QUESTION 1

Question 1 can in total give 20 points. All ten sub-questions are to be answered by either "True" or "False". No explanation is required. If you answer a sub-question correctly you get +2 points. A wrong answer gives -3 points. Not answering a sub-question results in 0 points. Nevertheless, please note that your total score from Question 1 will never be negative. A point total of zero will be carried forward if the sum of the scores from the ten sub-questions is negative.

a) Every Boolean function can be represented by a Bayesian network. **TRUE**

b) Suppose that none of the variables $X_1, X_2, \ldots, X_k$ have parents in a Bayesian network that in total consists of $n > k$ variables. Then the Bayesian net asserts that

$$\mathbf{P}(X_1, X_2, \ldots, X_k) = \mathbf{P}(X_1) * \mathbf{P}(X_2) * \ldots * \mathbf{P}(X_k).$$ **TRUE**

c) Bayesian networks are particularly well suited for detecting objects in a greyscale image if the pixels of the image are represented as an array of intensity values. **FALSE**

d) A perfectly rational agent will never loose in a game of Poker. **FALSE**

e) An agent that senses only partial information about the state of its environment can still be perfectly rational. **TRUE**

f) A system must think like a human to pass the Turing Test reliably. **FALSE**

g) The Chinese Room argument explains why the Weak AI hypothesis is not achievable. **FALSE**

h) Deep neural networks are called deep because they require deep engineering knowledge to be learned efficiently. **FALSE**

i) A key ingredient of convolutional neural networks is weight sharing. **TRUE**

j) The gradient of the loss function wrt. the network weights is not sufficient to learn the weights of a deep learning network. **FALSE**

QUESTION 2

Question 2 can give 18 points in total. Each of the three sub-questions has a maximum score of 6 points.

An agent inhibits a world with two states, named "**S**" and "**¬S**" (NOT S). At each point in time the agent does exactly one of the following two actions:

A1: This action has no effect. The agent remains in the same state as at the previous time-step.

A2: The state of the world is flipped. If the agent was in **S**, he will be in **¬S** after **A2** is performed; if he was in **¬S** he will be in **S** after **A2** is performed.

At each point in time the agent receives a reward, *R*, depending on which state the agent was in: If the agent is in state **S** the reward is 3, otherwise the reward is 2.

We want to find the optimal strategy π, that for each possible state of the world determines the rational action to perform.

a) Explain why the problem can be represented as a *Markov Decision Problem*. Make any additional assumption(s) you make explicit.

Process: 1:Observe; 2:Collect reward; 3:Select action; 4:Execute. Repeated in loop.
Assumptions: Fully observable, Markov assumption, transition process & rewards known and stationary, infinite time horizon

b) The Bellman-equations state that

$$U^*(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \cdot U^*(s')$$

Explain how this equation can help the agent determine the optimal strategy.

The U function determines the maximum expected discounted reward starting in s and behaving optimally thereafter. It is therefore a quantification of the *utility* of being there. If we have the U-function we know how to proceed: At any point in time take the action that maximizes utility (MEU principle). U-func is unknown, but must satisfy the Bellman equation (good if explained why). Hence, using Bellman we find U, therefore optimal strategy.

c) Calculate the U-values for the first three iterations of the value iteration scheme. After each iteration, you should also produce the associated strategy.
Set γ = 0.5, and start from initial U-values of zero.

Start with U0(S) = 0, U0(NOT S) = 0
U1(S) = 3 + 0.5 * max(0,0) = 3
U1(NOT S) = 2 + 0.5 * max(0,0) = 2
Strategy: If in S we choose A1. If in NOT S we choose A2.

U2(S) = 3 + 0.5 * max(2,3) = 4.5
U2(NOT S) = 2 + 0.5 * max(2,3) = 3.5
Strategy: If in S we choose A1. If in NOT S we choose A2.

U3(S) = 3 + 0.5 * max(4.5,3.5) = 5.25
U3(NOT S) = 2 + 0.5 * max(4.5,3.5) = 4.25
Strategy: If in S we choose A1. If in NOT S we choose A2.

The convergence point is U*(S)=6, U*(NOT S)=5. The strategy remains unchanged.

QUESTION 3

Question 3 can give 22 points in total. The scoring rule for each sub-question is given below.

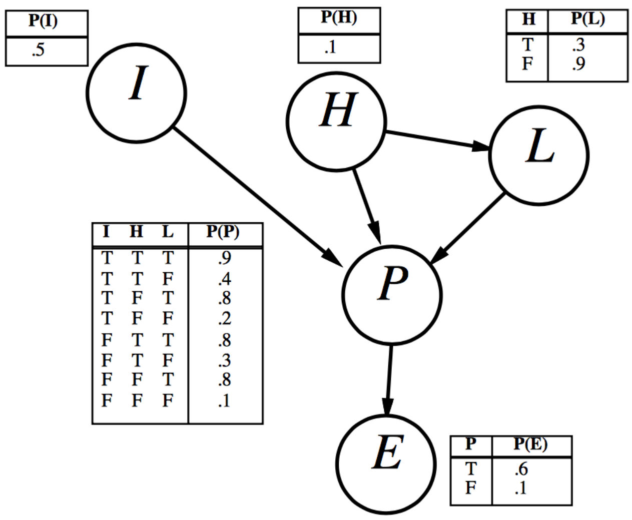

Figure 1: A simple Bayesian network with Boolean variables

In Question 3 we consider the simple Bayesian network model shown in Figure 1. The model describes the process of getting elected as the American president. All the variables are binary, and have the following meaning:

- *I:* Intelligent
- *H:* Honest
- *P:* Popular
- *L:* LotsOfCampaignFunds
- *E:* Elected

For each node we have given the associated conditional distribution. Note that only the required numbers are given, with the syntactic definition that, for instance, "P(H)" is a shorthand for P(H=True). With this number being given as 0.1, we know that (since all variables are binary) P(H=False) = 1 – P(H=True) = 0.9.

a) Considering only the *structure* of the model (and not the allocated conditional probability tables), which – if any – of the following statements are True:
i) **P**(I, L) = **P**(I) **P**(L) TRUE
ii) **P**(E|P, L) = **P**(E|P, L, H) TRUE
iii) **P**(P|I, H) ≠ **P**(P|I, H, L) TRUE
This sub-question is worth 3 points.

b) Calculate P(*I=True*, *H=True*, *L=False*, *P=True*, *E=False*).
This sub-question is worth 3 points.
P(*I=True*, *H=True*, *L=False*, *P=True*, *E=False*) =
P(I=True)*P(H=True)*P(L=False|H=True)*P(P=True|I=True,H=True,L=False)*P(E=False|P=True) =
0.5*.1*0.7*0.4*0.4 = 0.0056

c) Calculate the probability that someone is intelligent given that they are honest, have few campaign funds, and are elected.
This sub-question is worth 4 points.

P(I=True |H=True, L=False, E=True) =
P(I=True, H=True, L=False, E=True)/ P(H=True, L=False, E=True)
P(I=True, H=True, L=False, E=True) =
P(I=True, H=True, L=False, P=True, E=True) +
P(I=True, H=True, L=False, P=False, E=True)
= .5*.1*.7*.4*.6 + .5*.1*.7*.6*.1 = 0.0105
P(I=False, H=True, L=False, E=True) =
P(I=False, H=True, L=False, P=True, E=True) +
P(I=False, H=True, L=False, P=False, E=True)
= .5*.1*.7*.3*.6+.5*.1*.7*.7*.1=0.00875
P(I=True |H=True, L=False, E=True) =
P(I=True, H=True, L=False, E=True) / (P(I=True, H=True, L=False, E=True) +
P(I=False, H=True, L=False, E=True) =
0.0105 / (0.0105 + 0.00875) = 0.545

d) Suppose that we want to add the variable *R=RigsElection* to the network. Draw the new network and provide new or modified conditional probability tables as needed.
This sub-question is worth 6 points.
Different type of answers can be given here. One MUST have R the parent of E (influences of you are elected) and child of H. Can (but not a must) the child also of L and P. For the CPTs, any quantification for R is OK as long as the effect is "in the right direction" – e.g., higher prob of rigging for dishonest than honest). CPT for E should/could be as without R (that is, as in the original assignment) for R=False. Higher prob for being elected if R=True.

e) True or False: If there are two candidates in the race, then making two copies of the network will correctly represent the joint distribution over the two sets of variables.

This sub-question is to be answered by "True" or "False" only. A correct answer gives 2 points, a wrong answer results in -3 points. No answer gives zero points.

False. Only one guy is elected, and two copies do not capture this.

f) Suppose that you are considering making a large *Donation* to help elect your candidate. Draw the influence diagram/decision network (only the structure, conditional probability tables not required) obtained by adding the appropriate node(s) to Figure 1.
This sub-question is worth 4 points.
D would be the parent of L. Furthermore, we would have a value-node as the child of E. A link from D to the utility-node is required to get full score.

QUESTION 4:

Question 4 can give 20 points in total, with max-score for each sub-question given below.

a) Consider the four different training sets given in Figure 2 below.

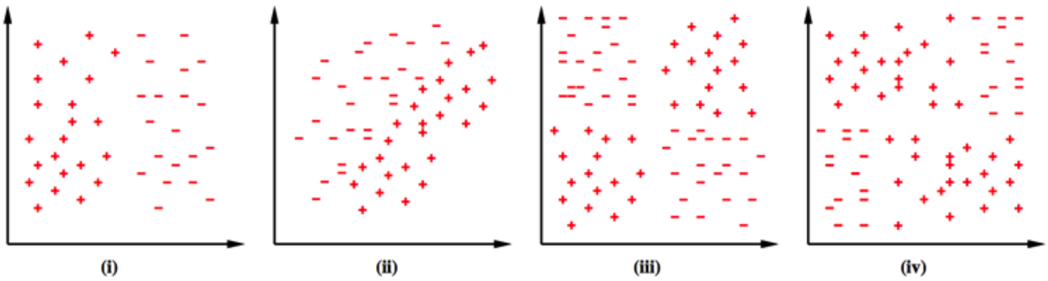

Figure 2 Four different training sets with two numerical input-attributes

Each training set has 2 continuous input attributes, one measured along the *x*-axis, the other along the *y*-axis. Each training example is either a member of the *positive class* (shown by a "+") or the *negative class* (shown by a "-").

For each of the four training sets, you should choose one machine learning method that can classify that training set correctly. Choose among the following:
- Decision trees
- Perceptron
- Case-based reasoning with Eucleadian distance measure
- Deep learning with convolutional filters

If more than one approach will be successful, you chose the simplest one. You do not have to select the same classifier for all four data sets.
This sub-question is worth 8 points: 2 points per dataset in Figure 2.
For full score the answers should be:
i) DT or Perceptron
ii) Perceptron – and none of the others
iii) DT – and none of the others
iv) DT – and none of the others
Answering CBR on either of the above datasets it will give 1 p.

b) ***Briefly*** explain the four steps of the CBR cycle.

<u>This sub-question is worth 6 points.</u>

Very short solution as follows:

Retrieve: Find closest case, Reuse: Use solution of closest case in the setting of the query, Revise: Check if it worked, Retain: Store solution in case-base.

More detail would be expected, e.g., something about the CBR cycle to put the explanation into context.

c) You wish to compare your new machine learning algorithm to decision trees. Your algorithm, called YOU, has an important parameter that users can set. Call this parameter θ and assume it only has three possible settings (1, 2, and 3).You have three standard datasets used in machine-learning research, call then A, B, and C. You split each data-set into a training set and a test set, train using the training set, and calculate the following test set accuracies:

Dataset	Decision Tree	YOU with θ=1	YOU with θ=2	YOU with θ=3
A	79%	75%	81%	91%
B	62%	67%	56%	61%
C	89%	84%	93%	87%

Which interpretation of these results is best?

1. YOU is better because for EACH dataset SOME setting of θ produces results better than the Decision Tree.
2. The Decision Tree is better because for SOME setting of θ it beats YOU on each dataset.
3. An additional dataset (a *validation set*) should have been used to independently choose a good setting for θ for each Dataset A, B and C. Then YOU should report only the score with that value of θ.
4. Θ should *not* be a settable parameter in YOU and instead should be always set to 3 because on average it produces the best accuracy. Therefore YOU is better than Decision Trees for dataset A, but poorer for datasets B and C.

State you selection (1, 2, 3 or 4) and ***briefly*** explain your answer.

<u>This sub-question is worth 6 points.</u>

The key point here is that we must consider how we use the data, and what info that should be available to us when YOU is specified. For 1 to work we need the accuracy on future test-sets to choose the best θ, and this is not viable. Same type of argument invalidates 2. 3 is a good strategy. We incorporate the choosing of θ as a part of the learning – we use the data to fix it. 4 is not good. Again we use the results of test-data to select θ, and we redice the quality of YOU by assuming it must be fixed. → 3 is the choice!