Mathematical Techniques for Computer Science Revision Notes

James Brown

May 8, 2017

Contents

1 Introduction **1**

2 Systems of Linear Equations **1**
2.1 Gaussian Elimination . . . 1
2.2 Special Cases for Gaussian Elimination . . . 1
2.2.1 Contradictory Equations . . . 1
2.2.2 Irrelevant Equations . . . 1
2.2.3 First Equation Unusable for Eliminating the First Unknown . . . 2
2.2.4 Can't Use Any of the Given Equations to Eliminate the First Unknown . . . 2
2.3 General Gaussian Elimination . . . 2
2.3.1 Extended Gaussian Elimination . . . 2

3 Fields **2**
3.1 Galois Field . . . 3

4 Analytic Geometry **3**
4.1 In the Plane . . . 3
4.1.1 Vectors . . . 3
4.1.2 Representing Straight Lines . . . 4
4.1.3 Intersecting Two Lines . . . 4
4.2 In Three Dimensions . . . 4
4.2.1 Planes . . . 4
4.2.2 Two-Point Description of a Line . . . 5
4.2.3 Three-Point Description of a Plane . . . 5
4.3 Vector Spaces . . . 5
4.3.1 Bases . . . 5
4.3.2 Codes . . . 5

5 A Different Way of Representing Lines and Planes **6**
5.1 Translating Between the Parametric and Equational Representations . . . 6
5.1.1 Lines . . . 6
5.1.2 Planes . . . 6

6 Matrices **7**

7 Sets **7**
7.1 Cardinality . . . 7

8 Functions and Relations **7**
8.1 Relations . . . 7
8.2 Functions . . . 7

9 Inductive Definitions **7**

10 Probability **7**
10.1 Discrete Random Variables . . . 7
10.2 Continuous Random Variables . . . 7

1 Introduction

These are notes I have written in preparation of the 2017 Mathematical Techniques for Computer Science exam. This year the module was run by Achim Jung (A.Jung@cs.bham.ac.uk).

2 Systems of Linear Equations

A linear equation which contains more than one variable will usually have many solutions. In order to pin this down to exactly one solution, so that all equations are satisfied simultaneously, we need as many equations are there are unknowns. Two unknowns, two equations required and so on.

In these systems of equations, we don't really care about the variable names. We introduce a new notation which is much more consise by removing variable names:

$$\left(\begin{array}{ccc|c} 1 & 5 & -2 & -11 \\ 3 & -2 & 7 & 5 \\ -2 & -1 & -1 & 0 \end{array}\right)$$

Figure 1: Consise notation for systems of linear equations

2.1 Gaussian Elimination

In Computer Science terms, this is best described as a recursive algorithm with a **base case** and a **general case**.

- **Base case:** There is only one equation and one unknown in the form $ax = b$. We can directly solve this: $x = b/a$.
- **General case:** There are n equations and each equation contains n unknowns. We use the first equation to *eliminate* the first unknown in the remaining $n - 1$ equations. We then recursively apply Gaussian elimination to the remaining $n - 1$ equations which each have $n - 1$ unknowns.

This method builds a staircase of zeroes beneath our variables, which is often called **echelon form**.

$$\left(\begin{array}{ccc|c} x_1 & * & * & * \\ 0 & x_2 & * & * \\ 0 & 0 & x_3 & * \end{array}\right)$$

2.2 Special Cases for Gaussian Elimination

2.2.1 Contradictory Equations

In the base case $ax = b$, it can happen that $a = 0$, so the equation is really $0 = b$. Furthermore, if it is the case that $b \neq 0$ then the equation is contradictory and cannot be solved. In this case, the algorithm should exit and indicate that a solution can not be found. This may be by raising an exception for example. An example of equations that may lead to this is as follows:

$$\begin{aligned} x_1 - 2x_2 &= 3 \\ 2x_1 - 4x_2 &= 7 \end{aligned}$$

2.2.2 Irrelevant Equations

This is like the previous special case, but both sides of the equation are zero (in $ax = b$, both a and b are zero). This means the equation does not constrain the value of x in any way. Due to

this, the value of the unknown can be chosen freely. Take the following equations:

$$x_1 - 2x_2 = 3$$
$$2x_1 - 4x_2 = 6$$

Here the second equation is redundant so we are only left with one equation for two unknowns. This gives the following assignments:

$$x_1 = 3 + x_2$$
$$x_2 : \text{ chosen freely}$$

2.2.3 First Equation Unusable for Eliminating the First Unknown

If the coefficient of the first unknown in the first equation is zero, not matter how much we add it to another equation the first variable of the other will not be affected. In order to get around this, we simply exchange the first equation with another one for which the first equation is not zero and run the algorithm on this rearranged setup.

2.2.4 Can't Use Any of the Given Equations to Eliminate the First Unknown

This is similar to the case where we have an irrelevant equation. It means that the first variable is not constrained at all by the given system. The algorithm to solve this should report back that the first variable can be chosen freely.

2.3 General Gaussian Elimination

We may encounter a situation where we have two equations for one unknown, which forces us to consider more general systems for linear equations where the number of unknowns does not neccesarily match the number of equations. Luckily, the algorithm does not need much adjustment, all we need to do is readjust the base case.

- **Base Case 1:** Only one unknown is left over but there may be more than one equation. We solve each equation independently and compare the answers. If they agree, that is what we return. If they disagree, then the system has no solution.
- **Base Case 2:** Only one equation is left over but more than one unknown appears in it (let's say m many). In this case, $m-1$ unknowns can be chosen freely and the other is computed from the equation. It does not matter which of the $m-1$ unknowns is chosen and which is computed. For example, if we are left with $x - 2y + z = 3$, we set $x = 3 + 2y - z$ and say y and z can be freely chosen.

2.3.1 Extended Gaussian Elimination

Once we have reached **echelon form** and checked that the system is not contradictory, we can elminate entries above the pivots (the staircase values). This helps us to write the general solution, as we can easily read of the values.

3 Fields

There are many **number systems** which are of interest to a computer scientist. The mathematical term for a number system is a **field**. The requirement is that the elements of a field can be added and multiplied. There also must be a 'zero' and a 'one' which must satisfy

$$x + 0 = x$$
$$x * 1 = x$$

All the usual rules of arithmetic are also valid:

$$x + y = y + x$$
$$x + (y + z) = (x + y) + z$$
$$x * y = y * x$$
$$x * (y * z) = (x * y) * z$$
$$x * (y + z) = x * y + x * z$$

We also require that the following equations can always be solved:

$$a + x = 0$$
$$a * x = 1 \text{ assuming } a \neq 0$$

From these, we can derive many other rules of arithmetic. For example, we can show $x * 0 = 0$ is true in any field.

3.1 Galois Field

One particularly useful finite field is GF(2), which has just two elements: zero and one. No field can have fewer than this. We define addition and multiplication in the following ways:

+	0	1
0	0	1
1	1	0

Figure 2: Addition in GF(2)

*	0	1
0	0	0
1	0	1

Figure 3: Multiplication in GF(2)

Gaussian elimination works exactly the same over GF(2) as it did with ordinary numbers. This fact is exploited extensively in coding theory and cryptography.

4 Analytic Geometry

4.1 In the Plane

Given two points with coordinates $\begin{pmatrix} p_1 \\ p_2 \end{pmatrix}$ and $\begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$, the distance d between them is computed with $d = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}$. This works for both negative and positive coordinates.

4.1.1 Vectors

We can also interpret a pair of numbers $\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$ as a movement in the plane v_1 units parallel to the x-axis and v_2 units parallel to the y-axis. Lower case letters with arrows above are used to notate vectors: $\overrightarrow{v}, \overrightarrow{w}, \overrightarrow{u}$ etc. A vector also has a length, once again computed by the Pythagorean theorem $|\overrightarrow{v}| = \sqrt{v_1^2 + v_2^2}$. This is the distance a point travels under the movement described by $\overrightarrow{v}$. A vector of length 1 is called a **unit vector**.

We define $\overrightarrow{0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ as the **null vector**. For all vectors $\overrightarrow{v} \neq \overrightarrow{0}$, there is a unit vector which points in the same direction. This is computed by:

$$\frac{\overrightarrow{v}}{|\overrightarrow{v}|} = \begin{pmatrix} v_1/|\overrightarrow{v}| \\ v_2/|\overrightarrow{v}| \end{pmatrix}$$

Movements via vectors can be composed: we write $\overrightarrow{v} + \overrightarrow{w}$ for the movement of $\overrightarrow{v}$ followed by $\overrightarrow{w}$. The result is a straight line movement with coordinates $\begin{pmatrix} v_1 + w_1 \\ v_2 + w_2 \end{pmatrix}$. Movements can also be shrunk by a **scalar** as such: $\frac{1}{2} \cdot \overrightarrow{v} = \begin{pmatrix} v_1/2 \\ v_2/2 \end{pmatrix}$

4.1.2 Representing Straight Lines

Given a point P and a vector $\overrightarrow{v} \neq \overrightarrow{0}$, we can consider all points X that you can reach from P by following some distance along the direction along the vector $\overrightarrow{v}$. We write $X = P + s \cdot \overrightarrow{v}$ for this, where s is allowed to be a positive or negative number. This is called the **parametic** representation of a line where the **parameter** is s. If we are given two different points, P and Q in the plane then this defines a straight line whose generating expresion is $X = P + s \cdot \overrightarrow{PQ}$

4.1.3 Intersecting Two Lines

If we have two lines and $Y = Q + t \cdot \overrightarrow{w}$ we can find their point of intersection as so. Intersection point satisfies $X = Y$ so

$$\begin{pmatrix} p_1 + sv_1 \\ p_2 + sv_2 \end{pmatrix} = P + s \cdot \overrightarrow{v} = Q + t \cdot \overrightarrow{w} = \begin{pmatrix} q_1 + tw_1 \\ q_2 + tw_2 \end{pmatrix}$$

This gives us two ordinary equations, one for each coordinate:

$$p_1 + sv_1 = q_1 + tw_1$$
$$p_2 + sv_2 = q_2 + tw_2$$

This has two unknown, s and t. s tells us how far in the direction of $\overrightarrow{v}$ we have to move from P in order to reach the point of intersection, and likewise for t and $\overrightarrow{w}$. We can compute either s or t through Gaussian elimination:

$$v_1 s - w_1 t = q_1 - p_1$$
$$v_2 s - w_2 t = q_2 - p_2$$

4.2 In Three Dimensions

For a coordinate system in three dimensions, we need a third axis which is usually called the z-axis. The z-axis must be orthogonal to the other two axes, and the unit of length must be the same on all three. The normal rules for vectors and points, such as length and distance between points apply.

4.2.1 Planes

The parametric representation of a plane has the form $X = P + s \cdot \overrightarrow{v} + t \cdot \overrightarrow{w}$. P is a point and $\overrightarrow{v}$ and $\overrightarrow{w}$ are vectors (which cannot be the null vector). Further, $\overrightarrow{w}$ must not be the same as $\overrightarrow{v}$, otherwise just a line will be generated.

Three points P, Q and R which are not all on the same line determine a plane:

$$X = P + s \cdot \overrightarrow{PQ} + t \cdot \overrightarrow{PR}$$

From this, we have a variety of intersection tasks we may wish to compute:

1. Test whether a given point lies on a give line or a plane
2. Find the intersection point between two lines
3. Find the intersection point between a line and a plane
4. Find the intersection line between two planes

All of these problems can be solved in the same way as described earlier, using a system of linear equations for each coordinate.

4.2.2 Two-Point Description of a Line

The vector that moves P to Q is denoted by $\overrightarrow{PQ}$, and has the coordinates $\begin{pmatrix} q_1 - p_1 \\ q_2 - p_2 \end{pmatrix}$ in 2D. The parametric representation for the line through P and Q can therefore be written as

$$X = P + s \cdot \overrightarrow{PQ}$$

Using the laws of vector algebra we can write this as follows:

$$\begin{aligned} X &= P + s \cdot (Q - P) \\ &= P + s \cdot Q - s \cdot P \\ &= (1 - s) \cdot P + s \cdot Q \end{aligned}$$

The **two-point description of a line** is therefore $X = (1 - s) \cdot P + s \cdot Q$. Sometimes this may be written as $X = s \cdot P + t \cdot Q$ with the side condition that $s + t = 1$.

4.2.3 Three-Point Description of a Plane

Just as we did in the previous section, we can also do the exact same as we did with lines to a plane in order to produce the **three-point description of a plane**. The three point description of a plane is $X = (1 - s - t) \cdot P + s \cdot Q + t \cdot R$.

4.3 Vector Spaces

We have discussed the operations and laws of vectors earlier. To repeat: we have a **null vector**, we can add two vectors and we can multiply vectors with a scalar. We can also solve the equation $\overrightarrow{v} + \overrightarrow{x} = \overrightarrow{0}$ for $\overrightarrow{x}$ by simply setting $\overrightarrow{x} = (-1) \cdot \overrightarrow{v}$. Any structure that carries these two operations and satisfies these laws is called a **vector space**.

We can go one step further an realise that scalars could come from any field $\mathbb{F}$, such as GF(2). Then we say that we have a 'vector space over $\mathbb{F}$'.

If $\overrightarrow{v}$ is an element of some **vector space** then we can generate a **subspace**(sometimes called a **sub-vector space**) by considering all vectors of the form $s \cdot \overrightarrow{v}$, which contains the null vector.

Another way to look at this is to look at our parametric representations. We pick a point and allow all movements from a subspace to act on this point. This construction leads to what in general is called an **affine subspace**. In other words, lines and planes are affine subspaces of 2D/3D.

4.3.1 Bases

We now know how to generate subspaces. This works well and yields the expected result unless we choose the generators unwisely. For example, if one of the generators is the null vector then that does not contribute anything. Also, if two generators point in the same direction (one is a scalar multiple of the other) then we are introducing redundancy as one of them could be dropped. A set of generators that can not be made any smaller is called a **basis** of the subspace. The number of elements of the basis is called the **dimension** of the subspace.

If we start with a set of generators and we are not sure whether is is redundant or not, then we can write the vectors as rows into a matrix and run Gaussian elimination (without any right hand side). The rows of the resulting echelon form will form a basis that generates the same subspace as the generating set from before.

4.3.2 Codes

Subspaces in $\mathrm{GF}(2)^n$ are used as linear **codes** in coding theory. Coding in this sense has nothing at all to do with programming or cryptography; its purpose is to spot and correct errors in the storage and transmission of data. Without coding, CDs and DVDs could not exist.

5 A Different Way of Representing Lines and Planes

Conside the linear equation $x_1 - 2x_2 = 3$. The solutions for this are given by

$$\begin{aligned} x_2 &: \text{ choose freely} \\ x_1 &= 3 + 2x_2 \end{aligned}$$

We can write the possible solutions in a vector form:

$$X = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 3 + 2x_2 \\ x_2 \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \end{pmatrix} + x_2 \cdot \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

From this, we can see all the solutions for this set of linear equations lies upon a line. This is true in general: the solution space of a linear equation in two unknowns is a line in 2D. The same is true for planes and systems with three unknowns.

5.1 Translating Between the Parametric and Equational Representations

5.1.1 Lines

If we are given the parametric representation of a line in 2D

$$X = P + s \cdot \overrightarrow{v} = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + s \cdot \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

and we want to find an equation in the form $ax_1 + bx_2 = d$ which describes the same line then all we need to do is set

$$\begin{aligned} a &= -v_2 \\ b &= v_1 \\ d &= ap_1 + bp_2 \end{aligned}$$

5.1.2 Planes

As we have just done for lines, we can do the same for planes. Given the parametric representation

$$X = P + s \cdot \overrightarrow{v} + t \cdot \overrightarrow{w} = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} + s \cdot \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} + t \cdot \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$$

and we are looking for an equation

$$ax_1 + bx_2 + cx_3 = d$$

which describes the same plane then all we need to do is set

$$\begin{aligned} a &= v_2w_3 - v_3w_2 \\ b &= v_3w_1 - v_1w_3 \\ c &= v_1w_2 - v_2w_1 \\ d &= ap_1 - bp_2 + cp_3 \end{aligned}$$

6 Matrices

7 Sets

7.1 Cardinality

8 Functions and Relations

8.1 Relations

8.2 Functions

9 Inductive Definitions

10 Probability

10.1 Discrete Random Variables

10.2 Continuous Random Variables

Index

affine subspace, 5

basis, 5

codes, 5

dimension, 5

echelon form, 1, 2

field, 2

gaussian elimination, 1, 4

linear equation, 1

null vector, 3

parametric, 4

scalar, 3
sub-vector space, 5
subspace, 5

two point description of a line, 5

unit vector, 3

vector space, 5