Homework 1 Template

Use this template to record your answers for Homework 1. Add your answers using LaTeXand then save your document as a PDF to upload to Gradescope. You are required to use this template to submit your answers. **You should not alter this template in any way** other than to insert your solutions. You must submit all 15 pages of this template to Gradescope. Do not remove the instructions page(s). Altering this template or including your solutions outside of the provided boxes can result in your assignment being graded incorrectly.

You should also export your code as a .py file and upload it to the **separate** Gradescope coding assignment. Remember to mark all teammates on **both** assignment uploads through Gradescope.

Instructions for Specific Problem Types

On this homework, you must fill in blanks for each problem. Please make sure your final answer is fully included in the given space. **Do not change the size of the box provided.** For short answer questions you should **not** include your work in your solution. Only provide an explanation or proof if specifically asked.

Fill in the blank: What is the course number?

10-703

Problem 0: Collaborators

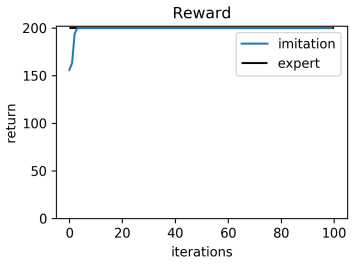
Enter your team members' names and Andrew IDs in the boxes below. If you worked in a team with fewer than three people, leave the extra boxes blank.

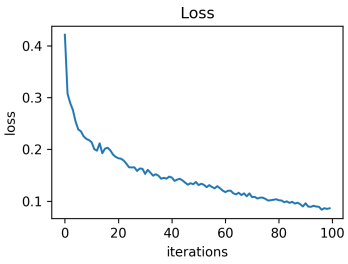
Name 1:	Akshay Sharma	Andrew ID 1:	akshaysh
Name 2:	Katayoon Goshvadi	Andrew ID 2:	kgoshvad
Name 3:	Keitaro Nishimura	Andrew ID 3:	knishimu

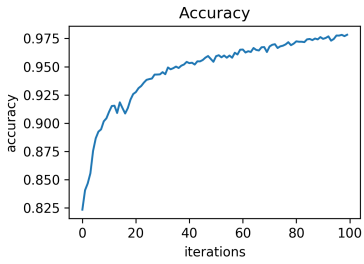
Problem 1: Behavior Cloning and DAGGER (50 pt)

1.1 Behavior Cloning (25 pt)

1.1.1 Plot Behavior Cloning (15 pt)

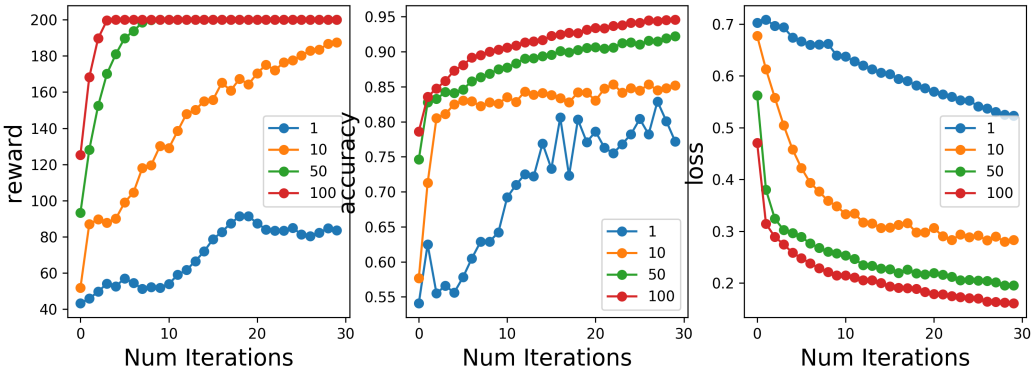






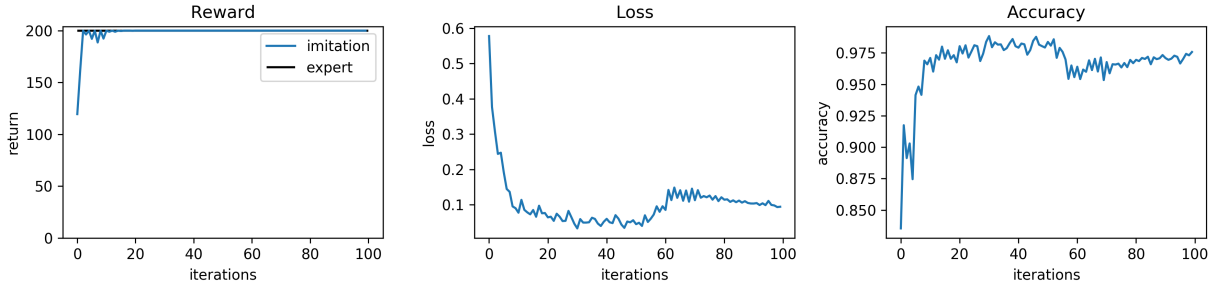
1.1.2 Plot Behavior Cloning with Varying Expert Episodes (10 pt)

The plots are property("reward","accuracy","loss") vs the number of iterations the model was trained for. Each curve in the plot is corresponding to the number of episodes that were used and has been averaged over all the seeds.

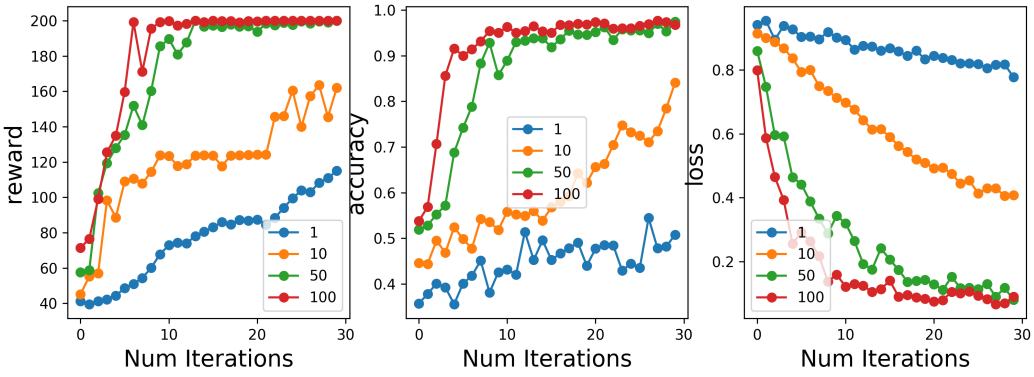


1.2 DAGGER (25 pt)

1.2.1 Plot DAGGER (10 pt)



1.2.2 Plot DAGGER with Varying Expert Episodes (10 pt)

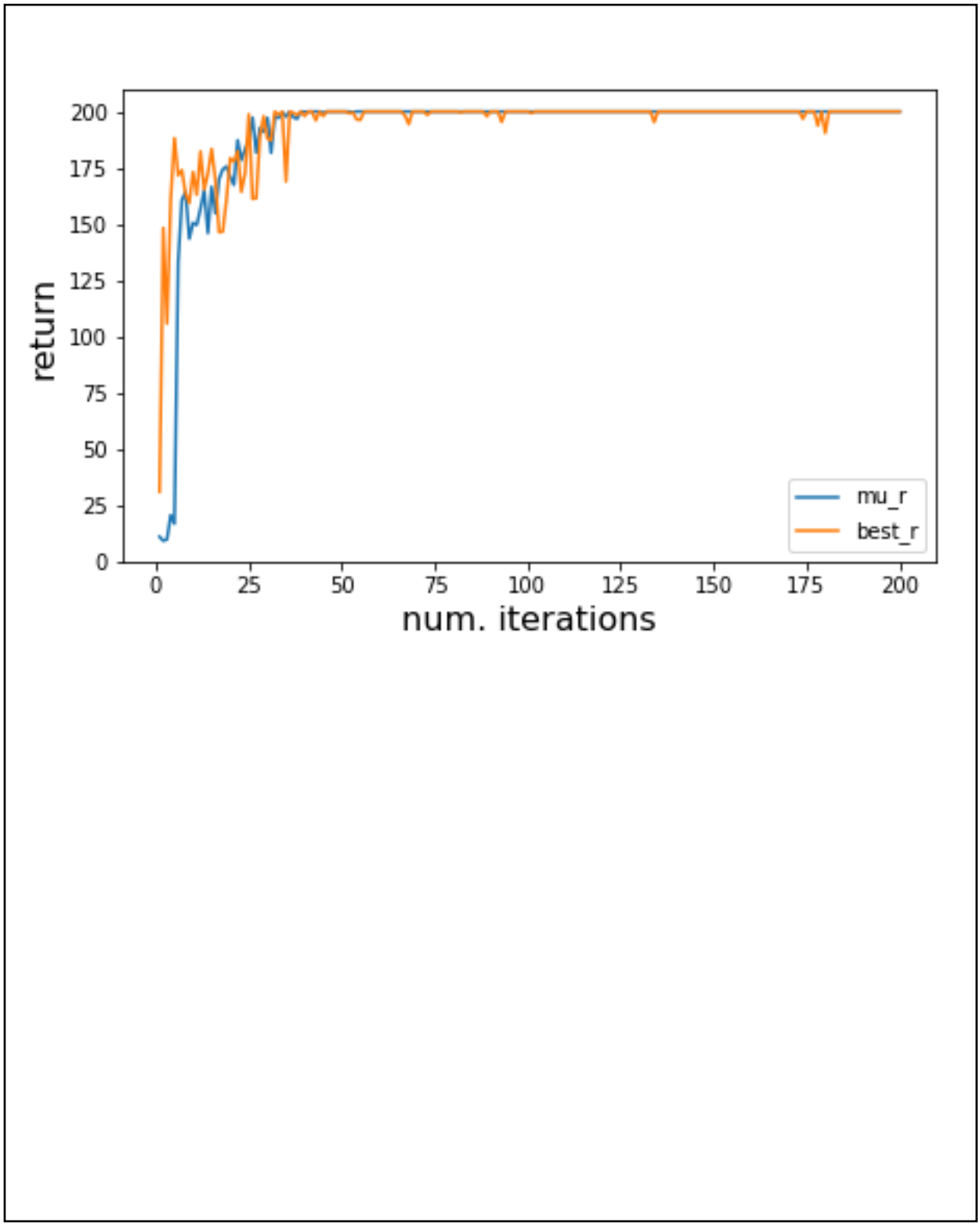


1.2.3 Compare Behavior Cloning and DAGGER (5 pt)

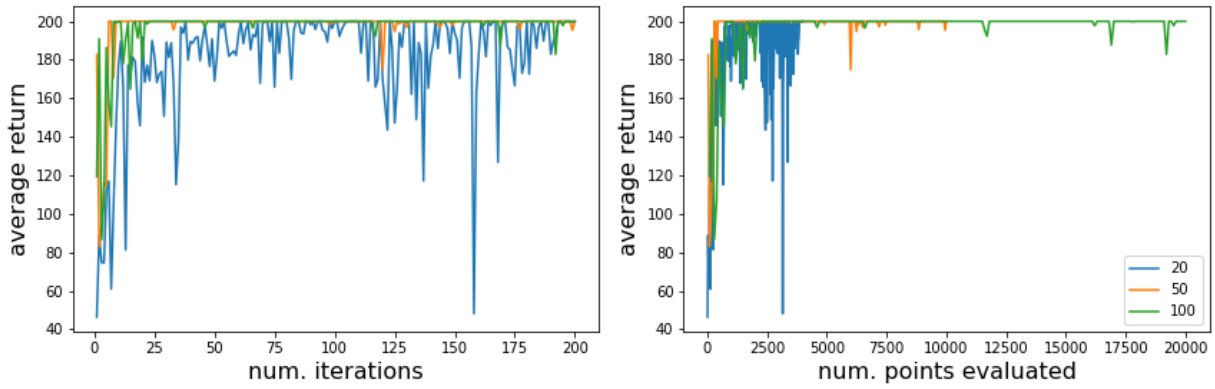
Based on the above graphs that we are getting from behavioural cloning and dagger, we can see that there exists more oscillation in the results of the dagger graphs. We expect this, because in behavioural cloning, the policy is trained on smaller number of states which is created based on a fixed expert policy. Whereas for dagger, every time that the policy is trained, a new set of states is generated based on our changing policy which makes distribution of on which it is trained larger, causing it to oscillate more.
This training on a larger distribution of states also allows the policy trained using dagger to converge faster albeit having more variance due to its generalization. In a more complex environment dagger might be slow initially because of the large variety of states the policy needs to explore, but once it has been trained on that wide variety, any new state that shows up is more or less within the distribution the policy has learned so far, which will allow faster convergence in the latter stages. The policy trained using behavior cloning (BC) has seen only the states which the expert visited, so it will perform well only on the small distribution of states from which expert's sees. If it sees any state which is out of that distribution or in other words a state which the expert never came across, the BC trained agent will not be able to perform well. Since the policy in BC is trained on smaller number of states which are more consistent, the graphs would be smooth.
To test the hypothesis that the policy trained by dagger would actually perform better in more complicated environments with uncertainties, we can test the two policies in order to evaluate their performance.
Finally, based on the graphs created with varying expert episodes, we can see that increasing amount of the expert data after some threshold will not affect the performance of the trained policy for both behavioural cloning and dagger.

Problem 2: CMA-ES (25 pts)

2.1 Plot CMA-ES (15 pts)



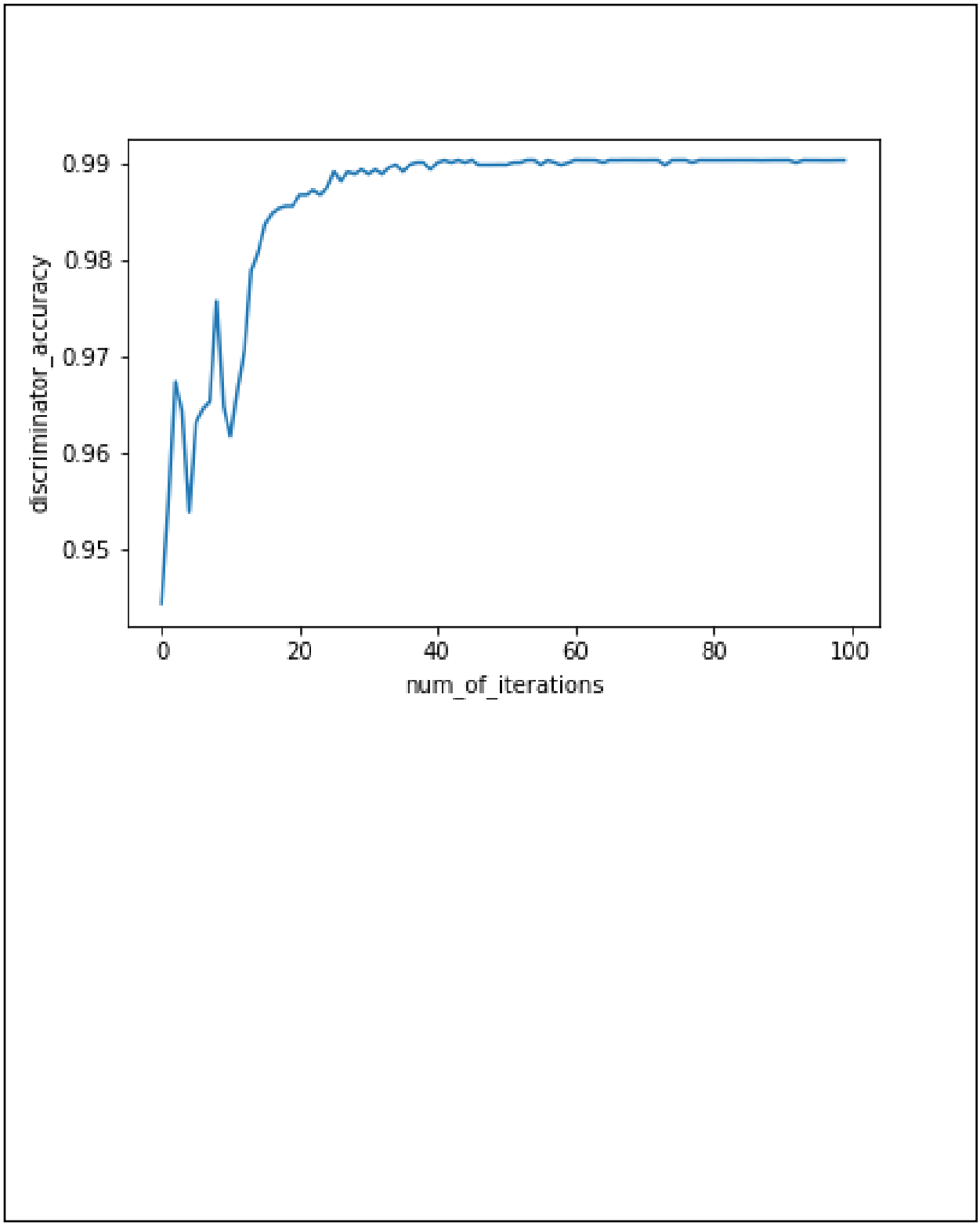
2.2 Plot CMA-ES with Varying Populations (10 pts)



The Y-axis label has a typo, it should be saying 'best return', in the original collab notebook the y axis was labeled "average return" and we did not change it. The plot is for the best return and not average return.

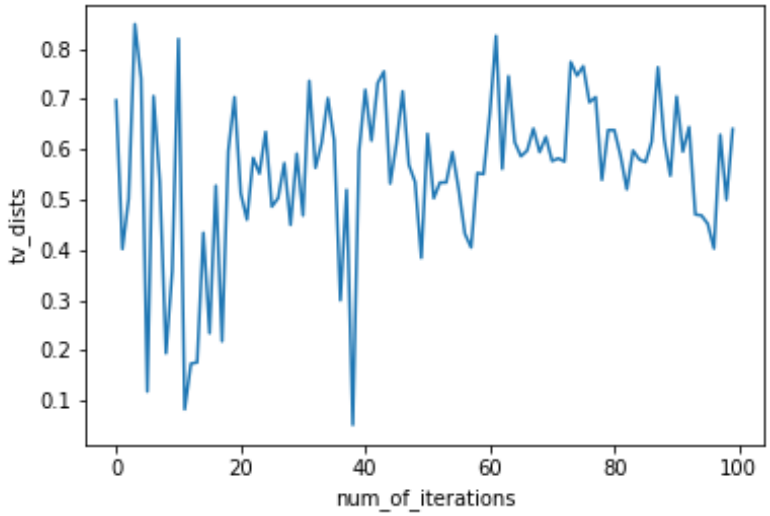
Problem 3: GAIL (25 pts)

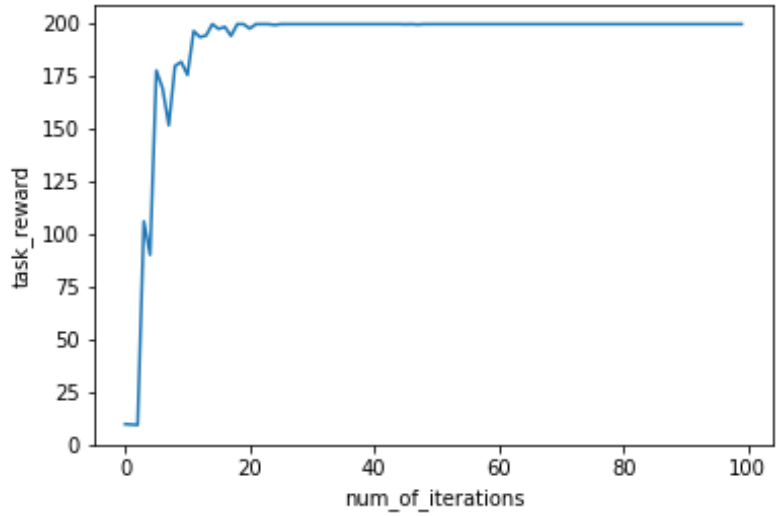
3.1 Plot Training Accuracy (5 pts)



3.2 Plot CMA-ES Task Reward and TV Distance (5 pts)

We observed that using only the log of discriminator probability as the reward will produce only negative rewards with wide range of variation for the policy which causes the agent to stop the episode early instead of accumulating negative rewards. This caused us adding one to the discriminator probability and getting log over that for reward function.

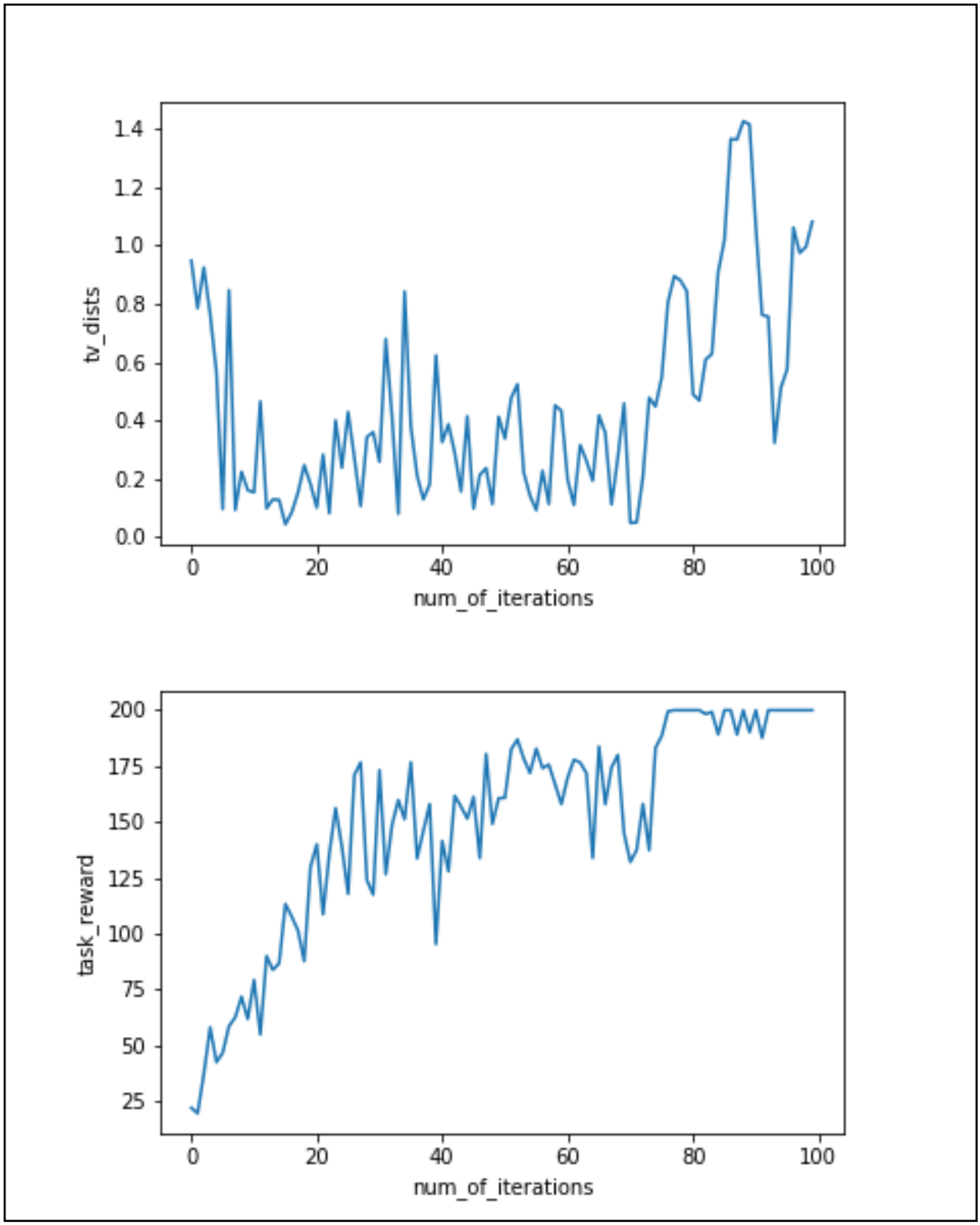




The TV-distance does not seem to be a very good metric to judge the performance of the policy in this environment because this metric only looks at the distribution of the x-coordinate of the cart. In the cartpole environment the policy can get higher reward if it learns to balance the pole for a longer time. The x-coordinate at which it does that is not that relevant. So it could very well happen that the learned policy's visited x-coordinate might be very different from that of the expert, but still the policy gets a high environment reward, which seems to be happening in the plots.

Note: We have also changed the way TV distance was being calculated in the code. Earlier the states generated before training the policy were being used to calculate it, but it should actually be done using states collected after training the policy.

3.3 Plot GAIL Task Reward and TV Distance (5 pts)

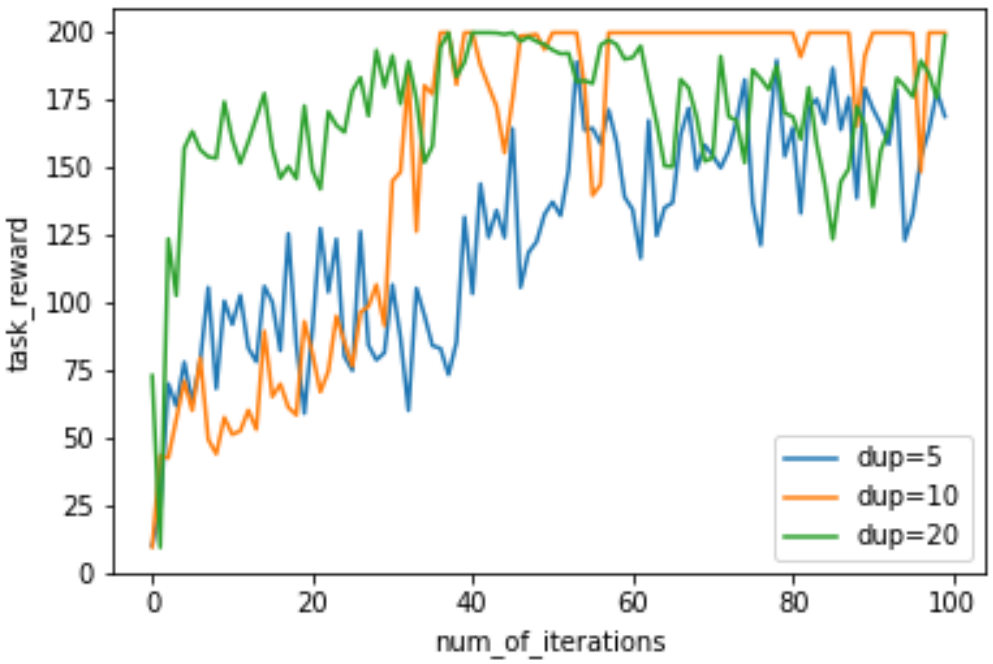


3.4 Vary Frequency(5 pts)

Describe your findings (3-5 sentences):

Training the discriminator too much makes it very good at discriminating between expert and student states, which results in lower rewards for the policy that causes slower training and convergence of the policy. So we expect to see that by increasing the number of times that we update our discriminator or in other words, increasing the accuracy of the discriminator, the overall performance of the policy in the environment and its gathered reward decrease which can be seen in the below graph as well. We got the reward task for different update periods of 5, 10 and 20(dup = discrimator update period).

Plot(s):



3.5 Overall Findings (5pts)

In behavioral cloning and dagger, we do not use any form of reward to train our policy and we just train a model to fit our (state-action) data which makes the trained policy more robust but in GAIL since we are using a reward which is computed by a changing discriminator, there exist more instability and oscillation in graphs. As the policy trained using dagger is trained on wider range of states it is more generalised and more robust in a stochastic environment with uncertainty whereas behavior cloning might give better results if the environment is simpler and the states visited by the agent are coming from the same distribution as the ones used to train the policy. As for GAIL, if we train a policy using GAIL with a reward function which properly encapsulates the task to be performed, we can expect a policy which is more generalised and robust and should perform better in more complex and uncertain environments. It is worth mentioning that behavioural cloning trains more smoothly than the other two methods and there exist more oscillation in both Dagger and Gail.

Extra (2pt)

Feedback (1pt): You can help the course staff improve the course by providing feedback. You will receive a point if you provide actionable feedback. What was the most confusing part of this homework, and what would have made it less confusing?

The assignment was pretty clear in stating what was needed in every question. The only thing, is that you provided the expert only as a tensorflow model, which does not allow us to work with other deep learning libraries. It would be good to provide experts for atleast Pytorch too, in case we want to write the code using Pytorch.

Time Spent (1pt): How many hours did you spend working on this assignment? Your answer will not affect your grade.

Alone	2
With teammates	20
With other classmates	0.5
At office hours	0.5