Literature Review

A Distributional Perspective on Reinforcement Learning

Abdelrahman Ahmed

28 August 2017

1 Introduction

This report assesses the research paper, 'A Distributional Perspective on Reinforcement Learning' by the authors, Marc G. Bellemare, Will Dabney and Remi Munos, published in the proceedings of the 34th International Conference on Machine Learning (ICML) in 2017. Bellemare et al.'s paper will be assessed on several criteria. Firstly, content is assessed through providing some background information and describing the methods and findings of the paper. Secondly, the novelty and innovation of the paper is described. Thirdly, the technical quality is examined. The presentation of the paper is then assessed on the flow and depth of their argument. Finally, possible applications for the research work and suggested improvements are identified.

2 Content

The content of Bellemare et al.'s paper is well-researched and offers new insights in the field of reinforcement learning. It shows the approach the authors undertook to demonstrate that modelling the variation in the reward value, rather than the typical average value, results in improved accuracy and training performance in reinforcement learning systems. This section of the report provides background information about reinforcement learning and the contents of the paper. It also examines the methods used by the authors, and their findings.

2.1 Background

Reinforcement learning (RL) is an area of machine learning that is inspired by psychological and neuroscientific perspectives on animal behaviour (Mnih et al., 2015). It is the problem of getting an agent to act in an environment in such a way as to maximise its rewards, by predicting the long-term impact of its actions. Unlike supervised learning, the agent is not given labelled data to

indicate whether their action is correct for a specific scenario (Sutton & Barto, 1998). In a typical reinforcement learning system, the algorithm predicts the average reward value it receives from multiple attempts at a task (Bellemare et al., 2017b).

Bellemare et al. (2017b, para. 2) argue that 'randomness is something we encounter everyday and has a profound effect on how we experience the world', and that these variations should be accounted for when designing algorithms. They show that it is possible to model these variations in the reward the agent receives, termed value distribution. The authors demonstrate that there is a variant of Bellman's equation (Figure 1) which can predict the possible outcomes without aggregating them into an average value, allowing the agent to 'model its own randomness' (Bellemare et al., 2017b, para. 12).

$$Q(x, a) = \mathbb{E}\, R(x, a) + \gamma\, \mathbb{E}\, Q(X', A').$$

Figure 1: Bellman's equation describes the value (Q) in terms of expected reward and the expected outcome of the random transition (x, a) -¿ (X', A'), and the discount factor (gamma) determines the importance of future rewards (Bellemare et al., 2017a)

2.2 Methods

Bellemare et al. (2017a) begin by providing some background information on value distributions and how distributional perspectives are currently being applied to other works, and the benefits of it applied to reinforcement learning. The authors then evaluate the theoretical results of the distributional equations and how they can be applied. Based on the theoretical results and other works, they propose and implement a new algorithm based on the distributional variant of Bellman's equation, where the average reward value output of a Deep Q-Network (DQN) agent is replaced with a distribution of possible values, or atoms, which can be adjusted. Bellemare et al. assess the performance of the algorithm against Atari 2600 games in the Arcade Learning Environment (ALE). An optimal number of atoms is found, 51, by varying the number of atoms and evaluating the training performance of the algorithm (Figure 2). The performance is also evaluated against a typical DQN agent, where it attained state-of-the-art performance (Bellemare et al., 2017a).

Figure 2: Varying the number of atoms in the distribution in a number of games, showing the average scores are improving (Bellemare et al., 2017a)

The figures below visualise the typical value distributions observed in Bellemare et al.'s experiments. The value distributions demonstrate how the agent determines the safe actions from the losing actions, where safe actions have similar distributions and the losing actions are assigned low or zero probability. For example, Figure 3 shows that the agent assigned a probability of zero to the three actions that will lead to the agent losing the game by firing their laser too early.

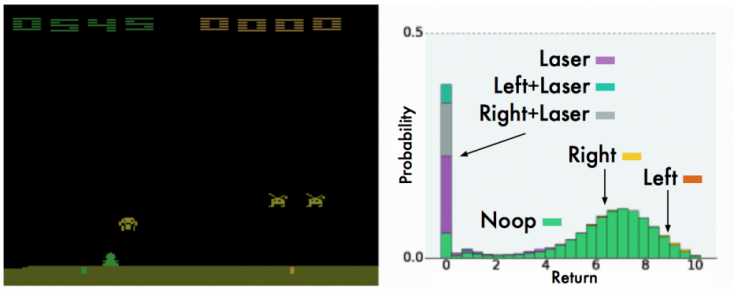

Figure 3: Agent playing the Atari 2600 game, Space Invaders. It shows a typical learned value distribution where the different colours indicate the different actions used in the game. Noop means no operation (Bellemare et al., 2017a)

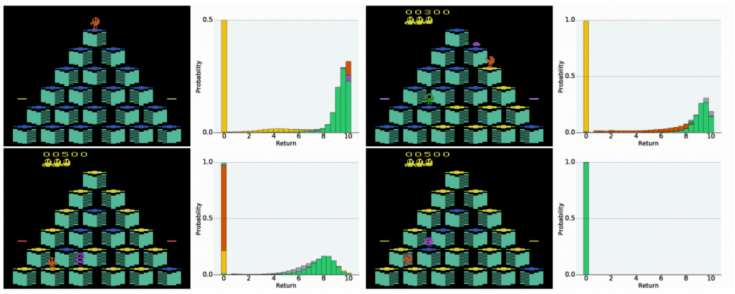

Figure 4: Agent playing the Atari 2600 game, Q*bert. Top, left and right: Predicting which actions are not recoverable. Bottom-Left: The distribution shows costly consequences for performing the wrong actions. Bottom-Right: The distribution shows the agent has made a big mistake (Bellemare et al., 2017a)

2.3 Findings

Bellemare et al. (2017a, p. 454) state that 'the distributional update keeps separated the low-value, "losing" event from the high-value, "survival" event, rather than average them into one (unrealizable) expectation', which shows why their approach is more successful. Bellemare et al. (2017a) achieved state-of-the-art results using the 51-atom agent (C51) across the suite of Atari 2600 games. They found that it significantly outperformed other algorithms, such as the DQN agent, and surpassed the current state-of-the-art results by a large number. Figure 5 shows that training performance of the C51 surpasses the performance of a fully trained DQN and a human player by a wide margin. It achieved 75% of a trained Deep Q-Network performance in 25% of the time.

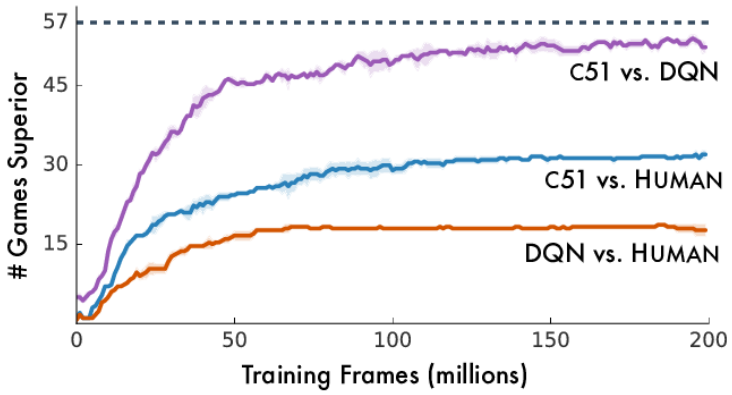

Figure 5: Performance comparison between the new algorithm (C51), the original DQN and human at the set of Atari 2600 games (Bellemare et al., 2017a)

They also observed surprising randomness in the Atari 2600 games, although the underlying emulator, Stella, is completely predictable. This inherent randomness is attributed to partial observability, where the agents cannot accurately predict when their score will increase. These findings highlight the limitation of the agent's understanding; however, it does not directly affect the performance.

Bellemare et al.'s new algorithm also managed to exceed the performance of other state-of-the-art algorithms, such as Double DQN, by a wide margin and achieved remarkable results in a gamut of Atari 2600 games. The C51 agent obtained a mean score improvement of 126% and a median of 21.5% on a normalised scole scale with respect to random and DQN agents, confirming the advantages and benefits of C51 and the value of the distributional perspective (Bellemare et al., 2017a).

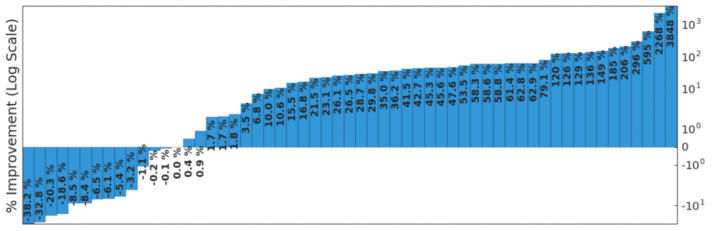

Figure 6: The per-game improvement percentage of the C51 agent compared to a Double Deep Q-Network agent (Bellemare et al., 2017a)

In addition to outperforming a standard Deep Q-Network (DQN) agent, the C51 agent showed significant improvement over a Double DQN agent as well.

Double DQN is an improvement upon the original Deep Q-Network which uses Double Q-Learning to reduce the agent's overestimation of values, achieving state-of-the-art results against the DQN (van Hasselt et al.). Figure 6 shows a significant percentage improvement of the C51 agent against the Double DQN agent per Atari game.

Bellemare et al.'s paper provides evidence that the distributional perspective leads to improved performance and more stable reinforcement learning. This shows that the distributional perspective is more powerful than expected on the tested range of problems, and thus it is likely that it will be beneficial to other areas in machine learning.

3 Innovation

Although a distributional perspective is not in itself a novel idea, it has only been used for specific purposes in reinforcement learning. Bellemare et al. (2017a) note previous works that used a distributional perspective. Dearden et al. (1998) modelled parametric uncertainty and Morimura et al. (2010) designed risk-sensitive algorithms. Bellamere et al. believe it has an important role to play in reinforcement learning algorithms. Their findings demonstrate that applying a distributional perspective leads to improved performance in reinforcement learning. They stated that 'it might just be the beginning for this approach' and that there's a possibility that 'every reinforcement learning concept could now want a distributional counterpart' (Bellemare et al., 2017b, para. 13).

Bellemare et al.'s work offers significant contributions to the field of reinforcement learning. They showed that it is possible and favourable to predict the potential outcomes rather than simply average them, by using a variant of Bellman's equation. The implementation of their ideas did not require substantial changes to the existing Deep Q-Network architecture, substituting the average reward value output with a distribution of 51 possible values. As well as updating the learning rule to reflect the transition to the distributional counterpart of Bellman's equation. This new architecture is called Categorical DQN.

4 Technical Quality

Bellemare et al. (2017a) acknowledged and discussed the related work of Morimura et al., who proposed solutions to learn distributions in reinforcement learning using the Monte Carlo estimation method. They then illustrated the aim of their paper is to build upon the work done by Morimura et al. and demonstrate the possibility of designing a practical distributional algorithm that is based

on Bellman's equation. This shows that the paper is well-researched and the authors are familiar with similar work in reinforcement learning.

Bellemare et al. provided supplementary material about the algorithm design, and figures about the evaluation and results. They included additional details and proofs of the distributional variant of Bellman's equation, and supplementary videos showcasing the change in reward distribution as the agent is training across different games such as Space Invaders, Pong and Seaquest. Although the authors did not provide the source code, the algorithm details, equations, proofs and pseudo code in the paper are sufficient to implement the new algorithm by amending a DQN agent to use a value distribution for reward outcomes. Albeit the paper is relatively new, several implementations of the paper using Tensorflow, a machine learning library, and OpenAI Gym, a learning environment that includes wide range of games including Atari, have been published on GitHub to replicate the results; however, at the time of writing, the results are incomplete.

Bellemare et al. conducted various experiments on the new algorithm and evaluated it against a typical Deep Q-Network (DQN) agent to compare their performance. They also adjusted the number of atoms in the new algorithm in an attempt to find the optimal number of atoms in the distribution. The algorithm exceeded the performance of the DQN agent and achieved state-of-the-art results in a number of classic Atari 2600 games.

5 Presentation

Bellamere et al.'s paper was well presented and easy to understand. The authors clearly describe the issues with the current reinforcement learning approach and why a distributional perspective would be favourable.

The authors provided the necessary details on how the algorithm was evaluated, including the learning environment and settings for the DQN architecture. They also presented their findings from varying the number of atoms in the new algorithm while testing its performance.

The paper was mostly easy to follow, aside from the section about finding the distributional variant of the Bellman's equation. This section would be difficult to follow and grasp without some prior knowledge in reinforcement learning. The authors however make a note to the reader that this section is skippable if the reader is interested in the end result of the algorithm, and the evaluation that follows.

In addition, the authors published a blog post on the DeepMind Blog to showcase the algorithm performance compared to a typical DQN agent by providing a real world example, train commute times, to further explain the reasoning behind the algorithm design and the importance of modelling value distributions in reinforcement learning without any technical jargon.

6 Application

Reinforcement learning has a wide range of applications. It has been in use for decades in playing games such as backgammon and checkers, robotics, elevator dispatching strategies and job scheduling (Sutton & Barto, 1998). More recently, DeepMind has introduced deep reinforcement learning by incorporating deep neural networks with Q-learning, a model-free reinforcement learning technique, to create a novel artificial agent, termed Deep Q-Network (DQN) to learn successful policies from high-dimensional inputs, such as pixels (Mnih et al., 2015). The DQN agent achieved state-of-the-art results compared to previous algorithms, and reached human-level performance across a set of 49 classic Atari 2600 games (Mnih et al., 2015).

Bellemare et al.'s contributions further develop deep reinforcement learning by replacing the Deep Q-Network agent's average reward outcome with a distribution of reward values. Any existing reinforcement learning algorithm can be revised with a distributional perspective, in order to achieve improved accuracy and performance (Bellemare et al., 2017b). They also argue that 'predicting the distribution over outcomes also opens up all kinds of algorithmic possibilities'. Appropriate action can be taken if the data observed is bimodal, i.e. take on two possible values. For example, using the train commute times, we can then check for train updates before leaving home to maximise the outcome. In addition to that, by modelling the distribution we can identify the safe choices when two of the choices have the same average value, by favouring the choice that varies the least. Predicting multiple outcomes has also been shown to improve the training performance of deep networks Bellemare et al. (2017b).

Bellemare et al.'s work opened up additional possibilities and may just be the beginning of this new approach. The research work can be improved further by applying a distributional perspective to other algorithms in machine learning, where a multitude of outcomes may be more beneficial than an average outcome. Improvements in performance will highlight the necessity of taking randomness into consideration when designing algorithms in general. In addition to achieving better performance in this learning environment, it would be interesting to see the research work applied to a real world problem where the impact of the improved performance can be realised.

7 Conclusion

Bellemare et al.'s paper demonstrated the importance of accounting for randomness in algorithm design and how it can be implemented in reinforcement learning algorithms. Overall this paper is of high technical quality and adds a great amount of value to machine learning and reinforcement learning. It was interesting to read, concise, well researched and presented. It is recommended that those researching reinforcement learning refer to this paper for information

on designing algorithms with a distributional perspective for better and more reliable reinforcement learning.

References

Bellemare, Marc G., Dabney, Will, and Munos, Rémi. A distributional perspective on reinforcement learning. In Precup, Doina and Teh, Yee Whye (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 449–458, International Convention Centre, Sydney, Australia, 06–11 Aug 2017a. PMLR. URL `http://proceedings.mlr.press/v70/bellemare17a.html`.

Bellemare, Marc G., Dabney, Will, and Munos, Rémi. Going beyond average for reinforcement learning, 24 July 2017b. URL `https://deepmind.com/blog/going-beyond-average-reinforcement-learning/`.

Mnih, Volodymyr, Kavukcuoglu, Koray, Silver, David, Rusu, Andrei A., Veness, Joel, Bellemare, Marc G., Graves, Alex, Riedmiller, Martin, Fidjeland, Andreas K., Ostrovski, Georg, Petersen, Stig, Beattie, Charles, Sadik, Amir, Antonoglou, Ioannis, King, Helen, Kumaran, Dharshan, Wierstra, Daan, Legg, Shane, and Hassabis, Demis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Sutton, Richard and Barto, Andrew G. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

van Hasselt, Hado, Guez, Arthur, and Silver, David. Deep reinforcement learning with double q-learning. AAAI Conference on Artificial Intelligence, pp. 2094–2100, Phoenix, Arizona. AAAI Press.