

Prof. Emilio Del Moral Hernandez
Escola Politécnica da Universidade de São Paulo
Departamento de Eng. De Sistemas Eletrônicos



emilio_del_moral@ieee.org
www.lsi.usp.br/~emilio

2

Reconhecimento de Padrões, Modelagem, Inteligência Computacional – Prof. Emilio Del Moral H. – ©

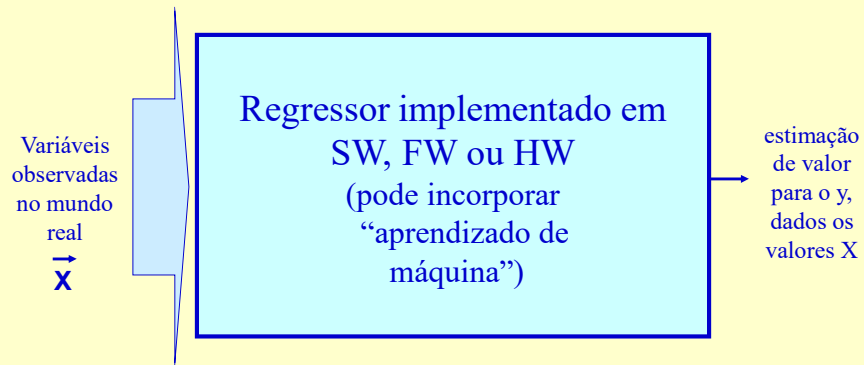
Alguns dos alvos nas temáticas / práticas / projetos deste curso e em outros que ministro

- **Modelagem envolvendo sistemas com múltiplas variáveis**
- **Uso de ferramentas que possibilitem representação de fenômenos não lineares (além dos lineares)**
- **Reconhecimento / Identificação / Classificação de “objetos” a partir de medidas múltiplas (vindas de múltiplos sensores, por exemplo)**
- **Regressão / previsão / estimação de grandezas analógicas a partir de medidas correlacionadas com tal grandeza**
- **Exploração de elementos para “automação parcial” da modelagem, via aprendizado de máquina**

3

Reconhecimento de Padrões, Modelagem, Inteligência Computacional – Prof. Emilio Del Moral H. – ©

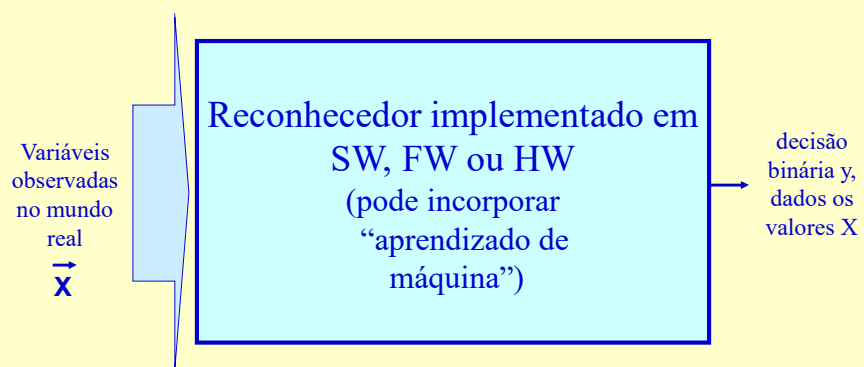
Regressor multivariado (em sw, fw ou hw)



4

Reconhecimento de Padrões, Modelagem, Inteligência Computacional – Prof. Emilio Del Moral H. – ©

Detector / reconhecedor multivariado (em sw, fw ou hw)

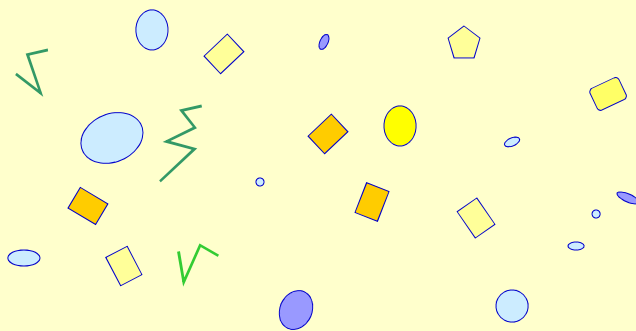


5

Reconhecimento de Padrões, Modelagem, Inteligência Computacional – Prof. Emilio Del Moral H. – ©

Uma metáfora para a tarefa de classificação de padrões genérica ou multiclases (classificação não binária)...

Universo de objetos observados ...

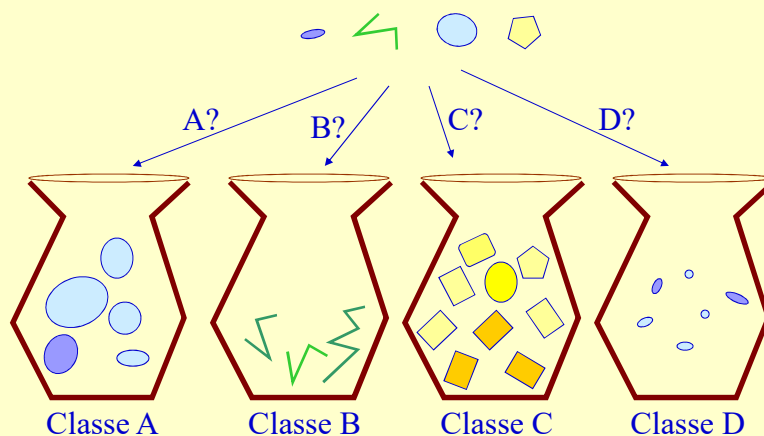


6

Reconhecimento de Padrões, Modelagem, Inteligência Computacional – Prof. Emilio Del Moral H. – ©

Uma metáfora para a tarefa de classificação de padrões genérica ou multiclases (classificação não binária)...

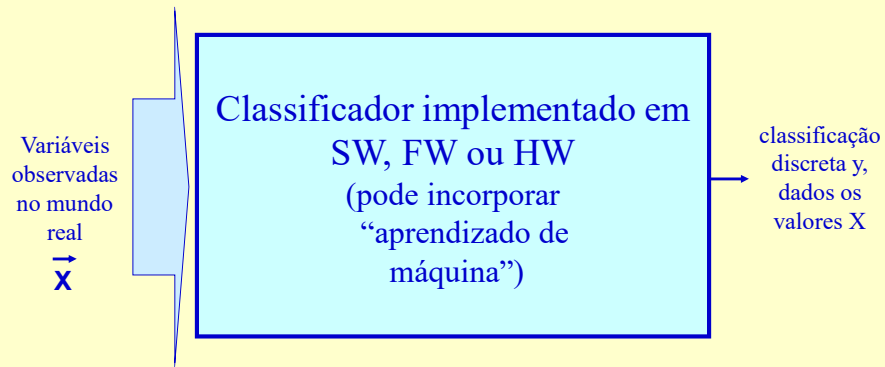
Um dado objeto específico observado é de que tipo ?



7

Reconhecimento de Padrões, Modelagem, Inteligência Computacional – Prof. Emilio Del Moral H. – ©

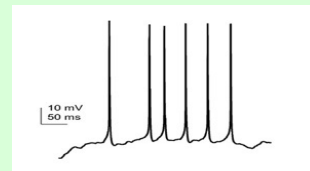
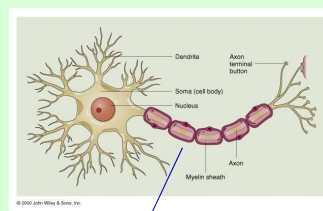
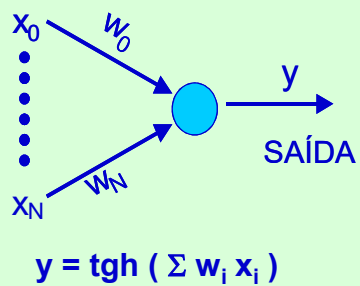
Classificador multivariado (em sw, fw ou hw)



8

Reconhecimento de Padrões, Modelagem, Inteligência Computacional – Prof. Emilio Del Moral H. – ©

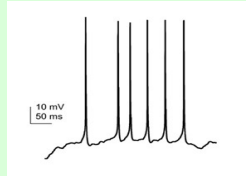
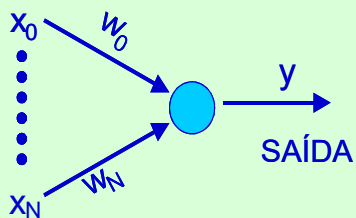
Neurônio Artificial - inspirado no biológico (ou também chamado de “nó” da rede neural)



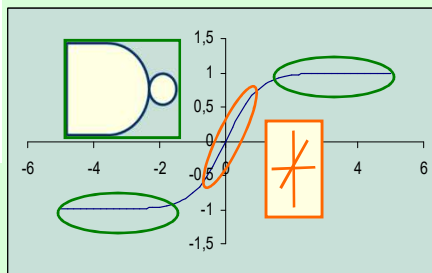
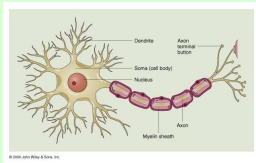
© Prof. Emilio Del Moral Hernandez

Computação linear e não linear, com codificação frequencial

10



$$y = f_T(\sum w_i x_i)$$

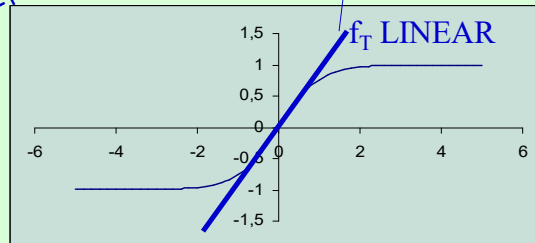
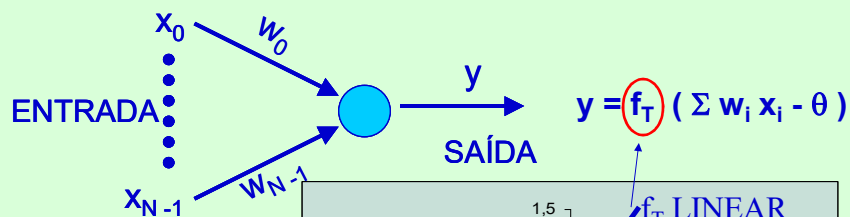


© Prof. Emilio Del Moral Hernandez

11

Caso particular do nó com comportamento linear ... $f_T(\arg) = k \cdot \arg$

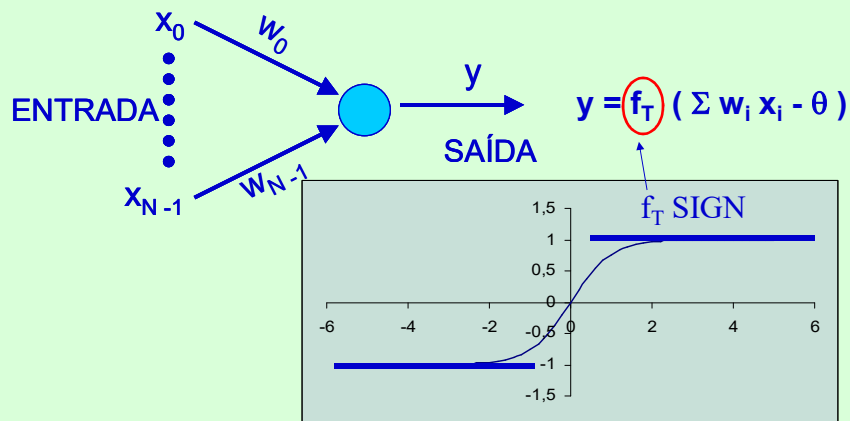
11



© Prof. Emilio Del Moral Hernandez

12

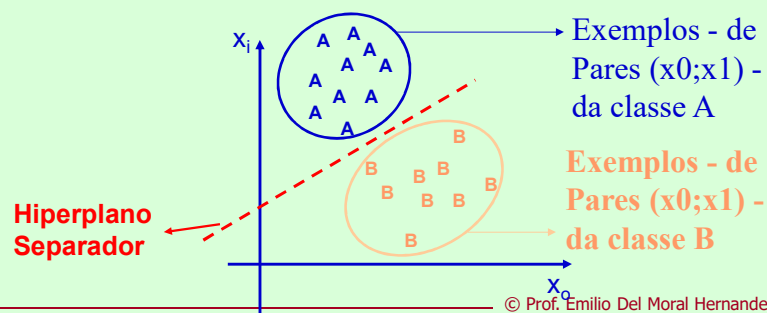
Caso particular do nó com comportamento binário ... $f_T(\text{arg}) = \text{sign}(\text{arg})$



© Prof. Emilio Del Moral Hernandez

Um pioneiro ... o Perceptron: $y = \text{sign}(\sum w_i x_i - \theta)$ (função de transferência tipo “degrau”)

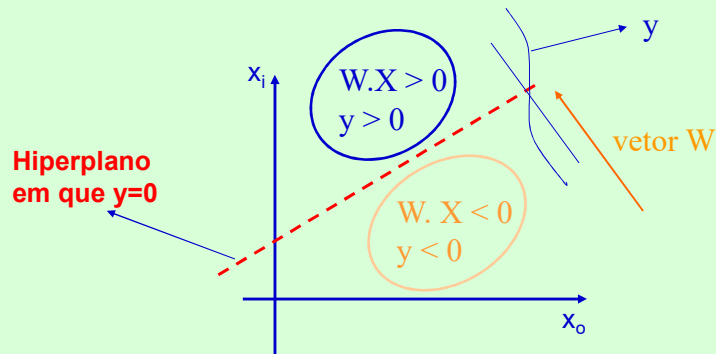
- Viabiliza a classificação de padrões com separabilidade linear
- O algoritmo de aprendizado adapta os Ws de forma a encontrar o hiperplano de separação adequado
- Aprendizado por conjunto de treinamento



© Prof. Emilio Del Moral Hernandez

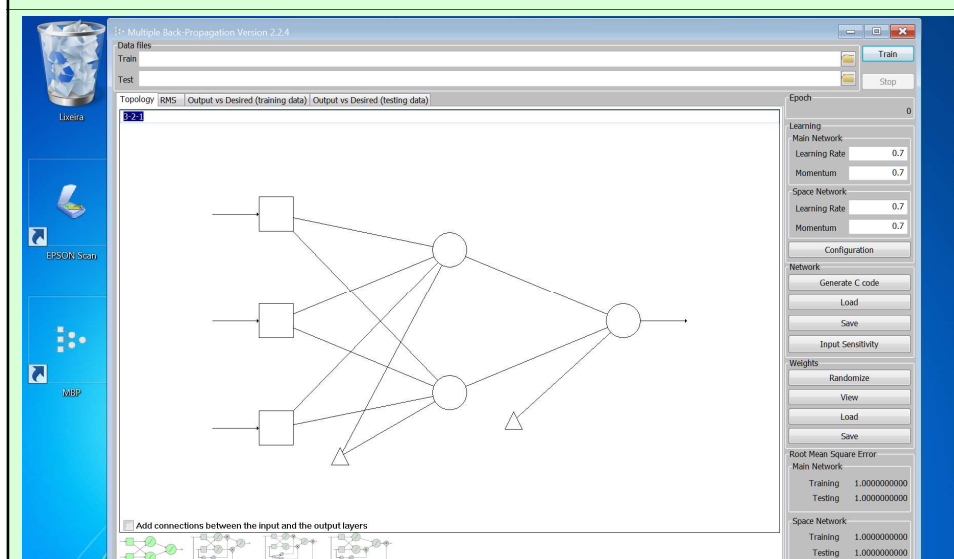
E se a saída do nosso problema não for digital? O "Perceptron Contínuo": $y = \text{tgh}(\sum w_i x_i - \theta)$

- Que problemas de entradas contínuas conseguimos atacar usando uma função de transferência tangente hiperbólica)



© Prof. Emilio Del Moral Hernandez

Indo além do neurônio "camaleão" simples Uma rede neural com 3 nós neurais ...



© Prof. Emilio Del Moral Hernandez

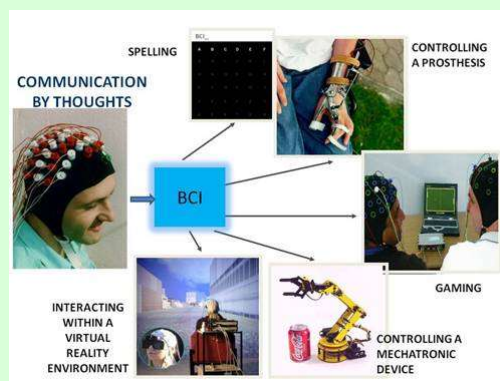
*Alguns exemplos de
sistemas de
reconhecimento
e de
regressão
multivariada, com os
quais tive contato e que
empregaram redes neurais
e técnicas relacionadas*

19

PSI2672 – Rec Padrões, Modelagem, Redes Neurais – Prof. Emilio Del Moral Hernandez – © 2016

Brain Computer Interfaces ... “um amplo
parênteses” ...

17

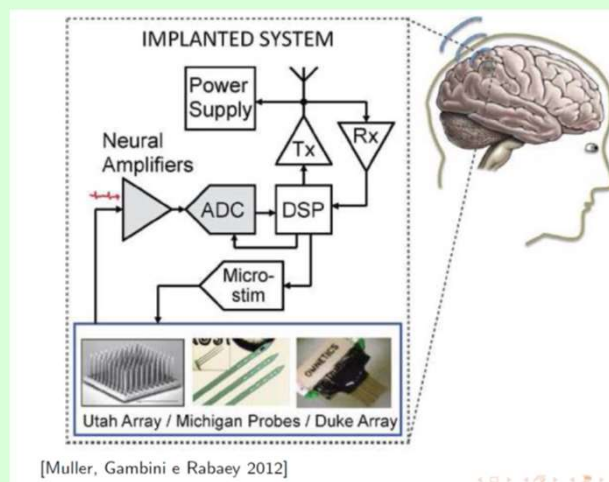


© Prof. Emilio Del Moral Hernandez

17

Brain Computer Interfaces – área do trabalho do doutorando Julio Cesar Saldaña

18



Slide - Contribuição de Julio Cesar Saldaña - EPUSP

© Prof. Emilio Del Moral Hernandez

19

Sistemas Implantáveis de Registro Neural e BCIs

19

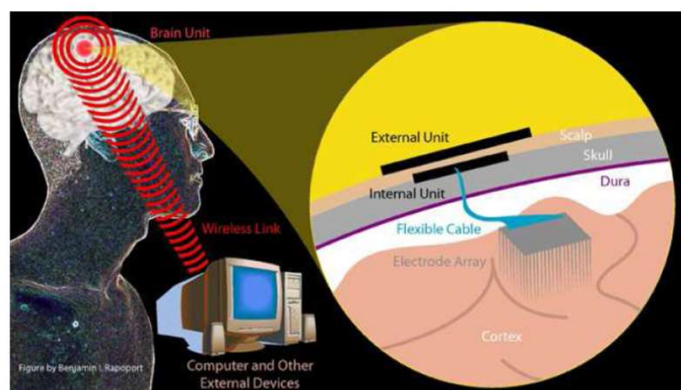


Figura 1: Imagem extraída do site do pesquisador Benjamin I. Rapoport

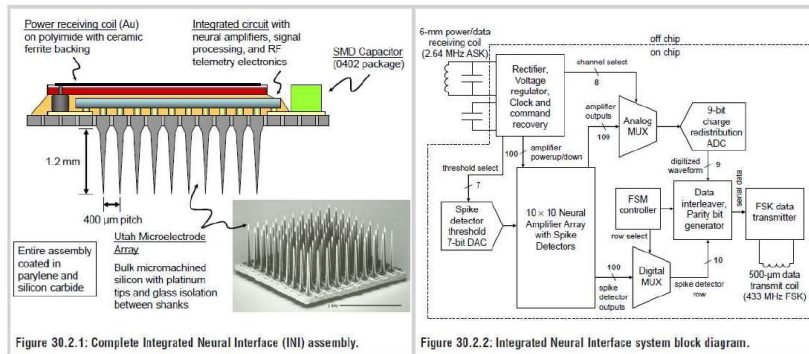
Slide - Contribuição de Julio Cesar Saldaña - EPUSP

© Prof. Emilio Del Moral Hernandez

10

Sistemas Implantáveis de Registro Neural e BCIs

20



[Harrison et al. 2006], [Harrison et al. 2007]

Slide - Contribuição de Julio Cesar Saldaña - EPUSP

© Prof. Emilio Del Moral Hernandez

21

Brain Computer Interfaces

21

Separação de impulsos elétricos ou Spike Sorting

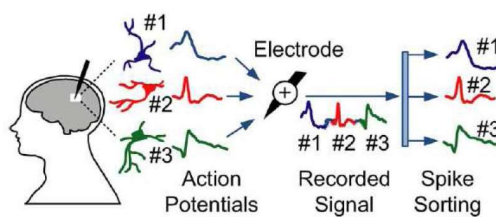


Figura 6: Extraída do artigo [Karkare, Gibson e Markovic 2013] (JSSC, Set. 2013)

Slide - Contribuição de Julio Cesar Saldaña - EPUSP

© Prof. Emilio Del Moral Hernandez

22

Alguns exemplos de projetos de 5º ano em PSI2672 da EC2 de anos anteriores (2011 a 2017); No 5o ano da EC3, atividades de projeto similares ocorrem na eletiva PSI3571.

Exemplos de projetos concebidos e realizados por alunos da disciplina EC2 PSI-2672 (na EC3, ~ PSI-3571)

The screenshot shows a web browser window with the URL <http://www.lis.usp.br/icone/psi2672/>. The page title is "PSI 2672 - Práticas em Reconhecimento de Padrões, Modelagem e Neurocomputação". On the left, there is a sidebar with links: "material", "bibliografia", "exemplos", and "2011". The main content area is titled "Projetos finais:" and lists seven projects with links to their respective PDF presentations:

1. Classificação em gêneros musicais ([gr_I_apresentacao_Musicas.pdf](#))
2. Implementação de um sistema para correção da medida de um sensor de pressão diferencial ([gr_II_Apresentacao_rna_sensor_pressao_03_01.pdf](#))
3. Análise da qualidade dos vinhos a partir de testes físicos-químicos ([gr_III_apresentacao_PSI2672 - Análise de vinhos.pdf](#))
4. Reconhecimento de placas de trânsito ([gr_IV_apresentacao_Reconhecimento_de_placas.pdf](#))
5. Reconhecedor de alcoolismo ([gr_V_apresentacao_Reconhecedor Alcoolismo.pdf](#))
6. Língua eletrônica ([gr_VI_apresentacao_lingua_eletronica.pdf](#))
7. Reconhecimento de dígitos ([gr_VII_apresentacao_Reconhecedor de digitos.pdf](#))

At the bottom, there is a link for "Informações sobre o programa do curso: [Programa PSI2672_2011.pdf](#)".

Materials diversos que podem ser uteis para seu aprendizado - Rascunhos do prof Emilio

Restrito Disponível se: Você faz parte de T-PSI3501-2018201

Repositório - Exemplos de sistemas (aproximadamente 50) para o reconhecimento ou regressão multivariada usando redes neurais, implementados por aluno de graduação em disciplinas do PSI (zipado de 36MB)

O que você deseja fazer com 333-Arquivo com slides de 50 Projetos em PSI2672 2011-2016 (1).zip (35.2 MB)?
De: edisciplinas.usp.br

333-Arquivo com slides de 50 Projetos em PSI2672 2011-2016

Início Compartilhar Exibir

Copiar Colar Recortar Copiar caminho Colar atalho Mover para Copiar para Excluir Renomear Nova pasta Novo item Fácil acesso Propriedades Editar Histórico Seleccionar Limpar sele Inverter sele

Área de Transferência Organizar Novo Abrir Seleccionar

« Emilio_2018 » PSI00-3501_2018-comMiguelArjona » StudioFiltrado3501 » 333-Arquivo com slides de 50 Projetos em PSI

Nome	Data de mo...	Tipo	Tamanho
2011-PSI2672 - Apresentacoes finais - 7 Projetos	08/08/2018 ...	Pasta compactada	6.014 KB
2012-PSI2672 - Apresentacoes finais - 6 Projetos	08/08/2018 ...	Pasta compactada	3.428 KB
2013-PSI2672 - Apresentacoes finais - 7 Projetos	08/08/2018 ...	Pasta compactada	2.799 KB
2014-PSI2672 - Apresentacoes finais - 7 Projetos	08/08/2018 ...	Pasta compactada	6.761 KB
2015-PSI2672 - Apresentacoes finais - 12 Projetos	08/08/2018 ...	Pasta compactada	10.561 KB
2016-PSI2672 - Apresentacoes finais - 10 Projetos	08/08/2018 ...	Pasta compactada	6.510 KB
Inventario com Projetos PSI 2672 de 2011 ate 2016	08/08/2018 ...	Adobe Acrobat Docu...	11 KB

Classificação automática de generos musicais

27

DESCRIÇÃO DO PROBLEMA

Categorizar músicas em gêneros automaticamente baseado em trechos de suas gravações e exemplos rotulados *a priori*

A SOLUÇÃO

1) Blues
2) Classical
3) Country
4) Disco
5) Hip-Hop
6) Jazz
7) Metal
8) Pop
9) Reggae
10) Rock

projeto de alunos em PSI-2672

© Prof. Emilio Del Moral Hernandez

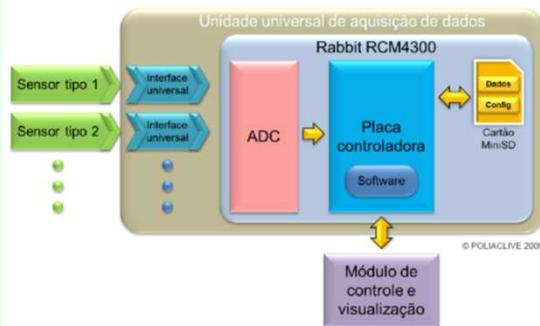
27

Sistema de fusão de sensores: por exemplo pressão e temperatura

28

Introdução

Eletrônica embarcada



projeto de alunos em PSI-2672

© Prof. Emilio Del Moral Hernandez

29

Língua Eletrônica para controle de qualidade alimentar e detecção de substâncias nocivas

29



1. Gordura
2. Proteína
3. Lactose
4. pH

- Sistema com 4 sensores;
- Sistema com 5 frequências diferentes em cada sensor;
 - $4 \times 5 = 20$ entradas para a RNA

projeto de alunos em PSI-2672

© Prof. Emilio Del Moral Hernandez

29

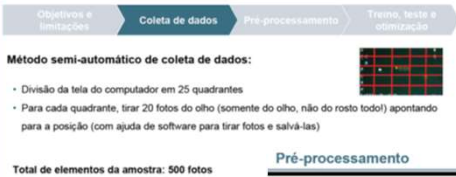
Seguidor do alvo dos olhos na tela do computador

Eye-tracker – Definição do problema

"Utilizar técnicas neurais para determinar a posição (x,y) para a qual os olhos do usuário estão apontados na tela"



Coleta de dados



Método semi-automático de coleta de dados:

- Divisão da tela do computador em 25 quadrantes
- Para cada quadrante, tirar 20 fotos do olho (somente do olho, não do rosto todo!) apontando para a posição (com ajuda de software para tirar fotos e salvá-las)

Total de elementos da amostra: 500 fotos

Pré-processamento



projeto de alunos em PSI-2672

Exemplo de modelagem / reconhecimento automático de padrões ... (tese de Liselene / Prof Miguel)

O procedimento de detecção do Vazamento

- * Haste de escuta
- * Correlacionador de ruído
- * Geofone

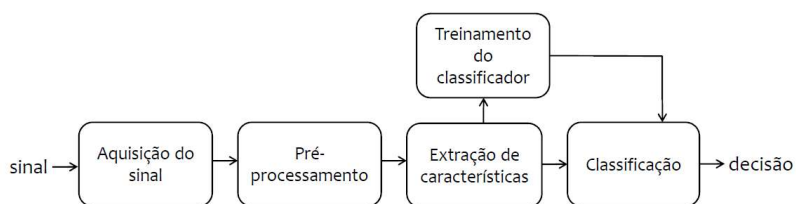


© Prof. Emilio Del Moral Hernandez

Exemplo de modelagem / reconhecimento automático de padrões ... (tese de Liselene / Prof Miguel)

32

Sistema de Detecção de Vazamento

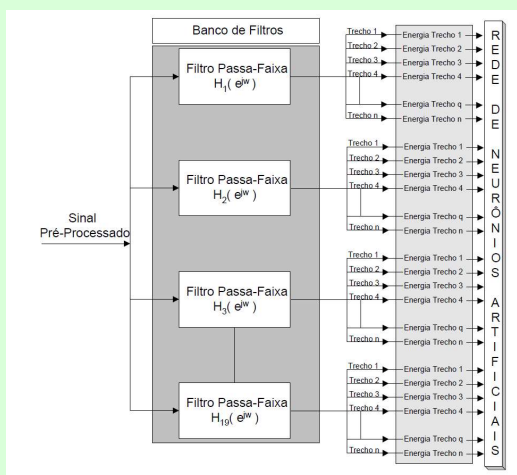


© Prof. Emilio Del Moral Hernandez

32

TCC em Reconhecimento de comandos de voz (Daniel e Gabriel) – sobe, desce, esquerda, direita

33

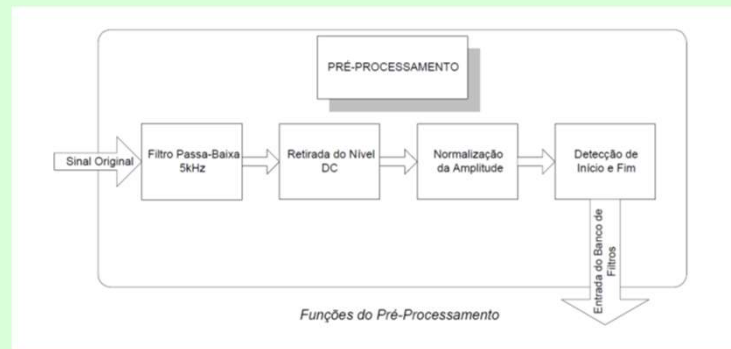


© Prof. Emilio Del Moral Hernandez

33

Exemplo de pré-processamentos num reconhecedor de comandos de voz - TCC de Daniel e Gabriel

34



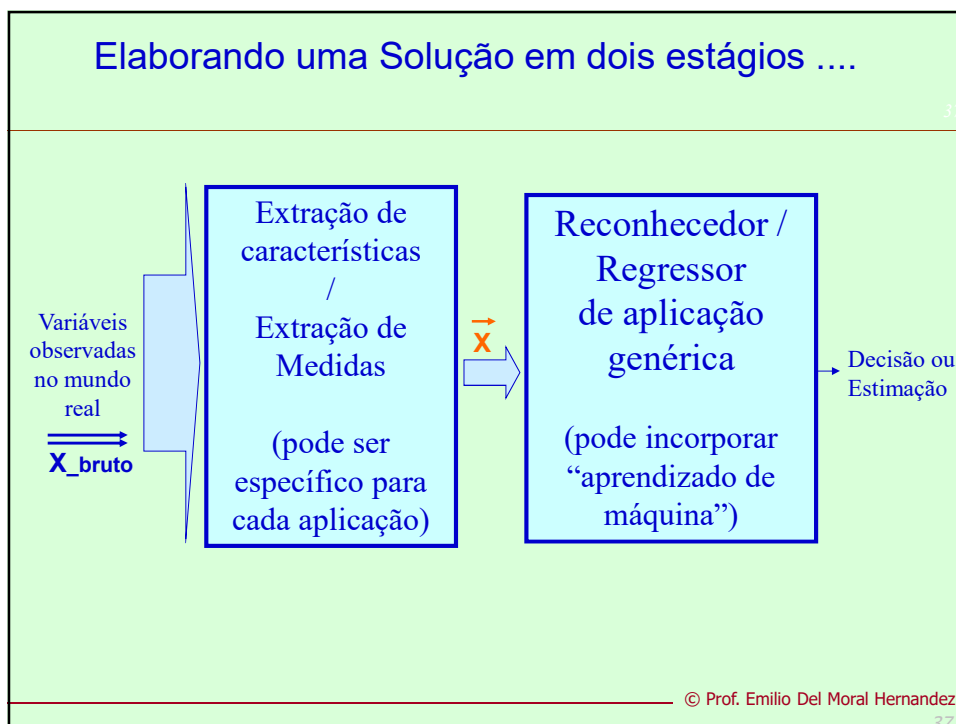
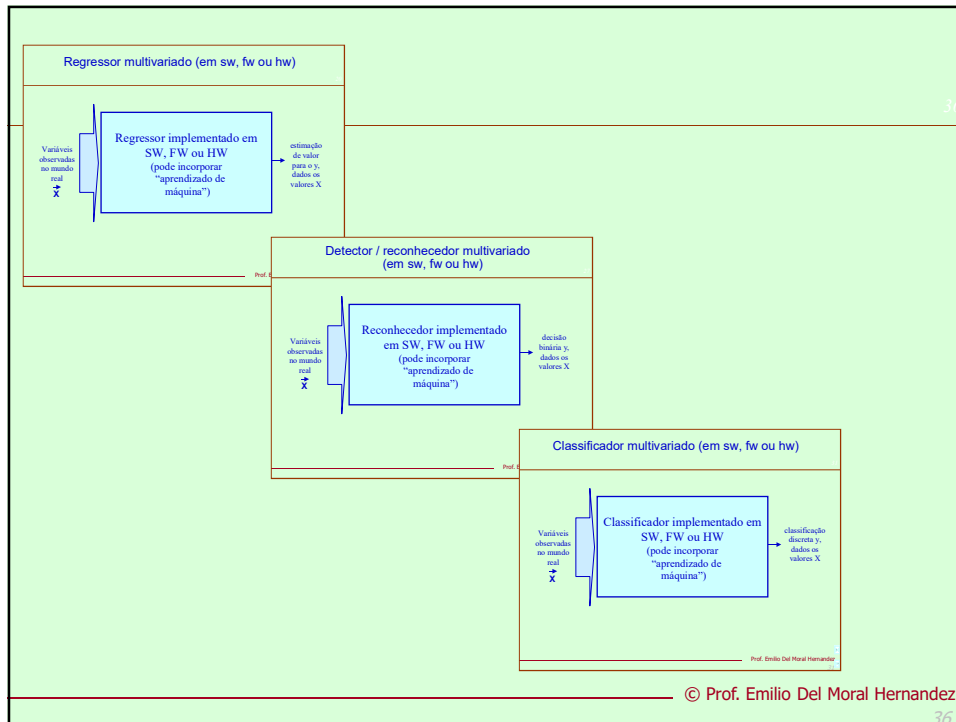
© Prof. Emilio Del Moral Hernandez

34

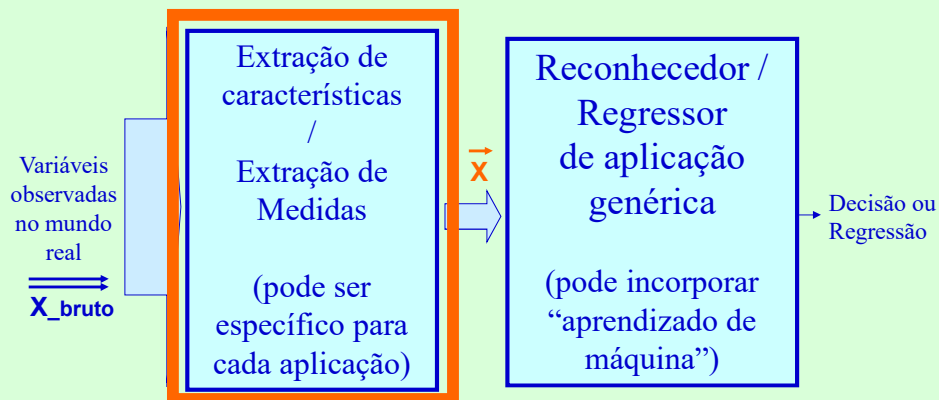
Em vários dos projetos de regressão e reconhecimento realizados (embora não em todos) houve a necessidade de uma solução envolvendo dois estágios ... Um primeiro responsável por extração de características / extração de medidas mais relevantes dos objetos "X" brutos, medidas essas normalmente específicas para cada aplicação, e um segundo estágio atuando sobre o vetor com essas diversas medidas, e que emprega alguma técnica para a regressão / identificação de padrões, independente da aplicação, como é o caso de redes neurais artificiais

© Prof. Emilio Del Moral Hernandez

35



... O 1o estágio gera um Vetor de Medidas, $\vec{X}$
(o segundo estágio operará sobre tal vetor)

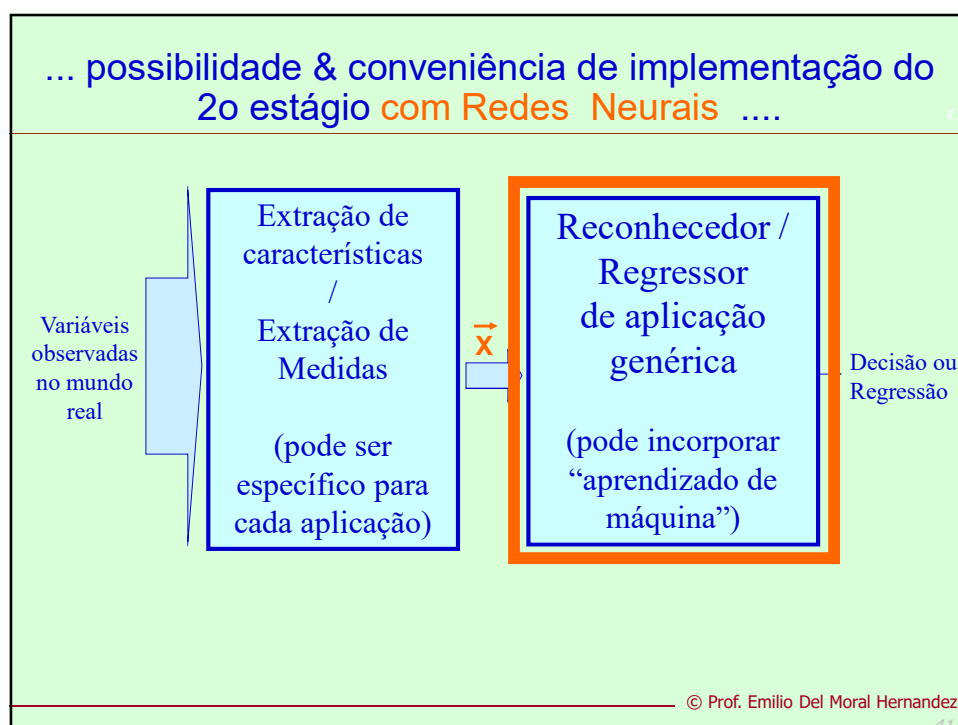
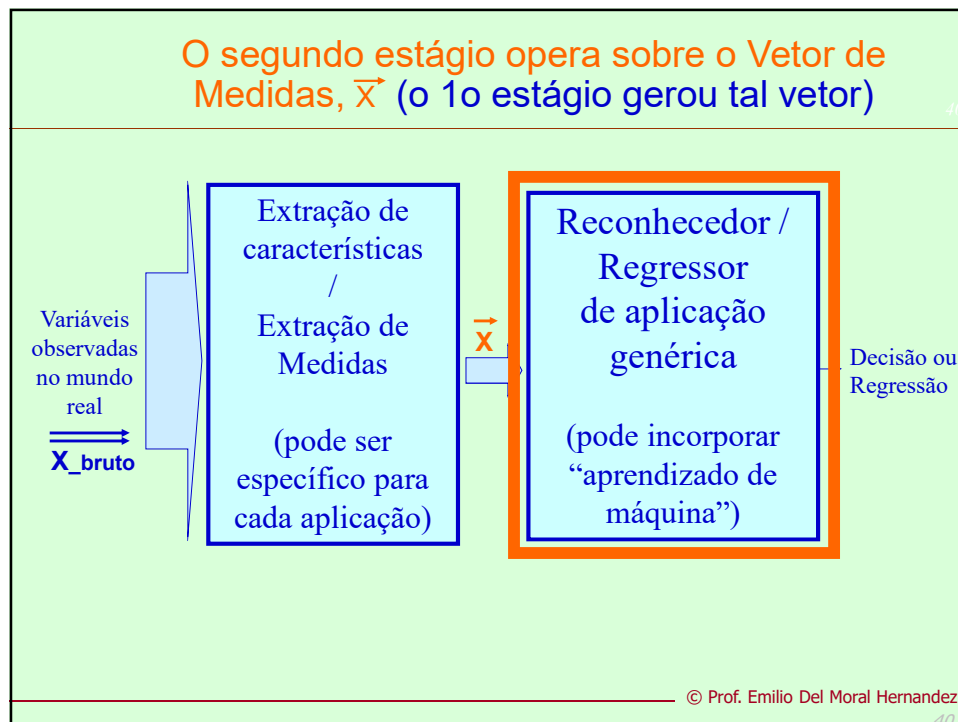


© Prof. Emilio Del Moral Hernandez

Alguns exemplos de grandezas componentes dos vetores de medidas X :

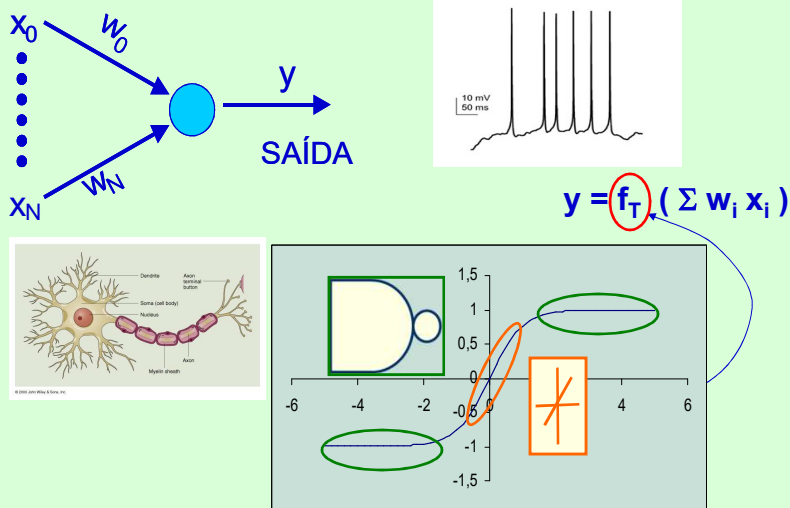
- Grandezas macroscópicas como energia do sinal, amplitude, frequência média ...
- Componentes de diversas harmônicas (análise em frequência)
- Componentes de análise tempo-frequência
- Intensidades luminosas ou intensidades em canais de cor (RGB por ex.)
- Histogramas de intensidades
- Principal Components (componentes principais – PCA)
- Medidas sobre séries temporais (médias móveis, por exemplo; medidas de dispersão / instabilidade localizadas)
- Medidas específicas à aplicação, experimentadas em problemas similares ao seu, relatadas na literatura técnica como sendo de sucesso

© Prof. Emilio Del Moral Hernandez



Computação linear e não linear, com codificação frequencial

44



© Prof. Emilio Del Moral Hernandez

44

*Falemos agora um pouco sobre
ambientes computacionais que
poderemos usar nos projetos ...*

47

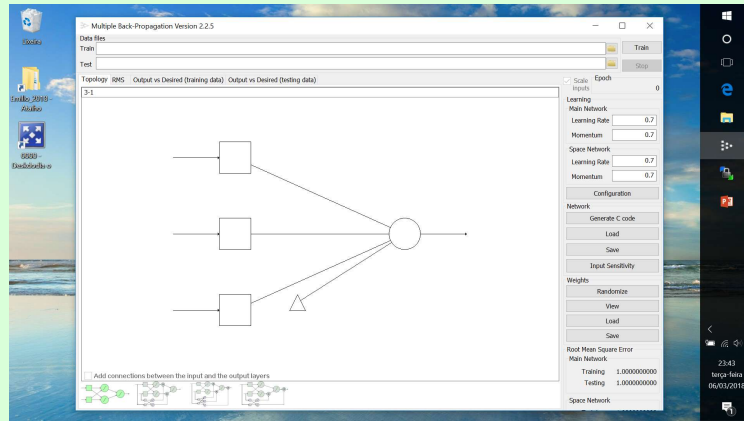
*MBP para prototipação rápida, Matlab,
etc ...*

*Me digam vocês de outros e iniciemos
uma seção no STOA para registro de
bons achados!*

47

Instale o MBP e digite "3-1" no campo Topology

48

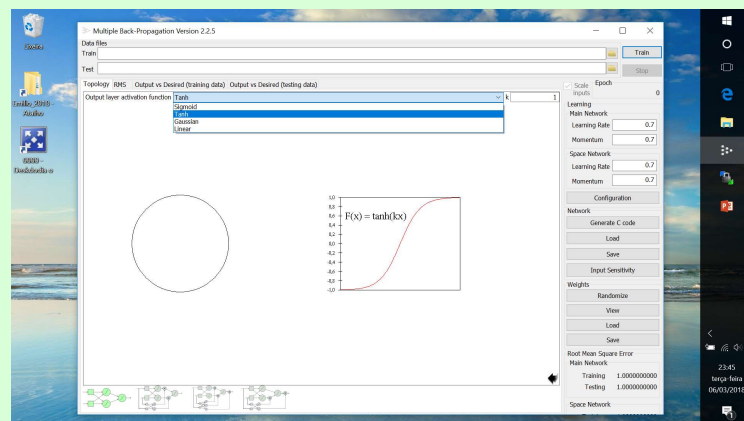


© Prof. Emilio Del Moral Hernandez

49

Depois escolha a função de ativação do nó neural

49

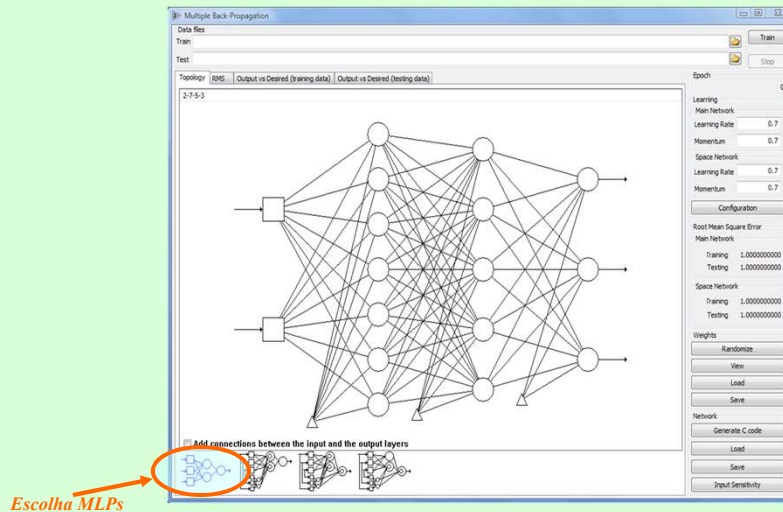


© Prof. Emilio Del Moral Hernandez

49

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

50

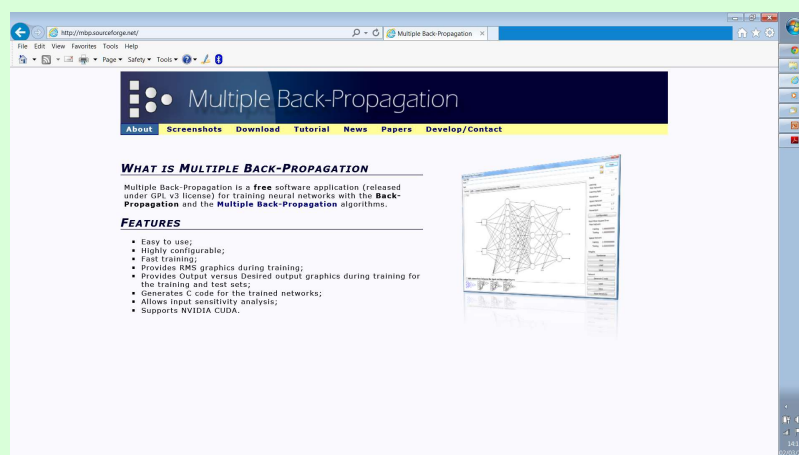


© Prof. Emilio Del Moral Hernandez

51

“Sugestão” ... visite os tutoriais do MBP - <http://mbp.sourceforge.net/> -
e instale-o no seu computador Windows.
(Na sala C1-10: o MBP deve estar instalado já no início do semestre)

51



© Prof. Emilio Del Moral Hernandez

51

MBP – uma plataforma didática para redes neurais gratuita, de fácil uso e com 12 excelentes tutoriais

52

site <http://mbp.sourceforge.net/>

TUTORIAL

1. Introduction (includes the MBP Algorithm)
2. Creating the training and the test datasets
3. Defining the topology of the neural networks
4. Configuring the activation functions of the neurons
5. Defining the neural network learning configuration
6. Training a neural network - Part I (regression)
7. Training a neural network - Part II (classification)
8. Copying data and graphics
9. Initialize, view, save and load the neural network weights
10. Load and save a neural network
11. Generate C code from a trained neural network
12. Analyzing the input sensitivity of a neural network

Ambiente desenvolvido pelo Prof. Noel Lopes e colaboradores
– Instituto Politécnico da Guarda – Portugal

© Prof. Emilio Del Moral Hernandez

52

Comece a usar o MBP em situações simples

53

Please properly cite our work if you find it useful. This supports future development. The following articles provided the foundation for Multiple Back-Propagation software (A full list of references is included with the software):

Lopes, N. and Ribeiro, B. (2010). A Strategy for Dealing With Missing Values by Using Selective Activation Neurons in a Multi-Topology Framework. IEEE World Congress on Computational Intelligence WCCI 2010.

Lopes, N. and Ribeiro, B. (2009). GPU Implementation of the Multiple Back-Propagation algorithm. In Proceedings of Intelligent Data Engineering and Automated Learning, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, volume 5784, pages 444-456.

Lopes, N. and Ribeiro, B. (2003). An Efficient Gradient-Based Learning Algorithm Applied to Neural Networks with Selective Activation Neurons. In Neural, Parallel & Scientific Computations, volume 11, pages 213-272. Dynamic Publishers.

Lopes, N. and Ribeiro, B. (2001). Hybrid learning in a multi neural network architecture. In IJCNN: IEEE International Joint Conference on Neural Networks, IJCNN), volume 4, pages 2738-2743. Washington D.C., USA.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010 Noel de Jesus Mendonça Lopes

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

This program can be freely obtained on the site <http://ftp.gnu.org/gnu/mbp/>

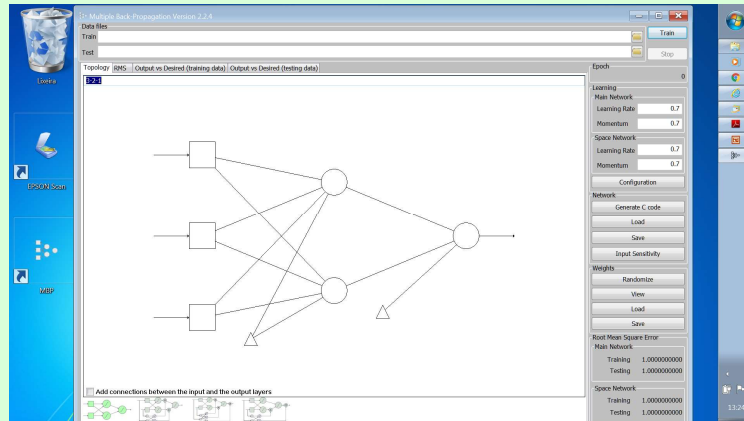
1. Add connections between the input and the output layers

© Prof. Emilio Del Moral Hernandez

53

Instale-o e digite "3-2-1" no campo Topology, ;-)

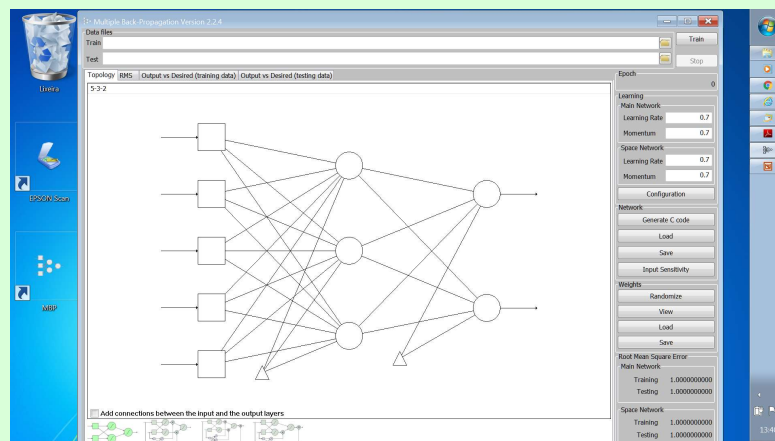
54



© Prof. Emilio Del Moral Hernandez

Instale-o e digite "5-3-2" no campo Topology, ;-)

55



© Prof. Emilio Del Moral Hernandez

12 tutoriais curtos narrados (~5 mins cada).

Abra um browser e acesse ... <http://mbp.sourceforge.net/tutorial.html>

56

TUTORIAL

1. Introduction (includes the MBP Algorithm)
2. Creating the training and the test datasets
3. Defining the topology of the neural networks
4. Configuring the activation functions of the neurons
5. Defining the neural network learning configuration
6. Training a neural network - Part I (regression)
7. Training a neural network - Part II (classification)
8. Copying data and graphics
9. Initialize, view, save and load the neural network weights
10. Load and save a neural network
11. Generate C code from a trained neural network
12. Analyzing the input sensitivity of a neural network

© Prof. Emilio Del Moral Hernandez

56

Tutorial 2 – criando 2 conjuntos empíricos, de treino e de teste

57

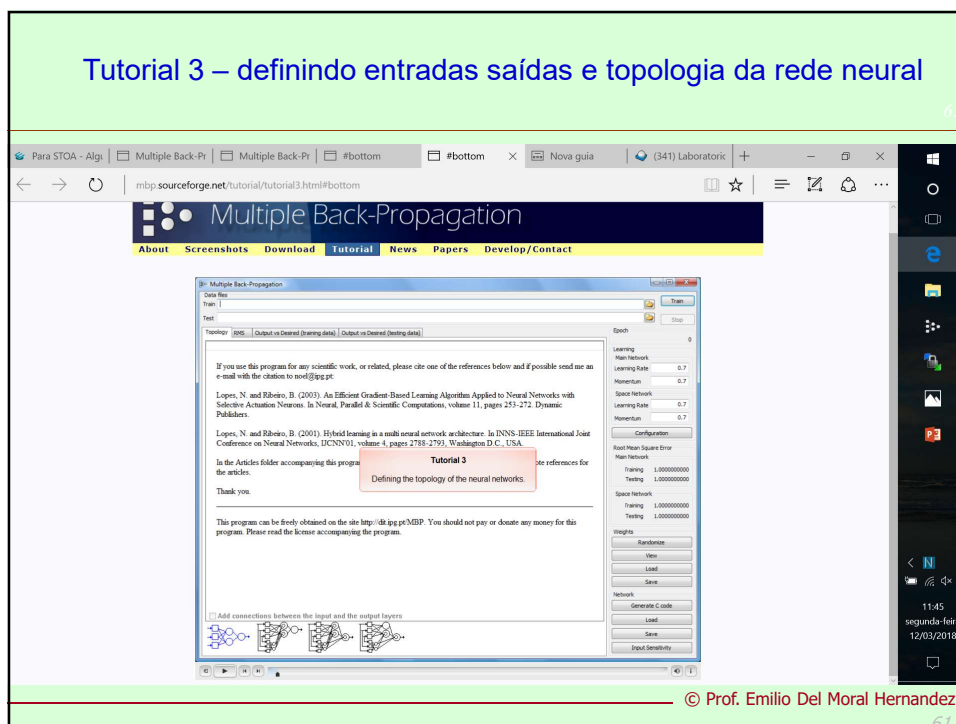
Tutorial 2
Creating the training and the test datasets.

© Prof. Emilio Del Moral Hernandez

57

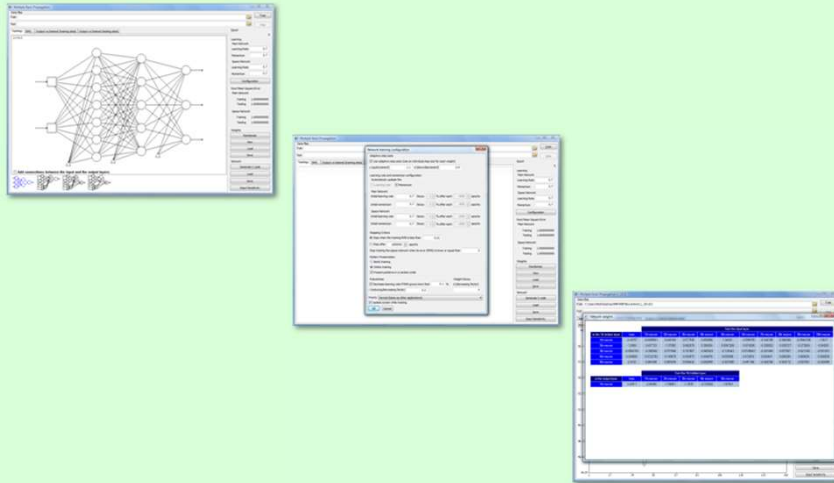


Tutorial 3 – definindo entradas saídas e topologia da rede neural



Algumas Telas do MBP

62



© Prof. Emilio Del Moral Hernandez

62

Conhecemos outros ambientes de modelagem úteis a esta disciplina além do MBP?

63

Vamos resgatar nas nossas trajetórias elementos que nos permitiriam modelagem multivariada / reconhecimento de padrões / regressão ?

Acrescentemos essas possibilidades ao seu diário de bordo do curso – um dossiê cumulativo de seus progressos e atividades, acessível ao professor.

© Prof. Emilio Del Moral Hernandez

63

Redes Neurais Artificiais

66

São: sistemas computacionais, de implementação em hardware ou software, que imitam as habilidades computacionais do sistema nervoso biológico, usando um grande número de processadores simples (neurônios artificiais) e interconectados entre si.

Emprestam da biologia:

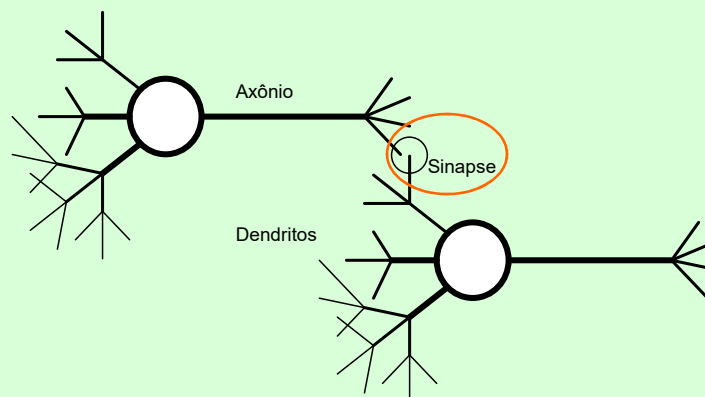
- A estrutura de processamento microscópico (processamento de informação de neurônios individuais)
- Em algum grau, aspectos da organização de redes neurais biológicas – como os neurônios se interligam
- O aprendizado através de exemplos (através de casos)

© Prof. Emilio Del Moral Hernandez

66

Cômputos mais complexos ... são realizados
pelo encadeamento de vários neurônios

67



A conexão entre um axônio de um neurônio e um dendrito de outro é denominada **Sinapse**

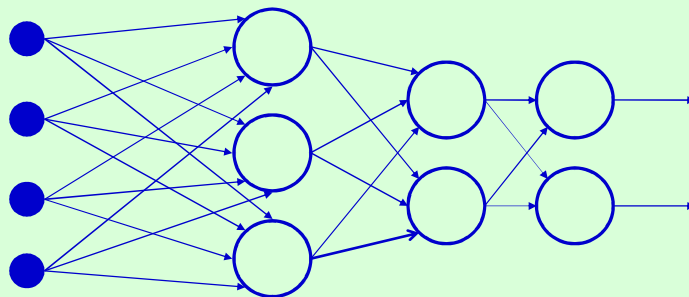
© Prof. Emilio Del Moral Hernandez

67

Foco deste Curso: o Multi Layer Perceptron (MLP)

68

- 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

68

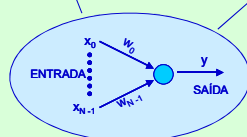
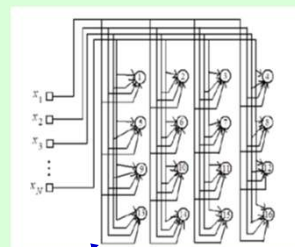
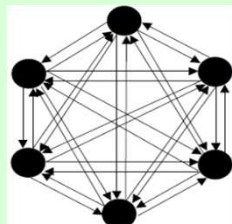
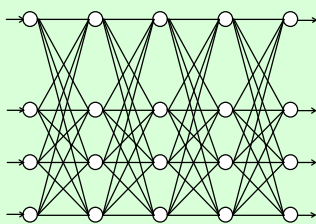
Três arquiteturas neurais importantes (abordadas em pósgrad – PSI 5886)

69

1) MLP
- Multi Layer
Perceptron

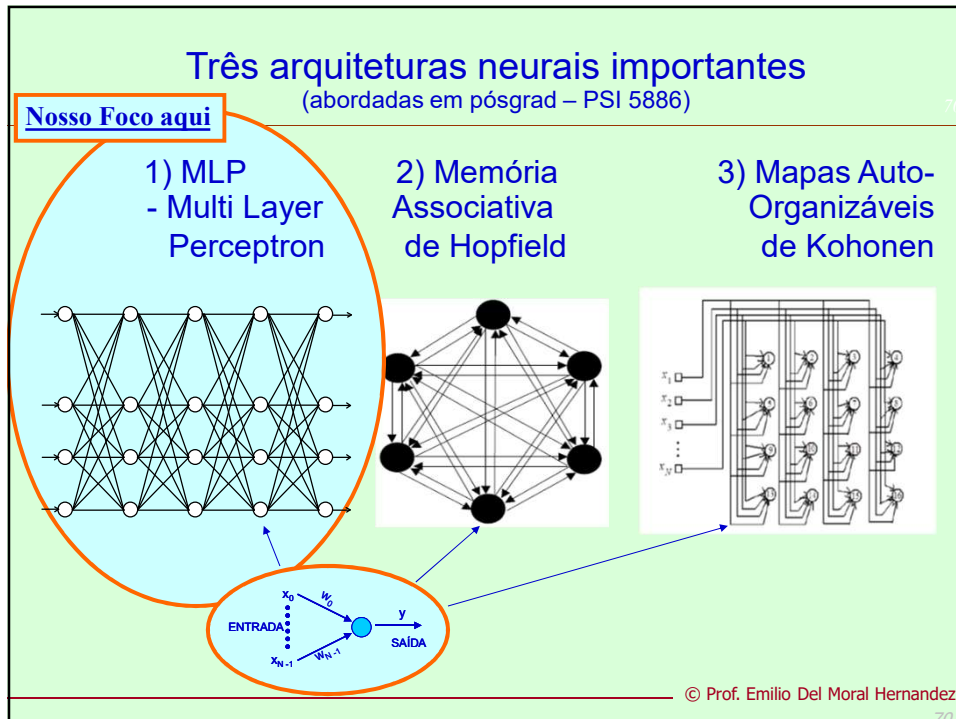
2) Memória
Associativa
de Hopfield

3) Mapas Auto-
Organizáveis
de Kohonen



© Prof. Emilio Del Moral Hernandez

69



Redes Neurais Artificiais

São: sistemas computacionais, de implementação em software, que imitam as habilidades computacionais do sistema nervoso biológico, usando um grande número de neurônios simples (neurônios artificiais) e interligados de maneira específica.

Empreendimento

A modelagem fiel da biologia está nas redes neurais?
Quanto? Onde? Quando?

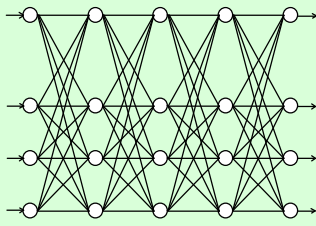
- A modelagem de processamento microscópico (processamento de informação de neurônios individuais)
- Em algum grau, aspectos da organização de redes neurais biológicas – como os neurônios se interligam
- O aprendizado através de exemplos (através de casos)

© Prof. Emilio Del Moral Hernandez

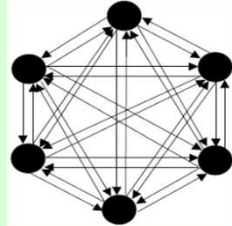
Três arquiteturas neurais importantes (abordadas em pósgrad – PSI 5886)

72

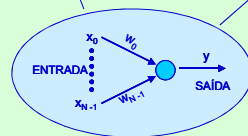
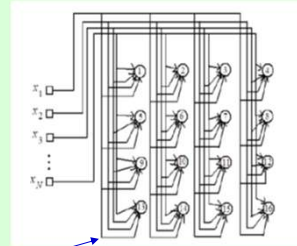
1) MLP
- Multi Layer
Perceptron



2) Memória
Associativa
de Hopfield



3) Mapas Auto-
Organizáveis
de Kohonen



© Prof. Emilio Del Moral Hernandez

73

*Como escolhemos
os valores dos
diversos w 's de um
MLP?*

73

© Prof. Emilio Del Moral Hernandez

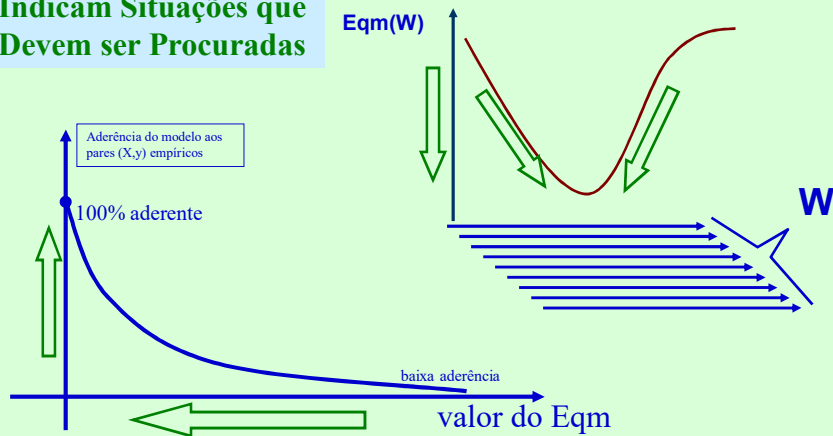
73

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

74

Devemos buscar Maximização da aderência = Mínimo Eqm possível

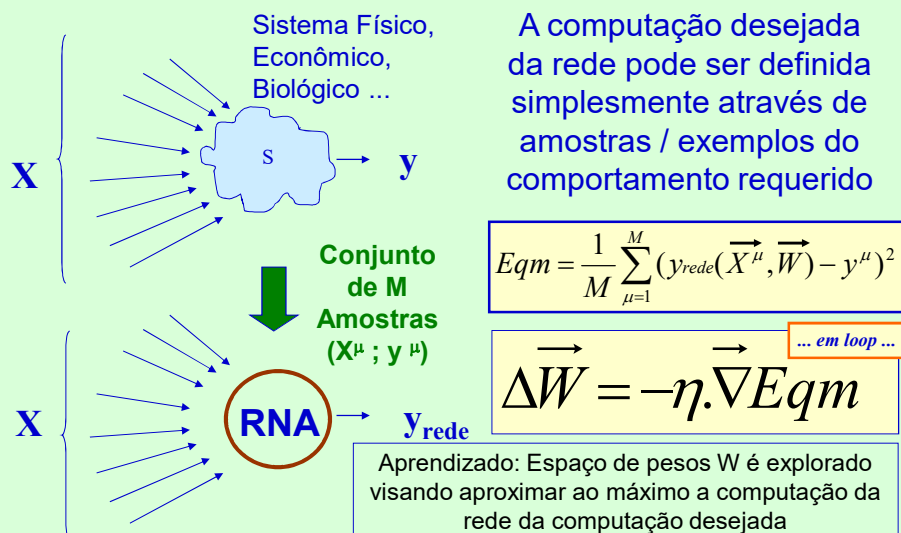
As Setas Verdes Indicam Situações que Devem ser Procuradas



74

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

75



© Prof. Emilio Del Moral Hernandez

75

Aprendizado do MLP por Error Back Propagation ...

$$\Delta \vec{W} = -\eta \cdot \vec{\nabla} E_{qm}$$

Gradiente de E_{qm} no espaço de pesos = $(\partial E_{qm}(W)/\partial w_1, \partial E_{qm}(W)/\partial w_2, \partial E_{qm}(W)/\partial w_3, \dots)$

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

76

© Prof. Emilio Del Moral – EPUSP

A modelagem fiel da biologia está nas redes neurais?
Quanto? Onde? Quando?

**Como escolhemos
os valores dos
diversos w 's de um
MLP?**

77

© Prof. Emilio Del Moral Hernandez

77

78

A modelagem fiel da biologia está nas redes neurais?
Quanto? Onde? Quando?

Como escolhemos os valores dos diversos w 's nas arquiteturas de Hopfield e Kohonen?

© Prof. Emilio Del Moral Hernandez

78

79

De onde vem o grande poder do MLP?

© Prof. Emilio Del Moral Hernandez

79

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

- 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)

© Prof. Emilio Del Moral Hernandez

80

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*)

Kurt Hornik showed in 1991^[2] that it is not the specific choice of the φ 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-increasing** function on $C(I_m)$ and $\epsilon > 0$, there exist an integer N and real constants α_i and b_i

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

as an approximate realization of the function f where f is independent of N

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

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

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

Formal statement [\[edit\]](#)

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

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

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

as an approximate realization of the function f where f is independent of N and

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

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

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

Formal statement [\[edit\]](#)

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

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

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

as an approximate realization of the function f where f is independent of N and

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

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

$y_{\text{rede}}(X)$

X

número de nós escondidos

sigmoidal

w_i : viés do nó escondido i

W_i : vetor de pesos do nó escondido i

elementos do vetor de pesos do nó linear de saída W_s

Kurt Hornik showed in 1991^[2] that it is not the specific choice of the activation function assumed to be linear. For notational convenience, only the single output is considered.

Formal statement [edit]

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

Let $\phi(\cdot)$ be a nonconstant, bounded, and monotonically-increasing function. If $C(I_m)$ and $\epsilon > 0$, there exist an integer N and real constants α_i and b_i such that

$$F(x) = \sum_{i=1}^N \alpha_i \phi(\omega_i x + b_i)$$

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

as an approximate realization of the function f where f is independent of N and ϵ .

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

Limite de erro

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

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.

310

G. Cybenko

313

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 functions 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

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

where $y_j \in \mathbb{R}^n$ and $\alpha_j, \theta_j \in \mathbb{R}$ are fixed, y_j^T is the transpose of y_j so that $y_j^T x$ is the inner product of y_j 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:

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

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 function.

* Date received: October 21, 1988. Date revised: February 17, 1989. This research was supported in part by NSF Grant DCR-8619103, ONR Contract N00014-86-G-0202 and DOE Grant DE-FG02-85ER22001.

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

303

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 Gamma 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 a 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 $\epsilon > 0$, there is a sum, $G(x)$, of the above form for which

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

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 $L^1(I_n)$ is the dual of $L^\infty(I_n)$. The notion of being discriminatory accordingly changes to the following: for $h \in L^\infty(I_n)$ the condition that

$$\int_{I_n} \sigma(y_j^T x + \theta_j) h(x) dx = 0$$

for all y_j and θ_j 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 $\epsilon > 0$, 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 $m(D) \geq 1 - \epsilon$ and

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

ed are quite powerful, we that remain to be answered approximation (or equivalently, simulation of a given quality?) (y) a role in determining the suspect quite strongly that i will require astronomical dimensionality that plagues Some recent progress con- sidered and the number found in [MSJ] and [BH], iness of the results of this more attention.

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

New York, 1972.

eralization?, *Neural Comput.* (to

tems and control, *IEEE Control*

h. Classifying learnable geometric

ndings of the 18th Annual ACM

p. 273–282.

and the Pompeiu problem, *Am.*

l nets using the Radon transform,

wo Hidden Layers are Sufficient,

University, 1988.

of linear combinations, *SIAM J.*

us mappings by neural networks,

IEEE Trans. Acoust. Speech Signal

lterward networks are universal

a Neural Net and Conventional

mal Classifiers, Technical Report,

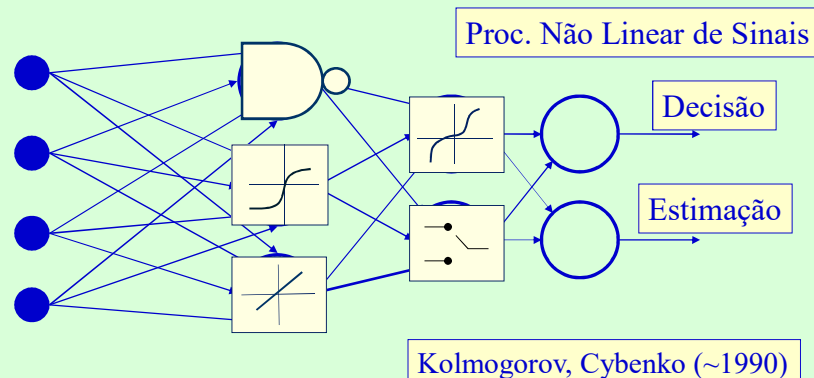
–475.

works by sigmoidal functions,

A, University of Lowell, 1988.

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

Para quem quiser avançar em direções distintas do MLP ... Pesquise os nichos de aplicação, as equações de operação entrada / saída e os algoritmos de aprendizado das Redes neurais Não MLP: Memórias associativas de Hopfield, Redes recorrentes, Mapas autoorganizados de Kohonen, Redes de RBFs, SVMs, ConvNNs,... etc etc.

© Prof. Emilio Del Moral Hernandez