i Front page

Examination paper for TDT4171 Methods in Artificial Intelligence

Date: 8. may 2025
Time: 0900-1300

Course contact: Helge Langseth
Present at the exam location: NO

Permitted examination support material: D: No printed or hand-written support material is allowed. A specific basic calculator is allowed.

OTHER INFORMATION

Read the questions carefully and make your own assumptions. In your answers, explain clearly what assumptions you have made and how you have understood or limited the assignment

If there are direct errors or omissions in the assignment set and you cannot make your own assumptions, please refer to the information about complaints regarding formal errors on the NTNU website “Explanation of grades and appeals”.

SPECIFIC INFORMATION FOR YOUR COURSE

Hand drawings:

For Question 1 you are meant to answer on handwritten sheets. **Other questions must be answered directly in Inspera.** At the bottom of the question, you will find a seven-digit code. Fill in this code in the top left corner of the sheets you wish to submit.

We recommend that you do this during the exam. If you require access to the codes after the examination time ends, click “Show submission”.

You are responsible for filling in the correct codes on the handwritten sheets. Therefore, read the cover sheet carefully. The Examination Office cannot guarantee that incorrectly completed sheets will be added to your assignment.

Weighting: Each question gives information about how many points it is worth. All subquestions inside a question count equally unless stated otherwise.

Withdrawing from the exam:

If you wish to submit a blank test/withdraw from the exam for another reason, go to the menu in the top right-hand corner and click “Submit blank”. This cannot be undone, even if the test is still open.

Access to your answers:

After the exam, you can find your answers under previous tests in Inspera. Be aware that it may take a working day until any hand-written material is available in “previous tests”.

1 Congestion modelling

Please model the following as a Bayesian network. **Your model is to be drawn on paper. See information on the front page of the examination paper.**

We are interested in traffic flow in a major city, in particular to be able to predict if there will be congestion or not. Our understanding of the domain is as follows:

- The traffic varies throughout the day, with congestion during rush-hour peaks, that happen around 0830 in the morning and 1500 in the afternoon.
- Traffic is typically more congested in nasty weather
- We may see more congested traffic if there is a major event in town, like a concert or sports event. These events are typically in the evenings.
- If there is some construction work going on, then this can cause large congestions. The effect is most pronounced during rush hours.
- In this example we will also assume that construction workers do not work in nasty weather. Furthermore, the workers only work between 0800 and 1600, and traffic is not prevented by the ongoing work outside this time period.

Make a Bayesian network describing this domain. For each variable you define, **you must also define the possible states that each variable can take.** You are asked to **make the Bayesian network so that it is as easy to understand as possible**. If you make additional assumptions about the domain then please write them down explicitly together with your model.

You should only make the qualitative structure (the directed acyclic graph), and you are **not asked to provide conditional probability tables**. However, make your model so that it would be as easy as possible for a domain expert to provide the required probabilities.

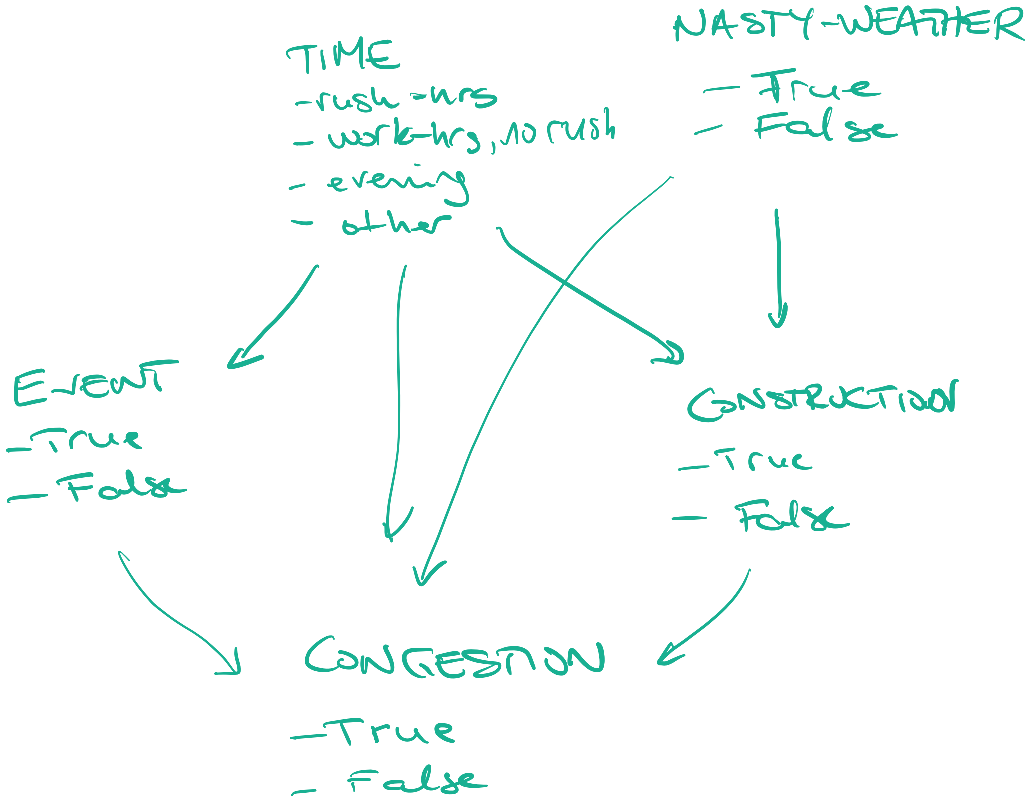
Maximum marks: 10

One solution is on the next page.

The main problematic issue in this domain is maybe the definition of time. We have mentioned “rush hours”, “evenings” and “working hours”. Rush hours are included in working hours, so state-space needs a bit of thinking. For full score we need to see potential to model all the different effects time have on the other variables. Just saying something «time is continuous variable that takes values from 0 to 24» is not full score because the implementation of the CPTs in this model will be hard.

States for other states are up for discussion, so be «understanding». If they are missing for the true/false variables (anything but the time-of-day) this should not lead to loss of points. If the states are completely missing for Time of day it is a loss of two points. Meaningful, but not-optiomal statespace (like continuous): 1 point loss.

Main goal is being able to model the effects encoded in the story. Meaningful graph with at least some relation to the story: At least 5 points.


TIME
-rush-hrs
- work-hrs, no rush
- evening
- other
NASTY-WEATHER
- True
- False
EVENT
- True
- False
CONSTRUCTION
- True
- False
CONGESTION
- True
- False

2 AI Foundation - True/False

Note! For this question, each sub-question gives +1 point if answered correctly, -1 if you make a mistake. No answer means 0 points. If the total from the question is negative, you will get 0 points.

Subquestion (a)

An AI model that passes the Turing test must have AGI (Artificial General Intelligence)

Select one alternative:

○ True

○ False ✓

Subquestion (b)

The Chinese Room argument is irrelevant when discussing modern large language models, because these models have attention layers

Select an alternative

○ True

○ False ✓

Subquestion (c)

Current state of the art mobile phones have sufficient computational power to learn the weights in a language model like ChatGPT from scratch.

Select an alternative

○ True

○ False ✓

Subquestion (d)

Development of AI models can lead to loss of privacy for human beings.

Select an alternative

○ True ✓

○ False

Subquestion (e)

Systems that exhibit Artificial Super Intelligence are already available to the general public.

Select an alternative

○ True

○ False ✓

Maximum marks: 5

3 AI Foundation - Issues with AI

Your responses in text fields are saved automatically

Give up to five potentially problematic issues related to the use of Artificial Intelligence systems. In your answer you can consider a future were these systems have become even more capable than they are today.

You can get up to 5 points from this question, with one point per distinct issue you mention. You do not have to elaborate on why each issue is problematic if this is obvious. Please stay to-the-point when providing your answer.

Fill in your answer here

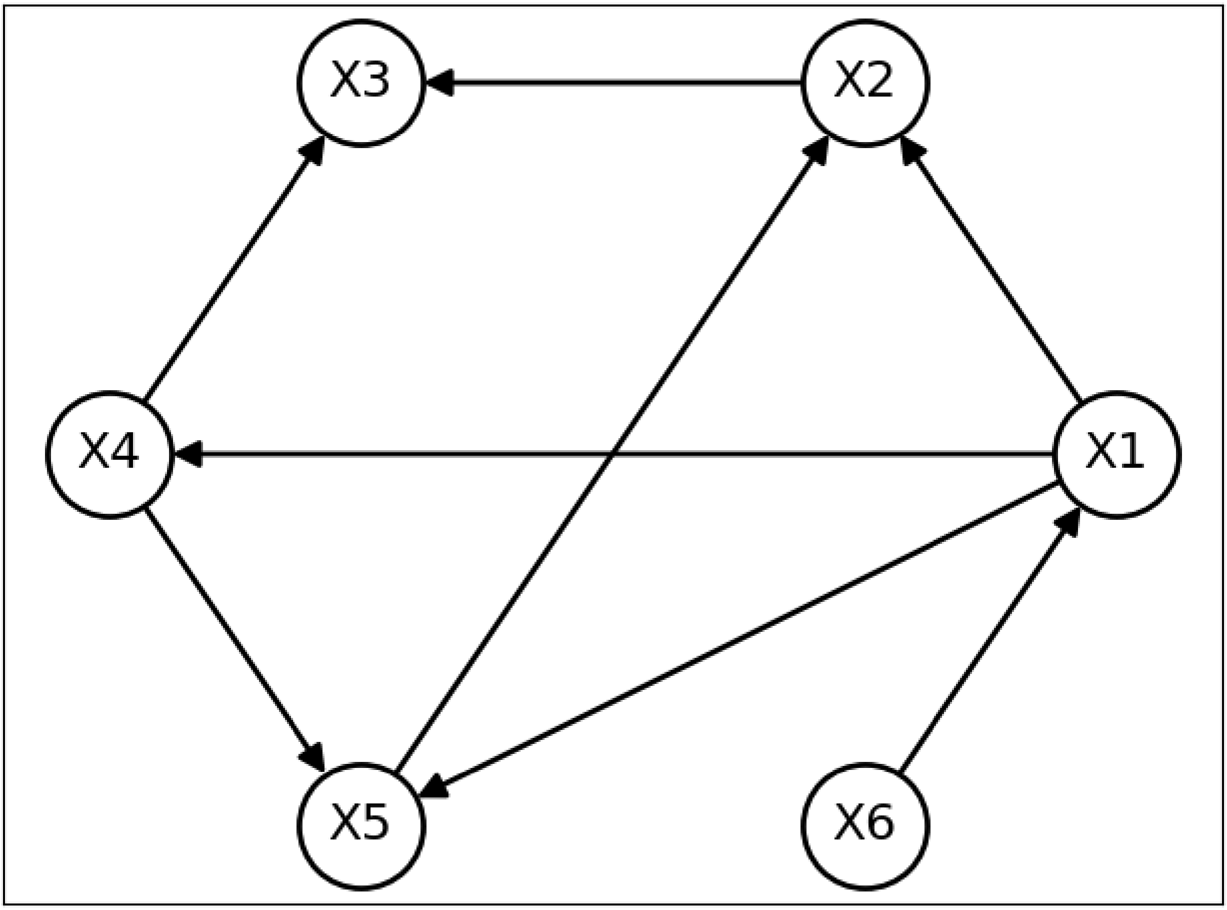
There are no “wrong” answers here as long as the issues are relevant. Anything like privacy/surevilance, autonomous weapons, loss of aoutonomy, loss of work, … should each give one point. We want to see diverse things here, though, so “I loose my job. My dad looses his job” would not be two points. No elaborations needed to get the points!

Maximum marks: 5

4 Conditional independence statements (1)

All questions below relate to this Bayesian network graph. We use set-notation when relating to conditioning sets of more than one variable, as an example we use "given {X_1 , X_5 }" to mean that both X_1 and X_5 are known.

Each question is scored with +1 point if answered correctly, -1 if answered wrongly, and 0 if no answer is given. If the sum from all these questions is negative, you will be awarded zero points (negative scores do not carry forward).



X_3 is independent of X_6

Select an alternative

○ True

○ False ✓

X_1 is independent of X_3 given X_5

Select one alternative:

○ True

○ False ✓

X_4 is independent of X_6 given $\{X_3, X_5\}$

Select an alternative

○ True

○ False ✓

X_5 is independent of X_6 given $\{X_1, X_3, X_4\}$

Select an alternative

○ True ✓

○ False

X_1 is independent of X_3 given $\{X_2, X_4, X_5, X_6\}$

Select an alternative

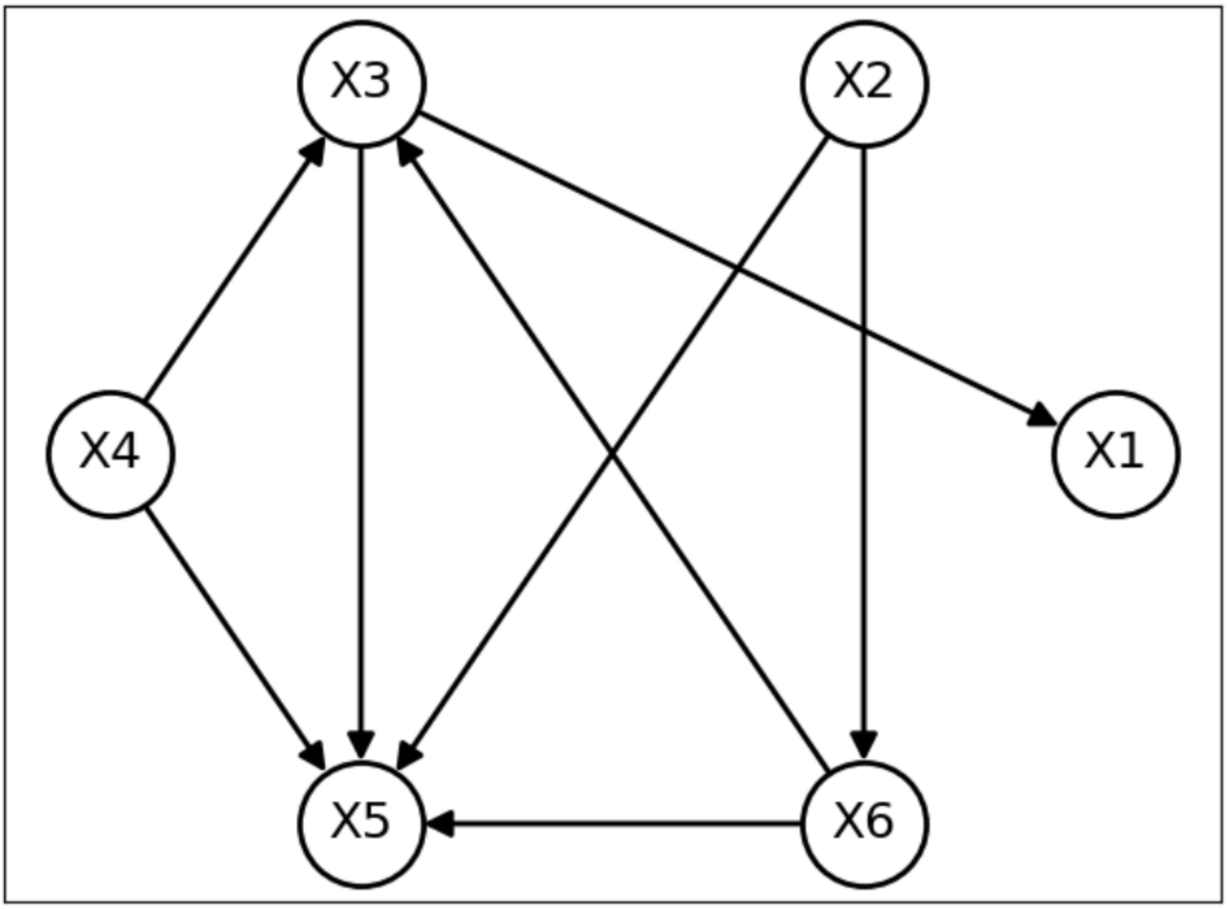
○ True ✓

○ False

Maximum marks: 5

5 Conditional independence statements (2)

All questions below relate to this Bayesian network graph. We use set-notation when relating to conditioning sets of more than one variable, as an example we use "given {X_1 , X_5 }" to mean that both X_1 and X_5 are known.



Each question is scored with +1 point if answered correctly, -1 if answered wrongly, and 0 if no answer is given. If the sum from all these questions is negative, you will be awarded zero points (negative scores do not carry forward).

X_1 is independent of X_5

Select an alternative

○ True

○ False ✓

X_1 is independent of X_6 given X_3

Select an alternative

○ True ✓

○ False

X_2 is independent of X_3 given {X_1 , X_5 }

Select an alternative

○ True

○ False ✓

X_1 is independent of X_4 given {X_2, X_3, X_6}

Select an alternative

○ True ✓

○ False

X_4 is independent of X_6 given {X_1, X_2, X_3, and X_5}

Select one alternative:

○ True

○ False ✓

Maximum marks: 5

6 Music Recommender System

Mood-Based Music Recommendation System

Note! For this question you will get **10 points** if every question is correctly answered. If you make at least one error, or leave at least one question unanswered, you will get **0 points**.

A music streaming service is developing a mood detection system to enhance their personalized recommendations. The system models a user's hidden **mood** (which cannot be directly observed) while tracking the type of songs the user chooses to play.

At each time *t*, the mood can be one of the two states

- Energetic (e)
- Relaxed (r)

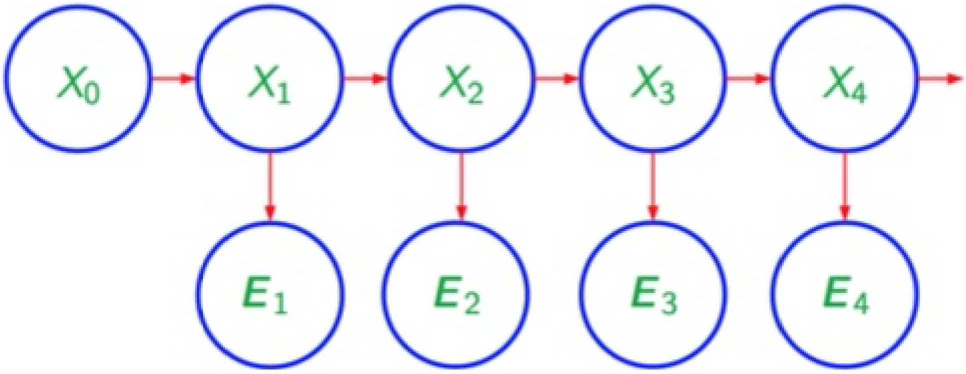
We do not observe the actual mood, but rather the observable **songs played** by the user. We assume that at each time *t* the user can play music from one of these two categories:

- Upbeat (u)
- Calm (c)

The service knows that users tend to remain in the same mood for periods of time but occasionally switch. Even when in a particular mood, users sometimes play songs that don't match their current mood (e.g., playing a calm song while in an energetic mood).

The recommender system will monitor the songs played by a user, and try to infer what the next song will be. The music streaming company feels that having this in place will serve as a good recommendation system: They will simply recommend what the system believe is coming next.

Notation: In the questions below we use X_t to denote the mood at time *t*, and E_t for the played song at the same time. Whenever we need a sum, we use standard notation to show what the sum is over. *Example:* $\sum_{x\in\{a,b\}} P(X=x)$ is the same as $P(X=a)+P(X=b)$. We use the shorthand *1 : t* do denote the sequence starting from time 1 and going up to and including time *t*. We will also use $\mathbf{X}_{1:t} = \{X_1, \ldots, X_t\}$ and similarly for $\mathbf{E}_{1:t}$. This is done to align the presentation here with the general presentation of Hidden Markov Models, as shown in this figure:



Sub-question (a):

Which assumptions - if any - are needed for the music streaming service to be able to use the Hidden Markov Model to provide recommendations to their users?

Select the correct alternatives:

□ The domain is stationary ✓

□ The domain has the Markov property ✓

□ The domain has a discount-factor $0 \leq \gamma < 1$.

□ The users are rational

Sub-question (b):

The music service wants to calculate the probability that the two first songs by a new user are upbeat, meaning the probability $P(e_1 = u, e_2 = u)$. Which of the alternatives below is correct? If you think that several equations are correct, you should choose the alternative that is simplest to calculate (requires fewest multiplications and additions)

Select one alternative to complete the equation $P(E_1 = u, E_2 = u) =$ ... :

○ $P(E_1 = u) \cdot P(E_2 = u)$

○ $\sum_{x \in \{e,r\}} P(E_1 = u | X_1 = x) P(E_2 = u | X_2 = x)$

○ $\sum_{x \in \{e,r\}} P(X_1 = x) P(X_2 = x | X_1 = x) P(E_1 = u | X_1 = x) P(E_2 = u | X_2 = x)$

○ None of the equations above gives the correct answer in general. ✓

Sub-question (c):

Which of the following inferences is most relevant for the music services in order to make next-song-recommendation to their users? If you think that several calculations are useful, choose the one that is closest to give a meaningful recommendation for the next song based on the observed behavior (the inference that when done, requires the fewest extra operations to provide the recommendation).

Select one alternative:

○ **Filtering** to find $\mathbf{P}(X_t|\mathbf{e}_{1:t})$.

○ **Smoothing** to find $\mathbf{P}(X_k|\mathbf{e}_{1:t})$ for some *k* with $\boldsymbol{0 \leq k < t}$.

○ **Prediction** to find $\mathbf{P}(X_{t+1}|\mathbf{e}_{1:t})$. ✓

○ **Most probable explanation:** $\arg\max_{\mathbf{x}_{1:t}} P(\mathbf{x}_{1:t}|\mathbf{e}_{1:t})$.

○ Neither of the inferences above are relevant for next-song-recommendation.

Maximum marks: 10

7 What user-group?

Consider the following situation:

- 50% of the players of a computer game are 40 years or older (henceforth called **old** users), and the remaining are below that age (henceforth called **young** users)
- 50% of the **old** users have completed Level 1 of the game after playing one hour
- 100% of the **young** users have completed Level 1 of the game after playing one hour

We learn that a randomly selected user who has played for one hour has completed Level 1.
What is the probability that the user is **old** ?

Give your answer as an integer percentage. If you for example believe the probability to be 0.508 then this is 50.8%, and you should answer 51 after rounding off to nearest integer.

P(user is **old**|User **completed** Level 1) = □ (33) %.

This question gives 5 points if answered correctly.

Maximum marks: 5

8 Color-game

In this task, each correct answer (each box you fill in) is worth 1 point. There is no penalty for wrong answers. If all answers are correct, you get an extra bonus so that your total score from this question will be 10 points. Give your answers with one digit after the decimal point, e.g., -2.3 or 4.1.

I am playing a game with a friend. At the beginning of the game I am choosing a color. Legal choices are **Red**, **Blue** and **Green**.

After I have made my decision, I flip a fair coin. Now, the winning of the game depends on the choice I make as well as the coin-flip. I know the following:

If I flip Heads, then the winning probability for each choice is

- Red: 10% chance of winning
- Green: 50% chance of winning
- Blue: 90% chance of winning

If I flip Tails, the probabilities are

- Red: 70% chance of winning
- Green: 50% chance of winning
- Blue: 30% chance of winning

Winning the game is worth 10 gold, and loosing is worth 0.

The first thing I wonder, is the expected worth of gold I win should I choose the different colors.

Subquestion 1:

Expected value of gold won if I choose **Red**: □ (4.0)

Expected value of gold won if I choose **Green**: □ (5.0)

Expected value of gold won if I choose **Blue**: □ (6.0)

Subquestion 2:

Next, I am interested in understanding how my winnings would improve if I could somehow trick my friend a bit. To fool him, I swap the fair coin with one that has **Tails on both sides**. Assuming I want to win as much gold as possible, and play rationally to achieve that goal, what is the expected amount of gold I win when using the fake coin?

Expected value of gold won when **playing optimally and flipping the fake coin**: □ (7.0)

Subquestion 3:

I feel bad about cheating, and rather want to negotiate with my friend. I am considering to offer him some gold if he accepts a rule-change. The change I propose is to alter the ordering of events so that I **first** flip the coin, and **knowing the outcome of the coin-flip** I thereafter choose my color.

Assume that my utility of gold is the same as the amount of gold, so for instance is the utility of getting 0 gold 0, the utility of getting 10 gold is +10.

How much, in expectation, should I value to know the outcome of the coin-flip prior to making my choice?

Value of Perfect Information: □ (2.0)

Maximum marks: 10

9 Rationality and subjectivity

Each correctly answered question below gives 1 point. Wrong answers do not lead to point-losses.

If two agents are both **rational**, then they must have the same underlying utility function.
Select one alternative:

○ True

○ False ✓

Rational behavior can be described using **mathematical** terms.
Select an alternative

○ True ✓

○ False

When a probability *P(x)* is **subjective** then two intelligent agents can hold different opinions about how the probability should be quantified even though both agents are rational.
Select an alternative

○ True ✓

○ False

If the probability distribution P(X=x, Y=y) over the two binary variables X and Y is **subjective**, then rationality does not require that the agent follows the axioms of probability calculus when, for instance, inferring its belief in the conditional event P(X=x|Y=y)

Select an alternative

○ True

○ False ✓

There always exists a utility-function that can be used to explain the behavior of a human being precisely.

Select an alternative

○ True

○ False ✓

Maximum marks: 5

10 Gradient descent learning

Your responses in text fields are saved automatically

Please note! INSPERA is unable to parse some of the text below on some computers, and will then show LaTeX commands (like "\boldsymbol" and "\mathbf") as plain text in red. If this happens to you, please read the equations by disregarding the red text, and you should still be able to read the Questions correctly.

Here you will get up to two points per sub-question. Please make your answers as short and to-the-point as possible.

Deep learning relies heavily on the **gradient descent** algorithm. The most important step of the algorithm can be described using this equation:

$$\mathbf{w}_{i+1} \leftarrow \mathbf{w}_i - \eta \cdot \nabla_{\mathbf{w}} L(\mathbf{w}_i)$$

Here we use w_i for the estimate of the weights at *the i*'th **iteration** of the algorithm, $L(\mathbf{w}_i)$ is the loss calculated over the dataset with model weights $\mathbf{w}_i$, and $\nabla_{\mathbf{w}} L(\mathbf{w}_i)$ is the gradient of this loss with respect to the weights, evaluated at the same point $\mathbf{w}_i$.

All the sub-questions below are related to this algorithm.

Subquestion (a):

- What is η in the equation above called?
- What is ballpark range of values for this parameter? (A fixed number is not required, but a description of a range, like "typically much larger than one", "typically much smaller than one, but larger than zero", "any positive value", "close to zero; can be positive or negative", etc.)

Fill in your answer here

This is the learning rate. It is typically a small positive number. Among the alternatives above, the correct choice is “typically much smaller than one but larger than zero”, but phrasing this is up to the student. Failing to give a reasonable range that shows lack of understanding of the purpose (e..g., saying any positive number) is a loss of at least one of the two available points.

Subquestion (b):

The gradient descent algorithm is **iterative***,* in that we do several rounds of learning, and make small refinements from one iteration to the next. Why is this necessary? The alternative you should compare to is to calculate the optimal weights directly (i.e., in a single pass through the training-data).

Fill in your answer here

We do not have access to the analytic solution (due to non-linearities in the transfer functions). Therefore we are unable to find the optimal weights directly. We can, however, find out how to IMPROVE, and that is what we keep on doing.

Subquestion (c):

The algorithm uses the **gradient** of the loss. Explain why the gradient is relevant. You should cover the following:

- **What property** of the gradient is the reason for it being used here?
- We **subtract** $\eta\nabla_{\mathbf{w}}L(\mathbf{w}_i)$ from the current weights $\mathbf{w}_i$ instead of **adding** it. Why?

Fill in your answer here

The gradient is a vector that points in the direction where the L-function increases the sharpest. This means that if we are to make a move of a given small length from the position in weight-space we are currently at, we should move in the direction of the gradient if we want to change L the most. Multiplying with -1 gives the direction with sharpest decrease. We minimize the loss, so that is the direction to move.

Subquestion (d):

One way to decide when to terminate the iterative scheme is to monitor how much we update the weights from one iteration to the next, i.e., to look at the length of the vector given by the difference $\mathbf{w}_{i+1} - \mathbf{w}_i$.

- Explain why this is **meaningful**.
- Does this stopping rule guarantee that the algorithm will find the **globally optimal** weights?

Fill in your answer here

If these weigts are close, the move we made was small. If the move was small, the length of the gradient vector is small, too. If the gradient vector is short, it is because the loss is changing slowly at our current position. -> It is (almost) a local optimum and we can stop. There is no guarantee of GLOBAL optimality.

Subquestion (e):

Let us look at the perceptron in particular. We assume that the transfer-function is the identity and that there is no offset ($b = 0$). The input is a single real number, and we will call that *x*. The calculated output is then given as *wx*, where *w* is the single trainable parameter of the perceptron.

Our training-data contains values of *x* paired up with the target-value of the response, *t,* meaning an observation in the dataset is *(x, t)*. We need to tune the weight *w* so that the generated output for an input *x* is as close to the target *t* as possible. You now get to know that the gradient-update step for this particular situation is calculated as

$$w_{i+1} \leftarrow w_i + \eta \sum_d x_d \cdot \mathbf{I}(t_d > w_i x_d).$$

Here, the sum indexed by *d* is over the training-data, $x_d \text{ and } t_d$ are the input and targeted output, respectively, for observation *d* and $\mathbf{I}(\phi)$ is a function that returns +1 if the proposition ϕ is true, and -1 otherwise.

Which loss-function $L(w)$ is used to result in this update-rule?

Fill in your answer here

This is the L1 loss (also known as mean absolute error, MAE, or something to that extent). Math formulation as sum of absolute difference between target at calculated is OK, too. MSE, squared error, L2 etc is wrong and gives zero points

Maximum marks: 10

11 Overfitting

Note! The sub-questions are scored by +1 point for correct answer, -1 for a wrong answer, and 0 points if you do not answer. If your total score from these five subquestions is negative, you will instead be awarded zero points.

Sub-question (a):

Overfitting in machine learning means that a machine learning model is trained using both expert knowledge as well as training data, while **underfitting** means that only information from the training data is used during training.

Select one alternative:

○ True

○ False ✓

Sub-question (b):

Weight-regularization as used in deep learning (when a term that increases with the length of the weight-vector is added to the loss) is a technique to avoid overfitting.

Select an alternative

○ True ✓

○ False

Sub-question (c):

The reason deep learning typically works better than a single perceptron in terms of getting a lower **training loss** is that the gradient descent algorithm scales well with respect to the number of observations in the training data, and therefore avoids overfitting.

Select an alternative

○ True

○ False ✓

Sub-question (d):

When one is to make a deep learning model to classify images, it is common to use convolutions. Using a convolutional layer ensures that the same mathematical operations are done on all parts of the input-image, and will reduce overfitting compared to a densely connected layer.

Select an alternative

○ True ✓

○ False

Sub-question (e):

When training a deep learning model, it is common to separate the available data into three parts: training-data, validation-data, and test-data. While training on the **training-data**, we monitor overfitting tendency by checking the system's ability (e.g., in terms of loss) on the **validation-data**. To estimate the system's ability to generalize on unseen data, we use the **test-data**.

Select an alternative

○ True ✓

○ False

Maximum marks: 5

12 Word embeddings and attention

Note! Each of the following sub-questions gives 1 point if answered correctly, and -1 point if answered incorrectly. If you do not give an answer you get 0 points. The minimum points obtainable from the question in total is zero points. **Importantly,** if you have answered all subquestions correctly, you will get **5 bonus-points**. Hence, the maximum score from this question is 10 points.

In this question we assume we have access to some word-embeddings in 3 dimensions. The embeddings are as follows:

- Man: (1, 2, 3)
- Woman: (2, 2, 3)
- Uncle: (3, 4, 5)
- Spain: (-1, -1, 1)
- Madrid: (-1, 1, 3)
- Norway: (-1, -2, 0)
- Oslo: (-1, 0, 2)
- Goat-cheese: (3, -2, 0)
- Food: (3, -1.5, 0.5)

Unless stated otherwise, we will use Eucledian distance in the following, hence the distance between the two vectors representations of Man and Woman is $\sqrt{(1-2)^2+(2-2)^2+(3-3)^2}=1$, etc.

Sub-question (a):
One of the major reasons why word embeddings have become popular in Deep Learning applications is that the embeddings encode semantic meanings of the words that cannot be captured by one-hot-encodings.
Select one alternative

○ True ✓

○ False

Sub-question (b):
When one learns word embeddings from data, it is very common to assume the unigram word model to reduce computational complexity.
Select one alternative

○ True

○ False ✓

Sub-question (c):

We do not have the embedding for the word **Italy**. A friend has come up with a handfull of suggestions. Which of the following representations are most meaningful, given the other embeddings.

Select one alternative:

○ (-1, -2, 0)

○ (-1, -2, 1) ✓

○ (2, 1,3)

○ There is not a single embedding out of the three above that makes more sense than all the others

Sub-question (d):

There is no information about the embedding for the word **aunt**, but we can estimate it by first calculating how to change a word of "**masculinity**" to one of "**femininity**". Using this strategy, what is a natural estimate for the embedding of **aunt**?

Select one alternative

○ None of the provided numerical answers captures the meaning of **aunt** correctly

○ (3.0 ,4.0 ,5.0)

○ (4.0, 4.0, 5.0) ✓

○ (2.0, 2.0, 3.0)

○ (2.5, 3.0, 4.0)

Sub-question (e):

We will now consider how a query in the sentence "Food from Spain" is handled by the attention-mechanism so central in the Transformer-model. We will consider the effect on the word "Food", and assume the query is the one-dimensional vector **q**=(2).

For each word in the sentence we get **three** vectors in the list below: The key-vector and the value-vector associated with the query in addition to the embedding

- **Food**: Key = (-3.0), Value = (1, 0.5, 0.5). Embedding = (3, -1.5, 0.5)
- **from:** Key = (-3.0), Value = (2, -2, -0.5). Embedding = (0, 0, 0)
- **Spain:** Key = (3.0), Value = (0, 0.5, 0.5). Embedding = (-1, -1, 1)

What is the embedding related to **food** after attending to the full sentence?

Remember that the process is that each word's importance first is calculated by taking the inner-product between query and key, and the attention weights are found by taking a softmax on the importance vector. The embedding is updated by adding the value vectors weighted by their respective attention-weights.

To simplify the calculations please note that we use at most one digit after the decimal point in our answers, and in this example it therefore suffices to approximate the softmax by rounding off each output to the nearest integer.

Select one alternative

○ (-1, -1, 1)

○ (0, 0, 0)

○ (3, -2, 0.5)

○ (3, -1.5, 0.5)

○ (3, -1, 1) ✓

○ (4, -1, 1)

○ All the numerical answers are wrong

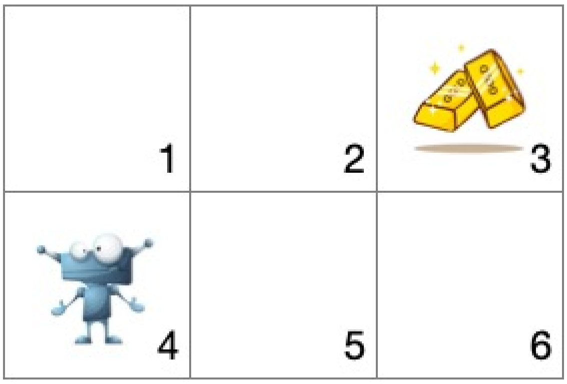
Maximum marks: 10

13 Which algorithm?

A robot is going to navigate a maze. It wants to do that in an optimal way, which means to accumulate reward over infinite time, using a discount-factor γ that is between 0 and 1. In this example, the reward is obtained from reaching the position in the maze that holds gold, which is worth +1 reward. Each hunt for gold is called an episode. An episode ends whenever the robot finds gold, or if the robot for some reason is destroyed. After an episode ends, the robot is put back to its starting position, and immediately starts a new episode. This loop is repeated forever.

To achieve its goal of accumulating lots of reward, the robot will develop a plan regarding how to behave when moving around in the maze. The following five sub-questions are about what algorithm to use to plan optimally in different situations. If it is possible to use the *Value Iteration* algorithm for sequential decision making, this is preferred by the robot, as it means that the robot does not have to do exploration in the maze. It can just move optimally according to the policy that was found prior to start of the episode. If the rules of the maze do not permit using that algorithm, the robot can consider *Q-learning* as a solution strategy for reinforcement learning. If that also does not work, the robot is out of luck (and may need to take other courses than "TDT4171 Methods in AI" to expand its understanding of optimal decision-making). When the solution-alternatives below refer to some algorithm "working", it means that the assumptions underlying that algorithm are fulfilled, and therefore the algorithm is guaranteed to (eventually) succeed. An algorithm "not working" means that an underlying assumption is violated.

The figure shows an example of the problem. For simplicity, the figure shows an extremely small maze, but think of the problem as rather being of much bigger size so that planning is computationally challenging.



Note! There is no penalty for wrong answers in this question. All sub-questions score equally, with up to 2 points each.

Sub-question (a):

At all points in time the robot knows its position in the maze. It knows what actions are available to it, and even though the effect of an action is stochastic, the robot knows the probabilistic model for what the effect of each action in each state is. That is, the transaction distribution that is sometimes written as $P(s'|a, s)$, is known to the robot.

Each time the robot enters a "cell" in the maze it will collect a reward based on its location. The robot knows these rewards in advance, meaning that the location of the gold (the source of the +1 reward) is known. In the example shown in the figure, we thus assume that

- The robot can see the IDs of each cell (e.g., it is now in location 4 and knows that).
- It knows that it can decide to go up, down, left or right.

- It knows the probabilistic effect of these actions.
- It knows that the gold is in location 3.

Which statement about algorithms available to the robot to optimize its behaviour is correct?
Select one alternative:

○ The robot can solve the problem using MDP/“Value Iteration”. ✓

○ MDP/“Value Iteration” does not work, but RL/“Q-learning” will work.

○ Neither MDP/"Value Iteration" nor RL/"Q-learning" will work in this situation because both algorithms make assumptions that are violated in this setting.

Sub-question (b):
The robot **looses its ability to differ between the locations** in the environment. In the figure above, you can think of this as the cell IDs being hidden to the robot. The robot nevertheless wants to behave optimally.

Which statement is correct?
Select one alternative

○ The robot can solve the problem using MDP/“Value Iteration”.

○ MDP/“Value Iteration” does not work, but RL/“Q-learning” will work.

○ Neither MDP/"Value Iteration" nor RL/"Q-learning" will work in this situation because both algorithms make assumptions that are violated in this setting. ✓

Sub-question (c):
The robot is repaired again, and has the same abilities as in part **(a)**. However, the rules of the maze is now changed: At each time-step, at just the moment when the robot makes its move, the **gold can be moved** to a new location. What location the gold is moved to is random, and is determined by probability distribution **P**(GoldLocation$_t$). This distribution is **known** to the robot.

Which statement is correct in this new situation?
Select one alternative

○ The robot can solve the problem using MDP/“Value Iteration”. ✓

○ MDP/“Value Iteration” does not work, but RL/“Q-learning” will work.

○ Neither MDP/"Value Iteration" nor RL/"Q-learning" will work in this situation because both algorithms make assumptions that are violated in this setting.

Sub-question (d):

A number of **monsters enter the maze**. They are spread around in different locations randomly, such that each location is equally likely to host a monster. The monsters stay fixed as soon as they have been positioned and will **stay there**, also from one episode to the next. However, the robot **does not know** the monsters' locations before bumping into them. If the robot enters a location containing a monster it will stop working. That means the episode is over. To let the robot know that this is a bad outcome, **the reward of meeting a monster is -1**. This number is **known**. The robot knows in advance that monsters have entered the maze, and it also knows the **probability distribution** used to determine the monsters' locations (i.e., that all cells were equally likely to host one of the monsters).

Which statement is correct in this new situation?

Select one alternative

○ The robot can solve the problem using MDP/“Value Iteration”.

○ MDP/“Value Iteration” does not work, but RL/“Q-learning” will work. ✓

○ Neither MDP/"Value Iteration" nor RL/"Q-learning" will work in this situation because both algorithms make assumptions that are violated in this setting.

Sub-question (e):

Starting from the situation described in part **(d)**, the robot is changed by getting a new motor-control-system. This leads to an erratic movement behaviour, and the transition-model (the one called *P(s'|a, s)* above) that the robot has known and used so far, is no longer correct. The robot still knows what actions to choose from when it is in a state *s*, and it knows that the effect of selecting an *a* is random, but it **no longer knows the transition distribution**.

Which statement is correct in this new situation?

Select one alternative

○ The robot can solve the problem using MDP/“Value Iteration”.

○ MDP/“Value Iteration” does not work, but RL/“Q-learning” will work. ✓

○ Neither MDP/"Value Iteration" nor RL/"Q-learning" will work in this situation because both algorithms make assumptions that are violated in this setting.

Maximum marks: 10

14 The cycle

Explain, in your own words, **the steps of the CBR cycle**. For each step you should

- **Explain briefly** what the step does.
- **Discuss briefly** how **domain knowledge** can be utilized in that step **-** or **say explicitly** that domain knowledge plays no role in that step.

This question is worth 5 points.

Fill in your answer here

Format

The 4 steps of the cycle should be known: Retrieve, reuse, revise, retain.
For each step, it is a bit up to the student to be clever about how to use knowledge. We should not award a claim that knowledge cannot be used if it can, but apart from that I think we should be accommodating. Note that the possibilities mentioned below do not give an exhaustive list:

* Retrieve: Find the case(s) in the case-base closest to current situation. It is meaningful to use domain knowledge to make the similarity metric: Things like “color is not important”, or “This only matters in context of some other variable taking a specific value”

* Reuse: Map the solution from the previous case to the target problem. One can use domain knowledge to define how this is done. Example: In some context/input situation, we adapt the solution in this way, in other situations we do it that way.

* Revision: Test the new solution (real world or simulation), and revise if needed. This is more often than not seen as an external thing, and here it is OK to say no domain knowledge used. (But if there are ideas of using it, then that is good too!)

* Retain: Store the resulting experience as a new case in the case base. There CAN be clever things going on here about how to choose if indeed to store the case (is it needed or not), etc. but also for Retain I am willing to accept answers stating that knowledge is not used as well as more creative ones using knowledge.

5 points available in total. Only naming the steps: 2 points. Naming steps and explain what they do, but nothing on domain knowledge: 3 points

Words: 0

Maximum marks: 5