# Chapter 1

# Basic Linear Algebra

## 1.1 Introduction

In general, an $m \times n$ matrix can be written as an uppercase letter $A$. Its elements are written as lowercase $a_{ij}$ in the matrix form

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & & \cdots & a_{2n} \\ a_{31} & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix} \tag{1.1}$$

In particular, note that $a_{ij}$ is the entry in the $i^{th}$ row and $j^{th}$ column. This is also frequently called the $ij$-entry of the matrix, for economy. Also note that the matrix has $m$ rows and $n$ columns. Matrices are always described as '(number of rows)×(number of columns) matrices'.

We call the elements $a_{11}$, $a_{22}$ and so on the diagonal elements since they go down the diagonal of the matrix. Frequently, we also use the term superdiagonal for the diagonal immediately above the main diagonal, and subdiagonal for those immediately below.

One particularly important matrix is the identity matrix, which we call $I$. It is a matrix with ones on the main diagonal, and zeros elsewhere. In more mathematical notation,

$$I = \begin{bmatrix} 1 & 0 & \cdots & & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0 & \cdots & & 0 & 1 \end{bmatrix} \tag{1.2}$$

We should consider fundamental arithmetic operations we've used before on real numbers on these matrices.

## 1.2 Matrix Addition and Subtraction

If we wish to calculate the matrix sum $A + B$, the mathematical definition is

$$[A + B]_{ij} = a_{ij} + b_{ij} \tag{1.3}$$

where $[A + B]_{ij}$ denotes the $ij$-entry of the sum of these matrices. For the matrix sum to be well-defined, $A$ and $B$ must match in dimension(both should be $m \times n$ matrices). Further, subtraction may be defined in a similar way, with similar restrictions on dimensions of matrix sizes.

## 1.3 Matrix Multiplication

Multiplication is not quite so straightforward, but is still defined in terms of operations on the elements of the matrix. In particular,

$$[A * B]_{ij} = \sum_{k=0} a_{ik} b_{kj} \tag{1.4}$$

where $A$ is an $m \times n$ matrix, $B$ is a $p \times q$ matrix. Please note the three crucial observations:

1. $AB$ is undefined if $n \neq p$ (the inner dimensions of the matrices being multiplied must match).
2. If $AB$ is defined, then it is a $m \times q$ matrix.
3. The matrices $AB$ and $BA$ are, in general, not equal.

One way to see part 2 from above is to note that we are not requiring $m = q$ or $m = n$, so if $A$ is a 2×3 matrix and $B$ is a $3 \times 2$ matrix, then $AB$ is a 2×2 matrix and $BA$ is a 3×3 matrix, and therefore they have no hope of being equal.

Another deeper and more interesting case is to note that even in the nontrivial case of two square matrices ($A$ and $B$ are $n \times n$), the product

$$[AB]_{ij} = a_{11}b_{11} + a_{12}b_{21} \tag{1.5}$$

and the product

$$[BA]_{ij} = b_{11}a_{11} + b_{12}a_{21} \tag{1.6}$$

are not necessarily equal. In particular, they are not equal unless $a_{12}b_{12} = a_{21}b_{12}$, which does not hold for infinitely many matrices.

The most major application of matrix multiplication(and indeed, perhaps even matrices themselves), arise from equations written in the form

$$\begin{aligned} a_{11}b_{11} + a_{12}b_{21} + &\quad \cdots + a_{1n}b_{n1} && (1.7)\\ a_{21}b_{11} + a_{22}b_{21} + &\quad \cdots + a_{2n}b_{n1} && (1.8)\\ &\vdots && (1.9)\\ a_{m1}b_{11} + a_{m2}b_{21} + &\quad \cdots + a_{mn}b_{n1} && (1.10)\\ & && (1.11) \end{aligned}$$

Note from our definitions above that this implies that we can write two matrices as a matrix multiplication(by going backwards from definitions above). These general matrices look like

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{11} & a_{12} & \cdots & a_{1n} \end{bmatrix} * \begin{bmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{n1} \end{bmatrix} = AB = A\vec{b} \tag{1.12}$$

As is typical in mathematical literature, $n \times 1$ matrices are typically denoted by the lowercase letter with an arrow above. An astute reader should verify this equivalence since it arises over and over in applied mathematics. Its importance can not be overstated.

Finally, it's important to consider what happens when we multiply a matrix by the identity. Intuitively, we know that if we multiply any number by 1, we should get the same number back. So we might suspect that if we multiplied any matrix by $I$, we'd get the same matrix back. And this turns out to be true. It is quite possible to work this out from the definition of matrix multiplication and the definition of the identity. Since the $ij$-entry of the identity is only 1 if $i = j$(and otherwise zero), the matrix product

$$[A * I]_{mn} = \sum_{k=0} a_{mk} i_{kn} = a_{mn} \tag{1.13}$$

Happily, this is what we desired.

## 1.4 Elementwise Matrix Multiplication

Sometimes, we would like to multiply each element of a matrix only to the corresponding element of the other matrix. This is know as elementwise matrix multiplication, and is defined as

$$[A. * B]_{ij} = a_{ij}b_{ij} \tag{1.14}$$

You should note that for this definition to make sense, $A$ and $B$ should have matching dimensions. You must also carefully consider which definition makes sense in each MATLAB multiplication you consider. Typically this method is useful for filtering, as well as some methods we'll consider later for solving many differential equations.

## 1.5 Matrix Division

Matrix division is not defined quite the same as the rest, but we'll cover it here anyways. In basic math, dividing by a number is the same as multiplying by it's inverse. For example, it doesn't matter whether we divide by 2 or multiply by $\frac{1}{2}$. We can use a similar definition to define matrix division, if we have a way to find the inverse of a matrix. It turns out that in general this might not be possible for all matrices, but for a lot of the important ones we'd like to write down, it ends up working out. In particular, when we wrote down the above equation $A\vec{b}$ above, it is possible to solve for $\vec{b}$ by this method.

To see this, suppose that $A\vec{b} = \vec{x}$. Then by multiplying the equation by $A^{-1}$ on both sides, we get $A^{-1}A\vec{b} = A^{-1}\vec{x}$. Recall that in regular mathematics, if we multiply a number by it's inverse, we get 1(identity). In matrix math, this ends up being exactly the identity, which we saw in the multiplication section just evaluates to the same vector as it's multiplied by. In other words, $\vec{b} = A^{-1}\vec{x}$.

It's worth noting that calculating the inverse is usually a pretty bad idea except for very small, particularly nice matrices. Instead, most people use the MATLAB backslash command, which you'll cover in class in some depth.

## 1.6 Elementwise Matrix Division

Elementwise division is very similar to elementwise multiplication, except dividing instead of multipling. It is probably even more often used, in fact. We define elementwise matrix division as

$$[A./B]_{ij} = a_{ij}/b_{ij} \tag{1.15}$$

Note that in elementwise multiplication and division, as in addition and subtraction, dimensions must match for the matrices.

## 1.7 Transpose

Sometimes we have a column vector and desire a row vector, or vice-versa. This operations is known as the transpose, and defined as

$$[A^T]_{ij} = A_{ji} \tag{1.16}$$

As you might notice, this definition is general enough to work for matrices too. We should just swap elements across the diagonal.

## 1.8 MATLAB Notes

### 1.8.1 Matrix Accesses

MATLAB has a very flexible method of accessing elements in a matrix. In the most general form, an access from A is of the form

```
A(n,m)
```

where $n$ and $m$ are (respectively) row and column selectors and can be scalars($j$ or 1) or vectors($3:2:42$, $[3,2,1]$). Also allowed is a single colon, which should be read as 'all of' either the rows or columns, depending on which position it is in. Finally, the keyword 'end' is allowed to set or retreive the last entry. Putting all of this together, $A(:,[1,end])$ should be read as 'take all rows of the first and last columns of the matrix A'. It is greatly beneficial to practice this until it becomes second nature.

### 1.8.2 Transpose

In MATLAB, the transpose operator works exactly as expected for real matrices, but has a quirk for matrices with complex elements. To explain, if $a = x + \sqrt{-1}y$ is a complex number, then $\bar{a} = x - \sqrt{-1}y$ is it's complex conjugate. Note that this implies for real numbers, $a = \bar{a}$, since $y = 0$. This also applies for matrices. If $A$ is a matrix with complex entries, then $\bar{A}$ is the complex-conjugate matrix and each entry has been replaced with its complex conjugate.

Now, the MATLAB operation

```
A'
```

calculates the transpose of the complex conjugate matrix, while

```
A.'
```

calculates the transpose without conjugating entries of the matrix. This shouldn't concern you much in this class, but explains the difference between the two.