Projeto Trainee
Reinforcement Learning

Deep Q-Networks

Grupo Turing Usp

Junho de 2020

Sumário

Capítulo 1: A Aventura

Este livro-texto é desenvolvido para ajudá-lo na sua jornada para codar o seu primeiro algoritmo de **Aprendizado por Reforço**. É esperado que, ao final dessa aventura, você seja capaz de identificar diversos conceitos dessa área e tenha consciência do funcionamento de um dos mais poderosos algoritmos da terra-média.

O Propósito

Nós, da **Guilda Turing**, desejamos aos novos membros o potencial de abordar formidáveis desventuras sem medo da adversidade! O pupilo deverá então enfrentar, mesmo que com o suporte do restante da guilda, o mais difícil passo: aquele que marca o início de sua jornada. É esse primeiro passo no abismo de RL que será incorporado neste lendário documento.

O Problema

O problema a ser encarado não será um oponente simples, será desejado dos novos membros a capacidade de resolver um ambiente ancião: O jogo ***Pong***. Lógico, nenhum auxílio será poupado para os que o buscarem, afinal, somos um grupo unido! E dedicaremos esse pergaminho de instruções para preparar os novatos para esse formidável confronto.

Opções

Apesar do problema que recomendamos que os novatos enfrentem seja Pong, se for do interesse do novo membro enfrentar outros oponentes, recomenda-se o ambiente *Cartpole*.

Aprofundamento no Ambiente

Prerequisito: ambição superior a 15

O jogo clássico do Atari: ***Pong***, foi o primeiro videogame lucrativo da história. Foi criado por Nolan Bushnell e Ted Dabney na forma de um console ligado a um monitor, movido a moedas. É um jogo eletrônico de duas dimensões que simula o tênis, onde um oponente tenta pontuar sobre o outro ao fazer a bola atravessar a linha de sua "raquete". Pong era visto pela máquina do atari como somente 128 bytes, uma quantidade ínfima para 2020 - esse arquivo é mais carregado do que isso. Com alguns cálculos, é possível ver que a RAM do atari poderia enxergar um total de 256^{128} estados diferentes.

Nesse projeto, a tarefa será uma versão simplificada desse problema. Ao invés de trabalhar diretamente com a RAM, o membro trabalhará com uma lista reduzida de variáveis de estado. No modo fácil são dois valores: as distâncias horizontal e vertical entre o jogador e a bola. Já no modo normal, são quatro: as coordenadas x e y da barra e da bola. A partir desses valores, é necessário determinar em qual direção o jogador deve mover a barra.

O leitor astuto terá percebido que o número total de estados ficou na verdade ainda maior com essa simplificação, visto que as variáveis de estado são números reais e, assim, podem assumir uma quase infinidade de valores. No entanto, esses valores já se encontram numa forma muito mais propícia aos feitiços; "ambição superior a 15" não é suficiente para a manipulação direta de memória RAM.

Para melhor compreensão

Todas as ideias apresentadas até aqui já foram aprofundadas no *Diário da Guilda Turing*.

Os conceitos e a terminologia básicas de RL foram introduzidos em [2].

O ambiente do Pong funciona como os outros ambientes do Gym, que já foram descrito em [5]. Adicionalmente, um ambiente semelhante ao usado aqui foi utilizado em [6].

Mais especificamente, o jogador controla a barra da esquerda, os estados são conforme descrito acima, as ações possíveis são:

0=parado, 1=baixo, 2=cima,

e as recompensas são:

- $+500$ quando o jogador faz um ponto,
- -500 quando o oponente faz um ponto,
- $+2000$ quando o jogador ganha,
- -2000 quando o oponente ganha.

O jogo funciona num esquema de "melhor de 7", ou seja, ganha o primeiro jogador que fizer quatro pontos. No entanto, a magia que sustenta o jogo é escassa; caso o jogo se alongue demais, a magia pode se exaurir, acabando com o jogo. Nesse caso, considera-se que o jogador demorou de mais para pontuar, de forma que ele não merece nenhuma recompensa.

A ambição do programador

Item dourado, raro

Um buff que o portador deste documento adquire, concedendo a ele +5 de coragem, *status* que será crucial para a conclusão dessa tarefa.

Os efeitos da ambição são reativados toda alvorada.

Regiões do Mapa

.

1. Taverna de RL

No Discord, direcionado às dúvidas gerais.

2. Pombo Correio dos Membros da Guilda

O lugar onde dúvidas não perduram.

- (61) 98333-6261 — BerBardo
- (19) 99850-2800 — MatsuMonstro
- (11) 94331-1181 — Ariel, o Tritão
- (11) 97114-3359 — Pedro

3. Stack Overflow

Taverna onde serão encontrados muitos dos membros, possui interessante efeito sobre programadores: quanto mais tempo é gasto nessa lendária taverna, maior a tentação de voltar.

Capítulo 2: O Algoritmo

A arma que os desejosos de ascensão na guilda deverão usar não é para os de coração fraco, será utilizado o lendário algoritmo: ***Deep Q-Learning***.

> Conforme lês tais palavras, sentes na tua pele o sangue e o suor que a lâmina e a bainha que tal ferramenta porta, tu te sentes reconfortado, pela esperança — nunca antes maior — da vitória.

Q-learning

A Arma que será utilizada, *Deep Q-Learning*, é uma derivação direta de outra mais antiga: *Q-learning*. Como as outras armas do arsenal de Aprendizado por Reforço, essa permite ao usuário reviver eventos até que esteja otimizado para a batalha.

Q-learning busca encontrar o valor de cada ação que pode ser tomada para cada estado, de forma a saber qual é o melhor golpe para cada situação. Ela foi revolucionária dentro da área por associar a habilidade de aprender *durante a batalha, e não somente ao seu fim* e por possuir duas vozes dissociadas que auxiliam o guerreiro: uma que dita suas ações e outra que dita o valor de cada uma delas, permitindo uma exploração para a melhora do combate.

Para melhor compreensão

Os dois conceitos que de forma jocosa aqui foram introduzidos são conhecidos como *Temporal Difference* e *Off-policy*, é sugerido, dado interesse do leitor, aprofundamento através do texto sobre Q-learning do *Diário da Guilda Turing* [6].

Entretanto, Q-learning apresenta um sério desfalque, o de que ele não é muito capacitado contra ambientes portadores de muitas ações e estados. Comumente, são armazenados numa estrutura de dados os valores estimados para cada ação em cada estado; essa seria a nossa "função"que relaciona um valor para cada par estado-ação. Mas, como foi dito anteriormente, se a recordação é boa, o ambiente do Pong (simplificado) uma quantidade imensa de estados diferentes e, portanto, uma quantidade ainda maior de pares estado-ação.

Deep-Q-Learning

Redes Neurais

As ***Redes Neurais*** são uma magia de altíssima flexibilidade disponíveis no arsenal do invocador *(pesquisador de inteligência artificial)*; ela reganhou força em torno de 2014 e não perdeu seu momento até então. Ela é, sem erros, conhecida por ser uma ótima *aproximadora de funções universal* e é a principal diferença de um algoritmo com *Q-learning* e um com *Deep-Q-Learning*.

Deep-Q-Learning

O nosso algoritmo, então, buscará as forças daquele de onde é derivado e as ampliará com o seu potencial de generalização. Ele deverá substituir a tabela, aproximando os valores de cada estado-ação através de padrões reconhecidos pelo seu extenso treinamento. Sendo, portanto, capaz de resolver Pong.

Capítulo 3: A trajetória

> Com a chegada do último capítulo, teu coração te pesa, sabes que a batalha há de chegar, mas a preparação recebida te conforta.

Início

Para o começo da batalha, é recomendado que o guerreiro busque por novas informações, explore as referencias fornecidas e busque entender a teoria ao redor do oponente. Será necessário um plano de guerra, mas a Guilda Turing está sempre ao lado daqueles que buscam glória.

Primeira Tarefa

Depois de adquirir alguns conhecimentos simples no caminho para a mestria do Aprendizado por Reforço, o aprendiz estará pronto para a ***Primeira Tarefa***.

A ***Primeira Tarefa*** consiste em escrever a magia `rodar_ambiente()` no pergaminho mágico `Exploracao.py`, que se encontra na biblioteca de pergaminhos disponibilizada no início desta jornada (o Github).

Esta magia deve conjurar o ambiente escolhido e rodar um episódio completo, além de utilizar o feitiço `env.action_space.sample()` do ambiente `env` para selecionar uma ação aleatória dentre as possíveis desse ambiente.

Caso o noviço não esteja familiarizado em escrever esse tipo de magia de conjuração, ele não deve se preocupar. O pergaminho mágico já contém as instruções necessárias para escrever uma magia básica de conjuração, e é uma tradição secular da guilda conjuradores experientes se prontificarem para ajudar jovens aprendizes.

Meio

Durante a batalha, o discípulo deverá se concentrar no problema a frente, é esperado que já tenha um bom conhecimento do problema escolhido e da arma empunhada. Ele não deve hesitar em demandar por ajuda aos companheiros de guilda e recorrer aos históricos de prévias batalhas.

> **Sério mesmo, turma**
>
> Peçam ajuda e leiam exemplos! Inicialmente, é essa a forma de se aprender.

Segunda Tarefa

Uma vez que a ***Primeira Tarefa*** foi completada, o guerreiro estará pronto para seu segundo, e mais complexo, desafio: a ***Segunda Tarefa***.

Neste confronto, o aprendiz deparará com seu maior obstáculo até agora, conquistar o tão temido ambiente de ***Pong***! E, para alcançar tal feito, ele deverá utilizar de todas as armas obtidas durante sua jornada.

Primeiramente, o discípulo deverá abrir seu segundo pergaminho mágico, `DQN.py`. Nele, a potente invocação da DQN será escrita, bem como a sua interação com o ambiente conjurado.

Esta tarefa não será nada fácil, qualquer guerreiro precisará de estudar os velhos pergaminhos da Guilda para atacar este problema. Portanto, recomendamos a leitura dos textos “Criando uma IA que aprende a jogar Pong” [6] e “Pouse um módulo lunar com DQN” [7], do *Diário da Guilda Turing*.

Por fim, lembre-se sempre de que nenhum aprendiz precisa seguir sua jornada sozinho! É, inclusive, recomendado a ele que recorra aos meios apontados no mapa, e procure a sabedoria dos mestres da guilda!

Desafio (Opcional)

Tendo conquistado o Pong, o discípulo se dá conta que o ambiente esteve desde o início no modo *fácil*. Os discípulos mais ousados podem, então, terminar sua jornada dominando a dificuldade *normal* do jogo. Para tanto, será necessário desabilitar o feitiço `easy_mode` presente no mesmo pergaminho mágico da tarefa anterior.

Fim

Nos finalmentes da batalha, é necessário lembrar que os oponentes de RL são conhecidos por estender a sua morte, é normal uma rede neural demorar para treinar, mas não desanime, algumas horas devem bastar!

Bibliografia

Material Interno

Slides e notebooks de RL [1]

Grupo Turing. *Slides e notebooks de RL*. URL: https://drive.google.com/drive/folders/1HhcD_yxAHfdDQipjV-1sPlZHSCW140Ze.

Diário da Guilda Turing

RL #1 — Introdução [2]

Enzo Cardeal Neves. "Aprendizado por Reforço #1 — Introdução". Em: *Turing Talks* (2020). URL: https://medium.com/turing-talks/aprendizado-por-refor%C3%A7o-1-introdu%C3%A7%C3%A3o-7382ebb641ab.

RL #2 — MDP (Parte 1) [3]

William Fukushima. "Aprendizado por Reforço #2 | Processo de Decisão de Markov (Parte 1)". Em: *Turing Talks* (2020). URL: https://medium.com/turing-talks/aprendizado-por-refor%C3%A7o-2-processo-de-decis%C3%A3o-de-markov-mdp-parte-1-84e69e05f007.

RL #3 — MDP (Parte 2 [4]

Gustavo Corrêa. "Aprendizado por Reforço #3 — Processo de Decisão de Markov (Parte 2)". Em: *Turing Talks* (2020). URL: https://medium.com/turing-talks/aprendizado-por-refor%C3%A7o-3-processo-de-decis%C3%A3o-de-markov-parte-2-15fe4e2a4950.

RL #4 — Gym [5]

Enzo Cardeal Neves. "Aprendizado por Reforço #4 — Gym". Em: *Turing Talks* (2020). URL: https://medium.com/turing-talks/aprendizado-por-refor%C3%A7o-4-gym-d18ac1280628.

Criando uma IA que aprende a jogar Pong [6]

Ariel Guerreiro. "Criando uma IA que aprende a jogar Pong". Em: *Turing Talks* (2020). URL: https://medium.com/turing-talks/criando-uma-ia-que-aprende-a-jogar-pong-f379b0170017.

Pouse um módulo lunar com DQN [7]

William Fukushima. "Pouse um módulo lunar com Deep Q-learning". Em: *Turing Talks* (2020). URL: https://medium.com/turing-talks/pouse-um-m%C3%B3dulo-lunar-com-deep-q-learning-1f4395ea764.

Outros Materiais

Reinforcement Learning Book [8]

Richard S. Sutton e Andrew G. Barto. *Reinforcement Learning. An Introduction*. Ed. por The MIT Press. 2ª ed. 2018. ISBN: 978-0-262-19398-6. URL: https://archive.org/details/rlbook2018.

Gym Documentation [9]

Gym. *Gym Documentation*. OpenAI. URL: https://gym.openai.com/docs/.