Sergei Popkov, UEF Summer School 2019 (“DRL for CG”), learning diary questions.

1. Answer the questions on the page 40 of “Introduction to machine learning” slides. (Regarding to the confusion matrix)

How many Estonian test samples we have ? **186**
What is the error rate for Russian dialect? Total = 590, Error count = 590 – 379 = 211, Error rate = 211/590 = 0.357627118644068 ~= **0.36**
How many Turkish samples are mistakenly classified as Arabic ? **10**
What are the weighted and unweighted error rates ? **The metric of error rate estimation, where the corresponding evaluation is biased due to disproportionate classes, are called weighted error rate (accuracy). The alternative approach provides more balanced solution estimating the error rates specified independently for each given class.**

2. Neural networks (especially deep networks) can overfit very easily. What happens in this phenomenon and what you can do to avoid it?

Overfitting is a machine learning phenomena that occurs when the model is fitting the training set perfectly, but unable to adequately fit to the newly presented data, therefore failing to serve the purpose the model was made for in the first place. There’s a lot of different techniques to avoid overfitting (creating cross-validation set, applying dropout to the neural network, adding more data, removing reduntant variables (features), applying regularization and early stopping, i.e. detecting the moment of overfitting (using the difference between errors on training and validation set to abort the fitting process, to name a few).

It is always important to think about the problem at hand carefully, since there are different ways to model to data to and get a solution. What is the difference between generative and discriminative models? And when (and why) you would select to use either of them? Excellent and advanced look on the state-of-the-art neural network (and at the same time machine learning in general) results are available in this online book (published 2016): http://www.deeplearningbook.org

Generative models are using joint distribution, find the closest match of the sample to determine its class (like clustering, in general) whereas discriminative models are usually applying conditional distribution, describing decision boundary for each class.

3. What does Q-function take as inputs? What does it return?

Q-function is a value function, taking the state and the action as inputs. It returns the accumulative reward expected to be received by the agent while performing given actions from the state provided as the function input. This function helps to find the best action available to maximize the reward.

4. What are the main steps of Q-learning process? Describe each step shortly.

1) Q-table initialization – prepare the model for the training process (for example, set all table values to zero)
2) Action choosing – choose action while in the given state using policy derived from Q-function
3) Action performing – take the chosen action, observe new state.
4) Reward computation – estimate the received reward to update the Q-table.
5) Q-table update – update the Q-values for each state and action according to the taken action, as well as the new observed state and corresponding reward.
Steps 2-5 are repeated until the timeout or terminal (final) state is reached.

5. What is the main difference between policy gradient and state-action based approaches (e.g. Q-learning)? Which one is better and why?

The state-action approaches are aimed to optimize the value function. The policy gradient approaches are trying to perform the policy search to learn it directly. Policy gradient may converge to the local optimum and is learning much slower than the state-action based approach, albeit Q-values may take much more resources to compute them. None of them is better than another, because the effectiveness of the certain approach depends on the particular situation, problem at hand and dataset.

6. What are the major differences between stochastic and deterministic policies?

The deterministic policies map the states to the actions (useful for deterministic environment with clear certain pattern between the chosen action and corresponding state), whereas the stochastic policies output a probability distribution over actions (useful for stochastic environment without certain conditions for the actions).

7. What are the roles of the actor and critic in the Actor-Critic algorithm?

The actor learns the best policy, the critic evaluates the value function. The actor tries to solve the problem, the critic provides feedback to update the policy and improve the overall quality of the model.

8. What is the main advantage of actor-critic over policy gradient?

Basically, the Actor-Critic algorithm can be represented as the mix of the policy gradient and the state-action based approaches. Actor and Critic, as independent parts (neural networks) of the unified system, correct their respective weights based on the actions of each other. That way, the AC model can overcome the primary shortage of the policy gradient approach, which is inability to distinguish the bad actions from the overall good (reward-wise) episodes (without proper, relatively large, amount of samples).

9. Why do we decrease the epsilon over time in the exploration-exploitation dilemma? What happens if we set the epsilon to 1 or 0 all the time?

When we keep exploring all the time (epsilon=1), we basically not acting much better than random untrainable agent; meanwhile, excessive exploitation mode (epsilon=0) can lead agent to stuck in the local optima, unable to see “the bigger picture” and try something new (which may work way better than “well-established approaches” and behaviour patterns discovered before).