My solution

1. Method

Supervised learning

2. Data processing

- “achieved_goal”: duplicated, removed.
- “robot_id”: it’s unclear whether the robots for last test are among those data providers, and consider the effect of possible maintenance work, this is also removed.
- “delay”: there exist unreasonable huge delay in real data, so data with delay higher than 0.5 are removed.

3. Network structure

- MLP with 3 hidden layers, each layer have 512 nodes.
- ReLU for each layer, except Tanh for output layer.

4. Optimization

- l1 loss
- batch size is 2048
- learning rate is 0.001, a lr_scheduler is setup, but not activated for training because of early stop
- I use simulation environment to check the training performance, and decide (the first performance peak) to stop training at 6 epochs for push, 16 epochs for lift. You can stop at the same epoch, or use simulation environment to search for a better model at similar ending point.

5. Deployment

For real world test, I find the cube pose data is quite noisy, so each time new pose data come, there is a valid check and smoothing process.

- Cube pose data process: if “delay” is less than 0.5 or (maybe “and” is better) “confidence” is greater than 0.6, cube pose is updated with moving average: cube_pose := 0.3 * cube_pose + 0.7 * new_cube_pose; otherwise, cube pose would be kept as the old value.

6. Other attempts

I also tried other methods, but their performance are not as good as this one. Possible reasons could be: 1. my code-level error; 2. my unfeasible hyperparameter setting, besides these I want to share my ideas about other reasons. (Very possible to be wrong)

- Add LSTM layer to MLP: the slippage and cube pose estimation error could disturb the hidden state.
- TD3-BC: Although there is argument that “Off-line RL is better than SL”, however most Off-line RL have (much) more hyperparameter than SL. To make Off-line RL more popular, maybe it’s better to find more robust algorithm.
  (TD3-BC: https://arxiv.org/abs/2106.06860)

Finally, I hope the solution would be a entry level benchmark.