National University of Singapore
School of Computing
CS3243 Introduction to AI

Tutorial 7: Logical Agents I (Solutions)

Please refer to **Appendix A** for notes on Knowledge Bases, and **Appendix B** for Propositional Logic Laws.

1. Verify the following logical equivalences. Cite the equivalence law used with each step of your working (refer to Appendix B for a list of these laws).

   (a) $\neg(p \lor \neg q) \lor (\neg p \land \neg q) \equiv \neg p$

   (b) $(p \land \neg(\neg p \lor q)) \lor (p \land q) \equiv p$

   Solution:

   (a)
   $$\begin{aligned} \neg(p \lor \neg q) \lor (\neg p \land \neg q) &\equiv (\neg p \land q) \lor (\neg p \land \neg q) && \text{(de Morgan's law)} \\ &\equiv \neg p \land (q \lor \neg q) && \text{(distributive law)} \\ &\equiv \neg p \land \mathbf{true} && \text{(complement law)} \\ &\equiv \neg p && \text{(identity law)} \end{aligned}$$

   (b)
   $$\begin{aligned} (p \land \neg(\neg p \lor q)) \lor (p \land q) &\equiv (p \land (p \land \neg q)) \lor (p \land q) && \text{(de Morgan's law)} \\ &\equiv ((p \land p) \land \neg q) \lor (p \land q) && \text{(associative law)} \\ &\equiv (p \land \neg q) \lor (p \land q) && \text{(idempotent law)} \\ &\equiv p \land (\neg q \lor q) && \text{(distributive law)} \\ &\equiv p \land \mathbf{true} && \text{(complement law)} \\ &\equiv p && \text{(identity law)} \end{aligned}$$

2. Victor would like to invite three friends, Alice, Ben, and Cindy to a party, but must satisfy the following constraints:

   (a) Cindy comes only if Alice does not come.

   (b) Alice comes if either Ben or Cindy (or both) comes.

   (c) Cindy comes if Ben does not come.

   Victor would like to know who will come to the party, and who will not. Help Victor by expressing each of the above three constraints in propositional logic, and then, using these constraints, determine who will attend his party.

   Solution: Let x represent Cindy coming, y represent Alice coming, and z represent Ben coming. The constraints can be represented as follows.

   (a) $x \Rightarrow \neg y$

   (b) $(z \vee x) \Rightarrow y \equiv (z \Rightarrow y) \wedge (x \Rightarrow y)$. Hence, we have:

      i. $z \Rightarrow y$

      ii. $x \Rightarrow y$

   (c) $\neg z \Rightarrow y$

   From (a) and (b)(ii), we have:

$$\begin{aligned}
(x \Rightarrow \neg y) \wedge (x \Rightarrow y) &\equiv (\neg x \vee \neg y) \wedge (\neg x \vee y) && \text{(implication law)} \\
&\equiv \neg x \vee (\neg y \wedge y) && \text{(distributive law)} \\
&\equiv \neg x \vee \mathbf{false} && \text{(negation law)} \\
&\equiv \neg x && \text{(identity law)}
\end{aligned}$$

   Since we know $\neg x$, and by the contrapositive of (c) we have $\neg x \Rightarrow z$, therefore we have $\neg x \wedge (\neg x \Rightarrow z) \implies z$. Finally, from (b)(i), we also know $z \wedge (z \Rightarrow y) \implies y$. Hence, Alice and Ben will come to Victor's party. However, Cindy will not.

3. Consider the following knowledge base.

   "All firetrucks are red"

   "All firetrucks are cars"

   "All cars have four wheels"

   (a) Assume that an inference algorithm, A_1, that takes the query sentence "a ferrari is a red car" and infers "a ferrari is a firetruck". Determine which of the following properties **does not** apply to A_1.

   Option 1: complete
   Option 2: sound
   Option 3: both of the above

   Solution: **Option 2** is the correct answer. As the inferred statement is not entailed by the knowledge base the algorithm cannot be sound. It could still be complete, as the question doesn't state if the algorithm inferred anything else.

   (b) Assume that an inference algorithm, A_2 is given the query sentence "a ferrari is a red car". Determine which of the following properties **would guarantee** that A_2 would infer the sentence "a ferrari has four wheels".

   Option 1: completeness
   Option 2: soundness
   Option 3: both of the above need to be combined

   Solution: **Option 1** is the correct answer. Completeness ensures that everything that is entailed by the knowledge base will be inferred, as the statement is entailed by the KB completeness guarantees its inference.

   (c) Determine if the following statement is True or False.

   "Two agents with the same knowledge base and different inference engines, both of which are complete and sound, always behave in the same way".

   Justify your answer.

Solution: False. The behavior of an agent describes its interaction with the environment. Different complete and sound algorithms will (given the same KB) result with the same inferences and a difference in how they derive that doesn't imply a difference in behavior.

However, the behavior of an agent is not only dependent on its knowledge to different agents might have completely different objectives and hence act differently even with the same knowledge.

4. Given the following logical statements, use truth-table enumeration to show that $KB \models \alpha$. In other words, write down all possible true/false assignments to the variables, the ones for which KB is true and the one for which α is true, and see whether one is a subset of the other.

(a)

$$\begin{aligned} KB =& (x_1 \lor x_2) \land (x_1 \Rightarrow x_3) \land \neg x_2 \\ \alpha =& x_3 \lor x_2 \end{aligned}$$

(b)

$$\begin{aligned} KB =& (x_1 \lor x_3) \land (x_1 \Rightarrow \neg x_2) \\ \alpha =& \neg x_2 \end{aligned}$$

Solution: For the first KB and α pair we have:

x_1	x_2	x_3	KB	α
$True$	$True$	$True$	$False$	$True$
$True$	$True$	$False$	$False$	$True$
$True$	$False$	$True$	$True$	$True$
$True$	$False$	$False$	$False$	$False$
$False$	$True$	$True$	$False$	$True$
$False$	$True$	$False$	$False$	$True$
$False$	$False$	$True$	$False$	$True$
$False$	$False$	$False$	$False$	$False$

Note that $\alpha = x_2 \vee x_3$ is false iff both x_2 and x_3 are false (rows 4 and 8 in the table above), and that KB evaluates to false in those cases as well; moreover, when $KB = \mathit{True}$ (row 3), α is true as well. Thus $KB \models \alpha$.

For the next KB and α pair, we have:

x_1	x_2	x_3	KB	α
True	*True*	*True*	*False*	*False*
True	*True*	*False*	*False*	*False*
True	*False*	*True*	*True*	*True*
True	*False*	*False*	*True*	*True*
False	*True*	*True*	*True*	*False*
False	*True*	*False*	*False*	*False*
False	*False*	*True*	*True*	*True*
False	*False*	*False*	*False*	*True*

In order to show that $KB \models \alpha$ we need to show that whenever KB is true, so is α. However this is not the case: note that in line 5 $KB = \mathit{True}$ but $\alpha = \mathit{False}$.

Appendix A: Notes on Knowledge Bases

A knowledge base KB is a set of logical rules that model what the agent knows. These rules are written using a certain language (or *syntax*) and use a certain truth model (or *semantics* which say when a certain statement is true or false). In propositional logic sentences are defined as follows

1. Atomic Boolean variables are sentences.
2. If S is a sentence, then so is $\neg S$.
3. If S_1 and S_2 are sentences, then so is:
   (a) $S_1 \wedge S_2$ "S_1 and S_2"
   (b) $S_1 \vee S_2$ "S_1 or S_2"
   (c) $S_1 \Rightarrow S_2$ "S_1 implies S_2"
   (d) $S_1 \Leftrightarrow S_2$ "S_1 holds if and only if S_2 holds"

We say that a logical statement a models b ($a \models b$) if b holds whenever a holds. In other words, if $M(q)$ is the set of all value assignments to variables in a for which a holds true, then $M(a) \subseteq M(b)$.

An inference algorithm $\mathcal{A}$ is one that takes as input a knowledge base KB and a query α and decides whether α is derived from KB, written as $KB \vdash_{\mathcal{A}} \alpha$. $\mathcal{A}$ is sound if $KB \vdash_{\mathcal{A}} \alpha$ implies that $KB \models \alpha$; $\mathcal{A}$ is complete if $KB \models \alpha$ implies that $KB \vdash_{\mathcal{A}} \alpha$.

Appendix B: Propositional Logic Laws

De Morgan's Laws	$\neg(p \vee q) \equiv \neg p \wedge \neg q$	$\neg(p \wedge q) \equiv \neg p \vee \neg q$
Idempotent laws	$p \vee p \equiv p$	$p \wedge p \equiv p$
Associative laws	$(p \vee q) \vee r \equiv p \vee (q \vee r)$	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
Commutative laws	$p \vee q \equiv q \vee p$	$p \wedge q \equiv q \wedge p$
Distributive laws	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
Identity laws	$p \vee False \equiv p$	$p \wedge True \equiv p$
Domination laws	$p \wedge False \equiv False$	$p \vee True \equiv True$
Double negation law	$\neg\neg p \equiv p$	
Complement laws	$p \wedge \neg p \equiv False \wedge \neg True \equiv False$	$p \vee \neg p \equiv True \vee \neg False \equiv True$
Absorption laws	$p \vee (p \wedge q) \equiv p$	$p \wedge (p \vee q) \equiv p$
Conditional identities	$p \Rightarrow q \equiv \neg p \vee q$	$p \Leftrightarrow q \equiv (p \Rightarrow q) \wedge (q \Rightarrow p)$