

Interview Problems for Data Science Postdoc position

Malachi Schram, Diana McSpadden, Kishansingh Rajput

Thomas Jefferson National Accelerator Facility

Oct 11, 2021

1 Overview, tools, and materials.

For the problems assigned below, we expect the applicant to use Python 3.X and [Tensorflow/Keras](#) to develop the machine learning code (RL policy agent, GAN generator and discriminator, and the driver scripts). The code needs to be made available the day prior to the panel interview on [Github](#). The applicant can use their *personal computer* and/or any *free and publicly* available computing resources. Please prepare to go over your code and summarize our results.

2 Problem Definition #1 (reinforcement learning)

The goal of this exercise is to solve the inverted pendulum problem using reinforcement learning. The inverted pendulum swing-up problem is a classic problem in the control literature. In this version of the problem, the pendulum starts in a random position, and the goal is to swing it up so it stays upright.

To solve this problem, implement the Deep Deterministic Policy Gradient (DDPG) algorithm in a python class and provide a script that coordinates the interaction between the agent and the environment. Please provides figures that capture the learning behavior of the agent. The applicant is expected to use the `Pendulum-v0` environment provided by [OpenAI gym](#).

3 Problem Definition #2 (generative models)

The goal of this exercise is to create a generative adversarial network to reproduce the tabular data found https://github.com/JeffersonLab/jlab_datascience_data/blob/main/eICU_age.npy. The dataset includes the ages of critical care patients seen across intensive care units in the United States. The original data is from the eICU Collaborative Research Database <https://eicu-crd.mit.edu/about/eicu/>.

To solve this problem implement a generative adversarial network (GAN) in python to produce the distribution of age for these patients. Your script must coordinate both the generator and discriminator models. Please provide figures to illustrate this coordination and compare the distribution from the generator model output with that of the true distribution.