### Sobre o Curso

#### Semanas

- 1 Aula por semana
  - ~1h30min
  - Presença: Quem estiver na chamada
- 1 Laboratório por semana\*
  - 2 semanas para entregar

Slides e Labs estarão disponíveis no Github

#### Entregando Laboratórios

Basta subir o código/notebook no github e enviar pra gente.

 De preferência use um único repositório para o curso todo.



#### Cronograma

- Aula 0: ML e Imitation Learning
- Aula 1: Conceitos e Modelagem (RL)\*
- Aula 2: Processos de Decisão de Markov\*
- Aula 3: Métodos de Diferença Temporal e *Deep* RL
- Aula 4: Métodos de Gradiente de Política
- Aula 5: Métodos de Actor-Critic
- Aula 6: Offline-RL
- Aula 7: Tópicos Avançados\*

#### Dúvidas e Discussões





## Aprendizado por Reforço

AULA - 0

Aprendizado de Máquina e Imitation Learning

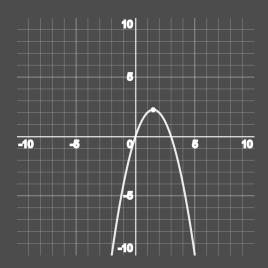
#### ROTEIRO

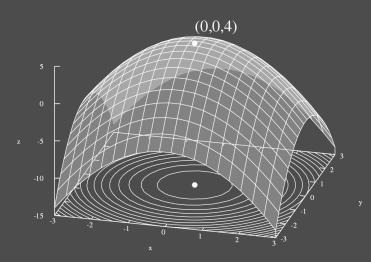
- Machine Learning
  - o Otimização
  - o Tipos de Aprendizado
  - Aprendizado Supervisionado
- Redes Neurais Artificiais
  - Visão Geral
- Imitation Learning
  - Como Funciona
  - Se Funciona
- PyTorch

## Machine Learning

#### Otimização

- O que é otimizar?
  - o Encontrar a melhor solução
- Maximizando ou Minimizando uma função



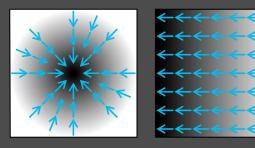


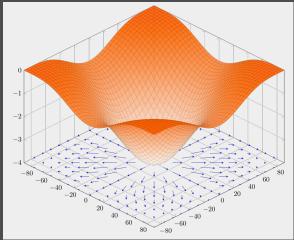
#### Gradiente

Direção do maior aumento em uma função



$$\nabla f(x, y, z) = \frac{\partial f}{\partial x} \mathbf{i} + \frac{\partial f}{\partial y} \mathbf{j} + \frac{\partial f}{\partial z} \mathbf{k}$$





#### O que é Aprendizado de Máquina?

- Otimizar funções quando não se sabe o formato
- Aprendendo Padrões

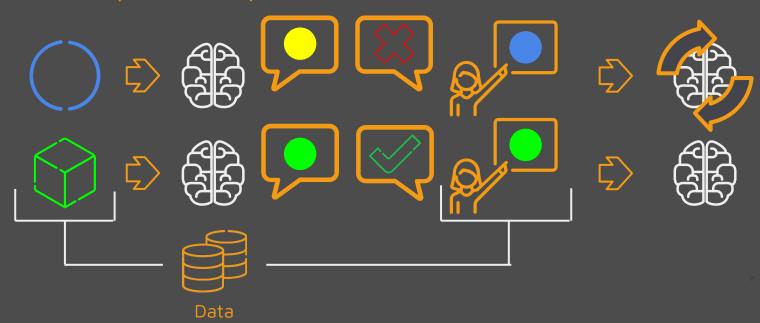
- O aprendizado de máquina permite sistemas aprender e melhorar sem programação explícita de comportamentos
- Imitando o Cérebro

#### Tipos de Aprendizado

#### Não Supervisionado: Supervisionado: Aprender padrões a Encontrar padrões partir de exemplos sem exemplos por Reforço: Aprender a partir de interação e experiência

#### Aprendizado Supervisionado

Com exemplos de comportamento correto



#### Depois do Treino

- Generalizar a partir de padrões "aprendidos"
- Reconhece os mesmos padrões em outras situações



#### O que está sendo otimizado?

- Função de Perda (*Loss Function*)
- Erro Quadrático Médio

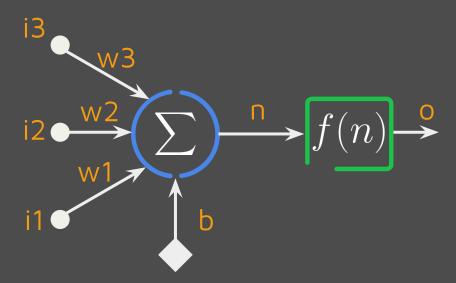
$$ext{MSE} = rac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y_i})^2.$$

- Diferença entre a resposta correta e a resposta do modelo
- Minimização (contra o gradiente)

# Redes Neurais Artificiais

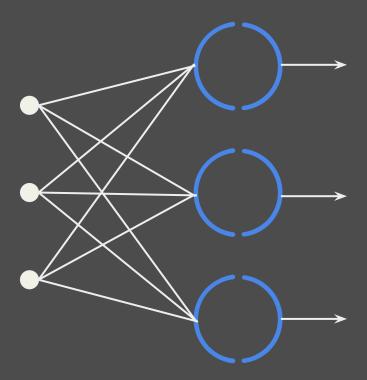
#### Neurônio Artificial

- Sinais de entrada são ponderados
- Somatória é passada por uma função
- Resultado é o sinal de saída



#### Camadas de Neurônios

- Vários neurônios juntos
- Entradas iguais
- Pesos diferentes
- Cada neurônio tem um sinal de saída

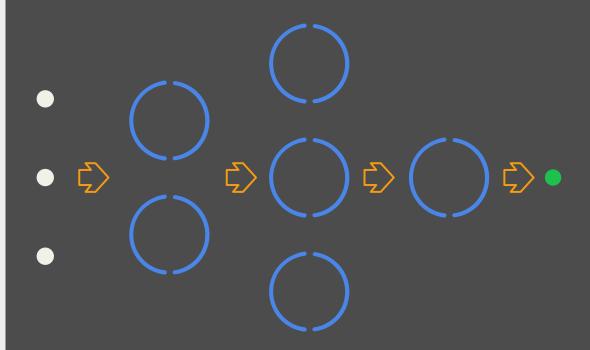


#### Rede Neural

- Camadas Sequenciais
- Entrada de um camada é a saída da anterior

#### Pesquisar em casa:

- Camadas Convolucionais
- Camadas Recorrentes



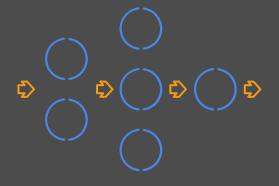
# Imitation Learning

#### Aprendendo por Imitação

Imagine que estamos tentando fazer um carro autônomo



situação observação





#### Aprendendo por Imitação

 Podemos gravar uma pessoa dirigindo e guardar as situações e ações que ela toma



#### **Behavior Cloning**

- Adquirir dataset de observações e ações do expert
- Treinar o modelo com aprendizado supervisionado, minimizando a função de perda

$$Data: \{(s_t, a_t^*), (s_{t+1}, a_{t+1}^*), (s_{t+2}, a_{t+2}^*)...\}$$

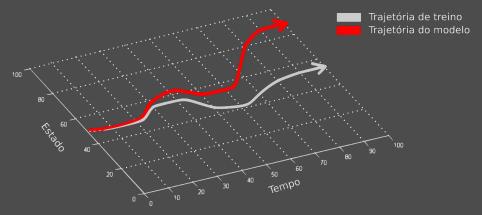
$$|L = -log(P(a^*|s))|$$

#### É diferente do Supervisionado?

- Sim, e não...
- Ambos otimizam a partir de exemplos
- Ambos precisam de um "expert" para anotar dados
- Problema sequencial
  - Não afeta a forma de treinar
  - Dificuldades específicas de problemas sequenciais

#### Funciona?

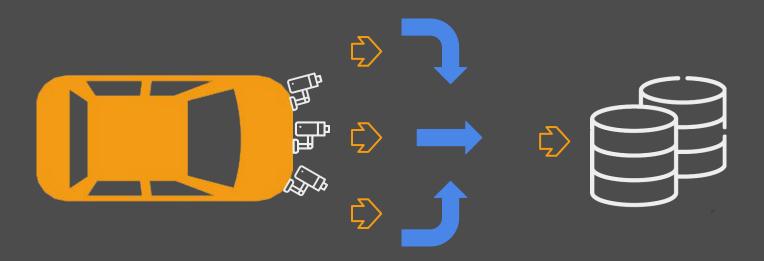
- Não tão fácil
- Dados mostram um bom comportamento
- Erros acumulam para situações muito diferentes



• É preciso ter REPRESENTATIVIDADE

#### NVIDIA

- Câmeras levemente laterais (extra)
- Ações com leves alterações p/ essas câmeras (mais dados)





#### Representatividade nos Dados

- Pegar mais dados diferentes do seu "expert"
  - o Situações onde ele normalmente não entraria
- Truques
  - NVIDIA
  - Câmeras levemente laterais (extra)
  - Ações com leves alterações para essas câmeras (mais dados)
- Adicionar dados de comportamento do próprio modelo ao dataset
  - DAgger
  - Colocar o modelo no ambiente e coletar dados
  - Humano/Expert categoriza os dados com as melhores ações



DAgger here reacts dynamically to an untrained obstacle

