

**i Exam: DAT440 / DIT471 Advanced topics in machine learning****Examiner and teacher:** Morteza Haghir Chehreghani, [morteza.chehreghani@chalmers.se](mailto:morteza.chehreghani@chalmers.se)

- The final exam is in the form of a digital exam (to be done on Inspera).
- The exam will take four hours.
- There are in total 34 questions: Most of the questions have one point/mark, and there are few questions with two or three points/marks. The mark of each question is specified. The total score is 40.
- The exam must be done individually, and you cannot use cheat-sheet or any other resources.
- You can have an ordinary calculator for the exam.
- The exam consists of only multiple-choice questions, where for each question, at least one option (and possibly more than one) can be correct. The correct number of options is specified for each question.
- There will be no negative score for wrong answers, but you need to choose all (and only) the correct answers to get the mark/score of the question.
- The examiner will visit the exam premises on two occasions to answer the clarification questions: i) one hour after the start of the exam, and ii) when an hour of the exam remains.
- You do not need to show your calculations for the questions.

1 For regret analysis of a bandit algorithm, in what settings is the bad event often negligible?

[Number of correct options: 2]

The probability of the clean event is significantly larger.



The total number of rounds is large.





The upper confidence interval is small.

The total number of arms is large.

Rätt. 1 av 1 poäng.

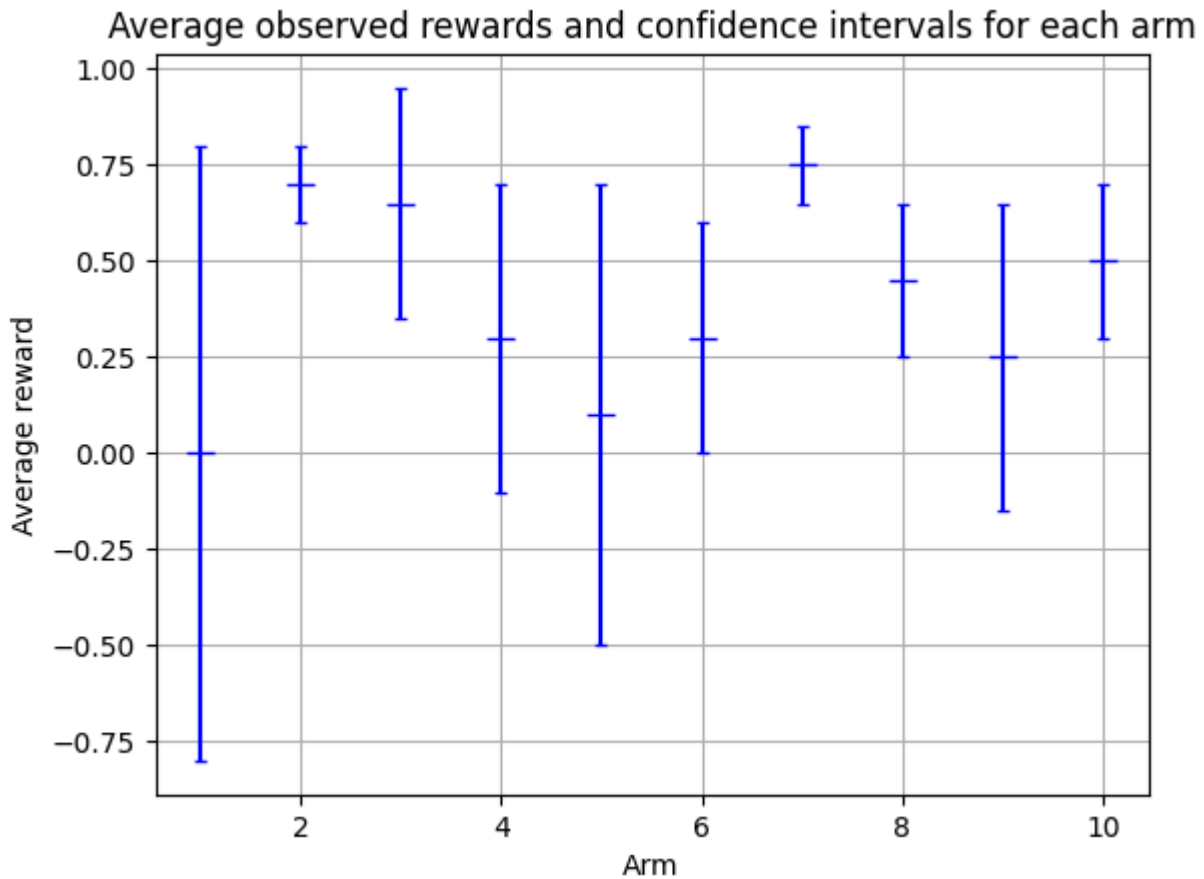
**2** Choose the correct statements about Epsilon-Greedy ( $\epsilon$ -greedy) in bandit problems.

[Number of correct options: 2]

- The exploration does not depend on the history of observed rewards. 
- The exploration is uniform over arms. 
- Epsilon-Greedy performs better than Greedy when the rewards are deterministic (fixed) but unknown.
- Compared to Greedy, Epsilon-Greedy becomes less effective when the rewards are noisy.

Rätt. 1 av 1 poäng.

3



Consider a stochastic bandit problem with a set of arms  $\mathcal{A}$ , and  $|\mathcal{A}| = 10$ . For an arbitrary round, the figure displays the average reward  $\bar{\mu}(a)$  up to this round for every arm  $a \in \mathcal{A}$  and corresponding confidence intervals. Which arm(s) will algorithm UCB1 and Greedy choose in this round?

Algorithm UCB1 will choose   (Arm 2, Arm 10, Arm 8, **Arm 3**, Arm 7, Arm 6, Arm 4, Arm 5, Arm 1, Arm 9).



Greedy will choose   (Arm 6, Arm 1, **Arm 7**, Arm 3, Arm 4, Arm 2, Arm 5, Arm 10, Arm 8, Arm 9).

Rätt. 1 av 1 poäng.

4

Choose the correct statement about Hoeffding's inequality.

[Number of correct options: 2]



- The probability of the (high-probability) event decreases as the time horizon  $T$  increases.
- It is only applicable when the time horizon  $T$  goes to infinity, which implies the average reward converges to the true mean as  $T$  goes to infinity.
- Can define the confidence radius and confidence interval used in many bandit algorithms. 
- With high probability, the largest difference between the average of all observed rewards and corresponding expected value is inversely proportional to the square root of the number of observations. 

Rätt. 1 av 1 poäng.

5 Which problems can be modelled as a multi-armed bandit problem?

Hint: Note that some of the problems can be modelled as a reinforcement learning problem, not a multi-armed bandit problem.

[Number of correct options: 2]

- Learn to win a game of chess against a professional player.
- Find optimal path from start to end in a maze without any prior knowledge about possible paths (such that the entire path cannot be planned in advance).
- Recommend songs to minimize the number of songs that you skip. 
- Compare the treatment effectiveness of different medications. 

Rätt. 1 av 1 poäng.

- 6 Consider the following game that proceeds over  $n$  rounds: In each round  $t \in \{1, \dots, n\}$ , you choose either to play or do nothing. If you do nothing, then your reward is  $X_t = 0$ . If you play, then your reward is  $X_t = 1$  with probability  $p$  and  $X_t = -1$  otherwise.

In terms of regret, what is the optimal way of choosing actions, i.e., the best algorithm, when  $p$  is known?

[Number of correct options: 1]

- Always choose to do nothing.
- If  $p > 0.5$ , choose to do nothing. Otherwise, choose to play.
- Always choose to play.
- If  $p > 0.5$ , choose to play. Otherwise, choose to do nothing.



Rätt. 1 av 1 poäng.

7

Choose the correct statements about the reward for a chosen arm in stochastic multi armed bandit problems.

[Number of correct options: 2]

- Depends on the state of the environment
- Depends on previously chosen arms
- Depends on the chosen arm
- Depends on an unknown distribution





Rätt. 1 av 1 poäng.

8 Table 1: Number of samples  $n_t(a)$  and total observed reward (return)  $s_t(a)$  of each  $a$  arm before round  $t$ .

$a$	1	2	3	4	5	6	7	8	9	10
$n_t(a)$	10	52	24	17	10	18	24	21	14	109
$s_t(a)$	0	36	10	5	0	6	10	8	3	92
$x_t(a)$	1	1	1	1	0	0	1	0	0	1

Consider a stochastic  $K$ -armed bandit problem with a set of arms  $\mathcal{A}$ , and  $|\mathcal{A}| = K$ . We define, at round  $t$ :  $a_t \in \mathcal{A}$  is the arm played by the agent,  $x_t(a) \in \{0, 1\}$  is the reward received by an agent if it plays arm  $a \in \mathcal{A}$ ,  $s_t(a) = \sum_{j=1}^{t-1} \mathbf{1}\{a_j = a\} x_j(a)$  is the cumulative reward of arm  $a$  before round  $t$ , and  $n_t(a) = \sum_{j=1}^{t-1} \mathbf{1}\{a_j = a\}$  is the number of rounds that arm  $a$  has been played before round  $t$ . Consider the scenario in Table 1, where an agent has played in a multi-armed bandit environment with  $K = 10$  arms up to round  $t$ . Note that the reward  $x_t(a)$  is revealed to the agent *if and only if* arm  $a \in \mathcal{A}$  is played in round  $t$ . With fixing the time horizon  $T = 1000$ , which arm will be chosen in rounds  $t$  and  $t + 1$  by the UCB1 algorithm with confidence radius  $r_t(a) = \sqrt{\frac{2 \log_e T}{n_t(a)}}$ ?

Round  $t$ :   (Arm 1, Arm 2, Arm 3, Arm 4, Arm 5, Arm 6, Arm 7, Arm 8, Arm 9, Arm 10)

Round  $t + 1$ :   (Arm 1, Arm 2, Arm 3, Arm 4, Arm 5, Arm 6, Arm 7, Arm 8, Arm 9, Arm 10)

Rätt. 2 av 2 poäng.

- 9 Consider a Bayesian  $K$ -armed bandit problem with a set of arms  $\mathcal{A}$ ,  $|\mathcal{A}| = K$ . Assume independent priors  $\mathbb{P}(\mu)$  given by the Beta distribution, likelihood  $\mathbb{P}(r|\mu)$  is given by the Bernoulli distribution and  $K = 4$  arms.

Assume you sample the following mean reward vectors from the posterior distribution in rounds  $t = 1, \dots, 4$ :

Round 1:  $\tilde{\mu}_1 = [0.7, 0.3, 0.6, 0.4]$  where each entry corresponds to one of the arms.

Round 2:  $\tilde{\mu}_2 = [0.5, 0.4, 0.2, 0.3]$  where each entry corresponds to one of the arms.

Round 3:  $\tilde{\mu}_3 = [0.4, 0.8, 0.6, 0.1]$  where each entry corresponds to one of the arms.

Round 4:  $\tilde{\mu}_4 = [0.3, 0.4, 0.3, 0.5]$  where each entry corresponds to one of the arms.

You use Thompson sampling to choose one arm in each round and observe the following sequence of rewards  $\{0, 1, 1, 0\}$  at different rounds (i.e., 0 in the first round, 1 in the second round, 1 in the third round, and finally, 0 in the fourth round).

Given that the prior is uniform at the first round, what are the values of posterior parameters in the next round  $t = 5$ ?

Hint 1: For observations (of random variables)  $x_1, \dots, x_t$ , the parameters of a Beta posterior distribution are given by  $(\alpha, \beta) = (\alpha_0 + \sum_{i=1}^t x_i, \beta_0 + t - \sum_{i=1}^t x_i)$ , where  $\alpha_0$  and  $\beta_0$  are the prior parameters.

Hint 2: A Beta distribution with parameters  $\alpha = \beta = 1$  yields a uniform distribution.

Note: Options are given in the following format:  $\{(\alpha^1, \beta^1), (\alpha^2, \beta^2), (\alpha^3, \beta^3), (\alpha^4, \beta^4)\}$ , where  $\alpha^i$  and  $\beta^i$  are the posterior parameters of arm  $i$ .

[Number of correct options: 1]

$\{(2,4), (2,2), (2,1), (1,2)\}$

$\{(2,2), (2,1), (1,1), (1,2)\}$

$\{(2,2), (3,2), (2,2), (2,3)\}$

$\{(1,1), (1,0), (0,0), (0,1)\}$



Rätt. 2 av 2 poäng.

- 10 Consider a stochastic bandit problem with  $K = 2$  arms with Gaussian rewards with means  $\mu_1$  and  $\mu_2$ , respectively. Assume that the first arm is the optimal arm, i.e.,  $\mu^* = \mu_1$ , and  $\mu_1 = \mu_2 + \Delta$  with  $\Delta > 0$ . It can be shown that the regret of the Explore-First algorithm will be upper bounded by

$$R(T) \leq \Delta \left( N + T \Phi \left( -\Delta \sqrt{\frac{N}{2}} \right) \right),$$

where  $\Phi$  denotes the cumulative distribution function (cdf) of the standard Gaussian distribution.

Provide a value  $N^*$  of  $N$  that minimizes the above regret upper bound.

Hint 1:  $\frac{\partial}{\partial N} \Phi(a(N)) = \phi(a(N)) \frac{\partial a(N)}{\partial N}$ , where  $\phi(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}$  denotes the probability density function (pdf) of the standard Gaussian distribution and  $a(N)$  is an arbitrary function of  $N$ .

Hint 2: Use the Lambert function  $W$ , defined as for  $y > 0$ ,  $W(y) \exp(W(y)) = y$ .

Note:  $\lceil x \rceil$  is the **ceiling function** which maps  $x$  to the least integer greater than or equal to  $x$ .

[Number of correct options: 1]

$N^* = \left\lceil W \left( \frac{T^2 \Delta^4}{32\pi} \right) \right\rceil$

$N^* = \left\lceil \frac{2}{\Delta^2} \right\rceil$

$N^* = \left\lceil \frac{T^2 \Delta^4}{32\pi} \right\rceil$

$N^* = \left\lceil \frac{1}{\Delta} W \left( \frac{T^2 \Delta^2}{16\pi} \right) \right\rceil$

$N^* = \left\lceil \frac{T^2 \Delta^2}{16\pi} \right\rceil$

$N^* = \left\lceil \frac{2}{\Delta^2} W \left( \frac{T^2 \Delta^4}{32\pi} \right) \right\rceil$





Rätt. 3 av 3 poäng.



- 11 Consider a Markov decision process for reinforcement learning with optimal policy  $\pi^*$ . Choose the correct statements about the Bellman optimality equation for the optimal action-value function  $q_*$ .

In these equations,  $a$  refers to an action,  $s$  refers to a state,  $r$  is a reward value, and  $\gamma$  is the discount factor.





[Number of correct options: 2]

- $q_*(s, a) = \sum_{s', r} p(s', r | s, a) [r + \gamma \mathbb{E}_{a' \sim \pi^*} [q_*(S', A') | S' = s']]$
- $q_*(s, a) = \sum_{s', r} p(s', r | s, a) [r + \gamma \max_{a'} q_*(s', a')]$  
- $q_*(s, a) = \max_a \sum_{s', r} p(s', r | s, a) [r + \gamma q_*(s', a)]$
- $q_*(s, a) = \mathbb{E}[R_{t+1} + \gamma \max_{a'} q_*(S_{t+1}, a') | S_t = s, A_t = a]$  

Rätt. 1 av 1 poäng.

- 12 Choose the correct statement(s) about the optimal policy in reinforcement learning.

[Number of correct options: 3]




- Optimal policies do not necessarily share the same state-value function.
- Optimal policies do not always share the same optimal action-value function. 
- Policy  $\pi$  is better than or equal to policy  $\pi'$  if and only if  $v_\pi(s) \geq v_{\pi'}(s)$  for all state  $s \in \mathcal{S}$ . 
- Optimal policies share the same optimal action-value function. 
- Optimal policies share the same state-value function. 
- Policy  $\pi$  is better than or equal to policy  $\pi'$  if and only if  $v_\pi(s) \geq v_{\pi'}(s)$  for at least one of the states  $s \in \mathcal{S}$ .

Delvis rätt. 0 av 1 poäng.

13 Choose the correct statements about state values  $v_\pi(\mathbf{s})$  for state  $\mathbf{s}$  under policy  $\pi$  in reinforcement learning.

In these equations,  $\mathbf{a}$  refers to an action,  $\mathbf{s}$  refers to a state,  $r$  is a reward value,  $\gamma$  is the discount factor, and  $G_t$  is the return.



[Number of correct options: 3]

- $v_\pi(\mathbf{s}) = \sum_a \pi(\mathbf{a}|\mathbf{s}) \sum_{s'} \sum_r p(s', r|\mathbf{s}, \mathbf{a}) [r + \gamma v_\pi(s')]$  
- $v_\pi(\mathbf{s}) = \mathbb{E}_\pi[G_t | S_t = \mathbf{s}]$  
- $v_\pi(\mathbf{s}) = \sum_a \pi(\mathbf{a}|\mathbf{s}) \sum_{s'} \sum_r p(s', r|\mathbf{s}, \mathbf{a}) [r + \gamma G_t]$
- $v_\pi(\mathbf{s}) = \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} | S_t = \mathbf{s}]$  
- $v_\pi(\mathbf{s}) = \mathbb{E}_\pi[R_{t+1} + \gamma G_t | S_t = \mathbf{s}]$

Rätt. 1 av 1 poäng.

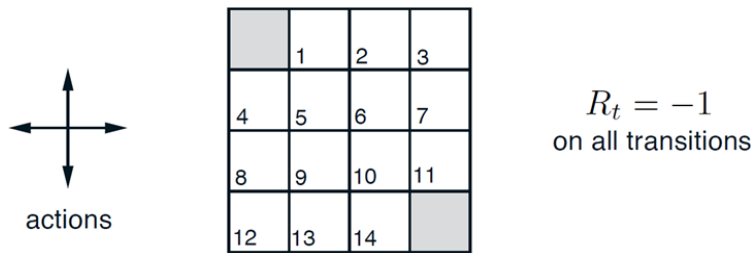
14 What is the concept of "reward hypothesis" in reinforcement learning?

[Number of correct options: 2]

- It determines what should be achieved in a reinforcement learning task. 
- It specifies how to reach the goals/purposes.
- It is equivalent to choosing actions with maximal expected/estimated reward at each time step.
- It indicates the goals and purposes can be modeled by the maximization of the expected value of the cumulative sum of rewards. 

Rätt. 1 av 1 poäng.

- 15 Consider the following reinforcement learning problem, where the agent from each of the numbered 14 cells aims to reach one of the goal cells specified by gray. Assume we want to solve the Policy Evaluation (Prediction) task using the Iterative Policy Evaluation method, where the actions are random in one of four possible directions. The immediate reward for each action (movement) is -1. You can assume an MDP for this problem with discount factor  $\gamma = 1$ .



After the second step of Iterative Policy Evaluation (i.e., for  $k=2$ ) the estimated state values are as following.

0.0	-1.7	-2.0	-2.0
-1.7	-2.0	-2.0	-2.0
-2.0	-2.0	-2.0	-1.7
-2.0	-2.0	-1.7	0.0

What is the state value of the cell highlighted by red in the next iteration (i.e., for  $k=3$ )?  
[Number of correct options: 1]

- 2.85
- 2.85
- 1.85
- 3
- 3
- 1.85





Rätt. 1 av 1 poäng.

**16** Choose the correct statements about Policy Improvement Theorem.

[Number of correct options: 2]

Let  $\pi$  and  $\pi'$  be any pair of deterministic policies such that for all  $s \in \mathcal{S}$  we have:  $q_\pi(s, \pi'(s)) \geq v_\pi(s)$ , where  $q_\pi$  and  $v_\pi$  respectively correspond to state value and action value functions. Then, the policy  $\pi$  cannot be worse than  $\pi'$ .

There is always at least one optimal policy. 

Let  $\pi$  and  $\pi'$  be any pair of deterministic policies such that for all  $s \in \mathcal{S}$  we have:  $q_\pi(s, \pi'(s)) \geq v_\pi(s)$ , where  $q_\pi$  and  $v_\pi$  respectively correspond to state value and action value functions. Then, the policy  $\pi'$  is as good as or better than  $\pi$ . 


It implies there is always exactly one optimal policy.


Rätt. 1 av 1 poäng.


**17** Choose the correct statements about Monte Carlo (MC) methods in reinforcement learning.

[Number of correct options: 2]

MC methods need some prior knowledge of the actual environment's dynamics.

Unlike Dynamic Programming, MC methods need an MDP. 

MC methods learn from experience. 



MC methods work based on averaging sample returns. 

MC methods perform the policy evaluation similar to Dynamic Programming methods.

Delvis rätt. 0 av 1 poäng.

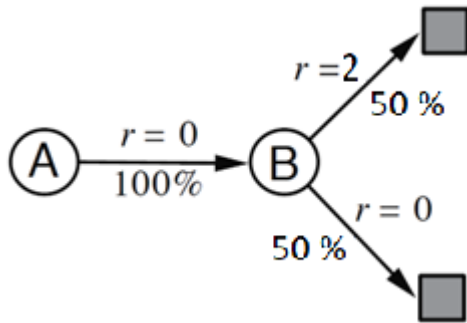
**18** Consider Monte Carlo ES (Exploring Starts) in reinforcement learning. Choose the correct statements about this method.

[Number of correct options: 2]

- It can be employed with a greedy policy. 
- It assumes a nonzero probability of being selected as the start for each state–action pair. 
- The exploration parameter  $\epsilon$  in  $\epsilon$ -greedy must be set very high (i.e., close to 1) with this method.
- It does not perform any policy improvement.
- It is required to follow an  $\epsilon$ -greedy policy with this method.

Rätt. 1 av 1 poäng.

- 19 Consider the following Markov reward process, with non-terminal states A and B, and the terminal states shown by gray squares. The parameter  $r$  specifies the reward for the respective transition.



Assume the following batch of episodes from the above Markov reward process.

- 1) A, 0, B, 2
- 2) B, 2
- 3) B, 0
- 4) B, 2
- 5) B, 0
- 6) B, 0

Choose the correct statements about state values computed in batch setting using MC (Monte Carlo) or TD (Temporal Difference) methods (assume the discount factor is 1).

[Number of correct options: 2]




- With MC we have:  $V(A) = 0$  and  $V(B) = 2$
- With MC we have:  $V(A) = 1$  and  $V(B) = 1$
- With TD we have:  $V(A) = 1$  and  $V(B) = 1$
- With MC we have:  $V(A) = 2$  and  $V(B) = 1$
- With MC we have:  $V(A) = 0$  and  $V(B) = 1$
- With TD we have:  $V(A) = 2$  and  $V(B) = 2$



Delvis rätt. 0 av 1 poäng.

**20** Choose the correct statements about double learning in reinforcement learning (in Temporal Difference methods).




[Number of correct options: 2]

- It does not affect the memory requirements.
- Two estimates for each state-action pair are learned, and both of them are updated on each play. 
- It separates the maximizing action from the estimate of its value. 
- It does not affect the amount of computation per step. 

Delvis rätt. 0 av 1 poäng.

**21** Choose the correct statements about  $n$ -step TD ( $n$ -step Temporal-Difference) methods.




[Number of correct options: 3]

- $n$ -step TD methods are equivalent to ordinary TD methods for  $n=0$ .
- $n$ -step TD methods enable bootstrapping to occur over multiple steps. 
- $n$ -step TD methods are equivalent to MC (Monte Carlo) methods for  $n=0$ .
- $n$ -step TD methods are less online (less immediate) compared to ordinary TD methods. 
- $n$ -step TD methods are equivalent to ordinary TD methods for  $n=1$ . 
- $n$ -step TD methods are equivalent to MC (Monte Carlo) methods for  $n=1$ .

Rätt. 1 av 1 poäng.

**22** Choose the correct statement(s) about Temporal-Difference (TD) learning in reinforcement learning.

[Number of correct options: 3]

- It assumes a model of the environment.
- It learns directly from raw experience (from interactions with the environment). 
- It assumes the state transition probabilities are given.
- It uses bootstrap, similar to Dynamic Programming. 
- It uses bootstrap, similar to Monte Carlo methods.
- It is a model-free method. 

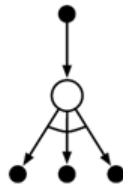
Rätt. 1 av 1 poäng.



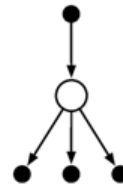
23 Consider the following three backup diagrams corresponding to Q-learning, expected sarsa, and sarsa.



(a)



(b)



(c)

Choose the correct statements.

[Number of correct options: 3]

- Backup diagram (a) is for expected sarsa.
- Backup diagram (b) is for expected sarsa.
- Backup diagram (a) is for sarsa.
- Backup diagram (c) is for Q-learning.
- Backup diagram (a) is for Q-learning.
- Backup diagram (b) is for Q-learning.
- Backup diagram (c) is for expected sarsa.
- Backup diagram (c) is for sarsa.
- Backup diagram (b) is for sarsa.



Rätt. 1 av 1 poäng.

- 24** Consider function approximation for prediction in reinforcement learning. In particular, we apply it for state value estimation of state  $S_t$ , i.e.,  $\hat{v}(S_t, \mathbf{w})$ , where  $\mathbf{w}$  corresponds to the parameters of the approximate function.

We use SGD (Stochastic Gradient Descent) to learn the parameters. They are updated as following, where  $\mathbf{w}_t$  corresponds to the estimate at time  $t$ .

$$\begin{aligned}\mathbf{w}_{t+1} &\doteq \mathbf{w}_t - \frac{1}{2} \alpha \nabla \left[ v_\pi(S_t) - \hat{v}(S_t, \mathbf{w}_t) \right]^2 \\ &= \mathbf{w}_t + \alpha \left[ v_\pi(S_t) - \hat{v}(S_t, \mathbf{w}_t) \right] \dots\dots\dots\end{aligned}$$

Choose the correct option to fill in the missing part.

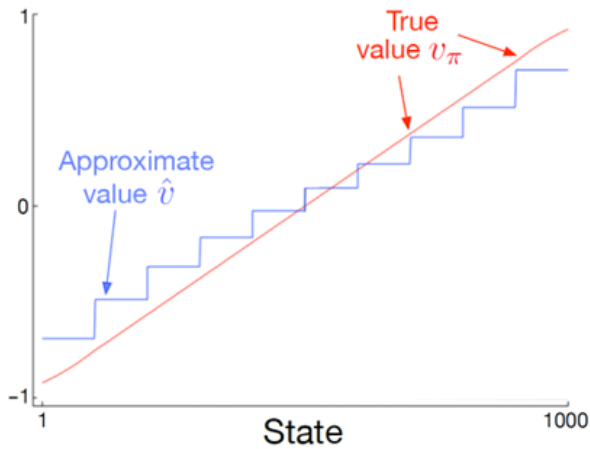
[Number of correct options: 1]

- $\mathbf{w}_t$
- $\nabla \hat{v}(S_t, \mathbf{w}_t)$  ✓
- $\nabla v_\pi(S_t)$  ✗
- $v_\pi(S_t)$
- $\nabla \mathbf{w}_t$
- $\hat{v}(S_t, \mathbf{w}_t)$

Fel. 0 av 1 poäng.

**25** Consider function approximation for prediction in reinforcement learning, applied for state value estimation, i.e.,  $\hat{v}(\mathcal{S}_t, \mathbf{w})$ , where  $\mathbf{w}$  corresponds to the parameters of the approximate function. We use the "state aggregation" method for  $\hat{v}(\mathcal{S}_t, \mathbf{w})$  along with SGD (Stochastic Gradient Descent) to learn the parameters.

We apply this method to a task with 1000 states (e.g., the 1000-state Random Walk problem). We obtain the following results about the estimated (approximate) and true state values.



Choose the correct statements about this problem.



[Number of correct options: 2]

- The update target used here is unbiased.
- The update target used here is based on Monte Carlo (MC).
- The update target used here is based on Temporal-Difference (TD). ✔
- The update target used here is biased. ✔

Rätt. 1 av 1 poäng.

**26** Choose the correct statement(s) about function approximation in reinforcement learning.



[Number of correct options: 2]

- It is useful for huge state spaces including visual images. 
- It is only applicable when the model of the environment is known.
- Compared to tabular reinforcement learning, it supports better the transfer of state values between similar states. 
- The approximate function must always be implemented using a deep neural network.

Rätt. 1 av 1 poäng.

**27** Consider the application of DQN to Atari games and choose the correct statements.

[Number of correct options: 2]

- The last layers of the neural network are fully connected layers. 
- It uses an  $\epsilon$ -greedy policy, with  $\epsilon$  increasing over time.
- The first layers of the neural network are convolutional layers, in order to capture the visual features from the game frames. 
- The last layers of the neural network are convolutional layers.
- The first layers of the neural network are fully connected layers.

Rätt. 1 av 1 poäng.

28 Consider the following DQN algorithm and choose the correct statements about that.



```

Initialize network  $\hat{q}$ 
Initialize target network  $\tilde{q}$ 
Initialize experience replay memory  $D$ 
Initialize the Agent to interact with the Environment
while not converged do
  /* Sample phase
   $\epsilon \leftarrow$  setting new epsilon with  $\epsilon$ -decay
  Choose an action  $a$  from state  $s$  using policy  $\epsilon$ -greedy( $\hat{q}$ )
  Agent takes action  $a$ , observe reward  $r$ , and next state  $s'$ 
  Store transition  $(s, a, r, s', done)$  in the experience replay memory  $D$ 

  if enough experiences in  $D$  then
    /* Learn phase
    Sample a random minibatch of  $N$  transitions from  $D$ 
    for every transition  $(s_i, a_i, r_i, s'_i, done_i)$  in minibatch do
      if  $done_i$  then
        |  $y_i = r_i$ 
      else
        |  $y_i = r_i + \gamma \max_{a' \in \mathcal{A}} \tilde{q}(s'_i, a')$ 
      end
    end
    Calculate the loss  $\mathcal{L} = 1/N \sum_{i=0}^{N-1} (\hat{q}(s_i, a_i) - y_i)^2$ 
    Update  $\hat{q}$  using the SGD algorithm by minimizing the loss  $\mathcal{L}$ 
    Every  $C$  steps, copy weights from  $\hat{q}$  to  $\tilde{q}$ 
  end
end

```




[Number of correct options: 2]

- The target network is used as the duplicate network. 
- When  $done_i$  is *true* then it does not use experience replay.
- It uses the MC (Monte Carlo) returns as update target.
- It does not use experience replay.
- It uses the TD (Temporal Difference) as update target. 

Rätt. 1 av 1 poäng.

29 Choose the correct statements about duplicate network used in DQN.

[Number of correct options: 2]

- It requires training two neural networks independently in parallel, but using the same data. 
- The use of duplicate network can avoid oscillations or divergence. 
- With this trick, the neural network used in the target (the target of Q-learning) is updated less frequently. 
- It makes the target for a Q-learning update depend on the most recent update of the neural network.



Delvis rätt. 0 av 1 poäng.

30 Consider the following Policy Gradient Theorem in policy-based reinforcement learning. Assume the policy parameters are  $\theta$  and the respective performance measure (objective) is  $J(\theta)$  which is defined as the value of the start state, i.e.,  $J(\theta) = v_{\pi_{\theta}}(s_0)$ . Moreover,  $\mu(s)$  is the state distribution, and  $q_{\pi}(s, a)$  is the action value function for pair  $(s, a)$  under policy  $\pi$ .

$$\nabla J(\theta) \propto \sum_s \mu(s) \sum_a q_{\pi}(s, a) \nabla \pi(a|s, \theta)$$

Choose the correct expression about gradient ascent update to maximize  $J(\theta)$ .

[Number of correct options: 1]

- $\theta_{t+1} = \theta_t + \alpha \sum_s \mu(s) \nabla \pi(a|s, \theta)$
- $\theta_{t+1} = \theta_t + \alpha \hat{q}(S_t, A_t, \mathbf{w}) \nabla \pi(A_t|S_t, \theta)$  
- $\theta_{t+1} = \theta_t + \alpha \mu(S_t) \nabla \pi(A_t|S_t, \theta)$
- $\theta_{t+1} = \theta_t + \alpha \sum_a \hat{q}(S_t, a, \mathbf{w}) \nabla \pi(a|S_t, \theta)$  

Fel. 0 av 1 poäng.

31 Consider REINFORCE, a policy gradient algorithm in reinforcement learning described below.

Choose the correct statement about this algorithm.

[Number of correct options: 1]

- It is based on MC (Monte Carlo).
- It is a first-visit method.
- It evaluates a given policy without modifying (improving) it.
- Policy improvement is based on  $\epsilon$ -greedy.



Rätt. 1 av 1 poäng.

32 Consider the following Policy Gradient Theorem in policy-based reinforcement learning. Assume the policy parameters are  $\theta$  and the respective performance measure (objective) is  $J(\theta)$  which is defined as the value of the start state, i.e.,  $J(\theta) = v_{\pi_\theta}(s_0)$ . Moreover,  $\mu(s)$  is the state distribution, and  $q_\pi(s, a)$  is the action value function for pair  $(s, a)$  under policy  $\pi$ .

$$\begin{aligned} \nabla J(\theta) &\propto \sum_s \mu(s) \sum_a q_\pi(s, a) \nabla \pi(a|s, \theta) \\ &= \mathbb{E}_\pi \left[ \sum_a \text{-----} \nabla \pi(a|S_t, \theta) \right] \end{aligned}$$

Choose the correct statement about the missing part.

[Number of correct options: 1]

- $q_\pi(S_t, a)$
- $\mu(S_t)$
- $q_\pi(s_0, a)$
- $\mu(s_0)$



Fel. 0 av 1 poäng.

33 Consider the following n-step Sarsa for estimating action values (control).

```

Initialize  $Q(s, a)$  arbitrarily, for all  $s \in \mathcal{S}, a \in \mathcal{A}$ 
Initialize  $\pi$  to be  $\varepsilon$ -greedy with respect to  $Q$ , or to a fixed given policy
Algorithm parameters: step size  $\alpha \in (0, 1]$ , small  $\varepsilon > 0$ , a positive integer  $n$ 
All store and access operations (for  $S_t, A_t$ , and  $R_t$ ) can take their index mod  $n + 1$ 

Loop for each episode:
  Initialize and store  $S_0 \neq$  terminal
  Select and store an action  $A_0 \sim \pi(\cdot|S_0)$ 
   $T \leftarrow \infty$ 
  Loop for  $t = 0, 1, 2, \dots$  :
    | If  $t < T$ , then:
    |   Take action  $A_t$ 
    |   Observe and store the next reward as  $R_{t+1}$  and the next state as  $S_{t+1}$ 
    |   If  $S_{t+1}$  is terminal, then:
    |      $T \leftarrow t + 1$ 
    |   else:
    |     Select and store an action  $A_{t+1} \sim \pi(\cdot|S_{t+1})$ 
    |    $\tau \leftarrow t - n + 1$  ( $\tau$  is the time whose estimate is being updated)
    |   If  $\tau \geq 0$ :
    |     X
    |     If  $\tau + n < T$ , then  $G \leftarrow G + \gamma^n Q(S_{\tau+n}, A_{\tau+n})$  ( $G_{\tau:\tau+n}$ )
    |     Y
    |     If  $\pi$  is being learned, then ensure that  $\pi(\cdot|S_\tau)$  is  $\varepsilon$ -greedy wrt  $Q$ 
  Until  $\tau = T - 1$ 

```

Choose the correct statements about the missing parts X and Y.

[Number of correct options: 2]

- $Y : Q(S_\tau, A_\tau) \leftarrow Q(S_\tau, A_\tau) + \alpha[G - Q(S_\tau, A_\tau)]$  ✓
- $Y : Q(S_\tau, A_\tau) \leftarrow Q(S_\tau, A_\tau) + \alpha[G - \max_a Q(S_\tau, a)]$  ✗
- $X : G \leftarrow \sum_{i=\tau+1}^{\min(\tau+n, T)} \gamma^{i-\tau-1} R_{i-\tau-1}$
- $Y : Q(S_\tau, A_\tau) \leftarrow Q(S_\tau, A_\tau) + \alpha G$
- $X : G \leftarrow \sum_{i=\tau+1}^{\min(\tau+n, T)} \gamma^{i-\tau-1} R_i$  ✓
- $X : G \leftarrow \sum_{i=\tau+1}^{\min(\tau+n, T)} \gamma^i R_i$

Delvis rätt. 0 av 2 poäng.



- 34 Consider the REINFORCE method with baseline, where the policy gradient theorem is generalized to include the baseline  $b(s)$ .

$$\nabla J(\theta) \propto \sum_s \mu(s) \sum_a \left( q_\pi(s, a) - b(s) \right) \nabla \pi(a|s, \theta)$$

$J(\theta)$  is the objective (performance measure),  $\mu(s)$  is the distribution of states,  $q_\pi$  is the action value function, and  $\pi$  is the policy function.

Consider the following expression and choose the correct statements.

$$\sum_a b(s) \nabla \pi(a|s, \theta) = X \quad \nabla \sum_a Y = Z$$

[Number of correct options: 3]

- $Y = b(s)$
- $Y = \pi(a|s, \theta)$  
- $Z = 1$  
- $X = \pi(a|s, \theta)$
- $Z = 0$  
- $X = b(s)$  

Delvis rätt. 0 av 2 poäng.