Grundlagen der Künstlichen Intelligenz

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel
M. Krawez, T. Schulte
Sommersemester 2018

Universität Freiburg
Institut für Informatik

Übungsblatt 1 — Lösungen

Aufgabe 1.1 (Möglichkeiten und Grenzen der KI)

Recherchieren Sie in der KI-Literatur bzw. im Internet, inwiefern folgende Probleme heutzutage mittels Computer- bzw. Robotereinsatz gelöst werden können:

(a) Spielen der Brettspiele Dame und Go.

Lösung:

Checkers is completely solved, optimal game guarantees a tie.

The game of Go has been regarded as a benchmark for AI for a long time. Since the number of possible games is much larger than, e.g., in chess, it often was claimed that the game requires human-like creativity. In 2016, Google's AlphaGo algorithm has beaten Lee Sedol, a world top Go player.

`http://en.wikipedia.org/wiki/Computer_Go`.

(b) Verarbeiten natürlicher Sprache in Echtzeit.

Lösung:

Natural language processing is still an open field of research. A major milestone recently was IBM's Watson computer, which was able to win the first place in the popular quiz show Jeopardy! after competing against former human winners. This involved understanding questions posed in natural language and searching for the answer through a large database of information.

`http://www.research.ibm.com/deepqa/deepqa.shtml`

(c) Autonomie unbemannter Fahr- und Flugzeuge (UGVs und UAVs).

Lösung:

In the context of the „DARPA Grand Challenge“ in 2005 the team of Sebastian Thrun from the Stanford University reached the goal of autonomous navigation of a vehicle through the desert at a route of 200 km.

By 2016, Google's Waymo cars have driven autonomously for over 2.7 millions kilometers on public roads, with only few accidents caused by the self-driving car.

More information:

`http://en.wikipedia.org/wiki/DARPA_Grand_Challenge` and
`http://en.wikipedia.org/wiki/Google_driverless_car`.

Autonomous aircrafts are comparably easy to construct:

http://www.ida.liu.se/ext/witas/ or
http://en.wikipedia.org/wiki/Unmanned_aerial_vehicle.

(d) Automatische Gesichtserkennung.

Lösung:

In a crowd the rate of correctly recognized people is around 80 percent. In well-controlled settings this rate can be high enough to be used for, e.g., access control in banks, in military or scientific facilities. In these use cases the person have to face the camera frontally and don't move for some time.

(e) Spielen von Computerspielen (z.B. klassische Atari-Spiele) wie ein Mensch.

Lösung:

The performance is of course dependent on the game at hand, but a big number of games can be approached by deep learning:

http://www.nature.com/nature/journal/v518/n7540/full/nature14236.html

(f) Komponieren von Musik.

Lösung:

Several approaches to automatic music generation exist, two most prominent algorithms are:

EMI

In a training phase the algorithm analyses several music pieces, then it tries to produce a new piece following the style of the input pieces.

http://www.computerhistory.org/atchm/algorithmic-music-david-cope-and-emi

Melomics

Instead of copying a given style, Melomics generates a new music piece from some seed value, following general rules of music composition.

http://geb.uma.es/melomics

(g) Turing-Test

Lösung:

Chat bots (programs which simulate a conversation by answering to user input) have a long history and have grown quite sophisticated in recent years. They typically use a large online data base, which is extended by analyzing chats with users. However, those bots are still far away from passing the Turing test, mainly because they cannot well understand the semantics and context of a conversation.

Schreiben Sie Ihre Erkenntnisse in jeweils 2–3 Sätzen auf.

Aufgabe 1.2 (Performanz und Nutzen)

(a) Was ist der Unterschied zwischen einer Performanzmessung und einer Nutzenfunktion?

Lösung:

Ein Performanzmessung bewertet die Leistung eines Agenten von außen, quasi aus der Sicht einer objektiven externen Instanz, während eine Nutzenfunktion dem Agenten selbst erlaubt, seine möglichen Aktionen zu bewerten.

(b) Beschreiben Sie den Zusammenhang zwischen Performanzmessung und Nutzenfunktion bei einem lernenden Agenten.

Lösung:

Ein lernender Agent verändert seine utility function auf Grund der Rückmeldung des „Kritikers“, die wiederum von der Performanzmessung abhängt.

Aufgabe 1.3 (Rationale Agenten)

(a) Geben Sie für jeden der folgenden Agenten eine Beschreibung des Performanz-Maßes, der Umgebung, der Aktuatoren und der Sensoren (**P**erformance **E**nvironment **A**ctuators **S**ensors) an:

(i) Monopoly spielen

(ii) Leichtathlet beim Weitsprung

(iii) 2048 spielen (`http://gabrielecirulli.github.io/2048`)

Lösung:

Agent	P	E	A	S
Monopoly	Besitz nach Spielende	Spielbrett, andere Spieler	Würfeln, Grundstücke kaufen, ...	Würfelaugen, Handlungen anderer Spieler, ...
Weitsprung	gesprungene Weite in Meter	Sportstadion	Muskeln des Athleten	Augen des Athleten, Schiedsrichterentscheidung (übertreten, Weite)
2048	höchstes Tile, Highscore	4 x 4 grid	4 Bewegungen	Zustand des Spiels

(b) Klassifizieren Sie die in (a) formalisierten Umgebungen der Agenten nach folgenden Kriterien:

- vollständig beobachtbar oder teilweise beobachtbar

- deterministisch oder stochastisch
- statisch oder dynamisch
- diskret oder kontinuierlich

Lösung:

- Monopoly: teilweise beobachtbar, stochastisch, statisch, diskret
- Weitsprung: vollständig beobachtbar, stochastisch, statisch (abgesehen vom Wind), kontinuierlich
- 2048: vollständig beobachtbar, stochastisch, statisch, diskret

Aufgabe 1.4 (Problemformalisierung)
Geben Sie für die folgenden Problemstellungen jeweils eine möglichst präzise Formulierung an, die aus Anfangszustand, Zustandsraum, Aktionen, Zieltest und einer Pfadkostenfunktion besteht:

- Sie wollen den Rubiks Zauberwürfel lösen.
  `http://de.wikipedia.org/wiki/Zauberw%C3%BCrfel`

  Lösung:

  Zustände: Die Zustände sind $9 \cdot 6 = 54$-Tupel der Farbe der Seitenflächen.

  Anfangszustand: "zufällige" Konfiguration der Farben

  Zieltest: Haben alle Seitenflächen eine einheitliche Farbe?

  Aktionen: Drehen einer der Seitenflächen um 90° (Oben, unten, links, rechts, vorne, hinten) eventuell auch Drehungen um 180° sowie der Mittelfläche

  Pfadkosten: Je nach Metrik und verfügbaren Aktionen 1 pro Aktion (oder 2 pro 180° Drehung)

- Sie wollen eine Karte von Europa mit nur vier Farben einfärben. Damit man die Grenzen einzelner Länder erkennen kann, ist es erforderlich, dass es keine Nachbarländer mit gleicher Einfärbung gibt.

  Lösung:

  Zustände: Die Zustände sind 49-Tupel der Farben *rot*, *grün*, *blau*, *gelb* oder *ungefärbt* (z.B.). Jeder Eintrag repräsentiert ein Land auf der Karte. Der Kompaktheit halber kann man die Farben mit den Zahlen 0 (*rot*) bis 4 (*ungefärbt*) kodieren.

  Anfangszustand: Alle Länder sind ungefärbt / alle Länder sind rot / jedes Land kann eine der 4+1=5 Farben [ungefärbt, r,g,b,y] haben.

  Zieltest: Alle Länder sind gefärbt, benachbarte Länder haben unterschiedliche Farben?

  Aktionen: Einem Land eine der 4 Farben zuweisen und vorher überprüfen ob ein Nachbarland schon die gleiche Farbe hat / ein Land umfärben, wenn ein Konflikt festgestellt wird.

  Pfadkosten: Jede Aktion verursacht Einheitskosten.

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel, Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

Exercise Sheet 2 — Solutions

Exercise 2.1 (Formalizing problems)

Formalize the following problems as precisely as possible, by defining the initial state, the state space, the set of actions, the goal test and the path cost function:

- You want to solve Rubik's Cube.
  `http://en.wikipedia.org/wiki/Rubik%27s_Cube`

  Solution:

  A Rubik's Cube consists of 26 unique miniature cubes that can be moved to different locations on the cubes six faces by some rotation mechanic. The miniature cube's face plates are colored in one of six colors.

  state space: The states are $9 \cdot 6 = 54$-tuples representing the different color configurations (9 face plates per side, 6 sides).

  initial state: An arbitrary color configuration.

  goal test: Are all cube faces uniformly colored?

  actions: The rotations constitute actions. Each rotation/action changes the configuration of the cube from one state to another.

  path costs: Each rotation induces a cost, for instance 1 (or 2 for 180° rotations). The path cost is the sum of the rotations' costs.

- You have to color a map of Europe with only four colors. In order for the national borders to be recognizable, no two neighboring countries may be assigned the same color.

  Solution:

  state space: The states are 49-tuples of colors, e.g., *red*, *green*, *blue*, *yellow*, or *uncolored*. Every entry represents a country on the map. The colors can be encoded with values $0-4$, i.e., 0 (*red*) to 4 (*uncolored*), for compactness' sake.

  initial state: All countries are uncolored / all countries are colored red / each country is either assigned one of the four colores or "uncolored".

  goal test: Are all countries colored and are neighbouring countries colored differently?

  actions: Assigning a color to a country given no neighboring country is assigned the same color.

  path costs: Each action induces a uniform cost of 1.

Exercise 2.2 (Search algorithms)

Prove each of the following statements:

(a) Breadth-first search is a special case of uniform-cost search.

Solution:

Breadth-first search always expands an unexpanded node of minimal depth, while uniform-cost search (UCS) always expands an unexpanded node with minimal path costs. If the action costs for UCS are constant 1, then a node has depth k exactly when its path costs are k. In particular, a node has minimal depth exactly when it has minimum path costs. The expansion criteria are therefore identical.

(b) Breadth-first search, depth-first search, and uniform-cost search are special cases of best-first search.

Solution:

It was shown in the previous exercise that breadth-first search is a special case of uniform-cost search. Thus, it remains to be shown that depth-first search (DFS) and uniform-cost search (UCS) are special cases of best-first search (BFS):

(a) DFS is a special case of BFS where $h(n) := -depth(n)$.

(b) UCS is a special case of BFS where $h(n) := g(n)$.

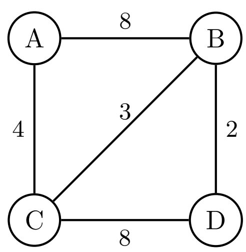
(c) Uniform-cost search is a special case of A* search.

Solution:

The evaluation function f of UCS is $f(n) = g(n)$. In contrast, the evaluation function of A* is $f(n) = g(n) + h(n)$. So UCS is a special case of A* where $h \equiv 0$.

Exercise 2.3 (Search)

Consider the graph depicted below. We are interested in a path from node **A** to node **D**. The cost of moving between two nodes is given by the respective edge weight. In the following, we refer to instances of the general TREE-SEARCH algorithm (not GRAPH-SEARCH).



	$h(x)$
A	8
B	3
C	4
D	0

(a) Perform a *Greedy Best-First Search* using heuristic h. In which order are the nodes expanded?

Solution:

Note: In the following tables, i denotes the iteration of the respective search algorithm. The search frontier contains the search nodes that are considered for expansion next. Each such node is a tuple, consisting of a label refering to a node in the problem graph and the respective f-value. Underlined nodes are expanded next.

i	search frontier
0	$\underline{(A, 8)}$
1	$\underline{(B, 3)}$ $(C, 4)$
2	$(C, 4)$ $\underline{(D, 0)}$

Expansion order: A, B, D

(b) Perform a $A^\star$ search using heuristic h. In which order are the nodes expanded?

Solution:

i	search frontier
0	$\underline{\text{(A,0+8)}}$
1	(B,8+3) $\underline{\text{(C,4+4)}}$
2	(B,8+3) (A,8+8) $\underline{\text{(B,7+3)}}$ (D,12+0)
3	(B,8+3) (A,8+8) (D,12+0) (A,15+8) (C,10+4) $\underline{\text{(D,9+0)}}$

Expansion order: A, C, B, D

(c) Perform a *Uniform Cost Search*. In which order are the nodes expanded?

Solution:

i	search frontier
0	$\underline{\text{(A,0)}}$
1	(B,8) $\underline{\text{(C,4)}}$
2	(B,8) (A,8) $\underline{\text{(B,7)}}$ (D,12)
3	$\underline{\text{(B,8)}}$ (A,8) (D,12) (A,15) (C,10) (D,9)
4	$\underline{\text{(A,8)}}$ (D,12) (A,15) (C,10) (D,9) (A,16) (C,11) (D,10)
5	(D,12) (A,15) (C,10) $\underline{\text{(D,9)}}$ (A,16) (C,11) (D,10) (B,16) (C,12)

Expansion order: A, C, B, B, A, D

(d) Complete the following definition:

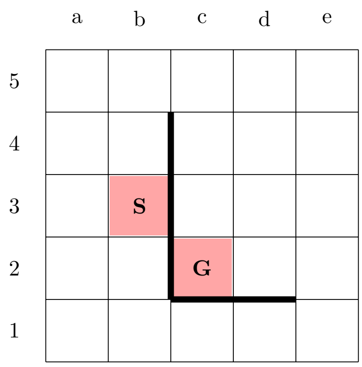
A heuristic h is *consistent* iff ...

Solution:

A heuristic h is *consistent* iff for all actions a leading from state s to state $s' : h(s) - h(s') \leq c(a)$, where $c(a)$ denotes the cost of action a.

Exercise 2.4 (A*)

A house cleaning robot tries to find the shortest path from S (start) to G (goal). The robot can move between horizontally or vertically connected grid cells, one cell in each step. If a wall (thick black line) lies in between two cells, the robot cannot move between them. Each step incurs a uniform cost of 1. Figure (a) shows the initial state, Figure (b) the heuristic value estimates of each cell.



(a) Initial state

	a	b	c	d	e
5	5	4	3	4	5
4	4	3	2	3	4
3	3	2	1	2	3
2	2	1	0	1	2
1	3	2	1	2	3

(b) Heuristic values

(a) Perform an A* search to find the shortest path from S to G. For all generated nodes, write down the respective g- and f-values in the corresponding grid cell. All other cells should be left blank. Use the graph-based variant of A* and re-open explored nodes when a better estimate becomes known (see slide 19 of 4-Informed-search-methods.pdf).

Solution:

	a	b	c	d	e
5	8	6	6	8	
4	6	4	6	8	
3	4	**S**	6	8	
2	4	2	**G**		
1	6	4	4	6	8

($e1$ is optional)

(b) Is the heuristic from Figure (b) admissible?

Solution:

A heuristic h is admissible if $h(n) \leq h^\star(n)$ for all n, where $h^\star$ is the optimal heuristic. I.e. h is admissible if it never over estimates the cost of the cheapest solution from n to a goal. The heuristic shown in figure (b) is admissible. (It's the Manhattan Distance heuristic.)

(c) Let $h^*(n)$ be the actual cost of the optimal path from n to the goal G. How many nodes does A* expand when using the h^* heuristic?

Solution:

The implementation presented in the lecture needs 7 expansions (b3, b4, b5, c5, c4, c3, c2).

Foundations of Artificial Intelligence

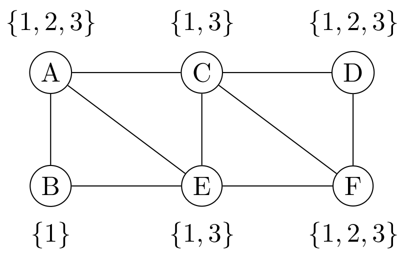
Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel,
Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

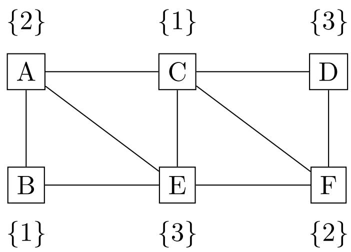
Exercise Sheet 4 — Solutions

Exercise 4.1 (Arc consistency)

Consider the constraint satisfaction problem given by the constraint graph below. The constraints are such that no two adjacent nodes have the same value. Establish arc-consistency.



Solution:



Exercise 4.2 (Satisfiability, Models)

(a) Decide for each of the following propositions whether they are valid, satisfiable or neither valid nor satisfiable.

(a) $\mathit{Smoke} \Rightarrow \mathit{Smoke}$

(b) $\mathit{Smoke} \Rightarrow \mathit{Fire}$

(c) $(\mathit{Smoke} \Rightarrow \mathit{Fire}) \Rightarrow (\neg \mathit{Fire} \Rightarrow \neg \mathit{Smoke})$

(d) $(\mathit{Smoke} \Rightarrow \mathit{Fire}) \Rightarrow ((\mathit{Smoke} \land \mathit{Heat}) \Rightarrow \mathit{Fire})$

(e) $\mathit{Spring} \Leftrightarrow \mathit{SunnyWeather}$

Solution:

In all cases it is possible to create a truth table in order to demonstrate validity or to disprove satisfiability. To show satisfiability, it is enough to find a single satisfiable variable assignment. Accordingly, validity is disproven by a single non-satisfiable variable assignment.

(a) *Smoke* $\Rightarrow$ *Smoke*: This expression simplifies to: $\neg S \vee S$.

S	$\neg S$	$\neg S \vee S$
0	1	1
1	0	1

Hence, valid (truth table) and thus also satisfiable.

(b) *Smoke* $\Rightarrow$ *Fire*: Satisfiable ($\{S \mapsto 1, F \mapsto 1\}$), but not valid ($\{S \mapsto 1, F \mapsto 0\}$)

(c) (*Smoke* $\Rightarrow$ *Fire*) $\Rightarrow$ ($\neg$*Fire* $\Rightarrow$ $\neg$*Smoke*):This expression simplifies to: $\neg(\neg S \vee F) \vee (F \vee \neg S)$.

S	F	$\neg(\neg S \vee F)$	$(F \vee \neg S)$	$\neg(\neg S \vee F) \vee (F \vee \neg S)$
0	0	0	1	1
0	1	0	1	1
1	0	1	0	1
1	1	0	1	1

Hence, valid (truth table) and thus also satisfiable.

(d) (*Smoke* $\Rightarrow$ *Fire*) $\Rightarrow$ ((*Smoke* $\wedge$ *Heat*) $\Rightarrow$ *Fire*): This expression simplifies to: $\neg(\neg S \vee F) \vee (\neg(S \wedge H) \vee F)$.

H	S	F	$\neg(\neg S \vee F)$	$\neg(S \wedge H) \vee F$	$\neg(\neg S \vee F) \vee (\neg(S \wedge H) \vee F)$
0	0	0	0	1	1
0	0	1	0	1	1
0	1	0	1	1	1
0	1	1	0	1	1
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	0	1	1

Valid (truth table) and thus also satisfiable.

(e) *Spring* $\Leftrightarrow$ *SunnyWeather*: Satisfiable ($\{Sp \mapsto 1, SW \mapsto 1\}$), but not valid ($\{Sp \mapsto 0, SW \mapsto 1\}$).

(b) Consider a vocabulary with only four propositions, A, B, C, and D. How many models are there for the following formulae? Explain.

(a) $(A \wedge B) \vee (B \wedge C)$

Solution:

Notation: *abcd* with $a, b, c, d \in \{0, 1, X\}$ as a short form for $\{A \mapsto a, B \mapsto b, C \mapsto c, D \mapsto d\}$. X as notation for: either 0 or 1 is possible.

Models for $A \wedge B$ are all assignments $11XX$ (i.e., four cases: 1100, 1101, 1110, 1111), Models for $B \wedge C$ are all assignments $X11X$ (also four cases). An assignment is a model of $(A \wedge B) \vee (B \wedge C)$ if and only if it is a model of $A \wedge B$ or a model of $B \wedge C$. Thus, there are six models in total, since 1110 and 1111 should not be counted twice.

(b) $A \vee B$

Solution:

The only assignments which are *not* a model of $A \vee B$ are models of $\neg A \wedge \neg B$, i.e., four assignments of the form $00XX$. The remaining 12 out of 16 assignments are models of $A \vee B$.

(c) $(A \leftrightarrow B) \wedge (B \leftrightarrow C)$

Solution:

Models have the form $000X$ or $111X$, i.e., four models in total.

Exercise 4.3 (CNF Transformation, Resolution Method)

The following transformation rules hold, whereby propositional formulae can be transformed into equivalent formulae. Here, φ, ψ, and χ are arbitrary propositional formulae:

$$\neg\neg\varphi \equiv \varphi \tag{1}$$
$$\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi \tag{2}$$
$$\varphi \vee (\psi \wedge \chi) \equiv (\varphi \vee \psi) \wedge (\varphi \vee \chi) \tag{3}$$
$$\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi \tag{4}$$
$$\varphi \wedge (\psi \vee \chi) \equiv (\varphi \wedge \psi) \vee (\varphi \wedge \chi) \tag{5}$$

Additionally, the operators $\vee$ and $\wedge$ are associative and commutative.

Consider the formula $((C \wedge \neg B) \leftrightarrow A) \wedge (\neg C \to A)$.

(a) Transform the formula into a clause set K using the CNF transformation rules. Write down the steps.

Solution:

Transformation to CNF:

$$\begin{aligned}((C \wedge \neg B) \leftrightarrow A) \wedge (\neg C \to A) &\equiv ((C \wedge \neg B) \to A) \wedge (A \to (C \wedge \neg B)) \wedge (\neg C \to A)\\ &\equiv (\neg(C \wedge \neg B) \vee A) \wedge (\neg A \vee (C \wedge \neg B)) \wedge (C \vee A)\\ &\equiv (\neg C \vee B \vee A) \wedge (\neg A \vee C) \wedge (\neg A \vee \neg B) \wedge (C \vee A)\end{aligned}$$

In Clause Normal Form we thus have

$$K = \{\{A, B, \neg C\}, \{\neg A, C\}, \{\neg A, \neg B\}, \{A, C\}\}.$$

(b) Afterwards, using the resolution method, show whether $K \models (\neg B \rightarrow (A \wedge C))$ holds.

Solution:

In order to show that $K \models \varphi$, it is sufficient to show that $K \cup \{\neg\varphi\} \models \bot$. Therefore, we first have to extend K by the clauses corresponding to $\neg(\neg B \rightarrow (A \wedge C))$, and then derive a contradiction (empty set) using resolution. Transformation of $\neg(\neg B \rightarrow (A \wedge C))$ to CNF:

$$\begin{aligned}\neg(\neg B \rightarrow (A \wedge C)) &\equiv \neg B \wedge \neg(A \wedge C)\\ &\equiv \neg B \wedge (\neg A \vee \neg C)\end{aligned}$$

i.e., $\{\{\neg B\}, \{\neg A, \neg C\}\}$.

Resolution (one of many possibilities, recommended notation):

$$\begin{aligned}
&\Big\{A, B, \neg C\Big\} && (1)\\
&\Big\{\neg A, C\Big\} && (2)\\
&\Big\{\neg A, \neg B\Big\} && (3)\\
&\Big\{A, C\Big\} && (4)\\
&\Big\{\neg B\Big\} && (5)\\
&\Big\{\neg A, \neg C\Big\} && (6)\\
&(1) + (5): \quad \Big\{A, \neg C\Big\} && (7)\\
&(2) + (6): \quad \Big\{\neg A\Big\} && (8)\\
&(4) + (7): \quad \Big\{A\Big\} && (9)\\
&(8) + (9): \quad \emptyset && (10)
\end{aligned}$$

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel,
Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020
University of Freiburg
Department of Computer Science

Exercise Sheet 5 — Solutions

Exercise 5.1 (Modeling, Proofs)

Consider the following knowledge base:

If the unicorn is mythical, then it is immortal, but if it is not mythical, then it is a mortal mammal. If the unicorn is immortal or a mammal, then it is horned. The unicorn is magical if it is horned.

Using this knowledge base, can you prove that the unicorn is (a) mythical, (b) magical or (c) horned? First, formalize the knowledge base with propositional logic. If a statement is valid or unsatisfiable, use resolution for prove. Else, write down one satisfying and one unsatisfying interpretation.

Solution:

The above statements can be formalized using the atomic propositions *mythical*, *mortal*, *mammal*, *magical* and *horned*:

$$mythical \rightarrow \neg mortal \tag{i}$$

$$\neg mythical \rightarrow mortal \wedge mammal \tag{ii}$$

$$\neg mortal \vee mammal \rightarrow horned \tag{iii}$$

$$horned \rightarrow magical \tag{iv}$$

Let $KB = (i) \wedge (ii) \wedge (iii) \wedge (iv)$ be the set of those four propositions, in CNF:

$$\Big\{\neg mythical, \neg mortal\Big\} \tag{1}$$

$$\Big\{mythical, mortal\Big\} \tag{2}$$

$$\Big\{mythical, mammal\Big\} \tag{3}$$

$$\Big\{mortal, horned\Big\} \tag{4}$$

$$\Big\{\neg mammal, horned\Big\} \tag{5}$$

$$\Big\{\neg horned, magical\Big\} \tag{6}$$

(a) We cannot tell from KB whether the unicorn is mythical or not, since there exist two models $I_{mythical}$ and $I_{\neg mythical}$ of KB with $I_{mythical} \models mythical$ and $I_{\neg mythical} \models \neg mythical$.

More specific:

$I_{mythical} = \{mythical \mapsto 1, mortal \mapsto 0, mammal \mapsto 0, magical \mapsto 1, horned \mapsto 1\}$

and

$I_{\neg mythical} = \{mythical \mapsto 0, mortal \mapsto 1, mammal \mapsto 1, magical \mapsto 1, horned \mapsto 1\}$.

Thus, $KB \not\models \neg mythical$ and $KB \not\models mythical$, since neither *mythical* nor $\neg$*mythical* are true in *all* models of KB. In other words, the unicorn could be mythical or not, depending on the choice of model for KB.

(b) $KB \models horned$ holds, since $KB \cup \{\neg horned\}$ is unsatisfiable. Proof:

Let

$$\Big\{\neg horned\Big\} \tag{7a}$$

then

$$(5) + (7a): \quad \Big\{\neg mammal\Big\} \tag{8a}$$
$$(4) + (7a): \quad \Big\{mortal\Big\} \tag{9a}$$
$$(3) + (8a): \quad \Big\{mythical\Big\} \tag{10a}$$
$$(1) + (10a): \quad \Big\{\neg mortal\Big\} \tag{11a}$$
$$(11a) + (9a): \quad \emptyset$$

(c) $KB \models magical$ holds, since $KB \cup \{\neg magical\}$ is unsatisfiable. Proof:

Let

$$\Big\{\neg magical\Big\} \tag{7b}$$

then

$$(6) + (7b): \quad \Big\{\neg horned\Big\} \tag{8b}$$
$$(5) + (8b): \quad \Big\{\neg mammal\Big\} \tag{9b}$$
$$(4) + (8b): \quad \Big\{mortal\Big\} \tag{10b}$$
$$(3) + (9b): \quad \Big\{mythical\Big\} \tag{11b}$$
$$(1) + (11b): \quad \Big\{\neg mortal\Big\} \tag{12b}$$
$$(12b) + (10b): \quad \emptyset$$

Exercise 5.2 (DPLL)

Use the Davis-Putnam-Logemann-Loveland (DPLL) procedure to check whether the given formulae ϕ_1 and ϕ_2 are satisfiable or not. Write down all steps carried out by the algorithm during the process. If you have to apply a splitting rule, split on variables in alphabetical order, trying *true* first, then *false*. Should the formula be satisfiable, please indicate the satisfying assignment.

(a)

$\phi_1 = (D \vee C) \wedge (\neg A \vee \neg D \vee B \vee \neg C) \wedge (A \vee C) \wedge (\neg C \vee \neg B) \wedge (\neg C \vee B \vee \neg A \vee D) \wedge (C \vee \neg D \vee B) \wedge (\neg D \vee \neg B \vee \neg A)$

Solution:

$$\begin{aligned}
\text{Splitting } A \to 1 \quad & (D \vee C) \wedge (\neg D \vee B \vee \neg C) \wedge (\neg C \vee \neg B) \wedge (\neg C \vee B \vee D) \wedge \\
& (C \vee \neg D \vee B) \wedge (\neg D \vee \neg B) \\
\text{Splitting } B \to 1 \quad & (D \vee C) \wedge \neg C \wedge \neg D \\
\text{Unit-propagation } C \to 0 \quad & D \wedge \neg D \\
\text{Unit-propagation } D \to 1 \quad & \bot \\
\text{Backtracking } B \to 0 \quad & (D \vee C) \wedge (\neg D \vee \neg C) \wedge (\neg C \vee D) \wedge (C \vee \neg D) \\
\text{Splitting } C \to 1 \quad & \neg D \wedge D \\
\text{Unit-propagation } D \to 1 \quad & \bot \\
\text{Backtracking } C \to 0 \quad & D \wedge \neg D \\
\text{Unit-propagation } D \to 1 \quad & \bot \\
\text{Backtracking } A \to 0 \quad & (D \vee C) \wedge C \wedge (\neg C \vee \neg B) \wedge (C \vee \neg D \vee B) \\
\text{Unit-propagation } C \to 1 \quad & \neg B \\
\text{Unit-propagation } B \to 0 \quad & \top
\end{aligned}$$

Satisfying assignment: $A \to 0; B \to 0; C \to 1; D \to 1$ or 0;

(b)

$\phi_2 = (D \vee \neg A \vee B) \wedge (\neg B \vee \neg C \vee A \vee D) \wedge (\neg B \vee \neg A) \wedge (B \vee \neg D) \wedge (A \vee C \vee \neg B) \wedge (\neg D \vee \neg C) \wedge (B \vee D)$

Solution:

$$\begin{aligned}
\text{Splitting } A \to 1 \quad & (D \vee B) \wedge \neg B \wedge (B \vee \neg D) \wedge (\neg D \vee \neg C) \wedge (B \vee D) \\
\text{Unit-propagation } B \to 0 \quad & D \wedge \neg D \wedge (\neg D \vee \neg C) \wedge D \\
\text{Unit-propagation } D \to 1 \quad & \bot \\
\text{Backtracking } A \to 0 \quad & (\neg B \vee \neg C \vee D) \wedge (B \vee \neg D) \wedge (C \vee \neg B) \wedge (\neg D \vee \neg C) \wedge (B \vee D) \\
\text{Splitting } B \to 1 \quad & (\neg C \vee D) \wedge C \wedge (\neg D \vee \neg C) \\
\text{Unit-propagation } C \to 1 \quad & D \wedge \neg D \\
\text{Unit-propagation } D \to 1 \quad & \bot \\
\text{Backtracking } B \to 0 \quad & \neg D \wedge (\neg D \vee \neg C) \wedge D \\
\text{Unit-propagation } D \to 1 \quad & \bot
\end{aligned}$$

The formula is unsatisfiable.

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel, Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

Exercise Sheet 6 — Solutions

Exercise 6.1 (Semantics of Predicate Logic)

Consider the Interpretation $\mathcal{I} =< \mathcal{D}, \cdot^{\mathcal{I}} >$ with

- $\mathcal{D} = \{0, 1, 2, 3\}$
- $\mathit{even}^{\mathcal{I}} = \{0, 2\}$
- $\mathit{odd}^{\mathcal{I}} = \{1, 3\}$
- $\mathit{lessThan}^{\mathcal{I}} = \{(0, 1), (0, 2), (0, 3), (1, 2), (1, 3), (2, 3)\}$
- $\mathit{two}^{\mathcal{I}} = 2$
- $\mathit{plus}^{\mathcal{I}} : \mathcal{D} \times \mathcal{D} \to \mathcal{D}, \mathit{plus}^{\mathcal{I}}(a, b) = (a + b) \mod 4$

and the variable assignment $\alpha = \{(x, 0), (y, 1)\}$.
Decide for the following formulae θ_i if $\mathcal{I}$ is a model for θ_i under α, i.e. if $\mathcal{I}, \alpha \models \theta_i$.
Explain your answer by formally applying the semantics.

(a) $\theta_1 = \mathit{odd}(y) \wedge \mathit{even}(\mathit{two})$

(b) $\theta_2 = \forall x\ (\mathit{even}(x) \vee \mathit{odd}(x))$

(c) $\theta_3 = \forall x \exists y\ \mathit{lessThan}(x, y)$

(d) $\theta_4 = \forall x\ (\mathit{even}(x) \Rightarrow \exists y\ \mathit{lessThan}(x, y))$

(e) $\theta_5 = \forall x\ (\mathit{odd}(x) \Rightarrow \mathit{even}(\mathit{plus}(x, y)))$

Solution:

(a) $\theta_1^{\mathcal{I}} =$
$(\mathit{odd}(y) \wedge \mathit{even}(\mathit{two}))^{\mathcal{I}} =$
$\mathit{odd}^{\mathcal{I}}(y^{\mathcal{I}}) \wedge \mathit{even}^{\mathcal{I}}(\mathit{two}^{\mathcal{I}}) =$
$\mathit{odd}^{\mathcal{I}}(1) \wedge \mathit{even}^{\mathcal{I}}(2) =$
$\top \wedge \top = \top$, since $1 \in \mathit{odd}^{\mathcal{I}}$ and $2 \in \mathit{even}^{\mathcal{I}}$.
Thus, $\mathcal{I}, \alpha \models \theta_1$.

(b) Let $\theta_2 = \forall x \phi_2$ with $\phi_2 = \mathit{even}(x) \vee \mathit{odd}(x)$.
Then $\mathcal{I}, \alpha \models \theta_2$ iff $\mathcal{I}, \alpha[x/d](\phi_2) = \top$ for all $d \in \mathcal{D}$.
Notice that here x is bound by a quantifier and therefore the variable assignment in the definition of α is overridden by $\alpha[x/d]$.

Since our universe $\mathcal{D}$ is finite (and small), we can evaluate the above for all $d \in \mathcal{D}$:

$\mathcal{I}, \alpha[x/0](\phi_2) = \mathit{even}^{\mathcal{I}}(0) \lor \mathit{odd}^{\mathcal{I}}(0) = \top \lor \bot = \top$
$\mathcal{I}, \alpha[x/1](\phi_2) = \mathit{even}^{\mathcal{I}}(1) \lor \mathit{odd}^{\mathcal{I}}(1) = \bot \lor \top = \top$
$\mathcal{I}, \alpha[x/2](\phi_2) = \mathit{even}^{\mathcal{I}}(2) \lor \mathit{odd}^{\mathcal{I}}(2) = \top \lor \bot = \top$
$\mathcal{I}, \alpha[x/3](\phi_2) = \mathit{even}^{\mathcal{I}}(3) \lor \mathit{odd}^{\mathcal{I}}(3) = \bot \lor \top = \top$

Thus, $\mathcal{I}, \alpha \models \theta_2$.

(c) Let $\theta_3 = \forall x \exists y \phi_3$ with $\phi_3 = \mathit{lessThan}(x, y)$.
Then $\mathcal{I}, \alpha \models \theta_3$ iff for all $d_1 \in \mathcal{D}$ there exists a $d_2 \in \mathcal{D}$ so that $\mathcal{I}, \alpha[x/d_1, y/d_2](\phi_3) = \top$.
In this case it is clear that for $d_1 = 3$ no such d_2 exists, more formally, $\mathit{lessThan}^{\mathcal{I}}(3, d_2) = \bot$ for all $d_2 \in \mathcal{D}$ and therefore $I, \alpha \not\models \theta_3$.

(d) Let $\theta_4 = \forall x \phi_4$ with $\phi_4 = \mathit{even}(x) \Rightarrow \exists y\ \mathit{lessThan}(x, y)$.
Then $\mathcal{I}, \alpha \models \theta_4$ iff $\mathcal{I}, \alpha[x/d_1](\phi_4) = \top$ for all $d_1 \in \mathcal{D}$
We again consider all cases, starting odd numbers, i.e., $d_1 \in \{1, 3\}$: $(\mathit{even}(x) \Rightarrow \exists y\ \mathit{lessThan}(x, y))^{\mathcal{I}} = \neg \mathit{even}^{\mathcal{I}}(d_1) \lor ... =$
$\neg \bot \lor ... = \top \lor ... = \top$.
For $d_1 \in \{0, 2\}$ we have $\neg \mathit{even}^{\mathcal{I}}(d_1) = \bot$, thus we have to find a corresponding $d_2 \in \mathcal{D}$ with $\mathcal{I}, \alpha[x/d_1, y/d_2](\mathit{lessThan}(x, y)) = \top$. Here, we have $(0, 1) \in \mathit{lessThan}^{\mathcal{I}}$ and $(2, 3) \in \mathit{lessThan}^{\mathcal{I}}$.
Summing up all of the above, we showed that $\mathcal{I}, \alpha \models \theta_4$.

(e) Let $\theta_5 = \forall x \phi_5$ with $\phi_5 = \mathit{odd}(x) \Rightarrow \mathit{even}(\mathit{plus}(x, y))$.
Then $\mathcal{I}, \alpha \models \theta_5$ iff $\mathcal{I}, \alpha[x/d](\phi_5) = \top$ for all $d \in \mathcal{D}$
$\mathcal{I}, \alpha[x/d](\phi_5) = \neg \mathit{odd}^{\mathcal{I}}(d) \lor \mathit{even}^{\mathcal{I}}(\mathit{plus}^{\mathcal{I}}(d, 1))$
Notice that here y is a free variable and therefore the assignment $[y/1]$ is applied.
Now we consider all cases for d:
$\mathcal{I}, \alpha[x/0](\phi_5) = \neg \mathit{odd}^{\mathcal{I}}(0) \lor \mathit{even}^{\mathcal{I}}(\mathit{plus}^{\mathcal{I}}(0, 1)) = \neg \mathit{odd}^{\mathcal{I}}(0) \lor \mathit{even}^{\mathcal{I}}(1) = \top \lor \bot = \top$
$\mathcal{I}, \alpha[x/1](\phi_5) = \neg \mathit{odd}^{\mathcal{I}}(1) \lor \mathit{even}^{\mathcal{I}}(\mathit{plus}^{\mathcal{I}}(1, 1)) = \neg \mathit{odd}^{\mathcal{I}}(1) \lor \mathit{even}^{\mathcal{I}}(2) = \bot \lor \top = \top$
$\mathcal{I}, \alpha[x/2](\phi_5) = \neg \mathit{odd}^{\mathcal{I}}(2) \lor \mathit{even}^{\mathcal{I}}(\mathit{plus}^{\mathcal{I}}(2, 1)) = \neg \mathit{odd}^{\mathcal{I}}(2) \lor \mathit{even}^{\mathcal{I}}(3) = \top \lor \bot = \top$
$\mathcal{I}, \alpha[x/3](\phi_5) = \neg \mathit{odd}^{\mathcal{I}}(3) \lor \mathit{even}^{\mathcal{I}}(\mathit{plus}^{\mathcal{I}}(3, 1)) = \neg \mathit{odd}^{\mathcal{I}}(3) \lor \mathit{even}^{\mathcal{I}}(0) = \bot \lor \top = \top$
Thus, $\mathcal{I}, \alpha \models \theta_5$.

Exercise 6.2 (Semantics of Predicate Logic)
Consider the Interpretation $\mathcal{I} =< \mathcal{D}, \cdot^{\mathcal{I}} >$ and the variable assignment α:

- $\mathcal{D} = \{a, b, c\}$
- $P^{\mathcal{I}} = \{(a, a), (a, b), (b, a), (b, b), (b, c), (c, a)\}$
- $Q^{\mathcal{I}} = \{a, b\}$
- $R^{\mathcal{I}} = \{(a, a), (a, b), (a, c), (b, c), (c, b)\}$

- $\alpha = \{(v, a), (w, b)\}$

Decide for the following formulae θ_i if $\mathcal{I}$ is a model for θ_i under α, i.e. if $\mathcal{I}, \alpha \models \theta_i$. Explain your answer by formally applying the semantics.

(a) $\theta_1 = \forall x\big(P(x, w) \Rightarrow Q(x)\big)$

(b) $\theta_2 = \exists x\big(R(v, x) \Rightarrow P(x, x)\big)$

(c) $\theta_3 = \forall x \forall y\big(R(x, y) \iff Q(y)\big)$

(d) $\theta_4 = \Big[\neg\forall x \forall y\big(Q(y) \lor P(x, y)\big)\Big] \land \Big[\exists z\big(Q(z) \lor P(w, z)\big)\Big]$

Solution:

(a) Let $\theta_1 = \forall x \phi_1$ with $\phi_1 = \big(P(x, w) \Rightarrow Q(x)\big)$.
Then $\mathcal{I}, \alpha \models \theta_1$ iff $\mathcal{I}, \alpha[x/d](\phi_1) = \top$ for all $d \in \mathcal{D}$.
Notice that here x is bound by a quantifier and therefore the variable assignment in the definition of α is overridden by $\alpha[x/d]$.
Also w is a free variable and therefore the assignment $[w/b]$ is applied.
Since our universe $\mathcal{D}$ is finite (and small), we can evaluate the above for all $d \in \mathcal{D}$:
$\mathcal{I}, \alpha[x/a](\phi_1) = \neg P^{\mathcal{I}}(a, b) \lor Q^{\mathcal{I}}(a) = \bot \lor \top = \top$
$\mathcal{I}, \alpha[x/b](\phi_1) = \neg P^{\mathcal{I}}(b, b) \lor Q^{\mathcal{I}}(b) = \bot \lor \top = \top$
$\mathcal{I}, \alpha[x/c](\phi_1) = \neg P^{\mathcal{I}}(c, b) \lor Q^{\mathcal{I}}(c) = \top \lor \bot = \top$
Thus, $\mathcal{I}, \alpha \models \theta_1$.

(b) Let $\theta_2 = \exists x \phi_2$ with $\phi_2 = \big(R(v, x) \Rightarrow P(x, x)\big)$.
Then $\mathcal{I}, \alpha \models \theta_3$ if there exists a $d \in \mathcal{D}$
so that $\mathcal{I}, \alpha[x/d](\phi_2) = \top$.
Also v is a free variable and therefore the assignment $[w/a]$ is applied.
One case which can be applied is $d = b$.
$\mathcal{I}, \alpha[x/b](\phi_2) = \neg R^{\mathcal{I}}(a, b) \lor P^{\mathcal{I}}(b, b) = \bot \lor \top = \top$
Thus $\mathcal{I}, \alpha \models \theta_2$.

(c) Let $\theta_3 = \forall x \forall y \phi_3$ with $\phi_3 = \big(R(x, y) \iff Q(y)\big)$.
Then $\mathcal{I}, \alpha \models \theta_3$ iff $\mathcal{I}, \alpha[x/d1][y/d2](\phi_1) = \top$ for all $d1$ and $d2 \in \mathcal{D}$.
However, it can does not satisfy by assigning x=a and y = c,
$\mathcal{I}, \alpha[x/a][y/c](\phi_3) = \big(R^{\mathcal{I}}(a, c) \iff Q^{\mathcal{I}}(c)\big) = \bot \iff \top = \bot$
Hence $\mathcal{I}, \alpha \not\models \theta_3$.

(d) $\mathcal{I}, \alpha \models \theta_4$
Let $\theta_4 = \neg\theta_4^a \land \theta_4^b$ with
$\theta_4^a = \forall x \forall y\big(Q(y) \lor P(x, y)\big)$ and
$\theta_4^b = \exists z\big(Q(z) \lor P(w, z)\big)$
Thus, $\mathcal{I}, \alpha$ must model θ_4^b but not θ_4^a.
$\mathcal{I}, \alpha \not\models \theta_4^a$, since for $y = c$, $c \notin Q^{\mathcal{I}}$ and $(c, c) \notin P^{\mathcal{I}}$
$\mathcal{I}, \alpha \models \theta_4^b$, since, for instance, $a \in Q^{\mathcal{I}}$.

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel, Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

Exercise Sheet 7 — Solutions

Exercise 7.1 (Planning)

Consider the following STRIPS-Task $\Pi = \langle \mathcal{S}, O, I, G \rangle$:

- $\mathcal{S}$: {X, Y, Z, G}
- $O : \{A, B, C, D, E, F\}$ where

$$\begin{aligned}
A &: pre(A) = \{X\}, & eff(A) &= \{Y, Z\} \\
B &: pre(B) = \{X\}, & eff(B) &= \{\neg X, Z\} \\
C &: pre(C) = \{\neg Y\}, & eff(C) &= \{Z\} \\
D &: pre(D) = \{\neg Z\}, & eff(D) &= \{Y\} \\
E &: pre(E) = \{\neg X, Y\}, & eff(E) &= \{\neg Y, G\} \\
F &: pre(F) = \{Z\}, & eff(F) &= \{\neg Z, G\}
\end{aligned}$$

- I: {X, Y}
- G: {G}

(a) State for each operator from O if it is applicable in I or not. For each applicable operator also give the resulting state after applying that operator in I.

Solution:

Operator	Applicable?	Resulting State
A	Yes	$\{X, Y, Z\}$
B	Yes	$\{Y, Z\}$
C	No	-
D	Yes	$\{X, Y\}$
E	No	-
F	No	-

(b) Give an applicable plan π that leads from I to G.

Solution:

$\pi = \langle B, E \rangle$, $\langle A, F \rangle$, $\langle B, F \rangle$, $\ldots$

Exercise 7.2 (Bayes' Rule)

In Freiburg 80% of all cars are red. You see a car at night that does *not* appear red to you. You know that you can correctly identify a red car only in 70% of the cases when the given car is red. And you can identify a non-red car correctly in 90% of the cases when the given car is non-red.

(a) List all conditional and non-conditional probabilities that you can determine directly from the task description. Note: Differentiate between the statement that a car *is red* and the statement that you have *seen a red car*.

(b) Compute the probability that the car is actually red, when you perceive a car as red in Freiburg at night.

Solution:

P(R): car is red
P(PR): car is perceived red
$P(R) = 0.8, P(\neg R) = 0.2$
$P(PR|R) = 0.7$
$P(\neg PR|\neg R) = 0.9$

$$P(R|PR) = \frac{P(PR|R) \cdot P(R)}{P(PR)}$$

$$= \frac{P(PR|R) \cdot P(R)}{P(PR|R) \cdot P(R) + P(PR|\neg R) \cdot P(\neg R)}$$

$$= \frac{0.7 \cdot 0.8}{0.7 \cdot 0.8 + 0.1 \cdot 0.2}$$

$$= \frac{0.56}{0.56 + 0.02} = \frac{0.56}{0.58} = \frac{28}{29}$$

Exercise 7.3 (Independence and Joint and Conditional Probabilities)

(a) A 6-sided die is rolled once. Which of the following events are independent? Show the probability values and reasoning.

- E : An even number is rolled
- O : An odd number is rolled
- T : A number ≥ 3 is rolled

Solution:

We know there are 6 possible outcomes for the roll of the die.

$P(E) = 0.5$	3 out of 6 possibilities are covered under the event
$P(O) = 0.5$	3 out of 6 possibilities are covered under the event
$P(T) = \frac{2}{3}$	4 out of 6 possibilities are covered under the event
$P(E \cap O) = 0 \neq P(E) * P(O) = 0.25$	E and O are disjoint events
$P(E \cap T) = \frac{1}{3} = P(E) * P(T) = \frac{1}{3}$	E and T cover 2 out of 6 possibilities
$P(O \cap T) = \frac{1}{3} = P(O) * P(T) = \frac{1}{3}$	O and T cover 2 out of 6 possibilities

By definition of independence, E and T are independent and O and T are independent.

(b) Make the joint probability distribution table for the events E and T.

Solution:

	$E = \mathit{False}$	$E = \mathit{True}$
T = *False*	0.167	0.167
T = *True*	0.333	0.333

(c) Calculate the conditional probability $P(\neg e \,|\, t)$.

Solution:

$P(\neg e \,|\, t) = P(\neg e \wedge t)/P(t) = 0.333/0.666 = 0.5$

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel,
Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

Exercise Sheet 8 — Solutions

Exercise 8.1 (Conditional Independence and Bayes Networks)

(a) Consider exercise 7.3 (a) in exercise sheet 7. You are given additional information that you rolled a number ≥ 2 (let's name this event U). Are any of the events E, O and T conditionally independent given this information? Show the probability values and reasoning.

Solution:

When given that U has occurred, there are 5 remaining possible outcomes for the roll of the die. Then

$$\begin{aligned}
P(E|U) &= 0.6 && \text{3 out of 5 possibilities are covered under the event}\\
P(O|U) &= 0.4 && \text{2 out of 5 possibilities are covered under the event}\\
P(T|U) &= 0.8 && \text{4 out of 5 possibilities are covered under the event}\\
P(E \cap O|U) &= 0 \neq P(E|U) * P(O|U) = 0.24 && \text{E and O are disjoint events}\\
P(E \cap T|U) &= 0.4 \neq P(E|U) * P(T|U) = 0.48 && \text{E and T cover 2 out of 5 possibilities}\\
P(O \cap T|U) &= 0.4 \neq P(O|U) * P(T|U) = 0.32 && \text{O and T cover 2 out of 5 possibilities}
\end{aligned}$$

We see that, by definition, none of the events are conditionally independent given U.

(b) Consider the following probability tables (all variables are binary, thus they can be either true or false):

$P(B)$
0.8

$P(A)$
0.6

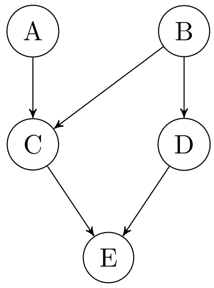
B	$P(D)$
F	0.5
T	0.1

A	B	$P(C)$
F	F	0.8
F	T	0.2
T	F	0.1
T	T	0.1

C	D	$P(E)$
F	F	0.1
F	T	0.3
T	F	0.9
T	T	0.5

Draw the corresponding Bayesian network and compute the probability $P(A, \neg B, \neg D, E)$.

Solution:

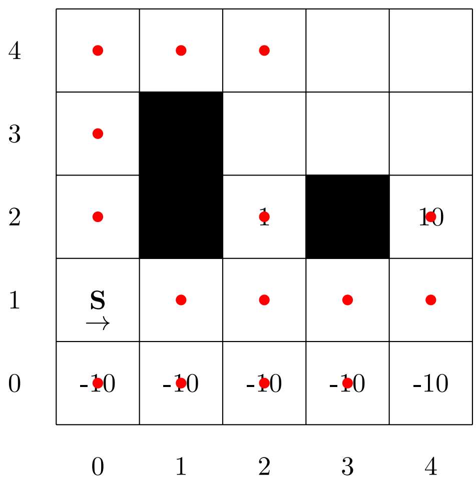


$$P(A, \neg B, \neg D, E)$$

$$\begin{aligned}
P(A, \neg B, \neg D, E) &= P(A)P(\neg B)P(\neg D|\neg B)\left(\sum_{c\in\{\neg C, C\}} P(c|A, \neg B)P(E|c, \neg D)\right) \\
&= 0.6 \cdot 0.2 \cdot 0.5 \cdot (0.1 \cdot 0.9 + 0.9 \cdot 0.1) \\
&= 0.6 \cdot 0.2 \cdot 0.5 \cdot (0.09 + 0.09) \\
&= 0.06 \cdot (0.18) = 0.0108
\end{aligned}$$

Exercise 8.2 (Sequential Decision Problems)

Consider the sequential decision problem in the following grid world. The agent wants to reach one of the goal states (terminal states with positive rewards) and wants to avoid falling of the cliffs (terminal states with negative rewards). The agent can carry out actions N (go north), E (go east), S (go south), or W (go west). The agent moves in the intended direction with a probability of 0.8, and with a probability of 0.1 each, it moves at right angles to the intended direction. If the agent bumps into a wall, it remains in its current state. The start state is marked **S** with the 1st action (E) marked under it. States are enumerated as in a Cartesian coordinate system. So, $\mathbf{S} = (0, 1)$.



(a) How large is the probability that the sequence of actions (E,E,E,E,N) leads to the goal state with a reward of 10? Which other states can be reached with this sequence of actions? Mark the corresponding states in the grid above.

Solution:

0.8^5

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel, Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

Exercise Sheet 9 — Solutions

Exercise 9.1 (Markov Decision Processes)

Consider the Markov decision process underlying the sequence decision problem for the grid world in exercise 8.2.

(a) Perform a policy-improvement step for the initial state **S** and the (non-optimal) policy E. Which is the optimal action in this state? The utility of all non-terminal states is assumed to be zero and rewards are assumed to be additive (i.e. not discounted).

Solution:

Let s be the Starting position. We have to calculate the expected utility for each action and then choose the action with the maximum expected utility. The expected utility for an action, with no immediate reward, is calculated as

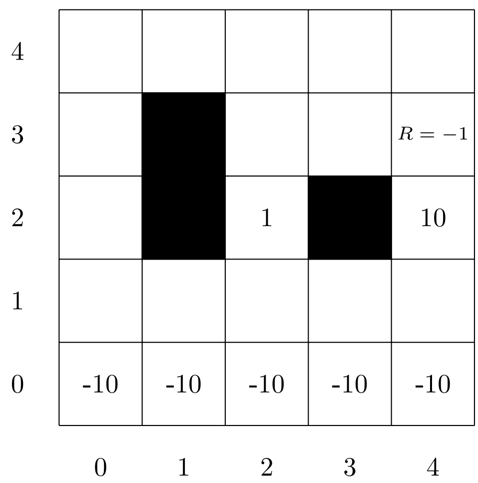
$$\sum_{s'} P(s, a, s')U(s').$$

So:

$$\begin{aligned}
\sum_{s'} P(s, \mathbf{North}, s')U(s') &= 0.8 * 0 + 0.1 * 0 + 0.1 * 0 = 0 \\
\sum_{s'} P(s, \mathbf{East}, s')U(s') &= 0.8 * 0 + 0.1 * (-10) + 0.1 * 0 = -1 \\
\sum_{s'} P(s, \mathbf{South}, s')U(s') &= 0.8 * (-10) + 0.1 * 0 + 0.1 * 0 = -8 \\
\sum_{s'} P(s, \mathbf{West}, s')U(s') &= 0.8 * 0 + 0.1 * (-10) + 0.1 * 0 = -1
\end{aligned}$$

The best action is $\arg\max_a \sum_{s'} P(s, a, s')U(s')$, i.e., **North**.

(b) Perform a first step of the value iteration algorithm for state (4,3) for the slightly **modified grid world** below. The immediate reward of this state is $R = -1$. Assume discounted rewards and a discount factor of $\gamma = 0.5$. Initially, all non-terminal states have a utility of zero.



Solution:

$$
\begin{aligned}
U'(s) &= R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s'|s,a)U(s') \\
& 0.5 * \begin{cases} 0.8 * 0 + 0.1 * 0 + 0.1 * 10 & (\rightarrow) \\ 0.8 * 0 + 0.1 * 0 + 0.1 * 0 & (\uparrow) \\ 0.8 * 0 + 0.1 * 0 + 0.1 * 10 & (\leftarrow) \\ 0.8 * 10 + 0.1 * 0 + 0.1 * 0 & (\downarrow) \end{cases} \\
&= -1 + 0.5 * 8 = 3
\end{aligned}
$$

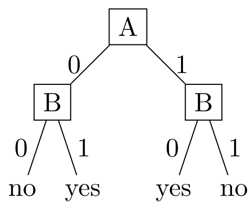
Exercise 9.2 (Decision Trees)
Specify decision trees representing the following boolean functions:

(a) A XOR B

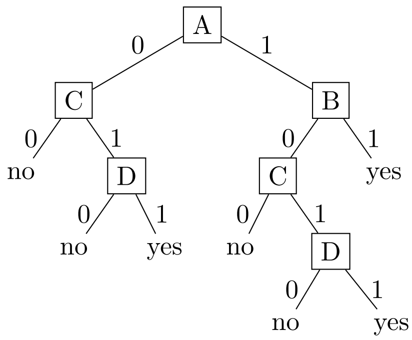
(b) $(A \wedge B) \vee (C \wedge D)$

Solution:

(a)



(b)


A
0
1
C
B
0
1
0
1
no
D
C
yes
0
1
0
1
no
yes
no
D
0
1
no
yes

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel,
Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

Exercise Sheet 10 — Solutions

Exercise 10.1 (Decision Trees)

No	Age	Engine power [kW]	**Risk**
1	< 25	< 100	low
2	< 25	> 200	high
3	≥ 25	> 200	high
4	≥ 25	$100 - 200$	low
5	< 25	$100 - 200$	high
6	≥ 25	< 100	low

Consider the data on car insurance risk in the table above. Produce a decision tree, which correctly classifies the insurance risk for the examples given, using the attributes *Age* and *Engine Power* in order of decreasing *information gain.* Give detailed calculations that justify the order in which the attributes are tested.
You can make use of the following values:
$\log_2(\frac{1}{3}) \approx -\frac{3}{2}$, $\log_2(\frac{2}{3}) \approx -\frac{1}{2}$, $\log_2(\frac{1}{2}) = -1$, $\log_2(1) = 0$.

Solution:

Entropy of the root node: $I(Risk) = I(\frac{1}{2}, \frac{1}{2}) = 1$
Remaining uncertainties after splitting on the different attributes:

$$R(EnginePower) = \frac{1}{3} \cdot I(1,0) + \frac{1}{3} \cdot I(0,1) + \frac{1}{3} \cdot I(\frac{1}{2}, \frac{1}{2}) = \frac{1}{3}$$

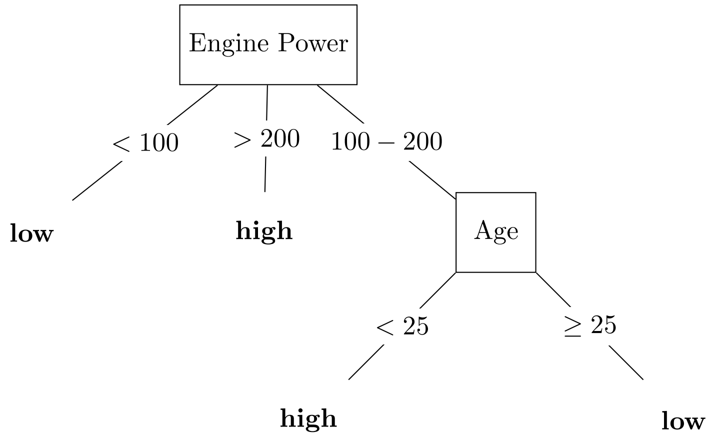
$$R(Age) = \frac{1}{2} \cdot I(\frac{1}{3}, \frac{2}{3}) + \frac{1}{2} \cdot I(\frac{2}{3}, \frac{1}{3}) = I(\frac{1}{3}, \frac{2}{3}) = -\frac{1}{3}\log_2(\frac{1}{3}) - \frac{2}{3}\log_2(\frac{2}{3}) = \frac{5}{6}$$

Gains are then:

$$Gain(EnginePower) = 1 - \frac{1}{3} = \frac{2}{3}$$

$$Gain(Age) = 1 - \frac{5}{6} = \frac{1}{6}$$

So the first split should be on attribute *Engine Power.* After that, splitting on *Age* will result in a clean split with no entropy left.



Exercise 10.2 (Best practices in ML)

When doing machine learning, it is good practice to split the dataset into a training/validation/test set.

- Which subset(s) should you use for the following tasks:
  (a) training models (R & D)[1]
  (b) guard against overfitting (R & D)
  (c) model selection (R & D)
  (d) progress reports (R & D)
  (e) train the model (product)[2]
  (f) evaluating the model (product)
- Which of these subsets should always be fixed a priori (before even looking at the data)?

Solution:

-

task	**phase**	**training set**	**validation set**	**test set**
training models	R & D	yes	no	no
guard against over-fitting	R & D	no	yes	no
model selection	R & D	no	yes	no
progress reports	R & D	no	yes	no
training the model	product	yes	maybe	no
evaluating the model	product	no	no	yes

- The test set should always be fixed a priori (and used only once, to evaluate the final model). Instances in validation/training sets may vary during R & D. However, having a sparsely-used, fixed subset that acts as a 'pseudo' test-set can be useful (e.g. for internal progress reports). Also, if examples in the validation set are frequently used (e.g during training, hyper-parameter tuning, etc.) failing to detect overfitting is a real risk.

[1] R & D: During research and development
[2] product: For the final product/publication

Foundations of Artificial Intelligence

Prof. Dr. J. Boedecker, Prof. Dr. W. Burgard, Prof. Dr. F. Hutter, Prof. Dr. B. Nebel,
Dr. rer. nat. M. Tangermann
T. Schulte, M. Krawez, R. Rajan, S. Adriaensen, K. Sirohi
Summer Term 2020

University of Freiburg
Department of Computer Science

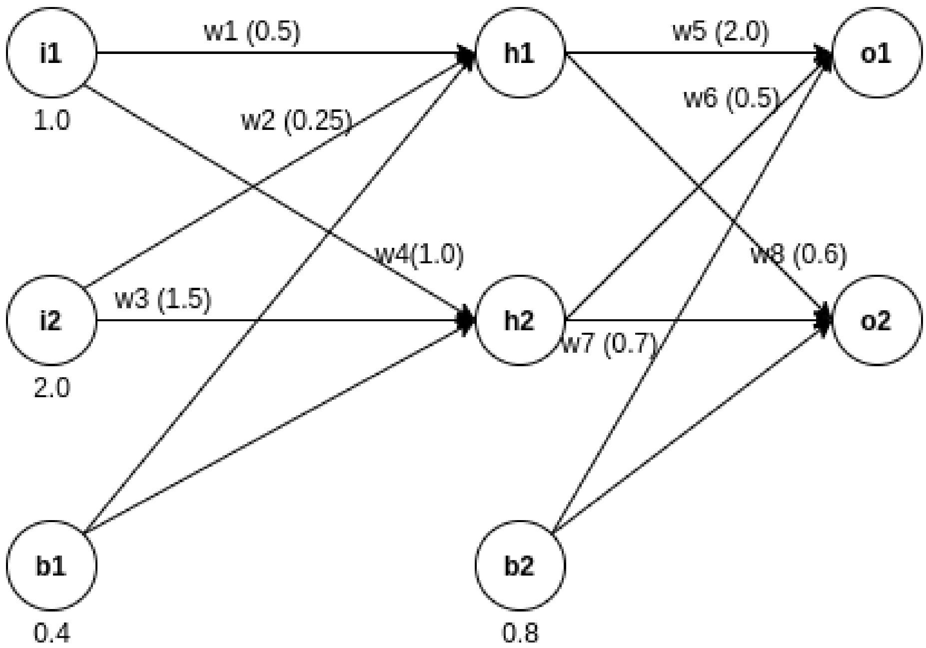
Exercise Sheet 11 — Solutions

Exercise 11.1 (Multi Layer Perceptron)

Given below is a structure of a multilayer perceptron with 2 inputs (i1 and i2), 2 hidden layers (h1 and h2), biases(b1 and b2) and one output layer(o). Each hidden and output layer output is activated using logistic sigmoid activation function:. :

a) Perform one forward pass with the values of parameters depicted with every variable in the network and calculate the outputs(o1,o2).

b) Calculate the mean square error given value of outputs (o1,o2) as (2.0,4.0).



Solution:

a) To solve the output of the network, the output of each node has to be calculated. Let the overall input of *h1* is denoted by *inh1* which is calculated as follows.

$inh1 = w1 * i1 + w2 * i2 + b1$
$inh1 = 0.5 * 1.0 + 0.25 * 2.0 + 0.4 = 1.4$

This is activated using logistic sigmoid activation function to give output, $outh1$.

$outh1 = 1/(1 + e^{-1.4}) = 0.8022$

Similarily we calculate the output of the node $h2$.
$inh2 = w4 * i1 + w3 * i2 + b1$
$inh2 = 1.0 * 1.0 + 1.5 * 2.0 + 0.4 = 4.4$

$outh2 = 1/(1 + e^{-4.4}) = 0.9878$

The output $outo_1$ is calculated as follows:
$ino_1 = w5 * outh1 + w6 * outh2 + b2$
$ino_1 = 2.0 * 0.8022 + 0.5 * 0.9878 + 0.8 = 2.8983$
$outo_1 = 1/(1 + e^{-2.8983}) = 0.9477$

The output oto_2 is calculated as follows:
$ino_2 = w8 * outh1 + w7 * outh2 + b2$
$ino_2 = 0.6 * 0.8022 + 0.7 * 0.9878 + 0.8 = 1.97278$
$outo_2 = 1/(1 + e^{-1.97278}) = 0.87791$

b) Mean sqaure error is defined as:

MSE $= \frac{1}{n} \sum_{i=1}^{n} (o_i - outo_i)^2$.
MSE $= (1/2) * ((o_1 - outo_1)^2 + (o_2 - outo_2)^2)$

Given values of o_1 and o_2 are 2.0 and 4.0 respectively.

MSE $= 0.5 * ((2.0 - 0.9477)^2 + (4.0 - 0.87791)^2)$
MSE $= 5.4273$

Exercise 11.2 (Convolutional Neural Network)

Given below is a sequence of operations in a small convolutional neural network(CNN) which takes input of shape (48 x 48 x 3). Calculate the output size and number of trainable paramters after each layer of the network.
conv1 and conv2 are the convolutional layers with given filter size f , stride s and output feature size o.

layer	**shape**	**parameters**
Input	(48,48,3)	0
conv1(f=3,s=1,o=8)		
conv2(f=5,s=1,o=16)		

Solution:

The output size of a convolution layer with input size nxn, filter size fxf, stride s and padding p is given by:

$$out = \frac{n-f-2p}{s} + 1$$

as no padiding is provided, the padding is 0.

For a filter size of fxf, previous feature size i, output feature size o, the number of learnable parameters are calculated as:

$$num_{param} = (i * f * f * o) + o.$$

For conv1, it will lead to:

$$num_{param} = (3 * 3 * 3 * 8) + 8 = 224.$$

Using these formulas, the complete table is as follows.

layer	**shape**	**parameters**
Input	(48,48,3)	0
conv1(f=3,s=1,o=8)	(46,46,8)	224
conv2(f=5,s=1,o=16)	(42,42,16)	3216