Proceeding Series of the Brazilian Society of Applied and Computational Mathematics, v. 6, n. 1, 2018.

Trabalho apresentado no XXXVII CNMAC, S.J. dos Campos - SP, 2017.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Utilização do Método Perceptron de Raio-ε Fixo para Aprendizado por Reforço

Lucas de Almeida Teixeira[1]
Saul de Castro Leite[2]
Departamento de Ciência da Computação, UFJF, Juiz de Fora, MG

1 Introdução

Problemas de aprendizado por reforço podem ser definidos como problemas nos quais um agente inteligente deve agir com base na observação do ambiente ao seu redor de modo a maximizar a soma total das recompensas imediatas que ele receberá no estado atual e nos estados subsequentes. O valor esperado desta soma de recompensas imediatas dado um estado inicial s e uma ação a é conhecido como função Q. É possível definir uma política ótima conhecendo-se a função Q ótima. Para aproximar essa função podem ser utilizadas técnicas de programação dinâmica que, usualmente, tratam o problema como um Processo de Decisão de Markov. Porém, existe o problema de representar a função Q computacionalmente.

A utilização de algoritmos de regressão para aproximar a função Q soluciona o problema de representação. Contudo, a aproximação deve ser realizada em cada iteração do algoritmo, o que pode levar a propagação de erros e consequentemente a divergência do algoritmo. Uma abordagem que vem gerando bons resultados é baseada no aprendizado de forma *off-line* e em lote, como o algoritmo *Fitted Q Iteration* [1]. Esta abordagem foi modificada em [2], em que o método de regressão foi substituído por uma rede neural, dando origem ao algoritmo *Neural Fitted Q Iteration*. Neste trabalho, é proposto a utilização do algoritmo de regressão *Perceptron* de Raio-ε Fixo [3] no papel do algoritmo de regressão no *Fitted Q Iteration*. Uma das principais vantagens desse método é que ele depende de um número menor de parâmetros em relação a outros algoritmos de regressão, e.g., as redes neurais, que dependem da escolha do número de camadas e nós adequado para ter sucesso.

2 Método Proposto

Suponha que durante uma simulação do ambiente (s_i, a_i) representa o estado do sistema e ação escolhida no tempo i, respectivamente. Métodos *Fitted Q iteration* são compostos dos seguintes passos: (a) iniciar a regressão Q_0 e fazer $k = 0$; (b) iterar os seguintes passos N vezes: (i) gerar pontos de treinamento $D = \{(x_i, y_i)\}_{i=1}^{m}$ a partir do problema de

[1]lucas.almeida1@ice.ufjf.br
[2]saul.leite@ufjf.edu.br

aprendizado por reforço, em que $x_i = (s_i, a_i)$, $y_i = r(s_i, a_i, s_{i+1}) + \gamma \min_b Q_k(s_{i+1}, b)$, γ é um fator de desconto e r é a função de recompensa do problema; (ii) fazer o treinamento de um algoritmo de regressão utilizando os dados D para gerar Q_{k+1}; (iii) fazer $k = k+1$ e voltar no passo (i). O método do *Perceptron* de Raio-ε Fixo é definido como o de minimizar a seguinte função de erro: $\sum_{i=1}^{m} \max\{0, |y_i - w \cdot x_i| - \epsilon\}$ em relação aos parâmetros w, em que ϵ é um valor fixo que define o tamanho máximo do erro permitido. A minimização é feita através do chamado gradiente estocástico. É importante ressaltar que esse método permite a utilização do chamado *kernel trick*, que possibilita regressões não lineares.

3 Experimentos

Nesta seção, os resultados comparativos entre o algoritmo proposto aqui e o algoritmo *Neural Fitted Q Iteration* são apresentados. Ambas abordagens foram testadas nos seguintes problemas clássicos de aprendizado por reforço: *Gridworld*, *Mountain Car* e *Pole Balancing*. Para a realização dos experimentos, utilizou-se uma abordagem similar a apresentada na seção de experimentos de [2]. Os dados apresentados foram obtidos ao realizar os experimentos repetidos por 5 vezes em cada um dos problemas citados acima para cada algoritmo. Um experimento consiste em $N = 250$ iterações de aprendizado e geração de novas amostras. Utilizou-se o kernel exponencial para fazer a regressão com o método do *Perceptron* de Raio-ε Fixo. A Tabela 1 mostra a média dos melhores resultados obtidos durante o aprendizado e a média do número de iterações até encontra-los.

Tabela 1: Média dos melhores resultados encontrados e do número de iterações médio.

	Gridworld		*Mountain Car*		*Pole Balancing*	
	Vitórias	Iter.	Vitórias	Iter.	Vitórias	Iter.
Rede Neural Artificial	85,46%	**30**,**4**	**99**,**94**%	125,4	32,02%	**34**,**4**
Perceptron de Raio-ε Fixo	**100**%	32,0	82,64%	**52**,**8**	**88**,**88**%	178,6

Os testes comparativos apontam que o algoritmo proposto possuí desempenho comparável ao *Neural Fitted* Q *iteration*, com o benefício de ter um número de parâmetros menor para sua configuração.

Agradecimentos

Agradecemos ao GETComp e a FAPEMIG pelo apoio durante a execução do projeto.

Referências

[1] D. Ernst, P. Geurts and L. Wehenkel. Tree-based batch mode reinforcement learning, *Journal of Machine Learning Research*, volume 6, pp. 503–556, 2005.

[2] M. Riedmiller. Neural Fitted Q Iteration – First Experiences with a Data Efficient Neural Reinforcement Learning Method, *European Conf. on Machine Learning*, pp. 317-328, 2005. DOI: 10.1007/11564096_32.

[3] R. C. S. N. P. Souza, R. F. Neto, S. C. Leite and C. C. H. Borges. Online algorithm based on support vectors for orthogonal regression, *Pattern Recognition Letters*, volume 34, pp. 1394–1404, 2013. DOI: 10.1016/j.patrec.2013.04.023.