COMP11120 Notes

Todd Davies

January 16, 2014

1 Discrete Structures

1.1 Terminology

A *structure* consists of certain *sets*. It also contains *elements* of these sets, *operations* on these sets and *relations* on these sets.

1.2 Number systems to learn

The following number must be learnt:

- $\mathbb{N}$ The set of natural numbers (all whole numbers from 0 to ∞)
- $\mathbb{Z}$ The set of integers (all whole numbers from $-\infty$ to ∞)
- $\mathbb{Q}$ The set of rational numbers (any integer divided by any other integer e.g. $\frac{5}{4} = 1.25$)
- $\mathbb{R}$ The set of real numbers (all finite and infinite decimal numbers)

1.2.1 Operations

Each number system has a set of valid operations that can be performed on elements in that system. Number systems only contain operations that will produce an output that is still within the number system.

For example, the number system $\mathbb{N}$ contains the operations of addition and multiplication. This is because the summation of any two positive integers will *always* be a member of $\mathbb{N}$, and the same goes for multiplication.

However, you may be wondering why subtraction and division aren't included in this number system. This is because for some numbers, the result of subtraction or division won't be inside the set $\mathbb{N}$. An example of this would be subtracting 4 from 2. Even though both of the operands are inside $\mathbb{N}$, the answer isn't.

Different sets may have different operations available. For example, you can concentrate any two members of the set $\mathbb{S}tring$ and end up with another $\mathbb{S}tring$.

Types of operation Operations that have an operand either side of them are called *infix* operations. An example of an infix operation is addition. Infix operations are also referred to as *binary* operations since they have *two* operands.

An example of other types of operations is a unary operation, which would take only one operand. Unary operations can be placed either before the operand (prefix) or after it (postfix).

Commutativity If an operation is commutative, the order of the operands doesn't matter. For example, addition is commutative since:

$$a + b = b + a$$

Subtraction however, isn't commutative:

$$a - b \neq b - a$$

Associativity An operation is associative if inserting or changing brackets doesn't change the outcome of the operation. For example, multiplication is associative since:

$$(a \times b) \times c = a \times (b \times c)$$

An operation may only be commutative or associative if it is commutative or associative for *all* elements of the set that the operation supports.

Distinguished elements A set may contain distinguished elements that have notable effects on certain operations in the set. An example is the number 1. If we multiply *something* by 1, then the result will always be the same *something*. The same goes for 0 with addition. Because of this, we refer to 1 and 0 as distinguished elements of the set $\mathbb{Z}$.

1.2.2 Relations

Each of the sets $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$ carries binary comparison relations $\leq$ and $<$. Different sets (such as $\mathbb{S}tring$) may have other relations, such as:

- Is a section of

- Is an initial section of
- Occurs in

All relations return values in the set $\mathbb{B}ool$.

1.3 Bases

Conventionally, we count using base 10. Base 10 includes, you guessed it, ten different symbols from 0 through to 9.

Sometimes however, it is convenient to count using different bases. Popular bases include:

Base n	**Member symbols**	**Name**
$n = 2$	$\mathbb{Z}_2 = \{0, 1\}$	Binary
$n = 8$	$\mathbb{Z}_8 = \{0, 1, 2, 3, 4, 5, 6, 7\}$	Octal
$n = 10$	$\mathbb{Z}_{10} = \{0, 1, \ldots, 7\}$	Decimal
$n = 16$	$\mathbb{Z}_{16} = \{0 - 9, A - F\}$	Hexadecimal

1.3.1 How to read numbers in any given base

The formula for reading a number in a given base is as follows:

$$\sum_{i=0}^{k} a_i b^i$$

Where the number you're trying to read takes the form $a_k, a_{k-1}, \ldots,$ b is the base you're using and i is the count from right to left of the digits in the number.

Example 1 Lets apply the formula to the base 10 number `27385`:

$$\begin{aligned}
27385 &= (5 \times 10^0) + (8 \times 10^1) + (3 \times 10^2) + (7 \times 10^3) + (2 \times 10^4) \\
&= (5 \times 1) + (8 \times 10) + (3 \times 100) + (7 \times 1000) + (2 \times 10000) \\
&= 5 + 80 + 300 + 7000 + 20000 \\
&= 27385
\end{aligned}$$

Example 2 Lets apply the formula to the base 16 number `F00BA4`:

$$\begin{aligned}
F00BA4 &= (4 \times 16^0) + (A \times 16^1) + (B \times 16^2) \\
&\quad + (0 \times 16^3) + (0 \times 16^4) + (F \times 16^5) \\
&= (4 \times 16^0) + (10 \times 16) + (11 \times 256) \\
&\quad + (0 \times 4096) + (0 \times 65536) + (15 \times 1048576) \\
&= 4 + 160 + 2816 + 0 + 0 + 15728640 \\
&= 15731620
\end{aligned}$$

1.3.2 Changing from base 10 to base n

In order to change into base n from base 10, we just repeatedly divide by n and use the remainder as the value for base n. Here are a few examples:

Example 1 Convert 893 into base 2.

$$
\begin{array}{rcll}
893 \div 2 & = & 446 & \text{r1} \\
446 \div 2 & = & 223 & \text{r0} \\
223 \div 2 & = & 111 & \text{r1} \\
111 \div 2 & = & 55 & \text{r1} \\
55 \div 2 & = & 27 & \text{r1} \\
27 \div 2 & = & 13 & \text{r1} \\
13 \div 2 & = & 6 & \text{r1} \\
6 \div 2 & = & 3 & \text{r0} \\
3 \div 2 & = & 1 & \text{r1} \\
1 \div 2 & = & 0 & \text{r1}
\end{array}
$$

Reading up from the bottom, we can see that the binary (base 2) representation is 1101111101.

Example 2 Convert 893 into base 9.

$$
\begin{array}{rcll}
893 \div 9 & = & 99 & \text{r2} \\
99 \div 9 & = & 11 & \text{r0} \\
11 \div 9 & = & 1 & \text{r2} \\
1 \div 9 & = & 0 & \text{r1}
\end{array}
$$

Reading up from the bottom, we can see that the nonal (base 9) representation is 1202.

Example 2 Convert 893 into base 16.

$$
\begin{array}{rcll}
893 \div 16 & = & 55 & \text{r13} \\
55 \div 16 & = & 3 & \text{r7} \\
3 \div 16 & = & 0 & \text{r3}
\end{array}
$$

Reading up from the bottom, we can see that the hexadecimal (base 16) representation is $3, 7, 13$ or $37D$.

1.4 A structure for the integers

The set $\mathbb{Z}$ of all integers $\{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$ includes the subset $\mathbb{N}$ of all natural numbers together with the negative integers. We will be using three basic binary operations on the carrier set $\mathbb{Z}$; addition, multiplication and subtraction.

The standard notation for these operations is $+$, $\times$ and $-$ and they are used as infix operations.

Sometimes different notations are used, for example, in many programming languages * is usually used for multiplication, and there is a convention to write xy as an abbreviation for $x \times y$.

Both + and x are **commutative** but - is not. This is because:

1. For all integers x,y both, $x + y = y + x$ and $xy = yx$
2. There are integers x,y with $x - y \neq y - x$.

Commutativity ensures that for most purposes the order in which the two arguments are consumed is irrelevant.

Both + and x are **associative** but - is not:

1. For all integers x,y,z both $(x+y)+z = x+(y+z)$, $(xy)z = x(yz)$.
2. There are integers x,y,z with $(x - y) - z \neq x - (y - z)$.

Associativity ensures that for most purposes brackets are not needed to punctuate an expression. For instance, we can make sense of $x + y + z$ and xyz since it doesn't matter where the brackets are, the resulting values are the same.

Below are the definitions of commutativity and associativity.

1. Any operation '$\circledast$' is **commutative** if and only if

$$a1 \circledast a2 = a2 \circledast a1$$

for all $a1, a2, \in, A$

2. Any operation '$\circledast$' is **associative** if and only if

$$(a1 \circledast a2) \circledast a3 = a1 \circledast (a2 \circledast a3)$$

for all $a1, a2, a3, \in, A$

Note:

N.b. Commutativity and associativity don't always have to go together. An example of such an operation is the average of two numbers.

1.5 The formal language

We use the formal language to talk about $\mathbb{Z}$. Expressions in the formal language are built up from atoms in a recursive fashion, so an expression may be $(x \circ y)$ where x and y are expressions and $\circ$ is one of the three operations $(+, -, \times)$. The brackets are important in the formal language, since they ensure that strings can only be parsed in one way.

Strings that contain literals that aren't neutral elements or have invalid bracketing are not valid in the formal language. Examples may include:

$$(x + y + z)$$

$$(x + 2)$$

$$x((y$$

We can use a parse tree to show how an expression is built up from it's identifiers $(a, b, c \ldots)$, constants $(0, 1 \ldots)$ and operations $(+, -, \times \ldots)$.

For example, the parse tree of $(x \times (y + z))$ is:

$$\dfrac{x \quad \dfrac{y \quad z}{(y+z)}\,(+)}{(x \times (y+z))}\,(\times)$$

Often, especially with large parse trees, it's a pain to write so many identifiers. Because of this, it is a convention to replace identifiers with dots after they've been used once, like so:

$$\dfrac{x \quad \dfrac{y \quad z}{\cdot}\,(+)}{\cdot}\,(\times)$$

In order to parse a parse tree, we must assign an appropriate value to each of the 'leaves' of the tree (i.e. all the identifiers) and let the values trickle down towards the root node of the tree where the evaluated answer will appear.

Lets use the above example again. Let $x = 4$, $y = 6$ and $z = 2$:

$$\dfrac{4 \quad \dfrac{6 \quad 2}{8}\,(+)}{32}\,(\times)$$

At this point, parse trees may seem very pointless, but this is because we're doing very simple arithmetic. However, when we start to define other operators that do unfamiliar things or don't use the *infix* notation, then using parse trees can be a big help!

> **Note:**
>
> The infix notation is when an operator is placed between two operands, e.g. $2+2$. Other notations include the *prefix* and *postfix* notations.

1.6 The properties of sets

All sets have some properties in common that we can use to manipulate them. Examples include membership, equality, inclusion etc.

1.6.1 Set membership

To indicate that an element is a member of a set, we use the $\in$ symbol. For example, to say that the element *true* is a member of the set $\mathbb{B}ool$, we would write:

$$true \in \mathbb{B}ool$$

Interestingly, we can parse this into English in many ways though. It could mean any of the following things:

- $true$ is an element of the set $\mathbb{B}ool$
- $true$ is an member of the set $\mathbb{B}ool$
- $true$ is contained in $\mathbb{B}ool$
- $\mathbb{B}ool$ contains $true$

Conversely, to indicate that an element is not a member of a set, we use the symbol $\notin$:

$$sheep \notin \mathbb{B}ool$$

Note that there is no concept of *order* or *repetition* in sets. This means that the following sets are all equal:

$$\{1, 2, 3\}$$

$$\{2, 3, 1\}$$

$$\{2, 3, 1, 2\}$$

$$\{3, 3, 3, 3, 3, 2, 1\}$$

1.6.2 Set equality

Sets are equal if they have exactly the same members. Note that as far as sets are concerned, duplicate members are treated as just one member.

The notation for set equality is very easy. To say the set X is equal to the set Y, we write:

$$X = Y$$

However, you must understand that this is only true if for each element a in X that element will also be contained in Y and for each element a in Y, that element will also be contained within X:

$$\text{For each } a, a \in X \leftrightarrow a \in Y$$

Note:

Sets with different descriptions can still be equal. Convince yourself that the set of integers where $y^3 < y$ is equal to the set of integers where $x < -1$

1.6.3 Set inclusion

If one set is a subset of another set, all the members of the first set are also found within the second set. In mathematical terms:

$$\text{For each } a, a \in X \rightarrow a \in Y$$

The notation for inclusion is $\subseteq$, so in the above example, we would write:

$$X \subseteq Y$$

Note:

If $X \subseteq Y$ and $Y \subseteq X$ then what else an we say about the relationship between X and Y?

1.6.4 The empty set

The empty set is a set that contains no members at all. It's symbol is $\emptyset$.

Because the empty set has no members, it is a subset of all other sets:

$$\emptyset \in A$$

Note:

This is because otherwise $x \in \emptyset$ would be true for some x where $x \notin A$. This is impossible since there are no elements in the $\emptyset$.

1.6.5 Singleton sets

For any entity a, we can form a set consisting only of a:

$$\{a\}$$

Be aware, a singleton set is not the same as the element contained within the set:

$$a \neq \{a\}$$

1.6.6 Set union

There are several ways of combining multiple sets together to create another set. One such method is set union, the symbol of which is $\cup$. The union of two sets is a set containing all the members of *both* sets. For example:

$$A = \{1, 3, 5, 7, 9\}$$

$$B = \{1, 1, 2, 3, 5, 8, 13\}$$

$$A \cup B = \{1, 2, 3, 5, 7, 8, 9, 13\}$$

We could also define the union of two sets mathematically, like so:

$$x \in A \cup B \leftrightarrow x \in A \text{ or } x \in B$$

1.6.7 Set intersection

Another way of combining sets is intersection. Intersecting two sets will produce a set of elements that belong to both of the sets. The symbol for intersection is $\cap$.

If we defined intersection mathematically, we would do so like this:

$$x \in A \cap B \leftrightarrow x \in A \text{ and } x \in B$$

1.6.8 Relative complement

The relative complement of two sets is all the members in the first set that aren't members of the second set. The symbol for the relative complement is $-$. Defined mathematically, we get:

$$x \in A - B \leftrightarrow x \in A \text{ and } x \notin B$$

1.6.9 The universal set

The universal set contains *all of the possible elements*. The notation we use to describe the universal set is S. We can define the De Morgan's laws using the universal set:

$$X' = S - X$$

1.7 Boolean algebra of sets

If we have a universal set, S, and consider only subsets of S. Then these are all the things that we have:

- Two distinguished subsets $\emptyset$ and S
- A unary operation $(.)'$ on such subsets
- Two binary operations; $\cap$ and $\cup$ on such subsets

This structure of operations and sets is known as **Boolean algebra**. Boolean algebra has many basic properties that we can exploit in order to manipulate expression into other forms. Find them on the flashcards that go with these notes.

2 Propositional Logic

2.1 The logical connectives

Most of these are very similar to the properties of sets that we came across in the boolean algebra section. Learn these:

Name	Symbol	Meaning
Negation	$\neg$	not
Conjunction	$\wedge$	and
Disjunction	$\vee$	or
Implication	$\implies$	implies
Bi-implication	$\iff$	iff

Note:

We use *inclusive* or in propositional logic, not *exclusive* or.

Note:

Iff is an abbreviation for *if and only if*.

All of these symbols are used as infixes, and so go in between two parameters. An exception to this however, is negation, which is a unary symbol, and is placed as a prefix before it's argument.

Note:

If a connective takes one parameter, then it's arity is 1, if it takes two then it has an arity of 2 etc

2.1.1 The truth tables of the logical connectives

Using a truth table, we can fully describe the behaviour of the connectives we have just described in the previous section:

p_1	p_2	$\neg p_1$	$p_1 \wedge p_2$	$p_1 \vee p_2$
T	T	F	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	F

p_1	p_2	$p_1 \implies p_2$	$p_1 \iff p_2$
T	T	T	T
T	F	F	F
F	T	T	F
F	F	T	T

A note about implication: In order to understand why the truth table for implication is as stated above, consider this example:

$$x > 3 \text{ implies } x > 1$$

Now take a look at the truth table for this expression:

x	$x > 3$	$x > 1$	$x > 3 \implies x > 1$
4	T	T	T
2	F	T	T
0	F	F	T

As you can see, there is no way to make the first expression ($x > 3$) true, but the second expression ($x > 1$) false. Henceforth, the first expression implies the second expression.

2.2 The formulae of propositional logic

In order to build up a piece of propositional logic, we must construct an expression from *atomic formulae* and connectives.

> **Note:**
>
> An example of an atomic formula is p, or maybe r_2. It's any single boolean variable.

There are three rules we can use to generate a formulae in PL:

1. Every atomic formula is a formula of PL
2. If A is a formula, then so is $\neg A$
3. If A, B are formulae, then so are $(A \wedge B)$, $(A \vee B)$, $(A \implies B)$, $(A \iff B)$

Syntactic conventions are of course in widespread use.

For example, when you have written an expression, it is usual to leave off the outermost parentheses, so $(p \wedge (r \vee q))$ would become $p \wedge (r \vee q)$.

> **Note:**
>
> This second convention is adopted for all associative operations, not just conjunction and disjunction.

It is also common to leave out parentheses from repeated uses of conjunction or disjunction, so $p \wedge (q \wedge r)$ would become $p \wedge q \wedge r$.

2.3 Truth valuations

A truth valuation is a list of allocations that define the values of variables in an expression, where:

$$\{p_1 = x_1, \ldots, p_n = x_n\}$$

> Note:
>
> An example truth valuation for the expression $p \wedge r$ might be $\{p = T, r = F\}$, for which the end value would be F.

In order to determine the end value of the PL formula, you must first construct a truth valuation of it's component atoms, and only then can the formula be evaluated.

2.4 Tautologies and contradictions

If the outcome of a formula is always T for every possible truth valuation, then the formula is said to be a **tautology**. An example of such an expression is $p \vee \neg p$.

In contrast, a formula that will yield an outcome of F for every possible truth valuation is named a **contradiction**. Such an example may be $p \wedge \neg p$.

If an expression is not a tautology and isn't a contradiction either, then it is named **satisfiable**.

2.5 Truth tables

A truth table lists all the possible truth valuations of an expression and the outcome of the expression for each. In this way, it is easy to determine whether the expression is a tautology, satisfiable or a contradiction.

It's a good idea to split an expression into several component parts inside a truth table so that it's easier to create. For example, take a look at the truth table for $x \vee (y \wedge z)$:

x	y	z	$y \wedge z$	$x \vee (y \wedge z)$
T	T	T	T	T
T	T	F	F	T
T	F	T	F	T
T	F	F	F	T
F	T	T	T	T
F	T	F	F	F
F	F	T	F	F
F	F	F	F	F

2.6 Logical equivalence

Two expressions are said to be logically equivalent if they have the same value for every truth valuation. The syntax two say two expressions A and B are logically equivalent is:

$$A \models\!\!\!\dashv B \tag{1}$$

2.7 Normal forms

A normal form is a specific format that we can re-arrange an expression in to so that it has certain attributes and is perhaps easier to manipulate or are more simple.

2.7.1 Negation Normal Form

Negation Normal Form (NNF) is build up of literals, conjunctions and disjunctions only. In order to get an expression into NNF, we have to do two things:

1. Remove all instances of implication and bi-implication using the logical identities.
2. Use De Morgan's laws to remove the negation of any brackets.

An important fact about Negation Normal Form is that *any formula* in the propositional language can be rearranged so that it is in NNF.

Note:

A literal is an atomic formula or the negation of an atomic formula (e.g. p or $\neg p$). Since NNF is built up of literals, conjunctions and disjunctions, this means you can't have negations of bracketed expressions since a bracketed expression is not a literal.

2.7.2 Conjunctive Normal Form

Conjunctive Normal Form (CNF) is a form where a series of expressions are separated by conjunctions. It is important to

note that any formula in CNF is also in NNF, since all the rules that apply to NNF also apply to CNF. Additional rules for CNF are:

- All the disjunctions must be inside expressions separated by conjunctions.
- There can be no nested brackets.

2.7.3 Disjunctive Normal Form

Disjunctive Normal Form (DNF) is the same as CNF, but the role of the conjunctions is reversed with that of the disjunctions.

To get any expression into CNF or DNF, it is often useful to get it into NNF first, This is because both forms (CNF and DNF) inherit all the rules from NNF, so it makes sense to get the formula into NNF first.

2.8 Testing for contradictions and tautologies using CNF and DNF

It is possible to find if an expression is a contradiction or a tautology just by looking at the truth table of it. In order to find if the expression is a tautology, you must re-arrange it into CNF, then look for a literal and the negation of the literal inside each bracket. If the expression exhibits this quality, then it is a tautology. Here's an example:

$$\neg p_1 \vee p_2 \vee p_3 \vee p_1 \tag{2}$$

An expression that is *not* a tautology might be:

$$(\neg p_1 \vee p_2 \vee p_3 \vee p_1) \wedge (p_1 \vee p_3) \tag{3}$$

This would be because the second expression doesn't contain a literal and the negation of the same literal.

Testing for contradictions is similar, but the expression must be in DNF instead of CNF.

2.9 Functional completeness

Functional completeness is when a languages contains sufficient symbols to enable it to express any conceivable expression. In the language of PL, this the minimum symbols for functional completeness are $\neg, \wedge, \vee$.

2.9.1 Finding a DNF equivalent of any truth table

Given any truth table, it is possible to construct the DNF equivalent of it.

In order to do this, you find a DNF formula for each row that is true in the truth table. The formula contains each of the inputs of the truth table, separated by conjunctions, and negated if their input value is false. Once that is done, then simply combine the formulae with disjunctions. This is best explained using an example:

x	y	z	output	
T	T	T	F	
T	T	F	T	$x \wedge y \wedge \neg z$
T	F	T	T	$x \wedge \neg y \wedge z$
T	F	F	F	
F	T	T	F	
F	T	F	F	
F	F	T	F	
F	F	F	T	$\neg x \wedge \neg y \wedge \neg z$

The DNF formula would be:

$$(x \wedge y \wedge \neg z) \vee (x \wedge \neg y \wedge z) \vee (\neg x \wedge \neg y \wedge \neg z) \tag{4}$$

Since *any* truth table can be represented in this way (using only negations, conjunctions and disjunctions), we can say that these three operators are a **functionally complete set of connectives**.

Note:

In fact, we can use De Morgan's laws to re-arrange the DNF equation so that we only need *either* conjunctions or disjunctions and negations, so there is an element of redundancy built in.

2.10 Predicates and quantifiers

A predicate is a function that returns a boolean. Examples may be:

$Animal(x)$	x is an animal
$Eats(x, y)$	x eats y
$Food(x)$	x is food

This allows us to form complicated, arbitrary statements such as:

$$(Animal(x) \wedge Food(y)) \implies Eats(x, y) \tag{5}$$

A quantifier is a way of specifying what a variable is, or could be. An example could be:

$$\text{For all x, y } (Animal(x) \wedge Food(x)) \implies Eats(x, y) \tag{6}$$

This would specify all the possible x's and y's for the expression. There are two such quantifiers, the **universal quantifier ($\forall$)** and the **existential quantifier($\exists$)**.

The universal quantifier specifies all elements, which in English, usually translates to *for all*. The existential quantifier specifies at least one element, usually translated to something like *there exists*, or *for some*.

Quantifiers can greatly change the meaning of a statement:

$\forall x, y\, (Animal(x) \wedge Food(y)) \implies Eats(x, y)$	All animals will eat all foods.
$\exists x, y\, (Animal(x) \wedge Food(y)) \implies Eats(x, y)$	There is an animal that will eat some specific item of food.

2.10.1 The negation of quantified statements.

There are identities to manipulate quantified statements that have been negated:

$$\begin{aligned} \neg\forall P(x) &\models\!\!\!\!\dashv \exists\neg P(x) \\ \neg\exists P(x) &\models\!\!\!\!\dashv \forall\neg P(x) \end{aligned}$$

3 Probability

3.1 Sample spaces and events

A **Sample Space** is a set of all the possible outcomes of an *experiment*. In this sense, an experiment is taken to be some occurrence that will produce one of many outcomes.

The notation we use for the sample space is Ω.

$$\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$$

An **event** is a set containing one ore more outcomes of an experiment. Since all outcomes are included in the set Ω, all events are also subsets of the sample space. This works both ways, any subset of Ω is an event. We use ω to represent an event.

An event is said to have occurred if the outcome of an experiment contains an element of that event.

Since all events are sets, it makes sense that we would have an empty set and a universal set. As with discrete logic (see page), we also use the empty set in probability. Here, we use it to describe an event that *never* occurs. The notation for the empty set is $\emptyset$. We have already discussed the universal set (see page) of probability, it's the sample space, Ω.

Note:

If Ω has n elements, then there are 2^n possible events.

To sum up events an sample spaces, here's an example. If toss a coin, the following events are generated:

Event	Description
$\emptyset$	Nothing happens. This will never occur.
$\{H\}$	The coin shows heads.
$\{T\}$	The coin shows tails.
Ω	The coin shows either heads or tails. This event always occurs.

Note:

Note, events can, of course have multiple elements. If we were examining a dice throw, we could define the event A to contain all the outcomes that are even numbers, so $A = \{2, 4, 6\}$

3.2 More set properties

All of the set operations we discussed in the Discrete Structures section of the notes still apply here, these can be found in section 1.6 on page .

However, there are three additional properties that will be useful in this section.

3.2.1 The number of items in a set

We can use the operator # to find the number of elements in a set. If we have an event A that contains the elements $\{\omega_1, \omega_2, \omega_3\}$ then the number of outcomes it contains is three, therefore $\#A = 3$.

3.2.2 Set difference

If we have two sets A and B, and B is a subset of A (i.e. $B \subset A$), then we can find the set difference between A and B.

The set difference is defined as **all the elements in A that are not in B.** The symbol for set difference is "\".

Mathematically, this is:

$$A \; B \; = \{\omega \in \Omega \mid \omega \in A \text{ and } \omega \notin B\}$$

3.2.3 Disjoint sets

If two sets are disjoint, then they share no common elements, that is to say that:

$$A \cap B = \emptyset$$

This is important in probability, since if two events are disjoint, then they will never occur at the same time.

3.3 Probability measures

A probability measure is a mapping between an event and the probability that the event will occur.

$$\begin{aligned} \mathbb{P} : \{&\textit{Collection of all events}\} \\ &\rightarrow [0, 1] \end{aligned}$$

This defines $\mathbb{P}$ as a number between 0 and 1 (i.e. a probability). Since a probability measure is defined for *all* events that can occur, then the following are true:

- $\mathbb{P}(\Omega) = 1$
- If A and B are disjoint events then $\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B)$
- If $\mathbb{P}(A) = 0$ then A will never occur.
- If $\mathbb{P}(A) = 1$ then A will always occur.

Note:

If $\mathbb{P}(A) = 1$, then that doesn't mean that $A = \Omega$, since there could be other events ω where $\mathbb{P}(\omega) = 0$

In order to verify whether any given mapping is a probability measure, you must check it against the first two of the above bullets; that the sum of all the mappings is 1 and the mapping of the union of any two disjoint events is the same as the sum of the separate mappings of the disjoint events.

3.3.1 Indicator functions

An indicator function will yield a value of 1 if an element ω is inside a set, and a value of 0 if the element is not inside the set. It can be defined as follows:

$$1_A(\omega_i) = \begin{cases} 1 \text{ if } \omega_i \in A \\ 0 \text{ if } \omega_i \notin A \end{cases}$$

We can therefore define the probability of an event A to be the sum of all the probabilities of the events inside A:

$$\mathbb{P}(A) = \sum_{i=1}^{n} 1_A(\omega_i)p_i$$

Note:

Note that n is the number of elements inside A.

3.4 Computing the probability of equally likely functions

If we have a sample space $\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$ where each element is equally likely to occur, then we can say that $p_i = \mathbb{P}(\{\omega_i\})$. Since all the elements are equally likely to occur, we can say that $p_1 = p_2 = \cdots = p_i$ and also that $p_1 + p_2 + \cdots + p_i = 1$.

Using ideas from the previous section, we can find the probability of any event A inside a sample space where all elements are equally likely. To do this, all we need to do is find the sum of all the elements in the sample space, but multiply the elements by the indicator function of A before we add them onto the sum:

$$\mathbb{P}(A) = \sum_{i=1}^{n} 1_A(\omega_i)p_i$$

However, since all the elements inside Ω are equally likely, that must mean that for any event ω_i, it's probability must be $\frac{1}{n}$ where n is the number of elements inside the sample space. If we take this into account, we can formulate the following formula:

$$\mathbb{P}(A) = \frac{1}{n}\sum_{i=1}^{n} 1_A(\omega_i)$$

However, we can make this even more simple; if we recognise that any element in A has the same probability of occurring as any element in Ω, then it's easy to realise that the probability of A occurring will be equal to the relationship between sizes of the sets A and Ω:

$$\mathbb{P}(A) = \frac{\#A}{\#\Omega}$$

3.4.1 Finding the binomial coefficient

The binomial coefficient is represented as $\binom{n}{k}$, both n and k are integers. It is pronounced n *choose* k.

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!}$$

3.4.2 Choosing items

If we have a set of three items a, b and c, then how many different ways can we choose two of those three items from the set?

This depends on two things. Does the order of the items matter, and can we choose an item twice?

If the order is important and we can choose an item many times, then we can choose the following items:

$$(a,a),(a,b),(a,c),(b,a),(b,b),(b,c),(c,a),(c,b),(c,c)$$

If the order is not important, then we have:

$$(a,a),(a,b),(a,c),(b,b),(b,c),(c,c)$$

If the order is important, but we can't choose an item twice, we get:

$$(a,b),(a,c),(b,a),(b,c),(c,a),(c,b)$$

Finally, if the order is not important and the items cannot be selected twice, then:

$$(a,b),(a,c),(b,c)$$

In order to calculate the number of possible different selections we can make in these different cases, we can use this table:

	Order important?	
	Yes	No
Without replacement	$\frac{n!}{(n-k)!}$	$\binom{n}{k}$
With replacement	n^k	Complicated

It's easy to verify that these are true; since:

- With replacement, order important:

$$\begin{aligned} n^k &= 3^2 \\ &= 9 \end{aligned}$$

- Without replacement, order important:

$$\begin{aligned}\frac{n!}{(n-k)!} &= \frac{3!}{(3-2)!} \\ &= 6\end{aligned}$$

- Without replacement and order is not important:

$$\begin{aligned}\binom{n}{k} &= \frac{n!}{k(n-k)!} \\ &= \frac{3!}{2(3-2)!} \\ &= \frac{6}{2} \\ &= 3\end{aligned}$$

Note:

See chapter 2 in the course provided notes for (copious) examples of the content covered in this section.

3.5 Conditioning

If an experiment is performed where we already know that a certain event will occur, then we can factor in an additional factor into our probabilistic theories. This factor could be called a condition.

An example is throwing a dice. If we define the event $B = \{5, 6\}$ then there is a probability of $\frac{1}{3}$ that the event B will occur. If we know in advance that B will occur, then the probabilities that the events outside B will occur suddenly drop to zero:

$$\mathbb{P}(\{1\}) = \cdots = \mathbb{P}(\{4\}) = 0$$

The probability that B will occur is 1, therefore the probability that either event in B will occur is $\frac{1}{2}$.

To clarify the following two things have just happened:

- All outcomes not in the event B have a probability 0.
- All outcomes in the event B have their probability *scaled* by a factor of $\frac{1}{\mathbb{P}(B)}$.

In a more general case, the probability that any event A occurs, given that the event B occurs is:

$$\frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}$$

The notation we use for conditional probability (i.e. the probability an event A will occur given that the event definitely will B occur) is:

$$\mathbb{P}(A|B)$$

Note:

It's correct to say that the definition of A given B is a probability measure. The probability measure would be concentrated on the event B rather than the whole sample space *Omega* since all events outside B are given the probability 0.

3.5.1 Independent events

If the probability of the event A does not change when we add the conditional that an event B will certainly occur and $\mathbb{P}(B) \geq 0$, then the events A and B are independent. This could be described as:

$$\mathbb{P}(A) \ = \mathbb{P}(A|B)$$

We can manipulate this mathematical definition of independence using the equations we described in the at the start of the conditionals section:

$$\begin{aligned}\mathbb{P}(A) \ = \mathbb{P}(A|B) \iff \mathbb{P}(A) &= \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} \\ \iff \mathbb{P}(A)\mathbb{P}(B) &= \mathbb{P}(A \cap B)\end{aligned}$$

Henceforth, we often take the last of these equations as our definition for independence.

Of course, it's easy to do the following too:

$$\begin{aligned}&\mathbb{P}(A)\mathbb{P}(B) \ = \mathbb{P}(A \cap B) \\ &\iff \mathbb{P}(B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(A)} \\ &\iff \frac{\mathbb{P}(B \cap A)}{\mathbb{P}(A)} \\ &\iff \mathbb{P}(B|A)\end{aligned}$$

$$\mathbb{P}(A|B) \iff \mathbb{P}(B|A)$$

So, an event is independent of another event if knowledge that one event occurs won't give you any extra information about how likely it is that the other event will occur. This doesn't mean that the events are disjoint (unless the probability of either was 0).

3.5.2 Partitions

Partitions are sections of a sample space that:

- Don't overlap.
- Include all elements inside the sample space between them.

So, suppose I had partitions $E_0 \dots E_n$, the following would hold:

- $E_i \cap E_i = \emptyset$ for and i and j
- $E_0 \cup E_1 \cup \dots \cup E_n = \Omega$

This means that all partitions are mutually disjoint.

> Note:
>
> The simplest partition is to take an event A and it's complement A^c.

Note that the probability of the union of any two partitions will be the sum of the probability of the separate partitions, since all partitions are disjoint.

3.5.3 The law of total probability

The law of total probability splits any event A up in terms of the partitions in the sample space:

$$A = (A \cap E_i) \cup (A \cap E_2) \cup \ldots \cup (A \cap E_n)$$

We can write this in terms of probabilities:

$$\mathbb{P}(A) = \mathbb{P}(A|E_1)\mathbb{P}(E_1) + \cdots + \mathbb{P}(A|E_n)\mathbb{P}(E_n)$$

3.5.4 Bayes Theorem

The derivation of Bayes theorem is quite easy; start with an event partitioned into m partitions, then take $\mathbb{P}(E_i|A)$ and use first the definition of conditional probability, then the multiplicative law, then the law of total probability. The process is described mathematically below:

$$\begin{aligned}
\mathbb{P}(E_i|A) &= \frac{\mathbb{P}(E_i \cap A)}{\mathbb{P}(A)} \\
&= \frac{\mathbb{P}(A|E_i)\mathbb{P}(E_i)}{\mathbb{P}(A)} \\
&= \frac{\mathbb{P}(A|E_i)\mathbb{P}(E_i)}{\mathbb{P}(A|E_1)\mathbb{P}(E_1) + \cdots + \mathbb{P}(A|E_m)\mathbb{P}(E_m)}
\end{aligned}$$

Bayes theorem is good if you have the probability of some event A that depends on the condition of the event B happening (i.e. you know $\mathbb{P}(A|B)$, but you instead want to know $\mathbb{P}(B|A)$.

3.5.5 The complementary law for conditional probabilities

The complementary law states that:

$$\mathbb{P}(A|B) = 1 - \mathbb{P}(A^c|B)$$

3.6 Random variables

A random variable is a mapping between the sample space Ω to real numbers ($\mathbb{R}$). A random variable is usually denoted by X.

$$\begin{aligned} X : \Omega \\ \rightarrow \mathbb{R} \end{aligned}$$

This means that every outcome in the sample space corresponds to a real number. The range of the random variable is very intuitive; if we were throwing a dice, then the range would be $\{1, 2, \ldots, 6\}$.

The sum of the probabilities of the random variable is always equal to 1:

$$\sum_{i=1}^{m} \mathbb{P}(X = r_i) = 1$$

3.6.1 Constructing new random variables from existing ones

Creating a random variable from another random variable is fairly easy; it's basically a composite function. If we have a random variable X and a function f then:

$$Y(\omega) = f(X(\omega)) \text{ for all } \omega \in \Omega$$

Note that the range of the new function (Y) may be different from the range of the old one (X).

3.6.2 Probability mass functions

Usually abbreviated to *pmf* a probability mass function is a function (denoted by p) that contains information regarding the probabilities of X in given ranges.

Since the output of a *pmf* is a probability, the range of a *pmf* is always $[0, 1]$.

Mathematically, the probability mass function of a random variable X is defined as:

$$\begin{aligned} p(r_i) &: \\ &= \mathbb{P}(X \\ &= r_i) \text{ for all } i > 1 \end{aligned}$$

Note:

The sum of all the values in a *pmf* function is always 1.

3.6.3 Cumulative distribution functions

A cumulative distribution function *cdf* is similar to a *pmf* except that:

- The value of the *cdf* is the sum of the *pmf* up to that point.
- A *cdf* can be generated for any function, not just discrete ones like a *pmf* can.

However, both the *cdf* and the *pmf* share the same range. The definition of *cdf* is:

$$F(x) \ = \mathbb{P}(X \leq x) \text{ for all } x \in \mathbb{R}$$

A graph of a *cdf* function contains all the information about the random variable it has mapped. We can deduce it's range, the *pmf* and where the graph jumps. Note that black dots on the graph indicate where a value is, and white dots at the same x coordinate show where the value isn't.

3.6.4 The mean of a random variable

Finding the average value of a random variable is comparable to doing so for a sequence of numbers where each number has a weight. If the numbers are numbered $\alpha_1 \dots \alpha_n$ and their weights are numbered $\omega_1 \dots \omega_n$ then the average value would be:

$$\text{Average value: } = \frac{\omega_1\alpha_1 + \cdots + \omega_n\alpha_n}{\omega_1 + \cdots + \omega_n}$$

Note that the mean is represented by $\mathbb{E}$ so:

$$\mathbb{E} = \frac{\omega_1\alpha_1 + \cdots + \omega_n\alpha_n}{\omega_1 + \cdots + \omega_n}$$

In this analogy, then the values for α would be members of the range of X and the values of ω would be the probabilities that these values would occur:

$$\mathbb{E}(X) = \frac{\mathbb{P}(X = r_1) \cdot r_1 + \cdots + \mathbb{P}(X = r_n) \cdot r_n}{\mathbb{P}(X = r_1) + \cdots + \mathbb{P}(X = r_n)}$$

However, since the denominator of this fraction is merely the sum of the probabilities of each of the elements in the range of X, it will always equal 1. Therefore:

$$\begin{aligned}\mathbb{E}(X) &= \mathbb{P}(X = r_1) \cdot r_1 + \cdots + \mathbb{P}(X = r_n) \cdot r_n \\ &= \sum_{i=1}^{n} r_i \mathbb{P}(X = r_i)\end{aligned}$$

In order to define the mean of a function of a random variable (as described in section 3.6.1), we don't need to apply the function to the probability of each element occurring, only to the element itself:

$$\mathbb{E}(f(X)) = \sum_{i=1}^{n} f(r_i) \mathbb{P}(X = r_i)$$

Of course, if the random variable $Y = f(X(r_i))$ then you can just find the average of Y instead:

$$\mathbb{E}(Y) = \sum_{i=1}^{n} r_i \mathbb{P}(Y = r_i)$$

We can say two more things about the means of random variables:

- If the random variable only has one element in it's range, then the mean of the random variable must be that element.
- For any $a, b \in \mathbb{R}$ we have

$$\mathbb{E}(aX + b) = a\mathbb{E}(X) + b$$

3.6.5 The variance and standard deviation of a random variable

The variance of a random variable is how much on average, a random variable strays from it's mean. This may seem strange, but it makes more sense when applied; a random variable that can take the values $\{-20, 0, 20\}$ will have a greater variance than one that has a range of $\{-2, 0, 2\}$, providing that their respective weightings are the same.

In order to find the variance of a random variable (X), we must first find it's mean (μ). We can then go through each element in the random variable and find how far it is from the mean of the random variable. Let Y denote the new random variable produced:

$$Y(\omega) = (X(\omega) - \mu)^2$$

In order to find the variance of the random variable, we need to find then average amount that the variable strays from it's mean, so we must find the mean of Y:

$$Var(X) = \mathbb{E}[(X - \mu)^2]$$

So, to sum it up, the variance is a measure of how far X is on average from it's mean μ. The larger the variance is, the larger the distance on average is.

Note that when finding the variance, it's imperative that you remember to take into account the probability of each element ω happening.

$$Var(X) = \sum_{i=1}^{m} (r_i - \mu)^2 \mathbb{P}(X = r_i)$$

3.6.6 Standard deviation

The standard deviation of a random variable is merely just the square root of it's variance:

$$SD(X) = \sqrt{Var(X)}$$

3.7 Distributions

A distribution is a combination of *range* and *pmf*. If two different random variables have the same range and the same pmf, then they have the same distribution.

3.7.1 The Bernoulli distribution

The Bernoulli distribution has a range of $R = \{0, 1\}$ and a pmf of $p_x(0) = 1 - p$ and $p_x(1) = p$.

An example of an experiment following this distribution would be the tossing of a coin.

If a random variable X follows the Bernoulli distribution (X *Bernou*
then:

$$\mathbb{E}[X] = p \text{ and } Var(X) = p(1 - p)$$

3.7.2 The Binomial distribution

The binomial distribution deals with situations where you have an experiment that can yield an outcome of *success* or *failure*. It's range is from $\{0, 1, \ldots, n\}$ since it accounts for the experiment being repeated n times.

The probability of having k number of successes in a row out of n tries is:

$$p_x(k) \ = \mathbb{P}(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

The mean of a Binomial distribution is $\mathbb{E}[X] = np$ and the variance is $Var(X) = np(1-p)$.

3.7.3 The Poisson distribution

The Poisson distribution is very similar to the Binomial distribution, but it is suited to applications where n is very large and p is very small. It predicts the probability of a number of k successes occurring within n tries of an experiment.

$$p_x(k) = \frac{\lambda^k}{k!} e^{-\lambda}$$

The mean and the variance both equal λ:

$$\mathbb{E}[X] \ = Var(X) = \lambda$$

3.7.4 The Geometric distribution

The geometric distribution finds the number of experiments you should do before getting your first success.

$$p_x(k) = (1-p)^{k-1}p \text{ for all } k \in \{0, 1, \dots\}$$

The mean and variance of a geometric distribution are:

$$\mathbb{E}[X] = \frac{1}{p}$$

$$Var(X) = \frac{1-p}{p^2}$$