Propositional Logic

Cesare Tinelli

The University of Iowa

Copyright 2001-05 — Cesare Tinelli and Hantao Zhang.

These notes are copyrighted material and may not be used in other course settings outside of the University of Iowa in their current or modified form without the express written permission of the copyright holders.
Readings

- Chap. 7 of [Russell and Norvig, 2003]
A logic is a triple \( \langle \mathcal{L}, S, R \rangle \) where

- \( \mathcal{L} \), the logic’s **language**, is a class of sentences described by a formal grammar.

- \( S \), the logic’s **semantics** is a formal specification of how to assign *meaning* in the “real world” to the elements of \( \mathcal{L} \).

- \( R \), the logic’s **inference system**, is a set of formal derivation *rules* over \( \mathcal{L} \).

There are several logics: propositional, first-order, higher-order, modal, temporal, intuitionistic, linear, equational, non-monotonic, fuzzy, . . .

We will concentrate on **propositional logic** and **first-order logic**.
Propositional Logic

Each sentence is made of

- **propositional variables** \((A, B, \ldots, P, Q, \ldots)\)
- **logical constants** \((\text{True}, \text{False})\).
- **logical connectives** \((\land, \lor, \Rightarrow, \ldots)\).

Every propositional variable stands for a basic *fact*.

**Examples:** *I’m hungry*, *Apples are red*, *Joe and Jill are married.*
Propositional Logic

Ontological Commitments
Propositional Logic is about facts in the world that are either true or false, nothing else.

Semantics of Propositional Logic
Since each propositional variable stands for a fact about the world, its meaning ranges over the Boolean values \{True, False\}.

Note: Do not confuse True, False, which are values (i.e., semantical entities) here with True, False which are logical constants (i.e., symbols of the language).
Propositional Logic

The Language

- Each propositional variable \((A, B, \ldots, P, Q, \ldots)\) is a sentence.

- Each logical constant \((\text{True}, \text{False})\) is a sentence.

- If \(\varphi\) and \(\psi\) are sentences, all of the following are also sentences.

\[
(\varphi) \quad \neg \varphi \quad \varphi \land \psi \quad \varphi \lor \psi \quad \varphi \Rightarrow \psi \quad \varphi \Leftrightarrow \psi
\]

- Nothing else is a sentence.
The Language of Propositional Logic

More formally, it is the language generated by the following grammar.

Symbols:
- Propositional variables: \(A, B, \ldots, P, Q, \ldots\)
- Logical constants:
  - True (true)
  - False (false)
  - \(\wedge\) (and)
  - \(\lor\) (or)
  - \(\Rightarrow\) (implies)
  - \(\neg\) (not)
  - \(\equiv\) (equivalent)

Grammar Rules:

\[
\begin{align*}
\text{Sentence} & ::= \text{AtomicS} | \text{ComplexS} \\
\text{AtomicS} & ::= \text{True} | \text{False} | A | B | \ldots | P | Q | \ldots \\
\text{ComplexS} & ::= (\text{Sentence}) | \text{Sentence} \text{ Connective} \text{ Sentence} | \neg \text{Sentence} \\
\text{Connective} & ::= \wedge | \lor | \Rightarrow | \equiv
\end{align*}
\]
Semantics of Propositional Logic

The meaning (value) of True is always True. The meaning of False is always False.

The meaning of the other sentences depends on the meaning of the propositional variables.

- **Base cases:** Truth Tables

<table>
<thead>
<tr>
<th>$P$</th>
<th>$Q$</th>
<th>$\neg P$</th>
<th>$P \land Q$</th>
<th>$P \lor Q$</th>
<th>$P \Rightarrow Q$</th>
<th>$P \Leftrightarrow Q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
</tbody>
</table>

- **Non-base Cases:** Given by reduction to the base cases

Ex: the meaning of $(P \lor Q) \land R$ is the same as the meaning of $A \land R$ where $A$ has the same meaning as $P \lor Q$. 
The Meaning of Logical Connectives: A Warning

Disjunction

- $A \lor B$ is true when $A$ or $B$ or or both are true (inclusive or).
- $A \oplus B$ is sometimes used to mean “either $A$ or $B$ but not both” (exclusive or).

Implication

- $A \Rightarrow B$ does not require a causal connection between $A$ and $B$.
  Ex: Sky-is-blue $\Rightarrow$ Snow-is-white
- When $A$ is false, $A \Rightarrow B$ is always true regardless of the value of $B$.
  Ex: Two-equals-four $\Rightarrow$ Apples-are-red
- Beware of negations in implications.
  Ex: Is-a-female-bird $\Rightarrow$ Lays-eggs
  $\neg$Is-a-female-bird $\Rightarrow$ $\neg$Lays-eggs
An assignment of Boolean values to the propositional variables of a sentence is an interpretation of the sentence.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>H</td>
<td>P \lor H</td>
<td>(P \lor H) \land \neg H</td>
<td>((P \lor H) \land \neg H) \Rightarrow P</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
</tbody>
</table>

Interpretations:
\{ P \mapsto \text{False}, H \mapsto \text{False} \}, \{ P \mapsto \text{False}, H \mapsto \text{True} \}, \ldots

The semantics of Propositional logic is compositional: the meaning of a sentence is defined recursively in terms of the meaning of the sentence’s components.
Semantics of Propositional Logic

The meaning of a sentence in general depends on its interpretation. Some sentences, however, have always the same meaning.

<table>
<thead>
<tr>
<th>$P$</th>
<th>$H$</th>
<th>$P \lor H$</th>
<th>$(P \lor H) \land \neg H$</th>
<th>$((P \lor H) \land \neg H) \Rightarrow P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
</tbody>
</table>

A sentence is

- **satisfiable** if it is true in **some** interpretation,
- **valid** if it is true in **every** possible interpretation.
Entailment in Propositional Logic

Given
- a set \( \Gamma \) of sentences and
- a sentence \( \varphi \),

we write

\[
\Gamma \models \varphi
\]

iff every interpretation that makes all sentences in \( \Gamma \) true makes \( \varphi \) also true.

\( \Gamma \models \varphi \) is read as “\( \Gamma \) entails \( \varphi \)” or “\( \varphi \) logically follows from \( \Gamma \).”
Entailment in Propositional Logic: Examples

\[
\begin{align*}
\{A, A \Rightarrow B\} & \models B \\
\{A\} & \models A \lor B \\
\{A, B\} & \models A \land B \\
\{\} & \models A \lor \neg A \\
\{A\} & \not\models A \land B \\
\{A \lor \neg A\} & \not\models A
\end{align*}
\]

<table>
<thead>
<tr>
<th></th>
<th>(A)</th>
<th>(B)</th>
<th>(A \Rightarrow B)</th>
<th>(A \lor B)</th>
<th>(A \land B)</th>
<th>(A \lor \neg A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>2.</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>3.</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>4.</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
</tbody>
</table>

22c:145 Artificial Intelligence, Fall'05 – p.13/31
Properties of Entailment

- $\Gamma \models \varphi$, for all $\varphi \in \Gamma$ (inclusion property of PL)

- If $\Gamma \models \varphi$, then $\Gamma' \models \varphi$ for all $\Gamma' \supseteq \Gamma$ (monotonicity of PL)

- $\varphi$ is valid if $\text{True} \models \varphi$ (also written as $\models \varphi$)

- $\varphi$ is unsatisfiable if $\varphi \models \text{False}$

- $\Gamma \models \varphi$ if the set $\Gamma \cup \{\neg \varphi\}$ is unsatisfiable
Two sentences $\varphi_1$ and $\varphi_2$ are **logically equivalent**, written

$$\varphi_1 \equiv \varphi_2$$

if $\varphi_1 \models \varphi_2$ and $\varphi_2 \models \varphi_1$.

**Note:**
- $\varphi_1 \equiv \varphi_2$ if and only if every interpretation assigns the same Boolean value to $\varphi_1$ and $\varphi_2$.
- Implication and equivalence ($\Rightarrow, \Leftrightarrow$), which are **syntactical entities**, are intimately related to entailment and logical equivalence ($\models, \equiv$), which are **semantical notions**:

$$\varphi_1 \models \varphi_2 \quad \text{iff} \quad \models \varphi_1 \Rightarrow \varphi_2$$

$$\varphi_1 \equiv \varphi_2 \quad \text{iff} \quad \models \varphi_1 \Leftrightarrow \varphi_2$$
Properties of Logical Connectives

- ∧ and ∨ are *commutative*
  \[ \varphi_1 \land \varphi_2 \equiv \varphi_2 \land \varphi_1 \]
  \[ \varphi_1 \lor \varphi_2 \equiv \varphi_2 \lor \varphi_1 \]

- ∧ and ∨ are *associative*
  \[ \varphi_1 \land (\varphi_2 \land \varphi_3) \equiv (\varphi_1 \land \varphi_2) \land \varphi_3 \]
  \[ \varphi_1 \lor (\varphi_2 \lor \varphi_3) \equiv (\varphi_1 \lor \varphi_2) \lor \varphi_3 \]

- ∧ and ∨ are mutually *distributive*
  \[ \varphi_1 \land (\varphi_2 \lor \varphi_3) \equiv (\varphi_1 \land \varphi_2) \lor (\varphi_1 \land \varphi_3) \]
  \[ \varphi_1 \lor (\varphi_2 \land \varphi_3) \equiv (\varphi_1 \lor \varphi_2) \land (\varphi_1 \lor \varphi_3) \]

- ∧ and ∨ are related by ¬ (DeMorgan’s Laws)
  \[ \neg(\varphi_1 \land \varphi_2) \equiv \neg \varphi_1 \lor \neg \varphi_2 \]
  \[ \neg(\varphi_1 \lor \varphi_2) \equiv \neg \varphi_1 \land \neg \varphi_2 \]
Properties of Logical Connectives

∧, ⇒, and ⇔ are actually redundant:

\[ \varphi_1 \land \varphi_2 \equiv \neg (\neg \varphi_1 \lor \neg \varphi_2) \]
\[ \varphi_1 \Rightarrow \varphi_2 \equiv \neg \varphi_1 \lor \varphi_2 \]
\[ \varphi_1 \Leftrightarrow \varphi_2 \equiv (\varphi_1 \Rightarrow \varphi_2) \land (\varphi_2 \Rightarrow \varphi_1) \]

We keep them all mainly for convenience.

Exercise. Use the truth tables to verify all the logical equivalences seen so far.
An inference system $\mathcal{I}$ for PL is a procedure that given a set $\Gamma = \{\alpha_1, \ldots, \alpha_m\}$ of sentences and a sentence $\varphi$, may reply “yes”, “no”, or runs forever.

If $\mathcal{I}$ replies positively to input $(\Gamma, \varphi)$, we say that $\Gamma$ derives $\varphi$ in $\mathcal{I}$, and write

$$\Gamma \vdash_{\mathcal{I}} \varphi$$

Intuitively, $\mathcal{I}$ should be such that it replies “yes” on input $(\Gamma, \varphi)$ only if $\varphi$ is in fact entailed by $\Gamma$.

---

$a$ Or, $\mathcal{I}$ derives $\varphi$ from $\Gamma$, or, $\varphi$ derives from $\Gamma$ in $\mathcal{I}$.
All These Fancy Symbols!

Note:

- $A \land B \Rightarrow C$ is a sentence, a bunch of symbols manipulated by an inference system $\mathcal{I}$.

- $A \land B \models C$ is a mathematical abbreviation standing for the statement: “every interpretation that makes $A \land B$ true, makes $C$ also true.”

- $A \land B \vdash_{\mathcal{I}} C$ is a mathematical abbreviation standing for the statement: “$\mathcal{I}$ derives $C$ from $A \land B$”.

In other words,

- $\Rightarrow$ is a formal symbol of the logic, which is used by the inference system.

- $\models$ is a shorthand we use to talk about the meaning of formal sentences.

- $\vdash_{\mathcal{I}}$ is a shorthand we use to talk about the output of the inference system $\mathcal{I}$. 
All These Fancy Symbols!

The connective $\Rightarrow$ and the shorthand $\models$ are related as follows.
The sentence $\varphi_1 \Rightarrow \varphi_2$ is valid (always true) if and only if $\varphi_1 \models \varphi_2$.

Example: $A \Rightarrow (A \lor B)$ is valid and $A \models (A \lor B)$

<table>
<thead>
<tr>
<th></th>
<th>$A$</th>
<th>$B$</th>
<th>$A \lor B$</th>
<th>$A \Rightarrow (A \lor B)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>2.</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>3.</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>4.</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
</tbody>
</table>
All These Fancy Symbols!

The shorthands $\models$ and $\vdash_I$ are related as follows.

- A **sound** inference system can derive only sentences that logically follow from a given set of sentences:
  
  $$\text{if } \Gamma \vdash_I \varphi \text{ then } \Gamma \models \varphi.$$ 

- A **complete** inference system can derive all sentences that logically follow from a given set of sentences:
  
  $$\text{if } \Gamma \models \varphi \text{ then } \Gamma \vdash_I \varphi.$$
Inference in Propositional Logic

There are two (equivalent) classes of inference systems of Propositional Logic:

- one based on truth tables (TT)
- one based on derivation rules (R)

**Truth Tables** The inference system TT is specified as follows:

\[ \{ \alpha_1, \ldots, \alpha_m \} \vdash_{TT} \varphi \quad \text{iff} \quad \text{all the values in the truth table of} \quad (\alpha_1 \land \cdots \land \alpha_m) \Rightarrow \varphi \quad \text{are True}. \]
The truth-tables-based inference system is sound:

\[ \alpha_1, \ldots, \alpha_m \vdash_{TT} \varphi \]

implies

truth table of \((\alpha_1 \land \cdots \land \alpha_m) \Rightarrow \varphi\)

all true

implies

\((\alpha_1 \land \cdots \land \alpha_m) \Rightarrow \varphi\)

is valid

implies

\(\models (\alpha_1 \land \cdots \land \alpha_m) \Rightarrow \varphi\)

implies

\((\alpha_1 \land \cdots \land \alpha_m) \models \varphi\)

implies

\(\models \alpha_1, \ldots, \alpha_m \models \varphi\)

It is also complete (exercise: prove it).

Its time complexity is \(O(2^n)\) where \(n\) is the number of propositional variables in \(\alpha_1, \ldots, \alpha_m, \varphi\).

We cannot hope to do better in general because a related, simpler problem (determining the satisfiability of a sentence) is NP-complete.

However, really hard cases of propositional inference are somewhat rare in practice.
An inference system in Propositional Logic can also be specified as a set $\mathcal{R}$ of inference (or derivation) rules.

- Each rule is just a *pattern* premises/conclusion.

- A rule applies to $\Gamma$ and derives $\varphi$ if
  - some of the sentences in $\Gamma$ match with the premises of the rule and
  - $\varphi$ matches with the conclusion.

- A rule is *sound* if the set of its premises entails its conclusion.
Some Inference Rules

- **And-Introduction**
  \[
  \frac{\alpha \quad \beta}{\alpha \land \beta}
  \]

- **And-Elimination**
  \[
  \frac{\alpha \land \beta}{\alpha}
  \]
  \[
  \frac{\alpha \land \beta}{\beta}
  \]

- **Or-Introduction**
  \[
  \frac{\alpha}{\alpha \lor \beta}
  \]
  \[
  \frac{\alpha}{\beta \lor \alpha}
  \]
Some Inference Rules (cont’d)

- Implication-Elimination (aka Modus Ponens)

\[
\begin{align*}
\alpha &\Rightarrow \beta \\
\alpha &
\end{align*}
\]

\[
\beta
\]

- Unit Resolution

\[
\begin{align*}
\alpha \lor \beta &
\end{align*}
\]

\[
\begin{align*}
\neg \beta &
\end{align*}
\]

\[
\alpha
\]

- Resolution

\[
\begin{align*}
\alpha \lor \beta &
\end{align*}
\]

\[
\begin{align*}
\neg \beta \lor \gamma &
\end{align*}
\]

\[
\alpha \lor \gamma
\]

or, equivalently,

\[
\begin{align*}
\neg \alpha &\Rightarrow \beta, \quad \beta \Rightarrow \gamma
\end{align*}
\]

\[
\neg \alpha \Rightarrow \gamma
\]
Some Inference Rules (cont’d)

- **Double-Negation-Elimination**
  
  \[ \neg\neg \alpha \vdash \alpha \]

- **False-Introduction**
  
  \[ \alpha \land \neg \alpha \vdash \text{False} \]

- **False-Elimination**
  
  \[ \text{False} \vdash \beta \]
Inference by Proof

We say there is a proof of \( \varphi \) from \( \Gamma \) in an inference system \( \mathcal{R} \) if we can derive \( \varphi \) by applying the rules of \( \mathcal{R} \) repeatedly to \( \Gamma \) and its derived sentences.

Example: a proof of \( P \) from \( \{(P \lor H) \land \neg H\} \)

1. \((P \lor H) \land \neg H\) by assumption
2. \(P \lor H\) by \(\wedge\)-elimination applied to (1)
3. \(\neg H\) by \(\wedge\)-elimination applied to (1)
4. \(P\) by unit resolution applied to (2),(3)

We can represent a proof more visually as a proof tree:

Example:

\[
\begin{array}{c}
(P \lor H) \land \neg H \\
\hline
P \lor H \\
\hline
P \\
\end{array}
\]

\[
\begin{array}{c}
(P \lor H) \land \neg H \\
\hline
\neg H \\
\hline
P
\end{array}
\]
More precisely, there is a proof of $\varphi$ from $\Gamma$ in $\mathcal{R}$ if

1. $\varphi \in \Gamma$ or,
2. there is a rule in $\mathcal{R}$ that applies to $\Gamma$ and produces $\varphi$ or,
3. there is a proof of each $\varphi_1, \ldots, \varphi_m$ from $\Gamma$ in $\mathcal{R}$ and a rule that applies to $\{\varphi_1, \ldots, \varphi_m\}$ and produces $\varphi$.

Then, the inference system $\mathcal{R}$ is specified as follows:

$$\Gamma \vdash_{\mathcal{R}} \varphi \iff \text{there is a proof of } \varphi \text{ from } \Gamma \text{ in } \mathcal{R}$$
An Inference System $\mathcal{R}$

- $\alpha \land \beta \quad \alpha \Rightarrow \beta$
- $\alpha \lor \beta \quad \alpha \lor \beta$
- $\neg \alpha \quad \alpha \land \neg \alpha$
- $\beta \quad \neg \beta \lor \gamma$
- False
Soundness of $\mathcal{R}$

The given system $\mathcal{R}$ is sound because all of its rules are.

Example: the Resolution rule

\[
\frac{\alpha \lor \beta, \neg \beta \lor \gamma}{\alpha \lor \gamma}
\]

<table>
<thead>
<tr>
<th></th>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>$\gamma$</th>
<th>$\neg \beta$</th>
<th>$\alpha \lor \beta$</th>
<th>$\neg \beta \lor \gamma$</th>
<th>$\alpha \lor \gamma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>2.</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>3.</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>4.</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>5.</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>6.</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>7.</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>8.</td>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
</tbody>
</table>

All the interpretations that satisfy both $\alpha \lor \beta$ and $\neg \beta \lor \gamma$ (4, 5, 6, 8) satisfy $\alpha \lor \gamma$ as well.
Soundness of $\mathcal{R}$

The given system $\mathcal{R}$ is sound because all of its rules are.

Exercise: prove that the other inference rules are sound as well.

Is $\mathcal{R}$ also complete?