Using Deep Learning tools for fitting Reinforcement Learning Models

Milena Rmus (milena_rmus@berkeley.edu)
UC Berkeley Department of Psychology, 2121 Berkeley Way
Berkeley, CA 94704 US

Jimmy Xia (jimmyxia@math.berkeley.edu)
UC Berkeley Department of Mathematics, 970 Evans Hall
Berkeley, CA 94704 US

Jasmine Collins (jazzie@berkeley.edu)
UC Berkeley Department of EECS, 2121 Berkeley Way
Berkeley, CA 94704 US

Anne Collins (annecollins@berkeley.edu)
UC Berkeley Department of Psychology, 2121 Berkeley Way
Berkeley, CA 94704 US

Abstract

Computational cognitive modeling has advanced our understanding of learning and decision-making. However, the set of models we use is often limited by technical constraints, such as feasibility of model-fitting. Most modeling methods require computing the likelihood of the data under the model (e.g. finding parameters that maximize it). However, many computational models have intractable likelihoods, and workarounds designed for this problem only work on a small subset of models with specific assumptions. To address this issue, we tested a method using deep learning tools to estimate model parameters without estimating intractable likelihoods. Our results show that we can adequately recover parameters using this end-to-end approach. Our work contributes an important new tool to the ongoing development of computational techniques that will enable researchers to consider a broader set of models and develop better theories of complex human cognition.

Keywords: computational modeling; reinforcement learning; hierarchical reinforcement learning; deep learning

Introduction

Reinforcement learning (RL) models have played an integral role in formalizing cognitive processes underlying reward-based learning. They have significantly contributed to our understanding of neural mechanisms (i.e. dopaminergic signals, (Schultz, Dayan, & Montague, 1997)), as well as individual differences (i.e. impaired computational mechanisms in clinical populations (Collins, Brown, Gold, Waltz, & Frank, 2014)) of reward-driven behavior.

Model-fitting methods commonly used to fit RL models are all dependent on evaluating the likelihood of the data (i.e. choices) under the given model (maximum likelihood estimation, MLE; maximum a posteriori, MAP; hierarchical Bayesian estimation) (Katahira, 2016; Lee, 2011; van Geen & Gerraty, 2021; Baribault & Collins, 2021). However, there are important classes of RL models with intractable likelihood; for example, some RL models assume that observed choices are dependent on discrete non-observable variables (i.e. what rule participants followed to generate choices). This can result in an intractable problem of integrating over uncertainty over non-observable variables, making it impossible to fit these models using likelihood-based methods. Various workarounds developed for models with intractable likelihood (Approximate Bayesian Computation; Probability Density Approximation; Inverse Binomial Sampling) (Turner et al., 2013; van Opheusden, Acerbi, & Ma, 2020) also do not apply broadly, because they assume independence between datapoints, which is not the case for many RL models. Thus, researchers often avoid considering the models with intractable likelihood, even if these models provide a more plausible theoretical account of the process they are interested in capturing.

We propose a novel model-fitting approach based on deep learning, with the aim of developing a general-purpose tool for fitting a wide range of RL models, including the ones with intractable likelihood.

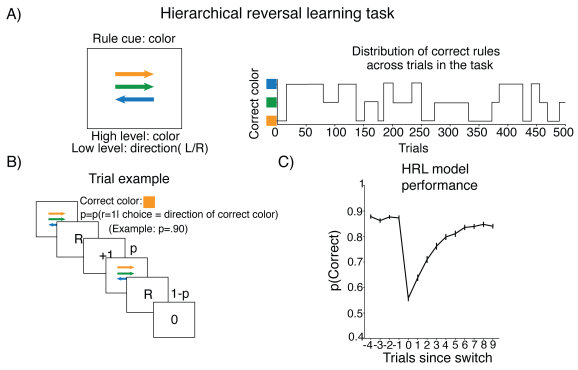

Figure 1: Task environment, and performance of simulated model. A) Hierarchical reversal learning task. B) Example of noisy feedback in the task. C) Hierarchical RL model with intractable likelihood can perform this task adequately.

Methods

While models with intractable likelihood cannot be fit using likelihood-based methods (and most alternatives), they can be easily simulated. We leveraged this property to create a large supervised learning dataset, and used an end-to-end neural network architecture to learn the mapping from simulated model trajectories to model parameters. As a proof of concept, we first benchmark the performance of parameter recovery for a simple RL model with a tractable likelihood, and then for a hierarchical RL models with an intractable likelihood.

Intractable likelihood dataset generation

Hierarchical reversal learning task. We developed a novel task environment, with a simple but plausible model with intractable likelihood. In the task, agents need to learn which arrow's direction to follow, by pressing left or right key, in order to get rewarded; the correct arrow changes unpredictably (Fig: 1A). If an agent chooses the side consistent with that of the correct arrow, it gets rewarded with high probability $p = 0.90$ (Fig: 1B); otherwise, it gets punished with the same high probability.

Hierarchical reinforcement learning model. We considered a Hierarchical reinforcement learning (HRL) (Fig: 1C) agent that tracks the value of each arrow, and chooses between the arrows noisily (with some tendency to repeat the choice from the previous trial): $p(arrow) \propto exp(\beta Q + \kappa\ \text{same}(arrow, arrow_{t-1}))$. The arrow the agent chooses is non-observable, as we only know which direction the agent chose. Following the choice of the arrow, the agent greedily chooses the direction of the chosen arrow (observable). The agent then updates the value of the selected arrow based on observed outcome: $Q_{t+1}(arrow) = Q_t(arrow) + \alpha(r - Q_t(arrow))$. To compute the likelihood of each subsequent choice we need to integrate over uncertainty of what the unobserved choice was on all past trials. The number of terms for this integration increases exponentially with each time-point, making the likelihood intractable beyond the first several trials.

Deep learning approach

The neural network (NN) structure used is inspired by previous work (Dezfouli et al., 2019). The NN consists of a recurrent neural network (RNN) with 70 bidirectional long short term memory (LSTM) cells, and a 4-layer feed-forward network (70 units in first two layers, 10 units in the third layer and p units in the output layer where $p =$ number of parameters). RNNs retain information across input sequences, making them suitable for data with sequential dependencies. The terminal state of the RNN is encoded into a p-dimensional space by the feed-forward network. We used ADAM optimizer, mean squared error (MSE) loss function, and rectified linear unit (ReLU) activation function in all layers (linear activation function in the last one). We trained the network using simulated agents and true parameter values, and validated the network performance on the out-of-sample validation set.

Results

We first simulated 30000 training and 3000 validation agents from a simple 2-parameter RL model (with tractable likelihood) on a different task. We trained the network for 600 epochs, with the batch size of 512. We compared the neural network performance (MSE loss) against the MSE of the standard method we also used to estimate model parameters (MAP). Both parameters of the model were well recovered using our DL approach (Fig:2A), with DL loss for validation data in the range of error margins of the standard method (MAP; Fig:2B). We can, therefore, justify the DL approach as it performs comparably to the standard method, when both approaches are applicable.

Next, we simulated 800000 training and 10000 validation agents from the HRL model, with fixed β. We trained the network to recover other parameters for 1200 epochs with a batch size of 1024. HRL learning rate recovery was adequate (Fig:2C), providing preliminary evidence of success in using DL tools for estimating parameters of RL models with intractable likelihood.

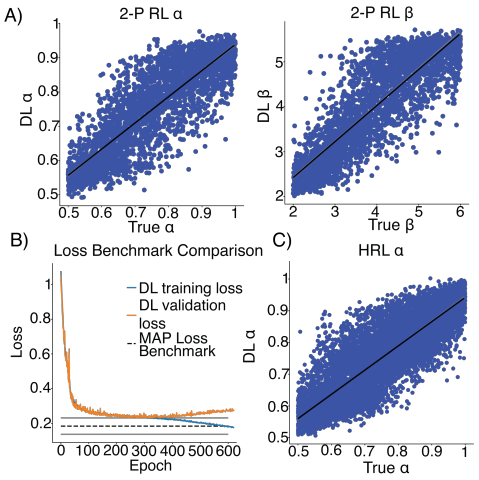

Figure 2: Successful recovery of a 2-parameter RL model with the DL approach (A), with loss within the bounds of that obtained with MAP (B). C) Successful recovery of the learning rate parameter from the model with intractable likelihood.

Discussion

Our results show that DL tools can be used for fitting RL models with intractable likelihood. We will further test the robustness of the DL approach (i.e. missing trials, different models), as well as experiment with different DL structures (i.e. transformers as advanced structures for sequential data (Devlin, Chang, Lee, & Toutanova, 2018)). Developing such tools could increase the range of cognitive models researchers can test, for which existing fitting methods cannot be used.

Acknowledgments

We thank Kshitiz Gupta, Yi Liu, Jaeyoung Park, Bill Thompson and Rich Ivry for their help.

References

Baribault, B., & Collins, A. (2021). Troubleshooting bayesian cognitive models: A tutorial with matstanlib.

Collins, A. G., Brown, J. K., Gold, J. M., Waltz, J. A., & Frank, M. J. (2014, October). Working Memory Contributions to Reinforcement Learning Impairments in Schizophrenia. *The Journal of Neuroscience*, *34*(41), 13747–13756.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dezfouli, A., Ashtiani, H., Ghattas, O., Nock, R., Dayan, P., & Ong, C. S. (2019). Disentangled behavioural representations. *Advances in neural information processing systems*, *32*.

Katahira, K. (2016, August). How hierarchical models improve point estimates of model parameters at the individual level. *Journal of Mathematical Psychology*, *73*, 37–58.

Lee, M. D. (2011). How cognitive modeling can benefit from hierarchical bayesian models. *Journal of Mathematical Psychology*, *55*(1), 1–7.

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*(5306), 1593–1599.

Turner, B. M., Forstmann, B. U., Wagenmakers, E.-J., Brown, S. D., Sederberg, P. B., & Steyvers, M. (2013, May). A Bayesian framework for simultaneously modeling neural and behavioral data. *NeuroImage*, *72*, 193–206.

van Geen, C., & Gerraty, R. T. (2021). Hierarchical bayesian models of reinforcement learning: Introduction and comparison to alternative methods. *Journal of Mathematical Psychology*, *105*, 102602.

van Opheusden, B., Acerbi, L., & Ma, W. J. (2020, January). Unbiased and Efficient Log-Likelihood Estimation with Inverse Binomial Sampling.