Summary for Safe Reinforcement Learning with Model Uncertainty Estimates

Siddharth Nayak

1 Introduction:

The paper 'Safe Reinforcement Learning with Model Uncertainty Estimates' is a work on getting knowledge about an agent's uncertainty in unknown scenarios and make them take actions which give more priority to safety rather than the goal/objective to be achieved. The authors use MC-Dropout and Bootstrapping method to give uncertainty estimates for actions taken by the agent. They combine this with reinforcement learning to get an agent which is more robust in novel scenarios and takes safer actions as compared to an uncertainty unaware model. This agent basically knows what it does not know and thus makes it take safer actions when it is in an unseen environment. The work in this paper is quite important in the current research field as safety is an important aspect in robots. Knowing the scenarios in which an agent is going to fail can help in devising new algorithms or methods to tackle the uncertainty of the agents to make them more reliable and transparent.

2 Model

The model uses a set of Long-Short-Term-Memory(LSTM) networks[2] to predict collision probabilities for a set of actions. They use Bootstrapping[7],[8] and MC-Dropout[6] to get a distribution over the prediction. Bootstrapping and MC-Droupout are used by the authors to get estimates of the model uncertainty which is quite impressive. They also train multiple LSTMs on different subsets of the dataset. The bootstrapping ensures that the model gives high certainty to the data-points which are similar to the ones in the dataset and lower certainty to the ones which are not.

3 Experiments

In the experiments the authors get the performance of the agent which is uncertainty aware against an agent which is not. The uncertainty aware agent does choose actions which make it avoid obstacles with more distance between it and the obstacle.

The action selected is determined by the following equation:

$$u^*_{t:t+h} = \underset{u \in U}{\operatorname{argmin}} \big(\lambda_v Var(P_{coll}) + \lambda_c E(P_{coll}) + \lambda_g t_{goal} \big)$$

The value of λ_g and λ_c have to be chosen in such a way that the predicted collision cost (terms involving λ_c and λ_v) is greater than goal cost. They do not multiply the variance term with the selected velocity as stopping or reducing velocity may not always be safe.

In one of the experiments, the author increases the value of λ_v to get better convergence. This method is quite impressive as this increases the penalty in highly uncertain actions over time. This makes the agent explore efficiently in directions of high model uncertainty in the early phases of training. This also helps in escaping local optimums. The effect of the increase in the value of λ_v is more prevalent in the case of challenging avoidance cases.

4 Possible Improvements

In the paper the authors collect the dataset by executing several episodes and storing the observation-action history. One of the ways to make it more robust is by adding adversarial noise in the states as well as actions while training as done in [3][4]. This will make the agent more robust to noise in the system. One of the things which could be tried out is by making the agent learn by having copies of itself in the environment motivated by the work on Asynchronous Policy Updates by Gu et.al[5]. One of the two following cases might happen:

- The agents will not converge to an optimum due to oscillations between the agents certainty.
- The agents will initially learn with an uncollaborative agents and then slowly converge towards a more robust agent.

5 Conclusion

Overall the paper is quite well written along with experimental results to prove the model effectiveness. This paper adds on to the current research in safe reinforcement learning. The paper content is well written and I could understand it well. I would like to thank Björn Lütjens for taking out his time to have a discussion about the paper and it's extensions.

6 References

[1]: Bjorn Lutjens, Michael Everett, Jonathan P. How. "Safe Reinforcement Learning with Model Uncertainty Estimates". ICRA 2019.

[2]:S. Hochreiter and J. Schmidhuber, Long short-term memory, Neural Comput., vol. 9, no. 8, pp. 17351780, Nov. 1997.

[3] A. Mandlekar, Y. Zhu, A. Garg, Li Fei, S Savarese. "Adversarially Robust Policy Learning: Active Construction of Physically-Plausible Perturbations". 2017

[4] L. Pinto, J. Davidson, R. Sukthankar, A. Gupta. "Robust Adversarial Reinforcement Learning". 2017

[5]: S. Gu, E. Holly, T. Lillicrap, S. Levine. "Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates".

[6]:Y. Gal and Z. Ghahramani, Dropout as a Bayesian approximation: Representing model uncertainty in deep learning, in Proceedings of the 33rd International Conference on Machine Learning (ICML-16), 2016.

[7]:B. Lakshminarayanan, A. Pritzel, and C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, in Advances in Neural Information Processing Systems 30. Curran Associates, Inc., 2017, pp. 64026413.

[8]:I. Osband, C. Blundell, A. Pritzel, and B. Van Roy, Deep exploration via bootstrapped dqn, in Advances in Neural Information Processing Systems 29. Curran Associates, Inc., 2016, pp. 40264034.