

Aprendizado de Máquina

Perceptron

Luiz Eduardo S. Oliveira

Universidade Federal do Paraná
Departamento de Informática
web.inf.ufpr.br/luizoliveira

Introdução

- Um dos maiores desafios em aprendizagem de máquina e reconhecimento de padrões é a definição de um conjunto de características com alto poder de discriminação.
- Isso não é uma tarefa trivial.
- Alguns argumentam que o ponto fraco nos métodos de aprendizagem de máquina está exatamente na definição de características.

Introdução

- Um dos maiores desafios em aprendizagem de máquina e reconhecimento de padrões é a definição de um conjunto de características com alto poder de discriminação.
- Isso não é uma tarefa trivial.
- Alguns argumentam que o ponto fraco nos métodos de aprendizagem de máquina está exatamente na definição de características.

Alternativa

Aprender as características a partir dos dados!

Introdução

- A inspiração vem da arquitetura do cérebro humano.
- Por décadas, os pesquisadores tentaram treinar arquiteturas profundas de redes neurais (com diversas camadas)
- Um experimento somente teve sucesso (CNN, Yann LeCun em 1998)
- Alto custo computacional e muitos dados são necessários para a aprendizagem dos modelos
- Anos 90 e 2000 foram dominados pelo SVM
- Os métodos começaram a se popularizar por volta de 2010 (paralelismo, GPU)

The screenshot shows the MIT Technology Review website interface. At the top, there is a navigation bar with a 'T' logo, 'HOME', 'MENU', 'CONNECT', 'THE LATEST', 'POPULAR', and 'MOST SHARED'. Below this, the main header features the MIT Technology Review logo and the title '10 BREAKTHROUGH TECHNOLOGIES 2013'. To the right of the title are three links: 'Introduction', 'The 10 Technologies', and 'Past Years'. The main content area displays a grid of ten technology cards. The first card, 'Deep Learning', is highlighted with a red rectangle. It contains the text: 'With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.' and a right-pointing arrow. The other cards are: 'Temporary Social Media' (Messages that quickly self-destruct could enhance the privacy of online communications and make people freer to be spontaneous.), 'Prenatal DNA Sequencing' (Reading the DNA of fetuses will be the next frontier of the genomic revolution. But do you really want to know about the genetic problems or musical aptitude of your unborn child?), 'Additive Manufacturing' (Skeptical about 3-D printing? GE, the world's largest manufacturer, is on the verge of using the technology to make jet parts.), 'Baxter: The Blue-Collar Robot' (Rodney Brooks's newest creation is easy to interact with, but the complex innovations behind the robot show just how hard it is to get along with people.), 'Memory Implants', 'Smart Watches', 'Ultra-Efficient Solar Power', 'Big Data from Cheap Phones', and 'Supergrids'. Each card has a right-pointing arrow at the bottom right.

Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart. →

Temporary Social Media

Messages that quickly self-destruct could enhance the privacy of online communications and make people freer to be spontaneous. →

Prenatal DNA Sequencing

Reading the DNA of fetuses will be the next frontier of the genomic revolution. But do you really want to know about the genetic problems or musical aptitude of your unborn child? →

Additive Manufacturing

Skeptical about 3-D printing? GE, the world's largest manufacturer, is on the verge of using the technology to make jet parts. →

Baxter: The Blue-Collar Robot

Rodney Brooks's newest creation is easy to interact with, but the complex innovations behind the robot show just how hard it is to get along with people. →

Memory Implants

Smart Watches

Ultra-Efficient Solar Power

Big Data from Cheap Phones

Supergrids

MIT Technology Review. April 23rd. 2013

Um breve histórico

- 1949, Donald Hebb
 - ▶ Definiu que a informação da rede fica armazenada nos pesos.
 - ▶ Propõe uma lei de aprendizagem específica para as sinapses dos neurônios
 - ★ Quando ocorre o aprendizado os pesos são atualizados
- 1953, McCulloch e Pitts
 - ▶ Estudaram o comportamento do neurônio biológico com o objetivo de criar um modelo matemático para este.
 - ▶ Sugeriram a construção de uma máquina baseada ou inspirada no cérebro humano.

Um breve histórico

- 1957/1958, Frank Rosenblatt
 - ▶ Estudos aprofundados
 - ▶ Pai da neuro-computação
 - ▶ Perceptron
 - ▶ Criados do primeiro neuro-computador a obter sucesso
 - ★ Mark I

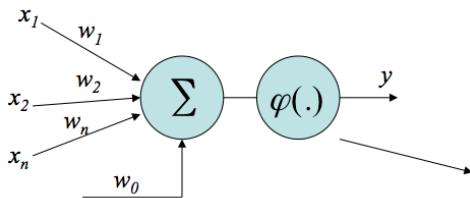


Um breve histórico

- 1958-1967
 - ▶ Várias pesquisas mal sucedidas.
- 1967-1982
 - ▶ Pesquisas silenciosas.
- 1986
 - ▶ Livro “Parallel Distributed Processing”.
 - ▶ Algoritmo eficaz de aprendizagem
- 1987
 - ▶ Primeira conferência IEEE IJCNN (International Joint Conference on Neural Networks)

Perceptron

- Primeira e mais primitiva estrutura de rede neural



$$y = \varphi\left(\sum w_i \times x_i + w_0\right)$$

A função de ativação normalmente utilizada no perceptron é a hardlim (threshold)

$$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$



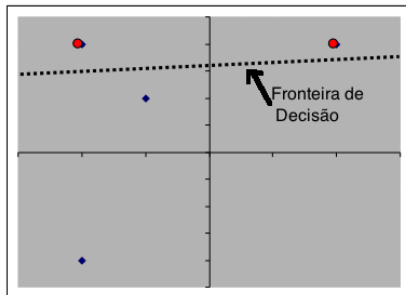
Algoritmo de Aprendizagem

- Iniciar os pesos e bias com valores pequenos, geralmente no intervalo $[0.3, 0.8]$
- Enquanto critério de parada não for alcançado
 - ▶ Aplicar um padrão de entrada com seu respectivo valor desejado de saída (t_j) e verificar a saída y da rede.
 - ▶ Calcular o erro da saída ($e = t_j - a$)
 - ▶ Se $e \neq 0$
 - ★ Atualizar pesos ($w_i = w_i^{old} + e \times x_i$)
 - ★ Atualizar bias ($b = b^{old} + e$)

Perceptron

- Considere o seguinte exemplo

Características (X)		Label (t)
2	2	0
-2	-2	1
-2	2	0
-1	1	1



Perceptron

Nesse exemplo, vamos inicializar os pesos e bias com 0, ou seja, $w = (0, 0)$ e $b = 0$

Apresentando o primeiro padrão (x_1) a rede

- $y = \text{hardlim}([0, 0][2, 2]^t + 0) = \text{hardlim}(0) = 1$
- Erro: $e = t_j - y = 0 - 1 = -1$
- Como o erro $\neq 0$, atualizam-se os pesos e bias
 - ▶ $W = W^{old} + e \times x_i = [0, 0] + (-1[2, 2]) = [-2, -2]$
 - ▶ $b = b^{old} + e = 0 + (-1) = -1$

Perceptron

Apresentando o segundo padrão (x_2) a rede

- $y = \text{hardlim}([-2, -2][-2, -2]^t + (-1)) = \text{hardlim}(7) = 1$
- Erro: $e = t_j - y = 1 - 1 = 0$
- Como o erro = 0, pesos e bias não precisam ser atualizados

Apresentando o terceiro padrão (x_3) a rede

- $y = \text{hardlim}([-2, -2][-2, 2]^t + (-1)) = \text{hardlim}(-1) = 0$
- Erro: $e = t_j - y = 0 - 0 = 0$
- Como o erro = 0, pesos e bias não precisam ser atualizados

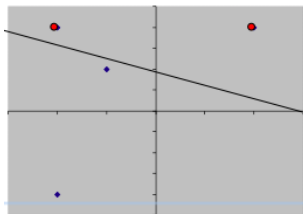
Perceptron

Apresentando o quarto padrão (x_4) a rede

- $y = \text{hardlim}([-2, -2][-1, 1]^t + (-1)) = \text{hardlim}(-1) = 0$
- Erro: $e = t_j - y = 1 - 0 = 1$
- Como o erro $\neq 0$, atualizam-se os pesos e bias
 - ▶ $W = W^{old} + e \times x_i = [-2, -2] + (1[-1, 1]) = [-3, -1]$
 - ▶ $b = b^{old} + e = -1 + 1 = 0$

Perceptron

- Como todos os padrões foram apresentados a rede, o processo começa novamente, utilizando os pesos encontrados no último passo da primeira época (iteração).
- O algoritmo acaba quando o erro for zero (todos os padrões forem classificados corretamente) ou quando um certo número de épocas for alcançado.
- Para esse exemplo, os pesos finais são $w = [-1, -3]$ e $b = 2$.



Perceptron

- Resolve problemas linearmente separáveis
- Entretanto, nem sempre os problemas são linearmente separáveis.

