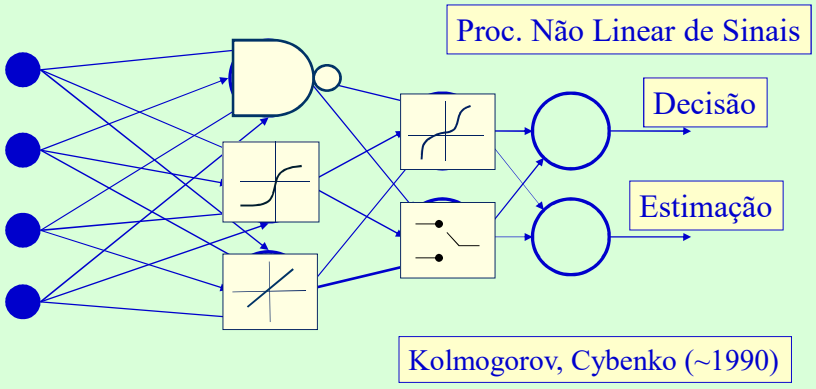
O Multi Layer Perceptron (MLP)

- Múltiplas entradas / Múltiplas saídas / Múltiplas camadas
- Variáveis (internas e externas) analógicas ou digitais
- Relações lineares ou não lineares entre elas



© Prof. Emilio Del Moral Hernandez

Resumindo os aspectos conceituais principais

1) Não linearidade com a função neural sigmoidal

2) Possibilidade de conjugar na mesma estrutura ...
   - Cálculos digitais
   - Cálculos lineares multivariáveis
   - Funções genéricas não lineares multivariáveis

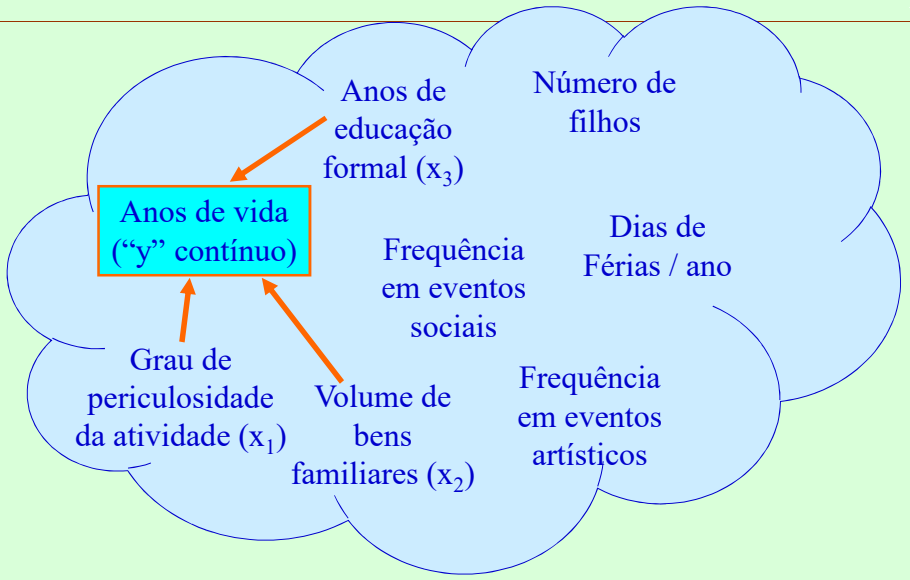
3) Comportamento adaptativo com aprendizado através de exemplos

---

- Problemas complexos, multidimensionais, não lineares e mesmo aqueles sem teoria conhecida
- Decisão automática, estimação, reconhecimento de padrões, classificação, processamento não linear de sinais, *clustering* multidimensional ...

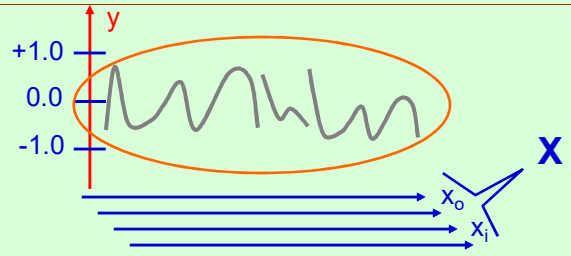
© Prof. Emilio Del Moral Hernandez

Um hipotético universo de variáveis inter-dependentes, passível de modelagem/ens

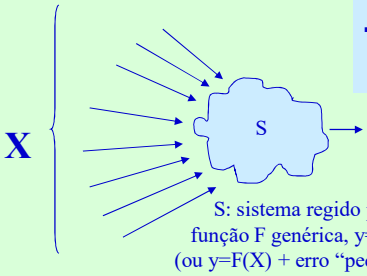


© Prof. Emilio Del Moral Hernandez

A função y(X) "a descobrir", num caso geral de função contínua y(X) ....



- ESTIMADOR ;
- MODELAGEM POR MAPEAMENTO X→y

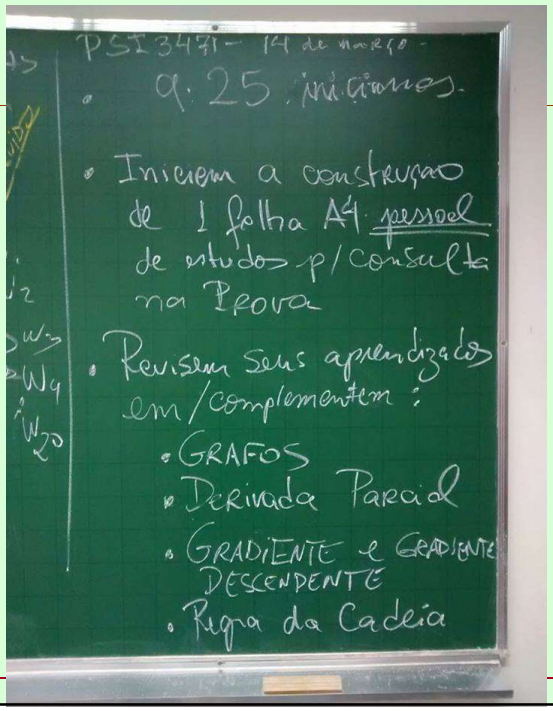


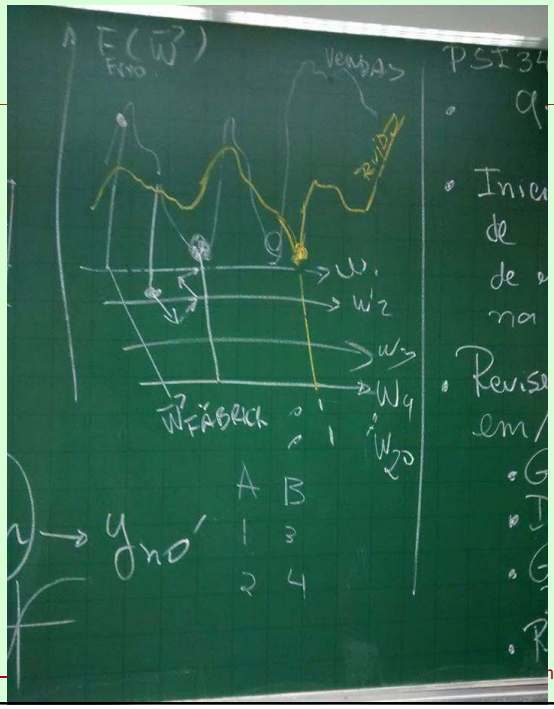
(.... VALORES CONTÍNUOS!!!)

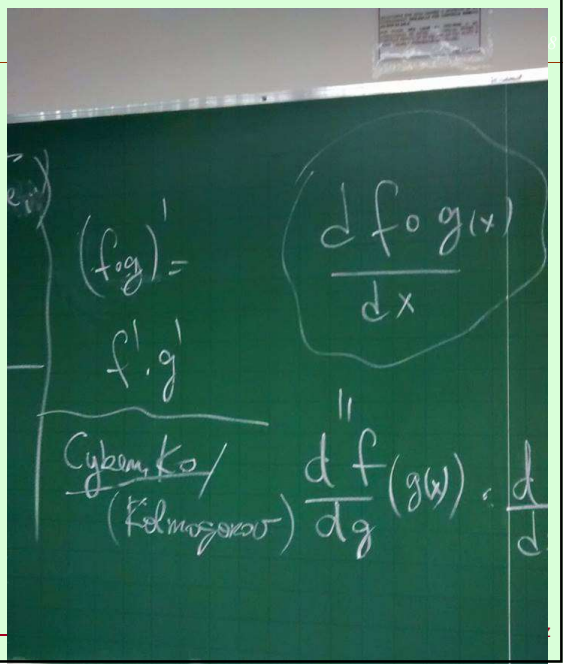
S: sistema regido por função F genérica, y=F(X) (ou y=F(X) + erro "pequeno")

© Prof. Emilio Del Moral Hernandez

Como escolhemos os valores dos diversos w´s ?



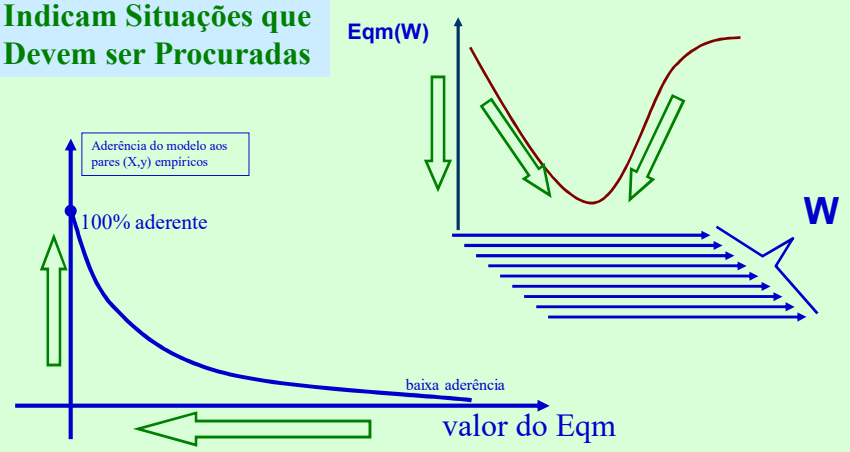

VENDAS
RUÍDO
W1
W2
W3
W4
W20
A B
1 3
2 4


(f∘g)' =
f'.g'
Cyklem, ko/
(Kolmogorov)

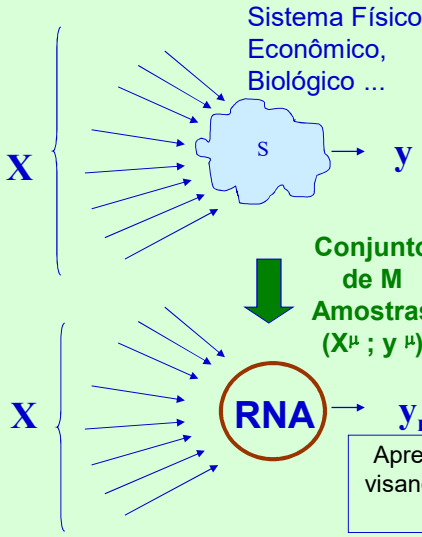
O que devemos buscar quando exploramos o espaço de pesos W buscando que a RNA seja um bom modelo?

Devemos buscar Maximização da aderência = Minimo Eqm possível

As Setas Verdes Indicam Situações que Devem ser Procuradas



Conjunto de treino em arquiteturas supervisionadas (ex. clássico: MLP com Error Back Propagation)



A computação desejada da rede pode ser definida simplesmente através de amostras / exemplos do comportamento requerido

$$Eqm = \frac{1}{M}\sum_{\mu=1}^{M}(y_{rede}(\vec{X}^{\mu},\vec{W}) - y^{\mu})^2$$

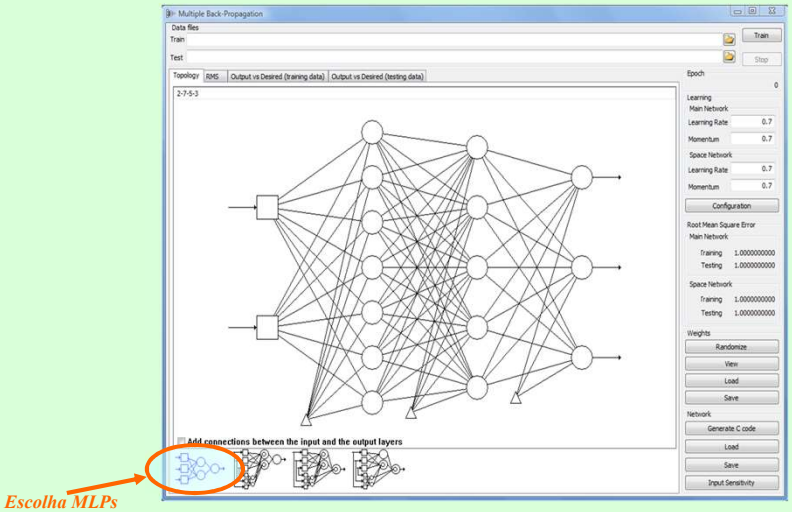
... em loop ...

$$\Delta\vec{W} = -\eta.\nabla Eqm$$

Aprendizado: Espaço de pesos W é explorado visando aproximar ao máximo a computação da rede da computação desejada

© Prof. Emilio Del Moral Hernandez

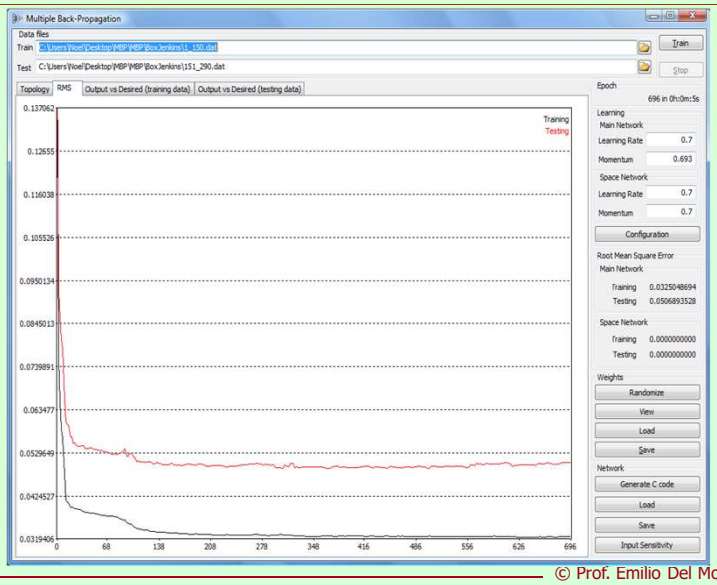
Exemplo de tela do ambiente MBP definindo uma Rede Neural do tipo MLP – Topology “2-7-5-3”



Escolha MLPs

Algumas Telas do MBP: acompanhando a queda do erro (RMS) da rede neural enquanto está aprendendo / adaptando seus pesos

(nota: O Eqm é um primo direto do RMS ... RMS = $Eqm^{1/2}$)



Aprendizado do MLP por Error Back Propagation ...

$$\Delta\vec{W} = -\eta.\vec{\nabla}Eqm_$$

Gradiente de Eqm no espaço de pesos = ($\partial Eqm(W)/\partial w_1$, $\partial Eqm(W)/\partial w_2$, $\partial Eqm(W)/\partial w_3$, ...)

Chegando às fórmulas das derivadas parciais, necessárias à Bússola do Gradiente

35

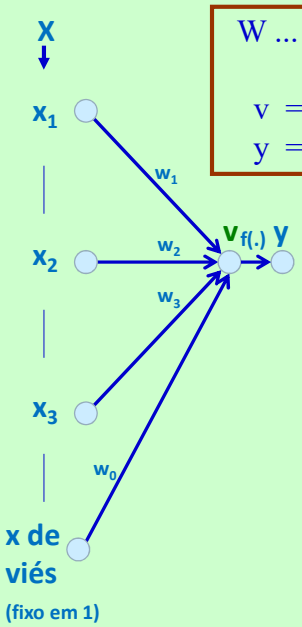
© Prof. Emilio Del Moral – EPUSP

Aplainando o terreno ... Para o gradiente descendente ...

Vamos falar um pouco sobre GRAFOS?

© Prof. Emilio Del Moral Hernandez

38



W ... inclui w_1 a w_3
e também o viés (w_0)

$v = X.W = \Sigma\, w_i\, x_i$

$y = f(v)$

Grafo de cálculos para um único nó neural ...

© Prof. Emilio Del Moral Hernandez

... erro da rede com relação ao conjunto de treinamento como um todo; simbologia (X^μ ; y^μ); Erro quadrático de exemplar (Eq^μ); Erro quadrático médio (Eqm)

Resumo de principais resultados em lousa ...

μ *identifica um de M exemplos de treinamento*

$$Eqm = \frac{1}{M}\sum_{\mu=1}^{M}(y_{rede}(\vec{X}^{\mu}) - y^{\mu})^2$$

$$Eqm = \frac{1}{M}\sum_{\mu=1}^{M}(y_{rede}(\vec{X}^{\mu}, \vec{W}) - y^{\mu})^2$$

© Prof. Emilio Del Moral Hernandez

Exemplo de regressão multivariada para estimação contínua usando MLP

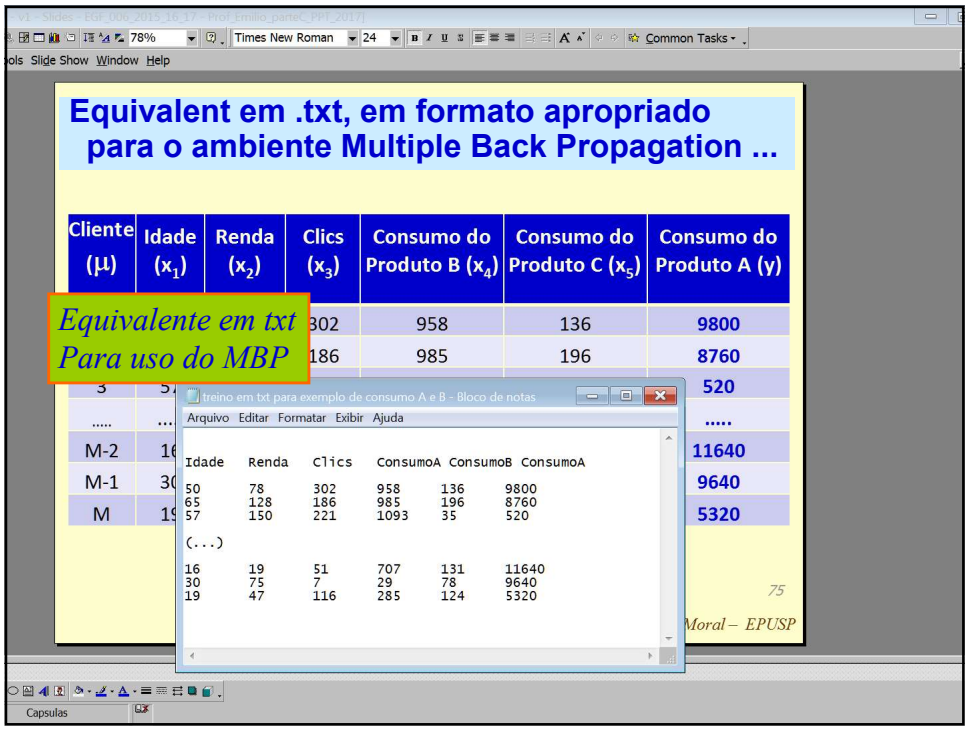
- O valor do y contínuo ... neste exemplo corresponde ao volume de consumo futuro num dado tipo de produto "A" a ser ofertado pela empresa a um cliente corrente já consumidor de outros produtos da empresa ("B" e "C"), volume esse previsto com base em várias medidas quantitativas que caracterizam tal indivíduo. ... Assim, y = Consumo do Produto A = $F(x_1,x_2, x_3, x_4, x_5)$.
- Consideremos 4 variáveis de entrada no modelo preditivo neural, ou seja, temos 5 medidas em X:
  - x_1: Idade do indivíduo
  - x_2: Renda mensal do indivíduo
  - x_3: Volume de clicks do indivído no website de exibição de produtos oferecidos pela empresa
  - x_4: Volume de consumo desse cliente observado para outro Produto B da mesma empresa
  - x_5: Volume de consumo desse cliente Produto C da mesma empresa
- Problema: desenvolver uma MLP para regressão contínua multivariada que permita estimar esse volume de consumo futuro y com base no conhecimento dos X e numa base de dados de aprendizado com esses dados X e y para 350 já clientes de universo populacional similar ao do novo consumidor potencial.

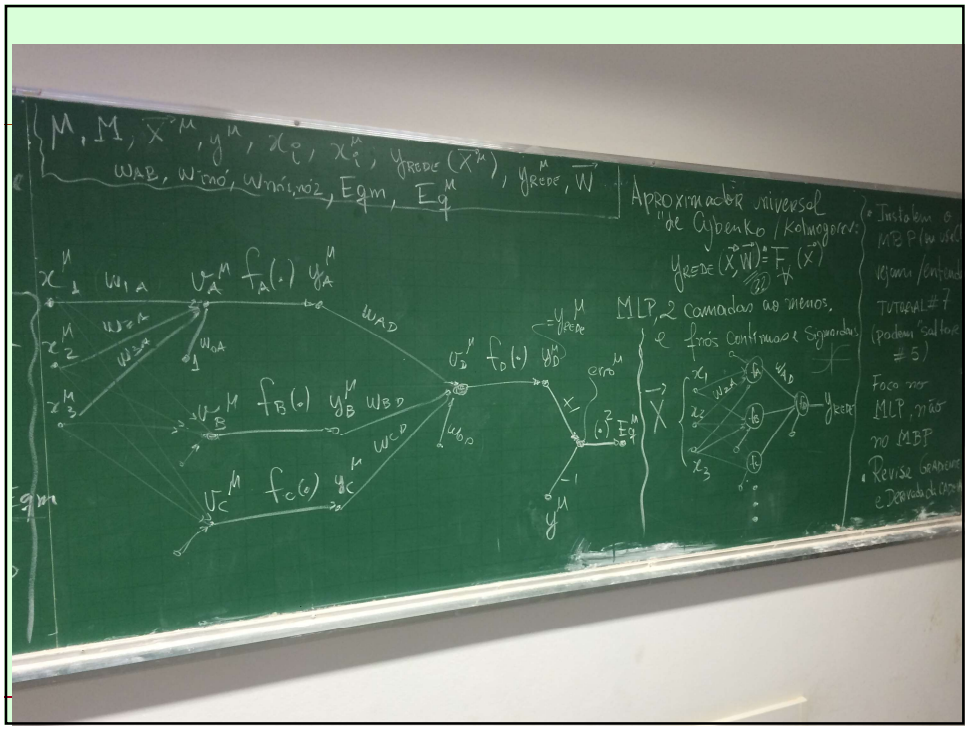
41

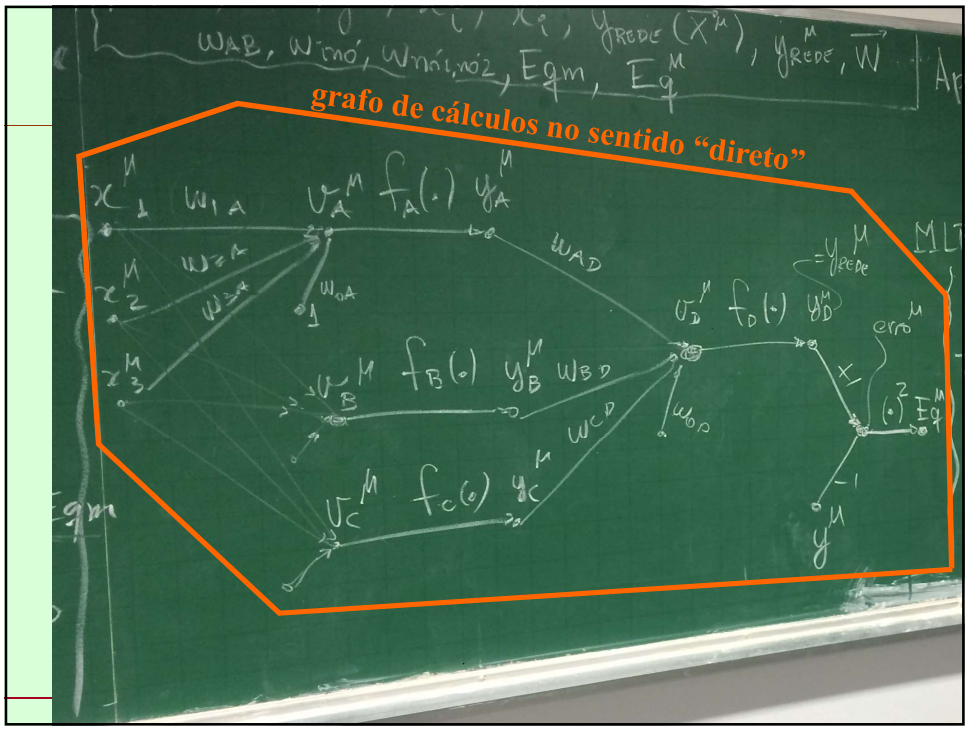
Exemplo de dados empíricos tabulados em Excel ...

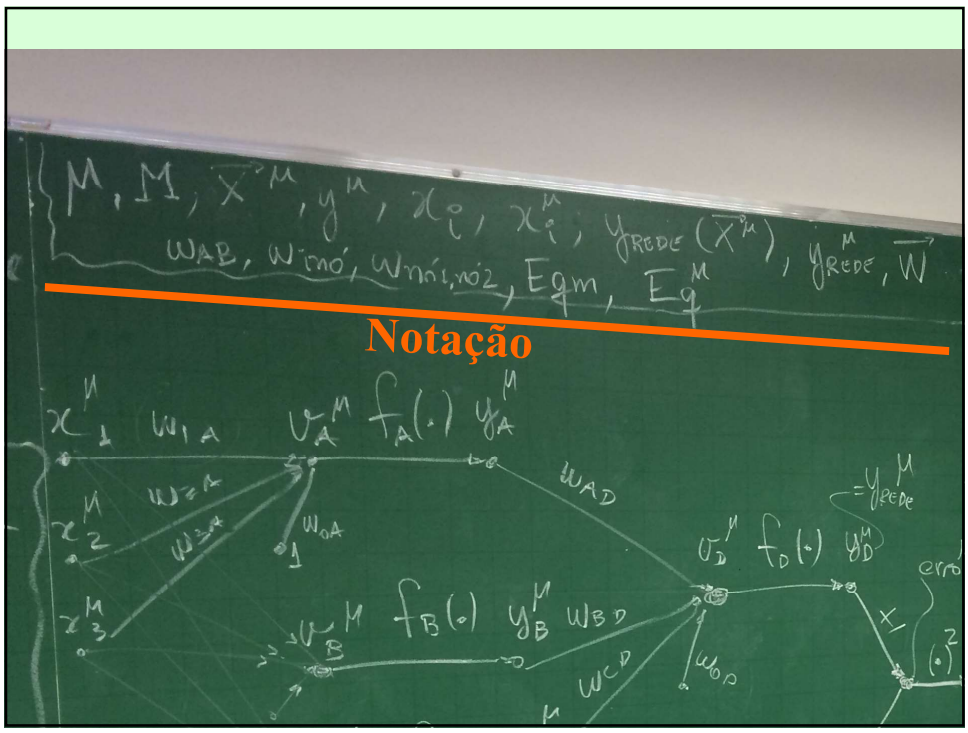
Cliente (μ)	Idade (x_1)	Renda (x_2)	Clics (x_3)	Consumo do Produto B (x_4)	Consumo do Produto C (x_5)	Consumo do Produto A (y)
1	50	78	302	958	136	**9800**
2	65	128	186	985	196	**8760**
3	57	150	221	1093	35	**520**
.....	.....	.....	.....	.....	.....	**.....**
M-2	16	19	51	707	131	**11640**
M-1	30	75	7	29	78	**9640**
M	19	47	116	285	124	**5320**

42

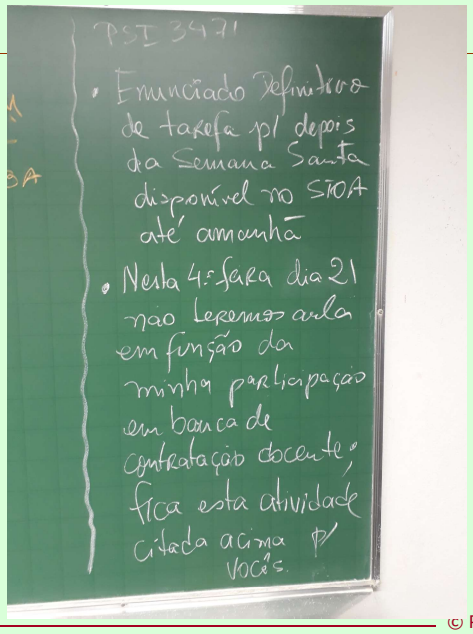

Equivalent em .txt, em formato apropriado para o ambiente Multiple Back Propagation ...
Cliente (μ)
Idade (x_1)
Renda (x_2)
Clics (x_3)
Consumo do Produto B (x_4)
Consumo do Produto C (x_5)
Consumo do Produto A (y)
Equivalente em txt
Para uso do MBP
302 958 136 9800
186 985 196 8760
520
.....
M-2 11640
M-1 9640
M 5320
treino em txt para exemplo de consumo A e B - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
Idade Renda Clics ConsumoA ConsumoB ConsumoA
50 78 302 958 136 9800
65 128 186 985 196 8760
57 150 221 1093 35 520
(...)
16 19 51 707 131 11640
30 75 7 29 78 9640
19 47 116 285 124 5320
75
Moral – EPUSP


Aproximador universal
MLP, 2 camadas ao menos,
e três Contínuas e Sigmoidais


grafo de cálculos no sentido "direto"


Notação

Avisos que detalhei na aula de 19 de março



© Prof. Emilio Del Moral Hernandez

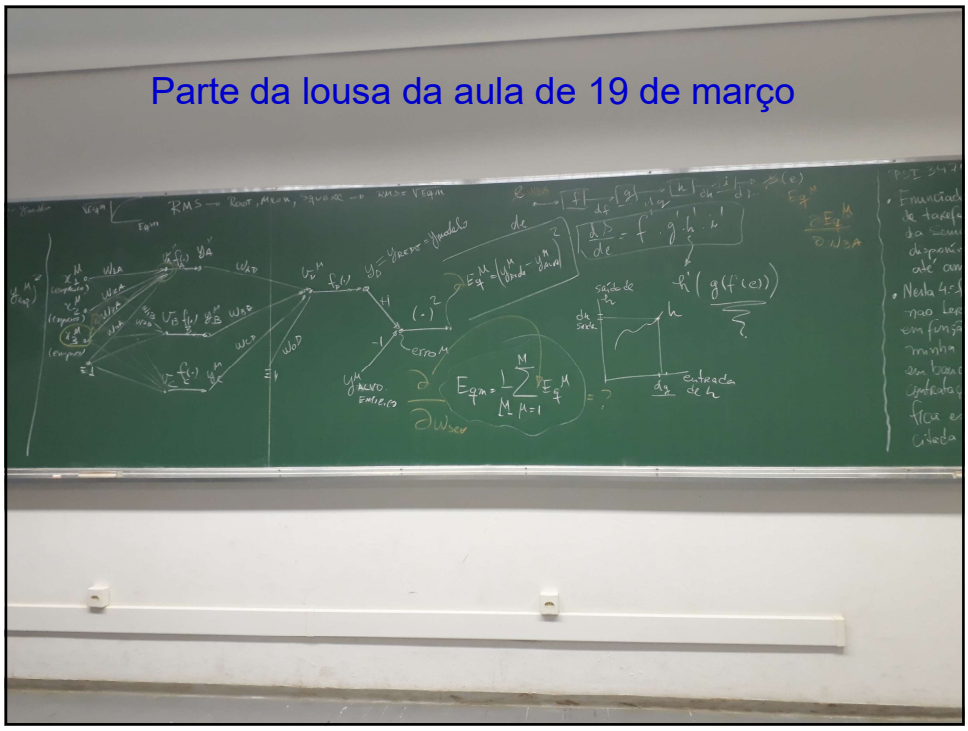
Enunciado preliminar! Pode ser ligeiramente modificado até 21/3

Tarefa para desenvolvimento próprio e entrega até a aula de 4ª f. da Semana Santa +1(dia 4/abril)

- No grafo detalhado de uma rede 3-3-1 como o da aula, identifique o peso sináptico "w_{meu}" da primeira camada definido pelo 1º dígito do seu NUSP como segue:
- 1º digito 1: w_{1A} ; 1º digito 2: w_{1B} ; 3: w_{1C} ; 4: w_{2A} ; 5: w_{2B} ; 6: w_{2C} ; 7: w_{3A} ; 8: w_{3B} ; 9: w_{3C}
- Assumindo 100 exemplares de treino (M=100), identifique o exemplar de treino "μ_{meu}" associado ao 2º e 3º dígitos de seu NUSP como segue:
- Se 2º e 3º dígitos são 0 e 1, μ = 01, se 2º e 3º dígitos são 0 e 2, μ = 02,
  se 2º e 3º são 0 e 3, μ = 03 … se 4 e 1, μ = 41 … e assim por diante
- a) Fixando a entrada X da rede no exemplar empírico μ , ou seja X = X^{μ} e usando como alvo (target) empírico y = y^{μ} , e visando o cálculo do gradiente do erro quadrático, necessário ao algoritmo de gradiente descendente (EBP – error back propagation), deduza com detalhe a expressão analítica da derivada parcial do erro quadrático do "seu" exemplar μ_{meu} com relação ao "seu" peso sináptico w_{meu}, ou seja, calcule ($\partial Eq^{\mu_meu} / \partial w_{meu}$) ; revise seus conceitos de derivada parcial e use a regra da cadeia na sua dedução.
- b) Agora empregue a formula analítica obtida em a) para um (X^{μ_meu}, y^{μ_meu}) empírico com valores razoáveis ao seu sistema regressor (já criado por você nas atividades de aula): defina valores numéricos e unidades das 4 grandezas envolvidas no cálculo, x_1^{μ}, x_2^{μ}, x_3^{μ}, e y^{μ} , e chegue ao valor numérico de ($\partial Eq^{\mu_meu} / \partial w_{meu}$)
- c) Especifique o que teve que assumir em a e b (por não definido no enunciado).

© Prof. Emilio Del Moral Hernandez

Parte da lousa da aula de 19 de março



De onde vem o grande poder do MLP?

© Prof. Emilio Del Moral Hernandez

Theorem of [Kolmogorov &] Cybenko:

- Kolmogorov:

  Given any F of many variables x_1, x_2, x_3, x_4 … for example, the complicated $F = [x_1.\sin(x_2) + \log(x_3)] / x_4$ + etc … or any other F, the following approximation can always be obtained …

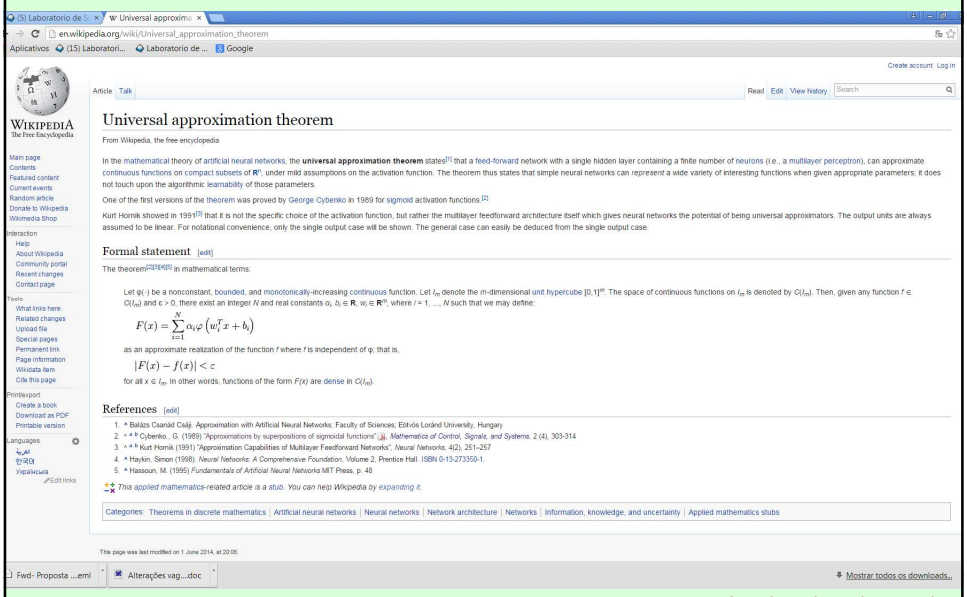
  $F(x_1, x_2, x_3, x_4 \ldots.)$ ~ linear combination and composition of a finite (limited) number of functions $g_k(v)$ of just one variable v, and we can have arbitrary precision in the approximation of F

- Cybenko: adapted Kolmogorov for the particular case in which the single argument functions g_k are approximated by a sum of sigmoidal functions … he noticed that several sigmoids shifted and scaled properly can approximate any g_k(scalar argument)

Cybenko concluded that any arbitrary F CAN be "implemented" by an ANN with sigmoidal nodes and just 1 hidden layer!!

© Prof. Emilio Del Moral Hernandez

Cybenko – Enunciado da Prova ... (premissas + resultado)



Universal approximation theorem

From Wikipedia, the free encyclopedia

In the mathematical theory of artificial neural networks, the **universal approximation theorem** states[1] that a feed-forward network with a single hidden layer containing a finite number of neurons (i.e., a multilayer perceptron), can approximate continuous functions on compact subsets of $\mathbf{R}^n$, under mild assumptions on the activation function. The theorem thus states that simple neural networks can *represent* a wide variety of interesting functions when given appropriate parameters; it does not touch upon the algorithmic learnability of those parameters.

One of the first versions of the theorem was proved by George Cybenko in 1989 for sigmoid activation functions.[2]

Kurt Hornik showed in 1991[3] that it is not the specific choice of the activation function, but rather the multilayer feedforward architecture itself which gives neural networks the potential of being universal approximators. The output units are always assumed to be linear. For notational convenience, only the single output case will be shown. The general case can easily be deduced from the single output case.

Formal statement [edit]

The theorem[2][3][4][5] in mathematical terms:

Let $\varphi(\cdot)$ be a nonconstant, bounded, and monotonically-increasing continuous function. Let I_m denote the m-dimensional unit hypercube $[0,1]^m$. The space of continuous functions on I_m is denoted by $C(I_m)$. Then, given any function $f \in C(I_m)$ and $\varepsilon > 0$, there exist an integer N and real constants $\alpha_i, b_i \in \mathbf{R}$, $w_i \in \mathbf{R}^m$, where $i = 1, \ldots, N$ such that we may define:

$$F(x) = \sum_{i=1}^{N} \alpha_i \varphi\left(w_i^T x + b_i\right)$$

as an approximate realization of the function f where f is independent of φ; that is,

$$|F(x) - f(x)| < \varepsilon$$

for all $x \in I_m$. In other words, functions of the form $F(x)$ are dense in $C(I_m)$.

References [edit]

1. ^ Balázs Csanád Csáji. Approximation with Artificial Neural Networks; Faculty of Sciences; Eötvös Loránd University, Hungary
2. ^ ^a ^b Cybenko. . G. (1989) "Approximations by superpositions of sigmoidal functions", *Mathematics of Control, Signals, and Systems*, 2 (4), 303-314
3. ^ ^a ^b Kurt Hornik (1991) "Approximation Capabilities of Multilayer Feedforward Networks", *Neural Networks*, 4(2), 251–257
4. ^ Haykin, Simon (1998). *Neural Networks: A Comprehensive Foundation*, Volume 2, Prentice Hall. ISBN 0-13-273350-1.
5. ^ Hassoun, M. (1995) *Fundamentals of Artificial Neural Networks* MIT Press. p. 48

This applied mathematics-related article is a stub. You can help Wikipedia by expanding it.

Categories: Theorems in discrete mathematics | Artificial neural networks | Neural networks | Network architecture | Networks | Information, knowledge, and uncertainty | Applied mathematics stubs

This page was last modified on 1 June 2014, at 20:06.

© Prof. Emilio Del Moral Hernandez

Entendamos ...

- as premissas da demonstração de Cybenko
- a notação não muito familiar a nós que ele usou
- o quão poderoso é o resultado que ele obteve
- como com passos simples podemos estender a sua aplicação

(ou ... *relaxando algumas das (apenas aparentes) limitações impostas nas premissas*)

© Prof. Emilio Del Moral Hernandez

Kurt Hornik showed in 1991[2] that it is not the specific choice of the a
assumed to be linear. For notational convenience, only the single out

Formal statement [edit]

The theorem[2][3][4][5] in mathematical terms:

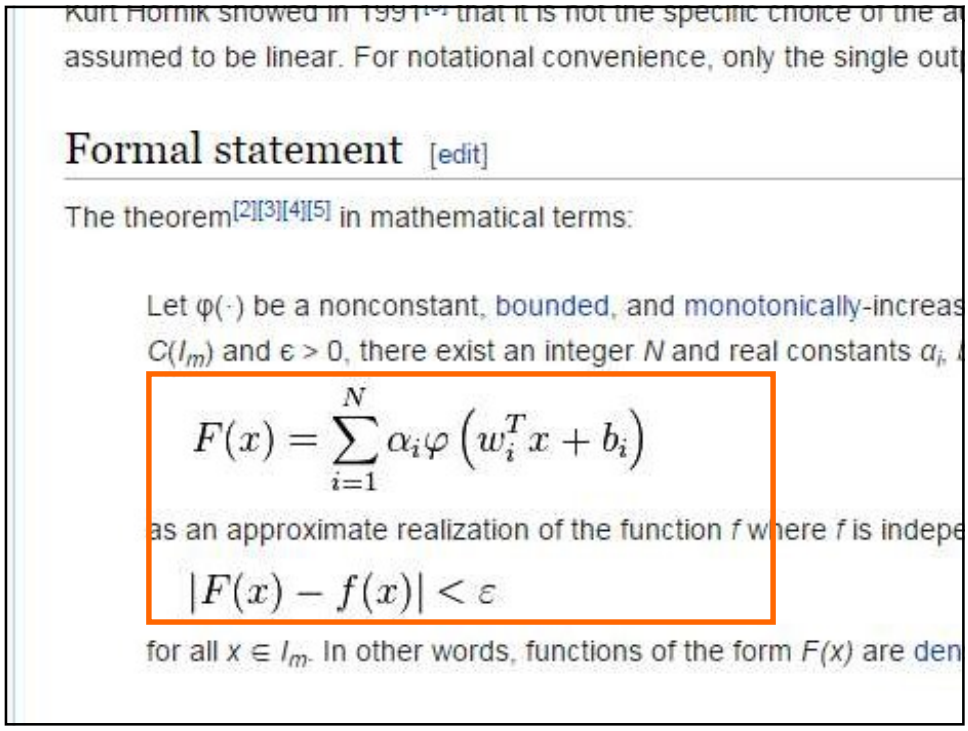
Let $\varphi(\cdot)$ be a nonconstant, bounded, and monotonically-increas
$C(I_m)$ and $\epsilon > 0$, there exist an integer N and real constants α_i,

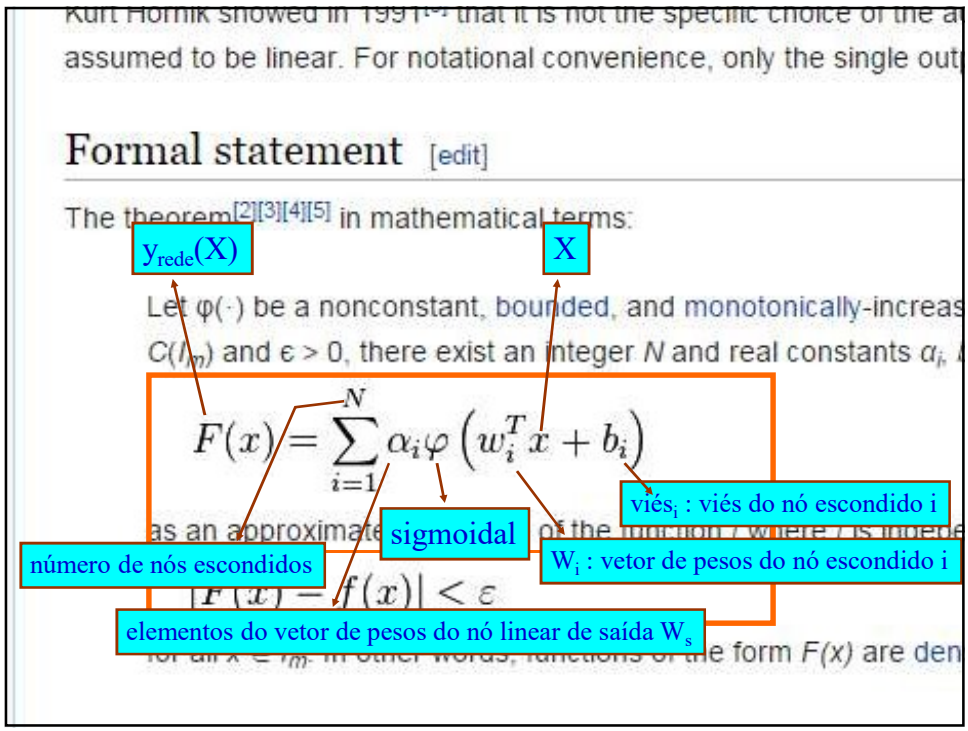
$$F(x) = \sum_{i=1}^{N} \alpha_i \varphi\left(w_i^T x + b_i\right)$$

as an approximate realization of the function f where f is indepe

$$|F(x) - f(x)| < \varepsilon$$

for all $x \in I_m$. In other words, functions of the form $F(x)$ are den


Kurt Hornik showed in 1991[2] that it is not the specific choice of the a
assumed to be linear. For notational convenience, only the single out
Formal statement [edit]
The theorem[2][3][4][5] in mathematical terms:
Let φ(·) be a nonconstant, bounded, and monotonically-increas
C(I_m) and ε > 0, there exist an integer N and real constants α_i,
$F(x) = \sum_{i=1}^{N} \alpha_i \varphi\left(w_i^T x + b_i\right)$
as an approximate realization of the function f where f is indepe
$|F(x) - f(x)| < \varepsilon$
for all x ∈ I_m. In other words, functions of the form F(x) are den


Kurt Hornik showed in 1991[2] that it is not the specific choice of the a
assumed to be linear. For notational convenience, only the single out
Formal statement [edit]
The theorem[2][3][4][5] in mathematical terms:
y_rede(X)
X
Let φ(·) be a nonconstant, bounded, and monotonically-increas
C(I_m) and ε > 0, there exist an integer N and real constants α_i,
$F(x) = \sum_{i=1}^{N} \alpha_i \varphi\left(w_i^T x + b_i\right)$
viés_i : viés do nó escondido i
sigmoidal
número de nós escondidos
W_i : vetor de pesos do nó escondido i
$|F(x) - f(x)| < \varepsilon$
elementos do vetor de pesos do nó linear de saída W_s
form F(x) are den

Kurt Hornik showed in 1991[2] that it is not the specific choice of the a

assumed to be linear. For notational convenience, only the single out

Formal statement [edit]

The theorem[2][3][4][5] in mathematical terms:

Let $\varphi(\cdot)$ be a nonconstant, bounded, and monotonically-increas

$C(I_m)$ and $\epsilon > 0$, there exist an integer N and real constants α_i,

$$F(x) = \sum_{i=1}^{N} \alpha_i \varphi\left(w_i^T x + b_i\right)$$

$y_{rede}(X)$ Fescondida_sistema(X)

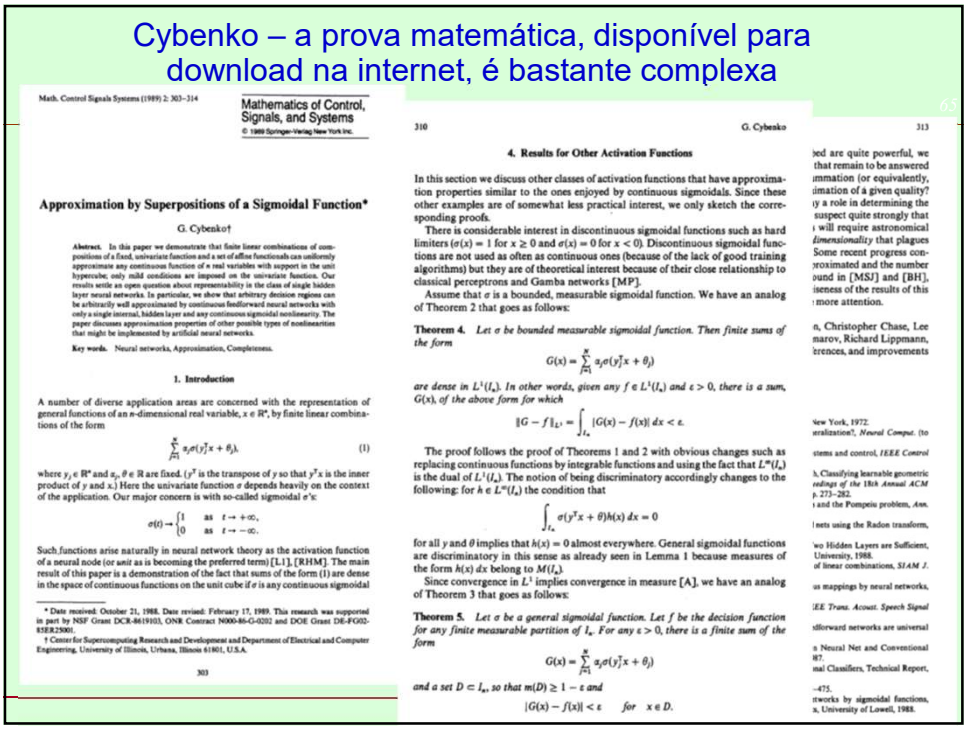
as an approximate realization of the function f where f is indepe

$$|F(x) - f(x)| < \varepsilon$$

Limite de erro

for all $x \in I_m$. In other words, functions of the form $F(x)$ are den

Cybenko – a prova matemática, disponível para download na internet, é bastante complexa



Math. Control Signals Systems (1989) 2: 303–314

Mathematics of Control, Signals, and Systems

© 1989 Springer-Verlag New York Inc.

Approximation by Superpositions of a Sigmoidal Function*

G. Cybenko†

Abstract. In this paper we demonstrate that finite linear combinations of compositions of a fixed, univariate function and a set of affine functionals can uniformly approximate any continuous function of n real variables with support in the unit hypercube; only mild conditions are imposed on the univariate function. Our results settle an open question about representability in the class of single hidden layer neural networks. In particular, we show that arbitrary decision regions can be arbitrarily well approximated by continuous feedforward neural networks with only a single internal, hidden layer and any continuous sigmoidal nonlinearity. The paper discusses approximation properties of other possible types of nonlinearities that might be implemented by artificial neural networks.

Key words. Neural networks, Approximation, Completeness.

1. Introduction

A number of diverse application areas are concerned with the representation of general functions of an n-dimensional real variable, $x \in \mathbb{R}^n$, by finite linear combinations of the form

$$\sum_{j=1}^{N} \alpha_j \sigma(y_j^T x + \theta_j), \qquad (1)$$

where $y_j \in \mathbb{R}^n$ and $\alpha_j, \theta_j \in \mathbb{R}$ are fixed. (y^T is the transpose of y so that $y^T x$ is the inner product of y and x.) Here the univariate function σ depends heavily on the context of the application. Our major concern is with so-called sigmoidal σ's:

$$\sigma(t) \to \begin{cases} 1 & \text{as } t \to +\infty, \\ 0 & \text{as } t \to -\infty. \end{cases}$$

Such functions arise naturally in neural network theory as the activation function of a neural node (or unit as is becoming the preferred term) [L1], [RHM]. The main result of this paper is a demonstration of the fact that sums of the form (1) are dense in the space of continuous functions on the unit cube if σ is any continuous sigmoidal

* Date received: October 21, 1988. Date revised: February 17, 1989. This research was supported in part by NSF Grant DCR-8619103, ONR Contract N000-86-G-0202 and DOE Grant DE-FG02-85ER25001.

† Center for Supercomputing Research and Development and Department of Electrical and Computer Engineering, University of Illinois, Urbana, Illinois 61801, U.S.A.

303

310 G. Cybenko

4. Results for Other Activation Functions

In this section we discuss other classes of activation functions that have approximation properties similar to the ones enjoyed by continuous sigmoidals. Since these other examples are of somewhat less practical interest, we only sketch the corresponding proofs.

There is considerable interest in discontinuous sigmoidal functions such as hard limiters ($\sigma(x) = 1$ for $x \geq 0$ and $\sigma(x) = 0$ for $x < 0$). Discontinuous sigmoidal functions are not used as often as continuous ones (because of the lack of good training algorithms) but they are of theoretical interest because of their close relationship to classical perceptrons and Gamba networks [MP].

Assume that σ is a bounded, measurable sigmoidal function. We have an analog of Theorem 2 that goes as follows:

Theorem 4. *Let σ be bounded measurable sigmoidal function. Then finite sums of the form*

$$G(x) = \sum_{j=1}^{N} \alpha_j \sigma(y_j^T x + \theta_j)$$

are dense in $L^1(I_n)$. In other words, given any $f \in L^1(I_n)$ and $\varepsilon > 0$, there is a sum, $G(x)$, of the above form for which

$$\|G - f\|_{L^1} = \int_{I_n} |G(x) - f(x)|\, dx < \varepsilon.$$

The proof follows the proof of Theorems 1 and 2 with obvious changes such as replacing continuous functions by integrable functions and using the fact that $L^\infty(I_n)$ is the dual of $L^1(I_n)$. The notion of being discriminatory accordingly changes to the following: for $h \in L^\infty(I_n)$ the condition that

$$\int_{I_n} \sigma(y^T x + \theta) h(x)\, dx = 0$$

for all y and θ implies that $h(x) = 0$ almost everywhere. General sigmoidal functions are discriminatory in this sense as already seen in Lemma 1 because measures of the form $h(x)\, dx$ belong to $M(I_n)$.

Since convergence in L^1 implies convergence in measure [A], we have an analog of Theorem 3 that goes as follows:

Theorem 5. *Let σ be a general sigmoidal function. Let f be the decision function for any finite measurable partition of I_n. For any $\varepsilon > 0$, there is a finite sum of the form*

$$G(x) = \sum_{j=1}^{N} \alpha_j \sigma(y_j^T x + \theta_j)$$

and a set $D \subset I_n$, so that $m(D) \geq 1 - \varepsilon$ and

$$|G(x) - f(x)| < \varepsilon \quad \text{for} \quad x \in D.$$

313

ed are quite powerful, we that remain to be answered mmation (or equivalently, imation of a given quality? y a role in determining the suspect quite strongly that will require astronomical *dimensionality* that plagues Some recent progress con- proximated and the number ound in [MSJ] and [BH], iseness of the results of this more attention.

n, Christopher Chase, Lee marov, Richard Lippmann, erences, and improvements

New York, 1972.
eralization?, *Neural Comput.* (to

stems and control, *IEEE Control*

), Classifying learnable geometric *eedings of the 18th Annual ACM* p. 273–282.
and the Pompeiu problem, *Ann.*

nets using the Radon transform,

wo Hidden Layers are Sufficient, University, 1988.
of linear combinations, *SIAM J.*

us mappings by neural networks,

IEE Trans. Acoust. Speech Signal

dforward networks are universal

n Neural Net and Conventional 87.
nal Classifiers, Technical Report,

–475.
tworks by sigmoidal functions, s, University of Lowell, 1988.