Logical Inference

COMPSCI 230 — Discrete Math

February 25, 2016
Logical Inference

1. Inference in Propositional Logic
   - Inference by Truth Table
   - Inference by Valid Inference Rules
   - Natural Inference
   - Jane’s Argument Revisited
Inference Example From Last Time

• If both Carl and Nathan come to the party, then Tom won’t: \((c \land n) \rightarrow \sim t\)

• If Heather comes to the party but neither Carl nor Samantha do, Peter will stay at home: 
\((h \land \sim(c \lor s)) \rightarrow \sim p\)

• Heather will be there, but Samantha is sick at home: 
\(h \land \sim s\)

• Premise:
\(\Phi \iff ((c \land n) \rightarrow \sim t) \land ((h \land \sim(c \lor s)) \rightarrow \sim p) \land (h \land \sim s)\)

• Jane’s conclusion: If both Nathan and Tom go to the party, then Peter stays at home: 
\(\psi \iff n \land t \rightarrow \sim p\)

• Is Jane’s inference \(\Phi \Rightarrow \psi\) valid? Is it an implication?

• Is \(\Phi \rightarrow \psi\) true for all values of \(c, n, t, h, s, p\)?
### Truth Table

\[
\Phi \Rightarrow \psi
\]

\[
[((c \land n) \rightarrow \sim t) \land ((h \land \sim (c \lor s)) \rightarrow \sim p) \land (h \land \sim s)] \rightarrow [n \land t \rightarrow \sim p]
\]

<table>
<thead>
<tr>
<th>(c)</th>
<th>(n)</th>
<th>(t)</th>
<th>(h)</th>
<th>(s)</th>
<th>(p)</th>
<th>(\alpha_1)</th>
<th>(\alpha_2)</th>
<th>(\omega_1)</th>
<th>(\omega_2)</th>
<th>(\tau_1)</th>
<th>(\tau_2)</th>
<th>(\phi)</th>
<th>(\psi)</th>
<th>(\phi \rightarrow \psi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
Proving Inferences by Truth Table

+ Always works in propositional logic
  $n$ atomic propositions, $2^n$ rows in the truth table
- The size of a proof by truth table is exponential in the number of propositions
- Does not help intuition
- There is a further problem...
No Truth Tables for Predicates

- $\forall x \ P(x)$ is equivalent to $P(T_1) \land P(T_2) \land \ldots$ ($T_i$ are the terms)
- $\forall x \ P(x)$ is generally an infinite conjunction
- $\exists x \ P(x)$ is equivalent to $P(T_1) \lor P(T_2) \lor \ldots$
- $\exists x \ P(x)$ is generally an infinite disjunction
- We cannot build truth tables in predicate logic
- Enter *valid inference rules*: Rules to transform predicates into other predicates while preserving truth
- To prove $\Phi \Rightarrow \psi$, transform $\Phi$ into $\psi$ by valid inference rules
- Crucial for predicate logic, can also be used for propositions
Inference by Valid Inference Rules

• Write the known facts $\Phi$ (use comma instead of $\land$)

$$(c \land n) \rightarrow \neg t \ , \ (h \land \neg(c \lor s)) \rightarrow \neg p \ , \ h \land \neg s$$

• Write the conclusion $\psi$

$$n \land t \rightarrow \neg p$$

• Inference rules are recipes to replace formulas with other formulas

• **Valid** inference rules preserve truth values

  They replace true formulas with other **true** formulas

• Keep applying valid inference rules starting with the known facts until the conclusion pops up!
Key Questions about Inference

• Q: How do we know if an inference rule is valid?
  • A: We prove it

• Q: How do we know which rules to apply to which formula?
  • A1: Try every rule on every formula (search)
  • A2: Luck (not recommended)
  • A3: Experience (recognize patterns)

• Q: Do I have enough inference rules?
  • A: Prove completeness of your inference system (hard)
Valid Inference

• If we know that formulas $\phi_1, \ldots, \phi_n$ are true
  ...meaning that $\phi_1 \land \ldots \land \phi_n$ is true
  and if we know that $\phi_1 \land \ldots \land \phi_n \Rightarrow \psi$
  then it is safe to add $\psi$ to what we know to be true

• The inference rule

\[
\begin{array}{c}
\phi_1 \\
\vdots \\
\phi_n
\end{array} \\
\hline
\psi
\]

is valid
Example of Valid Inference

• **Modus Ponens**

\[
\phi \rightarrow \psi \\
\phi \\
\hline
\psi
\]

• Proof of validity:

<table>
<thead>
<tr>
<th>$\phi$</th>
<th>$\psi$</th>
<th>$((\phi \rightarrow \psi) \land \phi) \rightarrow \psi$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 2 3</td>
</tr>
</tbody>
</table>

\[
((\phi \rightarrow \psi) \land \phi) \Rightarrow \psi
\]
Modus Ponens and Validity

- *Modus Ponens* says that if $\phi$ is true and if it is the case that if $\phi$ then $\psi$, then also $\psi$ is true.
- Validity says that *Modus Ponens* is always a correct inference.
- The conjunction of all the premises of the rule (above the line) implies the conclusion of the rule (below the line).
- “Valid” = “structurally correct.”
- The proof of validity justifies using *Modus Ponens* to replace some formulas with others.
Natural Inference

- Natural inference uses “customary” inference rules
- Some of these reflect typical reasoning patterns
- Examples:

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \rightarrow q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

- **Modus Ponens**
  \[
P \rightarrow Q \\
P \\
\hline 
Q
\]

- **Modus Tollens**
  \[
P \rightarrow Q \\
\sim Q \\
\hline 
\sim P
\]

- **Remove And**
  \[
P \land Q \\
\hline 
P
\]

- **Remove Double Not**
  \[
\sim \sim P \\
\hline 
P
\]

- **Add Or**
  \[
P \\
P \lor Q \\
\hline 
P
\]

- **Add Then**
  \[
Q \\
P \rightarrow Q \\
\hline 
P
\]
Reasoning by Cases

- May also need to *reason by cases*:
  Assume $P$ and see if the conclusion holds
  Then assume $\sim P$ and see if the conclusion holds
- If it does in both cases, then it always does
- Not obvious what to assume in what order
- Again, systematic search or experience
Example: Jane’s Inference

(1): \((c \land n) \rightarrow \sim t\)
(2): \((h \land \sim c \land \sim s) \rightarrow \sim p\)
(3): \(h \land \sim s\)

(\infty): n \land t \rightarrow \sim p

- Rewrote (2) with all “and” on the left for uniformity
- (3) and the premise of (2) differ by \(\sim c\)
  and \(c\) also appears in (1) but not in (\(\infty\))
- Perhaps reason by cases on \(c\)
  This gets us started with (2)
  and brings (1) a bit closer to (\(\infty\))
- Seems like a good move
Tentatively Assume $\sim c$

(1): $(c \land n) \rightarrow \sim t$
(2): $(h \land \sim c \land \sim s) \rightarrow \sim p$
(3): $h \land \sim s$
(4): $\sim c$
(5): $\sim p$

- **Modus ponens** on (3) $\land$ (4) $\land$ (2) yields $\sim p$
- **Add Then** on (5) yields $(\infty)$
- So if $\sim c$ then $(\infty)$ holds: Done
Tentatively Assume \( c \)

(1): \((c \land n) \rightarrow \neg t\)

(2): \((h \land \neg c \land \neg s) \rightarrow \neg p\)

(3): \(h \land \neg s\)

(4): \(c\)

• (2) is no longer useful, as its premise is now false

• Because of (4), the premise of (1) is equivalent to \( n \)

• Reason by cases on \( t \): If \( \neg t \), then the premise of \((\infty)\) is false, so \((\infty)\) is true

• If \( t \), then Modus tollens on (1) yields \( \neg n \), so again the premise of \((\infty)\) is false, and \((\infty)\) is true
Jane was Right

• We reasoned by cases
• No alternatives left
• In each case, Jane’s conclusion was true
• $\Phi \rightarrow \psi$ is a tautology, $\Phi \Rightarrow \psi$
• We inferred $\psi$ from $\Phi$ through valid inference rules and case-based reasoning
• Jane was right
Truth Tables versus Natural Inference

- Natural inference proof is smaller than truth table
- Devising the proof is harder:
  We need to figure out what rules to apply when and the right propositions for case-based reasoning
- A systematic search over all the rules, formulas, and cases is not necessarily simpler than a truth table
- Inference is all we can do in predicate logic, where truth tables are not possible
- There is a vast literature on making predicate-logic inference (more) efficient
- The programming language Prolog is based on Robinson’s one-rule inference method
- See optional readings for more
Interpreting Natural Inference

- In contrast with truth tables, we can translate our inference steps back into English
- Assume that Carl does not go to the party
- Then since Heather is coming but Samantha is not, Peter won’t attend, not to be left alone with Heather in Carl’s and Samantha’s absence
- So in this case Jane is right
- Now assume that Carl does go to the party
- Then, since Tom doesn’t want to be with both Nathan and Carl, Nathan and Tom cannot both be present
- So if they are both there, it means that Carl isn’t, and we are back to the first case
- So Jane is right