

Instituto Federal do Sudeste de Minas Gerais, Campus Barbacena

# Deep Learning para Previsão de Séries Temporais com Keras

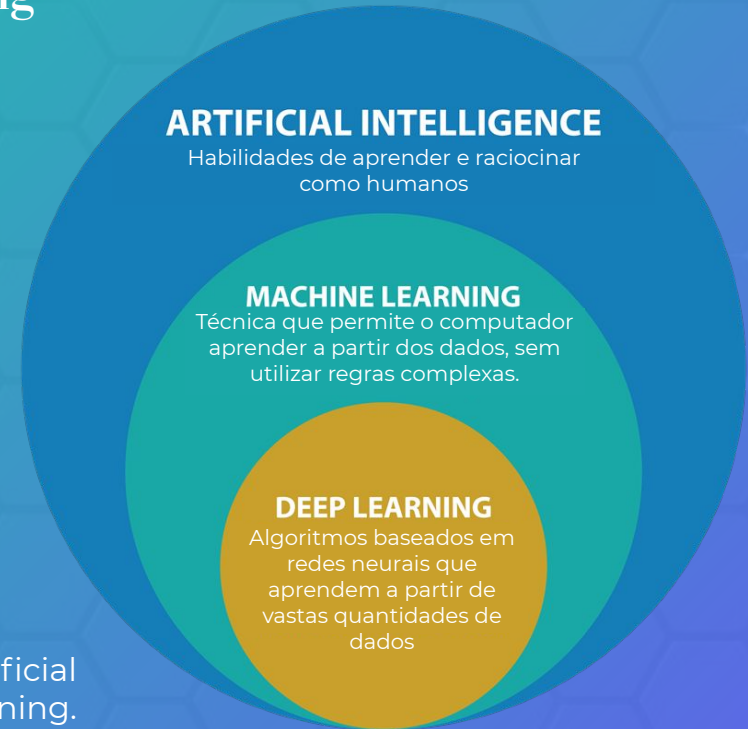


Guilherme Domith Ribeiro Coelho  
guilhermedomith@gmail.com

## Deep Learning

Algoritmos que utilizam **redes neurais profundas** e dependem de **muitos dados** para extrair características no seu treinamento. Este aprendizado é utilizado na tomada de decisões futuras.

Fig 1: Hierarquia das áreas de estudo Artificial Intelligence, Machine Learning e Deep Learning.



## Deep Learning

O que contribuiu para o deep learning ser popularizado?

- Maior disponibilidade de GPUs;
- Um grande volume de dados na internet a ser analisado;
- Problemas não linearmente separáveis;
- Ótimos resultados em reconhecimento de imagem.

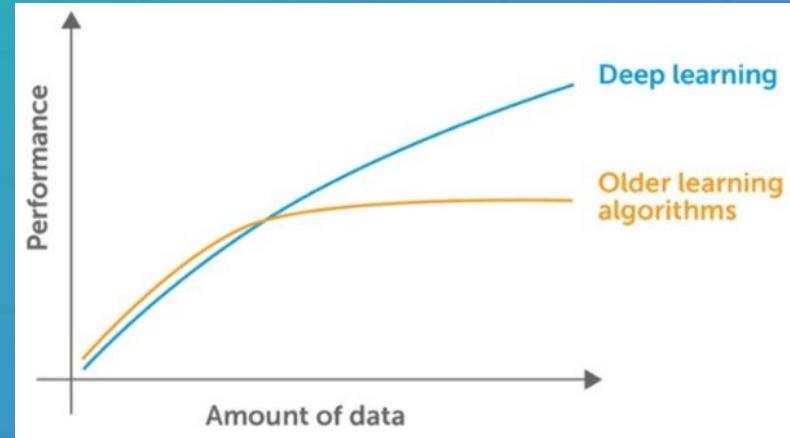


Fig 2: Comparação de Deep Learning com outros algoritmos em relação a quantidade de dados.

## Deep Learning

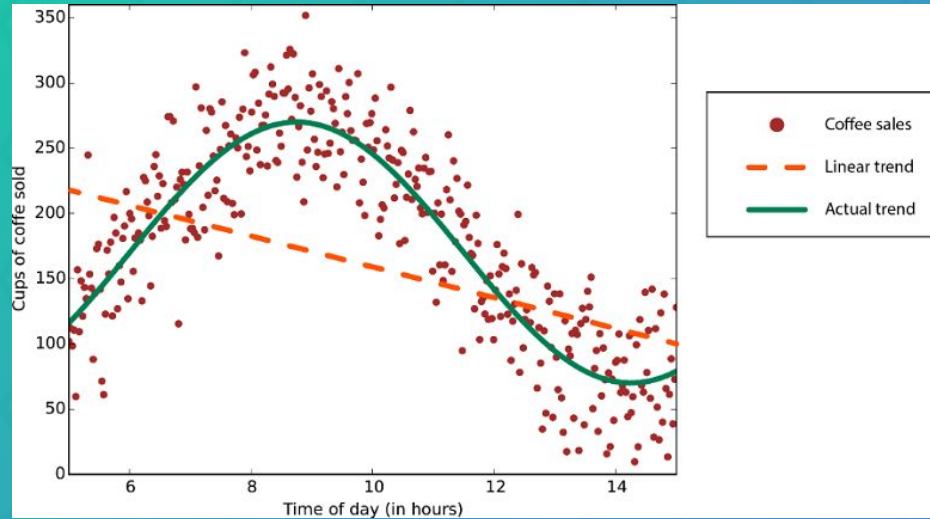


Fig 3: Exemplo de dados não linearmente separáveis.



## Deep Learning

Áreas em que o deep learning está sendo aplicado:

**Visão computacional** - Reconhecimento de objetos, segmentação semântica, descrição de cenário. Ex: Carros Autônomos;

**Reconhecimento da fala e PLN** - Ex. Cortana, Siri, Google Assistant e Google Tradutor;

**Healthcare** - Avanços na área da saúde, especificamente no diagnóstico por imagens. Ex: Detecção de tumores



6



## Deep Learning

Outras aplicações:

(Problemas também resolvidos com Machine Learning)

**Sistemas de Recomendação** - Ex: Netflix, Amazon e Spotify;

**Detecção de fraudes** - Ex: Nubank;

**Análise de sentimento** - Em vídeos, textos ou imagens. Ex:  
Monitoramento de marca em redes sociais

## Redes Neurais

**Modelo matemático** inspirado na estrutura neural de organismos inteligentes.

Uma grande rede neural artificial pode ter centenas ou milhares de **unidades de processamento**; já o cérebro de um mamífero pode ter muitos bilhões de neurônios.

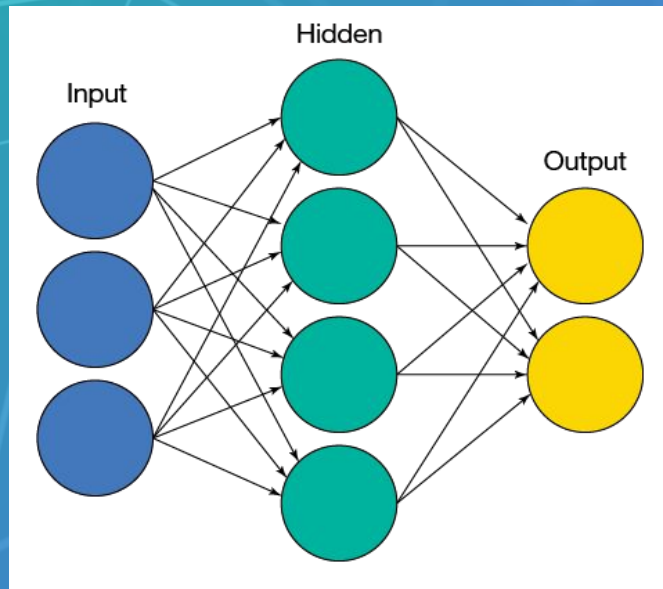


Fig 4: Rede neural com uma camada oculta de quatro neurônios.

## Redes Neurais

Matematicamente, o neurônio pode ser modelado como um somatório das entradas **multiplicado por pesos**, em que esse valor passa por uma **função de ativação**.



Fig 4: Modelo de um neurônio com os pesos e transformações a serem realizados sobre os dados de entrada, gerando uma saída.



## Redes Neurais

Alguns dos tipos de redes neurais:

**Convolutacional** - Usado especificamente na área de visão computacional (processamento de imagens e vídeos). Ex.: Reconhecimento Facial, Carros autônomos, e etc.

**Recorrente** - Usado em problemas que utilizam **dados sequenciais** (Séries temporais). Ex.: Prever o que uma pessoa irá fazer após uma determinada tarefa; Próxima palavra em um texto.

## Rede Neural Recorrente

Na Rede Neural Recorrente a camada oculta é conectada a ela mesma, passando uma resposta para frente e também se realimentando.

Sendo assim, além do que é aprendido em treinamento estas redes também levam em consideração os acontecimentos recentes.

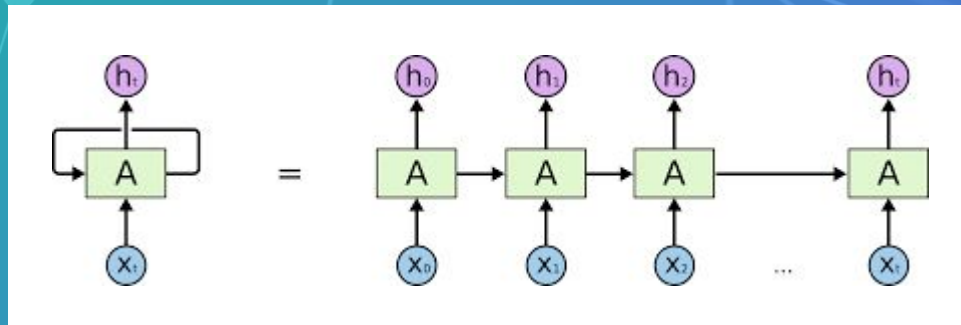


Fig 5: Exemplo de como o neurônio de uma RNN se comporta, analisando as demais iterações da rede neural.

11  
—x—  
Keras

**Keras** é uma **API de redes neurais** em Python, capaz de rodar em cima das bibliotecas de tensores TensorFlow, CNTK ou Theano. Ela provê uma estrutura que permite compilar redes neurais combinando camadas de diferentes dimensões e funções de ativação. Por padrão, utiliza **TensorFlow**.



theano

12

×

## Instalação

**Anaconda** (<https://www.anaconda.com/distribution/>)



matplotlib

NumPy

pandas  
 $y_i = \beta' x_i + \mu_i + \epsilon_i$



**Keras e TensorFlow:**

```
$ pip install keras
```

```
$ pip install tensorflow==2.0.0-alpha0
```

13  
—x—  
layers

Existe diferentes tipos de layers, mas as que utilizaremos são:

- **Dense:** Cada neurônio é ligado a todos os neurônios subjacentes;
- **Dropout:** Neurônios serão excluídos aleatoriamente no processo de treinamento para evitar overfitting;
- **LSTM:** Cada neurônio é ligado a todos os neurônios subjacentes e a ele mesmo. É utilizado para Redes Neurais Recorrentes (RNN).



## Optimizers

O **optimizer** é um dos parâmetros necessários para compilar o modelo do keras. Nele, deve-se informar o algoritmo que será utilizado para definir **como os pesos da rede neural são atualizados**.

Existem vários algoritmos optimizers. Porém, dois deles são mais utilizados, normalmente alcançam melhores resultados:

- **Adam**
- **RMSprop** (Recomendado para redes neurais recorrentes).

## Losses

Função que calcula a diferença entre os dados de teste e os dados de validação. Algumas das loss functions disponíveis:

- **Classificação Binária:** *binary\_crossentropy*, *hinge* e *squared\_hinge*;
- **Classificação Multiclasses:** *categorical\_crossentropy*, *sparse\_categorical\_crossentropy* e *kullback\_leibler\_divergence*;
- **Regressão:** *mean\_absolute\_error*, *mean\_squared\_error* e *mean\_squared\_logarithmic\_error*.

## Metrics

As **métricas** são utilizadas para fazer a **avaliação do modelo** durante o treinamento e para as previsões. Com ela é possível saber a acurácia do modelo, ou seja **medir a precisão**.

As métricas são opcionais para o Keras, mas podem ser informadas no momento de compilação do modelo.

## Funções de Ativação

Algumas das mais conhecidas funções de ativação:

- **Sigmoid:** Retorna valor entre 0 e 1.
- **Tahn:** Retorna valor entre -1 e 1.
- **Relu:** Retorna valor  $\geq 0$
- **Linear:** Retorna o próprio valor da soma
- **Softmax:** Retorna valores entre 0 e 1 (Classificação Multiclasses).

## Referências

Keras: The Python Deep Learning library. Disponível em:  
<[www.keras.io](http://www.keras.io)>

Uso De Redes Neurais Recorrentes Para Previsão De Séries  
Temporais Financeiras. Disponível em:  
<[www.dcc.ufmg.br/pos/cursos/defesas/1999M.PDF](http://www.dcc.ufmg.br/pos/cursos/defesas/1999M.PDF)>





## Referências

Deep Learning: do Conceito às Aplicações. Disponível em:  
<[www.medium.com/data-hackers/deep-learning-do-conceito-%C3%A0s-aplica%C3%A7%C3%B5es-e8e91a7c7eaf](http://www.medium.com/data-hackers/deep-learning-do-conceito-%C3%A0s-aplica%C3%A7%C3%B5es-e8e91a7c7eaf)>

Inteligência Artificial, Machine Learning e Deep Learning.  
Disponível em:  
<[www.medium.com/data-science-brigade/a-diferen%C3%A7a-e-entre-intelig%C3%Aancia-artificial-machine-learning-e-deep-learning-930b5cc2aa42](http://www.medium.com/data-science-brigade/a-diferen%C3%A7a-e-entre-intelig%C3%Aancia-artificial-machine-learning-e-deep-learning-930b5cc2aa42)>