A DIGRESSION ON ABSTRACT LINEAR ALGEBRA

YOUWEN WU

1. Introduction

Many introductory linear algebra classes focus on *application*. In general, this is a red herring and is engineer-speak for "we will teach you how to crunch numbers with no regard for conceptual understanding."

If you are a math major (or math-adjacent, such as Computer Science), this class is essentially useless for you. You will learn how to perform trivial numerical operations such as the *matrix multiplication*, *matrix-vector multiplication*, *row reduction*, and other trite tasks better suited for computers.

If you are taking this course, you might as well learn linear algebra properly. Otherwise, you will have to re-learn it later on, anyways. Completing a math course without gaining a theoretical appreciation for the topics at hand is an unequivocal waste of time. I have prepared this brief crash course designed to fill in the theoretical gaps left by this class.

2. Basic Notions

2.1. Vector spaces.

Before we can understand vectors, we need to first discuss *vector spaces*. Thus far, you have likely encountered vectors primarily in physics classes, generally in the two-dimensional plane. You may conceptualize them as arrows in space. For vectors of size > 3, a hand waving argument is made that they are essentially just arrows in higher dimensional spaces.

It is helpful to take a step back from this primitive geometric understanding of the vector. Let us build up a rigorous idea of vectors from first principles.

2.1.1. *Vector axioms.*

The so-called *axioms* of a *vector space* (which we'll call the vector space V) are as follows:

1. Commutativity: $u + v = v + u,\ \forall u, v \in V$
2. Associativity: $(u + v) + w = u + (v + w),\ \forall u, v, w \in V$
3. Zero vector: $\exists$ a special vector, denoted 0, such that $v + 0 = v,\ \forall v \in V$
4. Additive inverse: $\forall v \in V,\ \exists w \in V$ such that $v + w = 0$. Such an additive inverse is generally denoted $-v$
5. Multiplicative identity: $1v = v,\ \forall v \in V$
6. Multiplicative associativity: $(\alpha\beta)v = \alpha(\beta v)\ \forall v \in V, \text{scalars } \alpha, \beta$
7. Distributive property for vectors: $\alpha(u + v) = \alpha u + \alpha v\ \forall u, v \in V, \text{scalars } \alpha$
8. Distributive property for scalars: $(\alpha + \beta)v = \alpha v + \beta v\ \forall v \in V,\ \text{scalars } \alpha, \beta$

It is easy to show that the zero vector 0 and the additive inverse $-v$ are *unique*. We leave the proof of this fact as an exercise.

These may seem difficult to memorize, but they are essentially the same familiar algebraic properties of numbers you know from high school. The important thing to remember is which operations are valid for what objects. For example, you cannot add a vector and scalar, as it does not make sense.

Remark. For those of you versed in computer science, you may recognize this as essentially saying that you must ensure your operations are *type-safe*. Adding a vector and scalar is not just false, it is an *invalid question* entirely because vectors and scalars and different types of mathematical objects. See Chen, E. [1] for more.

2.1.2. *Vectors big and small.*

In order to begin your descent into what mathematicians colloquially recognize as *abstract vapid nonsense*, let's discuss which fields constitute a vector space. We have the familiar space where all scalars are real numbers, or $\mathbb{R}$. We generally discuss 2-D or 3-D vectors, corresponding to vectors of length 2 or 3; in our case, $\mathbb{R}^2$ and $\mathbb{R}^3$.

However, vectors in $\mathbb{R}$ can really be of any length. Discard your primitive conception of vectors as arrows in space. Vectors are simply arbitrary length lists of numbers (for the computer science folk: think C++ `std::vector`).

Example.

$$\begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{pmatrix}$$

Moreover, vectors need not be in $\mathbb{R}$ at all. Recall that a vector space need only satisfy the aforementioned *axioms of a vector space.*

Example. The vector space $\mathbb{C}$ is similar to $\mathbb{R}$, except it includes complex numbers. All complex vector spaces are real vector spaces (as you can simply restrict them to only use the real numbers), but not the other way around.

In general, we can have a vector space where the scalars are in an arbitrary field $\mathbb{F}$, as long as the axioms are satisfied.

Example. The vector space of all polynomials of degree 3, or $\mathbb{P}^3$. It is not yet clear what this vector may look like. We shall return to this example once we discuss *basis.*

2.2. **Vector addition. Multiplication.**

Vector addition, represented by $+$, and multiplication, represented by the $\cdot$ (dot) operator, can be done entrywise.

Example.

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} + \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix} = \begin{pmatrix} 1+4 \\ 2+5 \\ 3+6 \end{pmatrix} = \begin{pmatrix} 5 \\ 7 \\ 9 \end{pmatrix}$$

$$\begin{pmatrix}1\\2\\3\end{pmatrix} \cdot \begin{pmatrix}4\\5\\6\end{pmatrix} = \begin{pmatrix}1\cdot 4\\2\cdot 5\\3\cdot 6\end{pmatrix} = \begin{pmatrix}4\\10\\18\end{pmatrix}$$

This is simple enough to understand. Again, the difficulty is simply ensuring that you always perform operations with the correct *types*. For example, once we introduce matrices, it doesn't make sense to multiply or add vectors and matrices in this fashion.

2.3. **Vector-scalar multiplication.**

Multiplying a vector by a scalar simply results in each entry of the vector being multiplied by the scalar.

Example.

$$\beta \begin{pmatrix}a\\b\\c\end{pmatrix} = \begin{pmatrix}\beta\cdot a\\\beta\cdot b\\\beta\cdot c\end{pmatrix}$$

2.4. **Matrices.**

Before discussing any properties of matrices, let's simply reiterate what we learned in class about their notation. We say a matrix with rows of length m, and columns of size n (in less precise terms, a matrix with length m and height n) is a $m \times n$ matrix.

Given a matrix

$$A = \begin{pmatrix}1&2&3\\4&5&6\\7&8&9\end{pmatrix}$$

we refer to the entry in row j and column k as $A_{j,k}$.

2.4.1. *Matrix transpose.*

A formalism that is useful later on is called the *transpose*, and we obtain it from a matrix A by switching all the rows and columns. More precisely, each row becomes a column instead. We use the notation A^T to represent the transpose of A.

$$\begin{pmatrix}1&2&3\\4&5&6\end{pmatrix}^T = \begin{pmatrix}1&4\\2&5\\3&6\end{pmatrix}$$

Formally, we can say $\left(A_{j,k}\right)^T = A_{k,j}$.

References

1. Chen, E.: Digression on Type Safety, (2024)

University of California, Santa Barbara
Email address: youwen@ucsb.edu
URL: https://youwen.dev