

Aprendizado de máquina

Uma visão geral

André Pacheco
agcpacheco@inf.ufes.br

01 de março de 2019



- 1. Apresentação**
- 2. Introdução**
- 3. Conceitos básicos**
- 4. Deep learning**
- 5. Frameworks**



Apresentação

- **Nome:** André Pacheco
 - Graduado em Engenharia de Computação - UFES
 - Mestre em ciência da Computação - UFES
 - Doutorando em Ciência da Computação - UFES
- **Grupo de pesquisa:** Laboratório de Computação e Engenharia Inspirado na Natureza - LABCIN
- **Foco atual**
 - Reconhecimento de padrões de imagens médicas
- **Áreas de interesse**
 - Machine Learning
 - Deep Learning
 - Computação Natural



- Para aqueles que se interessarem pelo tema:
 - Eu tenho um blog sobre o assunto
 - <http://computacaointeligente.com.br>

Computação Inteligente

Sobre Algoritmos Tutoriais Artigos Cool stuffs Q.

Computação Inteligente

Máquinas aprendendo a solucionar problemas complexos
Conceitos, aplicações e desenvolvimento

[Blog](#)

 Imagem: Freepik

Difundir Conhecimento

A intenção principal deste blog é compartilhar conhecimento. Por isso, sempre utilizaremos um linguajar simples e direto em todos os nossos posts. Para mais informações acesse a sessão [sobre](#) do blog.

 Imagem: Freepik

Inteligência artificial

O tema principal deste blog é a inteligência artificial. Portanto, você encontrará diversos tutoriais, algoritmos, código e cíprijões em nossos posts. Abordamos tanto temas clássicos quanto os que estão no estudo da arte.

 Imagem: Freepik

Como citar o blog?

Está fazendo um trabalho de conclusão de curso e pretende citar um de nossos posts? Veja nossa página sobre [citacao](#). Em resumo, temos um DOI que você pode utilizar para nos citar.

Apresentação

Escopo



- Apresentar de maneira geral a área de aprendizado de máquina
- Discutir os principais tipos de problemas e algoritmos
- Discutir um pouco sobre redes neurais artificiais e o seu aprendizado
- Apresentar ferramentas para desenvolvimento



“A Ciência da Computação tem tanto a ver com o computador como a Astronomia com o telescópio, a Biologia com o microscópio, ou a Química com os tubos de ensaio. A Ciência não estuda ferramentas, mas o que fazemos e o que descobrimos com elas”

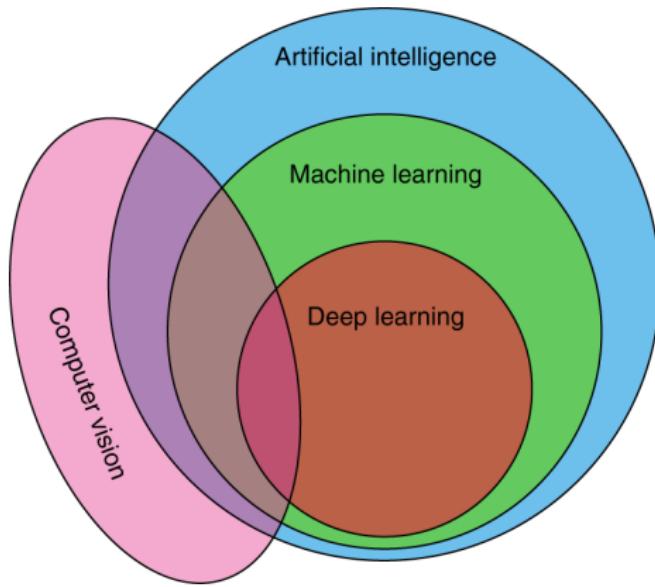
- Edsger Dijkstra



- Essa área é fortemente baseado em:
 - ▣ Algebra linear
 - ▣ Cálculo
 - ▣ Estatística
- Isso não quer dizer que essa apresentação dependa fortemente disso
 - ▣ É apenas um aviso para quem tiver interesse em se aprofundar na área
- Mas espera ai, que área?



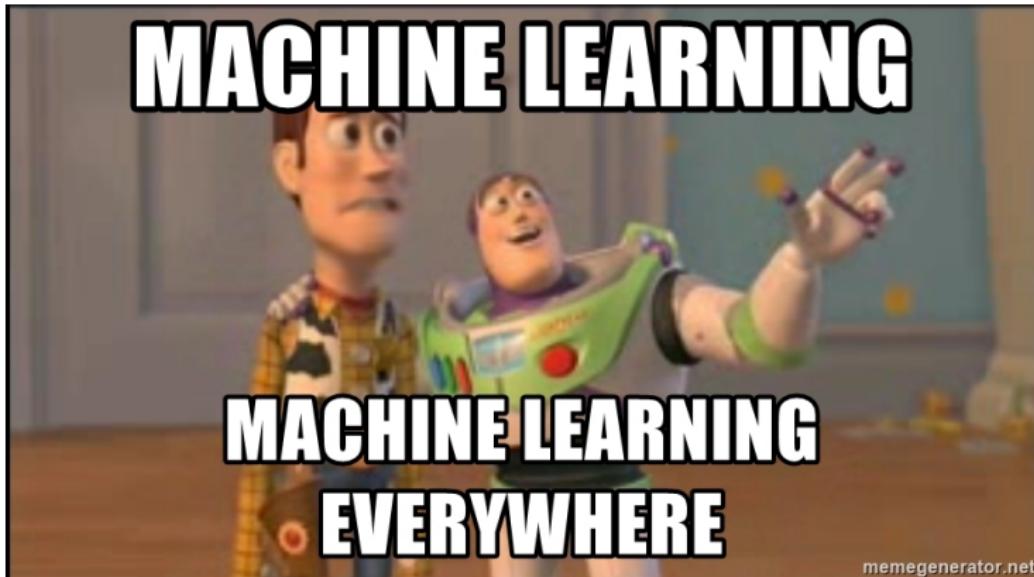
- Inteligência artificial, machine learning, deep learning, visão computacional etc etc etc



O que é o aprendizado de máquina?



- É uma área da computação que utiliza algoritmos que possuem as seguintes características:
 - Se baseia em modelos estatísticos
 - Executa uma tarefa específica sem instruções explícitas
 - Se baseia em padrões e inferência
- Além disso, os algoritmos são **treinados a partir de um conjunto de amostras**
 - Ter dado é essencial
 - Por isso **não são explicitamente programados**
- Exemplos:
 - **Rede neural artificial**, máquina de aprendizado extremo, K-vizinhos mais próximos, Naive Bayes, *Support Vector Machine*, Árvore de decisão etc.





- Basicamente são dois* tipos:
 - Supervisionado
 - Não supervisionado
- Ambos são **baseados em dados**
 - Conjuntos de treino, validação e teste
- Ambos são solucionados com algoritmos de aprendizado de máquina
 - Muda a forma de abordar o problema



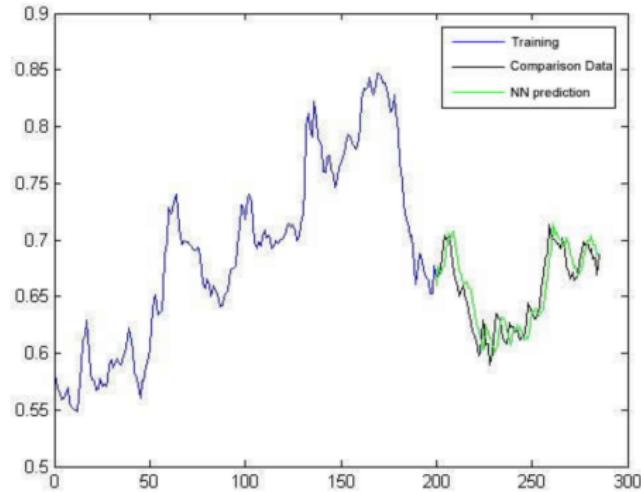
- Principais características:
 - É necessário um conjunto verdade
 - As amostras precisam de **conhecimento prévio**
 - Amostras rotuladas
 - Amostras possuem **entradas (x) e saídas(y)**
- **Principal objetivo:** encontrar uma função $f(x) = y$ que aproxime ao máximo a resposta do problema
 - Em outras palavras: dado x o algoritmo deve responder y

Conceitos básicos

Problemas supervisionados



- Dois problemas principais:
 - Classificação da dados
 - Predição de séries temporais
- Exemplo de série temporal:





- Classificar dados é algo tão comum que fazemos isso desde criança
 - Separar brinquedos por cores, por exemplo
- Abrange diversas áreas do conhecimento
 - Biologia, medicina, entretenimento, financeira etc.
- Exemplos não faltam:
 - Um médico quer determinar se um tumor é maligno ou benigno
 - Um empresa quer classificar tipos de vinho
 - Um banco quer determinar se um cliente deve ou não receber um tipo de financiamento
- Por ter tanta aplicação, é um dos tópicos mais ativos da área

Conceitos básicos

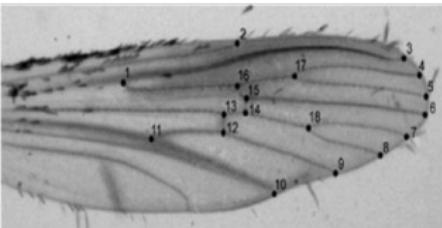
Classificação de dados



- Agora pense: por que o problema de classificação ocorre?

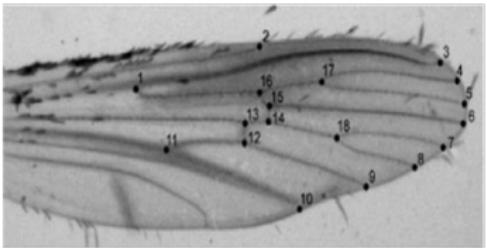


- Agora pense: por que o problema de classificação ocorre?
 - Por que não sabemos a priori qual rótulo um determinado "objeto" deve receber
 - Por exemplo: ao visualizar uma lesão na pele um dermatologista não sabe qual diagnóstico dar. **Ele precisa inferir!**
- Agora imagine o seguinte problema: como classificar insetos de acordo com sua asa?





- Ok, mas e se esse problema crescer um "pouquinho"?



-0.4933	0.0130
-0.0777	0.0832
0.2231	0.0861
0.2641	0.0462
0.2645	0.0261
0.2471	0.0003
0.2311	-0.0228
0.2040	-0.0452
0.1282	-0.0742
0.0424	-0.0966
-0.0674	-0.1108
-0.4102	-0.0163
-0.3140	0.0318
-0.1768	0.0341
0.0715	0.0509
-0.0540	0.0238
0.0575	-0.0059
-0.1401	-0.0240

127 ESPÉCIES

11 GÊNEROS

Classificação de dados



- Lembre-se: classificação é um problema supervisionado, portanto ele é baseado em **dados rotulados**
- Neste exemplo:
 - Entrada: 18 coordenadas nas asas
 - Saída: espécie e/ou gênero do inseto
- Resumindo: o problema de classificação consiste em **determinar o rótulo** de algum objeto, baseado em um conjunto de **atributos extraídos do mesmo**.
- Os atributos podem ser extraídos de maneira **explícita** ou **implícita**
- Principais algoritmos:
 - Redes neurais artificiais, árvore de decisão, suporte vector machine, K-vizinhos mais próximos.



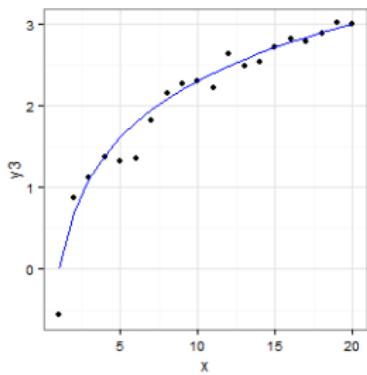
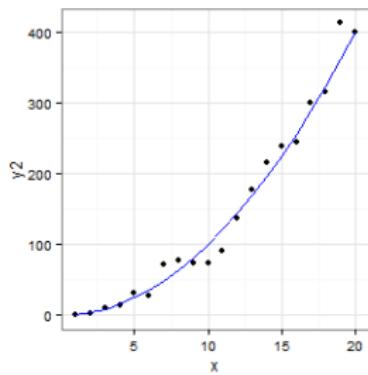
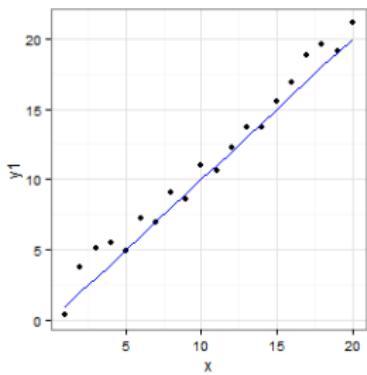
- Principais características:
 - Não possui conjunto verdade
 - Amostras não são rotuladas rotuladas
 - Amostras possuem **entradas (x)**
- **Principal objetivo:** encontrar padrões dos dados que ajudam a entender mais sobre eles
 - Não existem respostas corretas para aprender

Conceitos básicos

Problemas não supervisionados



- Dois problemas principais:
 - ▣ Clusterização (agrupamento)
 - ▣ Regressão
- Exemplo de regressão:



Conceitos básicos

Clusterização



- Similar ao problema de classificação, mas possui uma diferença primordial: **as amostras não são rotuladas!**
- O objetivo da clusterização é justamente determinar a quantidade de rótulos!
- Portanto, a classificação já parte do princípio que o número de rótulos já foi definido
 - ▣ Alguém, em algum momento, já clusterizou (mesmo que manualmente)



(a) Dados iniciais



(b) Dois clusters



(c) Três clusters



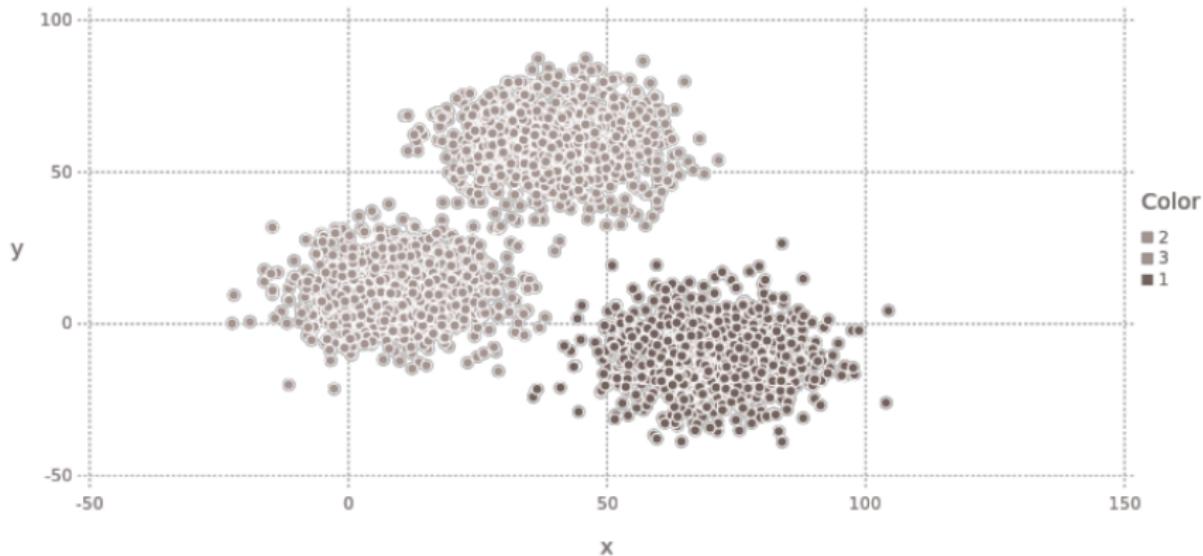
(d) Cinco clusters

Conceitos básicos

Clusterização



- Exemplo de clusterização: nível fácil

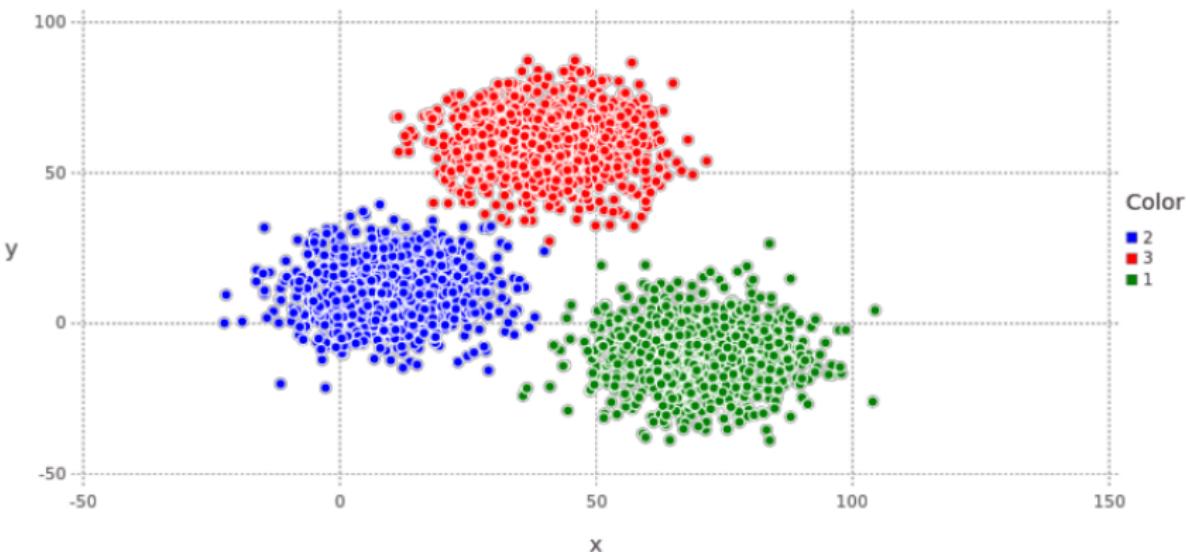


Conceitos básicos

Clusterização



- Exemplo de clusterização: nível fácil

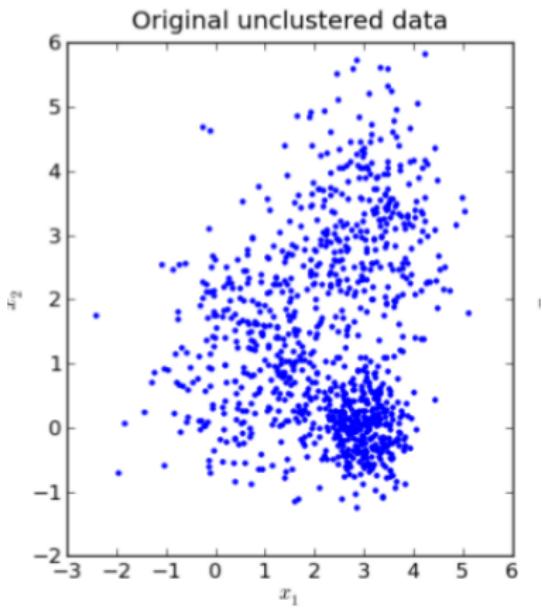


Conceitos básicos

Clusterização



- Exemplo de clusterização: nível médio/difícil

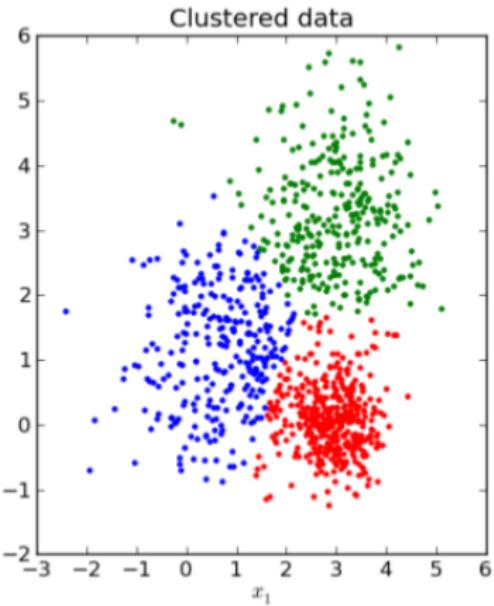
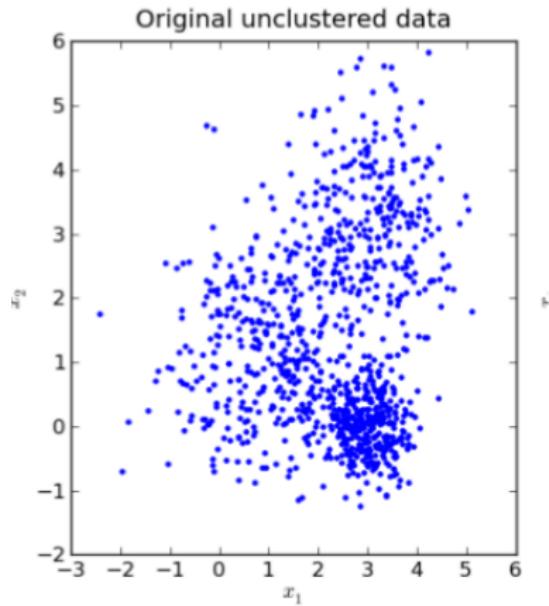


Conceitos básicos

Clusterização



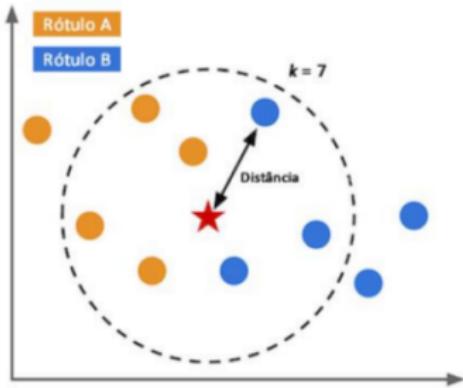
- Exemplo de clusterização: nível médio/difícil



Exemplo de um algoritmo de classificação



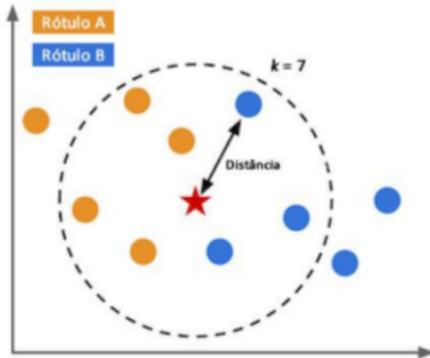
- Para ilustrar, vamos a um simples algortimos de classificação: o **K-vizinhos mais próximos (KNN)**
 - Simples e de fácil entendimento
 - Fácil de implementar
- Princípio básico: classificar amostras como semelhantes de acordo com sua distância cartesiana





- Pontos chaves:
 - Cálculo da distância
 - Escolha do valor de k
- Métrica de distância mais utilizada é a Euclidiana:

$$D = \sqrt{(p_1 - q_1)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (1)$$





- Não existe um valor ótimo para k
 - ▣ Escolhido de maneira empírica
 - ▣ Depende do problema em questão
- Quais são os pontos fracos deste algoritmo?



- Não existe um valor ótimo para k
 - Escolhido de maneira empírica
 - Depende do problema em questão
- Quais são os pontos fracos deste algoritmo?
 - Muito sensível a escolha de k
 - Calcular é custoso e pode tomar muito tempo se a base for grande (vamos abordar melhor isso, calma)

Conceitos básicos

Resumão



- O aprendizado de máquina é uma área da Ciência da Computação
 - Está dentro da inteligência artificial
 - Engloba deep learning
- Seus algoritmos não são desenvolvidos para uma tarefa específica
 - Existe um processo de treinamento
 - Altamente generalizável
- A matéria prima fundamental para aprendizado de máquina é **DADOS!**
- Existem uma gama de problemas difíceis que seus algoritmos podem ser aplicados

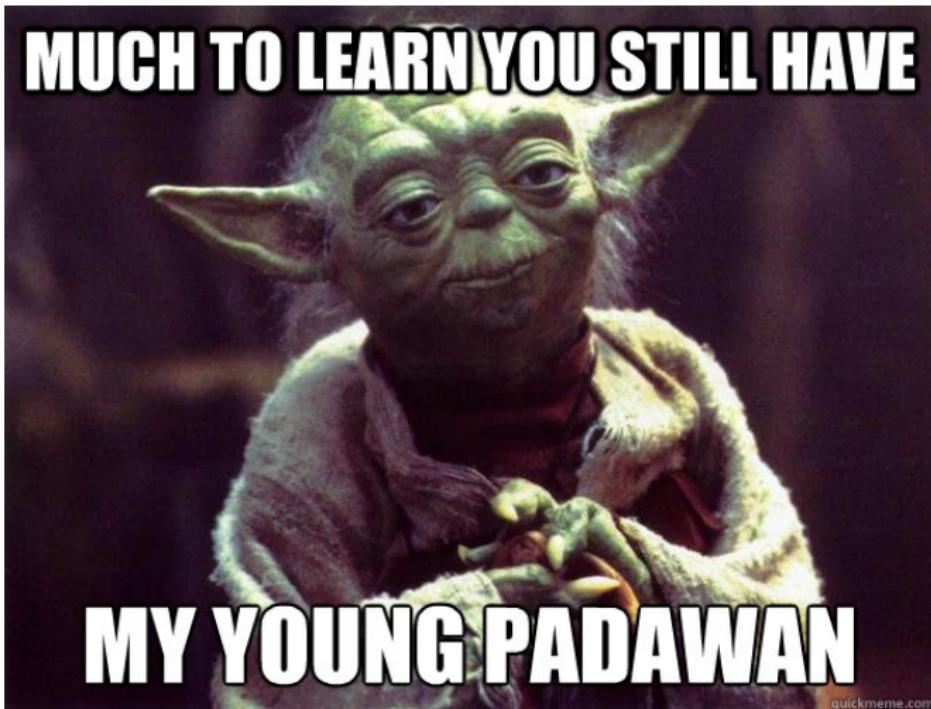
Conceitos básicos

Dúvidas?





Mas...



quickmeme.com

Deep Learning

Introdução



- A essencial do aprendizado de máquina é extrair informação de dados brutos e representá-los através de um **modelo matemático**
- Um dos principais algoritmos é a **rede neural artificial**
 - Não é uma relativamente nova
 - Desde a década de 50 ela é estudada
 - O poder de processamento limitou muito seu desenvolvimento
- Nas últimas décadas, problemas difíceis começaram a ser atacados com aprendizado de máquina:
 - Identificação de padrões em imagens
 - Tracking de veículos
 - Processamento de linguagem natural

Deep Learning

Introdução



- Porém, **quantidades massivas** de dados são necessárias para treinamento de um modelo descente
 - Consequentemente, modelos muito "profundos" começaram a ser necessários
- Durante um tempo, isso tornou-se impraticável
 - O poder de processamento de uma CPU não era suficiente
 - Um modelo poderia levar anos para ser treinado
- Eis que surge uma galera para resolver o problema!

Deep Learning

Introdução



- Os gamers!





- Ou melhor... as placas gráficas!
- Os jogos que estavam sendo desenvolvidos demandavam cada vez mais poder de computação
- A indústria de jogos fomentou o desenvolvimento de placas gráficas cada vez mais computacionalmente poderosas
- A NVIDIA foi a motor disso tudo foi!



Deep Learning

Placas gráficas



- Ok, mas qual a relação entre as placas gráficas e o aprendizado de máquina?
- A arquitetura do hardware de uma placa gráfica é bem diferente de uma CPU
 - É baseada Unidades de Processamento Gráfico (Graphics Processing Unit - GPU)
 - Baseado na classe SIMD (Single Instruction Multiple Threads)
 - Em outras palavras: **múltiplas threads** independentes são executadas de maneira concorrente utilizando uma instrução única
- Desde 2006, com lançamento da plataforma CUDA, GPUs vêm sendo utilizadas com **propósito geral de computação**.



- Uma CPU possui alguns núcleos otimizados para processamento serial e sequencial
- Uma GPU possui uma arquitetura paralela gigantesca
 - ▣ Milhares de núcleos
 - ▣ Menores e mais eficientes
 - ▣ Criados para lidar com **múltiplas tarefas simultaneamente!**
- Em outras palavras: **computação paralela!**
- Mas por que uma GPU foi construída assim?



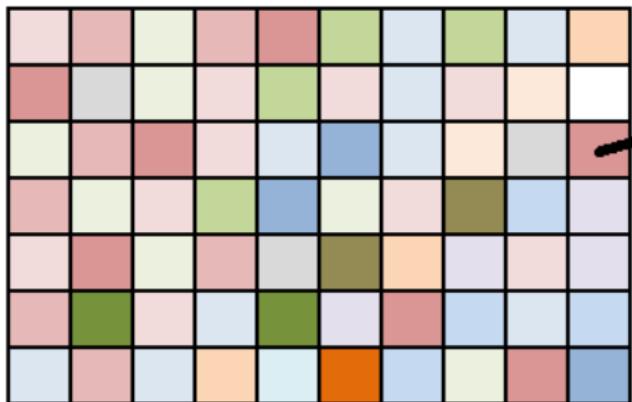
- Uma CPU possui alguns núcleos otimizados para processamento serial e sequencial
- Uma GPU possui uma arquitetura paralela gigantesca
 - Milhares de núcleos
 - Menores e mais eficientes
 - Criados para lidar com **múltiplas tarefas simultaneamente!**
- Em outras palavras: **computação paralela!**
- Mas por que uma GPU foi construída assim?
 - Qual é a matéria prima principal de um jogo?



- Uma CPU possui alguns núcleos otimizados para processamento serial e sequencial
- Uma GPU possui uma arquitetura paralela gigantesca
 - Milhares de núcleos
 - Menores e mais eficientes
 - Criados para lidar com **múltiplas tarefas simultaneamente!**
- Em outras palavras: **computação paralela!**
- Mas por que uma GPU foi construída assim?
 - Qual é a matéria prima principal de um jogo? **IMAGENS!**



- Uma imagem nada mais é que uma matriz!
 - Operações matriciais são altamente paralelizáveis!



RGB (218, 150, 149)

R = 11011010

G = 10010110

B = 10010101



- Esse ponto chamou a atenção dos pesquisadores da área de machine learning
- Os algoritmos começaram a ser programados para serem executados em GPUs
- A NVIDIA passou a desenvolver GPUs com arquitetura específica para área
- Com isso, a barreira do poder de computação limitada foi vencida
- Modelos cada vez mais grandes e complexos passaram a se tornar realidade
 - Eis que surge uma subárea: **Deep Learning**



- Esse ponto chamou a atenção dos pesquisadores da área de machine learning
- Os algoritmos começaram a ser programados para serem executados em GPUs
- A NVIDIA passou a desenvolver GPUs com arquitetura específica para área
- Com isso, a barreira do poder de computação limitada foi vencida
- Modelos cada vez mais grandes e complexos passaram a se tornar realidade
 - Eis que surge uma subárea: **Deep Learning**



- Para entender Deep Learning é necessário entender as redes neurais artificiais (RNA ou ANN)
- RNAs são modelos de processamento paralelo de informações constituído pela interconexão de unidades básicas de processamento, **o neurônio artificial**
 - São **inspiradas** no modelo neural humano
- Existem diversos modelos:
 - **Feed Forward Network**
 - **Convolution Neural Network**
 - Autoencoder
 - Restricted Boltzmann Machine
 - Recurrent Neural Network



A mostly complete chart of

Neural Networks

©2016 Fjodor van Veen - asimovinstitute.org

○ Backfed Input Cell

○ Input Cell

△ Noisy Input Cell

● Hidden Cell

○ Probabilistic Hidden Cell

△ Spiking Hidden Cell

● Output Cell

○ Match Input Output Cell

● Recurrent Cell

○ Memory Cell

△ Different Memory Cell

● Kernel

○ Convolution or Pool

Perceptron (P)



Feed Forward (FF)



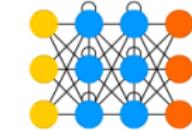
Radial Basis Network (RBF)



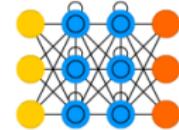
Deep Feed Forward (DFF)



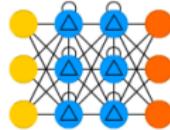
Recurrent Neural Network (RNN)



Long / Short Term Memory (LSTM)



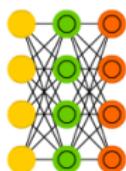
Gated Recurrent Unit (GRU)



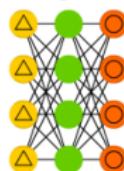
Auto Encoder (AE)



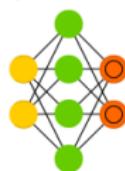
Variational AE (VAE)



Denoising AE (DAE)

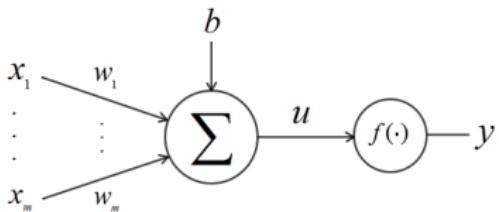


Sparse AE (SAE)





- Independente do modelo, tudo começa no neurônio!
- O modelo mais comum é o perceptron:
 - Pesos de conexão: w e b (armazenam o conhecimento)
 - $f(u)$ é uma não linearidade inserida no sistema
 - Por exemplo uma smoothRelu ou sigmoide



$$y = f(u) \rightarrow u = \left(\sum_{i=1}^m x_i w_i + b \right) \quad (2)$$

$$f(u) = \log(1 - e^{-u}) \quad (3)$$

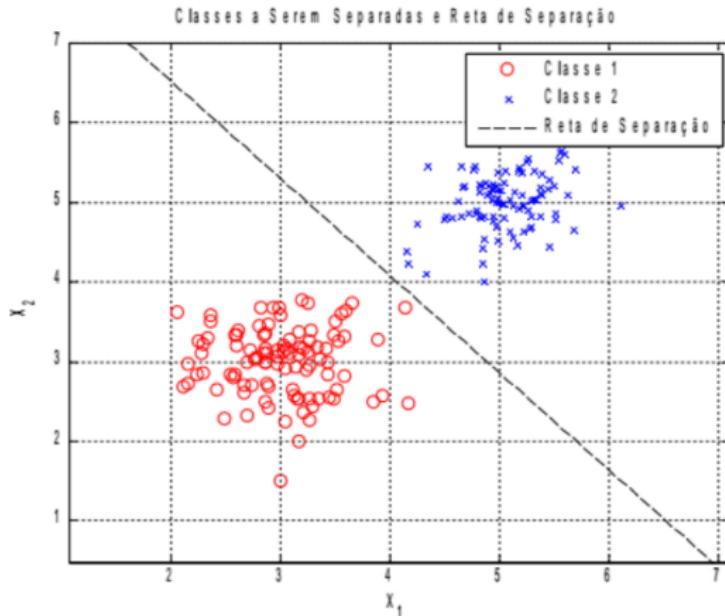


- Independente do modelo, tudo começa no neurônio!
- O modelo mais comum é o perceptron:
 - ▣ $f(u)$ é uma não linearidade inserida no sistema
 - ▣ Por exemplo uma smoothRelu ou sigmoide
- Se o neurônio possui apenas duas entradas:

$$u = \left(\sum_{i=1}^m x_i w_i + b \right) \rightarrow u = x_1 w_1 + x_2 w_2 + b \quad (4)$$

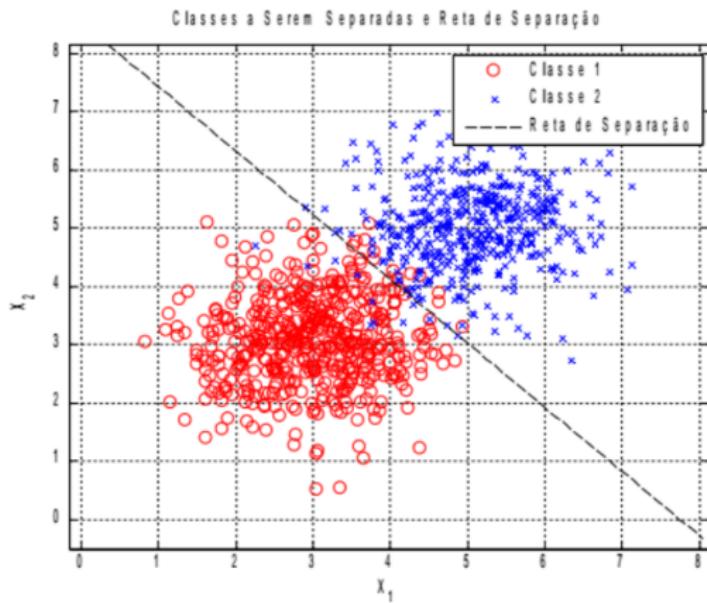


- A capacidade de um perceptron: problemas linearmente separáveis



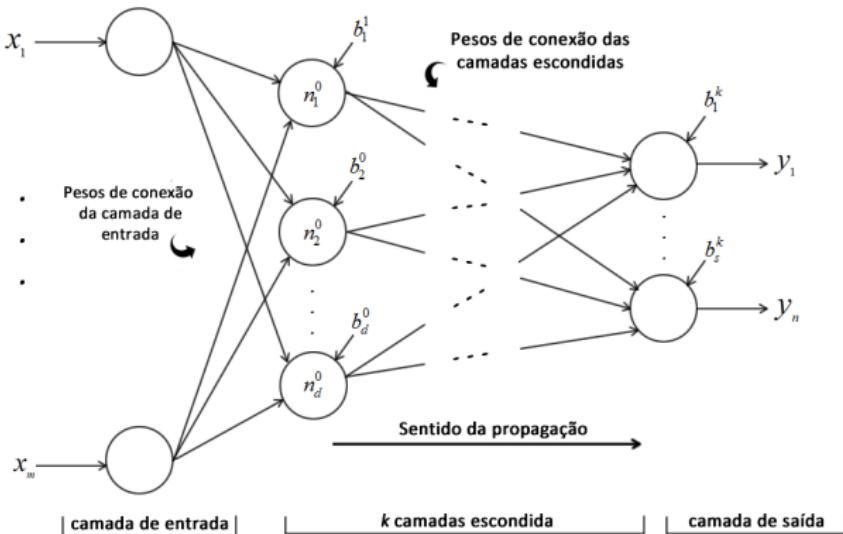


- A capacidade de um perceptron: problemas linearmente separáveis





- A junção de vários neurônios, forma uma rede neural artificial
- Como uma Feed Forward Network
 - Possui 3 tipos camadas: entrada, oculta(s), saída



Deep Learning

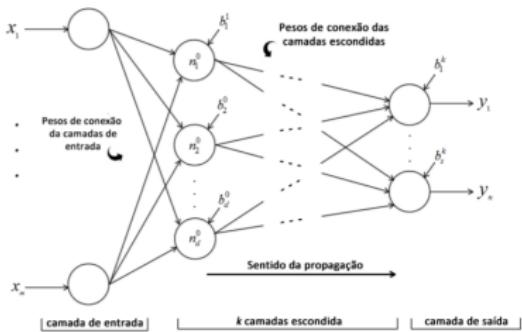
Redes neurais artificiais



- Como modelar matemáticamente uma rede neural?



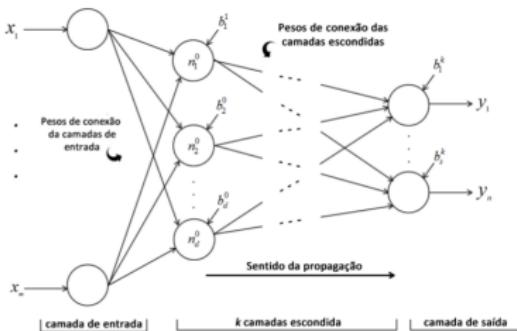
- Como modelar matemáticamente uma rede neural?
 - MATRIZES!



$$[n_1^0, n_2^0, \dots, n_d^0] = [x_1, \dots, x_m] \begin{bmatrix} w_{11}^0 & \cdots & w_{1d}^0 \\ \vdots & \ddots & \vdots \\ w_{m1}^0 & \cdots & w_{md}^0 \end{bmatrix} + [b_1^0, b_2^0, \dots, b_d^0]$$



- Como modelar matemáticamente uma rede neural?
 - MATRIZES!



$$[n_1^0, n_2^0, \dots, n_d^0] = [x_1, \dots, x_m, 1] \begin{bmatrix} w_{11}^0 & \cdots & w_{1d}^0 \\ \vdots & \ddots & \vdots \\ w_{m1}^0 & \cdots & w_{md}^0 \\ b_1^0 & \cdots & b_d^0 \end{bmatrix}$$



- Como determinar o número de neurônios de cada camada?
 - Os neurônios de entrada e saída são determinados pelo problema
 - Os neurônios ocultos, **empiricamente**
- Como determinar o número de camadas ocultas?



- Como determinar o número de neurônios de cada camada?
 - Os neurônios de entrada e saída são determinados pelo problema
 - Os neurônios ocultos, **empiricamente**
- Como determinar o número de camadas ocultas?
 - Também é feito de maneira empírica
 - Determina a profundidade da rede (daí o deep)



- Como determinar o número de neurônios de cada camada?
 - Os neurônios de entrada e saída são determinados pelo problema
 - Os neurônios ocultos, **empiricamente**
- Como determinar o número de camadas ocultas?
 - Também é feito de maneira empírica
 - Determina a profundidade da rede (daí o deep)
- Como determinar os pesos?



- Como determinar o número de neurônios de cada camada?
 - Os neurônios de entrada e saída são determinados pelo problema
 - Os neurônios ocultos, **empiricamente**
- Como determinar o número de camadas ocultas?
 - Também é feito de maneira empírica
 - Determina a profundidade da rede (daí o deep)
- Como determinar os pesos?
 - É necessário um **processo de treinamento**
 - A rede aprende com exemplos! (aprendizado supervisionado)*



- O processo de treinamento mais utilizado é o **backpropagation**
 - Baseado na descida do gradiente (agora que entra o cálculo)
 - É necessário uma medida de erro entre a saída real e a saída da rede
- Uma medida comum: o erro quadrático médio

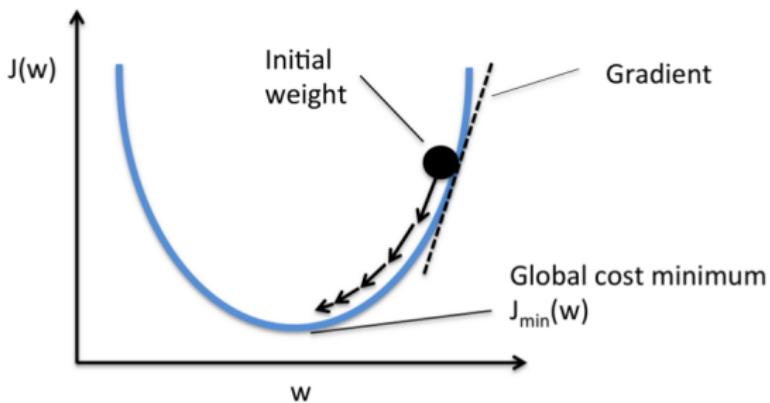
$$E = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2 \quad (5)$$

- A partir do erro, é calculado o gradiente:

$$W^{t+1} = W^t + \Delta W^t \rightarrow \Delta W = -\alpha \frac{\partial E}{\partial W} \quad (6)$$

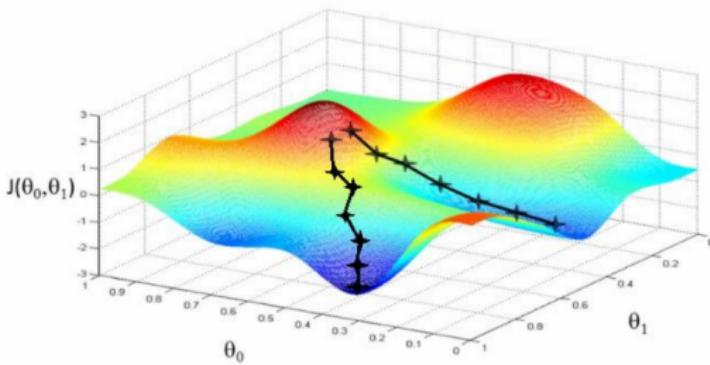


- A ideia do backpropagation:
 - ▣ Inicialização **estocástica** dos pesos
 - ▣ Executar a regra de atualização dos pesos diversas vezes até um critério de parada
 - ▣ Conseguem enxergar a influência de α ?





- A ideia do backpropagation:
 - Inicialização **estocástica** dos pesos
 - Executar a regra de atualização dos pesos diversas vezes até um critério de parada
 - Conseguem enxergar a influência de α ?





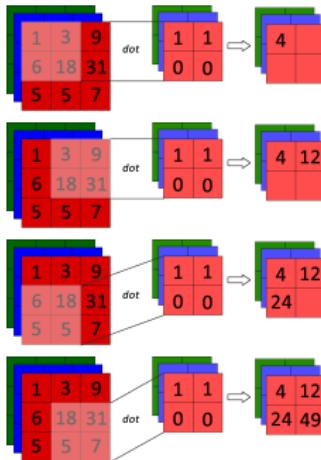


- Atualmente o motor do deep learning é as redes convolutivas
 - Proposta para trabalhar com imagens
 - Realiza diversos filtros de convolução nas imagens
 - Também treinada pelo backpropagation
- Possuem 3 camadas principais:
 - Convolutiva
 - Pooling
 - Totalmente conectada (FFN)



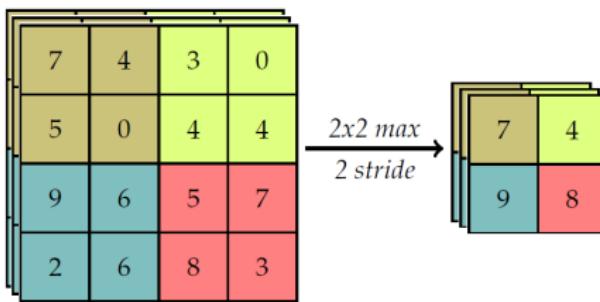
- Camada convolutiva

- Mais importante da rede
- Responsável por extrair características das imagens
- Composta por um conjunto de filtros





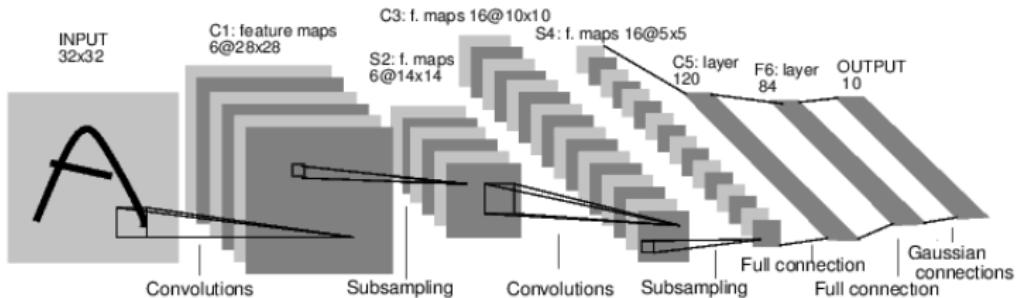
- Camada de pooling
 - Utilizada para reduzir o tamanho espacial das matrizes resultantes da convolução



- Camada totalmente conectada
 - É uma rede neural do tipo feed forward

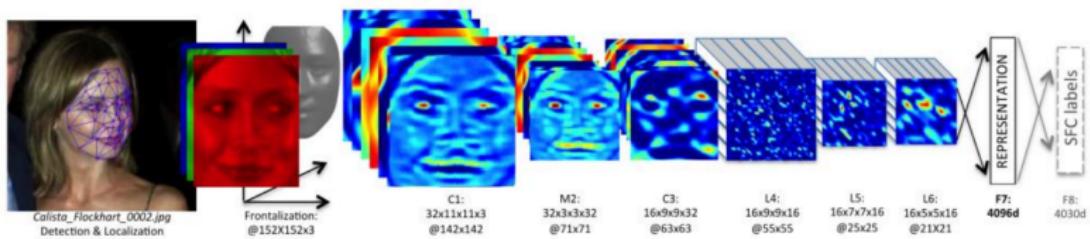


- Ilustração de uma CNN: Lenet5
 - Primeira rede a dar realmente certo!





- No fundo, o que uma CNN faz é encontrar padrões!

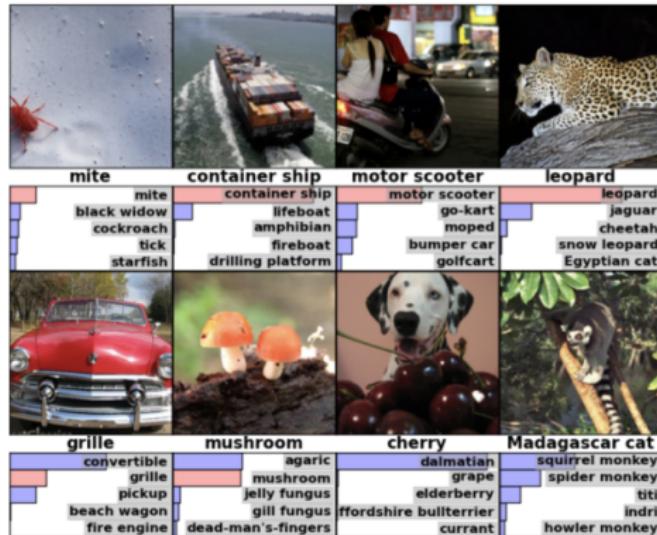


Deep Learning

Desafio ImageNet

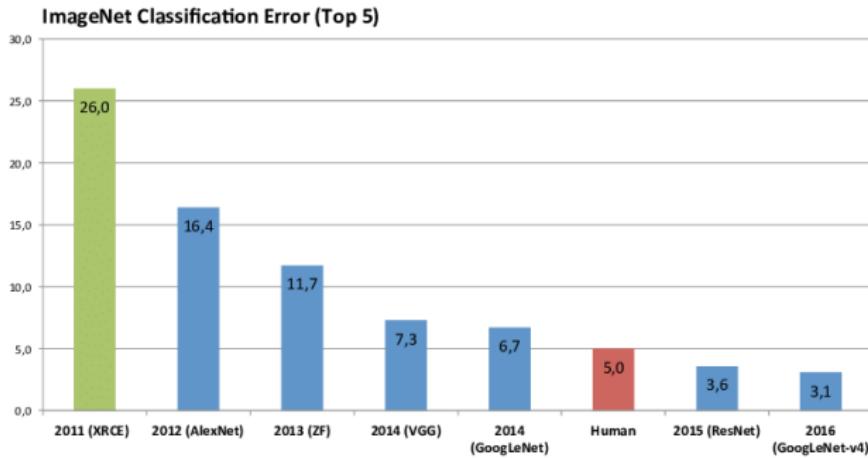


- É um desafio internacional de classificação de imagens
 - Perto de 1 bilhão de amostras
 - 1000 classes





- Primeira CNN a mudar o patamar do desafio: AlexNet
 - Primeiro grande boom do deep learning
 - Os olhos do mundo se voltaram para CNN
- Redes com mais de 100 camadas convolutivas começaram a ser propostas





Como iniciar na área?



- Basicamente a comunidade adotou **Python** como a linguagem de machine learning
 - NumPy, SciPy, Pandas, Matplotlib





- Um excelente framework para machine learning é o **Scikit-learning**
 - Dezenas de algoritmos implementados
 - Muitas métricas e funções para gerar gráfico



Como iniciar na área?



- Para deep learning, não tem como fugir da computação em GPUs
- Para isso, existem duas opções excelentes: Tensorflow e Pytorch



Obrigado

Qualquer dúvida, me envie um e-mail

