LGIC 010 Textbook

Prof. Scott Weinstein, Owain West, Grace Zhang

January 5, 2018

Contents

1 Prelude

1.1 Prerequisite Notation

Though the course has no specific mathematical prerequisites, a general familiarity with the set of integers and some of its basic properties will be assumed. We collect here some useful facts and notations that will appear from time to time throughout the course. We'll add more as the need arises.

1. Notations for important sets of numbers
   - $\mathbb{Z} = \{\ldots -2, -1, 0, 1, 2, \ldots\}$ (the integers)
   - $\mathbb{N} = \{0, 1, 2, \ldots\}$ (the non-negative integers, a.k.a. the natural numbers)
   - $\mathbb{N}^+ = \{1, 2, 3, \ldots\}$ (the positive integers)
2. Important facts about numbers
   - The Least Number Principle: If X is a nonempty subset of $\mathbb{N}$, then X has a least element.
   - Principle of Mathematical Induction: If X is a subset of $\mathbb{N}$, and $0 \in X$, and for every i, if $i \in X$, then $i + 1 \in X$, then $X = \mathbb{N}$.
   - The Pigeonhole Principle: If you distribute m pigeons into n pigeonholes and $m > n$, then some hole contains more than one pigeon.
3. Unique Factorization into Primes: Recall that $p \in \mathbb{N}^+$ is *prime* if and only if $p \neq 1$ and p is divisible only by 1 and p. Every $n \in \mathbb{N}^+$ with $n \neq 1$ can be written uniquely as $p_1^{a_1} \cdots p_n^{a_n}$ where each p_i is prime and each $a_i \geq 1$.

1.2 A Combinatorial Warmup

Combinatorics is, roughly, the part of mathematics which deals with counting things. Its techniques are general, and its results tangible. Throughout this book, we will use combinatorial problems as concrete examples of problems which can be considered and solved by means of logical techniques. To get our feet wet, let's consider the following principle and question.

Principle 1. ***The Pigeonhole Principle****: If you distribute m pigeons into n pigeonholes and $m \geq n + 1$, then some hole contains at least two pigeons.*

Example 1. *Is there a numerically diverse group of Philadelphians?*

(We call a group of people numerically diverse if no two people in the group have the same number of friends in the group - we assume groups are of size at least two and that friendship is always mutual.)

We will demonstrate that the answer is no by an application of the Pigeonhole Principle.

Proof. Suppose we have a group $G = \{1, \ldots, n\}$ of n people (we use numerals to name the people for privacy concerns). For brevity, let's write p_{ij} to signify that i is a friend of j. We assume friendship is symmetric, that is, if p_{ij}, then p_{ji}, for all $i, j \in G$, and irreflexive, that is, it is not the case that p_{ii}, for all $i \in G$. Let's write $f(i)$ for the number of friends of i, that is, the number of j such that p_{ji}. Since friendship is irreflexive, the possible values of f are the n numbers $0, 1, \ldots, n-1$. We are thinking of these values as the pigeonholes for application of the principle 1 and the members of G as being placed in these holes by f. We want to argue that the value of f must agree on at least two members of G. But so far, since we have n members of G and n pigeonholes into which they are sorted by f, we may not yet draw that conclusion via principle 1. But now we consider the question, "can f really take all the values from 0 to $n - 1$?" In particular, can it take on both the value 0 and the value $n - 1$? We argue that the answer is no. Suppose that there is some i with $f(i) = 0$, that is, for every j, it is not the case that p_{ji}. Then, by symmetry, for every j, it is not the case that p_{ij}. So, if i has no friends, then the maximum number of friends of any j is $n-2$, that is, f cannot take on the value $n-1$. Thus, the possible values of f are the $n-1$ numbers $0, \ldots, n-2$. But now, by principle 1, we can conclude that f takes on the same value for at least two members of G. This concludes our argument that there cannot be a numerically diverse group of Philadelphians. □

The above argument presupposes that there are finitely many Philedelphians. In fact, the theorem does not hold if we allow Philadelphia to have infinitely many people. As an exercise, try to describe a numerically diverse group of infinitely many Philadelphians.

The Pigeonhole Principle can take on a more general form, the *Mean Pigeonhole Principle*, which is as follows:

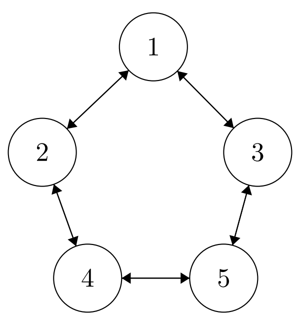
Principle 2. ***The Mean Pigeonhole Principle****: If you distribute m pigeons into n pigeonholes and $m \geq k \cdot n + 1$, then some hole contains at least $k+1$ pigeons.*

Note that Principle 1 is just the special case of Principle 2 for $k = 1$.

Example 2. *Say a group of people has three-mutuality if it contains either a group of three mutual friends or a group of three mutual strangers. How large a group of people can lack three-mutuality?*

We show that the largest such group has five members. To do this, we will give an example of a pattern of friendship among a group of five people that lacks three-mutuality, and show that every pattern of friendship among six or more people has three-mutuality. To show that every friendship pattern on six or more people lacks three-mutuality, we will use the Mean Pigeonhole Principle.

Proof. The diagram below shows a "friendship pentagon". Nodes represent people, and an edge between people represents friendship. It is easily checked that the diagram lacks 3-mutuality.



Next, we show that every group of $n \geq 6$ people must have 3-mutuality. Again, write p_{ij} to denote that i is a friend of j.

Let $G = \{1, \ldots, 6\}$ and sort the five people $2, \ldots, 6$ into two pigeonholes according to the truth value, true ($\top$) or false ($\bot$) of $p_{12}, \ldots, p_{16}$. That is, sort people $2, \ldots, 6$ into two groups, one group which are all friends of 1, and one group all of which are not friends with 1. By Principle 2, one of these holes, suppose it's the $\top$ one, contains at least three members of G.

Now, either two of these are friends, in which case they, together with 1 form a collection of three mutual friends, or none of them of friends, in which case they themselves form a collection of three mutual strangers. The argument is analogous, in the case that three members of G were sorted into the $\bot$ pigeonhole. □

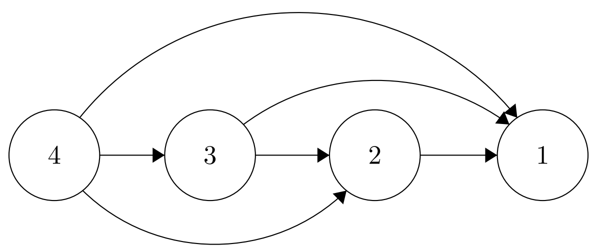
We might wonder whether every natural number n has a k such that every group of at least size k has n-mutuality. This happens to be true (try proving it!). The *Ramsey number* $R_{m,n}$ is the least k such that every set of k people must have either a group of m mutual friends or n mutual strangers. In the previous example, we shoed that $R_{3,3} = 6$. Higher ramsay numbers are much harder to compute. We know that

$R_{4,4} = 18$. $R_{5,5}$ is currently known to be between 43 and 48. $R_{6,6}$ is somewhere between 102 and 165.

As an exercise, prove that $R_{m,n} = R_{n,m}$ for all m, n.

"Suppose aliens invade the earth and threaten to obliterate it in a year's time unless human beings can find the Ramsey number for red five and blue five. We could marshal the world's best minds and fastest computers, and within a year we could probably calculate the value. If the aliens demanded the Ramsey number for red six and blue six, however, we would have no choice but to launch a preemptive attack." - Paul Erdos

Love differs from friendship in that there are narcissists (so we can't assume the relation is irreflexive) and is not always requited (so we can't assume the relationship is symmetric). This difference between friendship and love allows the existence of numerically diverse groups of lovers, that is, groups where each person in the group loves a different number of people in the group. Consider, for example, a group of four people, $\{1, 2, 3, 4\}$. Suppose that 1 doesn't love anyone, 2 loves 1, 3 loves both 1 and 2, and 4 loves all of $1, 2$, and 3, and that this exhausts all the love among our group of four. We achieve numerical diversity at the sacrifice of requital.



How many different patterns of love might obtain among a group of four people $\{1, 2, 3, 4\}$? Let's recycle the sentence letters and use p_{ij} to signify the statement that i loves j; note that 16 sentence letters would be required to record all the relevant statements. Since each pattern of love among $1, 2, 3, 4$ is determined by assigning one of the truth values $\top$ or $\bot$ to each of these 16 sentence letters, we can conclude that the number of such patterns is 2^{16}. Why? Because there are two assignments to p_{11} and for each of these, there are two assignments to p_{12}, and thus $2 \cdot 2 = 2^2$ assignments to them jointly (this observation is given the exalted title, "The Product Rule"). Thus, by iterating application of the product rule another fourteen times, we arrive at the conclusion that there are 2^{16} possible truth assignments to the 16 sentence letters.

$2^{16} = 65536$. It's kind of amazing that there are as many as 65,536 different potential love-scenarios at a table for four!

Friendship, as compared to love, is relatively tame in terms of the number of scenarios that might arise. Let's return to using p_{ij} to indicate that i and j are friends. In virtue of the fact that friendship is symmetric and irreflexive, a friendship-scenario is determined by assigning one of the truth values $\top$ or $\bot$ to each of the 6 sentence letters p_{ij}, for $1 \leq i < j \leq 4$. Hence, there are only $2^6 = 64$ possible patterns of friendship among the group of four, less than 1/1000 of the number of potential love-scenarios.

In general, how many possible friendship scenarios are there among a group of n people? Well, every pair can either be friends or not friends, so there are 2^{num_pairs} possibilities. How many pairs are their, in terms of n?

1.3 Review

Concept Review

- **Pigeonhole Principle**: If you have $n + 1$ pigeons and you try to fit them all into n holes, then there has to be at least one hole with $k > 1$ pigeons.
- **The Mean Pigeonhole Principle**: If you distribute m pigeons into n pigeonholes and $m \geq k \cdot n + 1$, then some hole contains at least $k + 1$ pigeons.
- **Product Rule**: If there are n ways to do a first action and m ways to do a second action, there are $n \cdot m$ ways to do both action 1 and action 2.

Problems

1. Let X be a set, $|X| = n$ (we write $|X|$ for the size of the set X). How many subsets does X have?
2. How many subsets of even size does a set X of size $n > 0$ have?
3. Prove that the Cartesian plane cannot be colored using only two colors (Red/Blue) such that all points 1 unit away from each other are different colors.
4. Prove that for any set of $n \geq 2$ numbers, there are 2 numbers whose difference is divisible by $n - 1$.
5. Show that for any $n \in \mathbb{N}$, there is a number k whose base ten numeral contains only "5"s and "0"s such that k is divisible by n.

Solutions

1. There are 2^n many subsets of a set of size n. To see why this is the case, note that every element of the set can be either in or not in any given subset. Hence there are two choices for each of the n elements of the set, and by the product rule 2^n choices in total.

2. 2^{n-1}. We show that for every X of size at least one, the number of even-size subsets of X is equal to the number of odd-size subsets of X; it then follows from the result of the preceding problem that the answer is 2^{n-1}.

   First, suppose that the size of X is odd. Then complementation induces a one-to-one correspondence between the odd-size and even-size subsets of X. That is, we associate to each odd-size subset $Y \subseteq X$, the even-size subset $X - Y$. If, on the other hand, the size $n > 1$ of X is even, we argue as follows. Let a be an element of X and consider the set $W = X - \{a\}$. Since the size of W is odd, we already know that it has the same number of subsets of even-size as it does of odd-size; that is, there are the same number of subsets of X of odd-size that exclude a as there are subsets of X of even-size that exclude a. From this it follows at once that also X has the same number of sets of even-size that include a as it does subsets of odd-size that include a. Thus, X has the same number of subsets of odd-size as it does subsets of even-size.

3. Consider an equilateral triangle with unit-length sides. We have three points pairwise one-unit apart and only two colors. The answer follows by application of the pigeonhole principle.

4. Note that there are $n - 1$ remainders when dividing by $n - 1$. Hence by the pigeonhole principle two of our n numbers must have the same remainder when divided by $n - 1$. Their difference is divisible by $n - 1$.

5. Consider the first $n + 1$ elements of the set $\{5, 55, 555...\}$. We know from above that this set has two numbers whose difference is divisible by n. Note that the difference of any two numbers in this set is written using only 5s and 0s.

2 Truth-Functional Logic

2.1 Introduction to Truth-Functional Logic

Throughout the course we will see a few different systems for formalizing statements. Each consists of a formal language to represent statements, and a way to interpret the meaning of statements in that language. Truth-functionalF logic is the simplest of these systems we will learn.

Components of Truth Functional Logic

1. Language (the *Syntax*)
   (a) sentence letters
   (b) connectives
2. Interpretation (the *Semantics*)
   (a) A function that assigns $\top$ or $\perp$ (true or false) to each sentence letter, called a **truth-assignment**
   (b) Fixed **truth-functional semantics** for each connective

Sentence letters such as $p, q, r, \ldots$ schematize statements (in natural language) which are true or false, and **connectives** such as $\wedge, \vee, \neg, \supset, \ldots$ are used to combine sentence letters into compound schemata.

Example 3. *We might say that p represents the statement "it is a Wednesday". This would be reasonable, since it's definitely either true or false that today is Wednesday. On the other hand, "is today Wednesday?" isn't a statement, so we wouldn't encode it as a sentence letter. Truth-functional logic deals with the truth or falsity of* statements *only.*

2.2 Basic Syntax of Truth-Functional Logic

Consider using the sentence letter p_{ij} to schematize the statement "i loves j," where $1 \leq i, j, \leq 4$. For example, p_{11} schematizes the statement "1 loves 1", or briefly, "1 is a narcissist."

Suppose we wish to schematize the following statements using those sentence letters:

1. all of 1, 2, 3, and 4 are narcissists;
2. none of 1, 2, 3, and 4 are narcissists;
3. at least one of 1, 2, 3, and 4 is a narcissist;
4. an odd number of 1, 2, 3, and 4 are narcissists.

In order to do so, we introduce the following truth-functional connectives. For each connective, we display its truth-functional interpretation via a table indicating the truth value of the compound schema as a function of the truth values of its components.

- Conjunction (and):

p	q	$(p \wedge q)$
$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$
$\bot$	$\bot$	$\bot$

- Negation (not):

p	$\neg p$
$\top$	$\bot$
$\bot$	$\top$

- Inclusive Disjunction (or)

p	q	$(p \vee q)$
$\top$	$\top$	$\top$
$\top$	$\bot$	$\top$
$\bot$	$\top$	$\top$
$\bot$	$\bot$	$\bot$

- Exclusive Disjunction (exclusive or, xor)

p	q	$(p \oplus q)$
$\top$	$\top$	$\bot$
$\top$	$\bot$	$\top$
$\bot$	$\top$	$\top$
$\bot$	$\bot$	$\bot$

- Material Conditional

p	q	$(p \supset q)$
$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\top$
$\bot$	$\bot$	$\top$

- Material Biconditional

p	q	$(p \equiv q)$
$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$
$\bot$	$\bot$	$\top$

Note that the truth/falsity of a compound schema is completely determined by, or purely a function of, the truth/falsity of its components. Hence, the term "truth-functional logic."

We can now schematize conditions 1 – 4 in the above example as follows.

S1: $((p_{11} \wedge p_{22}) \wedge p_{33}) \wedge p_{44}$

S2: $((\neg p_{11} \wedge \neg p_{22}) \wedge \neg p_{33}) \wedge \neg p_{44}$

S3: $((p_{11} \vee p_{22}) \vee p_{33}) \vee p_{44}$

S4: $((p_{11} \oplus p_{22}) \oplus p_{33}) \oplus p_{44}$

The first three are quite straightforward to verify; the fourth we will prove later in Proposition 1.

2.3 Basic Semantics of Truth-Functional Logic

Given a truth-functional schema like $((p \wedge q) \vee r)$, we cannot determine whether the schema is true or false unless we know whether p, q, and r are true or false. That is, any schema requires a truth-assignment to its sentence letters before it can be evaluated.

Definition 1 (Truth-assignment)**.** *Let X be a set of sentence letters. A truth-assignment A for X is a mapping which associates with each sentence letter $q \in X$ one of the two truth values $\top$ or $\bot$; we write $A(q)$ for the value that A associates to q.*

Suppose S is a truth-functional schema such that every sentence letter with an occurrence in S is a member of X. We say a truth assignment A for X satisfies such a schema S ($A \models S$) if and only if S receives the value $\top$ relative to the truth assignment A.

Example 4. *Take the schema $S = ((p \wedge q) \vee r)$, with truth assignment A such that $A(p) = \top$, $A(q) = \bot$, and $A(r) = \bot$, we have that S receives the value $\bot$. In other words A does not satisfy S. ($A \not\models S$).*

Interpreting the Material Conditional

Let's return to our potential lovers and restrict attention to just two of them, 1 and 2. How could express the statement that all love is requited among these two sweethearts? The natural mode of expression is: if 1 loves 2, then 2 loves 1, and if 2 loves 1, then 1 loves 2. This is a perfect candidate for using the material conditional.

Using the sentence letters $p_{11}, p_{12}, p_{21}, p_{22}$ as earlier interpreted, we can express the happy state that all love among 1 and 2 is requited by the schema

$$R : (p_{12} \supset p_{21}) \wedge (p_{21} \supset p_{12})$$

or, equivalently,

$$p_{12} \equiv p_{21}$$

In how many of the possible love scenarios among 1 and 2 is all love requited? Count the number of satisfying truth-assignments to R!

While the motivations for the truth-functional definitions for the other connectives normally seem evident to new logicians, the material conditional often gives people trouble. Let's consider generalized conditionals as a route to motivating the truth-functional interpretation of the conditional offered above. Of course, the statement "if an integer is divisible by six, then it is divisible by three," is true. Consider each of the following statements, which (intuitively) seem true:

- "If twelve is divisible by six, then twelve is divisible by three."
- "If three is divisible by six, then three is divisible by three."
- "If two is divisible by six, then two is divisible by three."

Note that the preceding sentences are in the form $\top \supset \top$, $\bot \supset \top$, and $\bot \supset \bot$ respectively.

Therefore, if the conditional involved is to be understood truth-functionally, then its interpretation must satisfy the conditions imposed by the first, third, and fourth rows of the material conditional's truth-table. On the other hand, the falsity of the conditional "if twelve is divisible by six, then twelve is divisible by seven," mandates the condition imposed by the second row of the truth-table.

An Inductive Proof

Let's do a simple inductive proof about truth-functional satisfaction, both to get your brain thinking inductively and to give an example of how induction works whgen dealing with logical sentences.

Proposition 1. *For every $n \geq 2$ and every set $X = \{q_1, \ldots, q_n\}$ of n distinct sentence letters, a truth assignment A for X satisfies the schema*

$$S_n : (\ldots(q_1 \oplus q_2) \ldots \oplus q_n)$$

if and only if A assigns an odd number of the sentence letters in X the value $\top$.

Proof. We prove the proposition by induction on n.

- Basis: Examination of the truth table for $\oplus$ suffices to establish the proposition for the case $n = 2$.
- Induction Step: Suppose the proposition holds for a number $k \geq 2$, that is, for every truth assignment A for $\{q_1, \ldots, q_k\}$, $A \models S_k$ if and only if A assigns an odd number of the sentence letters in $\{q_1, \ldots, q_k\}$ the value $\top$; this is our induction hypothesis. We proceed to show that the proposition also holds for $k + 1$. Let A' be an assignment to the sentence letters $\{q_1, \ldots, q_{k+1}\}$ and let A be its restriction to $\{q_1, \ldots, q_k\}$. We consider two cases. First, suppose that $A'(q_{k+1}) = \top$. In this case, $A' \models S_{k+1}$ if and only if $A \not\models S_k$ if and only if (by our induction hypothesis) A assigns an even number of the sentence letters $\{q_1, \ldots, q_k\}$ the value $\top$. Hence, if $A'(q_{k+1}) = \top$, then $A' \models S_{k+1}$ if and only if A' assigns an odd number of the sentence letters in $\{q_1, \ldots, q_{k+1}\}$ the value $\top$. On the other hand, suppose that $A'(q_{k+1}) = \bot$. In this case, $A' \models S_{k+1}$ if and only if $A \models S_k$ if and only if (by our induction hypothesis) A assigns an odd number of the sentence letters $\{q_1, \ldots, q_k\}$ the value $\top$. Hence, if $A'(q_{k+1}) = \bot$, then $A' \models S_{k+1}$ if and only if A' assigns an odd number of the sentence letters in $\{q_1, \ldots, q_{k+1}\}$ the value $\top$. This concludes the proof, since either $A'(q_{k+1}) = \top$ or $A'(q_{k+1}) = \bot$.

$\square$

The Centrality of Satisfaction

The satisfaction relation is the fundamental semantic relation. It is where language and the world meet; in the case to hand, language consists of truth-functional schemata and the possible worlds they describe are truth assignments to sentence letters. As the course progresses, we will encounter more textured representations of the world (relational structures) and richer languages to describe them (monadic and polyadic quantification theory). We now define some of the central notions of truth-functional logic in terms of satisfaction. These definitions will generalize directly to the more textured structures and richer languages we encounter later.

Definition 2. *For the following definitions, we suppose that S and T are truth-functional schemata and that A ranges over truth assignments to sets of sentence letters which include all those that occur in either S or T.*

- *S implies T if and only if for every truth assignment A, if $A \models S$, then $A \models T$.*
- *S is equivalent to T if and only if S implies T and T implies S.*
- *S is satisfiable if and only if for some A, $A \models S$.*
- *S is valid if and only if every truth assignment satisfies S.*

Examples of equivalence and the material biconditional

Try to see why the following are equivalent - either by appeal to your understanding of what the connective "means" or by going back to the truth tables.

- $p \oplus q$ is equivalent to $q \oplus p$ (commutativity of exclusive disjunction)
- $(p \oplus q) \oplus r$ is equivalent to $p \oplus (q \oplus r)$ (associativity of exclusive disjunction).
- $p \equiv q$ ius equivalent to $(p \supset q) \wedge (q \supset p)$

Note that both conjunction and inclusive disjunction are also commutative and associative, whereas the material conditional is neither.

Try to to think of examples of (binary) truth-functional connectives which are commutative but not associative, and associative but not commutative.

2.4 Review

Concept Review

Definitions

- A truth-assignment A for X is a function which maps every sentence letter $q \in X$ to either $\top$ or $\bot$. $A(q)$ is the notation for the value A associates with q.
- A schema S *implies* a schema T iff for all truth-assignments A, if $A \models S$ then $A \models T$.
- A schema S is *equivalent* to a schema T iff S and T are satisfied by exactly the same truth assignments (for all A, $A \models S$ iff $A \models T$).
- S is *satisfiable* iff there is a truth assignment that satisfies it (there exists an A such that $A \models S$)
- S is *valid* iff all truth assignments satisfy it (for all A, $A \models S$)

Syntax, Semantics The *syntax* of a logic is its solely syntactic representation, ie the sentence letters and connectives which form (purely formal) statements. The *semantics* of TF-logic are given by a *truth-assignment*, which associates with each letter a *truth-value*.

Satisfying Sentences The *truth-values* of the individual sentence letters in a chema are propagated to the whole schema by means of *truth-tables* which give fixed semantic interpretations to each of the *connectives*. We say that a truth-assignment A *satisfies* a sentence S (written $A \models S$) iff the sentence S evaluates to $\top$ under the truth-assignment A. Otherwise, we write $A \not\models S$ and say that A does not satisfy S.

Problems

1. Is "the University of Pennsylvania has a Logic major" a statement? If not, say why not.
2. Is "should I major in Logic?" a statement? Why or why not?
3. Using the sentence letters $p_i j$, $q \leq i, j \leq 4$ to stand for "person i loves person j". Schematize the following statements:
   (a) Person 1 loves everyone else.
   (b) There is a Shakespearean love triangle (ie no-one has their love requited) between people 1, 2, 3, and person 4 is a Scrooge (he does not love anyone, even himself).
   (c) Everyone loves, exclusively, people with numbers lower than themselves. Reverse capitalism!
4. How many truth-assignments to the given letters satisfy the following schema?
$$(p_1 \supset q_1) \wedge ... \wedge (p_5 \supset q_5)$$
5. How many truth-assignments, in terms of n, satisfy the following schema? Prove it by induction.
$$(p_1 \supset q_1) \wedge ... \wedge (p_n \supset q_n)$$
6. How many truth-assignments over the given letters satisfy satisfy the following schema?
$$p_1 \oplus p_2 \oplus p_3 \oplus p_4 \oplus p_5$$
7. Is the following sentence valid, satisfiable but not valid, or unsatisfiable?
$$(a \equiv b) \supset (a \vee \neg b)$$
8. Valid, satisfiable, or unsatisfiable?
$$(b \vee (b \supset a)) \wedge (\neg b \vee (a \supset b))$$
9. Valid, satisfiable, or unsatisfiable?
$$(a \equiv b) \wedge (b \equiv c) \wedge (a \oplus b)$$
10. How many truth-assignments for the given letters satisfy
$$(a \equiv b) \wedge (b \equiv c) \wedge (c \equiv d)$$
11. How many truth-assignments to the given letters satisfy
$$(a \oplus b) \vee (b \oplus c) \vee (c \oplus d)$$
12. I claim that if n people all shake hands with each other (once per pair), the total number of handshakes is $\frac{n(n-1)}{2}$. Prove this by induction.

Solutions

1. Yes, it is.

2. No, it is not. It is not a statement because it expresses a question, which is not determinitely true or false.

   With that being said, you should - of course - major in logic[a].

3. (a) $p_{11} \wedge p_{12} \wedge p_{13} \wedge p_{14}$

   (b) $((p_{12} \wedge p_{23} \wedge p_{31}) \vee (p_{13} \wedge p_{21} \wedge p_{32})) \wedge \neg(p_{41} \vee p_{42} \vee p_{43} \vee p_{44})$

   (c) $p_{21} \wedge p_{31} \wedge p_{32} \wedge p_{41} \wedge p_{42} \wedge p_{43}$

4. 3^5. Note that each of the terms of the form $p_i \supset q_i$ is satisfied in three cases (check the truth table) and apply the product rule.

5. 3^n for $n \geq 1$.

   BASE: checking the truth table verifies that there are $3^1 = 3$ satisfying assigmnents for the $n = 1$ case.

   INDUCT: suppose there are 3^n satisfying truth assignments to $S_n := (p_1 \supset q_1) \wedge ... \wedge (p_n \supset q_n)$. We want to show that there are 3^{n+1} satisfying assignments to $S_{n+1} := S_n \oplus (p_{n+1} \supset q_{n+1})$. Note that there are 3 satisfying assignments to S and s_{n+1} which satisfy $S \oplus s_{n+1}$. By hypothesis there are 3^n truth assignments satisfying $(p_1 \supset q_1) \wedge ... \wedge (p_n \supset q_n) = S_n$. By the product rule then, there are $3^n \cdot 3 = 3^{n+1}$ total satisfgying assigments to $S_n \oplus s_{n+1}$ as required.

6. $2^4 = 16$. Remember that there are 2^{n-1} ways to pick an odd-sized subset from n elements and that a sentence of the given form is satisfied iff an odd number of sentence letters are set to true.

7. This is valid. Note that if $a \equiv b$, $a \vee \neg b$ holds (since one of a or $\neg b$ must be true, hence the consequent is true, hence the conditional is true). If a is not equivalent to b, then the conditional holds because false implies anything.

8. Valid. If b is true, then the left conjunct is clearly true. If b is not true, then $b \supset a$ is true (false implies anything), hence the left conjunct is true as well. Similarly, if b is true then the right conjunct is true (since both falsity and truth imply truth), and if b is false then $\neg b$ is true, hence the right conjunct is true again.

9. Unsatisfiable. Since a is equivalent to b and b is equivalent to c, a is equivalent to c (in other words, $\equiv$ is *transitive*). But a can't be equivalent to c, because $a \oplus c$. So the sentence is unsatisfiable.

10. 2. Picking true/false for a fixes the truth-values of the remaining letters.

11. 14. To get this answer, we note that there are 16 (2^4) truth assignments in total, count the number which do not satisfy our sentence, and subtract that number from 16. The sentence is only not satisfied when each of a, b, c, d have the same truth-value, so there are 2 non-satisfying truth-assignments. This means there are $16 - 2 = 14$ satisfying truth assignments.

12. BASE CASE: $n = 2$. Two people shaking hands results in one handshake, and the formula gives us $\frac{2(2-1)}{2} = 1$ which is correct. Note that I pick $n = 2$ as the base case (not $n = 0$ or $n = 1$) because it doesn't really make sense to talk about those cases (since you need two people for a handshake).

    INDUCTIVE CASE: Assume that for n people, the number of handshakes (let's denote it H_n) is $H_n = \frac{n(n-1)}{2}$. We want to show (henceforth "wts") that for $n + 1$ people the number of handshakes is $H_{n+1} = \frac{(n+1)n}{2}$. The number of handshakes between $n+1$ people is clearly the number of handshakes for n people (H_n) plus n, since our new person must shake hands with the n others. So we have $H_{n+1} = H_n + n = \frac{n(n-1)}{2} + n = \frac{n^2-n+2n}{2} = \frac{n^2+n}{2} = \frac{(n+1)n}{2}$, which is what we wanted to show.

[a] Provided you like it and want to.

2.5 Expressive Completeness of Truth Functional Logic

Definition 3. *We use the symbol* $:=$ *to mean "is defined to be equal to".* $:=$ *expresses a* definition *of equality, whereas* $=$ *expresses a statement about equality.*

If you're a coder, $x := 10$ *is to logic/math as* `let x = 10` *is to JavaScript, whereas* $x = 10$ *is to logic as* `x === 10` *is to JavaScript.*

Propositions as a heuristic

It is sometimes useful to think of a schema S as expressing a proposition, to whit, the set of truth assignments A that satisfy S; of course, this needs to be relativized to a collection of sentence letters X which includes all those occurring in S. We suggested the notation:

$$\mathbb{P}_X(S) = \{A \mid A \text{ is a truth assignment for } X \text{ and } A \models S\}.$$

When we use this notation without the subscript X, we assume A is a truth assignment for exactly the set of sentence letters with occurrences in S.

Expressive Completeness

In the last section, we suggested using the notion of the proposition expressed by a schema as an intuitive vehicle for pursuing this investigation. Since the semantical correlate of a truth-functional schema is a set of truth assignments to some finite set of sentence letters, we can frame the question of the *expressive completeness of truth-functional logic* in terms of propositions. Let X be a non-empty finite set of sentence letters. We deploy the notation: $\mathbb{A}(X)$ for the set of truth assignments to the sentence letters X, and $\mathbb{S}(X)$ for the set of truth-functional schemata compounded from sentence letters all of which are members of X.

We provide the following inductive definition of $\mathbb{S}(X)$.

Definition 4. *Let* X *be a nonempty finite set of sentence letters.* $\mathbb{S}(X)$ *is the smallest set* $\mathbb{U}$ *(in the sense of the subset relation) satisfying the following conditions.*

- $X \subseteq \mathbb{U}$.
- *If* σ *and* τ *are strings over the finite alphabet* $X \cup \{), (, \neg, \supset, \equiv, \vee, \wedge, \oplus\}$*, and* $\sigma, \tau \in \mathbb{U}$*, then each of the strings* $\neg\sigma, (\sigma \supset \tau), (\sigma \equiv \tau), (\sigma \vee \tau), (\sigma \wedge \tau), (\sigma \oplus \tau)$ *belong to* $\mathbb{U}$.[1]

This is simply a formal way of saying that all of our sentences have to use only the letters from X and must be "well-built" in the sense that each connective has the correct number of arguments, with all the bracketing done correctly. For example, with $X := \{p, q, r\}$ then $S_1 := ((p \vee q) \wedge r)$ is well-built, whereas $S_2 := \vee p \wedge q$ is not.

If $\mathfrak{P} \subseteq \mathbb{A}(X)$, we call $\mathfrak{P}$ a *proposition over* X.

Let X be a non-empty finite set of sentence letters and let $\mathfrak{P}$ be a proposition over X. Is there a schema $S \in \mathbb{S}(X)$ such that $\mathbb{P}_X(S) = \mathfrak{P}$, ie can every proposition be expressed by some schema? In other words, is truth-functional logic *expressively complete*?

Theorem 1 (Expressive Completeness of Truth-functional Logic)**.** *Let* X *be a non-empty finite set of sentence letters and let* $\mathfrak{P}$ *be a proposition over* X*. There is a schema* $S \in \mathbb{S}(X)$ *such that* $\mathbb{P}_X(S) = \mathfrak{P}$.

[1] Here "$(\sigma \supset \tau)$" denotes the string with the initial symbol "(" concatenated with the string denoted by σ concatenated with the symbol "$\supset$" concatenated with the string denoted by τ and with terminal symbol ")", and likewise in all the other cases.

This looks complicated, but it really isn't. In natural language, what it's saying is this: *pick any subset* $\mathfrak{P}$ (your proposition) *of truth assignments for a set of sentence letters X. Then there is a truth-functional schema S using only letters from X ($S \in \mathbb{S}(X)$) which is true only of those truth-assignments which are in $\mathfrak{P}$ ($\mathbb{P}_X(S) = \mathfrak{P}$).* In other words, *every proposion can be "picked out" by some schema.* This is why it's called *expressive completeness*: truth-functional logic is "expressively complete" in that it can express every such proposition.

For the proof of Theorem 1, the following terminology and lemma will be useful.

Definition 5. *Let X be a non-empty finite set of sentence letters and let $S \in \mathbb{S}_X$.*

- *S is a* literal *over X just in case $S = p$ or $S = \neg p$, for some $p \in X$.*
- *S is a* term *over X just in case S is a conjunction of literals over X (we allow conjunctions of length 1).*
- *S is in disjunctive normal form over X if and only if S is a disjunction of terms over X (we allow disjunctions of length 1).*

If Λ is a set of literals over X we write $\bigwedge \Lambda$ to abbreviate a term which is formed as a conjunction of the literals in Λ. Similarly, if Γ is a set of terms over X we write $\bigvee \Gamma$ to abbreviate a schema in disjunctive normal form which is formed as a disjunction of the terms in Γ.

Example 5. *Let $S = \{a, b, c\}$. Then $\bigwedge S = a \wedge b \wedge c$, and $\bigvee S = a \vee b \vee c$.*

Lemma 1. *Let X be a non-empty finite set of sentence letters. For every $A \in \mathbb{A}(X)$ there is a schema T_A which is a term over X such that for every $A' \in \mathbb{A}(X)$*

$$A' \models T_A \quad \textit{if and only if} \quad A' = A.$$

Proof. Let X be a finite set of sentence letters and suppose $A \in \mathbb{A}(X)$. For each $p \in X$, let $l_p = p$, if $A \models p$, and let $l_p = \neg p$, if $A \not\models p$. Let $\Lambda = \{l_p \mid p \in X\}$ and let $T_A = \bigwedge \Lambda$. It is easy to verify that for every $A' \in \mathbb{A}(X)$, $A' \models T_A$ if and only if $A' = A$. □

Once you become a bit more familiar with the terminology, things will become much easier. Indeed, this lemma is really simple - in plain English, it says that *for every truth assignment, there is a schema which only uses logical ANDs that is satisfied by only that truth assignment.* When stated like that, of course, it seems obvious - if your truth assignment assigns true to p you should have p in your schema, and if your truth-assignment assigns false to p, your schema should include $\neg p$, with all the literals joined up together by ANDs.

The proof expresses that intuition symbolically - make sure you can understand the proof now, going over the relevant terminology and symbols if necessary. If you get stuck trying to interpret all that logical symbolism, please come into Office Hours and we'll be happy to help! Logic won't be any fun if the symbolism gets in the way of your ingenuity of understanding, so it's best if you take the time to get comfortable with all that at the start.

Proof of Theorem 1. Fix a finite non-empty set of sentence letters X and suppose $\mathfrak{P}$ is a proposition over X. If $\mathfrak{P} = \emptyset$, then pick $p \in X$ and note that $\mathbb{P}_X(p \wedge \neg p) = \mathfrak{P}$. Otherwise, for each $A \in \mathfrak{P}$, choose a term T_A, as guaranteed to exist by Lemma 1, such that for every $A' \in \mathbb{A}(X)$, $A' \models T_A$ if and only if $A' = A$. Let $\Gamma = \{T_A \mid A \in \mathfrak{P}\}$ and let $S = \bigvee \Gamma$. It is easy to verify that $\mathbb{P}_X(S) = \mathfrak{P}$. □

Once again, the main difficulty here is the symbolism - the proof expresses a simple intuition in symbolic form. Try rewriting this proof in your own words!

Corollary 1. *Every truth-functional schema is equivalent to a schema in disjunctive normal form.*

Corolloraries are Theorems which follow very simply or quickly from another (often proved-right-above) theorem. Whenever you see a corollary in a math textbook or notes, you should always make sure you understand why it's a consequence of the just-proven theorem!

2.6 The Power of a Truth-Functional Schema

We will introduce the following useful terminology.

Definition 6. *For the following, all schemata are drawn from $\mathbb{S}(X)$ for a fixed non-empty finite set of sentence letters X.*

- *A list of truth-functional schemata is succinct if and only if no two schemata on the list are equivalent.*
- *A truth-functional schema implies a list of schemata if and only if it implies every schema on the list.*
- *The power of a truth-functional schema is the length of a longest succinct list of schemata it implies.*

Example 6. *Let's consider $X = \{p, q, r\}$. What is the length of a longest succinct list of truth-functional schemata over X? We will arrive at the answer by proving an* upper bound *and a* lower bound *on this length.*

- *Upper bound: It is easy to verify that schemata S and S' are equivalent if and only if $\mathbb{P}(S) = \mathbb{P}(S')$. Hence, the length of a succinct list of schemata cannot exceed the number of propositions over X, that is, the number of subsets of the set $\mathbb{A}(X)$. The size of X is 3, so the size of $\mathbb{A}(X)$ is 2^3, since determining a truth assignment to X involves three binary choices (each letter can be assigned true or false, and you make that choice for each of the three letters). By the same reasoning, the number of propositions over X is 2^{2^3}, since determining a proposition involves deciding, for each of the 2^3 truth assignments, whether to include or omit it. Hence, the length of the longest succinct list is no more than $2^{2^3} = 2^8 = 256$.*
- *Lower bound: By Theorem 1, for every proposition over X, there is a schema expressing it. Since schemata expressing distinct propositions are not equivalent, it follows at once that there is a succinct list of schemata of length 256.*

So the longest such list is of length 256.

Example 7. *Let's compute the power, as defined above, of $p \wedge (q \vee r)$. Note that a schema S implies a schema S' if and only if $\mathbb{P}(S) \subseteq \mathbb{P}(S')$. Thus, the power of S is the number of sets Z satisfying the condition:*

$$\mathbb{P}(S) \subseteq Z \subseteq \mathbb{A}(X). \tag{1}$$

The size of $\mathfrak{P} = \mathbb{P}(p \wedge (q \vee r))$ is 3, so the size of $\mathbb{A}(X) - \mathfrak{P} = 5$. It follows at once that $2^5 = 32$ sets Z satisfy condition (1); hence, the power of $p \wedge (q \vee r)$ is 32.

Why is it that a schema S implies a schema S' if and only if $\mathbb{P}(S) \subseteq \mathbb{P}(S')$? Go back to the definition of both $\mathbb{P}(S)$ and schema implication if you need to.

Here's a simple way to think about it, once you know the definitions: if S implies S' then there is no truth assignment that satisfies S but does not satisfy S' (otherwise S wouldn't imply S', by the definition of schema implication). Hence everything that is true in S is true in S', or symbolically, $\mathbb{P}(S) \subseteq \mathbb{P}(S')$. When written that way, it seems really simple!

Often throughout your study of logic you will see things which, at the surface, look confusing like the statement we just considered. Make sure to always take the time to go back to definitions and understand things in your own words - it'll make logic *much* more satisfying.

Example 8. *Let's list the numbers which are powers of truth-functional schemata over $X = \{p, q, r\}$.*

- *First note that for every $S, S' \in \mathbb{S}(X)$ the power of S = the power of S' if and only if $|\mathbb{P}_X(S)| = |\mathbb{P}_X(S')|$, where we use $|U|$ to denote the number of members of the finite set U.*

- *In particular, if* $\mathfrak{P} = \mathbb{P}_X(S)$, *then the power of* $S = 2^{(8-|\mathfrak{P}|)}$.
- *It follows at once that for each* $S \in \mathbb{S}(X)$, *the power of* $S = 2^i$, *for some* $0 \leq i \leq 8$.

Generalizing our last example, suppose Y is a finite set of sentence letters with $|Y| = n$. In this case

- $|\mathbb{A}(Y)| = 2^n$, and
- for each $S \in \mathbb{S}(Y)$, if $\mathfrak{P} = \mathbb{P}_Y(S)$, then the power of $S = 2^{(2^n - |\mathfrak{P}|)}$.

Example 9. *What is the length of a longest succinct list of truth-functional schemata over* $X := p, q, r$ *each of which has power 32?*

Make sure you have all the relevant definitions in order - what does it mean for the power of a schema to be 32? What does it mean for a list of schemata to be succinct?

Well, from the definitions we know that a schema over $X := \{p, q, r\}$ *has power 32 if and only if exactly three truth assignments satisfy it (why?). Hence the length of a longest such succinct list is exactly the number of subsets of size three contained in a set of size eight (why a set of size 8, given that we have 3 sentence letters?). In the next section, we'll take a break from logic proper to learn a bit about how we would determine how many such subsets there are.*

A Combinatorial Interlude

Our leading question from the end of the last section brings us to an interlude on permutations and combinations: how many ways can we select k members of a set of size n? There is an ambiguity here: are we counting modes of selection, which involve the order of choices, or collections of members selected, where the order of selection is irrelevant? Once we recognize the ambiguity, we can proceed to count both. We will need notation for each, so let $(n)_k$ for the number of ordered sequences of k distinct elements that can be drawn from a set of size n and $\binom{n}{k}$ for the number of subsets of size k that are included in set of size n.

Let's first evaluate $(n)_k$, the number of ordered sequences of size k you can pick from a set of size n. Suppose we think of counting the ways n students could fill a row of length k in a lecture hall. Let's suppose the seats are labelled $1, 2, \ldots, k$. There are n choices for the student to fill seat 1; once that seat is filled, there are $n-1$ choices for the student to fill seat 2; and so on until there are $(n-k)+1$ choices for the student to fill seat k. Hence, by the product rule, there are $n \cdot (n-1) \cdots ((n-k)+1)$ ways of filling all k seats, that is, $(n)_k = n \cdot (n-1) \cdots ((n-k)+1)$.

Now that we have counted the number of ordered sequences, we can see how to count the number of subsets. By the same reasoning, each subset of size k appears as the content of $k \cdot (k-1) \cdots 2 \cdot 1$ ordered sequences of length k; this number is called k *factorial* and is often abbreviated as $k!$. Hence,

$$\binom{n}{k} = \frac{(n)_k}{k!}.$$

Observe that

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

from which it follows that

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

This last formulation makes transparent a symmetry in the values of $\binom{n}{k}$, namely, for every k between 0 and n, $\binom{n}{k} = \binom{n}{n-k}$. This accords nicely with the observation that complementation induces a one-one correspondence between the subsets of size k and the subsets of size $(n-k)$ that can be selected from a set of size n. Note also that it determines in a non-arbitrary way that the value of $0!$ is 1.

Consider picking a panel of three students from a class of 10. How many ways can you do this? Is it the same as the number of ways you could pick 7 of the 10 students to *not* be on the panel, using the non-picked students for the panel?

Let's not forget how this all began. Since the The length of the longest succinct list of schemata with power 32 is number of subsets of size three contained in a set of size eight, it follows that the length of the longest such list is $\binom{8}{3} = 56$.

Counting the Length of an "Implicational Anti-Chain"

Let's use our newfound ability to count selections to answer a different question: Is there a sequence of seventy schemata $S_1, \ldots, S_{70} \in \mathbb{S}(X)$ such that for every $1 \leq i \neq j \leq 70$, S_i does *not* imply S_j? Such a sequence of schemata is called an *implicational anti-chain* (of length 70).

As observed earlier, a schema $S \in \mathbb{S}(X)$ implies a schema $T \in \mathbb{S}(X)$ if and only if $\mathbb{P}_X(S) \subseteq \mathbb{P}_X(T)$. It follows that the answer to our question about an implicational anti-chain of length seventy will be the same as the answer to the following question about an anti-chain of length seventy with respect to the subset relation: Is there a list of seventy subsets of $\mathbb{A}(X)$, $P_1, \ldots, P_n$, such that for every $1 \leq i \neq j \leq 70$, P_i is *not* a subset of P_j? Note that if two finite sets, P and Q, have the same number of members, and P is not equal to Q, then P is not a subset of Q and Q is not a subset of P. Therefore, if there are seventy distinct subsets of $\mathbb{A}(X)$ all of the same size, then the answer to our question is yes. Since $\mathbb{A}(X)$ has eight members, a positive answer to our question followed immediately by evaluating

$$\binom{8}{4} = \frac{8 \cdot 7 \cdot 6 \cdot 5}{4 \cdot 3 \cdot 2 \cdot 1} = 70.$$

Prove that if two finite sets, P and Q, have the same number of members, and P is not equal to Q, then P is not a subset of Q and Q is not a subset of P.

Note that our argument merely shows that there is an implicational anti-chain of length 70; it does not establish that there is no longer implicational anti-chain consisting of schemata in $\mathbb{S}(X)$. This is, indeed, true, but a more sophisticated argument is required to establish this result, which follows from the famous Sperner's Theorem. For a reference on Sperner's Theorem, see[2].

[2] Van Lint and Wilson, *A course in combinatorics*, Chapter 6: Dilworth's theorem and extremal set theory.

2.7 Is There An Efficient Decision Procedure For Truth-Functional Logic?

It is easy to see that the finitary character of the semantics for truth-functional logic immediately yields an algorithm to decide the satisfiability of schemata of truth-functional logic. In particular, suppose $S \in \mathbb{S}(X)$ for some finite set of sentence letters X. Note first that for each truth-assignment $A \in \mathbb{A}(X)$ there is a simple and efficient algorithm, call it M, to determine whether $A \models S$. Thus, in order to test the satisfiability of S, we need only list $\mathbb{A}(X)$ in some canonical order $A_1, \ldots, A_{2^{|X|}}$ and use M to test whether the successive A_i satisfy S.

Come up with an algorithm for checking whether $A \models S$ for $A \in \mathbb{A}(X)$ and analyze its runtime complexity as a function of the length (in terms of the number of connectives) of S.

Of course, this algorithm is not efficient, in the sense that it's running time is potentially exponential in the length of its input. The question whether there is an efficient algorithm to decide the satisfiability of truth-functional schemata, called the $P = NP$ problem, is generally regarded as one of the most significant open mathematical problems of our time, and carries with it a $1,000,000 prize for its solution as well as eternal mathematical glory. For further information visit:

`http://www.claymath.org/millennium-problems/p-vs-np-problem`.

2.8 Review

Concept Review

Definitions

- A schema S *implies* a schema T iff for all truth-assignments A, if $A \models S$ then $A \models T$. In other words, S implies T iff the proposition expressed by S is a subset of the proposition expressed by T.
- A schema S is *equivalent* to a schema T iff S and T are satisfied by exactly the same truth assignments (for all A, $A \models S$ iff $A \models T$). In other words, S and T are equivalent if they express the same proposition.
- $\mathbb{P}_X(S) = \{A | A \text{ is a truth assignment over } X \text{ and } A \models S\}$ is *the proposition expressed by S*. It's the set of truth assignments that satisfy S (where truth assignments are restricted to those for sentence letters in the set X).
- $\mathbb{A}(X)$ is the set of all truth assignments over X.
- A list of TF-schemata is called *succinct* iff no two schema on the list are equivalent.
- The *power* of a schema S is the length of the longest succint list of schemata which S implies. In other words, it's the number of nonequivalent schema which S implies.

Fun With Counting There are $n!$ ways to order a list of n items. To see why, note that there are n choices for the first element, $n-1$ for the second, $n-2$ for the third, resulting in $n(n-1)(n-2)...(1) = n!$ orderings.

There are $(n)_k := \frac{n!}{(n-k)!}$ ways to pick an ordered list of k elements from n elements, $k \leq n$. As before, there are n choices for the first thing, $n-1$ for the second, all the way down to $n-k+1$ for the k^{th}. This gives us the answer $\prod_{i=n-k+1}^{n} i = \prod_{i=1}^{n} i / \prod_{i=1}^{n-k} i = \frac{n!}{(n-k)!}$

There are $\binom{n}{k} := \frac{n!}{(n-k)!k!}$ ways to pick a subset of k elements from n elements, $k \leq n$. There are $(n)_k$ ordered lists of size k from n. Since each subset of size k corresponds to $k!$ orderings of those lists, we divide out by $k!$ to get $\frac{n!}{(n-k)!k!}$, which we denoted as $\binom{n}{k}$, read "n choose k".

Expressive Completeness

For any (arbitrary) proposition, there is a truth-functional schema which expresses that proposition. We noted that a schema can pick out individual truth-assignments by conjoining literals for each of the sentence letters (for example, the truth assignment A_1 which maps $p = \top, q = \top, r = \top$ is picked out by the sentence $(p \wedge q \wedge r)$). Sentences of this form are called *terms*. We further noted that a disjunction of such terms (one for each truth-assignment in our proposition) was sufficient to express any proposition.

Power

Suppose we have a sentence S over n sentence letters which is satisfied by k truth assignments. Then power of S is 2^{2^n-k}. To see why this is the case, note that there are 2^n truth assignments for n sentence letters. If S is satisfied by k truth assignments, then those truth assignments must also satify T if S implies T. So we can't "choose" to include those k truth-assignments in our proposition expressed by T any more, because $\mathbb{P}_X(T)$ must include them. So we are left with $2^n - k$ truth-assignments which can be in T or not. Since each of these $2^n - k$ truth assignments can be either in or out of the proposition expressed by T, the power of S is then 2^{2^n-k}.

Problems

For the following problems, unless otherwise specified, let $X = \{p_1 p_, p_3, p_4\}$ be the set of sentence letters under consideration.

1. What is the power of $p_1 \equiv p_2$?
2. For four sentence letters as above, what is the length of the longest succinct list of schemata no two of which have the same power?
3. What is the length of the longest succint list of schemata (from four sentence letters) each having power 256?
4. What is the largest n such that the conjunction of any two schema of power n (with 4 sentence letters) is satisfiable?
5. How many ways can you choose 3 marbles from a bag of 15 marbles, assuming the marbles are all distinct? How many ways to take out all 15 marbles from the bag, one by one?
6. How many ways are there to arrange 10 people around a circular table, if we don't count rotations of the same order as being different?
7. Is there a schema of power 22? If so, give one. If not, explain why it's not possible.
8. How many non-equivalent schema over four letters have power greater than 256?

Solutions

1. $2^8 = 256$. For four sentence letters, $p_1 \equiv p_2$ has $2^3 = 8$ satisfying truth assignments. To see why this is the case, note that given a choice for p_1, p_2 is fixed. So we have two choices for p_1, one choice for p_2, and two choices each for p_3 and p_4.

   Plugging in to our formula, we find that the power is $2^{2^4-8} = 2^{16-8} = 2^8$.

2. 17. The power of a sentence S on n sentence letters with k satisfying truth assignments is 2^{2^n-k}. k can take any value from 0 through 16 inclusive when $n = 4$ (since we have $2^4 = 16$ truth-assignments), meaning that the power can be any one of $2^{16}, 2^{15}, ..., 2^0$.

3. $\binom{16}{8}$. A schema on four sentence letters has power $256 = 2^8$ when it is satisfied by 8 truth assignments (because $2^{2^4-8} = 2^8$). Since we have $2^4 = 16$ total truth assignments, the number of non-equivalent propositions of size 8 is the number of subsets of size 8 from 16, which is $\binom{16}{8}$.

4. $n = 2^7 = 128$. With four sentence letters, we have $2^4 = 16$ truth assignments. A schema has power 2^7 is satisfied by $16 - 7 = 9$ truth assignments. By the pigeonhole principle, two schemata of power 2^7 (hence both satisfied by 9 things) must have some satisfying truth-assignment in common (because $9 + 9 = 18 > 16$). Hence the conjunction of any two schemata of power 2^7 is satisfiable, because there must be a truth assignment that satisfies them both.

   Note that 2^7 is the highest power that works, because being satisfied by less than 9 truth-assignments (therefore having a greater power) would mean that the two sentences need not have a satisfying truth-assignment in common. For example, if both sentences were satisfied by 8 truth assignments each, those sets of satisfying truth-assignments could be disjoint, hence the conjunction of the two sentences would not be satisfiable.

5. $\binom{15}{3}$, 15!

6. 9!. There are 10! ways to order 10 people around the table, but that considers different rotations of the same order as different seating arrangements. Since there are 10 rotations of any such ordering, we divide 10! by 10, giving us the answer 9!.

7. No. The power of a schema is always some power of 2. 22 is not a power of 2.

8. $\sum_{i=0}^{7} \binom{16}{i}$. We have $2^4 = 16$ total truth-assignments. The power of a schema S on four sentence letters is greater than $256 = 2^8$ when S is satisfied by less than 8 truth-assignments (because our formula for power is 2^{2^n-k} with $n = 4$ in this case, hence power is greater than 2^8 when k is less than 8). Hence our answer equal to the number of schema that express a proposition of size 0, plus the number that express a proposition of size 1.... plus the number that express a proposition of size 7. Remember that $\binom{n}{k}$ represents the number of size-k subsets from n things, and since propositions are simply subsets of truth-assignments, we arrive at our answer $\sum_{i=0}^{7} \binom{16}{i}$.

3 Monadic Quantification Theory

3.1 Introduction to Monadic Quantification Theory

It's now time to graduate from our humble beginnings in Truth-Functional Logic. We will now begin to consider a more expressive logic, which we'll call *Monadic Quantification Theory*[3]. This is desirable because statements have significant logical form beyond the structure that can be exhibited in terms of truth-functional compounding. For example, the conjunction of the first two statements below implies, but does not truth-functionally imply, the third.

- All collies are mortal.
- Lassie is a collie.
- Lassie is mortal.

In order to analyze this example, consider the following statements:

- Lassie is a collie.
- Scout is a collie.
- Rin-Tin-Tin is a collie.

These statements share the *monadic predicate*[4] "◯ is a collie." Monadic predicates, unlike statements, are not true or false; rather, they are *true of* some objects and *false of* other objects. For example, "◯ is a prime number" is true of 2,3,5 and 7, and false of all even numbers greater than 2.

Definition 7 (Monadic Predicate)**.** *A* monadic predicate *is a property (which may or may not hold) of any single element. For example, "x is red" asserts that the monadic predicate "◯ is red" is true of x.*

Definition 8 (Extension of a Monadic Predicate)**.** [5] *The* extension *of a monadic predicate is the collection of objects of which the monadic predicate is true. For example, the extension of the monadic predicate "◯ is an even natural number" is the set* $\{0, 2, 4, 6...\}$.

You can think of the monadic predicate as "picking out" some subset of what you're talking about (your "universe of discourse"). The subset which the monadic predicate "picks out" is its extension.

What is the extension of the monadic predicate "◯ is a prime number less than 10"? What is the extension of the monadic predicate "◯ is an even prime number"?

Note that distinct monadic predicates might have the same extension - for example, the extensions of "◯ is a warm–blooded reptile" and "◯ is a better movie than Legally Blonde" are the same (namely, they are the emptyset[6]) We say that monadic predicates with the same extension are *coextensive.* Note that coextensive predicates are logically equivalent.

We will focus on statements whose truth depends only on the extensions of the monadic predicates which occur in them. We call such sentential contexts in which interchange of coextensive predicates preserves truth-value *extensional.* We will focus solely on extensional contexts - that is, we will distinguish sentences based only on their logical content. Our focus on extensional contexts is the natural continuation of our earlier focus on truth-functional contexts.

[3]An alternative name for this logic might be *Monadic First-Order Logic*

[4]Also known as a *unary relation.*

[5]Other treatments of logic forego the notion of a "monadic predicate" completely and only focus on extensions. We prefer to distinguish between the two, in order to to highlight that predicates are intensional (and hence cannot be objects of study in logic), whereas the extension of a predicate is a definite mathematical object which is a fair object of study.

[6]We maintain that Legally Blonde is the best movie ever and will not accept counterarguments.

3.2 Syntax

Open Sentences

Consider again the argument that "Lassie is a collie, and all collies are mortal. *Therefore* Lassie is mortal". Intuitively, the validity of this argument does not depend on the particular name "Lassie" being used; it would be equally valid with any name in place of "Lassie."

We can achieve this kind of generality by the use of variables in place of particular names. We will form new expressions called *open sentences* by putting variables $x, y, z, \ldots$ for the placeholders in monadic predicates. For example, "x is a collie" is an open sentence.

Open sentences are not statements. They are true or false with respect to assignments of values to the variables they contain. For example, the open sentence "x is an even number" is true with respect to the assignment $x := 16$ and false with respect to the assignment of $x := 17$. This gives a good justification of why we use the word "open" - ie, the truth of the sentence is an "open question" in absence of information about x. We use the notation $S[x|a]$ to denote the sentence resulting from the the substitution of the constant a for all free occurrences of the variable x in the sentence S.

We may, of course, form compounds of open sentences using truth-functional connectives. For example, the following open sentences are truth-functionally complex.

- $\frac{x}{6} = 0 \supset \frac{x}{3} = 0$.
- x is a collie and x weighs less than 300 kg.

We may use our prior understanding of the truth-functional connectives to determine the truth-values of such open sentences with respect to particular assignments of values to their variables.

The Existential Quantifier

Consider the statement that "there is a prime number". How would we express this? Supposing we had a sentence $P(x)$ which says that x is prime, we would want to say something along the lines of "there is an x such that $P(x)$"[7]. In order to do this, we introduce the existential quantifier $\exists$. Our sentence, "there exists an x such that $P(x)$" can then be written as

$$(\exists x)(P(x))$$

We say that the quantifier here *binds* x. In general, a quantifier Qx binds every instance of x in the outermost parentheses following it.

Note that $(\exists x)(P(x))$ has a truth-value, without any assignment to x. This is because every variable in the sentence is *bound* by a quantifier, and so no assignments need to be made. We call a sentence in which every variable is bound a *closed* sentence.

In the sentence $(\exists x)(F(x)) \wedge x > 3$, the first occurrence of x is bound, whereas the second is not. As such, the two instances of x *may refer to different elements of our universe.* In other words, the sentence is equivalent to $(\exists x)(F(x)) \wedge y > 3$, which makes it clear that the sentence is still open (since it needs an assignment to y to have a truth-value).

Note that a variable may have both free and bound occurrences within a single sentence:

- $(\exists x)$(x is an even number) $\wedge$ (x is a prime number);

and may have occurrences bound by distinct quantifiers:

- $(\exists x)$(x is an even number) $\wedge$ $(\exists x)$(x is a prime number).

[7]Note that $P(x)$ is an open sentence.

The Universal Quantifier

Let's now consider the universal quantifier, which allows us to say that a property holds of "everything". For example, we can render the statement

- all numbers are even or odd

as

- $(\forall x)$ [(x is an even number) or (x is an odd number)].

The last statement is true iff for any integer assignment to x, the open statement within the square brackets is satisfied. In other words, the statement is true for any integer substution for x. Given this interpretation, we are justified in reading the above sentence as "for all x, x is even or x is odd".

Note that context determines our *universe of discourse* - when we say "all numbers" in this context, we intend that the variable of quantification range over all integers and not, for example, all complex numbers.

Monadic Schemata

As we did in the case of truth-functional logic, we will introduce a schematic language for monadic quantificational logic. We specify the following categories of monadic schemata.

- A *one variable open schema* is a truth functional compound of expressions such as $Fx, Gx, Hx, \ldots$.
- A *simple monadic schema* is the existential or universal quantification of a one variable open schema with variable of quantification x.
- A *pure monadic schema* is a truth functional compound of simple monadic schemata.

3.3 Semantics

We now introduce *structures* as interpretations of monadic schemata. These play the role that truth-assignments played in the context of truth-functional logic, in that they bridge the gap between the syntactic objects of our (newly strengthened) language and their truth-values.

In order to specify a structure A for a schema S we need to

- specify a nonempty set U^A, the universe of A;
- specify sets $F^A, G^A, \ldots$ each of which is a subset of U^A as the extensions of the monadic predicate letters which occur in S;
- specify an element $a \in U^A$ to assign to the variable x, if x occurs free in S.

Item (1) sets aside all the objects that are under our consideration (ie, all the objects we're referring to when we quantify). Item (2) specifies for which elements of U^A have the properties F^A, G^A, etc (or, equivalently, which $x \in U^A$ are such that $x \in F^A$, etc). Item (3) makes sure that we substitute in definite values for free variables, if there are any, so that we can evaluate our sentence's truth.

When the variable x has no free occurrences in the schema S, we write $A \models S$ as shorthand for "the schema S is true in the structure A," alternatively "the structure A satisfies the schema S." Otherwise, we write $A \models S[a]$ as shorthand for "the structure A satisfies the schema S relative to the assignment of a to the variable x."

Validity, satisfiability, implication, and equivalence

We extend the notions of validity, satisfiability, implication, and equivalence to monadic quantificational schemata.

- A monadic schema S is *valid* if and only if for every structure $A, A \models S$.
- A monadic schema S is *satisiable* if and only if for some structure $A, A \models S$.
- A monadic schema S *implies* a monadic schema T if and only if for every structure A, if $A \models S$, then $A \models T$.
- Monadic schemata S and T are equivalent if and only if S implies T, and T implies S.

A schema being valid means it is true in all possible interpretations, or as logicians like to say, "all possible worlds". A schema is satisfiable if it's true in at least one possible world.

Show that if S implies T, $\mathbb{S}$ is the set of structures S is true in, and $\mathbb{T}$ is the set of structures that T is true in, then $\mathbb{S} \subseteq \mathbb{T}$.

3.4 Counting Satisfying Structures

Let's consider the problem of how to count the number of structures with a fixed universe of discourse that satisfy a given schema. For example, how many structures with universe of discourse $U = \{1, 2, 3, 4, 5, 6\}$ interpreting the monadic predicate letters F and G satisfy the schema

$$S : \quad (\forall x)(Fx \supset Gx).$$

From now on, we will use the notation $[n] := \{1, 2, ...n\}$

We observed that a structure A satisfies S if and only if $F^A \subseteq G^A$. So we need to determine the number, call it n, of pairs of subsets Y, Z of U with $Y \subseteq Z$. The idea then is to count all the possible sets which Z could be (there are $\sum_{i=0}^{6} \binom{6}{i}$ of these) and multiply by every set which Y could be, given Z (if Z is of size i, there are 2^i subsets of Z, so there are 2^i possibilities for Y). So, by using what we learned earlier about binomial coefficients, we see that

$$\begin{aligned} n &= \sum_{i=0}^{i=6} \binom{6}{i} 2^i \\ &= \sum_{i=0}^{i=6} \binom{6}{i} 2^i \cdot 1^{6-i} \\ &= (2+1)^6 \\ &= 3^6 \end{aligned}$$

The next to last equality is justified by the celebrated *Binomial Theorem*. For those of us with no taste for binomial coefficients, we move on to develop some theory which will give us a much simpler and direct combinatorial argument for the conclusion that $n = 3^6$.

Element Types

Consider the following four one variable open schemata; we will call them (element) types.

- $T_1(x) : Fx \wedge Gx$
- $T_2(x) : Fx \wedge \neg Gx$
- $T_3(x) : \neg Fx \wedge Gx$
- $T_4(x) : \neg Fx \wedge \neg Gx$

Note that a structure A satisfies the schema S if and only if it contains no element satisfying the type T_2. Since a structure is determined by the type of each of its elements, there are as many structures with universe U satisfying S as there are ways of sorting the members of U into the three remaining types. For each of the six members of U, there are three types into which it could be sorted, so by the product rule, the number of structures satisfying S is 3^6.

Counting Counterexamples to an Alleged Implication

If R and R^* are monadic schemata we say that a structure A is a *counterexample* to the claim that R implies R^* if and only if $A \models R$ and $A \not\models R^*$.

Note that R implies R^* iff the number of counterexamples as defined above is zero.

Let's continue with the preceding example and counted the number of counterexamples to the claim that the schema S implies the schema

$$T: \qquad (\forall x)(Gx \supset Fx).$$

Again, let's suppose that our structures have universe of discourse U and interpret exactly the monadic predicate letters F and G. If a structure A satisfies both S and T, then $F^A = G^A$. Hence, of the 3^6 structures satisfying S, the number that also satisfy T is 2^6. So the number of counterexamples to the claim that S implies T (ie, structures which satisfy S but not T) is $3^6 - 2^6$.

3.5 Decidability

Our next order of business it to establish the decidability of pure monadic schema, just as we did for truth-functional schema.[8] Our approach introduces notions that we will elaborate further, when we turn to study polyadic quantificational logic.

Three views of structures

As a warm-up to the main event, we noted that we now have three (equivalent) ways of viewing structures, each of which may contribute a useful perspective, depending on the problem to hand. These are

- the Canonical View, which consists of specifying the universe of discourse and extensions for each of the (finitely many) predicate letters in play,
- the Types View, which consists of specifying a universe of discourse and sorting it into types, that is, maximally specific descriptions that can be framed in terms of the predicate letters in play, and
- the Venn View, which pictures the extensions of all the predicate letters in play as intersecting regions contained in a rectangle that represents the universe of discourse.

Show that these three views are equivalent by giving a correspondence between them. For example, x being of type $F \wedge \neg G$ in the Types View corresponds to the statement that $x \in F \wedge x \notin G$ in the Canonical view.

The small model theorem

We will prove the following *Small Model Theorem* for monadic logic; the decidability of satisfiability of pure monadic schemata is a corollary to this result.

Theorem 2. *Let S be a pure monadic schema containing occurrences of at most n distinct monadic predicate letters. If S is satisfiable then there is a structure A of size at most 2^n such that $A \models S$.*

Why is decidability a corollary to Theorem 2? As a hint, think about why truth-functional logic is decidable.

Monadic similarity

The proof of Theorem 2 rests on a lemma, and so we first need to introduce some new concepts. In what follows, we will without loss of generality restrict our attention to monadic schemata in which only the predicate letters F and G occur[9]. First, a definition.

Definition 9. *We say that two structures A and B are* monadically similar *and write $A \approx_M B$ if and only if they satisfy exactly the same pure monadic schemata.*

Show that monadic similarity is an *equivalance relation*, ie it is reflexive ($A \approx_M A$), symmetric (if $A \approx_M B$, then $B \approx_M A$), and transitive (if $A \approx_M B$ and $B \approx_M C$, then $A \approx_M C$).

We now now turn towards developing our Lemma, which explores a sufficient condition for the monadic similarity of structures.

[8] See Goldfard Chs. 25-26 for additional exposition on the decidability of pure monadic schema.

[9] The choice of using only two variables F, G is simply for notational and instructional convenience. The generalization to n predicate letters is simple, and we will end up proving results for n variables, not just two.

Homomorphisms

A function h is a mapping from one set, called the *domain* of h to another set (it may be the same set), called the *range* of h. For every element a of the domain of h we write "$h(a)$" to denote the element of the range of h to which it is mapped. We sometimes call $h(a)$ the h *image* of a or the *image* of a under h. We sometimes use the notation

$$h : X \longrightarrow Y$$

to indicate that h is a function with domain X and range Y. If $h : X \longrightarrow Y$ we say that h is *onto* if and only if for every $b \in Y$ there is an $a \in X$ such that $h(a) = b$. In this case, we will also say that h is *surjective.*

Let A and B be structures. We call h a *homomorphism from* A *onto* B just in case h is an onto function with domain U^A and range U^B satisfying the following condition: for every monadic predicate letter P and every $m \in U^A$,

$$m \in P^A \quad \text{if and only if} \quad h(m) \in P^B.$$

If there is a homomorphism from A onto B, we say that B is a *surjective homomorphic image* of A.

Intuitively, a homomorphism is a function that loosely, *"preserves the arrangement"* of elements in its domain, ie elements which are of type P get mapped to an element in the range also of type P, etc.

Example

As an example, consider the following structures.

$$\begin{aligned} A: \quad & U^A = \{n \mid n \text{ is a positive integer.}\} \\ & F^A = \{n \mid n \text{ is an even positive integer.}\} \\ & G^A = \{n \mid n \text{ is a prime positive integer.}\} \end{aligned}$$

$$\begin{aligned} B: \quad & U^B = \{n \mid n \text{ is a positive integer.}\} \\ & F^B = \{n \mid n \text{ is an odd positive integer.}\} \\ & G^B = \{n \mid n \text{ is a prime positive integer.}\} \end{aligned}$$

Note that A and B both have the same regions occupied in their respective Venn diagrams, ie F^A and F^B are both nonempty, as are both G^A and G^B. However, there is no homomorphism from A onto B, nor any homomorphism from B onto A.

Prove the last assertion.

Although A and B are not homomorphic, we will shortly see that A and B have a common surjective homomorphic image, ie that there is a structure C such that there is a homomorphism from A onto C and a homomorphism from B onto C.

Homomorphisms and monadic similarity: the central lemma

The next lemma provides a useful sufficient condition for monadic similarity.

Lemma 2. *Let A and B be structures. If there is a homomorphism from A onto B, then A is monadically similar to B.*

Proof: Let A and B be structures and suppose that h is a homomorphism of A onto B. It suffices to show that for every simple monadic schema S,

$$A \models S \quad \text{if and only if} \quad B \models S,$$

since every pure monadic schema is a truth-functional compound of simple monadic schemata.

We begin by observing that for every $c \in U^A$ and every one variable open schema S, A makes S true with respect to the assignment of c to "x," if and only if B makes S true with respect to the assignment of $h(c)$ to "x." This follows immediately from the fact that h is a homomorphism.

Consider the simple schema S and suppose that S is the existential quantification of the the one variable open schema T. Suppose $A \models S$. Then, for some $c \in U^A$, A makes T true with respect to the assignment of c to "x." It follows that B makes T true with respect to the assignment of $h(c)$ to "x." Hence, $B \models S$.

Conversely, suppose $B \models S$. Then, for some $c \in U^B$, B makes T true with respect to the assignment of c to "x." Since h is surjective, there is a $d \in U^A$ with $h(d) = c$. It follows at once that A makes T true with respect to the assignment of d to "x." Hence, $A \models S$.

The case of universal quantification is handled similarly □

Write out the universal case formally. The argument should be very similar to the existential case, so make sure you understand that!

Types and monadic similarity

We recall our discussion of element types:

- $T_1(x) : Fx \wedge Gx$
- $T_2(x) : Fx \wedge \neg Gx$
- $T_3(x) : \neg Fx \wedge Gx$
- $T_4(x) : \neg Fx \wedge \neg Gx$

We say that a structure *realizes* a given type T_i just in case it makes the existential simple schema $(\exists x)T_i$ true (ie, tere is at least one element of type T_i).

Example 10. *The following structure realizes all four of the types listed above.*

$$A : U^A = \{1, 2, 3, 4\}, F^A = \{1, 3\}, G^A = \{1, 2\}$$

Moreover, the 14 proper substructures of A realize exactly the fourteen proper nonempty subsets of the types listed above.

Lemma 3 yields a useful necessary and sufficient condition for monadic similarity.

Lemma 3. *A and B realize the same types if and only if they are monadically similar.*

Proof: First, the forward direction. Suppose A, B realize the same types. Then there is a single structure C which is a surjective homomorphic image of both A and B (simply map every element of a given type in A or B onto a single element of that type in C).

Therefore, by our earlier result, A is monadically similar to C and B is monadically similar to C. It follows at once that A is monadically similar to B (as monadic similarity is an equivalent relation, and hence transitive).

The reverse direction follows immediately from the fact realization of a type is expressed by a pure monadic schema. □

In the above proof, C was a "canonical model" which realized the same types as A, B.

The small model theorem and the decidability of satisfiability

Theorem 2 is an immediate corollary to Lemma 3.

Proof (of Theorem 2): Suppose S is a schema involving only the predicate letters F, G. Then it follows at once from Lemma 3 and Example 10, that there is a collection X of 15 structures each of size ≤ 4, such that if S is satisfiable, then there is a structure $C_i \in X$ such that $C_i \models S$.

Why? Because there are only 15 canonical models. There are 2^4 subsets of types (since we have $2^2 = 4$ types), but only 15 of those are realizable (since for any nonempty model, at least one type must be realized). For $i \in [15]$, we let each C_i realize different types; since there are only 15 possible realized types, it follows that every structure has some C_i as its homomorphic image. These C_i are our canonical models.

More generally, there is a collection X of $2^{(2^n)} - 1$ structures each of size $\leq 2^n$ such that for any pure monadic schema S involving only the predicate letters "F_1," ... "F_n," if S is satisfiable, then there is a structure $A \in X$ such that $A \models S$. $\square$

There are 2^n types given n monadic predicate letters, so there are 2^{2^n} subsets of types, of which $2^{2^n} - 1$ are realizable.

Corollary 2. *For every schema S, if S is satisfiable, then there is an $1 \leq i \leq 15$ such that $A_i \models S$.*

Corollary 3. *There is a decision procedure to determine whether a pure monadic schema is satisfiable.*

Corollary 4.

Schema S implies schema T if and only if

$$\{i \mid A_i \models S \text{ and } 1 \leq i \leq 15\} \subseteq \{i \mid A_i \models T \text{ and } 1 \leq i \leq 15\}.$$

Schemata S and T are equivalent if and only if

$$\{i \mid A_i \models S \text{ and } 1 \leq i \leq 15\} = \{i \mid A_i \models T \text{ and } 1 \leq i \leq 15\}.$$

Show that these are all quick corollaries to the small model theorem.

3.6 Expressive Power

The expressive power of monadic quantification theory

With these results in hand, we proceed to analyze the expressive power of monadic schemata. For simplicity's sake, we'll continue to focus on the vocabulary consisting of the monadic predicate letters F and G. First, some definitions.

- A list of pure monadic schemata is *succinct* if and only if no two schemata on the list are equivalent.
- A pure monadic schema *implies a list of schemata* if and only if it implies every schema on the list.
- The *power* of a pure monadic schema is the length of a longest succinct list of pure monadic schemata it implies.

Now, the main question: how expressive is MQT?

Question 1. *What is the length of a longest succinct list of pure monadic schemata (in the vocabulary consisting of just the monadic predicate letters F and G)? In other words, how many propositions can MQT express?*

Answer: It follows immediately from Corollary 4, part (4) that the length of a longest such list is 2^{15}, since a schema is determined, up to equivalence, by which of the structures $A_1, \ldots, A_{15}$ satisfy it.

Question 2. *For which numbers n is there a schema S whose power is n?*

Answer: It follows from Corollary 4, parts (4) and (4), that the power of a schema S is determined by the size j of $\{i \mid A_i \models S \text{ and } 1 \leq i \leq 15\}$, in particular, the power of S is 2^{15-j}; for pure schemata S, j may be any number between 0 and 15. This answers Question 2.

Definition 10. *If X is a finite set, we write $|X|$ for the number of members of X.*

If S is a schema, we write $\mathsf{mod}(S, n)$ for the set of structures A such that $A \models S$ and $U^A = \{1, \ldots, n\}$.

Question 3. *What is the length of a longest succinct list of pure schemata S such that $\mathsf{mod}(S, 4) = 4$?*

Answer: Let $\mathbb{V} = \{A \mid U^A = \{1, 2, 3, 4\}\}$. Recall that $A \approx_M B$ if and only if for all pure monadic schemata S, $A \models S$ if and only $B \models S$. For $A \in \mathbb{V}$, let $\hat{A} = \{B \in \mathbb{V} \mid B \approx_M A\}$.

$\hat{A}$ is the monadic similarity *equivalence class* of A, ie all structures which are monadically similar to A. Generally, an equivalence class of an object N is the set of all objects which are equivalent to N under some equivalence relation.

In order to answer the question, it suffices to determine the size of $\hat{A}$ for each $A \in \mathbb{V}$. First, note that the size of $\hat{A}$ is determined by the number of types realized by A. We computed these sizes:

- If A realizes exactly 1 type, then the size of $\hat{A}$ is 1, since monadically similar structures realize the same types, and there is only 1 way to place 4 elements into one given type. There are $\binom{4}{1}$ structures in $\mathbb{V}$ satisfying exactly 1 type (since there are 4 choices of which type is realized).
- If A realizes exactly 2 types, then the size of $\hat{A}$ is $2^4 - 2$ (since there are $2^4 - 2$ ways of distrubuting 4 elements into two given types such that each type is nonempty). There are $\binom{4}{2}$ structures in $\mathbb{V}$ satisfying exactly 2 types (since there are $\binom{4}{2}$ choices for which two types are realized).
- If A realizes exactly 3 types, then the size of $\hat{A}$ is $\binom{4}{2} \cdot 3!$ (since there are $\binom{4}{2} \cdot 3!$ ways of distrubuting 4 elements into three given types such that each type is nonempty). There are $\binom{4}{3}$ structures in $\mathbb{V}$ satisfying exactly 3 types.

- If A realizes exactly 4 types, then the size of $\hat{A}$ is 4! (since there are 4! ways of ordering the 4 elements, with the i^{th} element in our order being places in T_i). There are $\binom{4}{4} = 1$ structures in $\mathbb{V}$ satisfying exactly 4 types.

If the size of $\hat{A}$ is confusing for any of the above, try to count these yourself! If you're still stuck, come into Office Hours and we'll be happy to help.

By Theorem 3, if $A \models S$ then for all $A_i \in \hat{A}$, $A_i \models S$. It follows that the answer to Question 3 is 1; in particular, one such list consists of the single schema

$$(\forall x)(Fx \wedge Gx) \vee (\forall x)(Fx \wedge \neg Gx) \vee (\forall x)(\neg Fx \wedge Gx) \vee (\forall x)(\neg Fx \wedge \neg Gx).$$

3.7 Review

Concept Review

Definitions

- A *one variable open schema* is a schema formed of predicates connected by truth-functional connectives (for example, $(Fx \wedge Gx)$).
- A *simple monadic schema* is a schema in which some quantifier binds a one-variable open schema (for example, $(\forall x)(Fx \wedge Gx)$).
- A *pure monadic schema* is schema which results from the truth-functional compounding of simple monadic schemata (for example, $(\forall x)(Fx \wedge Gx) \vee (\exists x)(Gx)$)
- A sentence S is satisfiable iff there is at least one structure A such that $A \models S$. Note that this is analogous to the corresponding definition for TF-logic, the only difference being that A is now a structure, not a truth-assignment.
- A sentence S is valid iff for all structures A, $A \models S$
- A sentence S implies a sentence T iff for all A, if $A \models S$, then $A \models T$
- Sentences S and T are equivalent iff they are satisfied by exactly the same structures.
- A structure A is said to "be a counterexample to the claim that a sentence S implies a sentence T" iff $A \models S$ and $A \not\models T$.
- A structure A is said to "witness the inequivalence of sentences S and T" iff ($A \models S$ and $A \not\models T$), or ($A \models T$ and $A \not\models S$).
- Structures A, B are said to be *monadically similar* ($A \approx_M B$) iff they satisfy the same sentences.
- A function h is *surjective* (or *onto*) if everything in the codomain is mapped to by h (equivalently, the image of the function - which is the set of all things that get mapped to - is equal to the codomain). In the language of first order logic, the criterion for surjetivity is

$$(\forall b \in B)(\exists a \in A)(h(a) = b)$$

- A *Homomorphism h from A onto B* satisfies the following three properties:
  - h is a function from A to B.
  - h is *surjective (onto).*
  - h is "structure preserving". This just means that the predicates which elements are members of are preserved under the homomorphism. You can think of this as elements of types in the Type View mapping onto elements in the corresponding type, or sections in the Venn View mapping onto their corresponding section. In the language of first order logic, the criterion for this is

$$(\forall \text{ predicates } P)(\forall a \in A)(a \in P^A \equiv h(a) \in P^B)$$

Quantifiers $\forall x$ is read "for all x", "for every x", or "for each x". A sentence of the form

$$(\forall x)(\text{some statement about } x)$$

is true just in case "some statement about x" is true no matter what x is.

$\exists x$ is read "there exists an x". A sentence of the form

$$(\exists x)(\text{some statement about } x)$$

is true just in case "some statement about x" is true of at least one thing x.

(Unary) Predicates *Predicates* are statements that are true of some things and not true of others. The correspond to the "some statement about x" segment of the aforementioned sentences. For example, "$\bigcirc$ is an even number" is a predicate that is true just of the elements of the set of even integers, and false of all other things. The set of things for which a predicate is true is called its *extension*. For example, the extension of "$\bigcirc$ is an even number" is the set $\{0, 2, -2, 4, -4...\}$.

Instead of writing out the full dscription for predicates each time we use them in a schema, we represent predicates by *predicate letters*. For example, we might say that Fx represents the statement that "x is an even number".

Structures In truth-functional logic, the truth of a sentence was relative to the truth-assignments to its sentence letters, and that only. Structures play the same role for logic with quantifiers as truth-assignments do for truth-functional logic.

A structure A consists of:

- A set called the *universe of* A, written U^A. This is the set of all things over which we are quantifying when we use a quantifier.
  - For example, if I asserted that $(\forall x)(x \text{ is even} \vee x \text{ is odd})$, you'd probably assume that I was talking about all integers, not all things in general. In this case, the universe would (implicitly) be the set of integers, then. This implicit universe is a function of natural language, which is context-dependent. In class, you'll normally explicitly be told that some set is the universe in consideration.
- Any number of monadic predicates (F^A, G^A, etc), all of which are (arbitrary) subsets of the universe.
  - Let F^A be the set of even numbers, and G^A be the set of odd numbers. Then, within the universe of integers, the statement $(\forall x)(Fx \vee Gx)$ is a true statement, asserting that $(\forall x)(x \text{ is even} \vee x \text{ is odd})$.

If a sentence is true relative to some structure, we write $A \models S$ and say that "A satisfies S" (as we did for TF-logic), "A is a model of S", or "S is true in A".

Bound vs. Free Quantifiers A variable x is said to be *bound* if it is within the scope of some quantifier. x is said to be *free* if it is not bound. The truth of sentences including free variables cannot be evaluated without an *assignment* of a value to those free variables. For example, the sentence "x is an even number" means nothing without some assignment of a value (say, 2 or 3) to x. However, the sentence $(\forall x)(x \text{ is an integer} \supset x \text{ is even})$ can be evaluated (and is, of course, false).

3 Views of Structures We have three equivalent ways of looking at strctures, which are

1. **The Canonical View**, which involves specifying the universe of discourse and extensions of predicates as sets.
2. **The Types View**, which involves drawing a table with sections for each "type".
3. **The Venn View**, which involves drawing a venn diagram, wherein the circles represent extensions of predicates.

Realizing Types We say that a structure realizes a type T_i iff there is some element of the structure in the quadrant T_i in the types view of our structure.

For example, with predicate letters F, G, the types would then be

- $T_1(x) : Fx \wedge Gx$
- $T_2(x) : Fx \wedge \neg Gx$
- $T_3(x) : \neg Fx \wedge Gx$
- $T_4(x) : \neg Fx \wedge \neg Gx$

and then we would say that a structure realizes type T_i iff it makes the sentence $(\exists x)(T_i(x))$ true.

The Small Model Theorem The Small Model Theorem states:

Theorem 3. *Let S be a pure monadic schema over n predicate letters. If S is satisfiable, then there is a structure A with $|A| \leq 2^n$ with $A \models S$.*

This immediately gives as a corrolary the algorithmic decidability of satisfiability for MQT. To see this, we noted the following equivalence (the *contrapositive*)

$$(p \supset q) \equiv (\neg q \supset \neg p)$$

Then the contrapositive of our theorem is: *if there is no structure A with $|A| \leq 2^n$ such that $A \models S$, then A is not satisfiable.* Thus, we have to check only a finite number of models to see if S is satisfiable or not, and hence satisfiability is algorithmically decidable.

Problems

For the later problems, it will be immensely useful to you to draw tables that look like this. Try to interpret the schema as statements about which quadrants of the table can/must have elements in them for the schema to be satisfied or falsified (this is the "types view").

	Fx	$\neg Fx$
Gx		
$\neg Gx$		

1. Are the following sentences equivalent? If not, does one imply the other?

$$S : (\forall x)(Px)$$

$$T : \neg(\exists x)(\neg Px)$$

2. Are the following sentences equivalent? If not, does one imply the other?

$$S : (\exists x)(Px)$$

$$T : \neg(\forall x)(\neg Px)$$

3. Are the following sentences equivalent? If not, does one imply the other?

$$S : (\forall x)(Px) \wedge (\forall x)(Qx)$$

$$T : (\forall x)(Px \wedge Qx)$$

4. Are the following sentences equivalent? If not, does one imply the other?

$$S : (\exists x)(Px) \wedge (\exists x)(Qx)$$

$$T : (\exists x)(Px \wedge Qx)$$

5. Are the following sentences equivalent? If not, does one imply the other?

$$S : (\forall x)(Px) \vee (\forall x)(Qx)$$

$$T : (\forall x)(Px \vee Qx)$$

6. Let

$$S : (\exists x)(Fx \wedge Gx) \wedge (\exists x)(\neg Fx \wedge Gx) \wedge (\exists x)(Fx \wedge \neg Gx) \wedge (\exists x)(\neg Fx \wedge \neg Gx)$$

$$T : (\forall x)(Fx \equiv Gx)$$

   (a) How many structures with universe $U = \{1, 2, 3\}$ are counterexamples to the claim that S implies T?

   (b) How many structures with universe $U = \{1, 2, 3, 4, 5\}$ are counterexamples to the claim that S implies T?

7. Let

$$S : (\forall x)(Fx \oplus Gx)$$

$$T : (\forall x)(Fx \equiv Gx)$$

How many structures are there with universe $U = \{1, 2, 3, 4, 5\}$ which witness the inequivalence of S and T?

8. Let

$$S : (\exists x)(Fx \wedge Gx)$$

$$T : (\forall x)(Fx \vee Gx)$$

How many structures are there with universe $U = \{1, 2, 3, 4, 5\}$ which witness the inequivalence of S and T?

9. Let

$$S : (\forall x)(Fx \oplus Gx)$$

$$T : (\forall x)(Fx) \oplus (\forall x)(Gx)$$

Given universe $U = \{1, 2, 3, 4, 5\}$, how many counterexamples are there to the claim that S implies T?

10. Let

$$S : (\forall x)(Fx \supset Gx)$$

$$T : (\forall x)(Fx) \supset (\forall x)(Gx)$$

Given universe $U = \{1, 2, 3, 4, 5\}$, how many counterexamples are there to the claim that S implies T?

Solutions

1. These are equivalent. The former asserts that everything has property P. The latter asserts that nothing does not have property P. These are equivalent statements.

2. These are equivalent. The former asserts that there is something with property P. The latter asserts that it is not the case that everything is not P. These are equivalent statements.

3. These are equivalent. The former asserts that everything is P, as well as that everything is Q. The latter asserts that everything is both P and Q.

4. These are not equivalent, but T implies S. S asserts that there is something that is P, and there is something that is Q. T asserts that there is something that is both P and Q. Clearly if T is true, S must be as well. Hence T implies S. The inequivalence of S and T is witnessed by the strcture A with $U^A = \{1, 2\}$, $P^A = \{1\}$, $Q^A = \{2\}$. Then $A \models S$ but $A \not\models T$.

5. These are not equivalent, but S implies T. S asserts that everything is P or everything is Q. T asserts that everything is at least one of P or Q. Clearly if S is true then so is T, hence S implies T. The inequivalence of S and T is witnessed by the strcture A with $U^A = \{1, 2\}$, $P^A = \{1\}$, $Q^A = \{2\}$. Then $A \models T$ but $A \not\models S$.

6. (a) 0. To satisfy S, there must be at least one element which is each of $(Fx \wedge Gx)$, $(Fx \wedge \neg Gx)$, $(\neg Fx \wedge Gx)$, and $(\neg Fx \wedge \neg Gx)$. We only have three elements though, so at least one of those categories won't have an element in it. Hence S can't be satisfied in this universe. Hence there are no structures A such that $A \models S$ and $A \not\models T$.

   (b) $\binom{5}{2} \cdot 4! = 240$. To satisfy S, there must be at least one element which is each of $(Fx \wedge Gx)$, $(Fx \wedge \neg Gx)$, $(\neg Fx \wedge Gx)$, and $(\neg Fx \wedge \neg Gx)$. Think of these categories as "boxes" into which we are placing elements of our universe (draw a table to help yourself think problems like this through!). There are $\binom{5}{2} \cdot 4!$ ways to satisfy S. The $\binom{5}{2}$ term results from choosing two items from U which will go into the same "box". Henceforth we think of these two elements as now being a "package deal". The 4! term is the number of ways we can order our four things (the three single elements, plus our "package deal") into the four "boxes". Note that any structure that satisfies S cannot satisfy T. Hence all of the structures satisfying S are counterexamples to the claim that S implies T, and we have our answer.

7. 2^6. Note that if a structure satisfies S it cannot satisfy T, and similarly if one satisfies T it cannot satisfy S. Hence it suffices to count the number of ways S, T can each be satisfied and add those results together.

   To satisfy S, every element of U must be either $(Fx \wedge \neg Gx)$ or $(\neg Fx \wedge Gx)$. This corresponds to two choices for each of our 5 elements, meaning there are 2^5 ways to satisfy S. Note that none of these structures satisfy T.

   Similar reasoning suffices to show that there are 2^5 ways to satisfy T. None of these structures satisfy S.

   Our answer is then $2^5 + 2^5 = 2^6$.

8. $4^5 - 2 \cdot 3^5 + 2^6 = 602$.

   Let's start by counting the number of structures which are counterexamples to the claim that T implies S, since that direction is easier. For T to be true, everything must be of the form $(Fx \wedge Gx)$, $(Fx \wedge \neg Gx)$, or $(\neg Fx \wedge Gx)$. For S to be false, nothing can be of the form $(Fx \wedge Gx)$. hence for T to be satisfied and S to be falsified, everything must be of the form $(Fx \wedge \neg Gx)$ or $(\neg Fx \wedge Gx)$. Since we have two choices per element then, there are 2^5 total ways to satisfy T and not satisfy S.

How many ways are there to satisfy S and not satisfy T? Let's begin by counting the ones that satisfy S. These are the structures which have at least one element of the form $(Fx \wedge Gx)$. There are 4^5 total assignments of our 5 elements to the four categories, 3^5 of which place no element in $(Fx \wedge Gx)$. Hence there are $4^5 - 3^5$ structures satifying S. From this number, we need to subtract the number that also satisfy T. 3^5 structures satisfy T (the structures in which every element is of the form $(Fx \wedge Gx)$, $(Fx \wedge \neg Gx)$, or $(\neg Fx \wedge Gx)$), so we subtract that from the number which satisfy S to get $4^5 - 2 \cdot 3^5$. Notice, however, that we *took away all structures whose elements were only of the form* $(Fx \wedge \neg Gx)$ *or* $(\neg Fx \wedge Gx)$ *twice - once when we were counting the number that satisfied* S*, and another time when counting the number that satisfed* T. Hence we "double subtracted" 2^5 possibilities, and we must add this back. This gives the result that there are $4^5 - 2 \cdot 3^5 + 2^5$ structures satisfying S which do not satisfy T.

Adding these two results together, we get $2^5 + (4^5 - 2 \cdot 3^5 + 2^5) = 4^5 - 2 \cdot 3^5 + 2^6 = 602$.

9. $2^5 - 2$. Note that S is satisfied by 2^5 truth assignments (each element can be in either the box $(Fx \wedge \neg Gx)$ or the box $(Gx \wedge \neg Fx)$). Of these, only two satisfy T (the one in which all elements are in $(Gx \wedge \neg Fx)$, and the one in which all elements are in $(Fx \wedge \neg Gx)$). Hence there are $2^5 - 2$ truth assignments which satisfy S and falsify T.

10. 0. There are 3^5 truth-assignments satisfying S. Of these, none falsify T.