3 Correct and exhaustive reasoning

Is an inference rule "correct"?

Is an inference system "exhaustive"?

Is an inference system "self consistent"?

What do the represented facts/conclusions "mean"?

3.1 Value assignments

Assign *values* to all *primitive* propositions
Specifies true "state of affairs"

E.g. truth theory

assign	$v(p) = t$	true
or	$v(p) = f$	false

for all primitive propositions p
(specifies complete state of affairs)

E.g. relevance theory

	$v(p) = \{t\}$	true
or	$v(p) = \{f\}$	false
or	$v(p) = \{t, f\}$	evidence for both
or	$v(p) = \{\}$	evidence for neither

E.g. 3-valued logic

	$v(p) = \{t\}$	true
or	$v(p) = \{f\}$	false
or	$v(p) = *$	undeterminable

All possible primitive assignments = all possible states of affairs

3.2 Evaluating compound propositions

Compositional semantics

> value of composite depends only on *values* of components
> (value functional semantics)

E.g. for truth theory

$$v(\neg\alpha) = \begin{cases} t & \text{if } v(\alpha) = f \\ f & \text{if } v(\alpha) = t \end{cases}$$

$$v(\alpha \wedge \beta) = \begin{cases} t & \text{if } v(\alpha) = t \text{ and } v(\beta) = t \\ f & \text{otherwise} \end{cases}$$

$$v(\alpha \vee \beta) = \begin{cases} t & \text{if } v(\alpha) = t \text{ or } v(\beta) = t \\ f & \text{otherwise} \end{cases}$$

$$v(\alpha \rightarrow \beta) = \begin{cases} f & \text{if } v(\alpha) = t \text{ and } v(\beta) = f \\ t & \text{otherwise} \end{cases}$$

Recursively evaluate compound propositions
– bottoms out in values of primitive propositions

Given a complete assignment to primitive propositions
– can evaluate every compound proposition

3.3 What do propositions *mean*?

A restriction on the possible state of affairs

Asserting α means that the state of affairs v makes α evaluate to t; i.e. $v(\alpha) = t$.

v is a state of affairs (truth assignment)

Terminology

- v *satisfies* α if $v(\alpha) = t$
- v *falsifies* α if $v(\alpha) = f$
- α is *satisfiable* if exists v such that $v(\alpha) = t$
- α is *falisfiable* if exists v such that $v(\alpha) = f$
- α is *unsatisfiable* (or *inconsistent*) if $v(\alpha) = f$ for all v
- α is *unfalsifiable* (or *valid*) if $v(\alpha) = t$ for all v
- α *entails* β if every v that makes α evaluate to t, makes β evaluate to t as well. Written $\alpha \models \beta$.

3.4 Do not confuse entailment with derivability!

Entailment $\alpha \models \beta$ and derivability $\alpha \vdash \beta$ are two completely different mathematical systems.

We now want to *relate* derivability and entailment

Assume truth value theory defines correctness

Want inference system to implement entailment

Correct inference

If $A \vdash \gamma$ then $A \models \gamma$ (also called *soundness*)

Exhaustive inference

If $A \models \gamma$ then $A \vdash \gamma$ (also called *completeness*)

3.5 Resolution rule is correct w.r.t. truth assignments

$$\frac{\alpha \vee \neg p,\ \beta \vee p}{\alpha \vee \beta}$$

Proof

Assume antecedent satisfied; that is, $v(\alpha \vee \neg p) = t$ and $v(\beta \vee p) = t$. Two cases.

Case 1, if $v(p) = t$. Then $v(\alpha) = v(\alpha \vee \neg p)$ and hence $v(\alpha \vee \neg p) = v(\alpha) = t$. This implies $v(\alpha \vee \beta) = t$.

Case 2, if $v(p) = f$. Then $v(\beta) = v(\beta \vee p)$ and hence $v(\beta \vee p) = v(\beta) = t$. This implies $v(\alpha \vee \beta) = t$.

Therefore, in both cases $v(\alpha \vee \beta) = t$. ∎

3.6 Resolution rule is not exhaustive w.r.t. truth assignments

E.g. $\top \models \neg p \vee p$ (i.e. $\neg p \vee p$ is valid)
but $\top \not\vdash \neg p \vee p$ using resolution

Note: Natural deduction system *is* correct and exhaustive with respect to truth assignments. For a proof see: D. E. Cohen (1987) Computability and Logic. Ellis Horwood. (Chapter 11)

3.7 Resolution rule *is* exhaustive w.r.t. deriving contradictions

If A is unsatisfiable (inconsistent) then resolution can derive contradiction (that is $A \vdash \bot$)

Proof

Let $A = \{\alpha_1, ..., \alpha_n\}$ (α's in strict clausal form). Let K be the number of "excess" propositional letters in A; defined as the sum over $\alpha_i \in A$ of the number of additional propositional letters in each α_i, not including the first letter if there is one. That is, for a given proposition α, $excess(\alpha) = 0$ if α has no propositional letters, and $excess(\alpha) = m - 1$ if α has m propositional letters for $m > 0$. Similarly, for a set of propositions A, $excess(A) = \sum_{\alpha_i \in A} excess(\alpha_i)$.

Proof is by induction on K, the number of excess propositional letters.

__Base case__: $K = 0$. A contains clauses only of the form $\top$, $\bot$, $\neg p$ or p.

Case 1: A contains $\bot$, done.

Case 2: A contains opposing pair $\neg p$, p. Then can derive $\bot$ by resolution rule, done.

Case 3: A contains no opposing pair nor $\bot$. Then one can satisfy all the clauses in A by assigning $v(p) = t$ if $p \in A$, and $v(p) = f$ if $\neg p \in A$. This makes every clause evaluate to t, which contradicts the assumption that A is unsatisfiable; done.

__*Induction hypothesis*__: Assume the theorem holds for $K - 1$ or fewer excess propositional letters

__*Induction step*__: Assume A is unsatisfiable and has K excess propositional letters for $K > 0$. We are going to show $A \vdash \bot$ in two steps by considering strengthenings of A that have fewer propositional letters than A and therefore fall under the induction hypothesis.

First notice that, since $K > 0$, there must be at least one clause $\alpha \in A$ that has two or more propositional letters. Without loss of generality, assume α is of the form $p \vee \beta$ (the case $\neg p \vee \beta$ is similar). Consider the strengthening $\alpha' = \beta$ and let $A' = A - \{\alpha\} \cup \{\alpha'\}$. Note that α' is a strengthening of α, and hence A' is a strengthening of A (since $\alpha' \models \alpha$ and therefore $A' \models A$). There are three immediate consequences: first, $excess(A') = excess(A) - 1$ by construction; second, A' is unsatisfiable since A is unsatisfiable by assumption and we have that $A' \models A$. Therefore, overall we obtain $A' \vdash \bot$ by the induction hypothesis.

Now consider two cases.

Case 1: $A' - \{\alpha'\} \vdash \bot$. But this immediately implies $A - \{\alpha\} \vdash \bot$ since $A' - \{\alpha'\} = A - \{\alpha\}$, so we are done.

Case 2: $A' - \{\alpha'\} \not\vdash \bot$. But here we know that $A' \vdash \bot$ by the induction hypothesis. Now consider a derivation of $\bot$ from A'. Since the only difference between A and A' is that the clause $\alpha = p \vee \beta$ has been changed to $\alpha' = \beta$ we must have that $A \vdash p$. (Just take the derivation of $\bot$ from A' and add a 'p' to the side of every resolution step that involves the clause $\alpha' = p \vee \beta$.) So we have established the first key consequence that $A \vdash p$.

Now we only need to show that $A \cup \{p\} \vdash \bot$ and we will be done. To complete the proof, we follow a similar argument to the above. Consider the strengthening $\alpha'' = p$ and let $A'' = A - \{\alpha\} \cup \{\alpha''\}$. Now note both that α'' is a strengthening of α and A'' is a strengthening of A (since $\alpha'' \models \alpha$ and hence $A'' \models A$). Thus once again we obtain three immediate consequences: first, $excess(A'') \leq excess(A) - 1$ since α had at least two propositional letters and α'' only has one; second, A'' is unsatisfiable since A is unsatisfiable by assumption and $A'' \models A$. Therefore, overall we obtain $A'' \vdash \perp$ by the induction hypothesis.

To finish Case 2, notice that the last two paragraphs have established that $A \vdash p$ and $A \cup \{p\} \vdash \perp$ respectively, which allows us to conclude $A \vdash \perp$, and we are done.

$\blacksquare$

3.8 Readings

Burris, Chapter 2.
Dean, Allen, Aloimonos, Chapter 3.
Russell and Norvig 2nd Ed., Chapters 7 and 9*.
Genesereth and Nilsson, Chapters 3* and 4*.
* - ignoring material on first order variables and quantification