Last Name: First Name:

Matriculation Number:

Seat:

Exam
Artificial Intelligence 2

August 2, 2022

	To be used for grading, do not write here											
prob.	1.1	1.2	2.1	2.2	3.1	3.2	4.1	4.2	4.3	5.1	Sum	grade
total	7	10	8	14	15	10	10	6	6	9	95	
reached												

The “solutions” to the exam/assignment problems in this document are supplied to give students a starting point for answering questions. While we are striving for helpful “solutions”, they can be incomplete and can even contain errors even after our best efforts. In any case, grading student’s answers is not a process of simply “comparing with the reference solution”, therefore errors in the “solutions” are not a problem in this case.
If you find “solutions” you do not understand or you find incorrect, discuss this on the course forum and/or with your TA and/notify the instructors. We will – if needed – correct them ASAP.

In the course Artificial Intelligence I/II we award 5 bonus points for the first student who reports a factual error (please report spelling/-formatting errors as well) in an assignment or old exam and 10 bonus points for an alternative solution (formatted in LaTeX) that is usefully different from the existing ones.

1 Probabilities

Problem 1.1 (Python) 7 pt

Consider the Python program below.

1. Which mathematical function does the method `foo` compute? 4 pt
2. Assume random variables X with domain $\{0, \dots, m-1\}$ and Y with domain $\{0, \dots, n-1\}$. Assume the following Python objects 3 pt
   - C holds the conditional probability distribution $P(X|Y)$, i.e., $C[i][j] = P(X = i|Y = j)$.
   - D holds the probability distribution $P(Y)$, i.e., $D[j] = P(Y = j)$.

   Complete the definition of E in the program below in such a way that it holds the probability distribution $P(X)$, i.e., $E[i] = P(X = i)$.
   (Hint: This requires relatively little code.)

```
def foo(a,b):
    l = len(a)
    m = len(a[0])
    n = len(b[0])
    res = []
    for i in range(l):
        row = []
        for j in range(n):
            s = 0
            for k in range(m):
               s += a[i][k] * b[k][j]
            row.append(s)
        res.append(row)
    return res


E =
```

Solution:

1. The matrix product $a \cdot b$.
2. `E=foo(C,[[x] for x in D])`.

   We also accepted `E=foo(C,D)`.

Grading:

1. 2 points if describing the loop structure correctly without mentioning matrix multiplication.
2. Partial credit depending on what students come up with.

Problem 1.2 (Calculations) 10 pt

Assume three Boolean random variables X, Y, Z, whose joint distribution $P(X, Y, Z)$ is given by

x	y	z	$P(X = x, Y = y, Z = z)$
$true$	$true$	$true$	a
$true$	$true$	$false$	b
$true$	$false$	$true$	c
$true$	$false$	$false$	d
$false$	$true$	$true$	e
$false$	$true$	$false$	f
$false$	$false$	$true$	g
$false$	$false$	$false$	h

1. In terms of a, b, c, d, e, f, g, h, give $P(X = true, Y = false)$. 3 pt
2. In terms of a, b, c, d, e, f, g, h, give $P(X = true|Y = false)$. 3 pt
3. Which of the following are true if Y and Z are conditionally independent given X? 4 pt
   (a) $a + b + e + f = a + c + e + g$
   (b) $a = (a + b) \cdot (a + c)/(a + b + c + d)$
   (c) $e = (f \cdot g)/h$
   (d) $a = e$

Solution:

1. $P(X = true, Y = false) = P(X = true, Y = false, Z = true) + P(X = true, Y = false, Z = false) = c + d$
2. $P(X = true|Y = false) = P(X = true, Y = false)/P(Y = false) = (c + d)/(c + d + g + h)$.
3. Only (b).

Grading:

1. 1.5 if similar
2. 1.5 if similar
3. Full points if only (b) marked. -1 point if (b) not marked and for every other marked statement. Thus, writing "none" or similar yields 2 points, but leaving the question blank is 0 points. If some statements are marked as true, the grader should cross out the unmarked ones to avoid manipulation during review.

---

2 Bayesian Reasoning

Note: When working with an upper case Boolean random variable X, you may abbreviate the event $X = true$ by the corresponding lower-case letter x. If you do that, make sure the distinction between upper and lower case letters is clear in your writing.

Problem 2.1 (Bayes' Rule) 8 pt

Assume you are trying to predict whether a particular topic comes up in an exam. You have collected the following data:

- 30% of all topics come up in the exam.
- 50% of all topics come up in the assignments.
- If a topic comes up in an exam, it was covered by an assignment 60% of the time.
- If a topic comes up in an exam, it also came up in a recent exam 80% of the time.

You model this situation using 3 Boolean random variables E (comes up in exam), A (covered by assignments), and R (came up in a recent exam).

1. By filling in the gaps below, state for each number in the text above, which probability it describes. 2 pt

   (a) $P(\quad E = true!) = 0.3$

   (b) $P(\quad A = true!) = 0.5$

   (c) $P(\quad A = true|E = true!) = 0.6$

   (d) $P(\quad R = true|E = true!) = 0.8$

2. When modeling this situation, is it reasonable to assume that A and R are stochastically independent? Why (not)? 2 pt

3. The topic you are interested in was covered by an assignment. Using Bayes' rule, determine the probability that it will come up in the exam. 4 pt

---

Solution:

1. $P(E = true) = 0.3$, $P(A = true) = 0.5$, $P(A = true|E = true) = 0.6$, $P(R = true|E = true) = 0.8$

2. No. Background knowledge indicates that A and R are often highly correlated (even if the details cannot be ascertained from the data given).

3. $P(E = true|A = true) = P(A = true|E = true) * P(E = true)/P(A = true) = 0.6 * 0.3/0.5 = 0.36$

Grading:

1. 0.5 points each.
2. 1 point each for answer and argument.
3. 2 points for the formula, 1 for plugging in the values, 1 for the result.

Problem 2.2 (Bayesian Networks) 14 pt

Consider the following situation:

- Covid can cause a sickness and/or fever.
- Fever itself is dangerous and can cause sickness.
- Tests can detect Covid. But a false-positive Covid test may cause sickness via a kind of Placebo effect.
- There are no other causal relationships.

You want to model this situation using Boolean random variables C (Covid infection), F (fever), S (sickness), and T (positive Covid test).

1. Give a good variable ordering for forming a Bayesian network for this situation. 3 pt
2. Give the resulting network. 3 pt
3. You have a fever, feel sick, and have tested positive for Covid. Now you want to determine if you have Covid. What are the query, evidence, and hidden variables? 2 pt
4. Assume your network is $C \to F \to T \leftarrow S$ (which *may or may not* be a correct solution to the above question). Which probabilities are stored in the conditional probability table of node T? 2 pt
5. Again using the network $C \to F \to T \leftarrow S$, give the formula for

$$P(C = true, T = true, S = true)$$

in terms of the entries of the conditional probability table of that network. 4 pt

Solution:

1. Causes should occur before effects, so e.g., $CFTS$ or $CTFS$.
2. $C \to F \to S$ and $C \to T \to S$ and $C \to S$.
3. Query: C, evidence: F, S, T, hidden: none.
4. The probability distribution $P(T|F, S)$, i.e., $P(T = x|F = y, S = z)$ as a function of Booleans x, y, z.
5.

$$P(c, t, s) = P(c, t, s, f) + P(c, t, s, \neg f)$$

$$= P(c) \cdot P(f|c) \cdot P(s|f, c) \cdot P(t|c, f, s) + P(c) \cdot P(\neg f|c) \cdot P(s|\neg f, c) \cdot P(t|c, \neg f, s)$$

$$= P(c) \cdot P(f|c) \cdot P(s) \cdot P(t|f, s) + P(c) \cdot P(\neg f|c) \cdot P(s) \cdot P(t|\neg f, s)$$

$$= P(c) \cdot P(s) \cdot \big(P(f|c) \cdot P(t|f, s) + P(\neg f|c) \cdot P(t|\neg f, s)\big)$$

Grading:

1. 2 point for minor mistake. 1 point if vaguely similar.
2. -0.5 points for each missing/false edge.
3. -0.5 per variable.
4. 1 point for the explanation, 1 for formal correctness.
5. -1 for every mistake. The last line of the solution is not necessary.

---

3 Markovian Reasoning

Problem 3.1 (Hidden Markov Models) 15 pt

Consider the following situation:

- You make daily observations about your business B. Each day business is either good (b_1) or bad (b_2).
- You know this is caused by the weather W, which can be rainy (w_1), cloudy (w_2), or sunny (w_3).
- You have previously obtained the following information:
  - when the weather is rainy, your business is good 36% of the time,
  - when the weather is cloudy, your business is good 84% of the time,
  - when the weather is sunny, your business is good 90% of the time,
  - half the time, the weather is the same as on the previous day,
  - when the weather changes from one day to the next, each change is equally likely.

You want to model this situation as a hidden Markov model with two families of random variables indexed by day number d.

1. Give the state and evidence variables and their domains. 2 pt
2. How can you tell that the sensor model is stationary here? 1 pt
3. Complete the following sentences:
   (a) The transition model T is given by the matrix 2 pt
   $$T = \begin{pmatrix} \quad\quad\quad\quad\quad\quad \end{pmatrix} \quad \text{where} \quad T_{ij} = P(W_{d+1} = w_j | W_d = w_i).$$
   (b) The sensor model S is given by the matrix 2 pt
   $$S = \begin{pmatrix} \quad\quad\quad\quad \end{pmatrix} \quad \text{where} \quad S_{ij} = P(B_d = b_j | W_d = w_i).$$
4. Let $\mathbf{w}$ be a 3-dimensional vector whose coefficients sum to 1. What is the intuitive meaning of the property $T \cdot \mathbf{w} = \mathbf{w}$? 2 pt
5. It was sunny yesterday, and your business is good today. You want to use filtering to obtain the probability distribution of today's weather.
   You proceed as follows:

(a) Give the recursive filtering equation for $f_{1:d+1}$. 1 pt

(b) Give the initial value $f_{1:0}$ to use in this case. 1 pt

(c) Give the diagonal sensor matrix O_1 to use in this case. 1 pt

(d) Compute the resulting distribution. 3 pt
Fully compute all values including the normalization. (This does not require approximations or a calculator.)

Solution:

1. State variables $W_d \in \{w_1, w_2, w_3\}$, evidence variables $B_d \in \{b_1, b_2\}$

2. The business-weather relation is the same for each day.

3. (a) $T = \begin{pmatrix} 0.5 & 0.25 & 0.25 \\ 0.25 & 0.5 & 0.25 \\ 0.25 & 0.25 & 0.5 \end{pmatrix}$

   (b) $S = \begin{pmatrix} 0.36 & 0.64 \\ 0.84 & 0.16 \\ 0.9 & 0.1 \end{pmatrix}$

4. $\mathbf{w}$ is a probability distribution of the weather that is a fixed point of the transition model, i.e., the distribution will stay the same when predicting the future.

5. We compute $f_{1:1}$ by applying the filtering equation once.

   (a) $f_{1:d+1} = \alpha(O_{d+1} \cdot T^t f_{1:d})$

   (b) $f_{1:0} = \langle 0, 0, 1 \rangle$

   (c) $O_1 = \begin{pmatrix} 0.36 & 0 & 0 \\ 0 & 0.84 & 0 \\ 0 & 0 & 0.9 \end{pmatrix}$

   (d) $f_{1:1} = \alpha \cdot O_1 \cdot T \cdot f_{1:0} = \alpha\langle 0.36 \cdot 0.25, 0.84 \cdot 0.25, 0.9 \cdot 0.5 \rangle$
   $= 4/3\langle 0.09, 0.21, 0.45 \rangle = \langle 0.12, 0.28, 0.6 \rangle$.

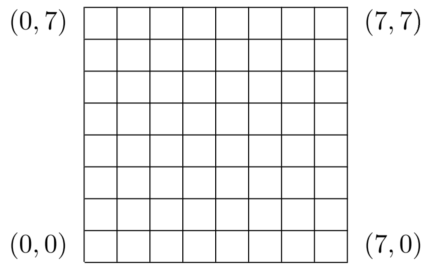
Grading:

1. 1 point each; -0.5 for minor mistakes
2. no partial credit
3. -0.5 for minor mistakes
4. 1 point for stationary/fixed point/etc., 1 point for $\mathbf{w}$ being a distribution for the weather.
5. (a) -0.5 for mistakes
   (b) -0.5 for mistakes
   (c) -0.5 for mistakes
   (d) 1 point for plugging in correctly, 1 point for the calculation, 1 point for the normalization.

Problem 3.2 (Utility and Decision Processes) 10 pt

Consider an agent moving on an 8×8 grid as indicated in the picture below.

The agent can move up, down, right, and left except where restricted by the edges of the grid. Every action results in moving one step in that direction with probability 75% and no move otherwise. The agent's goal is to get to the location $(7,7)$.



1. Choose an appropriate reward function and model this situation as a Markov Decision Process. 4 pt

2. Give an optimal policy π^*. 2 pt

3. Now ignore the rewards, and assume we use a fixed utility $i + j$ for the field (i, j). Compute the expected utility of moving up once in state $(1, 1)$. 2 pt

4. Now assume the agent is unable to tell whether an action resulted in a move or not. Explain informally how that would change the modeling. 2 pt

---

Solution:

1. One possible model is
   - $S = \{0, \ldots, 7\}^2$
   - $A((i,j)) = \{u, d, l, r\} \setminus E_i \setminus F_j$ where $E_0 = \{l\}$, $E_7 = \{r\}$, $F_0 = \{d\}$, $F_7 = \{u\}$ and $E_i = F_j = \emptyset$ otherwise
   - $P(s'|a, s)$ is 0.75 if s' is the result of moving a from s, 0.25 if $s' = s$, 0 otherwise
   - A typical choice is any function R that is high for $(7,7)$ and slightly negative for other states. E.g., $R(s) = 1$ for $s = (7,7)$ and $R(s) = -0.1$ otherwise.

2. Any policy that maps state $(7,7)$ to d or l and every other state to any legal action that is u or r. E.g., $\pi^*(s) = u$ if $u \in A(s)$, otherwise $\pi^*(s) = r$ if $r \in A(s)$, otherwise $\pi^*((7,7)) = d$.

3. $EU(u) = 0.75 \cdot U((1,2)) + 0.25 \cdot U((1,1)) = 2.25 + 0.5 = 2.75$.

4. We would need a POMDP. A state in the POMDP is a so-called belief state, a probability distribution for the MDP-state that the agent is in.

Grading:

1. 1 point each for states, actions, transition model, rewards. There are other reward functions that are technically correct and that the grader may have to be evaluated based on whether they lead an agent to the goal.
2. 1.5 points if minor mistake, 1 point if recognizable, 0.5 if something correct
3. 1 point for the formula, 1 for the calculation.
4. 1 point for POMDP, belief state, or similar, 1 point for using a probability distribution over states.

---

4 Learning

Problem 4.1 (Decision Trees and Lists) 10 pt

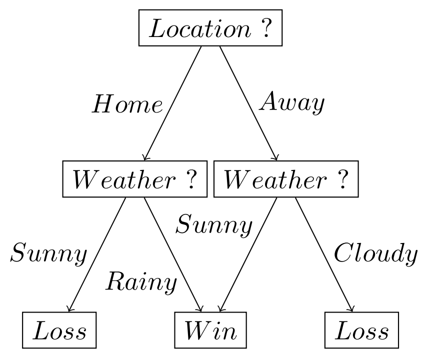
You observe the set of values below for 6 games of a sports team. You want to predict the result based on weather, location, and opponent.

#	Weather	Location	Opponent	Result
1	Rainy	Home	Weak	Win
2	Sunny	Away	Weak	Win
3	Sunny	Home	Strong	Loss
4	Sunny	Away	Weak	Win
5	Cloudy	Away	Strong	Loss
6	Sunny	Home	Strong	Loss

1. Draw the decision tree that arises from choosing attributes according to the order $Location, Weather, Opponent$. 3 pt
2. Give all minimal sets A of attributes such that $A \succ Result$. 3 pt
3. How can such a minimal set A be exploited to find a decision tree? 1 pt
4. You want to build a decision list for the result using tests with literals of the form $attribute = number$. Give the shortest possible decision list. 1 pt
5. Now stop using the above observations. Instead, assume some set of observations for which a decision list exists, and assume that the determination $Weather, Location \succ Result$ holds. In the worst case, what is the number of tests in the shortest decision list? 2 pt

Solution:

1. The tree is



2. $\{Weather, Location\}$ and $\{Opponent\}$

3. Only the attributes in A are needed in the tree. So smaller sets yield smaller trees.

4. If $Opponent = Weak$ then $Result = Win$ else $Result = Loss$.

5. 5 (not longer because we can use the determination and use one test for every one of the 6 combinations of values (no test is needed for the last one because we can use the final else-case); not shorter because we might indeed need those 6 cases)

Grading:

1. -0.5 for a decision list, -0.5 if for missing leaf node labels, or similar representation errors, -1 for including the redundant attribute, -0.5 for other mistakes; at least 2 points for an essentially correct solution, at least 1.5 points if recognizably similar to solution, 1 point if some decision tree.
2. 1.5 for a correct set. If more than 2 sets given, -0.5 for each additional set.
3. 0.5 points for each sentence.
4. -0.5 for small mistakes or for a non-minimal list.
5. 1 point for 6. No other partial credit.

Problem 4.2 (Statistical Learning) 6 pt

You observe the values below for 20 games of a sports team. You want to predict the result based on weather and opponent.

Weather	Opponent	Number of wins	losses
Rainy	Weak	3	1
Cloudy	Weak	0	1
Sunny	Weak	4	2
Rainy	Strong	0	2
Cloudy	Strong	2	3
Sunny	Strong	0	2

1. What is the hypothesis space for this situation, seen as an inductive learning problem? 2 pt
2. Explain whether we can learn the function by building a decision tree. 2 pt
3. To apply Bayesian learning, we model this situation as a Bayesian network $W \rightarrow R \leftarrow O$ using random variables W (weather), O (opponent), and R (game result). What are the resulting entries of the conditional probability table for the cases 2 pt
   (a) $P(W = rainy) =$ 3/10!
   (b) $P(R = win|O = weak) =$ 7/11!

Solution:

1. The set of functions $\{Rainy, Cloudy, Sunny\} \times \{Weak, Strong\} \rightarrow \{Win, Loss\}$.
2. It does not. Even all attributes together, i.e., *Weather* and *Opponent*, do not determine the result. So no decision tree exists.
3. $P(W = rainy) = 3/10$ and $P(R = win|O = weak) = 7/11$

Grading:
1. 1 point for a set of functions relating attributes, 1 for correctness
2. 1 point for the answer, 1 for the explanation.
3. 1 point each

Problem 4.3 (Support Vector Machines) 6 pt

Consider a set of points $\mathbf{x} = \langle \mathbf{x}_1, \mathbf{x}_2 \rangle$ in $\mathbb{R}^2$ that are classified as either $y = +1$ or $y = -1$.

1. Give the hypothesis space for finding a linear separator. 2 pt
2. Given a linear separator $h(\mathbf{x})$, which formula computes the classification y of a vector $\mathbf{x}$? 1 pt
3. What is the point of transforming a dataset into a higher-dimensional space? 1 pt

4. In this context, briefly discuss the usefulness of the transformation $F(\langle \mathbf{x}_1, \mathbf{x}_2 \rangle) = \langle \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_1 + \mathbf{x}_2 \rangle$. 2 pt

---

Solution:

1. The set of functions $\mathbf{w} \cdot \mathbf{x} + b$ for real numbers $\mathbf{w}_1$, $\mathbf{w}_2$, b. Alternatively, one can use $\mathbb{R}^3$ with some explanation that it holds the tuples ($\mathbf{w}_1$, $\mathbf{w}_2$, b).
2. $y = sgn(h(\mathbf{x}))$
3. It's possible that no linear separator exists for the original dataset, but a linear separator exists for the transformed dataset in the bigger space.
4. It is not useful because it is linear. If a linear separator exists afterwards, one existed before as well.

Grading:

1. 1 point for the general idea, 1 point for formal correctness.
2. No partial credit
3. 1 point for the explanation.
4. 1 point for the claim, 1 for the argument.

---

5 Natural Language Processing

Problem 5.1 (Information Retrieval) 9 pt

Consider the following three texts

- d_1: "The man is tall."
- d_2: "The tall man sees the woman."
- d_3: "The woman shouts."

Let $D = \{d_1, d_2, d_3\}$.

Below we use alphabetical order for the vector components:

is, man, sees, shouts, tall, the, woman

Simplify all results as much as possible without introducing approximate values.

1. What is the idea of cosine similarity for comparing a query against the documents in D? 2 pt
2. Give the vector $tf(\_, d_2)$. 2 pt
3. Give the vector $idf(\_, D)$. 2 pt
4. For $d \in D$ and a word t, give the definition of $tfidf(t, d, D)$. 1 pt
5. What is the benefit of using $tfidf$ instead of tf for using cosine similarity? 2 pt

Solution:

1. The query and each document are represented as a vector representing word frequencies. Vectors pointing in the same directions are considered similar. So the documents can be ranked by the angle between them and the query.
2. $tf(\_, d_2) = \langle 0, 1/6, 1/6, 0, 1/6, 1/3, 1/6 \rangle$.
3. $idf(\_, D) = \log_{10}(3/\langle 1, 2, 1, 1, 2, 3, 2 \rangle) = \langle k, l, k, k, l, 0, l \rangle$ with $k = \log_{10} 3$ and $l = \log_{10} 1.5$.
4. $tfidf(t, d, D) = tf(t, d) \cdot idf(t, D)$
5. $tfidf$ gives more weight to words that occur in fewer documents. Otherwise, many documents would falsely appear similar just because the most common words appear in most of them.

Grading:

1. 1 point for representing documents and query as vectors, 1 for the ranking by angle.
2. -0.5 per wrong entry
3. -0.5 per wrong entry
4. -0.5 for minor mistakes.
5. 1 point for the method, 1 for the goal.