THESIS PROSPECTUS

Reinforcement Meta-Learning (REML) or Learning to Learn by Gradient Descent as a Markov Decision Process

Matthew Stachyra

Sep 2023

1 INTRODUCTION

A deep learning model of a target function is always constrained by available data in the target domain. Few data examples limits the ability of the model to represent the target function at unknown points. Trying to learn under this constraint is known as few shot learning. One paradigm applied to manage this constraint is meta-learning or "learning to learn". This thesis contributes a new meta-learning algorithm. We are using the definition of meta-learning as proposed initially by Thrun et al in 1998: an algorithm is learning to learn if its performance at each task (where there is more than 1 task, more than 1 performance measure, and more than 1 training experience) is expected to improve with the number of tasks and training experiences on these tasks. The new meta-learning algorithm is Reinforcement Meta-Learning (REML), which casts learning to learn as a parameterized markov decision process.

REML is composed of a supervisory agent in a system with the models it generates for provided tasks in a domain. The supervisory agent is implemented with reinforcement learning with a parameterized action space, and the models are implemented as deep neural networks in the environment of the meta-learner agent. The supervisory agent composes and trains the models layer by layer for each task. In this way the reinforcement learning meta-learner is responsible for both hyperparameters and parameters.

This is to my knowledge the first work to use reinforcement learning as the meta-learner in a model agnostic manner akin to MAML (model agnostic meta-learning) proposed by Finn et al 2018. Other works at the intersection of reinforcement learning and meta-learning include RL-driven hyperparameter search (i.e., neural architecture search) and meta-reinforcement learning, where both the inner and outer loops are RL agents. The performance of REML will be evaluated for regression, classification, and reinforcement learning tasks on the same benchmarks as the MAML (Model agnostic meta-learning) paper by Finn et al 2018. As time allows, or as future work, I will investigate certain convenient properties inherent to this design. One of these may be robustness to learning on unrelated tasks, relative to offline trained tasks. I hypothesize that because layers in REML are composed individually, they have a more expressive quality in their availability to be sequenced in different combinations. This is as opposed to a single initial set of policy parameters adapted for each task, as is in MAML.

The research questions this work seeks to answer are: (1) can REML enable fast learning for new tasks?, (2) can REML be model agnostic and perform for regression, classification, and reinforcement learning?, and (3) is REML robust to unrelated tasks at meta test time?

To answer these research questions, preliminary questions answered are: (1) how to design a neural network using reinforcement learning?, (2) how to train a neural network using reinforcement learning?, (3) how to transfer learning across tasks using reinforcement learning?, and (4) how to enable meta-learning with reinforcement learning as the meta-learner?

1.1 COMMITTEE

- Jivko Sinapov
- Elaine Short
- Liping Liu

1.2 ALGORITHM

Algorithm 1 Reinforcement Meta-learning (REML)

Require: : α, β: step-size hyperparameters
Require: T: set of tasks t
Require: $f_{\theta_{super}}$: RL agent parameterized θ_{super}
Require: L: set of layers l
  Randomly initialize θ_{super}
  Randomly initialize all $l \in L$ with θ_{sub}
  for $t_i \in T$ **do** ▷ this constitutes one epoch of training
    Sample d from $\mathcal{D} = \{\boldsymbol{x}^{(j)}, \boldsymbol{y}^{(j)}\}$ for t_i
    Pass d through an initial layer in L to get initial state s_k
    while not done **do** ▷ capped by number of training steps
      Get next layer $l_k \in L$ via $f_{\theta_{super}}(s_k)$
      Expand θ_{sub} with l_k's θ
      Evaluate $\mathcal{L}_{\theta_{super}}$ for chosen f_{super}
      Evaluate $\mathcal{L}_{\theta_{sub}}$ for chosen f_{sub}
      $\theta_{super} \leftarrow \theta_{super} + \alpha \nabla \mathcal{L}_{\theta_{super}}$
      $\theta_{sub} \leftarrow \theta_{sub} + \beta \nabla \mathcal{L}_{\theta_{sub}}$
      Sample d from $\mathcal{D} = \{\boldsymbol{x}^{(j)}, \boldsymbol{y}^{(j)}\}$ for t_i
      Update state s_k via $f_{\theta_{sub}}(d)$

References

[1] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, 29, 2016.

[2] Yutian Chen, Matthew W Hoffman, Sergio Gómez Colmenarejo, Misha Denil, Timothy P Lillicrap, Matt Botvinick, and Nando Freitas. Learning to learn without gradient descent by gradient descent. In *International Conference on Machine Learning*, pages 748–756. PMLR, 2017.

[3] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[4] Jessica B Hamrick, Andrew J Ballard, Razvan Pascanu, Oriol Vinyals, Nicolas Heess, and Peter W Battaglia. Metacontrol for adaptive imagination-based optimization. *arXiv preprint arXiv:1705.02670*, 2017.

[5] Sepp Hochreiter, A Steven Younger, and Peter R Conwell. Learning to learn using gradient descent. In *Artificial Neural Networks—ICANN 2001: International Conference Vienna, Austria, August 21–25, 2001 Proceedings 11*, pages 87–94. Springer, 2001.

[6] Eric Mitchell, Rafael Rafailov, Xue Bin Peng, Sergey Levine, and Chelsea Finn. Offline meta-reinforcement learning with advantage weighting. In *International Conference on Machine Learning*, pages 7780–7791. PMLR, 2021.

[7] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

[8] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *arXiv preprint arXiv:1807.05960*, 2018.

[9] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural

networks. In *International conference on machine learning*, pages 1842–1850. PMLR, 2016.

[10] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

[11] Sebastian Thrun and Lorien Pratt. Learning to learn: Introduction and overview. In *Learning to learn*, pages 3–17. Springer, 1998.

[12] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.