CS2002 Logic Lecture 3

Truth Tables and more on Formalisation

Ian Gent

This Time

- Some Key Terms
- Another example of formalization
- Boolean Algebra
- A larger example of formalization
- (if we have time) How To Prove Things in Logic

Some key terms

- A logic formula can be one or more of these
  - **Valid**
    - Every way of assigning the variables T/F makes the formula true
  - **Tautology**
    - Exactly the same as valid
  - **Invalid**
    - Not valid
    - There is at least one assignment which makes the formula untrue
  - **Satisfiable**
    - At least one assignment to T/F makes formula true
  - **Unsatisfiable**
    - No assignment can make the formula true
- Try to understand relationships between these
  - Both for a formula and the negation of a formula

Operator Precedence

- Simplified notation for formulae can be used by declaring a **precedence order** on operators. We give a higher priority to the unary negation operator $\neg$ than the binary conjunction $\wedge$.

$$A \vee (\neg B \wedge C)$$

- No standard precedence between and/or, so use brackets to clarify where necessary
- But note we don't need to write $A \vee ((\neg B) \wedge C)$

On Boolean Expressions

- Boolean Algebra deals with binary variables and logical operations on those variables
- Use truth tables to represent expressions
- A Boolean expression should be given in its simplest form
- When using AND/OR/NOT/XOR and other symbols for which there are standard gates
  - We can translate a Boolean expression into a logic circuit
- Unnecessary terms => unnecessary components in physical circuits

Equivalence

- A and B propositional formulae are called **equivalent** if A$\leftrightarrow$B is a **tautology**
- Can prove this by truth tables in two ways
    - build a truth table for E$\leftrightarrow$F
        - Show that every line is true
    - build truth tables for E and F and compare
        - Every line should end in the same way in each table

From English to Logic

If I ate lunch in St Andrews, I went to Costa or Dino’s.
If I didn’t have a Burger, then I wasn’t at Costa.
If I was at the Doll’s House, I had a Burger.
I didn’t have a Burger.

Did I eat lunch in St Andrews?

From English to Logic

State atomic propositions
And what they mean in English

I ate lunch in St **A**ndrews	A
I had a **B**urger	B
I went to **C**osta	C
I went to Dino’s	D

From English to Logic

Encode using those propositions

If I ate lunch in St Andrews, I went to Costa or Dino's.	$A \rightarrow (C \vee D)$
If I didn't have a Burger, then I wasn't at Costa	$\neg B \rightarrow \neg C$
If I was at Dino's, I had a Burger	$D \rightarrow B$
I didn't have a Burger	$\neg B$

Query: Does it follow I didn't eat lunch in St Andrews?

Can we deduce $\neg A$ from the above statements?

Here's the truth table…

A	B	D	C	$A \to (C \vee D)$	$\neg B \to \neg C$	$D \to B$	$\neg B$	$(A \to \ldots) \wedge (\neg B \to \neg C) \wedge (D \to B) \wedge \neg B$
F	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**
F	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**
F	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**
F	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**
F	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**
F	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**
F	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**
F	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**
T	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**F**
T	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**
T	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**
T	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**
T	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**
T	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**
T	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**
T	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**

Query: How do we use it to answer our question?

Here’s the truth table…

A	B	D	C	$A \to (C \lor D)$	$\neg B \to \neg C$	$D \to B$	$\neg B$	$(A \to \ldots) \land (\neg B \to \neg C) \land (D \to B) \land \neg B$
F	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**
F	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**
F	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**
F	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**
F	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**
F	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**
F	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**
F	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**
T	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**F**
T	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**
T	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**
T	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**
T	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**
T	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**
T	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**
T	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**

Look at every row where the whole formula is true

Is ¬A true in ALL those rows?

In this case, yes.

So it follows I did not eat lunch in St Andrews

Proving things in Logic

- We showed that ¬A had to be true whenever all of the formula was true
- We used a truth table to do this
- But there are other ways
- One is Boolean Algebra (in a moment)
- Another is DPLL (in next lecture)

Quiz…

- If a formula X is a tautology…
  - Is X valid?
  - Is X satisfiable?
  - Is ¬X satisfiable?
- If a formula X is satisfiable
  - Is X a tautology?
  - Is X unsatisfiable?
  - Is ¬X unsatisfiable?
  - Is ¬X satisfiable?

Boolean Algebra

- An algebra for manipulating objects that can have **two values**: true and false, 1 and 0, or simply T and F
- Operations can be applied to these objects or variables, leading to **Boolean expressions** (logical formulae)
- Operators can be completely described by a **truth table**
- Common **Boolean operators** include AND, OR, NOT, XOR, IMPLIES
- Boolean Algebra includes **variables**
  - also called “**logic variables”,** “**propositions”, “propositional variables”**
  - often written as single letters, e.g. A, B, C…
  - each variable can only be true or false

Laws of Boolean Algebra

A Law is an equation, e.g.

$$A \wedge \text{True} = A$$

This is always true: a “tautology”.
Can **instantiate** this law with any formula for A

Say we want to prove**:** $\mathbf{(p \wedge True) \wedge True = p}$

Instantiate rule above with $A = (p \wedge \text{True})$:

$$(p \wedge \text{True}) \wedge \text{True} = (p \wedge \text{True}) \qquad (1)$$

Now Instantiate with $A = p$:

$$(p \wedge \text{True}) = p \qquad (2)$$

Substitute (2) in the Right Hand Side of (1)
QED

Lots of laws

$A \wedge 0 = 0$ $A \wedge 1 = A$ $A \vee 0 = A$ $A \vee 1 = 1$	**Zero and Unit Laws** **0 is false** **1 is true**
$A \wedge A = A$ $A \vee A = A$	**Idempotence Laws**
$A \wedge B = B \wedge A$ $A \vee B = B \vee A$	**Commutative Laws**
$(A \wedge B) \wedge C = A \wedge (B \wedge C)$ $(A \vee B) \vee C = A \vee (B \vee C)$	**Associative Laws** It is ok to write e.g. $A \vee B \vee C$

Writing a Proof

To prove: $(0 \wedge p) \vee q = q$

$$
\begin{aligned}
(0 \wedge p) \vee q &= (p \wedge 0) \vee q && \{\wedge \text{ Commutative }\} \\
&= 0 \vee q && \{\ \wedge \text{ Zero }\} \\
&= q \vee 0 && \{\vee \text{ Commutative}\} \\
&= q && \{\vee \text{ Zero }\}
\end{aligned}
$$

Say **which** law applies but skip details of the instantiation

Lots *more* laws in a minute

Before I give you even more laws…

The key thing is not to learn them by rote

But two things:

1. Convince yourself each one of them is true

2. Be able to use them when they are written down in front of you

Laws and Duality

The laws given so far in this lecture are **complete**

Any equivalence which is correct can be proved using them

The laws obey a **duality**

Swap $\vee$ and $\wedge$ and 1 and 0 in any law, gives another correct law

I won't prove these statements, but they are true

Lots more laws

$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$ $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$	Distribution Laws
$\neg(A \wedge B) = \neg A \vee \neg B$ $\neg(A \vee B) = \neg A \wedge \neg B$	De Morgan's Laws
$\neg 1 = 0$ $\neg 0 = 1$ $A \vee \neg A = 1$ $A \wedge \neg A = 0$	Complement Laws
$\neg\neg A = A$	Double Negation
$A \vee (A \wedge B) = A$ $A \wedge (A \vee B) = A$	Absorption Laws

Another Example

Prove: $p \land \neg(q \lor p) = 0$

Another Example

Prove: $p \wedge \neg(q \vee p) = 0$

$p \wedge \neg(q \vee p)$	$= p \wedge (\neg q \wedge \neg p)$	{De Morgan}
	$= p \wedge (\neg p \wedge \neg q)$	{Commutative}
	$= (p \wedge \neg p) \wedge \neg q$	{Associative}
	$= 0 \wedge \neg q$	{∧ Complement}
	$= 0$	{∧ Zero }

Exercise:

Use Boolean Algebra to Simplify:

1. $p \lor \neg(q \land p)$
2. $\neg(p \land q) \land (\neg p \lor q) \land (p \lor p)$
3. $\neg(p \land q) \land (\neg p \lor q) \land (\neg q \lor q)$
4. $\neg p \land (p \lor q) \lor (q \lor p) \land (p \lor \neg q)$

Larger Example of Formalisation

- More complicated logical problem
- Encode in Logic
- Look at later stages next time

Example Problem

There are three suspects for a murder: Adams, Brown, and Clark. Adams says "I didn't do it. The victim was an old acquaintance of Brown's. But Clark hated him." Brown states "I didn't do it. I didn't know the guy. Besides I was out of town all the week." Clark says "I didn't do it. I saw both Adams and Brown downtown with the victim that day; one of them must have done it." Assume that the two innocent men are telling the truth, but that the guilty man might not be. Write out the facts as sentences in Propositional Logic, and use propositional reasoning to solve the crime.

Formalisation (1)

Variables

A: **A**dams is guilty

B: **B**rown is guilty

C: **C**lark is guilty

K: Brown **K**new the victim

H: Clark **H**ated the victim

O: Brown was **O**ut of town

Formalisation (2)

Adams Says: $\neg A \land K \land H$
Brown Says: $\neg B \land \neg K \land O$
Clark Says: $\neg C \land K \land \neg O \land (A \lor B)$

But … one of them may be lying …

Formalisation (3)

Guilty man *may* be lying
I.e. non guilty people telling truth:

$\neg A \rightarrow (\neg A \wedge K \wedge H)$
$\neg B \rightarrow (\neg B \wedge \neg K \wedge O)$
$\neg C \rightarrow (\neg C \wedge K \wedge \neg O \wedge (A \vee B))$

Formalisation (4)

One of them is guilty

$A \lor B \lor C$

Two of them are innocent

I.e. *Only* one is guilty

$\neg(A \land B)$

$\neg(A \land C)$

$\neg(B \land C)$

Complete set of statements

$\neg A \rightarrow (\neg A \wedge K \wedge H)$

$\neg B \rightarrow (\neg B \wedge \neg K \wedge O)$

$\neg C \rightarrow (\neg C \wedge K \wedge \neg O \wedge (A \vee B))$

$A \vee B \vee C$

$\neg(A \wedge B)$

$\neg(A \wedge C)$

$\neg(B \wedge C)$

How to PROVE Things in Logic

- Key word is “prove”
- I.e. give a guarantee that something is correct
- This is typically the point of converting to logic
- E.g. Hardware verification
  - We want to KNOW that the hardware computes what we think it does
  - Given the assumptions we make hold true
  - E.g. we might assume cosmic rays don’t affect how gates work

How to Prove Things in Logic

- We want absolute certainty
- Which is of course hard to get in general
- But Logic is well suited to providing
- Remember we even talked about some terms for certainty
  - “valid” and ”tautology” give certainty of truth
  - “unsatisfiable” gives certainty of falsity

Reminder: Some key terms

- A logic formula can be one or more of these
  - **Valid**
    - Every way of assigning the variables T/F makes the formula true
  - **Tautology**
    - Exactly the same as valid
  - **Invalid**
    - Not valid
    - There is at least one assignment which makes the formula untrue
  - **Satisfiable**
    - At least one assignment to T/F makes formula true
  - **Unsatisfiable**
    - No assignment can make the formula true

A Valid Argument

- “Valid” has another meaning as well as synonym for tautology
- An argument is VALID if the conclusion necessarily follows from the assumptions
- In terms of Propositional Logic
  - If we have a set of assumptions A1, A2, A3, …
    - Each a formula of propositional logic, maybe very complex
  - And we have a conclusion C
    - Again possibly complex
  - We have a valid argument that the conclusion follows from the assumptions iff it is impossible for all of A1, A2, A3, … to be true and C to be false

Valid Argument = Valid Formula

- We have a valid argument that the conclusion follows from the assumptions

  iff it is impossible for all of A1, A2, A3, … to be true and C to be false
- Which is true iff the following formula is valid (a tautology)

  $(A1 \land A2 \land A3 \land \ldots) \rightarrow C$

- Note the tight link between the meanings of valid

Reductio ad absurdum in general

- “Reduction to an absurdity”
    - Also called “proof by contradiction”
- If the argument is valid, then $(A1 \land A2 \land A3 \land \dots) \rightarrow C$ is a tautology
- The only way $(A1 \land A2 \land A3 \land \dots) \rightarrow C$ can be false is if all the A’s are true and C is false
- If this is impossible then $A1 \land A2 \land A3 \land \dots \land \neg C$ is unsatisfiable
- If we deduce falsity (an absurdity) then this must be unsatisfiable
- So one way to prove an argument valid is to prove falsity from the assumptions and the negation of the conclusion
- We’ll see this in DPLL later

Next time …

- How to prove things in Logic
- Reductio ad Absurdum
- DPLL Algorithm
- Solving the whodunnit!