}_1, \boldsymbol{e}_2\rangle_A$, $\|\boldsymbol{e}_1\|_A$ and $\|\boldsymbol{e}_2\|_A$.

Exercise 1.48. Prove that an $n \times n$ symmetric positive definite matrix is invertible.

Our original notion of inner product is now the special case $A = I$, All of the ideas you've met for inner products generalize very easily to this more general definition. You'll have to trust me that this generalization is useful, and not some otiose abstraction. The Cauchy-Schwarz inequality also holds for this more general notion of inner product.

Theorem 1.5. Let $\boldsymbol{a}, \boldsymbol{b}$ be any vectors in $\mathbb{R}^n$ and let $A \in \mathbb{R}^{n\times n}$ be any symmetric positive definite matrix. Then

$$|\langle \boldsymbol{a}, \boldsymbol{b}\rangle_A| \le \|\boldsymbol{a}\|_A \|\boldsymbol{b}\|_A,$$

with equality if and only if the vectors are linearly dependent.

Proof. We simply replace $\langle \cdot, \cdot \rangle$ by $\langle \cdot, \cdot \rangle_A$ and $\|\cdot\|$ by $\|\cdot\|_A$ in the proof of Theorem 1.2. □

Exercise 1.49. Let

$$A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Prove that A is positive definite. Apply the Gram-Schmidt algorithm using the inner product induced by A to the vectors $\boldsymbol{e}_1 = (\,1 \quad 0 \quad 0\,)^T$, $\boldsymbol{e}_2 = (\,0 \quad 1 \quad 0\,)^T$ and $\boldsymbol{e}_3 = (\,0 \quad 0 \quad 1\,)^T$.

There's a very simple way to generate symmetric positive definite matrices. Let $P = (\boldsymbol{p}_1 \quad \boldsymbol{p}_2 \quad \cdots \quad \boldsymbol{p}_n) \in \mathbb{R}^{n \times n}$ be any invertible matrix, which just means that the columns $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n$ are linearly independent. In particular, $P\boldsymbol{x} = 0$ implies $\boldsymbol{x} = 0$. Then the matrix $A = P^T P$ satisfies

$$\boldsymbol{x}^T A \boldsymbol{x} = \boldsymbol{x}^T P^T P \boldsymbol{x} = (P\boldsymbol{x})^T (P\boldsymbol{x}) = \|P\boldsymbol{x}\|^2 \geq 0,$$

with equality if and only if $P\boldsymbol{x} = 0$, that is, if and only if $\boldsymbol{x} = 0$.

Exercise 1.50. Let $P \in \mathbb{R}^{n \times n}$ be any invertible matrix and let $M = PP^T$. Is M symmetric positive definite? Give an example of a symmetric 2×2 matrix that is not positive definite.

In fact, *every* symmetric positive definite matrix arises in this way. Furthermore, the proof is extremely important, because it introduces the *Cholesky factorization*.

Theorem 1.6. Let $A \in \mathbb{R}^{n \times n}$ be any symmetric positive definite matrix. Then there exists an invertible matrix $P \in \mathbb{R}^{n \times n}$ such that $A = P^T P$. Furthermore, we can choose P to be upper triangular, in which case we say that $A = P^T P$ is a Cholesky factorization of A.

Proof. Let $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n$ be any n linearly independent vectors in $\mathbb{R}^n$. Using the inner product

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle_A = \boldsymbol{a}^T A \boldsymbol{b}$$

induced by A, we apply the Gram-Schmidt algorithm to generate n vectors $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n$ satisfying

$$\langle \boldsymbol{u}_j, \boldsymbol{u}_k \rangle_A = \delta_{jk}, \qquad 1 \leq j, k \leq n.$$

In other words, setting $U = (\boldsymbol{u}_1 \quad \boldsymbol{u}_2 \quad \cdots \quad \boldsymbol{u}_n) \in \mathbb{R}^{n \times n}$, we have

$$U^T A U = I_n.$$

Since U is invertible (Exercise: Why?), we can set $P = U^{-1}$, whence $A = P^T P$.

If we choose $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n$ to be the columns of the identity matrix, that is, $\boldsymbol{v}_1 = \boldsymbol{e}_1, \ldots, \boldsymbol{v}_n = \boldsymbol{e}_n$, then U is upper triangular. Hence P is also upper triangular. □

Theorem 1.6 characterizes the symmetric positive definite matrices in a rather straightforward way that allows us to deduce some more useful properties with ease.

Proposition 1.7. Let $A \in \mathbb{R}^{n \times n}$ be a symmetric positive definite matrix. Then

$$A_{kk} > 0, \qquad \text{for } 1 \leq k \leq n,$$

and

$$|A_{jk}| < \sqrt{A_{jj}A_{kk}}, \qquad \text{when } j \neq k.$$

Proof. By Theorem 1.6, we can find an invertible matrix $P = (\, \boldsymbol{p}_1 \quad \cdots \quad \boldsymbol{p}_n \,)$ for which $A = P^T P$. Hence $A_{jk} = \langle \boldsymbol{p}_j, \boldsymbol{p}_k \rangle$ and $A_{kk} = \|\boldsymbol{p}_k\|^2 > 0$ (if $\boldsymbol{p}_k = 0$, then the columns of P would not be linearly independent, contradicting the invertibility of P). By the Cauchy-Schwarz inequality, we deduce

$$|A_{jk}| = |\langle \boldsymbol{p}_j, \boldsymbol{p}_k \rangle| < \|\boldsymbol{p}_j\| \|\boldsymbol{p}_k\| = \sqrt{A_{jj}A_{kk}},$$

and the inequality is strict because $\boldsymbol{p}_j$ and $\boldsymbol{p}_k$ are linearly independent, being the columns of an invertible matrix. □

Exercise 1.51. Let $A \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. Show that $A_{kk} = \boldsymbol{e}_k^T A \boldsymbol{e}_k$. Why does this imply $A_{kk} > 0$?

Thus Theorem 1.6 implies that, if $A \in \mathbb{R}^{n\times n}$ is symmetric positive definite, then we can find a lower triangular matrix $L \in \mathbb{R}^{n\times n}$ such that $A = LL^T$. However, although we could in principle simply apply the proof given above, there is an easier way. I'll present this method in an example, and then prove that the algorithm works.

Example 1.14. Let

$$A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 5/2 & -1 \\ 0 & -1 & 5/2 \end{pmatrix} \tag{1.32}$$

which you may assume is positive definite. We shall construct a lower triangular matrix $L = (\, \boldsymbol{\ell}_1 \quad \boldsymbol{\ell}_2 \quad \boldsymbol{\ell}_3 \,) \in \mathbb{R}^{3\times 3}$ for which $A = LL^T$. Using Exercise 1.14, we have

$$A = \boldsymbol{\ell}_1\boldsymbol{\ell}_1^T + \boldsymbol{\ell}_2\boldsymbol{\ell}_2^T + \boldsymbol{\ell}_3\boldsymbol{\ell}_3^T.$$

However

$$\boldsymbol{\ell}_2\boldsymbol{\ell}_2^T = \begin{pmatrix} 0 \\ \times \\ \times \end{pmatrix} (\, 0 \quad \times \quad \times \,) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \times & \times \\ 0 & \times & \times \end{pmatrix} \tag{1.33}$$

and

$$\boldsymbol{\ell}_3\boldsymbol{\ell}_3^T = \begin{pmatrix} 0 \\ 0 \\ \times \end{pmatrix} (\, 0 \quad 0 \quad \times \,) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \times \end{pmatrix}, \tag{1.34}$$

which implies

$$A_{j1} = (\boldsymbol{\ell}_1\boldsymbol{\ell}_1^T)_{j1} = (\boldsymbol{\ell}_1)_j(\boldsymbol{\ell}_1)_1, \qquad 1 \leq j \leq 3.$$

Setting $j = 1$, we see that one choice is $(\boldsymbol{\ell}_1)_1 = \sqrt{A_{11}} = \sqrt{2}$. For $j > 1$, we then have $(\boldsymbol{\ell}_1)_{j1} = A_{j1}/\sqrt{A_{11}}$, or

$$\boldsymbol{\ell}_1 = \frac{1}{\sqrt{A_{11}}} (\text{first column of } A) = \frac{1}{\sqrt{2}} \begin{pmatrix} 2 \\ -1 \\ 0 \end{pmatrix}.$$

Now

$$\boldsymbol{\ell}_1 \boldsymbol{\ell}_1^T = \frac{1}{2} \begin{pmatrix} 2 \\ -1 \\ 0 \end{pmatrix} \begin{pmatrix} 2 & -1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 1/2 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Thus

$$A^{(1)} \equiv A - \boldsymbol{\ell}_1 \boldsymbol{\ell}_1^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & -1 \\ 0 & -1 & 5/2 \end{pmatrix}.$$

By (1.33, 1.34), we have

$$A^{(1)}_{j2} = (\boldsymbol{\ell}_2 \boldsymbol{\ell}_2^T)_{j2} = (\boldsymbol{\ell}_2)_j (\boldsymbol{\ell}_2)_2,$$

whence we choose $(\boldsymbol{\ell}_2)_2 = \sqrt{A^{(1)}_{22}} = \sqrt{2}$ and

$$\boldsymbol{\ell}_2 = \frac{1}{\sqrt{A^{(1)}_{22}}} \left(\text{second column of } A^{(1)}\right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 2 \\ -1 \end{pmatrix}.$$

Therefore

$$\boldsymbol{\ell}_2 \boldsymbol{\ell}_2^T = \frac{1}{2} \begin{pmatrix} 0 \\ 2 \\ -1 \end{pmatrix} \begin{pmatrix} 0 & 2 & -1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & -1 \\ 0 & -1 & 1/2 \end{pmatrix},$$

and

$$A^{(2)} \equiv A^{(1)} - \boldsymbol{\ell}_2 \boldsymbol{\ell}_2^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

Hence

$$\boldsymbol{\ell}_3 = \begin{pmatrix} 0 \\ 0 \\ \sqrt{2} \end{pmatrix} = \frac{1}{\sqrt{A^{(3)}_{33}}} \left(\text{third column of } A^{(2)}\right).$$

Recombining our columns to form L, we obtain

$$L = \begin{pmatrix} \sqrt{2} & 0 & 0 \\ -1/\sqrt{2} & \sqrt{2} & 0 \\ 0 & -1/\sqrt{2} & \sqrt{2} \end{pmatrix}. \tag{1.35}$$

Exercise 1.52. Prove that the matrix of equation (1.32) is positive definite.

Let's consider the method of the last example more abstractly. Accordingly, let $A \in \mathbb{R}^{n\times n}$ be any symmetric positive definite matrix. Then $A_{11} > 0$ and we can define the vector

$$\boldsymbol{\ell}_1 = \frac{1}{\sqrt{A_{11}}} \begin{pmatrix} A_{11} \\ A_{21} \\ \vdots \\ A_{n1} \end{pmatrix}. \tag{1.36}$$

In other words, we have (*cf.* Exercise 1.5)

$$\boldsymbol{\ell}_1 = \frac{A\boldsymbol{e}_1}{\sqrt{A_{11}}} = \frac{A\boldsymbol{e}_1}{\sqrt{\boldsymbol{e}_1^T A \boldsymbol{e}_1}}. \tag{1.37}$$

We then form the matrix

$$A^{(1)} = A - \boldsymbol{\ell}_1 \boldsymbol{\ell}_1^T, \tag{1.38}$$

which is symmetric (why?). Now the first row and column of A and $\boldsymbol{\ell}_1\boldsymbol{\ell}_1^T$ are identical, because

$$(\boldsymbol{\ell}_1\boldsymbol{\ell}_1^T)_{j1} = (\boldsymbol{\ell}_1)_j(\boldsymbol{\ell}_1)_1 = \frac{A_{j1}}{\sqrt{A_{11}}}\frac{A_{11}}{\sqrt{A_{11}}} = A_{j1},$$

which implies that the first row and column of $A^{(1)}$ consist entirely of zeros. To illustrate this, we write

$$A^{(1)} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & \times & \times & \cdots & \times \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & \times & \times & \cdots & \times \end{pmatrix},$$

where $\times$ denotes any element that is not guaranteed to be zero. Writing this in the form

$$A^{(1)} = \begin{pmatrix} 0 & \cdots & 0 \\ \vdots & B & \\ 0 & & \end{pmatrix},$$

so that $B \in \mathbb{R}^{(n-1)\times(n-1)}$, we prove the following key result.

Theorem 1.8. The $(n-1)\times(n-1)$ matrix B is symmetric positive definite.

Proof. We must show that, if $\boldsymbol{v} \in \mathbb{R}^n$ is not the zero vector and $\boldsymbol{v}^T\boldsymbol{e}_1 = 0$, then

$$\boldsymbol{v}^T \left(A - \boldsymbol{\ell}_1\boldsymbol{\ell}_1^T\right) \boldsymbol{v} > 0.$$

Now

$$\begin{aligned} \boldsymbol{v}^T \left(A - \boldsymbol{\ell}_1 \boldsymbol{\ell}_1^T \right) \boldsymbol{v} &= \boldsymbol{v}^T A \boldsymbol{v} - (\boldsymbol{\ell}_1^T \boldsymbol{v})^2 \\ &= \boldsymbol{v}^T A \boldsymbol{v} - \frac{(\boldsymbol{e}_1^T A \boldsymbol{v})^2}{\boldsymbol{e}_1^T A \boldsymbol{e}_1} \\ &= \frac{\|\boldsymbol{v}\|_A^2 \|\boldsymbol{e}_1\|_A^2 - (\langle \boldsymbol{e}_1, \boldsymbol{v} \rangle_A)^2}{\|\boldsymbol{e}_1\|_A^2} \\ &> 0, \end{aligned}$$

by the Cauchy-Schwarz inequality — we get strict inequality because, by hypothesis, the vectors $\boldsymbol{e}_1$ and $\boldsymbol{v}$ are orthogonal, and hence linearly independent. □

With this result in hand, we know that we can repeat our construction, because $A_{22}^{(1)} = B_{22} > 0$, by Proposition 1.7, and we can define

$$\boldsymbol{\ell}_2 = \frac{1}{\sqrt{A_{22}^{(1)}}} \begin{pmatrix} 0 \\ A_{22}^{(1)} \\ \vdots \\ A_{n2}^{(1)} \end{pmatrix},$$

and then form

$$A^{(2)} = A^{(1)} - \boldsymbol{\ell}_2 \boldsymbol{\ell}_2^T,$$

whose first and second rows and columns contain only zeros. Further, because of our last theorem, the bottom right $(n-2) \times (n-2)$ of $A^{(2)}$ will also be positive definite symmetric, so that we can continue our algorithm.

What's the point of the Cholesky factorization? We can use it to solve the linear system $A\boldsymbol{x} = \boldsymbol{y}$ when A is symmetric positive definite, as follows. First, let's assume we have calculated $A = LL^T$ using the algorithm given in the last exercise. We first solve $L\boldsymbol{z} = \boldsymbol{y}$ using forward substitution. Then we solve $L^T \boldsymbol{x} = \boldsymbol{z}$ using back substitution. This is the preferred method for solving such systems for $n \leq 1000$, or $n \leq 10^4$ on faster machines.

Exercise 1.53. Solve

$$A\boldsymbol{x} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix},$$

where A is given by (1.32).

1.7. Least squares problems

Let $A = (\boldsymbol{a}_1 \quad \boldsymbol{a}_2 \quad \cdots \quad \boldsymbol{a}_n) \in \mathbb{R}^{m \times n}$, where $m > n$. In general, the linear system

$$A\boldsymbol{x} = \boldsymbol{y}, \qquad \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{y} \in \mathbb{R}^m,$$

has no solution. However, many practical problems generate systems of this form where, typically, the number m of equations is vastly greater than the number n of unknowns. Further, such applications often have the property that it is possible to choose $\boldsymbol{x}$ so that $\boldsymbol{y} - A\boldsymbol{x}$ is "small". The problem is to choose the best $\boldsymbol{x}$, in some sense. This section deals with one fundamental solution to this problem, but it's important to understand that it's not the only solution. We shall begin with a longish example.

Example 1.15. For a simple pendulum of length ℓ, the period of oscillation is given approximately by

$$T \approx 2\pi\sqrt{\ell/g}, \tag{1.39}$$

where g denotes the acceleration due to gravity. If we choose many lengths, and let $\boldsymbol{L} = (\sqrt{\ell_1} \quad \cdots \quad \sqrt{\ell_n})^T$ and measure the corresponding periods $\boldsymbol{T} = (T_1 \quad T_2 \quad \cdots \quad T_n)^T$, then we should have two vectors for which

$$\boldsymbol{T} \approx C\boldsymbol{L},$$

where $C = 2\pi/\sqrt{g}$. Once we've estimated the best C, in some sense, then we can estimate g via the equation $g = 4\pi^2/C^2$, so that this experiment provides a simple method for estimating the acceleration due to gravity. How do we estimate C?

One method is to choose C to minimize the sum of squares

$$S(C) = \sum_{k=1}^{n} (T_k - CL_k)^2 = \|\boldsymbol{T} - C\boldsymbol{L}\|^2.$$

Thus

$$S(C) = C^2\|\boldsymbol{L}\|^2 - 2C\langle \boldsymbol{T}, \boldsymbol{L}\rangle + \|\boldsymbol{T}\|^2,$$

and, because the coefficient of C^2 is positive, this quadratic possesses exactly one minimum which occurs when $dS/dC = 0$, that is,

$$0 = \frac{dS}{dC} = 2C\|\boldsymbol{L}\|^2 - 2\langle \boldsymbol{T}, \boldsymbol{L}\rangle,$$

or

$$C = \langle \boldsymbol{T}, \boldsymbol{L}\rangle / \|\boldsymbol{L}\|^2,$$

and this is the *best least squares estimate* for C.

We can also picture this geometrically, because the vector $\langle \boldsymbol{T}, \boldsymbol{L}\rangle \boldsymbol{L}/\|\boldsymbol{L}\|^2$ is the *projection* of $\boldsymbol{T}$ onto the line $\{\lambda\boldsymbol{L} : \lambda \in \mathbb{R}\}$. This interpretation yields another way to determine C: we choose C so that $\boldsymbol{T} - C\boldsymbol{L}$ is perpendicular to $\boldsymbol{L}$, so that

$$\langle \boldsymbol{T} - C\boldsymbol{L}, \boldsymbol{L}\rangle = 0,$$

which again implies $C = \langle \boldsymbol{T}, \boldsymbol{L}\rangle / \|\boldsymbol{L}\|^2$.

Now we address our original problem. We want to choose $\boldsymbol{x} \in \mathbb{R}^n$ minimizing the quadratic

$$\begin{aligned} Q(\boldsymbol{x}) &\equiv \|\boldsymbol{y} - A\boldsymbol{x}\|^2 = \|\boldsymbol{y} - \sum_{k=1}^{n} x_k \boldsymbol{a}_k\|^2 \qquad (1.40) \\ &= (\boldsymbol{y} - A\boldsymbol{x})^T(\boldsymbol{y} - A\boldsymbol{x}) \\ &= \boldsymbol{y}^T\boldsymbol{y} - (A\boldsymbol{x})^T\boldsymbol{y} - \boldsymbol{y}^T(A\boldsymbol{x}) + (A\boldsymbol{x})^T(A\boldsymbol{x}) \\ &= \|\boldsymbol{y}\|^2 - 2\boldsymbol{x}^T A^T\boldsymbol{y} + \boldsymbol{x}^T A^T A\boldsymbol{x} \qquad (1.41) \end{aligned}$$

Setting $\boldsymbol{\mu} = A^T\boldsymbol{y}$ and $G = A^T A$, we have

$$Q(\boldsymbol{x}) = \|\boldsymbol{y}\| - 2\boldsymbol{\mu}^T\boldsymbol{x} + \boldsymbol{x}^T G\boldsymbol{x}. \qquad (1.42)$$

Thus

$$\nabla Q(\boldsymbol{x}) = 2(G\boldsymbol{x} - \boldsymbol{\mu}) \qquad (1.43)$$

and the Hessian matrix is given by

$$D^2 f(\boldsymbol{x}) = 2G. \qquad (1.44)$$

This enables us to state and prove our main theorem.

Theorem 1.9. Let $A \in \mathbb{R}^{m\times n}$ have linearly independent columns, and let $\boldsymbol{y} \in \mathbb{R}^m$. Then the vector $\boldsymbol{x}^*$ defined by the so called *normal equations*

$$A^T A\boldsymbol{x}^* = A^T\boldsymbol{y} \qquad (1.45)$$

is the unique vector minimizing the sum of squares

$$Q(\boldsymbol{x}) = \|\boldsymbol{y} - A\boldsymbol{x}\|^2, \qquad (1.46)$$

and this is called the *least squares solution of* $A\boldsymbol{x} = \boldsymbol{y}$.

Proof. Let $A = (\,\boldsymbol{a}_1 \quad \boldsymbol{a}_2 \quad \cdots \quad \boldsymbol{a}_n\,)$. Now $A^T A$ is non-negative definite. Indeed,

$$\boldsymbol{c}^T A^T A\boldsymbol{c} = \|A\boldsymbol{c}\| = \|\sum_{k=1}^{n} c_k \boldsymbol{a}_k\|^2 \geq 0,$$

with equality if and only if $\boldsymbol{c} = 0$, because the columns of A are linearly independent. Thus $A^T A$ is symmetric positive definite, and therefore invertible, which implies that there's a unique vector $\boldsymbol{x}^*$ satisfying the normal equations (1.45), and hence, by Proposition 1.4 minimizing the quadratic. It's a minimum because the Hessian matrix $A^T A$ is symmetric positive definite. □

Example 1.16. Use the normal equations to solve the linear system

$$\begin{pmatrix} 3 & 65 \\ 4 & 0 \\ 12 & 13 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

in the least squares sense. You should find that $x_1 = 0.090587$, $x_2 = 0.010515$, to 5 decimal places.

In practice, forming the normal equations is almost always a bad idea. For reasons that are essentially beyond the scope of the course, it can lead to large computational errors in floating point arithmetic. Further, it's often less efficient to form the matrix A^TA. The preferred method at present uses the QR factorization we've already discussed in some detail. Therefore suppose we have calculated a QR factorization $A = QR$, where $Q \in \mathbb{R}^{m\times m}$ is orthogonal and $R \in \mathbb{R}^{m\times n}$ is upper triangular. By Proposition 1.1,

$$\|\boldsymbol{y} - A\boldsymbol{x}\|^2 = \|Q^T(\boldsymbol{y} - A\boldsymbol{x})\|^2 = \|Q^T\boldsymbol{y} - R\boldsymbol{x}\|^2. \tag{1.47}$$

Now we can write

$$Q^T\boldsymbol{y} = \alpha + \boldsymbol{\beta},$$

where

$$\alpha = \begin{pmatrix} (Q^T\boldsymbol{y})_1 \\ (Q^T\boldsymbol{y})_2 \\ \vdots \\ (Q^T\boldsymbol{y})_n \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\beta} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ (Q^T\boldsymbol{y})_{n+1} \\ \vdots \\ (Q^T\boldsymbol{y})_m \end{pmatrix}.$$

Because R is upper triangular, we also have $(R\boldsymbol{x})^T\boldsymbol{e}_k = 0$, for $n < k \leq m$. Hence, by Pythagoras' theorem

$$\|\boldsymbol{y} - A\boldsymbol{x}\|^2 = \|Q^T\boldsymbol{y} - R\boldsymbol{x}\|^2 = \|(\alpha - R\boldsymbol{x}) + \boldsymbol{\beta}\|^2 = \|\alpha - R\boldsymbol{x}\|^2 + \|\boldsymbol{\beta}\|^2 \geq \|\boldsymbol{\beta}\|^2,$$

with equality if and only if $R\boldsymbol{x} = \boldsymbol{\beta}$, which, if the diagonal elements of R are nonzero, we can solve by back substitution. Let's state this formally.

Algorithm 1.3. (The QR factorization via Givens Rotations) Let $A \in \mathbb{R}^{m\times n}$, $m > n$, have linearly independent columns, and let $\boldsymbol{y} \in \mathbb{R}^m$. For each column $j \in \{1, 2, \ldots, n\}$, and for each row $k \in \{j+1, j+2, \ldots, m\}$, we apply a Givens rotation to rows j and k to zeroize element A_{kj}; call these rotations $\{Q_1, Q_2, \ldots\}$. Thus we reduce A to upper triangular form (*triangularize* A is the jargon!). We also apply each rotation to $\boldsymbol{y}$ in turn.

Thus we obtain

$$\cdots Q_2 Q_1 A = R = \begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1n} \\ & R_{22} & & R_{2n} \\ & & \ddots & \vdots \\ & & & R_{nn} \\ & & & \\ & & & \\ & & & \\ & & & \end{pmatrix} \quad \text{and} \quad \boldsymbol{z} = \cdots Q_2 Q_1 \boldsymbol{y}.$$

Finally, we solve

$$\begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1n} \\ & R_{22} & & R_{2n} \\ & & \ddots & \vdots \\ & & & R_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \\ \cdots \\ z_n \end{pmatrix},$$

if R is invertible, obtaining the least squares solutions of the original linear system.

Exercise 1.54. The diagonal elements $R_{11}, \ldots, R_{nn}$ are all nonzero if and only if the columns of A are linearly independent. Prove this.

Exercise 1.55. Show that $A^T A = R^T R$. Hence prove that R is invertible if the columns of A are linearly independent.

Example 1.17. We shall use Givens rotations to obtain the least squares solution of the linear system

$$A \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

where the matrix A is given by (1.21). The Givens rotations required to triangularize A were computed in Example 1.8. It is easily checked that

$$G_{23}(\psi) G_{13}(\phi) G_{12}(\theta) \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{19}{13} \\ \frac{501}{13\sqrt{3665}} \\ \frac{41}{\sqrt{3665}} \end{pmatrix} = \begin{pmatrix} 1.46154 \\ 0.63659 \\ 0.67725 \end{pmatrix},$$

to 5 decimal places. Therefore we simply solve the 2×2 linear system

$$\begin{pmatrix} 13 & 27 \\ 0 & \sqrt{3665} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1.46154 \\ 0.63659 \end{pmatrix},$$

which provides $x_1 = 0.090587$, $x_2 = 0.010515$.

Finally, let's state our algorithms in a more formal way. The following pseudo-code is intended for students who are familiar with a programming language, but it should be readily apprehended by all.

Algorithm 1.4. Let $A \in \mathbb{R}^{m\times n}$, $m > n$. The following calculations will reduce A to an upper triangular matrix (*triangularize is the jargon!*) by applying Givens rotations.

For $j = 1, 2, \ldots, n$

  For $k = j+1, j+2, \ldots, m$

    Let $c = A_{jj}/(A_{jj}^2 + A_{kj}^2)^{1/2}$, $s = A_{kj}/(A_{jj}^2 + A_{kj}^2)^{1/2}$.

    Replace $\begin{pmatrix} A_{jj} & A_{jj+1} & \cdots & A_{jn} \\ A_{kj} & A_{kj+1} & \cdots & A_{kn} \end{pmatrix}$

    by $\begin{pmatrix} c & s \\ -s & c \end{pmatrix}\begin{pmatrix} A_{jj} & A_{jj+1} & \cdots & A_{jn} \\ A_{kj} & A_{kj+1} & \cdots & A_{kn} \end{pmatrix}$

On completion, the resulting matrix $R \equiv A$ is upper triangular.

The next algorithm describes the least squares solution of an over-determined set of linear equations using Givens rotations.

Algorithm 1.5. Let $A \in \mathbb{R}^{m\times n}$, where $m > n$ and let $\boldsymbol{y} \in \mathbb{R}^m$.

For $j = 1, 2, \ldots, n$

  For $k = j+1, j+2, \ldots, m$

    Let $c = A_{jj}/(A_{jj}^2 + A_{kj}^2)^{1/2}$, $s = A_{kj}/(A_{jj}^2 + A_{kj}^2)^{1/2}$.

    Replace $\begin{pmatrix} A_{jj} & A_{jj+1} & \cdots & A_{jn} & y_j \\ A_{kj} & A_{kj+1} & \cdots & A_{kn} & y_k \end{pmatrix}$

    by $\begin{pmatrix} c & s \\ -s & c \end{pmatrix}\begin{pmatrix} A_{jj} & A_{jj+1} & \cdots & A_{jn} & y_j \\ A_{kj} & A_{kj+1} & \cdots & A_{kn} & y_k \end{pmatrix}$

The resulting matrix $R \equiv A$ is now upper triangular. We now compute the solution of the linear system

$$\begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1n} \\ & R_{22} & \cdots & R_{2n} \\ & & \ddots & \\ & & & R_{nn} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

by back substitution.

1.8. Solutions to Exercises

Solution 1.1. For any square matrix C, $(C + C^T)^T = C^T + C$, because $(C^T)^T = C$. Similarly, $(C - C^T)^T = C^T - C = -(C - C^T)$.

Setting $S = (A - A^T)/2$ and $T = (A + A^T)/2$, we have $A = S + T$, where S is skew-symmetric and T is symmetric. The decomposition is unique because $S_1 + T_1 = S_2 + T_2$ implies $S_1 - S_2 = T_2 - T_1$, and any matrix that is both symmetric and skew-symmetric is the zero matrix.

Solution 1.2. A column vector is simply an $n \times 1$ matrix, whilst a row vector is a $1 \times n$ matrix. Applying the definition of matrix multiplication, we deduce that $\boldsymbol{a}^T\boldsymbol{a}$ is a 1×1 matrix – a scalar – and $\boldsymbol{a}\boldsymbol{a}^T$ is an $n \times n$ matrix. Further, using (1.8) with $p = 1 = r$ and $q = n$, we obtain

$$\boldsymbol{a}^T\boldsymbol{a} = (\boldsymbol{a}^T\boldsymbol{a})_{11} = \sum_{\ell=1}^{n} (\boldsymbol{a}^T)_{1\ell}(\boldsymbol{a})_{1\ell} = \sum_{\ell=1}^{n} a_\ell^2.$$

Similarly, setting $p = n = r$ and $q = 1$ in (1.8), we find

$$(\boldsymbol{a}\boldsymbol{a}^T)_{jk} = \sum_{\ell=1}^{1} (\boldsymbol{a})_{j\ell}(\boldsymbol{a}^T)_{\ell k} = a_j a_k,$$

as required.

Solution 1.3. For any vector $\boldsymbol{w} \in \mathbb{R}^n$, we have

$$A\boldsymbol{w} = (\boldsymbol{u}\boldsymbol{v}^T)\boldsymbol{w} = \boldsymbol{u}(\boldsymbol{v}^T\boldsymbol{w}),$$

using the associativity of matrix multiplication (see next exercise). We see that2 A maps every vector in $\mathbb{R}^n$ to a multiple of $\boldsymbol{u}$. Thus $\boldsymbol{u}$ is itself an eigenvector, with eigenvalue $\boldsymbol{v}^T\boldsymbol{w}$. Moreover, for any vector $\boldsymbol{w}$ orthogonal to $\boldsymbol{v}$, we have $A\boldsymbol{w} = 0$, an eigenvector with eigenvalue zero. Since the set of solutions of the equation $\boldsymbol{w}^T\boldsymbol{v} = 0$ forms an $(n-1)$-dimensional subspace of $\mathbb{R}^n$ (see `M2P1`), we can choose any basis of this space to complete our set of n eigenvectors.

Solution 1.4. Let $A \in \mathbb{R}^{p\times q}$, $B \in \mathbb{R}^{q\times r}$ and $A \in \mathbb{R}^{r\times s}$. Then

$$((AB)C)_{jk} = \sum_{\beta=1}^{r} (AB)_{j\beta}C_{\beta k} = \sum_{\beta=1}^{r}\sum_{\alpha=1}^{q} A_{j\alpha}B_{\alpha\beta}C_{\beta k},$$

whilst

$$(A(BC))_{jk} = \sum_{\alpha=1}^{q} A_{j\alpha}(BC)_{\alpha k} = \sum_{\alpha=1}^{q}\sum_{\beta=1}^{r} A_{j\alpha}B_{\alpha\beta}C_{\beta k},$$

and these are equal because the order of summation is a finite sum is irrelevant.

Solution 1.5. Method I: Set $\boldsymbol{x} = \boldsymbol{e}_j$ and $\boldsymbol{y} = \boldsymbol{e}_k$ in Example 1.3.

Method II: By definition of matrix multiplication, and using the Kronecker delta notation,

$$(A\boldsymbol{e}_k)_\ell = \sum_{m=1}^{n} A_{\ell m}(\boldsymbol{e}_k)_m = \sum_{m=1}^{n} A_{\ell m}\delta_{km} = A_{\ell k}.$$

Thus

$$\boldsymbol{e}_j^T A\boldsymbol{e}_k = \sum_{\ell=1}^{n}(\boldsymbol{e}_j)_\ell(A\boldsymbol{e}_k)_\ell = \sum_{\ell=1}^{n}\delta_{j\ell}A_{\ell k} = A_{jk}.$$

1.3.

Solution 1.6. You should find that

$$P_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{for} \quad P_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Thus P_1 (P_2) projects onto the first (second) coordinate axis.

Solution 1.7. This exercise applies the fact that $(AB)^T = B^T A^T$. Indeed, we have

$$(\boldsymbol{w}\boldsymbol{w}^T)^T = (\boldsymbol{w}^T)^T\boldsymbol{w}^T = \boldsymbol{w}\boldsymbol{w}^T.$$

Thus $P = P^T$, using the elementary properties $(A+B)^T = A^T + B^T$ and $(\lambda A)^T = \lambda A^T$, for any matrices A, B for which their sum is defined, and for any scalar λ.

For any vector $\boldsymbol{v} \in \mathbb{R}^n$, we have

$$P\boldsymbol{v} = \left(I_n - \frac{\boldsymbol{w}\boldsymbol{w}^T}{\boldsymbol{w}^T\boldsymbol{w}}\right)\boldsymbol{v} = \boldsymbol{v} - \frac{(\boldsymbol{w}\boldsymbol{w}^T)\boldsymbol{v}}{\boldsymbol{w}^T\boldsymbol{w}} = \boldsymbol{v} - \frac{\boldsymbol{w}(\boldsymbol{w}^T\boldsymbol{v})}{\boldsymbol{w}^T\boldsymbol{w}},$$

using associativity of matrix multiplication. Hence

$$\boldsymbol{w}^T P\boldsymbol{v} = \boldsymbol{w}^T\boldsymbol{v} - \frac{(\boldsymbol{w}^T\boldsymbol{w})(\boldsymbol{w}^T\boldsymbol{v})}{\boldsymbol{w}^T\boldsymbol{w}} = 0.$$

Solution 1.8. Here A and B are square matrices of the same order, $n \times n$ say. Now

$$\text{trace } AB = \sum_{j=1}^{n}(AB)_{jj} = \sum_{j=1}^{n}\sum_{k=1}^{n} A_{jk}B_{kj}.$$

Similarly,

$$\text{trace } BA = \sum_{k=1}^{n}(BA)_{kk} = \sum_{k=1}^{n}\sum_{\ell=1}^{n} B_{k\ell}A_{\ell k}.$$

But these are identical, because the order of summation in a finite sum is immaterial.

Solution 1.9. By definition of matrix multiplication,

$$(A\boldsymbol{e}_j)_k = \sum_{m=1}^{n} A_{km}(\boldsymbol{e}_j)_m = \sum_{m=1}^{n} A_{km}\delta_{jm} = A_{kj} = (\boldsymbol{a}_j)_k.$$

Solution 1.10. We have

$$(AB)_{jk} = \sum_{\ell=1}^{q} A_{j\ell}B_{\ell k} = \sum_{\ell=1}^{q} A_{j\ell}(\boldsymbol{b}_k)_\ell = (A\boldsymbol{b}_k)_j.$$

Solution 1.11. Simply note that

$$(A^T)_{jk} = A_{kj} = (\boldsymbol{a}_j)_k = (\boldsymbol{a}_j^T)_k.$$

Solution 1.12. By definition of matrix multiplication,

$$(AB)_{jk} = \sum_{\ell=1}^{q} A_{j\ell}B_{\ell k} = \sum_{\ell=1}^{q} (\boldsymbol{a}_j)_\ell(\boldsymbol{b}_k)_\ell = \boldsymbol{a}_j^T\boldsymbol{b}_k.$$

Solution 1.13. An easy exercise: just use Examples 1.24 and 1.25.

Solution 1.14. By definition of matrix multiplication,

$$(PQ)_{\alpha\beta} = \sum_{\gamma=1}^{m} P_{\alpha\gamma}Q_{\gamma\beta} = \sum_{\gamma=1}^{m} (\boldsymbol{p}_\gamma)_\alpha(\boldsymbol{q}_\gamma)_\beta.$$

Solution 1.15. Straightforward calculation yields $x_1 = 7/12$, $x_2 = 11/24$ and $x_3 = -1/6$.

Solution 1.16. At the first step of back substitution, we require one operation to form $x_n = y_n/U_{nn}$. At the kth step, for $k = n-1, n-2, \ldots, 2, 1$, we require $n-k$ multiplications, $n-k$ additions or subtractions, and one division. Assuming all operations have equal cost for simplicity, we obtain $2n - 2k + 1$ operations. Thus the total cost is given by

$$C(n) = 1 + \sum_{k=1}^{n-1}(2n - 2k + 1),$$

and now recall the elementary fact $1 + 2 + \cdots + m = m(m+1)/2$.

Solution 1.17. The easiest way to do this is to use the fact that $\det U = U_{11}U_{22}\cdots U_{nn}$, so that $\det U \neq 0$ if and only if every diagonal element is nonzero. Here's another way: If every diagonal element of U is nonzero, then we can use back substitution to solve $U\boldsymbol{x} = \boldsymbol{y}$ for every vector $\boldsymbol{y}$. Hence U is invertible. Conversely, if we can choose an integer j for which $U_{jj} = 0$, then U maps the j linearly independent vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_j$ on a subspace of span $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{j-1}\}$. In particular, the vectors $U\boldsymbol{e}_1, \ldots, U\boldsymbol{e}_j$ must be linearly dependent, so that U cannot be invertible.

Solution 1.18. Let $L \in \mathbb{R}^{n\times n}$ be an invertible lower triangular, so that every diagonal element is nonzero. To solve $L\boldsymbol{x} = \boldsymbol{y}$, we first set $x_1 = y_1/L_{11}$, then, for $k = 2, 3, \ldots, n$, we form

$$x_k = \left(y_k - \sum_{\ell=1}^{k-1} L_{k\ell} x_\ell \right) / L_{kk}.$$

Solution 1.19. I shall only deal with the upper triangular case, the lower triangular case being extremely similar. Thus let $U = \begin{pmatrix} \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_n \end{pmatrix} \in \mathbb{R}^{n\times n}$ be any invertible upper triangular matrix – in other words every diagonal element is nonzero. Let $U^{-1} = \begin{pmatrix} \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_n \end{pmatrix}$. Thus the columns of the inverse are given by the equations

$$U\boldsymbol{v}_j = \boldsymbol{e}_j, \qquad \text{for } j = 1, 2, \ldots, n.$$

Applying the back substitution algorithm, we see that $x_k = 0$ for $k > j$, which is equivalent to the statement that U^{-1} is upper triangular.

Solution 1.20. For a matrix-vector, we compute

$$(A\boldsymbol{v})_j = \sum_{k=1}^{n} A_{jk} x_k, \qquad 1 \le j \le n,$$

which clearly requires n^2 multiplications and $(n-1)n$ additions. For matrix multiplication, we use Equation (1.5), which requires n multiplications and $n-1$ additions for each of the n^2 elements in the matrix product.

Solution 1.21. Easy, but I'll provide an answer for $n = 10^7$. Here the operation count is $10^{21}/6$. Since our computer performs 10^8 operations every second, this requires $10^{13}/6$ seconds, or $10^6/9 \approx 10^5$ years.

Solution 1.22. If we first calculate AB, then this costs $\mathcal{O}(n^3)$ operations. However, calculating $\boldsymbol{y} = B\boldsymbol{x}$, and then forming $A\boldsymbol{y}$, only requires $\mathcal{O}(n^2)$ operations. (Consider the saving when $n = 100$, say.)

Solution 1.23. We have

$$\langle \boldsymbol{v}, \boldsymbol{u} \rangle = \sum_{k=1}^{n} v_k u_k = \sum_{k=1}^{n} u_k v_k = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$$

and

$$\begin{aligned} \langle \alpha_1 \boldsymbol{u}_1 + \alpha_2 \boldsymbol{u}_2, \boldsymbol{v} \rangle &= \sum_{k=1}^{n} \left(\alpha_1 (\boldsymbol{u}_1)_k + \alpha_2 (\boldsymbol{u}_2)_k \right) v_k \\ &= \alpha_1 \sum_{k=1}^{n} (\boldsymbol{u}_1)_k v_k + \alpha_2 \sum_{k=1}^{n} (\boldsymbol{u}_2)_k v_k \\ &= \alpha_1 \langle \boldsymbol{u}_1, \boldsymbol{v} \rangle + \alpha + 2 \langle \boldsymbol{u}_2, \boldsymbol{v} \rangle. \end{aligned}$$

The third relation is derived in the same fashion.

Solution 1.24. We have

$$\begin{aligned} \|\boldsymbol{a}+\boldsymbol{b}\|^2 &= \langle \boldsymbol{a}+\boldsymbol{b}, \boldsymbol{a}+\boldsymbol{b}\rangle \\ &= \langle \boldsymbol{a},\boldsymbol{a}\rangle + \langle \boldsymbol{a},\boldsymbol{b}\rangle + \langle \boldsymbol{b},\boldsymbol{a}\rangle + \langle \boldsymbol{b},\boldsymbol{b}\rangle \\ &= \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 + 2\langle \boldsymbol{a},\boldsymbol{b}\rangle. \end{aligned}$$

Solution 1.25. For $j \neq k$, we have

$$\begin{aligned} \int_{-\pi}^{\pi} e_j(t)\overline{e_k(t)}\,dt &= \int_{-\pi}^{\pi} e^{i(j-k)t}\,dt \\ &= \left[\frac{e^{i(j-k)t}}{i(j-k)}\right]_{-\pi}^{\pi} \\ &= 0, \end{aligned}$$

because, setting $g(t) = \exp(i(j-k)t)$, we note that $g(t+2\pi) = g(t)$, remembering that $\exp(2\pi i) = 1$.

Solution 1.26. In an n-dimensional vector space, any collection of $m > n$ vectors is linearly dependent. Since a set of m nonzero orthogonal vectors is linearly independent, we must have $m \leq n$. To see that nonzero orthogonal vectors are linearly independent, note that $\sum_{k=1}^{m} c_k \boldsymbol{a}_k = 0$ implies

$$0 = \langle \boldsymbol{a}_j, \sum_{k=1}^{m} c_k \boldsymbol{a}_k\rangle = \sum_{k=1}^{m} c_k \langle \boldsymbol{a}_j, \boldsymbol{a}_k\rangle = c_j \|\boldsymbol{a}_j\|^2.$$

Since each $\boldsymbol{a}_j$ is nonzero, we have $\|\boldsymbol{a}_j\| > 0$, whence $c_j = 0$; this is true for every $j \in \{1, 2, \ldots, n\}$.

Solution 1.27. By construction, $\boldsymbol{v}_k$ is orthogonal to $\boldsymbol{q}_j$, for $1 \leq j < k$. But $\boldsymbol{v}_j$ and $\boldsymbol{a}_j$ are linear combinations of $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_j$. Applying Gram-Schmidt, we get $\boldsymbol{q}_1 = (\,1 \quad 0 \quad 0 \quad 0\,)^T$,

$$\boldsymbol{v}_2 = \boldsymbol{a}_2 - (\boldsymbol{a}_2^T \boldsymbol{q}_1)\boldsymbol{q}_1 = (\,0 \quad 1 \quad 0 \quad 0\,)^T,$$

whence $\boldsymbol{q}_2 = \boldsymbol{v}_2$, and

$$\boldsymbol{v}_3 = \boldsymbol{a}_3 - (\boldsymbol{a}_3^T \boldsymbol{q}_1)\boldsymbol{q}_1 - (\boldsymbol{a}_3^T \boldsymbol{q}_2)\boldsymbol{q}_2 = (\,0 \quad 0 \quad 0 \quad -1\,)^T,$$

so that $\boldsymbol{q}_3 = \boldsymbol{v}_3$.

Solution 1.28. By associativity of matrix multiplication, we have

$$\begin{aligned}(QR)\boldsymbol{e}_k &= Q(R\boldsymbol{e}_k) = Q\left(\sum_{\ell=1}^{k} r_{\ell k}\boldsymbol{e}_\ell\right)\\ &= \sum_{\ell=1}^{k} r_{\ell k} Q\boldsymbol{e}_\ell\\ &= \sum_{\ell=1}^{k} r_{\ell k}\boldsymbol{q}_\ell,\end{aligned}$$

as required.

Solution 1.29. Method I: Note that the columns are orthonormal.

Method II: By direct computation, $Q^TQ = I$.

In both cases, we use the elementary trigonometric identity $\cos^2\theta + \sin^2\theta = 1$.

Solution 1.30. We have $(U_1U_2)^T(U_1U_2) = U_2^T(U_1^TU_1)U_2 = U_2^TU_2 = I$, using Example 1.2.

Solution 1.31. Example 1.2 applied to the $n \times n$ matrix $\boldsymbol{v}\boldsymbol{v}^T$ yields the formula $(\boldsymbol{v}\boldsymbol{v}^T)^T = (\boldsymbol{v}^T)^T\boldsymbol{v}^T = \boldsymbol{v}\boldsymbol{v}^T$. Since $\boldsymbol{v}^T\boldsymbol{v}$ is a scalar, we deduce that ρ is a symmetric matrix. Thus

$$\begin{aligned}\rho^T\rho &= \rho^2\\ &= I - 4\frac{\boldsymbol{v}\boldsymbol{v}^T}{\boldsymbol{v}^T\boldsymbol{v}} + 4\frac{(\boldsymbol{v}\boldsymbol{v}^T)(\boldsymbol{v}\boldsymbol{v}^T)}{(\boldsymbol{v}^T\boldsymbol{v})^2}\\ &= I - 4\frac{\boldsymbol{v}\boldsymbol{v}^T}{\boldsymbol{v}^T\boldsymbol{v}} + 4\frac{\boldsymbol{v}(\boldsymbol{v}^T\boldsymbol{v})\boldsymbol{v}^T}{(\boldsymbol{v}^T\boldsymbol{v})^2}\\ &= I - 4\frac{\boldsymbol{v}\boldsymbol{v}^T}{\boldsymbol{v}^T\boldsymbol{v}} + 4\frac{\boldsymbol{v}\boldsymbol{v}^T}{\boldsymbol{v}^T\boldsymbol{v}}\\ &= I,\end{aligned}$$

using associativity of matrix multiplication.

For $n = 2$ and $\boldsymbol{v} = (\,0 \quad 1\,)^T$, we obtain

$$P = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

which is reflection in the subspace orthogonal to $\boldsymbol{v}$. In general, ρ simply reflects vectors in the $(n-1)$-dimensional subspace of $\mathbb{R}^n$ orthogonal to the vector $\boldsymbol{v}$.

Solution 1.32. We only have to prove that the columns of Q are orthonormal, and this is obvious.

Solution 1.33. The 2×2 rotation matrix

$$Q_1 = \begin{pmatrix} a/(a^2+c^2)^{1/2} & c/(a^2+c^2)^{1/2} \\ -c/(a^2+c^2)^{1/2} & a/(a^2+c^2)^{1/2} \end{pmatrix}$$

satisfies

$$Q_1 \begin{pmatrix} a \\ c \end{pmatrix} = \begin{pmatrix} (a^2+c^2)^{1/2} \\ 0 \end{pmatrix}.$$

Thus $Q = Q_1^T$ and $R = Q_1 A$.

Solution 1.34. This is Example 1.17.

Solution 1.35. Apply the Cauchy-Schwarz inequality to the vectors $\boldsymbol{x} = (\, x_1 \quad \cdots \quad x_n \,)^T$ and $\boldsymbol{e} = (\, 1 \quad \cdots \quad 1 \,)^T$.

Solution 1.36. The Cauchy-Schwarz inequality implies the bound

$$-\|\boldsymbol{a}\|\|\boldsymbol{b}\| \leq \langle \boldsymbol{a}, \boldsymbol{b} \rangle \leq \|\boldsymbol{a}\|\|\boldsymbol{b}\|,$$

so that

$$\|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 - 2\|\boldsymbol{a}\|\|\boldsymbol{b}\| \leq \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 + 2\langle \boldsymbol{a}, \boldsymbol{b} \rangle \leq \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 + 2\|\boldsymbol{a}\|\|\boldsymbol{b}\|,$$

and this is the required inequality. Their geometric interpretation is simply that the sum of the lengths of the two shorter sides of a triangle cannot exceed the length of the longest side.

Solution 1.37. This is simple, so I'll only provide an answer for $\sqrt{65}$. We let $f(x) = x^{1/2}$, $f'(x) = (1/2)x^{-1/2}$ and $f''(x) = (-1/4)x^{-3/2}$. Setting $a = 64$ and $h = 1$ in the first displayed equation of this subsection, we obtain the linear approximation $\ell(65) = f(64) + f'(64) = 8 + (1/16)$ and the quadratic approximation $p(65) = \ell(65) + (1/2)f''(64)$. The rest is left to you.

Solution 1.38. We need only note that $\partial f / \partial x_j = \delta_{jp}$, for $1 \leq j \leq n$.

Solution 1.39. We have

$$\frac{\partial f}{\partial x_j} = \cos x_j \prod_{\ell=1, \ell \neq j}^{n} \sin x_\ell$$

and, for $j \neq k$,

$$\frac{\partial^2 f}{\partial x_j \partial x_k} = \cos x_j \cos x_k \prod_{\ell=1, \ell \neq j,k}^{n} \sin x_\ell,$$

whilst

$$\frac{\partial^2 f}{\partial x_j^2} = -f(\boldsymbol{x}).$$

Solution 1.40. If $f(\boldsymbol{x}) = \boldsymbol{a}^T\boldsymbol{x} = a_1x_1 + \cdots + a_nx_n$, then

$$(\nabla f(\boldsymbol{x}))_j = \frac{\partial f}{\partial x_j} = \sum_{k=1}^{n} a_k\delta_{jk} = a_j,$$

that is, $\nabla f(\boldsymbol{x}) = \boldsymbol{a}$.

Solution 1.41. For any two functions $u(\boldsymbol{x})$ and $v(\boldsymbol{x})$, we have

$$\nabla\left(u(\boldsymbol{x}) + v(\boldsymbol{x})\right) = \nabla u(\boldsymbol{x}) + \nabla v(\boldsymbol{x}),$$

because of the elementary relation

$$\frac{\partial}{\partial x_j}(u+v) = \frac{\partial u}{\partial x_j} + \frac{\partial v}{\partial x_j}.$$

Setting $h_1(\boldsymbol{x}) = a + \boldsymbol{b}^T\boldsymbol{x}$, $h_2(\boldsymbol{x}) = \boldsymbol{x}^TA\boldsymbol{x}/2$, and applying Example 1.10, we obtain the result.

Solution 1.42. This question does *not* assume that A is a symmetric matrix. Setting $f(\boldsymbol{x}) = \boldsymbol{x}^TA\boldsymbol{x}$, we see that

$$\nabla h(\boldsymbol{x}) = h(\boldsymbol{x})\nabla f(\boldsymbol{x}).$$

Using the technique of Example 1.10, we see that

$$\nabla f(\boldsymbol{x}) = \left(A + A^T\right)\boldsymbol{x}.$$

Solution 1.43. The function $f(\boldsymbol{x})$ has a local maximumat $\boldsymbol{x} = \boldsymbol{a}$ if there is a positive number δ such that

$$f(\boldsymbol{a}) \geq f(\boldsymbol{a} + h\boldsymbol{u})$$

for every unit vector $\boldsymbol{u} \in \mathbb{R}^n$ and $|h| \leq \delta$.

Solution 1.44. One example is $\boldsymbol{a} = 0$ and $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^4$.

Solution 1.45. By direct calculation,

$$\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2(x^2+y^2) - 2xy = 2(x-y/2)^2 + \frac{3}{2}y^2 \geq 0,$$

with equality if and only if $x - y/2 = 0$ and $y = 0$, that is $x = y = 0$.

Solution 1.46. Suppose A is positive definite. Then $a = \boldsymbol{e}_1^TA\boldsymbol{e}_1 > 0$. Following the hint, we obtain

$$\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = cy^2 + a\left[(x+by/a)^2 - b^2y^2/a^2\right] = a(x+by/a)^2 + \frac{y^2}{a}(ac-b^2).$$

Setting $y = a^{1/2}$ and $x = -by/a$, we find

$$0 < \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = ac - b^2.$$

Conversely, if $a > 0$ and $ac - b^2 > 0$, then

$$\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \geq 0$$

with equality if and only if $y = 0$ and $x + by/a = 0$, that is $x = y = 0$.

Solution 1.47. We find $\langle \boldsymbol{e}_1, \boldsymbol{e}_2 \rangle_A = \boldsymbol{e}_1^T A \boldsymbol{e}_2 = -1$ and $\|\boldsymbol{e}_1\|_A = \|\boldsymbol{e}_2\|_A = \sqrt{2}$.

Solution 1.48. If A is symmetric positive definite, then $A\boldsymbol{x} = 0$ implies $\boldsymbol{x}^T A \boldsymbol{x} = 0$, which implies $\boldsymbol{x} = 0$. (A is invertible if and only if $A\boldsymbol{x} = 0$ implies $\boldsymbol{x} = 0$.)

Solution 1.49. We have

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}^T A \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + z^2,$$

with equality if and only if $x = y = z = 0$. Hence A is positive definite.

Solution 1.50. The matrix P^T is invertible because $\det P^T = \det P \neq 0$. Hence $\boldsymbol{x}^T M \boldsymbol{x} = \boldsymbol{x}^T P P^T \boldsymbol{x} = \|P^T \boldsymbol{x}\|^2 \geq 0$, with equality if and only if $P^T \boldsymbol{x} = 0$. Of course, M is symmetric because $M^T = (PP^T)^T = (P^T)^T P^T = PP^T = M$.

Solution 1.51. If S is any skew-symmetric matrix, that is $S^T = -S$, then $\boldsymbol{x}^T S \boldsymbol{x} = 0$, because

$$\boldsymbol{x}^T S \boldsymbol{x} = \left(\boldsymbol{x}^T S \boldsymbol{x}\right)^T = \boldsymbol{x}^T S^T \boldsymbol{x} = -\boldsymbol{x}^T S \boldsymbol{x},$$

which implies $\boldsymbol{x}^T S \boldsymbol{x} = 0$. (Here I'm using the simple fact that every 1×1 matrix is symmetric.) Thus, given any symmetric positive definite matrix A, we have

$$\boldsymbol{x}^T (A + S) \boldsymbol{x} = \boldsymbol{x}^T A \boldsymbol{x},$$

so that $A + S$ is positive definite, but not symmetric.

Solution 1.52. This exercise has really occurred before, but here's a direct solution. We have

$$\boldsymbol{e}_k^T A \boldsymbol{e}_k = \sum_{\ell=1}^{n} (\boldsymbol{e}_k^T)_\ell (A\boldsymbol{e}_k)_\ell = \sum_{\ell=1}^{n} \delta_{k\ell} (A\boldsymbol{e}_k)_\ell = (A\boldsymbol{e}_k)_k = \sum_{m=1}^{n} A_{km} \delta_{km} = A_{kk}.$$

Since $\boldsymbol{e}_k \neq 0$, we must have $A_{kk} = \boldsymbol{e}_k^T A \boldsymbol{e}_k > 0$.

Solution 1.53. We've just found an invertible lower triangular matrix L for which $A = LL^T$, so that $\boldsymbol{x}^T A \boldsymbol{x} = \|L^T \boldsymbol{x}\|^2 \geq 0$, with equality iff $\boldsymbol{x} = 0$.

Solution 1.54. The easiest way to do this is to use the *next* exercise! This was a deliberate trick to remind you that examination questions, and parts thereof, need not be attempted in the order set.

Solution 1.55. If the columns of A are linearly independent, then the matrix A^TA is positive definite, and therefore invertible (see earlier exercises). Thus

$$0 \neq \det A^TA = \det(R^TR) = (\det R)^2\,,$$

so that R is invertible also. (Here I've used the facts that $\det(PQ) = \det P \det Q$, $\det(P^T) = \det P$, and $\det P \neq 0$ if and only if P is invertible.)

Conversely, if R is invertible, then, setting $A = (\,\boldsymbol{a}_1 \quad \cdots \quad \boldsymbol{a}_n\,)$, we obtain

$$\|\sum_{k=1}^{n} x_k \boldsymbol{a}_k\|^2 = \boldsymbol{x}^T A^T A\boldsymbol{x} = \boldsymbol{x}^T R^T R\boldsymbol{x} = \|R\boldsymbol{x}\|^2 \geq 0,$$

with equality if and only if $\boldsymbol{x} = 0$. Hence the columns of A are linearly independent.

2. An Introduction to Eigendecompositions

In this section, our primary purpose is to prove that a real symmetric matrix A can be *orthogonally diagonalized*, that is, there exists an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ and a diagonal matrix $D \in \mathbb{R}^{n\times n}$ for which $D = Q^TAQ$. We need to borrow a theorem from `M2P1`:

Theorem 2.1. Let K be any closed and bounded subset of $\mathbb{R}^n$ and let $f\colon K \to \mathbb{R}$ be a continuous function. Then f is bounded and attains its bounds.

In other words, there are real numbers m and M such that

$$m \leq f(\boldsymbol{x}) \leq M, \qquad \text{for every } \boldsymbol{x} \in K.$$

Furthermore, there exist points $\boldsymbol{x}_m, \boldsymbol{x}_M \in K$ for which $f(\boldsymbol{x}_m) = m$ and $f(\boldsymbol{x}_M) = M$.

The next exercise should clarify the succeeding theorem.

Exercise 2.1. Let $\boldsymbol{u}$ be a unit vector and let $\boldsymbol{w}$ be any vector such that $\boldsymbol{w}^T\boldsymbol{v} = 0$ for every vector $\boldsymbol{v}$ orthogonal to $\boldsymbol{u}$. Prove that $\boldsymbol{w}$ is a multiple of $\boldsymbol{u}$.

Proposition 2.2. Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix and define $f(\boldsymbol{x}) = \boldsymbol{x}^TAvx$, let $K = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\| = 1\}$ be the unit sphere in n-dimensions, and let $\boldsymbol{u} \in K$ be a point at which f achieves its minimum value on K. Then $\boldsymbol{u}$ is an eigenvector for A. Further, its eigenvalue is the number $\boldsymbol{u}^TA\boldsymbol{u} \in \mathbb{R}$.

Proof. We have $f(\boldsymbol{u}) \leq f(\boldsymbol{x})$, for every unit vector $\boldsymbol{x}$. Let $\boldsymbol{v}$ be any unit vector orthogonal to vu and consider the function

$$g(\theta) = f(\boldsymbol{u}\cos\theta + \boldsymbol{v}\sin\theta), \qquad \theta \in \mathbb{R}.$$

Thus g is simply f restricted to the circle formed by the intersection of the unit sphere K and the 2-dimensional subspace spanned by $\boldsymbol{u}, \boldsymbol{v}$. Now

$$\begin{aligned} g(\theta) &= f(\boldsymbol{u}\cos\theta + \boldsymbol{v}\sin\theta) \\ &= (\boldsymbol{u}\cos\theta + \boldsymbol{v}\sin\theta)^T A\,(\boldsymbol{u}\cos\theta + \boldsymbol{v}\sin\theta) \\ &= \cos^2\theta \boldsymbol{u}^T A\boldsymbol{u} + \boldsymbol{v}^T A\boldsymbol{v}\sin^2\theta \boldsymbol{u}^T A\boldsymbol{v}\sin 2\theta. \end{aligned}$$

Hence

$$g'(\theta) = \left(\boldsymbol{v}^T A\boldsymbol{v} - \boldsymbol{u}^T A\boldsymbol{u}\right)\sin 2\theta + 2\cos^2\theta \boldsymbol{u}^T A\boldsymbol{v}.$$

Now we have $g(0) \leq g(\theta)$, for all $\theta \in \mathbb{R}$, which implies, recalling A is symmetric,

$$0 = g'(0) = 2\boldsymbol{u}^T A\boldsymbol{v} = 2(A\boldsymbol{u})^T\boldsymbol{v}.$$

Since $\boldsymbol{v}$ can be *any* unit vector orthogonal to $\boldsymbol{u}$, this implies $A\boldsymbol{u}$ is a multiple of $\boldsymbol{u}$, as required – this is Exercise 2.1. □

Now we can prove our main result.

Theorem 2.3. Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. Then there exists an orthogonal matrix $Q = (\,\boldsymbol{q}_1 \quad \boldsymbol{q}_2 \quad \cdots \quad \boldsymbol{w}_n\,) \in \mathbb{R}^{n\times n}$ such that $D := Q^T AQ$ is a diagonal matrix. Furthermore, we have $A\boldsymbol{q}_j = D_{jj}\boldsymbol{q}_j$, for $1 \leq j \leq n$.

Proof. We use induction on the size n of the matrix. The theorem is plainly true when $n = 1$.

Pick any unit vector $\boldsymbol{u}_1$ at which $f(\boldsymbol{x}) = vx^T A\boldsymbol{x}$ attains its minimum value on the unit sphere K. By Proposition 2.2, $\boldsymbol{u}_1$ is an eigenvector of A, say $A\boldsymbol{u}_1 = \lambda_1\boldsymbol{u}_1$. Now choose any unit vectors $u_2, u_3, \ldots, u_n$ orthogonal to vu_1 and define the orthogonal matrix $Q_1 = (\,\boldsymbol{u}_1 \quad \cdots \quad \boldsymbol{u}_n\,) \in \mathbb{R}^{n\times n}$. Then

$$(Q_1^T AQ_1)_{jk} = \boldsymbol{u}_j^T A\boldsymbol{u}_k, \qquad 1 \leq j,k \leq n,$$

whence

$$(Q_1^T AQ_1)_{1j} = (Q^T AQ)_{j1} = \boldsymbol{u}_j^T A\boldsymbol{u}_1 = \lambda_1\boldsymbol{u}_j^T\boldsymbol{u}_1 = \delta_{j1},$$

or

$$A^{(1)} = Q_1^T AQ_1 = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & B & \\ 0 & & & \end{pmatrix},$$

where $B \in \mathbb{R}^{(n-1)\times(n-1)}$ is, of course, still symmetric. By induction hypothesis, there is an orthogonal $Q_2 \in \mathbb{R}^{(n-1)\times(n-1)}$ for which $E = Q_2^T BQ_2$ is

diagonal. Hence, if we define

$$Q_3 = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & Q_2 & \\ 0 & & & \end{pmatrix},$$

then

$$Q_3^T A^{(1)} Q_3 = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & E & \\ 0 & & & \end{pmatrix} \equiv D,$$

and Q_3 is orthogonal (check this!). Setting $Q = Q_1 Q_3$, we have the required orthogonal diagonalization. □

As a simple consequence of this result, the eigenvalues of A, being the minimum values of A on certain unit spheres, must be real numbers.

3. Least Squares Problems II

We've already extended the definition of inner product once. In this section, we become more abstract still.

Definition 3.1. Let V be a real vector space. An *inner product* is just a function $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$ satisfying the following conditions:

(i) $\langle \lambda x + \mu y, z \rangle = \lambda \langle x, z \rangle + \mu \langle y, z \rangle$;
(ii) $\langle y, x \rangle = \langle x, y \rangle$.
(iii) $\langle x, x \rangle \geq 0$, with equality if and only if $x = 0$.

Every inner product is associated with a *norm* defined by

$$\|x\| = +\sqrt{\langle x, x \rangle}, \qquad x \in V. \tag{3.1}$$

These definitions are in their most general form, but almost all of the inner products we'll consider fall into two categories. Because our vectors can now be functions, we shall follow the convention of advanced numerical analysis and *not* use bold type to indicate vectors in this section.

(i) **Discrete inner products:** Given positive numbers $w_0, w_1, \ldots, w_n$, we define

$$\langle f, g \rangle = \sum_{k=0}^{n} w_k f(x_k) g(x_k). \tag{3.2}$$

The numbers $w_0, \ldots, w_n$ are usually called *weights*.

(ii) **Continuous inner products:** Given a continuous function $w(x)$ that is non-negative on the interval $[a, b]$ and has at most finitely many

zeros in that interval, we define

$$\langle f, g \rangle = \int_a^b w(x) f(x) g(x)\, dx. \tag{3.3}$$

We usually call $w(x)$ the weight function.

The justification for studying the above is simple: they arise in many different areas of mathematics and its applications. The general definition allows us to treat all of these special cases at once.

Much of the rest of this section restates theory you've already met, but stated in the more general form. You should be aware that the properties of inner products covered earlier all generalize to this setting. In particular, the Cauchy-Scwarz inequality holds, its proof being unchanged.

Example 3.1. Let $w(x)$ be any positive continuous function defined on the interval $[a, b]$. Then, for any continuous function $f(x)$, the Cauchy-Schwarz inequality implies the bound

$$\left(\int_a^b w(x) f(x)\, dx \right)^2 \leq \int_a^b w(t)\, dt \int_a^b w(x) f(x)^2\, dx,$$

with equality if and only if $f(x)$ is a constant function. (Just set $g(x) = 1$ in the inequality.)

We can now state the abstract form of the least squares problem: Let V be an inner product space and let U be a subspace of V with basis $\{\phi_1, \ldots, \phi_m\}$. Given any point $v \in V$, find a best approximation $u^* \in U$.

Example 3.2. To continue the previous example, we let U be the vector space of polynomials of degree $m - 1$. Given any function $f \in V$, we want to construct a polynomial $q = p^* \in U$ minimizing the integral

$$\int_a^b (f(x) - q(x))^2\, dx, \qquad q \in U.$$

In other words, we want to find a vector $\lambda^* = [\lambda_1^*, \ldots, \lambda_m^*] \in \mathbb{R}^m$ minimizing the quadratic

$$E(\lambda) = \|v - \sum_{k=1}^m \lambda_k \phi_k\|^2, \tag{3.4}$$

for all $\lambda = [\lambda_1, \ldots, \lambda_m]^T \in \mathbb{R}^m$. Expanding the quadratic using the inner product, we obtain

$$E(\lambda) = \|v\|^2 - 2 \sum_{k=1}^m \lambda_k \langle v, \phi_k \rangle + \sum_{j=1}^m \sum_{k=1}^m \lambda_j \lambda_k \langle \phi_j, \phi_k \rangle. \tag{3.5}$$

It's useful to state 3.5 in matrix form: let G be the $m \times m$ matrix given by

$$G_{jk} = \langle \phi_j, \phi_k \rangle, \qquad 1 \le j, k \le m, \tag{3.6}$$

and define the vector $\mu \in \mathbb{R}^m$ by

$$\mu_j = \langle v, \phi_j \rangle, \qquad 1 \le j \le m. \tag{3.7}$$

Then

$$E(\lambda) = \|v\|^2 - 2\lambda^T \mu + \lambda^T G \lambda. \tag{3.8}$$

The entire theory of least squares rests on analysis of this quadratic. It is evident that any stationary point must satisfy the equation

$$\nabla E(\lambda^*) = 0 \qquad \text{where } \nabla E(\lambda) = 2(G\lambda - \mu), \tag{3.9}$$

or, equivalently,

$$G\lambda^* = \mu. \tag{3.10}$$

The equations forming (3.10) are still called the "normal equations". The form of G is sufficiently important to deserve a name: it's called a *Gram matrix*, and we sometimes write $G(\phi_1, \ldots, \phi_m)$ to indicate its dependence on $\phi_1, \ldots, \phi_m$. In fact the normal equations are necessary and sufficient for λ^* to be a global minimum of the quadratic E in this more general setting, as we shall soon see.

Lemma 3.1. Let $\phi_1, \ldots, \phi_m$ be any vectors in an inner product space and let $G = G(\phi_1, \ldots, \phi_m)$ be the corresponding Gram matrix. The G is non-negative definite and symmetric. Furthermore, it is positive definite if and only if the vectors $\phi_1, \ldots, \phi_m$ are linearly independent.

Proof. Let $h \in \mathbb{R}^m$. We have

$$h^T G h = \sum_{j=1}^m \sum_{k=1}^m h_j h_k \langle \phi_j, \phi_k \rangle = \langle \sum_{j=1}^m h_j \phi_j, \sum_{k=1}^m h_k \phi_k \rangle = \| \sum_{k=1}^m h_k \phi_k \|^2 \ge 0,$$

with equality if and only if $\sum_{k=1}^m h_k \phi_k = 0$. If the vectors $\phi_1, \ldots, \phi_m$ are linearly independent, then the last equation can only hold when every coefficient $h_k = 0$, in which case G is positive definite. □

Corollary 3.2. Let $\nabla E(\lambda^*) = 0$. Then $E(\lambda^* + h) \ge E(\lambda^*)$ for every $h \in \mathbb{R}^m$. Thus any solution of the normal equations provides a global minimum for E.

Proof. We have

$$\begin{aligned} E(\lambda^* + h) &= \|v\|^2 - 2\mu^T(\lambda^* + h) + (\lambda^* + h)^T G(\lambda^* + h) \\ &= E(\lambda^*) + 2h^T(G\lambda^* - \mu) + h^T G h \end{aligned}$$

$$\begin{aligned} &= E(\lambda^*) + h^T \nabla E(\lambda^*) + h^T G h \\ &= E(\lambda^*) + h^T G h \\ &\geq E(\lambda^*), \end{aligned}$$

by Lemma 3.1. □

The next result is closely related to Corollary 3.2. Geometrically, it states that the least squares solution $u^* = \sum_{k=1}^m \lambda_k^* \phi_k$ is the orthogonal projection of v onto the subspace U; we're simply dropping a perpendicular.

Theorem 3.3. The vector $u^* = \sum_{k=1}^m \lambda^* \phi_k \in U$ minimizes the quadratic E defined in (3.4) if and only if

$$\langle v - u^*, u \rangle = 0 \qquad \text{for every } u \in U.$$

Proof. The vector $u^* \in U$ minimizes (3.4) if and only if $\nabla E(\lambda^*) = 0$, by Corollary 3.2. Setting $u = \sum_{k=1}^m \lambda_k \phi_k$, we have

$$\langle v - u^*, u \rangle = \lambda^T \mu - \lambda^T G \lambda^* = \lambda^T \nabla E(\lambda^*) = 0,$$

using (3.7) and (3.9). □

Example 3.3. Let $V = C[0,1]$, the vector space of continuous, real-valued functions defined on the interval $[0,1]$. We let the inner product be given by

$$(f,g) = \int_0^1 f(x) g(x)\, dx$$

and the corresponding norm is

$$\|f\|^2 = \int_0^1 f(x)^2\, dx.$$

Our basis functions will be $\phi_k(x) = x^k$, for $0 \leq k \leq n$, and our problem is to find numbers $\lambda_0^*, \ldots, \lambda_n^*$ minimizing the integral

$$E(\lambda) = \|f - \sum_{k=0}^n \lambda_k \phi_k\|^2 = \int_0^1 \Big| f(x) - \sum_{k=0}^n \lambda_k x^k \Big|^2\, dx.$$

The normal equations are

$$G\lambda^* = \mu,$$

where the elements of the Gram matrix are

$$G_{jk} = \int_0^1 x^{j+k}\, dx = \frac{1}{j+k+1}, \qquad 0 \leq j,k \leq n,$$

and

$$\mu_j = \int_0^1 f(x) x^j\, dx, \qquad 0 \leq j \leq n.$$

This Gram matrix is usually called a *Hilbert matrix*. Hilbert matrices are far too ill-conditioned for practical work unless n is tiny; our first glimpse of the fact that normal equations are usually to be avoided unless the basis $\{\phi_1, \ldots, \phi_m\}$ is chosen rather carefully.

Example 3.4. Let A be a $m \times n$ real matrix whose columns are denotes $a_1, a_2, \ldots, a_n$. In general, we cannot solve the linear system

$$A\lambda = v,$$

where $\lambda \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$. However, it is sometimes necessary to construct an approximate solution. One approach is to choose $\lambda^* \in \mathbb{R}^n$ to minimize the quadratic

$$E(\lambda) = \|v - A\lambda\|^2 = \|v - \sum_{k=1}^{n} \lambda_k a_k\|^2, \qquad \lambda = [\lambda_1, \ldots, \lambda_n]^T \in \mathbb{R}^n.$$

The normal equations for this least squares problem are $G\lambda^* = \mu$, where

$$G_{jk} = a_j^T a_k \quad \text{and} \quad \mu_j = a_j^T v \quad \text{for} \quad 1 \le j, k \le n.$$

In other words, we have the system

$$A^T A\lambda^* = A^T v.$$

Again, these equations are often too ill-conditioned to be of practical value.

We have observed that the Gram matrix $G(\phi_1, \ldots, \phi_m)$ can often have an unsuitably large condition number – after all, the horrid Hilbert matrix is a Gram matrix. One remedy for this problem is to choose an orthogonal basis $\psi_1, \ldots, \psi_m$ for the space spanned by $\phi_1, \ldots, \phi_m$, because then the new Gram matrix $G(\psi_1, \ldots, \psi_m)$ is diagonal and the solution of the normal equations reduces to m divisions. The *Gram-Schmidt* algorithm provides a poor tool for constructing such orthogonal bases. Fortunately, there's a much more accurate algorithm that requires less computation also. This is the *three-term recurrence relation of orthogonal polynomials*, which is studied in the next section.

4. Orthogonal Polynomials

In this section our inner product will either be discrete

$$\langle f, g \rangle = \sum_{k=1}^{n} w_k f(x_k) g(x_k), \tag{4.1}$$

where $(w_k)_{k=1}^n$ is some set of positive numbers, or continuous

$$\langle f, g \rangle = \int_a^b w(x) f(x) g(x)\, dx, \tag{4.2}$$

where the weight function w is a non-negative continuous function on the open interval (a, b) with at most finitely many zeros in (a, b). Our first topic is to construct monic orthogonal polynomials $\phi_0, \ldots, \phi_n$ from the monomials $1, x, \ldots, x^n$, where "monic" simply means that the coefficient of the highest degree term is one. The whole point of orthogonal polynomials is their simplification of the least squares problem: the polynomial $p^*(x)$ minimizing $\|f - p\|$ for all $p \in \mathbb{P}_n$ is given by the simple formula

$$p^*(x) = \sum_{k=0}^{n} \left(\frac{\langle f, \phi_k \rangle}{\langle \phi_k, \phi_k \rangle} \right) \phi_k(x),$$

because the Gram matrix

$$G(\phi_0, \phi_1, \ldots, \phi_n) = \begin{pmatrix} \|\phi_0\|^2 & & & \\ & \|\phi_1\|^2 & & \\ & & \ddots & \\ & & & \|\phi_n\|^2 \end{pmatrix}$$

is diagonal.

Exercise 4.1. Is $4x^2 + 2x + 1$ monic? Is $x^2 + 3x + 1$ monic?

But how do we calculate orthogonal polynomials? For hand computation, such as examination problems, the Gram-Schmidt algorithm suffices.

The key trick is the observation that $\phi_{n+1}(x) - x\phi_n(x)$ is a polynomial of degree n, so that

$$\phi_{n+1}(x) - x\phi_n(x) = \sum_{k=0}^{n} c_k \phi_k(x). \tag{4.3}$$

If $\phi_0, \ldots, \phi_n$ are orthogonal polynomials, then

$$c_j \langle \phi_j, \phi_j \rangle = \langle \phi_{n+1} - x\phi_n, \phi_j \rangle, \qquad 0 \le j \le n. \tag{4.4}$$

In fact, all but two of these coefficients vanish.

Theorem 4.1. Orthogonal polynomials satisfy the *three term recurrence relation*

$$\phi_{n+1}(x) = (x - a_n)\phi_n(x) - b_n \phi_{n-1}(x), \qquad n \ge 1, \tag{4.5}$$

where

$$a_n = \frac{\langle x\phi_n, \phi_n \rangle}{\|\phi_n\|^2} \qquad \text{and} \qquad b_n = \frac{\|\phi_n\|^2}{\|\phi_{n-1}\|^2}. \tag{4.6}$$

Proof. Equation (4.4) implies the relation

$$c_j \|\phi_j\|^2 = -\langle \phi_n, x\phi_j \rangle, \qquad 0 \le j \le n,$$

because ϕ_{n+1} is orthogonal to every polynomial of degree n. Furthermore,

$x\phi_j(x)$ is a polynomial of degree $j+1$, so that c_j is zero unless $j \in \{n-1, n\}$. Hence

$$c_{n-1} = -\frac{\langle \phi_n, x\phi_{n-1} \rangle}{\|\phi_{n-1}\|^2} \qquad \text{and} \qquad c_n = -\frac{\langle \phi_n, x\phi_n \rangle}{\|\phi_n\|^2}.$$

Finally, we obtain the relations

$$\langle \phi_n, x\phi_{n-1} \rangle = \langle \phi_n, x\phi_{n-1} - \phi_n \rangle - \|\phi_n\|^2 = -\|\phi_n\|^2,$$

because $x\phi_{n-1} - \phi_n$ is a polynomial of degree $n-1$, and therefore orthogonal to ϕ_n. A simple algebraic rearrangement then provides (4.5) and (4.6). □

Theorem 4.1 requires ϕ_0 and ϕ_1 initially. Of course, $\phi_0(x) \equiv 1$. For ϕ_1, we must have $\phi_1(x) = x - a_0$, say, and $\langle \phi_0, \phi_1 \rangle = 0$. Thus $a_0 = \langle x, \phi_0 \rangle / \|\phi_0\|^2$, which satisfies the recurrence relation (4.5) if we define $\phi_{-1}(x) = 0$ and $b_0 = 0$.

I've already mentioned that the three-term recurrence relation is suitable for use in floating point arithmetic. However, it enjoys another advantage: to form ϕ_n requires the calculation of only $\mathcal{O}(n)$ inner products. However, Gram-Schmidt requires $\mathcal{O}(n^2)$ operations, a severe disadvantage unless n is tiny. This latter case includes most examination questions, so it's useful to give some examples of Gram-Schmidt in action. The next lemma provides a very simple pair of results that are nevertheless rather useful.

Lemma 4.2. We shall say that $f(x)$ is *even* if $f(-x) = f(x)$, for all x. We'll call it an *odd* function if $f(-x) = -f(x)$, for all x. Then, for any positive number A, we have

$$\int_{-A}^{A} f(x)\,dx = 2\int_{0}^{A} f(x)\,dx,$$

when $f(x)$ is an even function, and

$$\int_{-A}^{A} f(x)\,dx = 0,$$

when $f(x)$ is an odd function.

Proof. If $f(x) = f(-x)$, for all x, then

$$\int_{-A}^{A} f(x)\,dx = \int_{-A}^{0} f(x)\,dx + \int_{0}^{A} f(x)\,dx = \int_{0}^{A} \left(f(-x) + f(x)\right)\,dx = 2\int_{0}^{A} f(x)\,dx.$$

A similar manipulation proves that the integral $\int_{-A}^{A} f = 0$ when $f(x)$ is an odd function. □

Example 4.1. Let the inner product be

$$\langle f, g \rangle = \int_{-1}^{1} f(x)g(x)\,dx.$$

It is easily checked that $\phi_1(x) = x$. Let us calculate ϕ_2 from first principles: we write $\phi_2(x) = x^2 + \alpha x + \beta$ and note the equations $0 = \langle \phi_0, \phi_2 \rangle = \langle \phi_1, \phi_2 \rangle$, that is

$$0 = \int_{-1}^{1} x^2 + \alpha x + \beta \, dx = \int_{-1}^{1} x^3 + \alpha x^2 + \beta x \, dx.$$

Because the integral of an odd function vanishes on $[-1, 1]$, we obtain

$$\alpha = 0 \qquad \text{and} \qquad \beta = -(1/2) \int_{-1}^{1} x^2 \, dx = \int_{0}^{1} x^2 \, dx = 1/3.$$

Thus $\phi_2(x) = x^2 - 1/3$. The reader is encouraged to calculate ϕ_3 and ϕ_4 using (i) the Gram-Schmidt algorithm and (ii) the three term recurrence relation; only masochists will prefer the former.

Exercise 4.2. Let the inner product be

$$\langle f, g \rangle = \int_{-2}^{2} x^6 f(x) g(x) \, dx.$$

Compute the monic orthogonal polynomials $\phi_k(x)$, for $0 \leq k \leq 3$.

Example 4.2. Show that the polynomials $T_k(x) = \cos(k \cos^{-1} x)$, $-1 \leq x \leq 1$, are orthogonal with respect to the inner product

$$(f, g) = \int_{-1}^{1} f(x) g(x) (1 - x^2)^{-1/2} \, dx.$$

We have

$$\langle T_j, T_k \rangle = \int_{-1}^{1} \cos(j \cos^{-1} x) \cos(k \cos^{-1} x) (1 - x^2)^{-1/2} \, dx = \int_{0}^{\pi} \cos(j\theta) \cos(k\theta) \, d\theta,$$

using the change of variable $x = \cos\theta$. It is not hard to show that this integral vanishes when $j \neq k$, but you should check this.

These are the *Chebyshev polynomials*. We shall see another of their surprising properties in the section on polynomial interpolation.

Exercise 4.3. Prove that the Chebyshev polynomials satisfy the three term recurrence relation

$$T_{n+1}(x) + T_{n-1}(x) = 2x T_n(x), \qquad n \geq 1.$$

(Hint: recall that $\cos(n+1)\theta + \cos(n-1)\theta = 2 \cos\theta \cos n\theta$.)

Some inner products are defined by integrals over infinite intervals.

Example 4.3. Let

$$\langle f, g \rangle = \int_{0}^{\infty} e^{-x} f(x) g(x) \, dx.$$

Let's compute the monic orthogonal polynomials ϕ_0, ϕ_1 and ϕ_2. A useful trick for this example is the identity

$$\int_0^\infty e^{-x} x^m \, dx = m!, \tag{4.7}$$

for any nonnegative integer m. (To show this, let

$$I_n = \int_0^\infty e^{-x} x^n \, dx$$

and use integration by parts to show that $I_n = nI_{n-1}$. Since $I_0 = 1$, a simple induction completes the derivation.)

We set $\phi_0(x) \equiv 1$ and $\phi_1(x) = x - a$. We determine a using the orthogonality relation $0 = \langle 1, x-a \rangle = 1-a$, or $a = 1$. We now set $\phi_2(x) = x^2+bx+c$ and note that

$$0 = \langle 1, x^2 + bx + c \rangle = 2 + b + c$$

and

$$0 = \langle x, x^2 + bx + c \rangle = 6 + 2b + c.$$

Solving these linear equations yields $\phi_2(x) = x^2 - 4x + 2$.

Solution 4.1. The polynomial $4x^2 + 2x + 1$ is not monic, but $x^2 + 3x + 1$ is monic.

Solution 4.2. We set $\phi_0(x) = 1$ and $\phi_1(x) = x - a$. We see that $a = 0$, because of the orthogonality relation

$$0 = \langle 1, \phi_1 \rangle = \int_{-2}^2 x - a \, dx = 4a.$$

Thus $\phi_1(x) = x$. Setting $\phi_2(x) = x^2 + bx + c$, we have

$$0 = \langle x, x^2 + bx + c \rangle = \int_{-2}^2 x^3 + bx^2 + cx \, dx = 2b \int_0^2 x^2 \, dx = 16b/3,$$

whence $b = 0$. To determine c, we use

$$\langle 1, x^2 + c \rangle = \int_{-2}^2 x^2 + c \, dx = 2 \int_0^2 x^2 \, dx + 8c,$$

or $c = -(1/4) \int_0^2 x^2 \, dx = -2/3$. Thus $\phi_2(x) = x^2 - 2/3$.

Solution 4.3. The definition $T_k(x) = \cos(k \cos^{-1} x)$ implies that, when $x = \cos\theta$, the claimed recurrence is an immediate consequence of the given trigonometric identity.

5. Polynomial Interpolation

Let $z_0, z_1, \ldots, z_n$ be different complex numbers and let $f_0, \ldots, f_n$ be any given complex numbers. We want to construct a polynomial p of degree n for which $p(z_j) = f_j$, $0 \leq j \leq n$. Such a polynomial is called an *interpolating polynomial*, and we say that p *interpolates* the data $\{(z_j, f_j) : j = 0, 1, \ldots, n\}$. We shall let $\mathbb{P}_n$ denote the vector space of polynomials of degree n.

Exercise 5.1. The quadratic $p(z) = (z^2 - \pi^2/4)/(-\pi^2/4)$ interpolates $f(z) = \cos z$ at the points $\{-\pi/2, 0, \pi/2\}$.

Lemma 5.1. Let

$$\ell_j(z) = \prod_{k=0, k\neq j}^{n} \frac{z - z_k}{z_j - z_k}, \qquad 0 \leq j \leq n. \tag{5.1}$$

Then $\ell_r(z_s) = \delta_{rs}$, $0 \leq r, s, \leq n$ and $\ell_r \in \mathbb{P}_n$.

Proof. By construction, $\ell_r(z_s) = 0$ when $r \neq s$, because the product in (5.1) contains the term $(z - z_s)$. However, $\ell_r(z_r) = 1$, because then every term in (5.1) occurs in both numerator and denominator. □

These polynomials $\ell_0, \ell_1, \ldots, \ell_n$ are useful because they allow us to write down a very simple expression for the polynomial interpolant.

Proposition 5.2. The interpolating polynomial $p \in \mathbb{P}_n$ for the data $\{(z_j, f_j) : 0 \leq j \leq n\}$ is given by

$$p(z) = \sum_{j=0}^{n} f_j \ell_j(z), \qquad z \in \mathbb{C}. \tag{5.2}$$

Proof. Equation 5.1 implies $p(z_k) = \sum_{j=0}^{n} f_j \delta_{jk} = f_k$, $0 \leq k \leq n$. □

Equation 5.2 is called the *Lagrange form of the interpolating polynomial*. Unfortunately, the Lagrange form is almost useless in practical work, although it's sometimes useful for theoretical work. It can also be useful in examination questions.

Example 5.1. The quadratic polynomial satisfying $p(0) = \alpha$, $p(1) = \beta$ and $p(4) = \gamma$ is

$$p(z) = \alpha \frac{(z-1)(z-4)}{(0-1)(0-4)} + \beta \frac{z(z-4)}{(1-0)(1-4)} + \gamma \frac{z(z-1)}{(0-4)(1-4)}.$$

Uniqueness of the interpolant is easily settled.

Uniqueness requires a simple lemma.

Lemma 5.3. Let $p(z) = a_0 + a_1 z + a_2 z^2 + \cdots + a_n z^n$, where $z \in \mathbb{C}$ and

$a_0, a_1, \ldots, a_n \in \mathbb{C}$. Then $p(z)$ has at most n distinct zeros in $\mathbb{C}$ unless $a_0 = a_1 = \cdots = a_n = 0$.

Proof. The lemma is plainly true when $n = 0$ or $n = 1$. We then proceed by induction. Thus let us assume that every polynomial of degree less than n has at most n different zeros, unless every coefficient is zero. Given any polynomial $p(z)$ of degree $n + 1$, *either* p has a root, say $p(w) = 0$, *or* p has no roots. If the latter condition is valid, then there's nothing further to demonstrate. If the former is valid, then $(z - w)$ is a factor of $p(z)$. Thus we can write $p(z) = q(z)(z - w)$, and the roots of p are w and the roots of q. However, by induction hypothesis, q can have at most n different roots. Thus p can have, in total, at most $n + 1$ different roots. □

Aside: The last lemma is a very simple version of the great Fundamental Theorem of Algebra: a polynomial of degree n with complex coefficients has exactly n complex zeros if we count multiple zeros multiply. (Thus $(z - 2)^2$ has two zeros.) This theorem will be proved in the Complex Analysis course next term.

Proposition 5.4. There is exactly one interpolating polynomial $p \in \mathbb{P}_n$ when the points $z_0, z_1, \ldots, z_n$ are distinct.

Proof. Existence was shown in Proposition 5.2, so we address uniqueness. Thus let p and q be interpolating polynomials of degree n. Their difference $p - q$ is a polynomial of degree n that vanishes at the $n + 1$ different points $z_0, \ldots, z_n$. Hence $p - q$ vanishes identically, using the last lemma. □

Exercise 5.2. Let $h > 0$ and let $p \in \mathbb{P}_2$ be the quadratic interpolating f at $\{-h, 0, h\}$. Show that

$$\int_{-h}^{h} p(x)\,dx = \frac{h}{3}\Big(f(-h) + 4f(0) + f(h)\Big),$$

which you should recognize as Simpson's rule.

Example 5.2. You've already met the Lagrange form when computing partial fractions. Let $w_0, w_1, \ldots, w_n$ be different complex numbers. We shall compute the scalars $\alpha_0, \alpha_1, \ldots, \alpha_n$ in the partial fraction decomposition

$$\frac{1}{(z - w_0)(z - w_1)\cdots(z - w_n)} = \sum_{j=0}^{n} \frac{\alpha_j}{z - w_j}.$$

Let us set $f(z) \equiv 1$. The Lagrange form of the polynomial interpolating f at $w_0, w_1, \ldots, w_n$ is

$$1 = \sum_{j=0}^{n} \ell_j(z),$$

by 5.2. Dividing both sides by $(z-w_0)\cdots(z-w_n)$ yields the expression

$$\alpha_j = \Big(\prod_{k=0,k\neq j}^{n} (w_j - w_k) \Big)^{-1}, \qquad j = 0, 1, \ldots, n.$$

The Lagrange form of the interpolating polynomial is useful when n is small and in theoretical work. However, it is particularly inconvenient if we have constructed $p_{n-1} \in \mathbb{P}_{n-1}$ interpolating data $\{(z_j, f_j) : 0 \leq j \leq n-1\}$ and are then given a new datum (z_n, f_n), because we almost have to start the calculation from scratch. Fortunately a more compact form is available. The key idea is to let $p \in \mathbb{P}_n$ take the form

$$p_n(z) = p_{n-1}(z) + C(z-z_0)(z-z_1)\cdots(z-z_{n-1}), \qquad z \in \mathbb{C}. \tag{5.3}$$

We see that $p_n(z_j) = p_{n-1}(z_j) = f_j$, for $0 \leq j \leq n-1$, so we do not disturb our previous interpolant at these points. Of course we choose C to satisfy the equation

$$f_n = p_{n-1}(z_n) + C \prod_{k=0}^{n-1} (z_n - z_k). \tag{5.4}$$

Obviously C depends on f and $z_0, z_1, \ldots, z_n$. A traditional notation is

$$C = f[z_0, z_1, \ldots, z_n], \tag{5.5}$$

so that 5.3 becomes

$$p_n(z) = p_{n-1}(z) + f[z_0, z_1, \ldots, z_n](z-z_0)(z-z_1)\cdots(z-z_{n-1}). \tag{5.6}$$

The number $f[z_0, \ldots, z_n]$ is called a *divided difference*, because of the method used to calculate these numbers described below. Note that the coefficient of highest degree for p_n does not depend on the order in which we take the points. In other words, if we replace $z_0, z_1, \ldots, z_n$ by $z_{\pi 0}, z_{\pi 1}, \ldots, z_{\pi n}$, for any permutation π of the numbers $\{0, 1, \ldots, n\}$, then $f[z_{\pi 0}, \ldots, z_{\pi n}] = f[z_0, \ldots, z_n]$. Another way to see this is the following explicit expression for $f[z_0, \ldots, z_n]$, which is sometimes useful in theoretical work.

Proposition 5.5. We have

$$f[z_0, z_1, \ldots, z_n] = \sum_{j=0}^{n} \frac{f(z_j)}{\prod_{k=0,k\neq j}^{n} (z_j - z_k)}. \tag{5.7}$$

Further, $f[z_0, \ldots, z_n] = 0$ when f is a polynomial of degree less than n.

Proof. We just equate the coefficients of z^n in $p_n(z) = \sum_{j=0}^{n} f(z_j)\ell_j(z)$, using Proposition 5.2. Moreover, if $f(z) = z^\ell$ and $\ell < n$, then the coefficient of z^n in p_n is zero. But this highest degree coefficient is $f[z_0, \ldots, z_n]$. $\square$

Exercise 5.3. Show that

$$\sum_{j=0}^{n} \frac{z_j^m}{\prod_{k=0,k\neq j}^{n}(z_j - z_k)} = \delta_{mn},$$

for $m = 0, 1, \ldots, n$.

Recurring equation 5.6, and defining $f[z_0] = f(z_0)$, yields the explicit expression

$$p_n(z) = f[z_0] + f[z_0, z_1](z - z_0) + f[z_0, z_1, z_2](z - z_0)(z - z_1) + \cdots \\ + f[z_0, z_1, \ldots, z_n](z - z_0)(z - z_1)\cdots(z - z_{n-1}),$$

and this is called the *Newton form* of the interpolating polynomial.

It's **important** to understand that $f[z_0, \ldots, z_\ell]$ is the coefficient of highest degree for the polynomial $p_\ell \in \mathbb{P}_\ell$ interpolating the data $\{(z_k, f_k) : 0 \leq k \leq \ell\}$.

Example 5.3. The Newton form of the quadratic polynomial satisfying $p(0) = f(0)$, $p(1) = f(1)$ and $p(4) = f(4)$ is

$$p(z) = f[0] + f[0, 1]z + f[0, 1, 4]z(z - 1).$$

You'll see how to calculate the coefficients shortly.

The recursion used to calculate divided difference and justifying the suitability of their name is derived in the following key theorem.

Theorem 5.6. For any distinct complex numbers $z_0, z_1, \ldots, z_n, z_{n+1}$ the divided differences satisfy

$$f[z_0, \ldots, z_{n+1}] = \frac{f[z_0, \ldots, z_n] - f[z_1, \ldots, z_{n+1}]}{z_0 - z_{n+1}}. \tag{5.8}$$

Proof. We introduce two polynomials: (i) $p \in \mathbb{P}_n$ interpolates $\{(z_k, f_k) : 0 \leq k \leq n\}$, and (ii) $q \in \mathbb{P}_n$ interpolates $\{(z_k, f_k) : 1 \leq k \leq n + 1\}$. Thus the coefficients of highest degree for p and q are $f[z_0, \ldots, z_n]$ and $f[z_1, \ldots, z_{n+1}]$, respectively. The key **trick** is now the observation that the polynomial $r \in \mathbb{P}_{n+1}$ interpolating at all $n + 1$ points satisfies

$$r(z) = \frac{(z - z_{n+1})p(z) - (z - z_0)q(z)}{z_0 - z_{n+1}}, \tag{5.9}$$

because it is unique, by Proposition 5.4, and it is easily checked that the right hand side of (5.9) interpolates at $z_0, \ldots, z_{n+1}$: an exercise for the reader. Now the coefficient of highest degree in r is $f[z_0, \ldots, z_{n+1}]$, so equating the coefficients of highest degree is (5.9) yields (5.8). □

Exercise 5.4. Check that (5.9) holds when $n = 1$ and $x_k = k$.

Exercise 5.5. Let $f(z) = (\alpha - z)^{-1}$. Show that

$$f[z_0, z_1, \ldots, z_n] = \frac{1}{(\alpha - z_0)(\alpha - z_1)\cdots(\alpha - z_n)}.$$

Proposition 5.3 is extremely important. It's the basis of the algorithm for divided differences and the Newton form of the interpolating polynomial. We form the lower triangular matrix

$$\begin{pmatrix} x_0 & f(x_0) & & & & & \\ x_1 & f(x_1) & f[x_0,x_1] & & & & \\ x_2 & f(x_2) & f[x_1,x_2] & f[x_0,x_1,x_2] & & & \\ \vdots & \vdots & \vdots & \vdots & \ddots & & \\ x_{n-1} & f(x_{n-1}) & f[x_{n-2},x_{n-1}] & f[x_{n-3},x_{n-2},x_{n-1}] & \cdots & f[x_0,x_1,\ldots,x_{n-1}] & \\ x_n & f(x_n) & f[x_{n-1},x_n] & f[x_{n-2},x_{n-1},x_n] & \cdots & f[x_1,x_2,\ldots,x_n] & f[x_0,\ldots,x_n] \end{pmatrix}.$$

In practice, only the first two columns of this lower triangular matrix are stored. The diagonal elements are those needed for the Newton form of the interpolating polynomial, so it's usual for the second column to be overwritten by subsequent columns; that's $\mathcal{O}(n)$ rather than $\mathcal{O}(n^2)$ numbers to be stored. At completion, the second column should contain the diagonal elements of the matrix, that is

$$[f(x_0), f[x_0,x_1], f[x_0,x_1,x_2], f[x_0,x_1,x_2,x_3], \ldots, f[x_0,x_1,\ldots,x_{n-1}], f[x_0,\ldots,x_n]\,]^T.$$

A polynomial can be evaluated at any point using *Horner's rule* (nested multiplication), which we state here for completeness.

Algorithm for polynomial evaluation: Let $p(z) = a_0 + a_1 z + a_2 z^2 + \cdots a_n z^n$ and let α be any complex number. The following algorithm calculates the value $p(\alpha)$ of the polynomial at $z = \alpha$.

Set $s := a_n$.
For $k = n-1, n-2, \ldots, 1, 0$, set
$s := s * z + a_k$.

This is faster than naively evaluating $p(\alpha) = a_0 + a_1 * \alpha + a_2 * \alpha * \alpha + \cdots + a_n * \alpha * \cdots * \alpha$.

Exercise 5.6. Assume you're using a computer for which every arithmetic operation takes the same amount of time. Calculate (i) the number of operations required to apply Horner's rule, and (ii) the number of operations for the naive method.

What about the error in polynomial interpolation?

Theorem 5.7. Let $p \in \mathbb{P}_n$ interpolate f at n distinct complex numbers

$z_0, z_1, \ldots, z_n$. Then the error $e = f - p$ satisfies the equation

$$e(w) = f[z_0, z_1, \ldots, z_n, w] \prod_{k=0}^{n} (w - z_k), \qquad w \in \mathbb{C}. \tag{5.10}$$

Proof. If we add a new interpolation point z_{n+1}, then the Newton interpolating polynomial $q \in \mathbb{P}_{n+1}$ is given by

$$q(z) = p(z) + f[z_0, z_1, \ldots, z_n, z_{n+1}] \prod_{k=0}^{n} (z - z_k).$$

Hence

$$f(z_{n+1}) = p(z_{n+1}) + f[z_0, z_1, \ldots, z_n, z_{n+1}] \prod_{k=0}^{n} (z - z_k).$$

Since z_{n+1} can be *any* point, we can write $w = z_{n+1}$, which completes the proof. □

This result is of little use for error bounds unless we can bound $f[z_0, z_1, \ldots, z_n, w]$ from above in some way. Now the first mean value theorem implies the equation

$$f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f'(\xi),$$

for some point $\xi \in [x_0, x_1]$. There is an important result for divided differences generalizing this remark that's essentially a form of the mean value theorem you'll meet in real analysis. We shall use this relation to express the error in terms of the maximum modulus of the $(n+1)$st derivative of f.

Theorem 5.8. Let f have continuous $(n+1)$st derivative and let $x_0 < x_1 < \cdots < x_n$ be **real** numbers. Then there is a point $\xi \in [x_0, x_n]$ such that

$$f[x_0, x_1, \ldots, x_n] = \frac{f^n(\xi)}{n!}. \tag{5.11}$$

Proof. Let $p_n \in \mathbb{P}_n$ interpolate f at $x_0, \ldots, x_n$. Then the error function $e = f - p_n$ has at least $n+1$ zeros in $[x_0, x_n]$. Hence its derivative e' has at least n zeros in $[x_0, x_n]$, and its second derivative e'' has at least $n-1$ zeros. Continuing in this way, we deduce that $e^{(n)}$ has at last one zero, ξ say, in $[x_0, x_n]$. But then

$$0 = e^{(n)}(\xi) = f^{(n)}(\xi) - f[x_0, \ldots, x_n] n!,$$

as required. □

Corollary 5.9. Let f have continuous $(n+1)$st derivative and let $x_0, x_1, \ldots, x_n$ be different real numbers. If $p_n \in \mathbb{P}_n$ is the interpolating polynomial, then

the error $e_n = f - p_n$ satisfies

$$|e_n(x)| \leq \frac{M \prod_{k=0}^{n} |x - x_k|}{(n+1)!}, \qquad x \in [a, b], \tag{5.12}$$

where $M = \max\{|f^{(n+1)}(t)| : a \leq t \leq b\}$.

Proof. This is immediate from the last two theorems. □

Example 5.4. Rework the last example for general a and b.

Example 5.5. Let $f(x) = \cos x$. Rework the last example.

Thes examples might suggest that increasing the number of interpolation points always decreases the error. This is *not* so, as you may see in exercises.

Example 5.6. Let $f(x) = \exp(x)$ and let $a = -1/2$, $b = 1/2$. If the interpolation points are always contained within the interval $[-1/2, 1/2]$, then the error of interpolation satisfies

$$|e_n(x)| \leq \frac{e}{(n+1)!}, \qquad -1/2 \leq x \leq 1/2.$$

In other words, the error is *tiny*, what ever the choice of interpolation points. In fact, this is true whenever the function being interpolated is complex differentiable at every point of the complex plane. It is certainly **not** true for general functions, as we shall shortly see.

Equation (5.12) suggests the following problem: Find interpolation points $(x_k)_{k=0}^{n-1}$ minimizing

$$\max_{-1 \leq x \leq 1} \left(\prod_{k=0}^{n-1} |x - x_k| \right), \tag{5.13}$$

which occurs when we want to minimize upper bound (5.12) on the interval $[-1, 1]$. It is easy to see that equally spaced points are bad; try it on *Mathematica*. In fact, the minimum value of (5.13) occurs when

$$\prod_{k=0}^{n-1} x - x_k = 2^{1-n} \cos(n \cos^{-1} x).$$

This was discovered by the great Russian mathematician Chebyshev.

6. Gaussian quadrature

Suppose we need to calculate the integral

$$I = \int_a^b w(x) f(x)\, dx, \tag{6.1}$$

where the weight function w is positive and continuous. One approach is to choose different points $x_0, x_1, \ldots, x_n$ in the interval $[a, b]$ and to interpolate f at these points using a polynomial p of degree n. We can then let our approximation be

$$I_{\text{approx}} = \int_a^b w(x)p(x)\,dx. \tag{6.2}$$

Using the Lagrange polynomials provides a useful formula for I_{approx} in terms of the numbers $f(x_0), f(x_1), \ldots, f(x_n)$. We recall that the Lagrange polynomials are given by

$$\ell_j(x) = \prod_{k=0, k\neq j}^{n} \frac{x - x_k}{x_j - x_k}, \qquad 0 \leq j \leq n, \tag{6.3}$$

and satisfy

$$\ell_j(x_k) = \delta_{jk}, \qquad 0 \leq j, k \leq n. \tag{6.4}$$

Proposition 6.1. Let

$$w_k = \int_a^b w(x)\ell_k(x)\,dx, \qquad 0 \leq k \leq n. \tag{6.5}$$

Then

$$\int_a^b w(x)p(x)\,dx = \sum_{k=0}^{n} w_k p(x_k) \tag{6.6}$$

for every polynomial p of degree n.

Proof. An easy exercise. □

The next obvious question is where to choose the interpolation points $x_0, x_1, \ldots, x_n$. The Runge Phenomenon implies that equally spaced points can be disastrous for large n if f is a meromorphic function whose poles are sufficiently close to the interval $[a, b]$. One approach is to consider the error $I - I_{\text{approx}}$ when f is a polynomial of degree exceeding n. Thus $f - p$ is a polynomial vanishing at the points $x_0, x_1, \ldots, x_n$, and we can write

$$f(x) - p(x) = q(x)\pi(x), \tag{6.7}$$

where q is another polynomial and $\pi(x) = \prod_{k=0}^{n}(x - x_k)$. Hence

$$\begin{aligned} I - I_{\text{approx}} &= \int_a^b w(x)\Big(f(x) - p(x)\Big)\,dx \\ &= \int_a^b w(x)q(x)\pi(x)\,dx \\ &= \langle q, \pi \rangle, \end{aligned} \tag{6.8}$$

and this error vanishes if and only if q and π are orthogonal with respect to the ambient inner product. Therefore, if we can choose $x_0, \ldots, x_n$ so that $\pi = \phi_{n+1}$, then $I = I_{\text{approx}}$ when q is a polynomial of degree n, because ϕ_{n+1} is orthogonal to every polynomial of degree n. This amounts to interpolating at the zeros of ϕ_{n+1}. Of course, we do not yet know if these distinct zeros exist, and this is the subject of the next lemma.

Lemma 6.2. The orthogonal polynomial ϕ_n has n different zeros in the interval $[a, b]$.

Proof. Let σ denote the number of sign changes of ϕ_n in the interval $[a, b]$. If $\sigma < n$, then we can choose a polynomial p of degree less than n such that $p(x)\phi_n(x) > 0$ except at the (at most n) zeros of ϕ_n, as follows: Let $z_1, \ldots, z_\sigma$ denote the points at which ϕ_n changes sign and define $q(x) = (x - z_1) \cdots (x - z_\sigma)$; either q or $-q$ is a suitable choice for p. Hence $\langle p, \phi_n \rangle = \int_a^b w(x)p(x)\phi_n(x)\,dx > 0$. Since ϕ_n is orthogonal to polynomials of degree less than n, we conclude that p must have degree n, that is $\sigma = n$. Thus we have shown that ϕ_n changes sign at n different points in $[a, b]$. □

Theorem 6.3. Let $x_0, \ldots, x_n$ be the zeros of ϕ_{n+1} and let the weights $w_0, \ldots, w_n$ be chosen as in Proposition 6.1. Then

$$\int_a^b w(x)p(x)\,dx = \sum_{k=0}^{n} w_k p(x_k)$$

for every polynomial p of degree $2n + 1$.

Proof. We just follow the argument given before Lemma 6.2. □

An alternative proof is often given whose technique is interesting.

Proof. **(Second proof of Theorem 6.3)** Let p be any polynomial of degree $2n + 1$. Polynomial division yields the equation

$$p(x) = q(x)\phi_n(x) + r(x),$$

where q and r are polynomials of degree n. Thus

$$\sum_{k=0}^{n} w_k p(x_k) = \sum_{k=0}^{n} w_k q(x_k)\phi_n(x_k) = \sum_{k=0}^{n} w_k r(x_k)$$

and

$$\begin{aligned} \int_a^b w(x)p(x)\,dx &= \int_a^b w(x)q(x)\phi_n(x)\,dx + \int_a^b w(x)r(x) \\ &= \langle q, \phi_n \rangle + \int_a^b w(x)r(x)\,dx \end{aligned}$$

$$= \int_a^b w(x)r(x)\,dx.$$

However,

$$\sum_{k=0}^{n} w_k r(x_k) = \int_a^b w(x)r(x)\,dx,$$

by Proposition 6.1, since r has degree n. □

The method described in the statement of Theorem 6.3 is called "Gaussian quadrature". Quadrature is an old name for numerical integration, derived from the Latin *quadrare*, to make square, the idea being that one can approximate the area under a curve by squares (as in the Riemann sum approach to integration theory).

Example 6.1. Find points x_0, x_1 and weights w_0, w_1 such that

$$\int_{-1}^{1} f(x)\,dx = w_0 f(x_0) + w_1 f(x_1)$$

when f is a cubic.

Following Theorem 6.3, we set $n = 1$ and construct the orthogonal polynomial ϕ_2 of degree two. This calculation was the subject of Example 4.1, where it was shown that $\phi_2(x) = x^2 - 1/3$. Thus the zeros are $x_0 = -1/\sqrt{3}$ and $x_1 = 1/\sqrt{3}$. All that remains is to find weights such that

$$\int_{-1}^{1} f(x)\,dx = w_0 f(-1/\sqrt{3}) + w_1 f(1/\sqrt{3})$$

for every linear polynomial. There is no need to use the full apparatus of Proposition 6.1 in such a simple problem. Indeed, setting $f(x) = 1$ and $f(x) = x$ swiftly yields the equation $w_0 = w_1 = 1$.

Gaussian quadrature is best possible, in the sense that no (6.6) cannot hold for all polynomials of degree $2n + 2$. To see this, just let $p(x) = \prod_{k=0}^{n}(x - x_k)^2$. Thus $p \in \mathbb{P}_{2n+2}$ is strictly positive except at its zeros $x_0, \ldots, x_n$. Hence

$$\int_a^b w(x)p(x)\,dx > 0 = \sum_{k=0}^{n} w_k p(x_k).$$

Example 6.2. Find weights and nodes such that

$$\int_{-1}^{1} p(x)\,dx = w_0 p(x_0) + w_1 p(x_1) + w_2 p(x_2)$$

for every quartic polynomial p.

A simple calculation (left to the reader) shows that $\phi_3(x) = x^3 - (3/5)x$,

so that the nodes are 0 and $\pm\sqrt{3/5}$. To find the weights, we just let $p \in \{1, x, x^2\}$. We find that the Gaussian quadrature rule is then

$$\int_{-1}^{1} f(x)\,dx \approx \frac{1}{9}\Big(5f(-\sqrt{3/5}) + 8f(0) + 5f(\sqrt{3/5})\Big).$$

It is interesting to compare this with Simpson's rule

$$\int_{-1}^{1} f(x)\,dx \approx \frac{1}{3}\Big(f(-1) + 4f(0) + f(1)\Big),$$

which is only exact for cubics. When $f(x) = \cos x$, we find $\int_{-1}^{1} \cos x\,dx = 2\sin 1 = 1.6829$. The Simpson's rule approximation is $(1/3)(2\cos 1 + 4) = 1.6935$, whereas the Gaussian quadrature rule gives $(1/9)(10\cos\sqrt{3/5}+8) = 1.6830$.

Example 6.3. Quadrature rules play an important part in the derivation of Runge-Kutta methods. Suppose we want to solve $y' = f(t)$. We know that

$$y(t_n + h) = y(t_n) + \int_{t_n}^{t_n+h} f(\tau)\,d\tau = y(t_n) + h\int_0^1 f(t_n + h\sigma)\,d\sigma.$$

Now any quadrature rule provides an approximation of the form

$$\int_0^1 g(\sigma)\,d\sigma \approx \sum_{\ell=0}^{m} w_\ell g(\sigma_\ell).$$

Thus we obtain the recurrence

$$y(t_n + h) = y(t_n) + h\sum_{\ell=0}^{m} w_\ell f(t_n + h\sigma_\ell),$$

a method of Runge-Kutta type. The Gaussian quadrature rule is an obvious candidate and yields some useful Runge-Kutta methods. The general theory, when $y' = f(t, y)$, is sadly much more involved.

Example 6.4. Suppose we need to estimate the heat energy Q in a metal rod of length L. Thus

$$Q = C\int_0^L T(x)\,dx,$$

where $T(x)$ is the temperature in the rod and C is a constant. In practical work, Q must be estimated from a finite number of temperature measurements taken along the rod. Since we're approximating an integral, the Gaussian quadrature points are appropriate.

REFERENCES

G. H. Golub and C. F. V. Loan (1994), *Matrix Computations*, Johns Hopkins University Press.