

Relatório LAMIA - Kaíque Medeiros Lima

1 - Vídeo: O Que é Aprendizado de Máquina (I)

Redes Neurais e Machine Learning | Nerdologia Tech

Introdução:

As redes neurais são como vários nodos respondendo sim ou não, alguns com mais peso que outros, com o tempo, esses nodos aprendem entre si e se calibram, treinando com os dados recebidos.

Teorema de Aproximação Universal:

Uma rede com uma camada pode codificar as melhores decisões para resolver o problema, mas pode ser muito grande precisando de muito treino.

Otimizar Redes Neurais:

Rede profunda: Organização dos nodos em várias camadas, assim “peneirando” os dados.

Um jeito de refinar as decisões é utilizar os erros como base para correção de erros. Algoritmo de Backpropagation.

Especialistas: Avaliam apenas alguns tipos de itens, os quais são mais treinados no assunto, podendo até pular camadas.

Introdução ao Machine Learning (ML de Zero a 100, parte 1)

Linguagem: Python

Reconhecimento de mãos no jokenpô: Pode parecer fácil pelo olho humano, mas um computador fazer isso é outra história.

Programação tradicional: Regras + dados = resposta.

Machine Learning: Respostas + dados = regras.

$$x = -1, 0, 1, 2, 3, 4$$

$$y = -3, -1, 1, 3, 5, 7$$

$$y = 2x - 1$$

Código de reconhecimento de padrão de números do vetor:

```
# define o modelo, camada única, único neurônio: units = 1
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1])])

# calcula a qualidade da suposição usando a função de perda, se calibrando
model.compile(optimizer='sgd', loss='mean_squared_error')

xs = np.array([-1.0, 0.0, 1.0, 2.0, 3.0, 4.0], dtype = float)
ys = np.array([-3.0, -1.0, 1.0, 3.0, 5.0, 7.0]), dtype = float)

# fará isso 500, treinando nas 500 vezes, encaixe x em y e tente 500 vezes
model.fit(xs, ys, epochs = 500)

print(model.predict([10.0]))
```

a