

--- Aulas do Prof Emilio em PSI3471 da #1 (em 02/março) à #13 (em 29/abril) ---

Plano aproximado de temáticas aula a aula / semana a semana – versão de 01 de março
(atividades com entregas simples aproximadamente semanais ocorrerão durante as aulas)

#1 (02/março – 2ªf) Conceitos em regressão linear e não linear. Regressão univariada e Regressão multivariada; Contrastes com técnica mais simples já conhecida: a regressão linear univariada (reta média). Diferenças principais entre 1) a Regressão e 2) o Reconhecimento / Detecção / Identificação de Padrões Multivariados. Conceitos em Reconhecimento / Detecção / Identificação de Padrões. Entendendo primeiro as ferramentas de modelagem mais simples: as ferramentas lineares multivariadas ... elementos matemáticos em regressão linear multivariada e em discriminadores lineares multivariados. Dados empíricos para calibração de parâmetros em regressores e em discriminadores; o aprendizado supervisionado.

#2 (04/março – 4ªf) Neurônios biológicos como base para os neurônios artificiais; equações e propriedades matemáticas do neurônio artificial isolado, relação do neurônio isolado com os regressores e com os discriminadores lineares multivariados.

#3 (09/março – 2ªf) Foco nas Redes de Neurônios Artificiais (RNAs) concatenando múltiplos neurônios; arquiteturas neurais diversas; arquiteturas MLP – Multi Layer Perceptron; Relação de RNAs com a implementação de regressores e reconhedores **NÃO LINEARES MULTIVARIADOS**. O ensaio computacional de redes neurais – ambientes simples como o MBP – Multiple Back Propagation – e ambientes mais complexos alternativos, de sua escolha.

#4 (11/março – 4ªf) Grafos da computação em redes neurais do tipo MLP; notação matemática para pesos sinápticos, variáveis de entrada, saída e variáveis intermediárias no MBP; exemplares empíricos / observações empíricas para o treinamento supervisionado em MLPs; aprendizado com base na propagação reversa do erro (EBP – Error Back Propagation).

© Prof. Emilio Del Moral Hernandez

3

Computação linear e não linear, com codificação freqüencial

SAÍDA

$y = f_T(\sum w_i x_i)$

Saturação

(Freq. Sat.)

estímulos sublimiar

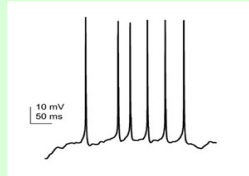
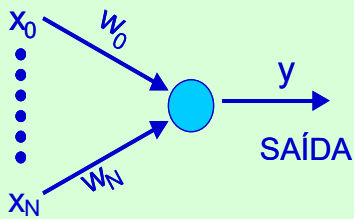
Volume de Estímulo

© Prof. Emilio Del Moral Hernandez

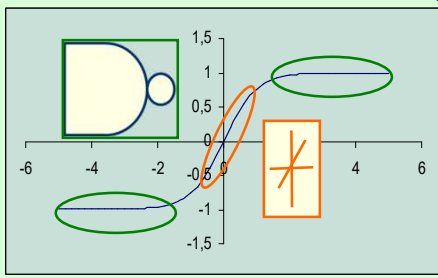
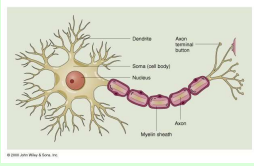
10

Computação linear e não linear, com codificação frequencial

11



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



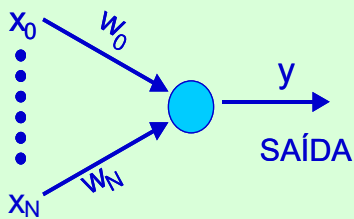
© Prof. Emilio Del Moral Hernandez

11

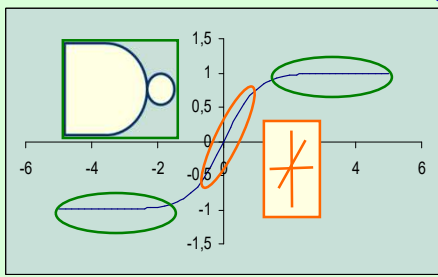
11

Computação linear e não linear, com codificação frequencial

12



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



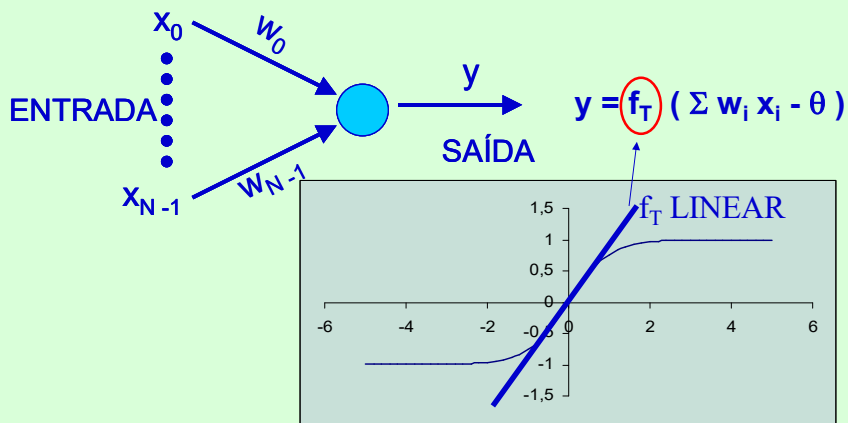
© Prof. Emilio Del Moral Hernandez

12

12

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

13



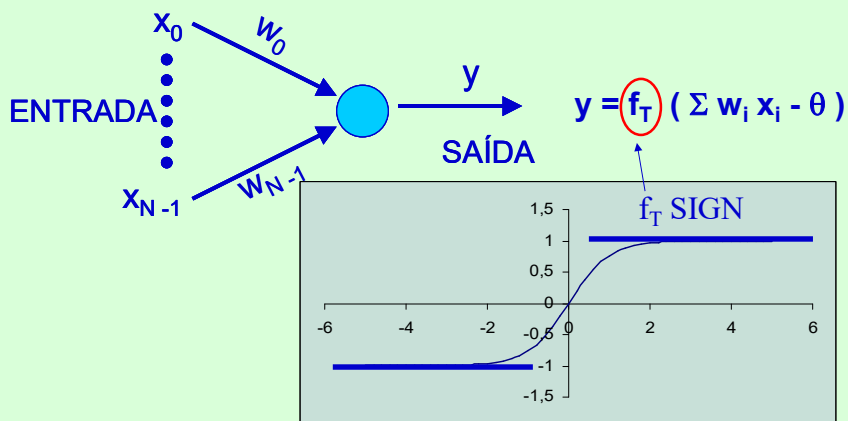
© Prof. Emilio Del Moral Hernandez

13

13

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

14



© Prof. Emilio Del Moral Hernandez

14

14

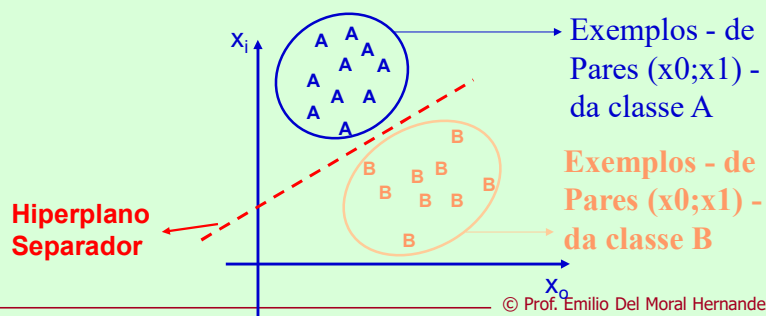
Um pioneiro ... o Perceptron de Roseblatt:

$$y = \text{sinal}(\sum w_i x_i - \theta)$$

(função de ativação tipo "degrau")

16

- 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



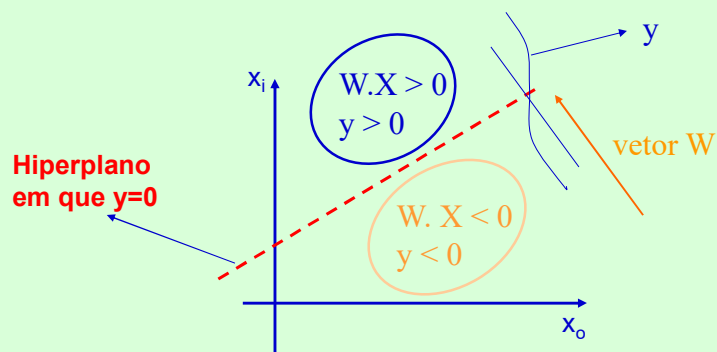
16

E se a saída do nosso problema não for digital?

O "Perceptron Contínuo": $y = \text{tgh}(\sum w_i x_i - \theta)$

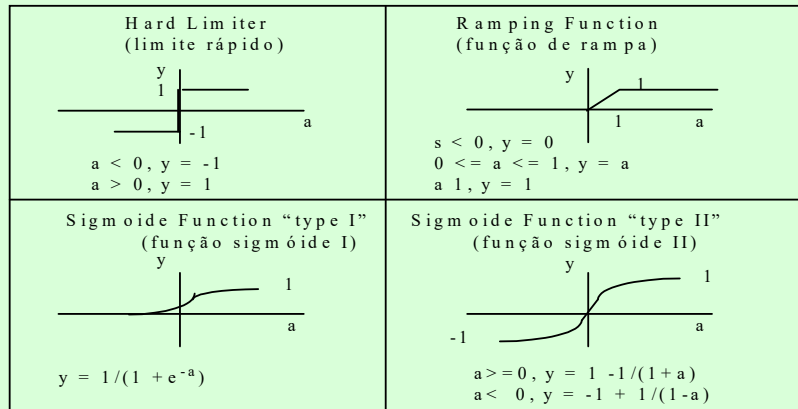
17

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



17

Outras funções de ativação não lineares (nem todas necessariamente disponíveis no ambiente MBP / no seu ambiente de simulação)

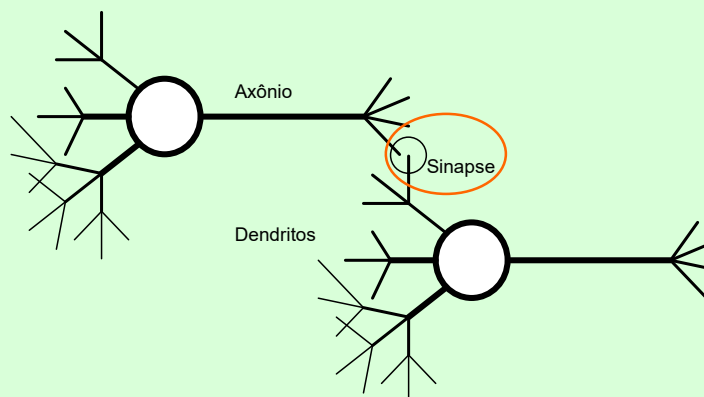


- Com escalamento do argumento, pode-se abarcar os universos digital e analógico / linear e não linear simultaneamente

© Prof. Emilio Del Moral Hernandez

18

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



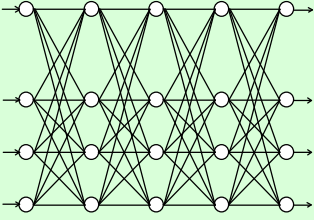
A conexão entre um axônio de um neurônio e um dendrito de outro é denominada **Sinapse**

© Prof. Emilio Del Moral Hernandez

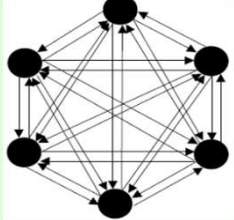
19

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

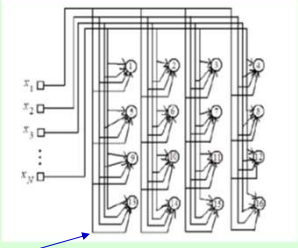
1) MLP
- Multi Layer
Perceptron

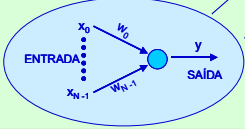


2) Memória
Associativa
de Hopfield



3) Mapas Auto-
Organizáveis
de Kohonen





ENTRADA x_0 w_0 y SAÍDA
 x_{n-1} w_{n-1}

© Prof. Emilio Del Moral Hernandez

20

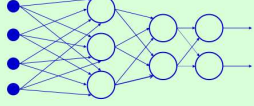
Três arquiteturas neurais importantes (abordadas em pósgrad – PSI 5886)

Nosso Foco aqui

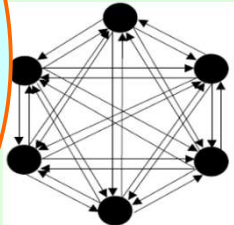
1) MLP
- Multi Layer
Perceptron

Foco deste Curso:
o Multi Layer Perceptron (MLP)

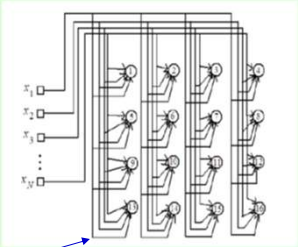
- Múltiplas entradas / Múltiplas saídas / Múltiplas camadas
- Variáveis (internas e externas) analógicas ou digitais
- Relações lineares ou não lineares entre elas

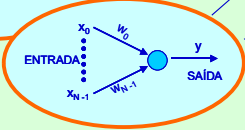


2) Memória
Associativa
de Hopfield



3) Mapas Auto-
Organizáveis
de Kohonen





ENTRADA x_0 w_0 y SAÍDA
 x_{n-1} w_{n-1}

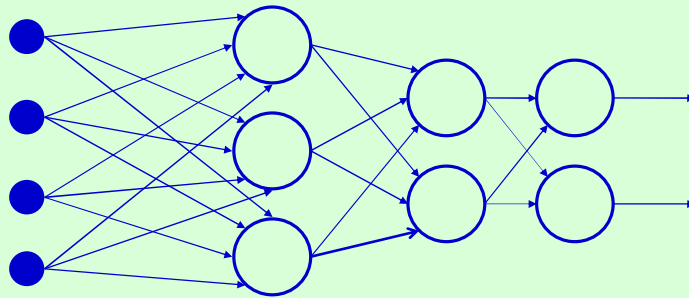
© Prof. Emilio Del Moral Hernandez

21

Foco deste Curso: o Multi Layer Perceptron (MLP)

24

- 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

24

24

Aparte ...

Vocês certamente já ouviram falar de ...

- Deep Learning
- Redes Neurais Profundas
- Redes Neurais Convolucionais

Pois é ... São temas bem quentes do momento e que estão muito relacionados com a arquitetura MLP que vocês estão aprendendo aqui em PSI3471!!

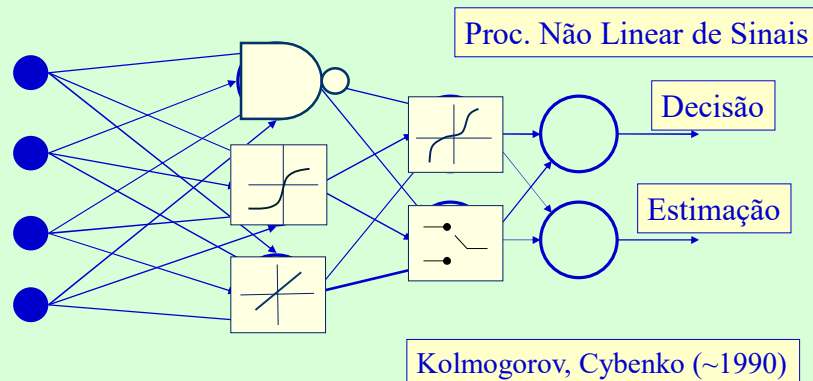
© Prof. Emilio Del Moral Hernandez

26

26

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

27

Resumindo os aspectos conceituais principais

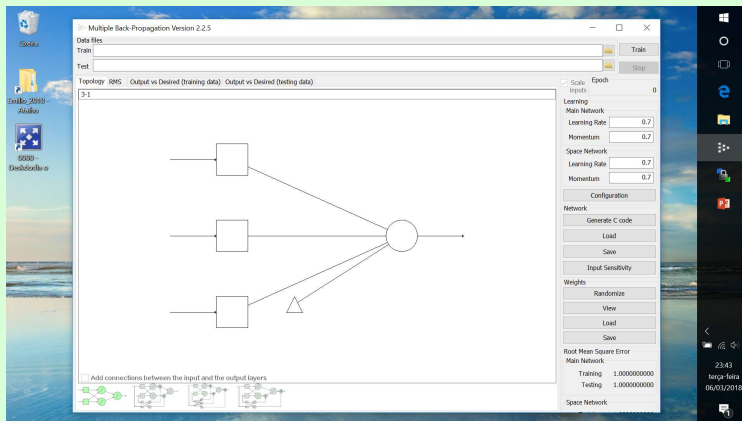
- 1) Não linearidade com a função neural sigmoide
 - 2) Possibilidade de conjugar na mesma estrutura ...
 - Cálculos digitais
 - Cálculos lineares multivariáveis
 - Funções genéricas não lineares multivariáveis
 - 3) Comportamento adaptativo com aprendizado através de exemplos
-
- Problemas complexos, 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

28

Sugestão: Instale o MBP e digite "3-1" no campo Topology (ou antes veja os tutoriais)

30



© Prof. Emilio Del Moral Hernandez

30

30

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

32

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

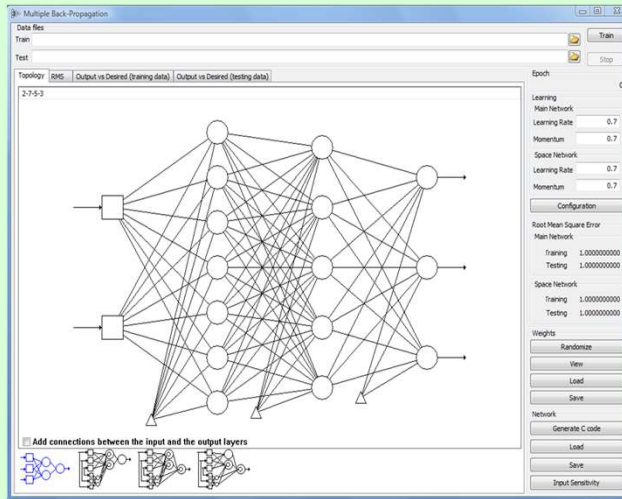
© Prof. Emilio Del Moral Hernandez

32

32

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

33

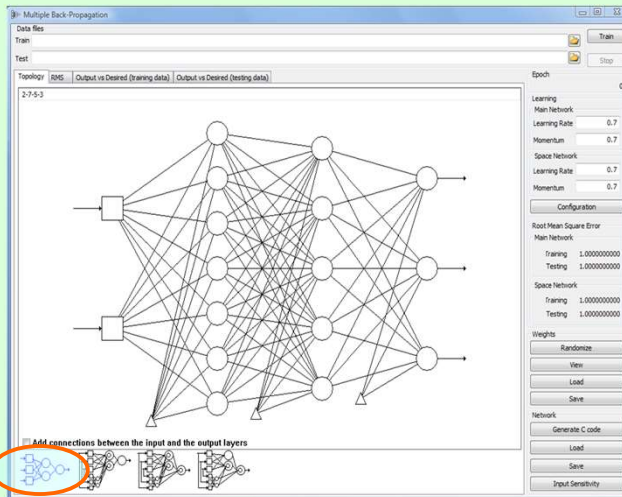


© Prof. Emilio Del Moral Hernandez

33

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

34



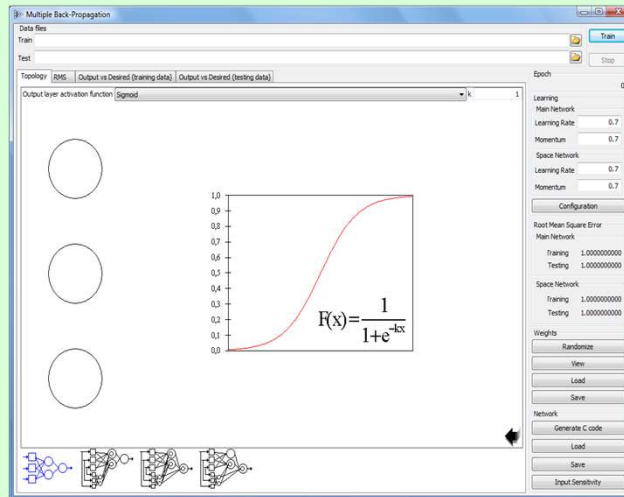
Escolha MLPs

© Prof. Emilio Del Moral Hernandez

34

Algumas Telas do MBP Mudando a função do nó neural

35



© Prof. Emilio Del Moral Hernandez

35

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

36

© Prof. Emilio Del Moral Hernandez

36

Conexão de RNAs com elementos de “Aprendizado de Máquina” / “Machine Learning”

37

O “aprendizado do modelo” é feito de maneira automática a partir de casos / de exemplos concretos: a definição matemática do modelo é feita a partir de um conjunto rico de exemplos numéricos empíricos de pares (X,y)

Conhecimento rico de exemplares / casos $(X^\mu ; y^\mu)$: Temos M observações empíricas $(X^\mu ; y^\mu)$, onde μ identifica cada observação, e varia entre 1 a M



y_{modelo} = cálculo neural (ou seja, somas ponderadas e com tgh's) que opera sobre as componentes do vetor X $(x_1, x_2, x_3 \dots)$; Esse cálculo neural é calibrado (via escolha dos valores dos ponderadores w's) a partir de M pares empíricos $(X^\mu ; y^\mu)$

37

37

Foco da semana que vem ... Aprendizado da rede neural

#5 (16/março – 2ªf) Foco no aprendizado da Rede Neural MLP – O Gradiente descendente e a otimização de pesos sinápticos com base no conjunto de treino e EBP; dedução das fórmulas do EBP, em sala de aula e em conjunto com os alunos: trabalho focado num peso sináptico específico da rede, escolhido pelo professor para máxima complexidade da dedução.

#6 (18/março – 4ªf) Discussão da extensão simples das deduções do EBP para demais pesos sinápticos; redundâncias nos cálculos dos diversos pesos da rede neural e otimização do esforço computacional. Regra “Delta” de aprendizado de Widrow, para neurônio isolado; Aprendizado por EBP recursivo, camada a camada.

#7 (23/março – 2ªf) Foco em Reconhedores – Circuitos lógicos genéricos e suas implementações através de computação neural; relação com discriminadores. Construindo reconhedores de padrões com fronteiras de separação genéricas: separadores lineares; bancos de separadores lineares como aproximadores efetivos de fronteiras genéricas.

#8 (25/março – 4ªf) Técnicas de medida de qualidade em reconhedores de padrões: matriz de confusão, conceitos de falsos positivo e negativo, especificidade e sensibilidade. Curvas ROC.

#9 (30/março – 2ªf) Foco agora em Regressores - Casos simples de aproximação de funções univariadas. Teorema de Cybenko: o MLP como aproximador universal de funções multivariadas; implicações práticas do teorema para a implementação de regressores e reconhedores de padrões não lineares multivariados genéricos.

#10 (01/abril – 4ªf) Medidas de qualidade diversificadas para regressores multivariados (distintas do erro quadrático médio); Flutuação do desempenho do modelo com as particulares amostras de treino e de teste e técnicas de reamostragem; técnica de validação cruzada, k-fold cross validation e leave one out. Sobreajuste / sobreaprendizado / perda de generalização em regressão polinomial e em redes neurais; limitação do número de nós neurais para evitar o sobreajuste e otimizar a generalização da rede neural; partição do volume de observações em conjuntos de treino, validação e teste.

06 e 08 de abril: Semana Santa – não há aula

© Prof. Emilio Del Moral Hernandez

38

38

39

Para STOA - Alguns | Multiple Back-Propa | #bottom | Multiple Back-Propa | Multiple Back-Propa | (342) Laboratorio de +

mbp.sourceforge.net/tutorial/tutorial2.html#bottom

Multiple Back-Propagation

About Screenshots Download Tutorial News Papers Develop/Contact

Multiple Back-Propagation

Date file: Train: Test: Epoch: 0

Learning Rate: 0.7
Momentum: 0.7
Speed Network: 0.7
Learning Rate: 0.7
Momentum: 0.7

Configuration

Least Mean Square Error
Main Network: Training: 1.0000000000
Speed Network: Training: 1.0000000000

Weights
Randomize
New
Load
Save

Network
Generate C code
Load
Save
Input Sensitivity

If you use this program for any scientific work, or related, please cite one of the references below and if possible send me an e-mail with the citation to emh@ipg.pt

Lopes, N. and Ribeiro, B. (2003). An Efficient Gradient-Based Learning Algorithm Applied to Neural Networks with Selective Activation Neurons. In *Neural, Parallel & Scientific Computation*, volume 11, pages 253-272. Dynamic Publishers.

Lopes, N. and Ribeiro, B. (2001). Hybrid learning in a multi neural network architecture. In *INNS-IEEE International Joint Conference on Neural Networks, IJCNN'01*, volume 4, pages 2788-2793, Washington D.C., USA.

In the Articles folder accompanying this program you will find links and the BibTex, ACM Ref and EndNote references for the articles.

For MBP to understand the data files, they must obey to the following conditions:

1. The columns containing the data must be separated by white spaces and its number must remain fixed.
2. The first line of the data file may optionally be a title line, where you may have a description of the columns. Just remember that the title columns cannot have white spaces because they are used as column separators. You may however use any other characters, such as the underscore, to separate words in the title columns.
3. MBP recognizes only numbers as data. Moreover only the dot (.) is recognized as the decimal separator and no thousands separator is recognized.

X	Y
0.218959	0.196178
0.047045	0.162958
0.678865	0.833997
0.679296	0.602405
0.934693	0.623243
0.383502	0.838679
0.519416	0.671648

Add connections between the input and the output layers

© Prof. Emilio Del Moral Hernandez

39

39

40

Para STOA - Alguns | Multiple Back-Propa | #bottom | Multiple Back-Propa | Multiple Back-Propa | (342) Laboratorio de +

mbp.sourceforge.net/tutorial/tutorial2.html#bottom

Multiple Back-Propagation

About Screenshots Download Tutorial News Papers Develop/Contact

Multiple Back-Propagation

Date file: Train: Test: Epoch: 0

Learning Rate: 0.7
Momentum: 0.7
Speed Network: 0.7
Learning Rate: 0.7
Momentum: 0.7

Configuration

Least Mean Square Error
Main Network: Training: 1.0000000000
Speed Network: Training: 1.0000000000

Weights
Randomize
New
Load
Save

Network
Generate C code
Load
Save
Input Sensitivity

Tetrahydrocortisone Pregnenetriol Type

3.0	1.3	A
1.9	1.4	A
9.1	0.6	B
9.2	7.9	C
2.6	0.1	A
3.9	0.6	B
3.8	0.2	A

Still you can code characteristics that have non-numeric values transforming them into numeric characteristics.

used Learning Algorithm Applied to Neural Networks with

In *Computational*, volume 11, pages 253-272. Dynamic

in several network architectures. In *INNS-IEEE International Joint*

ages 2788-2793, Washington D.C., USA.

and links and the BibTex, ACM Ref and EndNote references for

Thank you.

This program can be freely obtained on the site <http://ds.ipg.pt/MBP>. You should not pay or donate any money for this program. Please read the license accompanying the program.

Add connections between the input and the output layers

© Prof. Emilio Del Moral Hernandez

40

40

41

Multiple Back-Propagation

Para STQA - Alguns | Multiple Back-Propa | #bottom | Multiple Back-Propa | Multiple Back-Propa | (342) Laboratorio de +

mbp.sourceforge.net/tutorial/tutorial2.html#bottom

Multiple Back-Propagation

About Screenshots Download Tutorial News Papers Develop/Contact

Multiple Back-Propagation

Data files

Train: [] Train

Test: [] Test

Topology: [] Output as Desired (training data) [] Output as Desired

For example, if a variable has three possible values (A, B and C) you can replace it with three columns of data, so that each column would have a value of one for all the lines where the characteristic presents the corresponding values and a value of zero for the remaining lines.

Tetrahydrocortisone	Pregnenetriol	Type	Type-A	Type-B	Type-C
3.0	1.3	A	1	0	0
1.9	1.4	A	1	0	0
9.1	0.6	B	0	1	0
9.2	7.9	C	0	0	1
2.6	0.1	A	1	0	0
3.9	0.6	B	0	1	0
3.8	0.2	A	1	0	0

Thank you.

This program can be freely obtained on the site <http://dx.doi.org/10.1002/9781118111111.ch11>. You should not pay or donate any money for this program. Please read the license accompanying the program.

Configuration

Epoch: 0

Main Network

Learning Rate: 0.7

Momentum: 0.7

Spool Network

Learning Rate: 0.7

Momentum: 0.7

Configuration

Most Mean Square Error

Main Network

Training: 1.0000000000

Testing: 1.0000000000

Spool Network

Training: 1.0000000000

Testing: 1.0000000000

Weights

Randomize

View

Load

Save

Network

Generate C code

Load

Save

Input Sensitivity

© Prof. Emilio Del Moral Hernandez

41

41

42

Tutorial 3 – definindo entradas saídas e topologia da rede neural

42

Multiple Back-Propagation

Para STQA - Alg... | Multiple Back-Pr | #bottom | Nova guia | (341) Laboratoric +

mbp.sourceforge.net/tutorial/tutorial3.html#bottom

Multiple Back-Propagation

About Screenshots Download Tutorial News Papers Develop/Contact

Multiple Back-Propagation

Data files

Train: [] Train

Test: [] Test

Topology: [] Output as Desired (training data) [] Output as Desired

If you use this program for any scientific work, or related, please cite one of the references below and if possible send me an e-mail with the citation to emdel@ig.com.br.

Lopes, N. and Ribeiro, B. (2003). An Efficient Gradient-Based Learning Algorithm Applied to Neural Networks with Selective Activation Neurons. In *Neural, Parallel & Scientific Computation*, volume 11, pages 253-272. Dynamic Publishers.

Lopes, N. and Ribeiro, B. (2001). Hybrid learning in a multi neural network architecture. In *INNS-IEEE International Joint Conference on Neural Networks, IJCNN'01*, volume 4, pages 2758-2763. Washington D.C., USA.

In the Articles folder accompanying this program you will find references for the articles.

Thank you.

This program can be freely obtained on the site <http://dx.doi.org/10.1002/9781118111111.ch11>. You should not pay or donate any money for this program. Please read the license accompanying the program.

Configuration

Epoch: 0

Main Network

Learning Rate: 0.7

Momentum: 0.7

Spool Network

Learning Rate: 0.7

Momentum: 0.7

Configuration

Most Mean Square Error

Main Network

Training: 1.0000000000

Testing: 1.0000000000

Spool Network

Training: 1.0000000000

Testing: 1.0000000000

Weights

Randomize

View

Load

Save

Network

Generate C code

Load

Save

Input Sensitivity

© Prof. Emilio Del Moral Hernandez

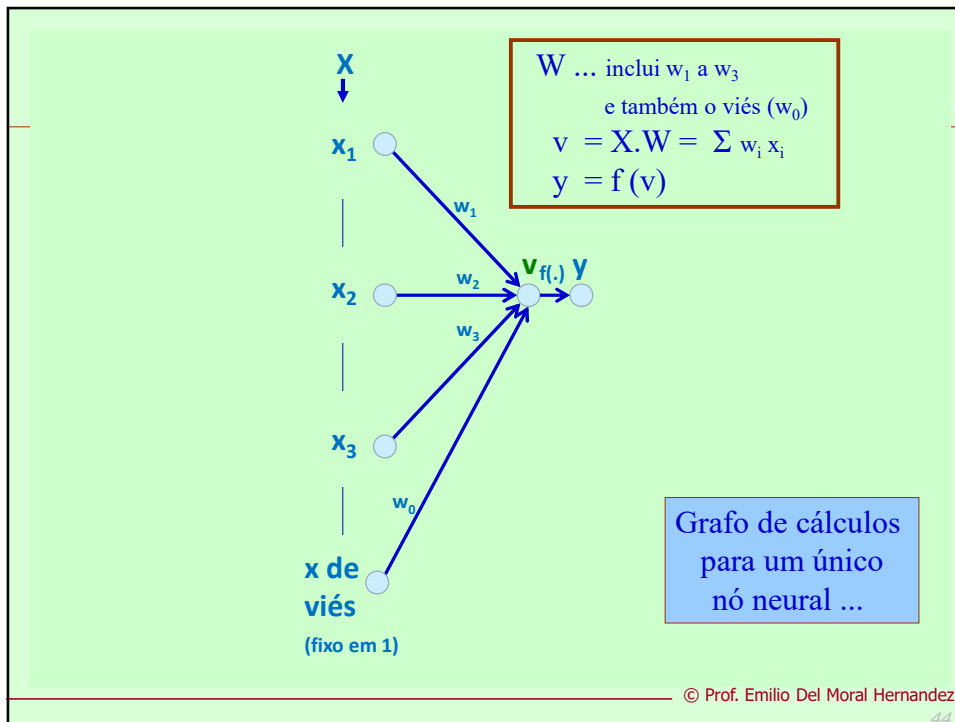
42

42

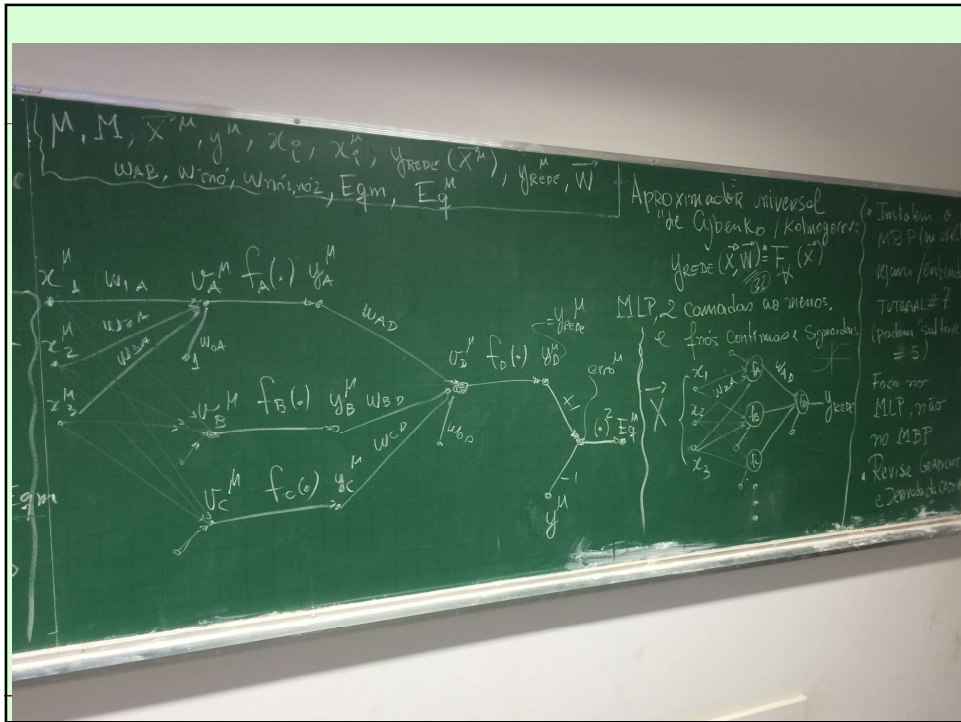
- 1) Definindo mais formalmente a computação da rede neural
- 2) Preparando o terreno para o aprendizado de MLPs ...

Vamos falar um pouco sobre GRAFOS?

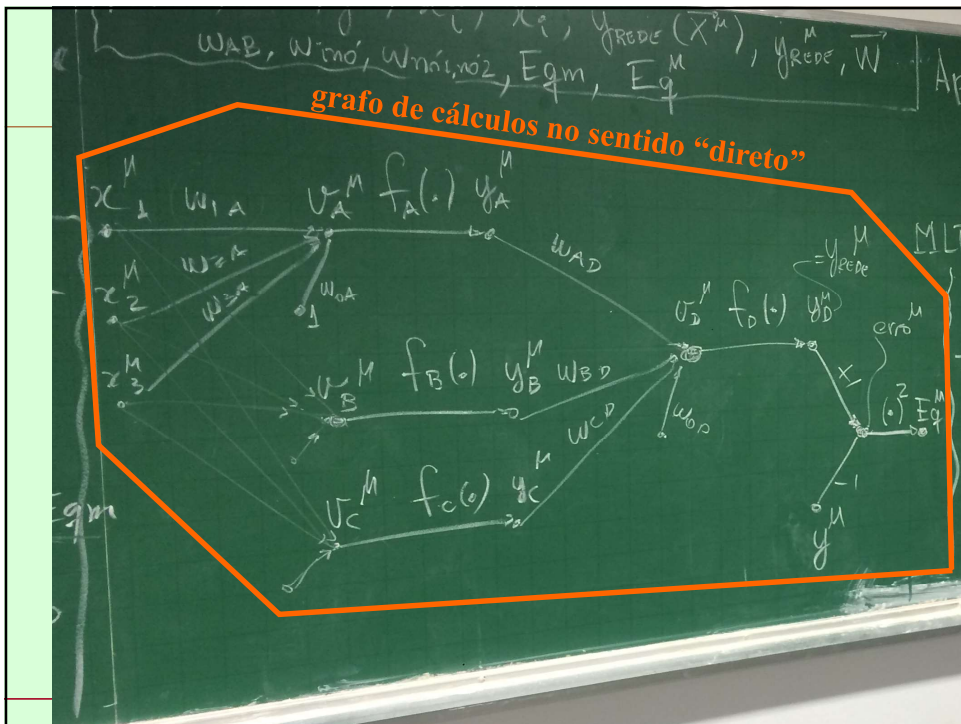
43



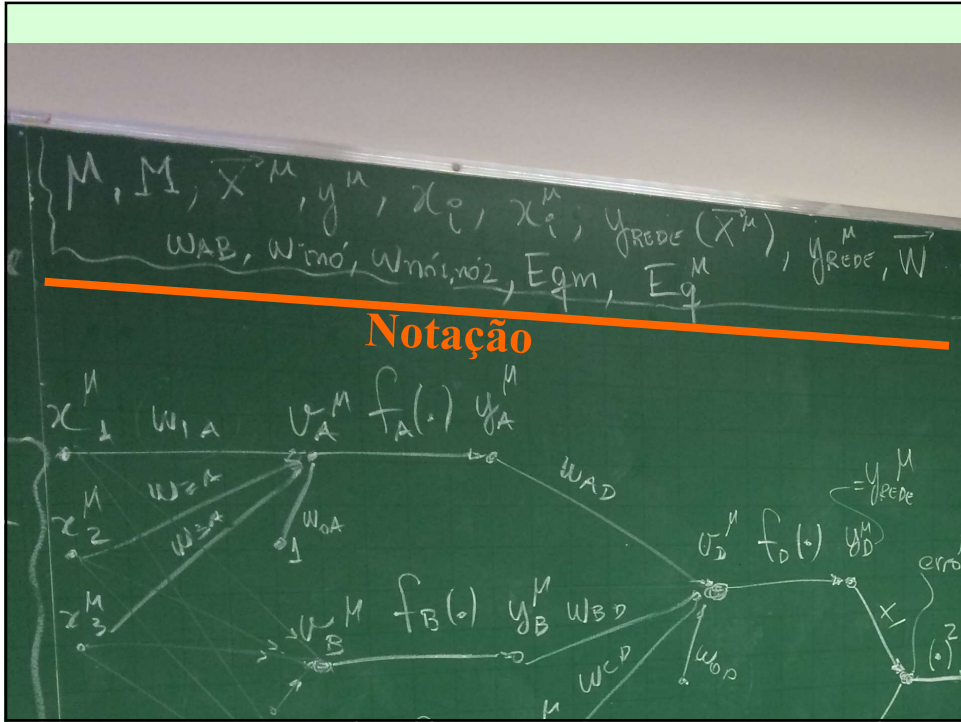
44



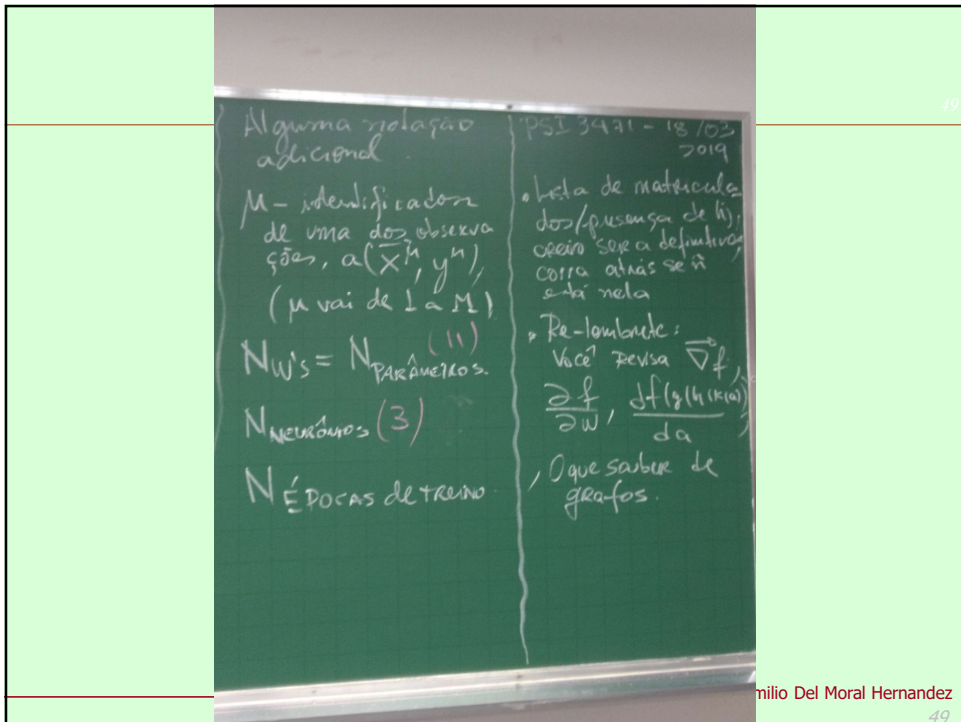
46



47

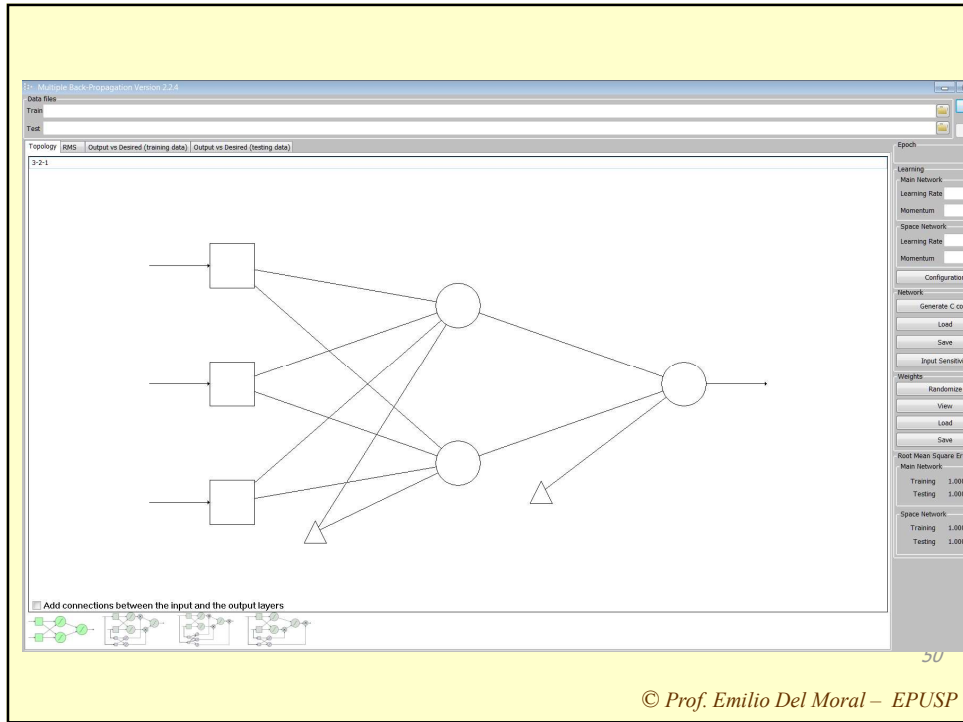


48

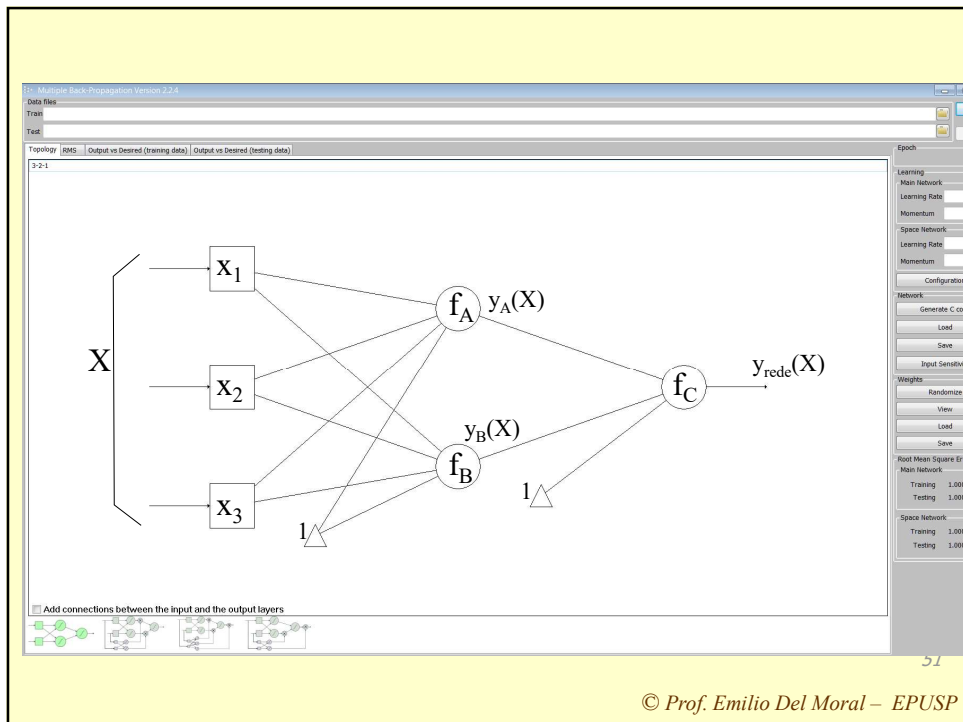


49

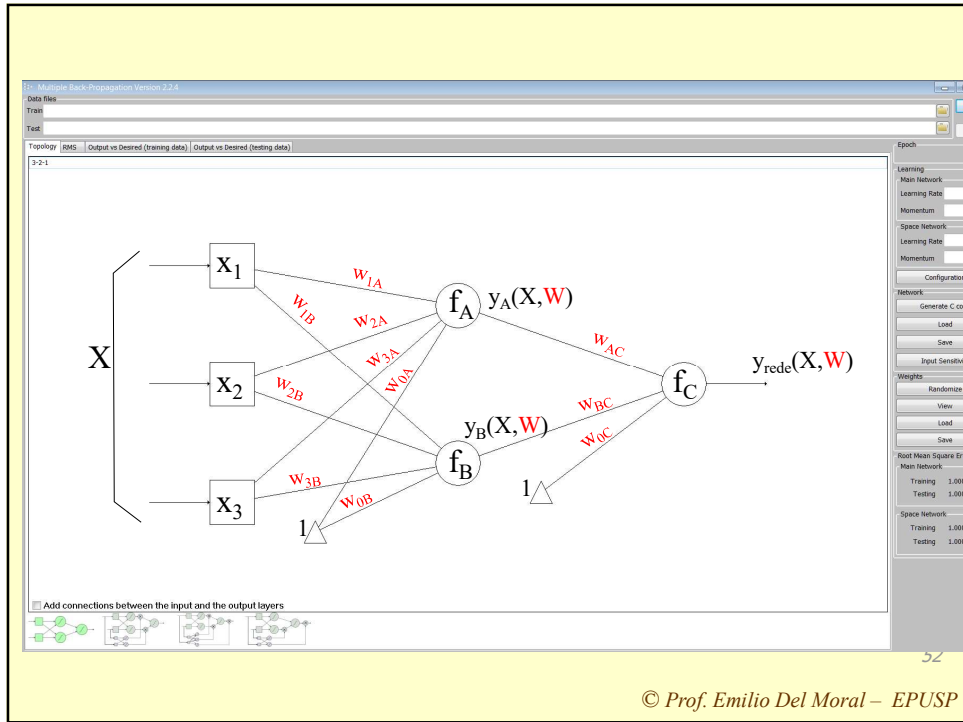
nilio Del Moral Hernandez



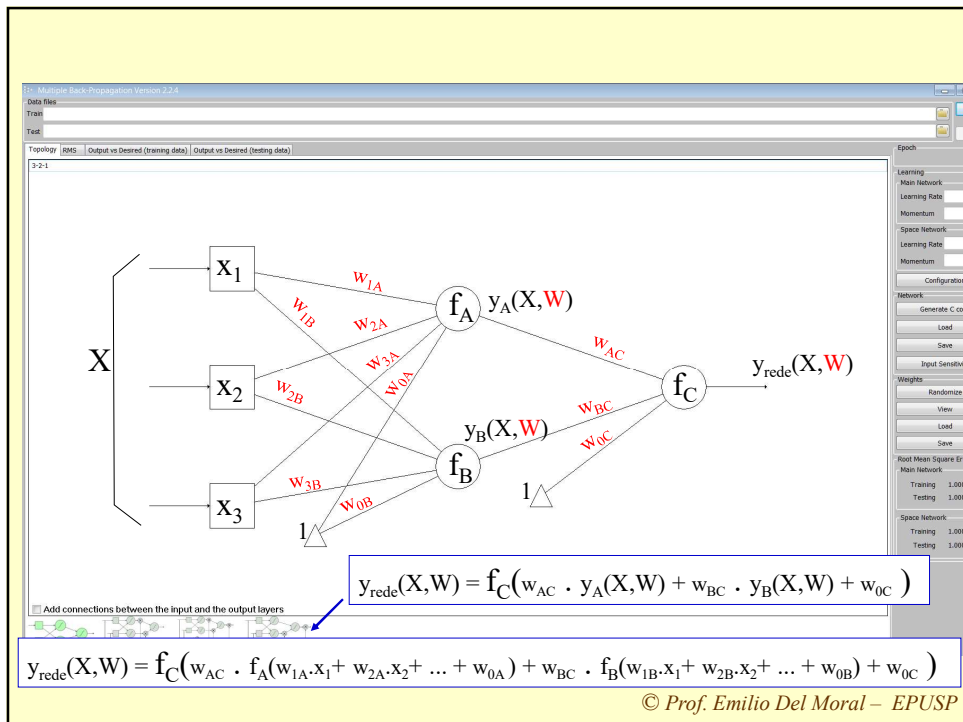
50



51



52



53

... 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)

54

Resumo de principais resultados em lousa ...

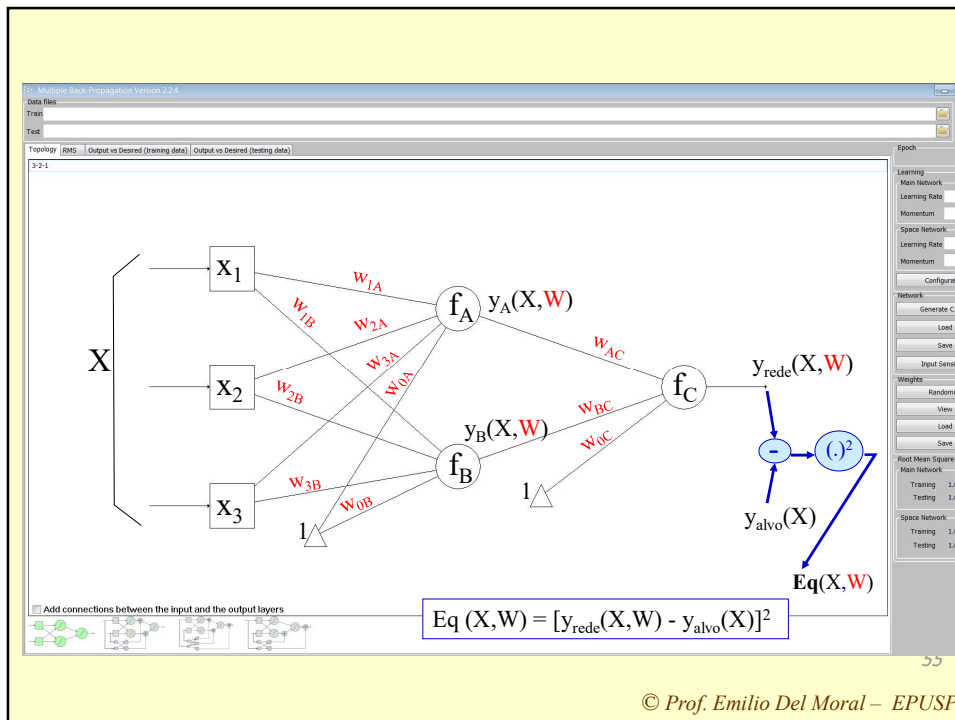
μ identifica um de M exemplos de treinamento

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

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

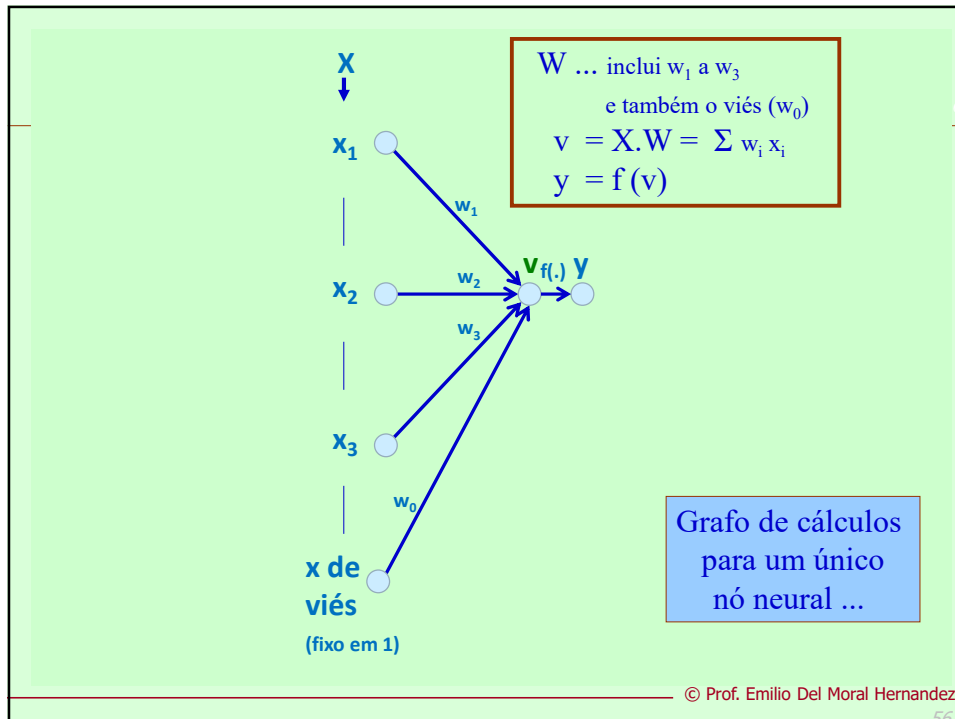
© Prof. Emilio Del Moral Hernandez

54



© Prof. Emilio Del Moral – EPUSP

55



56

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, $y = \text{Consumo do Produto A} = F(x_1, x_2, x_3, x_4, x_5)$.
- 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 indivíduo 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.

57

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

58

Equivalente 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)
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

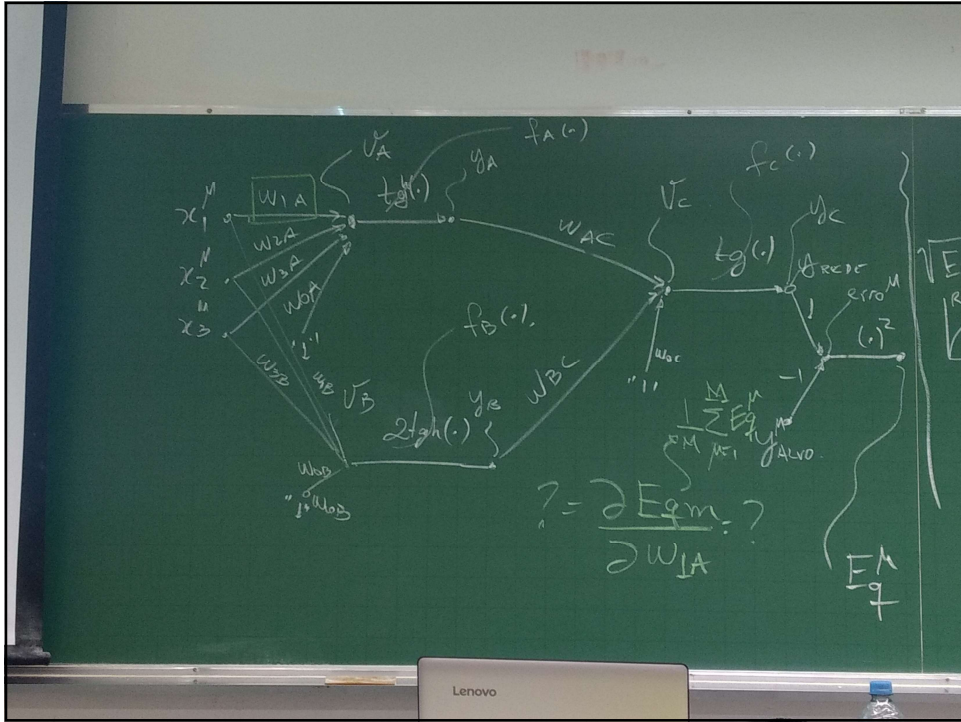
*Equivalente em txt
Para uso do MBP*

```

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
    
```

Moral - EPUSP

59



60