1 GSW… Linear Algebra

Linear algebra is the algebra of linear equations: the term *linear* being used in the same sense as in linear functions, such as:

$$y = ax + c \tag{0.1}$$

which is the equation of a straight *line*.

Of course, if we only have one input variable x and one output variable y, then the entire subject of linear algebra would consist of finding solutions to the equations like the one above, and apart from inverting it to find:

$$x = \frac{y - c}{a} \tag{0.2}$$

there's not much else of interest you can do. Things only get interesting when we have a set of input variables x_i, and a set of output variables y_i, and each output depends on several of the inputs. In the general case, we can write something like:

$$\begin{aligned} y_1 &= a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n \\ y_2 &= a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \ldots + a_{2n}x_n \\ &\ldots \\ y_m &= a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \ldots + a_{mn}x_n \end{aligned} \tag{0.3}$$

which is more usually written in the form of column vectors **y** and **x** and a matrix **A**, where the elements of the vector **y** are the terms $y_1 \ldots y_m$, the elements of vector **x** are the terms $x_1 \ldots x_n$ and the elements of the matrix **A** are the terms $a_{11} \ldots a_{mn}$,

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \ldots \\ y_m \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ a_{31} & a_{32} & \ldots & a_{3n} \\ \ldots & \ldots & \ldots & \ldots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \ldots \\ x_n \end{bmatrix} \tag{0.4}$$

or in the notation of summation signs:

$$y_i = \sum_{j=1}^{n} a_{ij} x_j \tag{0.5}$$

This allows the whole set or *system* of *simultaneous equations* to be written in the much more convenient form of:

$$\mathbf{y} = \mathbf{A}\,\mathbf{x} \tag{0.6}$$

If we know the vector **x** and the matrix **A** then calculating the vector **y** is easy, however if we know **y** and **A** but not **x**, then we've got a much more difficult task. This is the basic problem of linear algebra: how to find **x** given **y** and **A**.

Now, if you've read the chapter about matrices, you'll know that there is such a thing as the inverse of a matrix, and that if we knew the inverse matrix of **A**, we could write:

$$\mathbf{A}^{-1}\mathbf{y} = \mathbf{A}^{-1}\mathbf{A}\,\mathbf{x} = \mathbf{I}\mathbf{x} = \mathbf{x} \tag{0.7}$$

where **I** is the unit matrix. At first sight, it might appear that all we have to do is find the inverse matrix $\mathbf{A}^{-1}$, and then calculate **x** according to:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{y} \tag{0.8}$$

All of which is fine, except…

- …if the matrix **A** is not square, then it doesn't have an inverse,
- …even if the matrix **A** is square, it might not have an inverse,
- …if the matrix **A** is large, then it can take a very large number of calculations (and hence a long time) to work out the inverse, and it would be nice not to have to do this.

Most of linear algebra is about finding ways to deal with these problems.

1.1 A Taxonomy of Linear Simultaneous Equations

The general set of linear equations

$$\mathbf{y} = \mathbf{A}\,\mathbf{x} \tag{0.9}$$

contains a number of different cases, all of which have to be considered separately. I'll try and list them, with examples and a few comments, here. We'll be looking at techniques to deal with them in later chapters.

1.1.1 A is Square and Invertible

This is the most straightforward and simplest case. If **A** is a square matrix, then the vector **x** and **y** have equal numbers of elements. That means we have the same number of unknowns (elements in **x**) as we have equations (elements in **y**). If **A** is invertible, then there will be a unique solution, all we have to do is find it. Finding the matrix inverse $\mathbf{A}^{-1}$ is one way, however it is rather time-consuming, especially for large matrices. A few short-cuts would be useful.

1.1.1.1 Triangular Matrices and Back-Substitution

Using triangular matrices and the process of back-substitution is one popular short-cut. Given the set of equations:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{0.10}$$

if we can find a way to convert them into the form:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ 0 & b_{22} & b_{23} & b_{24} \\ 0 & 0 & b_{33} & b_{34} \\ 0 & 0 & 0 & b_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{0.11}$$

where the matrix is an *upper-triangular* matrix (one in which all the terms underneath the main diagonal are zero) we can immediately see that we've modified the original set of equations into the form:

$$\begin{aligned} z_1 &= b_{11}x_1 + b_{12}x_2 + b_{13}x_3 + b_{14}x_4 \\ z_2 &= b_{22}x_2 + b_{23}x_3 + b_{24}x_4 \\ z_3 &= b_{33}x_3 + b_{34}x_4 \\ z_4 &= b_{44}x_4 \end{aligned} \tag{0.12}$$

which with a simple re-arrangement gives:

$$\begin{aligned} x_4 &= \frac{z_4}{b_{44}} \\ x_3 &= \frac{z_3 - b_{34}x_4}{b_{33}} \\ x_2 &= \frac{z_2 - b_{23}x_3 - b_{24}x_4}{b_{22}} \\ x_1 &= \frac{z_1 - b_{12}x_2 - b_{13}x_3 - b_{14}x_4}{b_{11}} \end{aligned} \tag{0.13}$$

and from these equations, it's easy to work out the elements of the vector **x** one at a time, starting with x_4, then using the value of x_4 to work out x_3, then using these values to work out x_2, and so on. This process is called *back-substitution*, since you start with the last element in **x** (in this case x_4) and work backwards to x_1.

The classic method to convert the equations into this form is known as *Gaussian elimination*. More details on how this works in the chapter on Gaussian Elimination.

There's an important slight variation on this idea as well. Suppose we could convert the system of equations into the form:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} b_{11} & 0 & 0 & 0 \\ b_{21} & b_{22} & 0 & 0 \\ b_{31} & b_{32} & b_{33} & 0 \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{0.14}$$

where the matrix is in now in *lower-triangular* form. Then we can equally easily calculate the elements of **x** using a similar process, but this time starting with the value of x_1, and working down to the last value x_4. This is called *forward-substitution*.

1.1.1.2 Substitution and Decomposition

Taking this idea one stage further: suppose we could convert the system of equations into something of the form:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} b_{11} & 0 & 0 & 0 \\ b_{21} & b_{22} & 0 & 0 \\ b_{31} & b_{32} & b_{33} & 0 \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ 0 & c_{22} & c_{23} & c_{24} \\ 0 & 0 & c_{33} & c_{34} \\ 0 & 0 & 0 & c_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{0.15}$$

If we can do this, then we can work out **x** using a sort of two-stage substitution process: first, find a vector **p** that satisfies:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} b_{11} & 0 & 0 & 0 \\ b_{21} & b_{22} & 0 & 0 \\ b_{31} & b_{32} & b_{33} & 0 \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} \tag{0.16}$$

using forward-substitution, then do a back-substitution to work out **x** from the equation:

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ 0 & c_{22} & c_{23} & c_{24} \\ 0 & 0 & c_{33} & c_{34} \\ 0 & 0 & 0 & c_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{0.17}$$

Why bother with a two-stage process? Because in many cases we can convert the matrix **A** into the product of a lower-triangular and upper-triangular matrix quite easily, and when we do, the values of $z_1 \ldots z_4$ are equal to the values of $y_1 \ldots y_4$, so we don't need to do anything to the vector **y** at all. If there are a lot of sets of equations to solve with different values of **y** and **x** but the same matrix **A** (quite common in practice), this can save a lot of time.

This process of converting a matrix into the product of two (or sometimes more than two) other matrices is known as *matrix decomposition*. The particular case illustrated here is known as an *LU-decomposition*, since the matrix **A** is being expressed as the product of a lower-triangular matrix **L** and an upper triangular matrix **U**. See the chapter on Matrix Decompositions for more about this and related methods.

1.1.1.3 Example of a Square Invertible System

Consider the equations:

$$\begin{aligned} -1 &= x_1 + 2x_2 - x_3 \\ 3 &= 2x_1 - x_2 - 2x_3 \\ 2 &= -x_1 - 2x_2 + 2x_3 \end{aligned} \tag{0.18}$$

writing these in matrix form gives:

$$\begin{bmatrix} -1 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -1 \\ 2 & -1 & -2 \\ -1 & -2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{0.19}$$

and this matrix can be inverted:

$$\begin{bmatrix} 1 & 2 & -1 \\ 2 & -1 & -2 \\ -1 & -2 & 2 \end{bmatrix}^{-1} = \begin{bmatrix} 1.2 & 0.4 & 1 \\ 0.4 & -0.2 & 0 \\ 1 & 0 & 1 \end{bmatrix} \tag{0.20}$$

which allows a unique solution to be readily determined:

$$\begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -1 \\ 2 & -1 & -2 \\ -1 & -2 & 2 \end{bmatrix}^{-1} \begin{bmatrix} -1 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 1.2 & 0.4 & 1 \\ 0.4 & -0.2 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix} \tag{0.21}$$

with $x_1 = 2$, $x_2 = -1$ and $x_3 = 1$. (More details on how to calculate matrix inverses coming up in the chapter on Gaussian Elimination.)

1.1.2 A is Square and Degenerate

Not all square matrices have inverses. Consider the set of simultaneous equations:

$$\begin{aligned} -1 &= x_1 + 2x_2 \\ -2 &= 2x_1 + 4x_2 \end{aligned} \tag{0.22}$$

In matrix notation, these give:

$$\begin{bmatrix} -1 \\ -2 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{0.23}$$

Try and inverse this matrix, and you'll find that you can't[1]. What's gone wrong? We've got two equations in two unknowns, why can't we solve them? In this case, the answer is, I hope, quite easy to spot. We don't really have two equations, we've only got one. The second equation is just twice the first equation. It doesn't give us any more information.

These problem cases are not always so easy to spot. For example, consider the equations:

$$\begin{aligned} -1 &= x_1 + 2x_2 - x_3 \\ 3 &= 2x_1 - x_2 - 2x_3 \\ -4 &= -x_1 + 3x_2 + x_3 \end{aligned} \tag{0.24}$$

Again, in matrix notation, this gives:

[1] MATLAB, for example, will give a message saying "Warning: Matrix is singular to working precision", which is just another way of saying this matrix has no inverse. It's analogous to trying divide by zero. In fact, if you try and work out the inverse of this matrix, dividing by zero is exactly the problem you'll be faced with at some point in the calculations.

$$\begin{bmatrix} -1 \\ 3 \\ -4 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -1 \\ 2 & -1 & -2 \\ -1 & 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{0.25}$$

and again, try and invert this matrix, and you'll find that you can't. It's the same problem: we don't have three equations giving new information here, it's possible to derive any of these equations from the others. (For example, add the second and third equations together, and you'll get the first one.)

In general, we call such sets of equations *not linearly independent*. In other words, we can express at least one of the equations in terms of a linear combination of the other equations.

The number of linearly independent equations that the matrix expression represents is known as the *rank* of the matrix (so the rank of the matrix in equation (0.25) is two, and the rank of the matrix in equation (0.23) is one). A matrix in which the rows are not linearly independent is known as *degenerate* or *rank-deficient*, and it can't be inverted. Any square matrix in which the rows are linearly independent is known as a *full-rank* matrix, and it can be inverted.

Put this another way: a matrix of rank N contains enough information to solve a system of simultaneous equations with N unknowns, but no more.

For the system of equations in equation (0.25), with a matrix of rank two and three unknown elements in **x**, there is not enough information to find a single unique solution. There are an infinite number of solutions. (This is known as an *underdetermined system*.) We're free to choose one of the elements of **x** to be anything we like, and then solve for the other two. Without some additional knowledge about the problem and what would be a 'good' solution to find, there's not much else we can do.

This 'additional knowledge' often comes in the form of a requirement to minimise the total length of the vector **x**, and we're left with the more interesting task of finding the solution for **x** that has the minimum possible length. There's more about that problem later in this chapter, as well as in the chapter on Matrix Calculus.

1.1.2.1 Contradicting Equations

There's a slight variation on this problem of rank-deficient matrices and underdetermined systems. Suppose we had two simultaneous equations of the form:

$$\begin{aligned} -1 &= x_1 + 2x_2 \\ -3 &= 2x_1 + 4x_2 \end{aligned} \tag{0.26}$$

Now we've got a real problem: there are no solutions. The two equations contradict each other[2]. There are no possible values of x_1 and x_2 that satisfy both these equations. Now what?

If this happens in a real calculation involving physical measurements, the most likely explanation is that there has been an error in one of the measurements, perhaps due to noise, and the equations that you're really trying to solve are:

[2] I hope this is obvious: if not, try multiplying the first equation by two, and then comparing it to the second equation.

$$\begin{aligned} -1 - e_1 &= x_1 + 2x_2 \\ -3 - e_2 &= 2x_1 + 4x_2 \end{aligned} \tag{0.27}$$

where e_1 and e_2 are unknown error terms. Re-write these in the form:

$$\begin{aligned} -1 &= x_1 + 2x_2 + e_1 \\ -3 &= 2x_1 + 4x_2 + e_2 \end{aligned} \tag{0.28}$$

and you can see that what we actually have here is two linearly-independent equations in four unknowns: x_1, x_2, e_1 and e_2. That's a system with insufficient information (in other words another underdetermined system). If we wrote it in matrix format, we'd get:

$$\begin{bmatrix} -1 \\ -3 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 1 & 0 \\ 2 & 4 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ e_1 \\ e_2 \end{bmatrix} \tag{0.29}$$

and we no longer have a square system. Assuming that the error terms have a zero mean and the same standard deviation, the most likely value of the vector **x** is given by the *least-square method*, which minimises the sum of the squares of the moduli[3] of the error terms:

$$\|\mathbf{e}\|^2 = |e_1|^2 + |e_2|^2 \tag{0.30}$$

Treating the error terms as a vector **e**, so the equations are written:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \tag{0.31}$$

this approach minimises the length of this error vector, which is often the best thing to do (in the sense that it gives the answer for **x** that is most likely to be right: but remember the requirement that for this to be the best solution, all the error terms should have the same standard deviation). More details about how to do this calculation in the Geometric Interpretation section at the end of this chapter, and the chapter on Matrix Calculus.

1.1.3 A is Skinny: Redundant Information

Sometime we don't have a square system to start with. For example, consider the equations:

$$\begin{aligned} -1 &= x_1 + 2x_2 \\ 3 &= 2x_1 - x_2 \\ -4 &= -x_1 + 3x_2 \end{aligned} \tag{0.32}$$

Take any two of these equations, and you can calculate that $x_1 = 1$ and $x_2 = -1$. You can throw the other one away, you don't need it. In matrix terms, we'd write:

[3] Taking the modulus of each term just in case they are complex. In communications engineering, we have to deal with a lot of complex vectors and matrices.

$$\begin{bmatrix} -1 \\ 3 \\ -4 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 2 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{0.33}$$

and we could determine that the rank of the matrix is 2, so there is sufficient information to solve the equations for two unknowns[4]. We can just ignore one of the equations, and in this case, it doesn't matter which one. That's not always true: for example, consider the three equations:

$$\begin{aligned} -1 &= x_1 + 2x_2 \\ 3 &= 2x_1 - x_2 \\ 1 &= -x_1 - 2x_2 \end{aligned} \tag{0.34}$$

The third equation is minus one times the first equation. In this case throwing away the second equation would be a really bad idea: it would result in two identical equations (apart from the factor of minus one). You have to be a bit careful about which equations you choose to ignore.

Fortunately, there are simple techniques for determining which equations to throw away, and they don't add much to the number of calculations required to solve the system of equations. Again, more about these in the chapter on Gaussian Elimination.

1.1.4 A is Skinny: Contradicting Information

Just like the case of a rank-deficient matrix with contradicting equations, the usual case here is that there is some error in the measurements used to find the numbers, and instead of:

$$\begin{aligned} -1 &= x_1 + 2x_2 \\ 3 &= 2x_1 - x_2 \\ 0 &= -x_1 + 3x_2 \end{aligned} \tag{0.35}$$

which does not have any solutions for x_1 and x_2 since these equations contradict, the real problem is actually to solve:

$$\begin{aligned} -1 - e_1 &= x_1 + 2x_2 \\ 3 - e_2 &= 2x_1 - x_2 \\ 0 - e_3 &= -x_1 + 3x_2 \end{aligned} \tag{0.36}$$

which, in matrix form, could be written as a fat matrix with insufficient information:

[4] MATLAB users can find the rank of a matrix using the function `rank(A)`.

$$\begin{bmatrix} -1 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 1 & 0 & 0 \\ 2 & -1 & 0 & 1 & 0 \\ -1 & 3 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ e_1 \\ e_2 \\ e_2 \end{bmatrix} \tag{0.37}$$

or (more usually) as:

$$\begin{bmatrix} -1 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 2 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \tag{0.38}$$

and again, we usually want to find the solution of this set of equations that has the minimum values of the error terms. As noted before, the most popular technique is the *least-square method*, which minimises the sum of the squares of the error terms, here:

$$\|\mathbf{e}\|^2 = {e_1}^2 + {e_2}^2 + {e_3}^2 \tag{0.39}$$

1.1.5 A is Skinny: Insufficient Information

This is a real pathological case, but since it can exist, for completeness, I'll mention it. Consider the set of equations:

$$\begin{aligned} -1 &= x_1 + 2x_2 \\ -2 &= 2x_1 + 4x_2 \\ -3 &= 3x_1 + 6x_2 \end{aligned} \tag{0.40}$$

Hopefully you can see the problem: this might look like three equations, so that in matrix form this would look like:

$$\begin{bmatrix} -1 \\ -2 \\ -3 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{0.41}$$

but really there is only one piece of information here: the second equation is just twice the first equation, and the third equation three times the first equation. There is no new information about **x** in the second and third equations. The rank of this matrix is one.

That gives us a rank one matrix, with two unknowns. That's an underdetermined system: we don't have enough information to solve it, there are an infinite number of possible solutions.

The point is that it doesn't really matter what shape the matrix is, the important thing is the rank of the matrix: if that is equal to the number of unknown elements in **x**, then we've got enough information to find a unique solution. If it's less than the number of unknown elements in **x**, we've got an underdetermined system, and there will either be an infinite number of solutions, or none at all. If there are an infinite number, we'll need more information to know which solution is the best one to choose.

1.1.6 A is Fat: Insufficient Information

The more usual case of insufficient information is when the matrix **A** is fat (i.e. has more columns than rows). Consider the equations:

$$\begin{aligned} -1 &= x_1 + 2x_2 - x_3 \\ 3 &= 2x_1 - x_2 - 2x_3 \end{aligned} \tag{0.42}$$

which in matrix notation, gives:

$$\begin{bmatrix} -1 \\ 3 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -1 \\ 2 & -1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{0.43}$$

This matrix has a rank of two (the rows are not multiples of each other, so these really are two independent equations), but there are three elements in **x**. This is a very similar case to that of the rank-deficient square system, however in this case we don't have the possibility of contradicting equations and no solutions. (You could think of this set of equations as a square set where one of the redundant equations has already been deleted.)

However, it's still an underdetermined system. We don't have enough information to find an exact solution, so we have to choose from the infinite number of solutions. One possible choice is the solution that minimises the length of the vector **x**, and this is another variation on the *least-square method*, which in this case minimises the square of the length of **x**:

$$\|\mathbf{x}\|^2 = |x_1|^2 + |x_2|^2 + |x_3|^2 \tag{0.44}$$

rather than the length of an error vector **e**. Again, more on how to solve these coming up.

1.2 A Geometric Interpretation of Matrices

If you're anything like me, and prefer to think in pictures, all these dry equations don't really help get a good understanding of what's going on. It's better to use pictures, and this turns out to be a very useful way to think about these problems. Vectors can represent a point in space, so a vector **x** with *n* elements represents a unique point in *n*-dimensional space (sometimes written $\mathbb{R}^N$). Multiply this vector by a matrix **A**, and the result will be another vector **y**, representing another point in space.

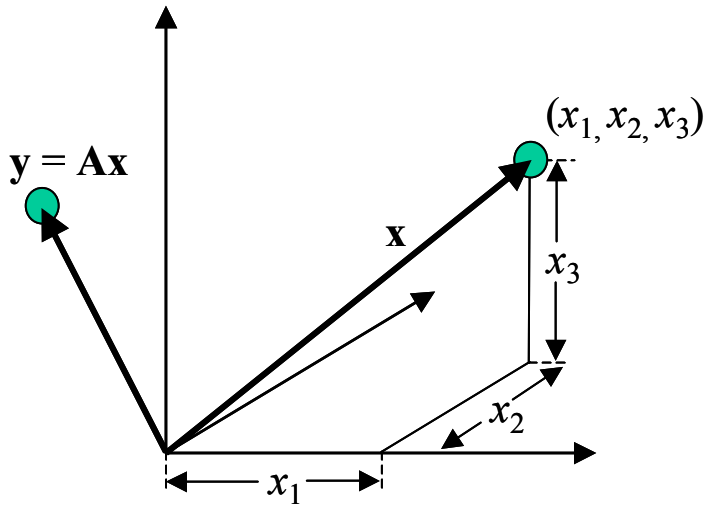

Figure 1-1 The Vectors x and y as Points in Three-Dimensional Space

The numbers of element in **x** and **y** determines the *dimensionality* of the spaces of **x** and **y**. For example, a vector with two elements represents a point on a plane, a two-dimensional surface; a vector with three elements represents a point in three-dimensional space. A vector with ten elements represents a point in ten-dimensional space, which is a lot harder to think about, and even harder to draw, but the general idea of a point in space is still useful.

1.2.1 Geometry and Full-Rank Square Matrices

If the matrix **A** is square, then the vectors **x** and **y** will have the same number of elements, and hence represent points in the same dimensionality (e.g. they both define 2-dimensional planes, or three-dimensional spaces, etc). If the matrix is invertible, and **y** = **Ax**, then for any vector **x**, we can find a unique vector **y**, and for any vector **y** we can find a unique vector **x**. All we need to do is find the inverse mapping from a point **y** to a point **x**, and that's provided by the inverse matrix. We just have to solve:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{y} \tag{0.45}$$

1.2.2 Geometry and Skinny Matrices

If the matrix **A** is skinny (i.e. has more rows than columns, and therefore the vector **x** has less elements than the vector **y**), then there will be some vectors **y** which do not correspond to any vector **x**. In other words, only a subset of the possible values of **y** can be produced from **Ax**. Being linear equations, these subsets are linear spaces: lines, planes, 3-dimensional spaces, 4-dimensional spaces, etc. If the number of elements in the vector **y** is bigger than the number of elements in the vector **x**, then there's a problem: there may not be a value of **x** for which **Ax** = **y**.

For example, consider the equations **y** = **Ax**, where **y** has three elements, and **x** has two:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 2 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{0.46}$$

The vector **x** has two-elements, and can therefore be thought of as a point on a plane. The vector **y** has three elements, and can be thought of as a point in three-dimensional space. The set of possible values of **Ax** then corresponds to the mapping of a 2-dimensional plane into the 3-dimensional space of the possible values of **y**[5]. All these points **Ax** will lie on a plane in 3-dimensional space.

If the value of **y** happens to lie in this plane, then we have linearly-dependent equations, and we can find a unique solution for **x**, just by ignoring one of the equations. However, if the value of **y** doesn't lie in this plane, then we have a case of contradicting equations, and we can't find a value for **x** that satisfies **y** = **Ax**. The best we can do is find the value of **x** for which **Ax** is as close as possible to the given value of **y**. This is exactly the same thing as minimising the length of the error vector **e** in the expression **y** – **e** = **Ax**, since **e** is a vector representing the distance between **y** and **Ax**.

It's a bit hard to draw, but I've tried in the figure below:

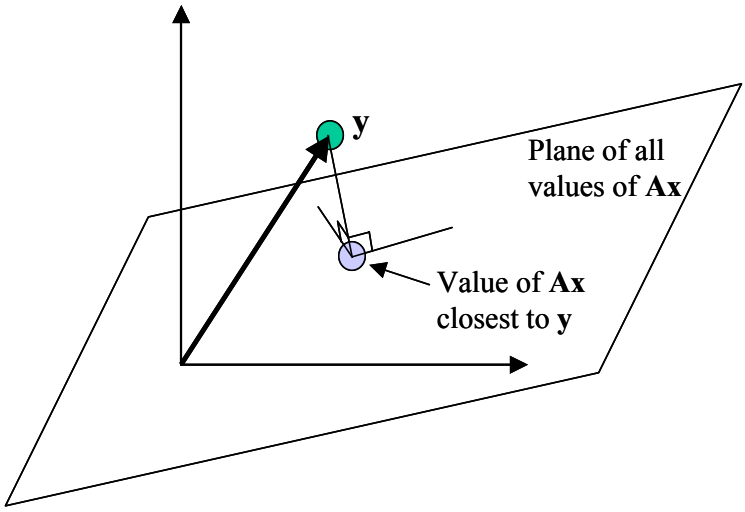

Figure 1-2 Least-Square Solution to y = Ax in Three Dimensions

It's probably easier to visualise in the case where **x** has just one element, and **y** has two: in this case the set of possible solutions to **y** = **Ax** are a line in the 2-dimensional space of **y**, which is just a plane. In that case, the least-square solution could be shown as in this figure:

[5] It's always a plane, and not a spherical surface, or a paraboloid, or any other curved surface. The equations would have to be non-linear to transform a plane into a curved surface, and this is linear algebra.

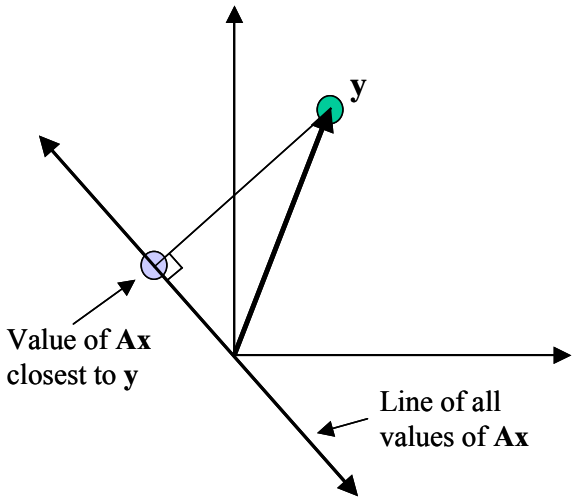

Figure 1-3 Least Square Solution to y = Ax in Two Dimensions

This geometric interpretation of matrices and vectors allows us to find this *least-squares* solution in a simple way. Remember, for a least-squares solution, we're looking for the value of **x** that allows us to solve the equation:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} = \begin{bmatrix} 1 & 2 \\ 2 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \tag{0.47}$$

for the minimum possible length of vector of **e**. A quick reminder: since all possible values of **x** map to a plane in the three-dimensional space of vector **y**, if we have a value of **y** that does not lie on this plane, we can't find a value of **x** for which the error is zero; the best we can do to minimise the error vector **e** is find the value of **x** for which **Ax** is as close as possible to the given value of **y**.

Now the clever bit: the smallest possible error vector **e** is perpendicular to all the vectors that lie in the region **Ax**. Again, this is easiest to draw in two-dimensions, where **y** has two elements, and **x** just one, so that all vectors **Ax** are points on a line:

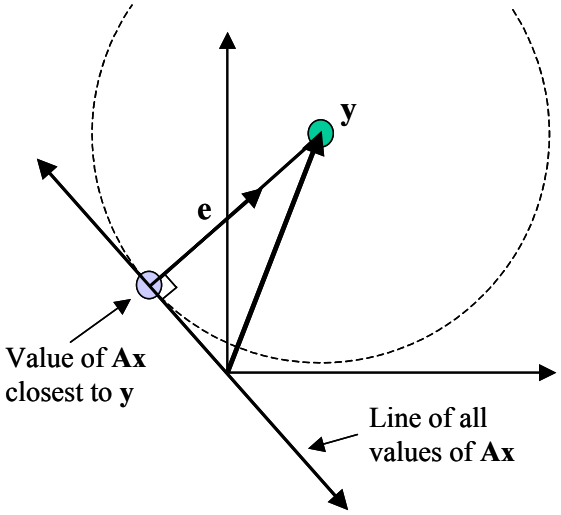

Figure 1-4 The Least Squares Solution as a Perpendicular

Consider a circle drawn around the point **y** (in three dimensions it would be a sphere, and so on). The smallest circle (sphere, etc) that reaches the line given by all the possible points **Ax** just touches the line, so the line must be a tangent to the circle.

The line from **y** to the nearest point on **Ax** is the error vector **e**, since **y** = **Ax** + **e**, and this must be a radius of the circle. The radius to a point on a circle is perpendicular to the tangent that passes through the same point. (In three-dimensions, the radius to a point on a sphere is perpendicular to the tangent plane at that point: a tangent plane being a plane that just touches, but does not cross the surface. And so on, in higher numbers of dimensions.)

Then, to calculate the least-square value of **x**, let $\mathbf{x}_1$ and $\mathbf{x}_2$ be any two possible values of **x**. Then $\mathbf{Ax}_1$ and $\mathbf{Ax}_2$ will be any two points on the plane containing all the points **Ax** in 3-dimensional space (or any two points on the line of points **Ax** in two-dimensional space.) The line joining them together, the vector $\mathbf{Ax}_1 - \mathbf{Ax}_2$ must be perpendicular to the error vector **e**, so[6]:

$$\left(\mathbf{Ax}_1 - \mathbf{Ax}_2\right)^H .\mathbf{e} = 0 \tag{0.48}$$

Therefore:

$$\begin{aligned} \mathbf{y} &= \mathbf{Ax} - \mathbf{e} \\ \left(\mathbf{Ax}_1 - \mathbf{Ax}_2\right)^H \mathbf{y} &= \left(\mathbf{Ax}_1 - \mathbf{Ax}_2\right)^H \mathbf{Ax} - \left(\mathbf{Ax}_1 - \mathbf{Ax}_2\right)^H \mathbf{e} \\ \left(\mathbf{Ax}_1 - \mathbf{Ax}_2\right)^H \mathbf{y} &= \left(\mathbf{Ax}_1 - \mathbf{Ax}_2\right)^H \mathbf{Ax} \end{aligned} \tag{0.49}$$

Now, using the matrix identity $(\mathbf{AB})^H = \mathbf{B}^H\mathbf{A}^H$ (see the chapter on matrices), we have:

$$\begin{aligned} \left(\mathbf{x}_1 - \mathbf{x}_2\right)^H \mathbf{A}^H\mathbf{y} &= \left(\mathbf{x}_1 - \mathbf{x}_2\right)^H \mathbf{A}^H\mathbf{Ax} \\ \left(\mathbf{x}_1 - \mathbf{x}_2\right)^H \left(\mathbf{A}^H\mathbf{y} - \mathbf{A}^H\mathbf{Ax}\right) &= 0 \end{aligned} \tag{0.50}$$

and since $\mathbf{x}_1$ and $\mathbf{x}_2$ can be any two values of **x**, the only way to ensure that this expression is always true is to take:

$$\begin{aligned} \mathbf{A}^H\mathbf{y} &= \mathbf{A}^H\mathbf{Ax} \\ \mathbf{x} &= \left(\mathbf{A}^H\mathbf{A}\right)^{-1}\mathbf{A}^H\mathbf{y} \end{aligned} \tag{0.51}$$

This is known as the *normal equation*[7]. It gives the best possible value of **x** to use, in the sense that it minimises the length of the error term[8]. It's a general result[9], not restricted to the case

---

[6] Using the definition of perpendicular that two vector are perpendicular if $\mathbf{x}^H\mathbf{y} = 0$. This is an extension to the more familiar case of requiring the dot product to be zero to cover the case of vectors with complex elements. For more on why this is a good way to define "perpendicular" in these cases, see the chapter on Vectors.

[7] That's 'normal' in the sense of 'perpendicular'. It's the solution for which the error vector **e** is perpendicular to the vector space **Ax**.

[8] The matrix $\mathbf{B} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H$ has some similarities to an inverse of **A**: for example, **ABA** = **A**, and **BAB** = **B**. For this reason, **B** is sometimes called a *pseudoinverse matrix* of **A**.

[9] One word of warning: the normal equation doesn't always work in the case of rank-deficient matrices. Consider:

[continued on next page…]

where **y** has three-dimensions and **x** has two, although it's rather harder to picture what's happening with any more dimensions.

(Note that $\mathbf{A}^H\mathbf{A}$ is a positive-definite Hermitian matrix. This is useful, since it means that it only has real positive eigenvalues, and therefore that iterative techniques such as the method of steepest descents will converge. If you've no idea what that means, don't worry, there's more details coming up in later chapters.)

1.2.3 Geometry and Fat Matrices

The case of a fat matrix is common as well. Suppose the matrix **A** is fat (has more columns than rows), so that the vector **x** has less elements than the vector **y**. For example:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -1 \\ 2 & -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{0.52}$$

Now the vector **x** has three-elements, and can be thought of as a point in a 3-dimensional space. The vector **y** has two elements, and can be thought of as a point on a 2-dimensional plane. In this case, there will be multiple values of **x** that map to the same point **y**. Being linear equations, all the possible solutions **x** of the equation $\mathbf{y} = \mathbf{A}\mathbf{x}$ will lie on a straight line[10].

This is the case of an infinite number of solutions, and in many cases the solution we want to find is the one that minimises the length of the vector **x**. We can use a similar geometric insight here. Thinking of a vector as a line from the origin to a point in space, the solution for **x** we want is the one closest to the origin; and the shortest line from the origin to a straight line is perpendicular to the straight line.

So, let **w** be any solution to the equation $\mathbf{y} = \mathbf{A}\mathbf{w}$ (in other words, a vector representing any point on the line of solutions), and let **x** be the solution we want (the point on the line of solutions closest to the origin). Then, since these values are both solutions to the original equation, we can write:

$$\mathbf{y} = \mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{w} \tag{0.53}$$

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}$$

the matrix **A** has a rank of one, which means that all terms of the form **Ax** lie on a single line in the three-dimensional vector space of **y** (in this case, the line $y_1 = y_2/2 = y_3/3$). The problem here is that the vector **x** has two elements, and therefore represents a point in two-dimensional space. This matrix **A** is mapping a set of points in two-dimensional space onto a line. This reduction in the number of dimensions implies that there are an infinite number of values of **x** that map to the closest value of **Ax** to **y**, and therefore there is no unique solution for **x**.

For the normal equation to give a solution, the rank of the matrix **A** must be equal to the number of elements in **x**.

[10] If the vector **x** had two more elements than the vector **y**, all the solutions would lie on a plane in the vector space of **x**. If there were three more elements in **x** than **y**, all the solutions would lie somewhere in a three-dimensional space. And so on.

and we also know that the line from **w** to **x** is perpendicular to the line from the origin to **x**, so:

$$\left(\mathbf{w}-\mathbf{x}\right)^{H}.\mathbf{x}=0 \tag{0.54}$$

and from these equations, we can derive that:

$$\mathbf{x}=\mathbf{A}^{H}\left(\mathbf{A}\mathbf{A}^{H}\right)^{-1}\mathbf{y} \tag{0.55}$$

(See the problems and solutions for how this is done.) This is a general result as well, although again it doesn't work in cases where the rank of the matrix **A** is less than the number of elements in **y**. (In these cases there are usually no solutions for **x** that satisfy **y** = **Ax**, and the derivation assumed that there were an infinite number of solutions.)

1.3 Questions

1) In the taxonomy in this chapter, I didn't consider the case of fat matrices with redundant or contradicting information. Is this case possible? How would you approach a system of simultaneous linear equations such as these:

$$\begin{aligned}1&=x_1+2x_2-x_3\\1&=-x_1-2x_2+x_3\end{aligned} \tag{0.56}$$

2) What can you say about the set of simultaneous equations:

$$\begin{aligned}-1&=x_1+2x_2-x_3\\-1&=-x_1+2x_2+x_3\\3&=-2x_1-4x_2+2x_3\\2&=x_1-2x_2-x_3\end{aligned} \tag{0.57}$$

Do you have not enough information to solve for **x**, or too much? How could you go about finding a solution for **x** in this case?

3) In the case of a square, invertible matrix **A**, simplify the normal equation $\mathbf{A}^{H}\mathbf{y}=\mathbf{A}^{H}\mathbf{A}\mathbf{x}$.

4) Suppose you're given the equations to solve:

$$\begin{aligned}-1+e_1&=x_1+2x_2\\-3+e_2&=2x_1+4x_2\end{aligned} \tag{0.58}$$

and you're told that the error in the value –1 (e_1) has a zero mean and a standard deviation of 0.1, and the error in the value of –3 (e_2) has a zero mean and a standard deviation of 0.5. What's the best guess you can make of the value of **x**?

5) Suppose you have a matrix **A** for which neither $\mathbf{A}^{H}\mathbf{A}$ nor $\mathbf{A}\mathbf{A}^{H}$ are invertible. In terms of the geometric interpretation, what is happening here? How might a least-squares solution be found?

6) Prove that when there are an infinite number of solutions to the equations **y** = **Ax**, the solution that minimises the length of the vector **x** is given by:

$$\mathbf{x} = \mathbf{A}^H \left(\mathbf{A}\mathbf{A}^H \right)^{-1} \mathbf{y} \tag{0.59}$$

(Hint: first prove that this is a solution to $\mathbf{y} = \mathbf{A}\mathbf{x}$, then show that $(\mathbf{w} - \mathbf{x})^H.\mathbf{x} = 0$ for all values of $\mathbf{w}$ that also satisfy $\mathbf{y} = \mathbf{A}\mathbf{w}$.)

7) Show that the pseudoinverse matrix $\mathbf{B} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H$ satisfies $\mathbf{ABA} = \mathbf{A}$, and also that the matrix $\mathbf{AB}$ is Hermitian (i.e. that $\mathbf{AB} = (\mathbf{AB})^H$).

8) Can $\mathbf{B} = \mathbf{A}^H(\mathbf{A}\mathbf{A}^H)^{-1}$ also be described as a pseudoinverse? Does it share the same properties as the pseudoinverse $(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H$? In what circumstances would each be an appropriate choice?