Propositional Logic: Syntax and Semantics

Supratik Chakraborty
IIT Bombay

Notation

- Variables: $p, q, r, \ldots$
  - Represent *propositions* or *declarative statements*
- Constants: $\top, \bot$
- Operators or connectives:
  - $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$, (,)
    - We don't need all of them, but convenient to have them ...

Rules for constructing formulas

- Every variable constitutes a formula
- The constants $\top$ and $\bot$ are formulae.
- If φ is a formula, so are $\neg\varphi$ and (φ)
- If φ_1 and φ_2 are formulas, so is $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$, $\varphi_1 \rightarrow \varphi_2$, $\varphi_1 \leftrightarrow \varphi_2$

Every formula formed using above rules (only) is called *well-formed*.

Propositional formulas as strings and trees

- Alphabet (over which strings are constructed):
  - Set of variable names, e.g. $\{p_1, p_2, q_1, q_2, \ldots\}$
  - Set of constants $\{\top, \bot\}$
  - Fixed symbols $\{\neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$
- Well-formed formula: string formed according to rules on prev. slide
  - $(p_1 \vee \neg q_2) \wedge (\neg p_2 \rightarrow (q_1 \leftrightarrow \neg p_1))$
  - $p_1 \rightarrow (p_2 \rightarrow (p_3 \rightarrow p_4))$
- Well-formed formulas can be represented using trees
  - Consider the rules on prev. slide as telling how to construct a tree bottom-up
  - Parse trees: Obviate the need for '(' and ')'

Semantics of Propositional Logic

Consider a formula φ with n variables. Let 0 represent "false" and 1 represent "true"

- $[\![\varphi]\!] : \{0,1\}^n \rightarrow \{0,1\}$
- **Semantics is a function**
  - Often represented in tabular form: Truth Table
- Indicates truth value of formula, given truth values of all variables

Rules of semantics

- $[\![\neg\varphi]\!] = 1$ iff $[\![\varphi]\!] = 0$.
- $[\![\varphi_1 \wedge \varphi_2]\!] = 1$ iff $[\![\varphi_1]\!] = [\![\varphi_2]\!] = 1$.
- $[\![\varphi_1 \vee \varphi_2]\!] = 1$ iff at least one of $[\![\varphi_1]\!]$ or $[\![\varphi_2]\!]$ evaluates to 1.
- $[\![\varphi_1 \rightarrow \varphi_2]\!] = 1$ iff at least one of $[\![\varphi_1]\!] = 0$ or $[\![\varphi_2]\!] = 1$.
- $[\![\varphi_1 \leftrightarrow \varphi_2]\!] = 1$ iff both $[\![\varphi_1 \rightarrow \varphi_2]\!] = 1$ and $[\![\varphi_2 \rightarrow \varphi_1]\!] = 1$.

Semantics example

$[\![(p \vee s) \rightarrow (\neg q \leftrightarrow r)]\!]$

Semantics example

$[\![(p \vee s) \rightarrow (\neg q \leftrightarrow r)]\!]$

Write out the truth table:

p	q	r	s	$p \vee s$	$\neg q$	$\neg q \leftrightarrow r$	$(p \vee s) \rightarrow (\neg q \leftrightarrow r)$

Semantics example

$$[\![(p \vee s) \rightarrow (\neg q \leftrightarrow r)]\!]$$

Write out the truth table:

p	q	r	s	$p \vee s$	$\neg q$	$\neg q \leftrightarrow r$	$(p \vee s) \rightarrow (\neg q \leftrightarrow r)$
0	0	0	0	0	1	0	1

Semantics example

$[\![(p \vee s) \rightarrow (\neg q \leftrightarrow r)]\!]$

Write out the truth table:

p	q	r	s	$p \vee s$	$\neg q$	$\neg q \leftrightarrow r$	$(p \vee s) \rightarrow (\neg q \leftrightarrow r)$
0	0	0	0	0	1	0	1
0	0	0	1	1	1	0	0

Semantics example

$$[\![(p \lor s) \to (\neg q \leftrightarrow r)]\!]$$

Write out the truth table:

p	q	r	s	$p \lor s$	$\neg q$	$\neg q \leftrightarrow r$	$(p \lor s) \to (\neg q \leftrightarrow r)$
0	0	0	0	0	1	0	1
0	0	0	1	1	1	0	0
0	0	1	0	0	1	1	1
0	0	1	1	1	1	1	1
0	1	0	0	0	0	1	1
0	1	0	1	1	0	1	1
0	1	1	0	0	0	0	1
0	1	1	1	1	0	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	0
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	1
1	1	0	0	1	0	1	1
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	0
1	1	1	1	1	0	0	0

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \lor q$, $p \land q$

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \vee q$, $p \wedge q$
  - Both φ and $\neg\varphi$ may be satisifable.

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \vee q$, $p \wedge q$
  - Both φ and $\neg\varphi$ may be satisfiable.
- **unsatisfiable** or **contradiction** iff $[\![\varphi]\!] = 0$ for all assign of vars
  - E.g. $p \wedge \neg p$,

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \vee q$, $p \wedge q$
  - Both φ and $\neg\varphi$ may be satisifable.
- **unsatisfiable** or **contradiction** iff $[\![\varphi]\!] = 0$ for all assign of vars
  - E.g. $p \wedge \neg p$, $p \wedge ((p \rightarrow q) \wedge \neg q)$

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \vee q$, $p \wedge q$
  - Both φ and $\neg\varphi$ may be satisifable.
- **unsatisfiable** or **contradiction** iff $[\![\varphi]\!] = 0$ for all assign of vars
  - E.g. $p \wedge \neg p$, $p \wedge ((p \rightarrow q) \wedge \neg q)$
  - At most one of φ and $\neg\varphi$ can be unsatisfiable.

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \vee q$, $p \wedge q$
  - Both φ and $\neg\varphi$ may be satisifable.
- **unsatisfiable** or **contradiction** iff $[\![\varphi]\!] = 0$ for all assign of vars
  - E.g. $p \wedge \neg p$, $p \wedge ((p \rightarrow q) \wedge \neg q)$
  - At most one of φ and $\neg\varphi$ can be unsatisfiable.
- **valid** or **tautology** iff $[\![\varphi]\!] = 1$ for all assign of vars
  - E.g. $p \vee \neg p$,

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \lor q$, $p \land q$
  - Both φ and $\neg\varphi$ may be satisifable.
- **unsatisfiable** or **contradiction** iff $[\![\varphi]\!] = 0$ for all assign of vars
  - E.g. $p \land \neg p$, $p \land ((p \rightarrow q) \land \neg q)$
  - At most one of φ and $\neg\varphi$ can be unsatisfiable.
- **valid** or **tautology** iff $[\![\varphi]\!] = 1$ for all assign of vars
  - E.g. $p \lor \neg p$, $p \lor \neg(p \land q)$

Semantics-based terminology

A formula φ is

- **satisfiable** or **consistent** iff $[\![\varphi]\!] = 1$ for some assign of vars
  - E.g. $p \vee q$, $p \wedge q$
  - Both φ and $\neg\varphi$ may be satisifable.
- **unsatisfiable** or **contradiction** iff $[\![\varphi]\!] = 0$ for all assign of vars
  - E.g. $p \wedge \neg p$, $p \wedge ((p \rightarrow q) \wedge \neg q)$
  - At most one of φ and $\neg\varphi$ can be unsatisfiable.
- **valid** or **tautology** iff $[\![\varphi]\!] = 1$ for all assign of vars
  - E.g. $p \vee \neg p$, $p \vee \neg(p \wedge q)$
  - φ is valid iff $\neg\varphi$ is unsatisfiable.

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \vee q)$,

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \vee q)$, $\neg p \models (p \rightarrow q)$

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \lor q)$, $\neg p \models (p \rightarrow q)$
  - Denoted $\varphi \models \varphi_1$
  - Equivalently, $\varphi \rightarrow \varphi_1$ is valid

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \vee q)$, $\neg p \models (p \rightarrow q)$
  - Denoted $\varphi \models \varphi_1$
  - Equivalently, $\varphi \rightarrow \varphi_1$ is valid
- is **semantically equivalent** to φ_1 iff $\varphi \models \varphi_1$ and $\varphi_1 \models \varphi$
  - Identical truth tables, (obviously) an equivalence relation

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \vee q)$, $\neg p \models (p \rightarrow q)$
  - Denoted $\varphi \models \varphi_1$
  - Equivalently, $\varphi \rightarrow \varphi_1$ is valid
- is **semantically equivalent** to φ_1 iff $\varphi \models \varphi_1$ and $\varphi_1 \models \varphi$
  - Identical truth tables, (obviously) an equivalence relation
  - E.g. $p \rightarrow q$ and $\neg p \vee q$

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \vee q)$, $\neg p \models (p \rightarrow q)$
  - Denoted $\varphi \models \varphi_1$
  - Equivalently, $\varphi \rightarrow \varphi_1$ is valid
- is **semantically equivalent** to φ_1 iff $\varphi \models \varphi_1$ and $\varphi_1 \models \varphi$
  - Identical truth tables, (obviously) an equivalence relation
  - E.g. $p \rightarrow q$ and $\neg p \vee q$
  - Equivalently, $\varphi \leftrightarrow \varphi_1$ is valid

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \vee q)$, $\neg p \models (p \rightarrow q)$
  - Denoted $\varphi \models \varphi_1$
  - Equivalently, $\varphi \rightarrow \varphi_1$ is valid
- is **semantically equivalent** to φ_1 iff $\varphi \models \varphi_1$ and $\varphi_1 \models \varphi$
  - Identical truth tables, (obviously) an equivalence relation
  - E.g. $p \rightarrow q$ and $\neg p \vee q$
  - Equivalently, $\varphi \leftrightarrow \varphi_1$ is valid
- is **equisatisfiable to** φ_1 iff either both φ and φ_1 are satisfiable or both are unsatisfiable
  - E.g. $p \wedge q$ and $r \vee s$

Semantics-based terminology

A formula φ

- **semantically entails** φ_1 iff $[\![\varphi]\!] \preceq [\![\varphi_1]\!]$ for all assign of vars, where 0 (false) $\preceq$ 1 (true)
  - E.g. $p \models (p \vee q)$, $\neg p \models (p \rightarrow q)$
  - Denoted $\varphi \models \varphi_1$
  - Equivalently, $\varphi \rightarrow \varphi_1$ is valid
- is **semantically equivalent** to φ_1 iff $\varphi \models \varphi_1$ and $\varphi_1 \models \varphi$
  - Identical truth tables, (obviously) an equivalence relation
  - E.g. $p \rightarrow q$ and $\neg p \vee q$
  - Equivalently, $\varphi \leftrightarrow \varphi_1$ is valid
- is **equisatisfiable to** φ_1 iff either both φ and φ_1 are satisfiable or both are unsatisfiable
  - E.g. $p \wedge q$ and $r \vee s$
  - Semantic equivalence implies equisatisfiability, not vice versa

Fun with semantics!

Two syntactically different formulas may be semantically equivalent!

E.g. $\varphi_1 : p \rightarrow (q \rightarrow r)$, $\varphi_2 : (p \wedge q) \rightarrow r$, and $\varphi_3 : (q \wedge \neg r) \rightarrow \neg p$

Fun with semantics!

Two syntactically different formulas may be semantically equivalent!

E.g. $\varphi_1 : p \rightarrow (q \rightarrow r)$, $\varphi_2 : (p \wedge q) \rightarrow r$, and $\varphi_3 : (q \wedge \neg r) \rightarrow \neg p$

Truth table way of checking equivalence (or not):

p	q	r	$q \rightarrow r$	$p \wedge q$	$q \wedge \neg r$	$\neg p$	φ_1	φ_2	φ_3

Fun with semantics!

Two syntactically different formulas may be semantically equivalent!

E.g. $\varphi_1 : p \rightarrow (q \rightarrow r)$, $\varphi_2 : (p \wedge q) \rightarrow r$, and $\varphi_3 : (q \wedge \neg r) \rightarrow \neg p$

Truth table way of checking equivalence (or not):

p	q	r	$q \rightarrow r$	$p \wedge q$	$q \wedge \neg r$	$\neg p$	φ_1	φ_2	φ_3
0	0	0	1	0	0	1	1	1	1

Fun with semantics!

Two syntactically different formulas may be semantically equivalent!

E.g. $\varphi_1 : p \rightarrow (q \rightarrow r)$, $\varphi_2 : (p \wedge q) \rightarrow r$, and $\varphi_3 : (q \wedge \neg r) \rightarrow \neg p$

Truth table way of checking equivalence (or not):

p	q	r	$q \rightarrow r$	$p \wedge q$	$q \wedge \neg r$	$\neg p$	φ_1	φ_2	φ_3
0	0	0	1	0	0	1	1	1	1
0	0	1	1	0	0	1	1	1	1
0	1	0	0	0	1	1	1	1	1
0	1	1	1	0	0	1	1	1	1
1	0	0	1	0	0	0	1	1	1
1	0	1	1	0	0	0	1	1	1
1	1	0	0	1	1	0	0	0	0
1	1	1	1	1	0	0	1	1	1

Fun with semantics!

Two syntactically different formulas may be semantically equivalent!

E.g. $\varphi_1 : p \rightarrow (q \rightarrow r)$, $\varphi_2 : (p \wedge q) \rightarrow r$, and $\varphi_3 : (q \wedge \neg r) \rightarrow \neg p$

Truth table way of checking equivalence (or not):

p	q	r	$q \rightarrow r$	$p \wedge q$	$q \wedge \neg r$	$\neg p$	φ_1	φ_2	φ_3
0	0	0	1	0	0	1	1	1	1
0	0	1	1	0	0	1	1	1	1
0	1	0	0	0	1	1	1	1	1
0	1	1	1	0	0	1	1	1	1
1	0	0	1	0	0	0	1	1	1
1	0	1	1	0	0	0	1	1	1
1	1	0	0	1	1	0	0	0	0
1	1	1	1	1	0	0	1	1	1

Works, but doesn't scale! 2^n rows for n propositions

Semantic reasoning without truth tables?

Yes, **proof rules**