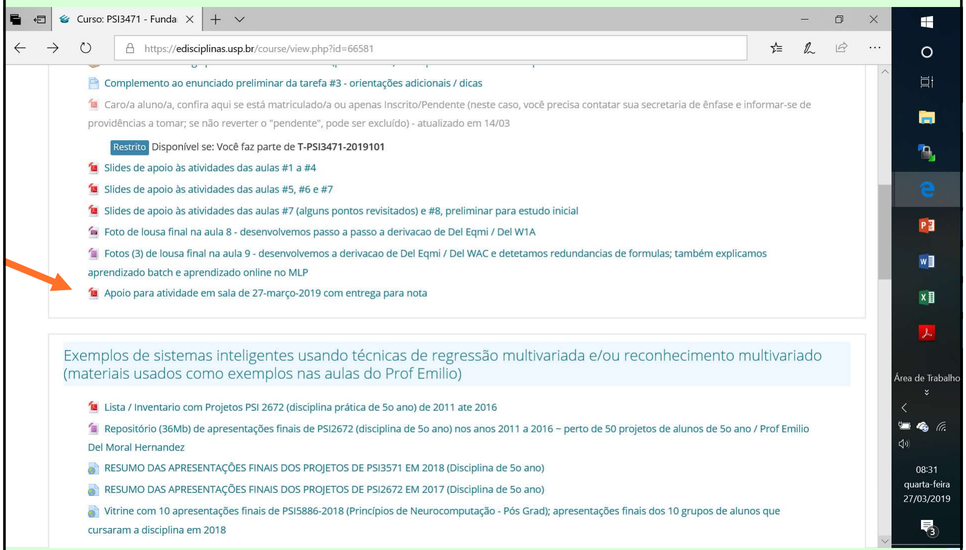
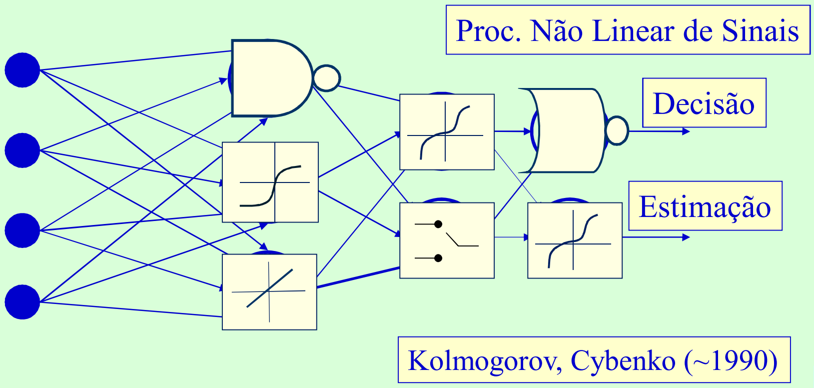


Prof. Emilio Del Moral Hernandez

3

O Multi Layer Perceptron (MLP)

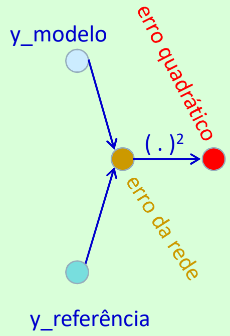
- Múltiplas entradas / Múltiplas saídas / Múltiplas camadas
- Variáveis (internas e externas) analógicas ou digitais
- Relações lineares ou não lineares entre elas



Prof. Emilio Del Moral Hernandez

5

Modelo Neural no modo de Treinamento

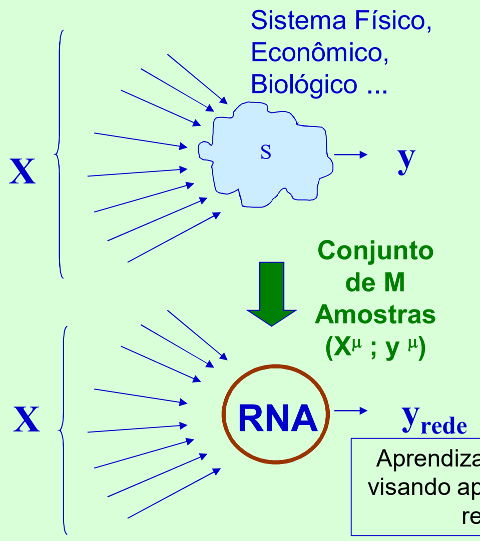


... erro da rede com relação ao conjunto de treinamento como um todo; simbologia (X^μ ; y^μ); Erro quadrático de exemplar (Eq^μ); Erro quadrático médio (Eqm)

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

μ *identifica um de M exemplos de treinamento*

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

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

... em loop ...

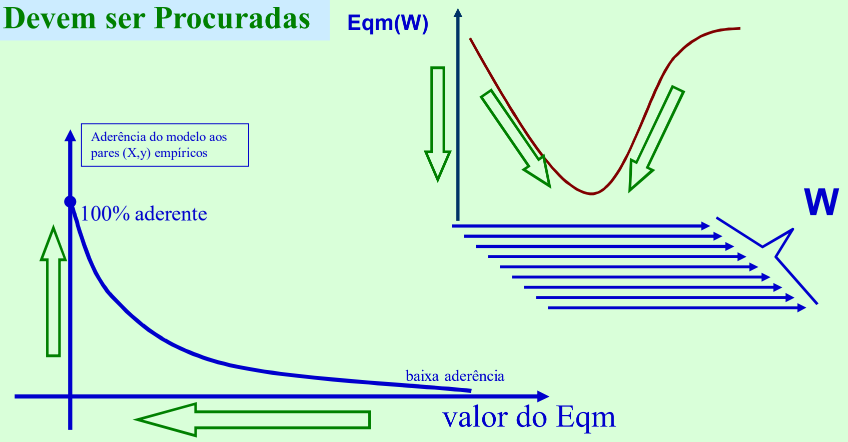
$$\Delta\vec{W} = -\eta.\vec{\nabla}Eqm$$

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

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



Método do Gradiente Aplicado aos nossos MLPs: a partir de um W#0, temos aproximações sucessivas ao Eqm mínimo, por repetidos pequenos passos DeltaW, sempre contrários ao gradiente ...

- "Chute" um W inicial para o "Wcorrente", ou "W melhor até agora"
- Em loop até obter Eqm zero, ou baixo o suficiente, ou estável:
  - Determine o vetor gradiente do Eqm, nesse espaço de Ws
  - Em loop varrendo todos os M exemplos (X^{μ};y^{μ}),
    - Calcule o gradiente de Eq^{μ} associado a um exemplo μ, e vá varrendo μ e somando os gradientes de cada Eq^{μ}, para compor o vetor gradiente de Eqm, assim que sair deste loop em μ ;
    - Cada cálculo como esse, envolve primeiro calcular os argumentos de cada tangente hiperbólica e depois usar esses argumentos na regra da cadeia das derivadas necessárias
  - Tire a média dos M gradientes individuais e dê um passo Delta ΔW nesse espaço, com direção e magnitude dados por $-\eta$*vetor gradiente (Eqm)

Prof. Emilio Del Moral Hernandez

10

Processo de refinamentos graduais a cada iteração ...

W#0	Eqm#0	GradEqm(W#0)	DeltaW#0 = - n.GradEqm(W#0)
W#1 (= W#0 + DeltaW#0)	Eqm#1 (< Eqm#0)	GradEqm(W#1)	DeltaW#1 = - n.GradEqm(W#1)
W#2 (= W#1 + DeltaW#1)	Eqm#2 (< Eqm#1)	GradEqm(W#2)	DeltaW#2 = - n.GradEqm(W#2)
W#3 (= W#2 + DeltaW#2)	Eqm#3 (< Eqm#2)	GradEqm(W#3)	DeltaW#3 = - n.GradEqm(W#3)
W#4 (= W#3 + DeltaW#3)	Eqm#4 (< Eqm#3)	GradEqm(W#4)	DeltaW#4 = - n.GradEqm(W#4)
...	...	...	...
W#k (= W#k-1 + DeltaW#k-1)	Eqm#k (< Eqm#k-1)	GradEqm(W#k)	DeltaW#k = - n.GradEqm(W#4)
...	...	...	...

Prof. Emilio Del Moral Hernandez

11

Comparando adaptação de pesos por visitas (em loop) unificadas / completas ao lote de M exemplo antes de cada adaptação, *versus* M adaptações parciais, exemplo a exemplo, com M subvisitas individuais aos M componentes do lote (revisitado em loop também): aprendizado em batch *versus* aprendizado online

W#0	Eqm#0	GradEqm(W#0)	DeltaW#0 = - n.GradEqm(W#0)
W#1 (= W#0 + DeltaW#0)			

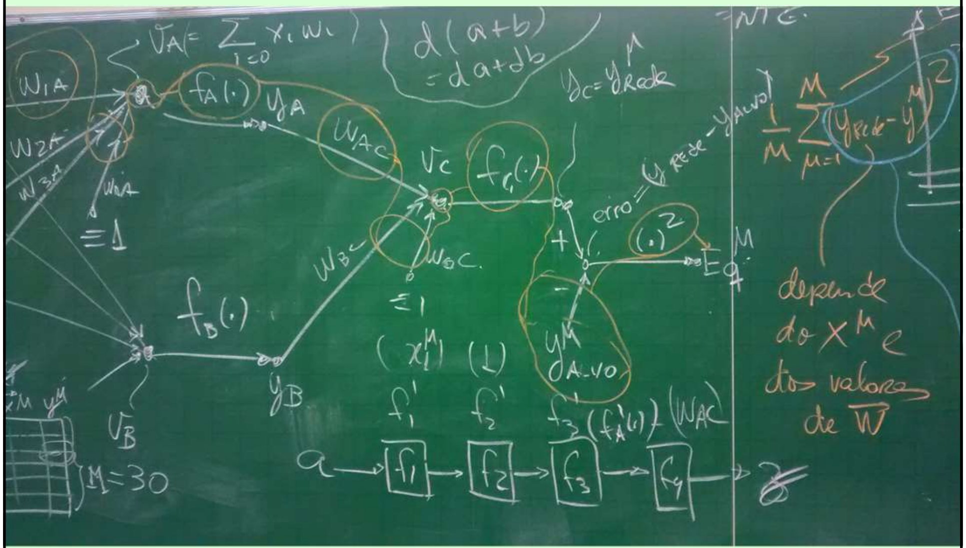
$$\text{Grad(Eqm)} = \Sigma_\mu \text{ Grad}(\text{Eq}^\mu) / M$$

$$\begin{aligned} \text{DeltaW\#0} &= - \eta.\text{Grad(Eqm)} \\ &= \Sigma_\mu - \eta.\text{ Grad}(\text{Eq}^\mu) / M \\ &= \Sigma_\mu \text{ DeltaW\#0}^{\mu} / M \\ &\sim \Sigma_\mu \text{ DeltaonlineW\#0}^{\mu} / M \end{aligned}$$

Treinamento em Batch (batelada) versus Online (serial) :

- O Grad(Eq$^\mu$) associado a apenas um exemplo específico (X$^\mu$;y$^\mu$) é obtido como desenvolvido nas aulas anteriores do módulo, devemos calcular tantas derivadas parciais ? Eq$^\mu$ / ? w_xx quantos pesos w_xx existam na rede.
- Fazemos esses cálculos através, por exemplo, das técnicas de retropropagação e dos passos direto e reverso. Assim calculamos todas as componentes ? Eq$^\mu$ / ? w_xx e compomos o gradiente de Eq$^\mu$, referente a um μ específico (cada μ define um para entrada / saída de treino).
- ~~Cumulativamente, somamos esses M gradientes de exemplar, Grad (Eq$^\mu$), para ao final da varredura de μ, chegarmos ao vetor médio Grad (Eqm)= Σ Grad (Eq$^\mu$)/M~~

Adaptamos "online" os pesos w's: Wnovo=Wanterior –η.Grad (Eq$^\mu$)



Falemos algo sobre as
Redes Neurais Convolucionais –
ou ConvNets –
ou Convolutional Neural Networks

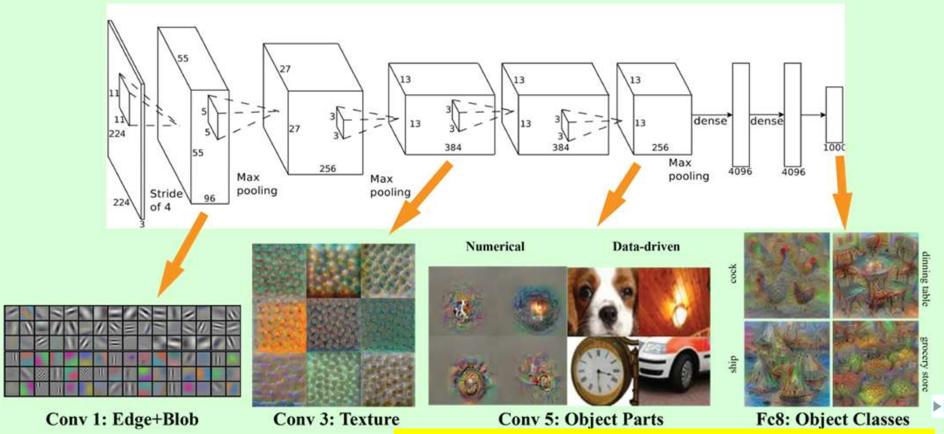
Trata-se de outra técnica atual de grande emprego no contexto de Deep Learning e que também traz algumas soluções para o sobreaprednizado em aplicações de alta dimensão de entrada

Mark Cappello Ferreira de Sousa

Classificação em Redes Neurais Convolucionais

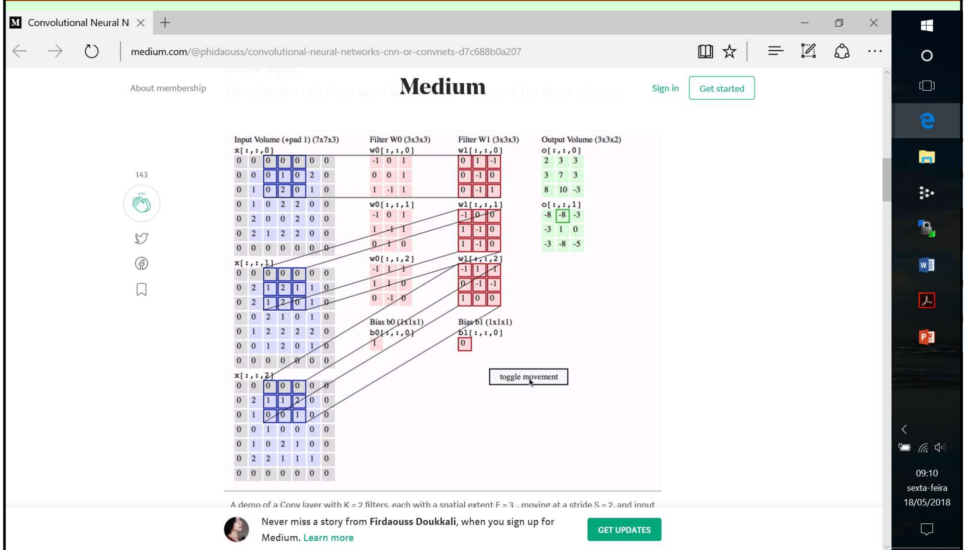
Convoluções
(Extração de características)

Fully Connected
(MLP)



figuras ilustrativas – extraídas da internet

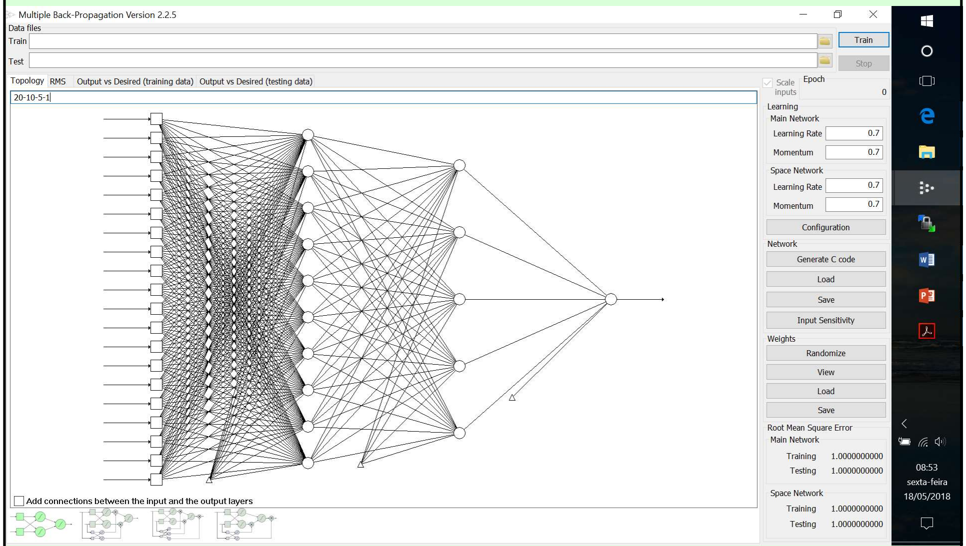
23



© Prof. Emilio Del Moral Hernandez

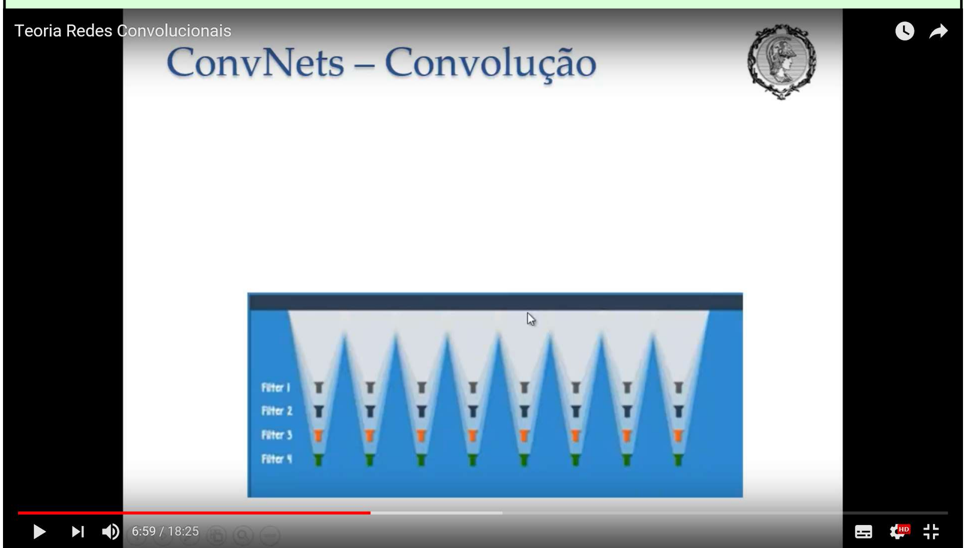
24

Num MLP tradicional, quantos parâmetros “w´s” temos só na primeira camada (com digamos 10 neurônios) se a imagem de entrada tiver 1M pixels?



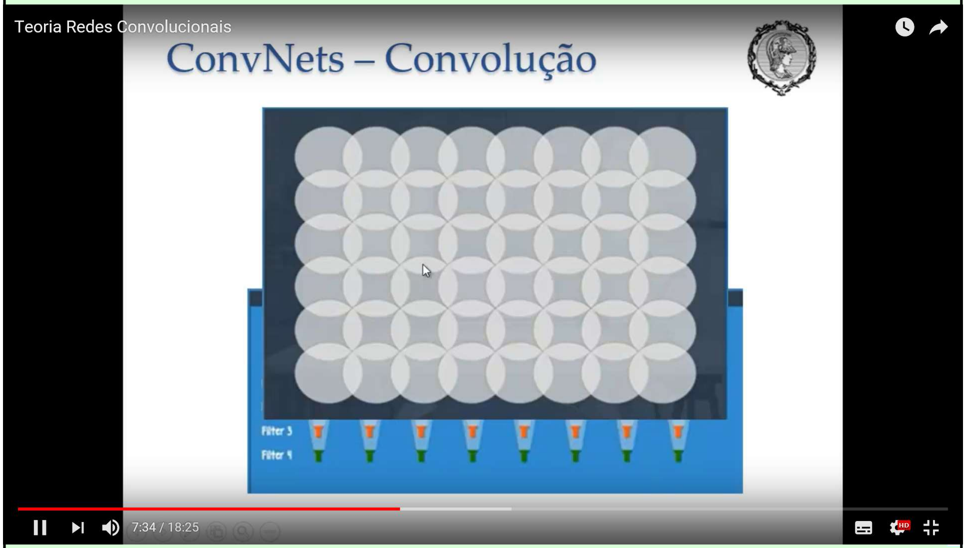
© Prof. Emilio Del Moral Hernandez

25



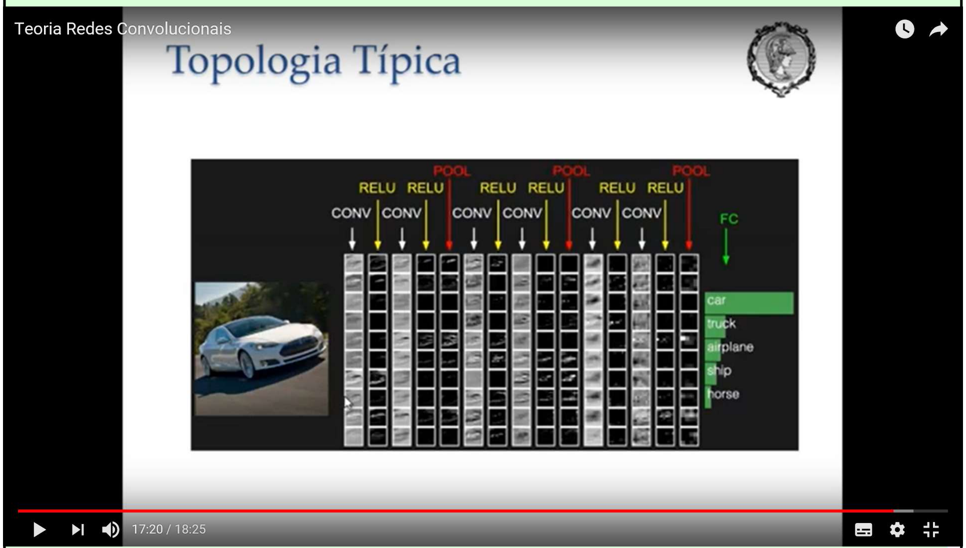
© Prof. Emilio Del Moral Hernandez

26



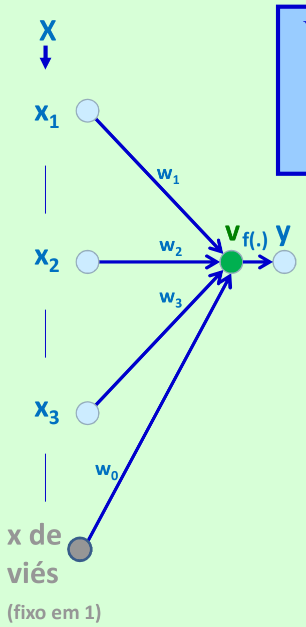
Teoria Redes Convolucionais
ConvNets – Convolução
Filter 3
Filter 4
7:34 / 18:25

27



Teoria Redes Convolucionais
Topologia Típica
POOL POOL POOL
RELU RELU RELU RELU RELU RELU
CONV CONV CONV CONV CONV CONV
FC
car
truck
airplane
ship
horse
17:20 / 18:25

28



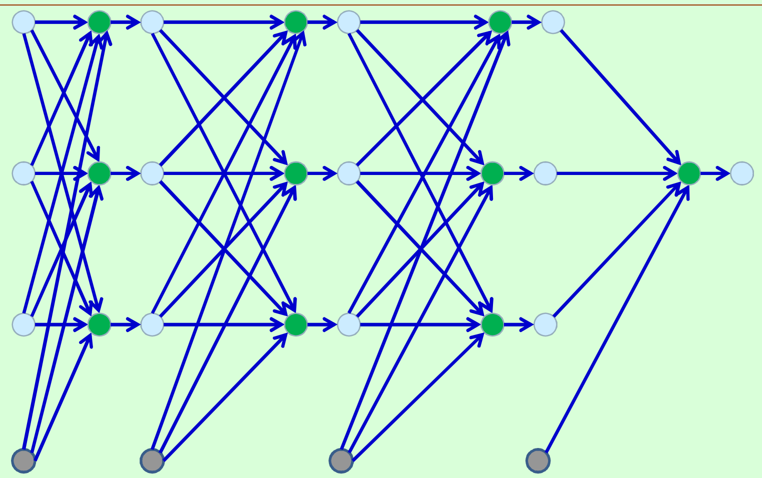
W … inclui w_1 a w_3
e também o viés (w_0)

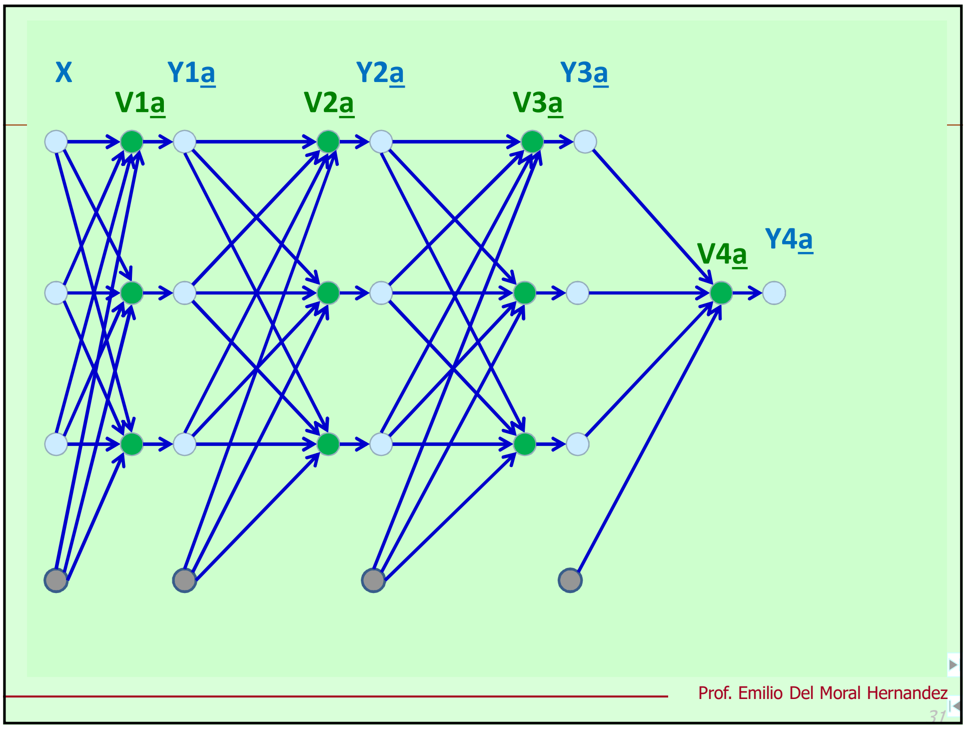
$v = X.W = \Sigma\, w_i\, x_i$

$y = f(v)$

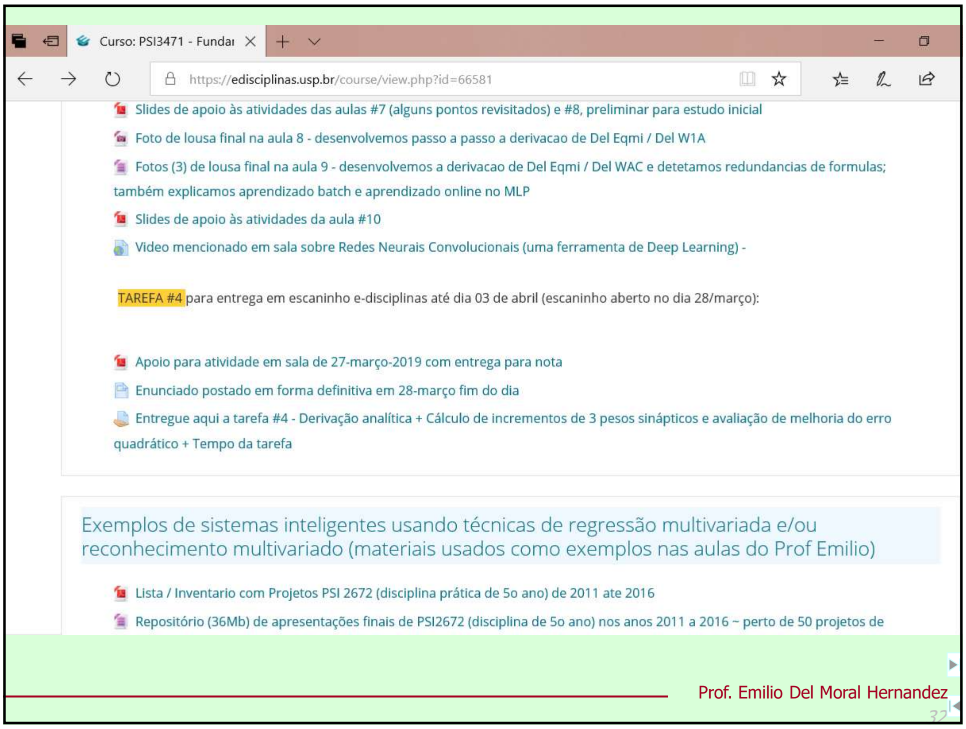
Grafo de cálculos para um único nó neural …

Modelo Neural com vários nós, no modo de operação: Um X de entrada sendo mapeado em $y_{rede}(X)$



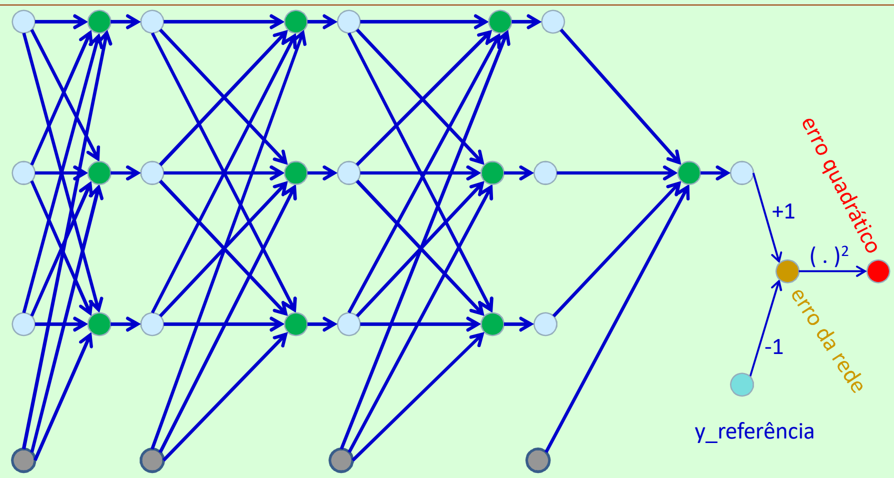

X
Y1a
V1a
Y2a
V2a
Y3a
V3a
V4a
Y4a
Prof. Emilio Del Moral Hernandez

31


Curso: PSI3471 - Fundar
https://edisciplinas.usp.br/course/view.php?id=66581
Slides de apoio às atividades das aulas #7 (alguns pontos revisitados) e #8, preliminar para estudo inicial
Foto de lousa final na aula 8 - desenvolvemos passo a passo a derivacao de Del Eqmi / Del W1A
Fotos (3) de lousa final na aula 9 - desenvolvemos a derivacao de Del Eqmi / Del WAC e detetamos redundancias de formulas; também explicamos aprendizado batch e aprendizado online no MLP
Slides de apoio às atividades da aula #10
Video mencionado em sala sobre Redes Neurais Convolucionais (uma ferramenta de Deep Learning) -
TAREFA #4 para entrega em escaninho e-disciplinas até dia 03 de abril (escaninho aberto no dia 28/março):
Apoio para atividade em sala de 27-março-2019 com entrega para nota
Enunciado postado em forma definitiva em 28-março fim do dia
Entregue aqui a tarefa #4 - Derivação analítica + Cálculo de incrementos de 3 pesos sinápticos e avaliação de melhoria do erro quadrático + Tempo da tarefa
Exemplos de sistemas inteligentes usando técnicas de regressão multivariada e/ou reconhecimento multivariado (materiais usados como exemplos nas aulas do Prof Emilio)
Lista / Inventario com Projetos PSI 2672 (disciplina prática de 5o ano) de 2011 ate 2016
Repositório (36Mb) de apresentações finais de PSI2672 (disciplina de 5o ano) nos anos 2011 a 2016 ~ perto de 50 projetos de
Prof. Emilio Del Moral Hernandez

32

Modelo Neural no modo de Treinamento (supervisionado) por minimização do erro quadrático [e Avaliação Qualidade]



"Regra Delta" (Widrow) para o aprendizado em nó simples:

Gradiente Descendente = Regra Δ p/ aprendizado de um nó:

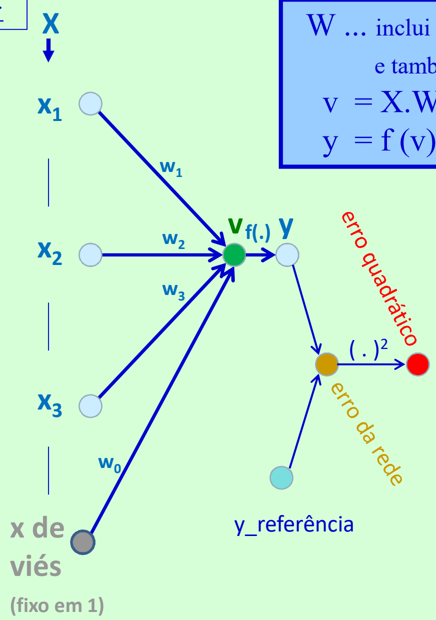
$\Delta w_i = - x_i.f'(v).2.erro$

... ou, vetorialmente:

$\Delta W = - X.f'(v).2.erro$

(ΔW = correções de W)

... em loop ...



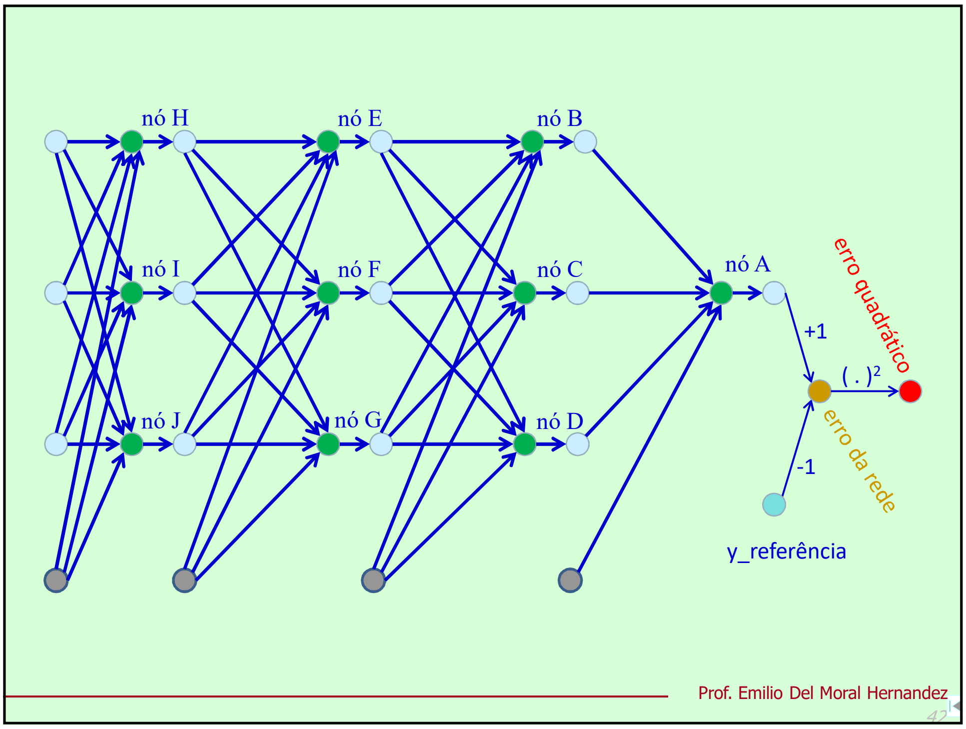
W ... inclui w_1 a w_3

e também o viés (w_0)

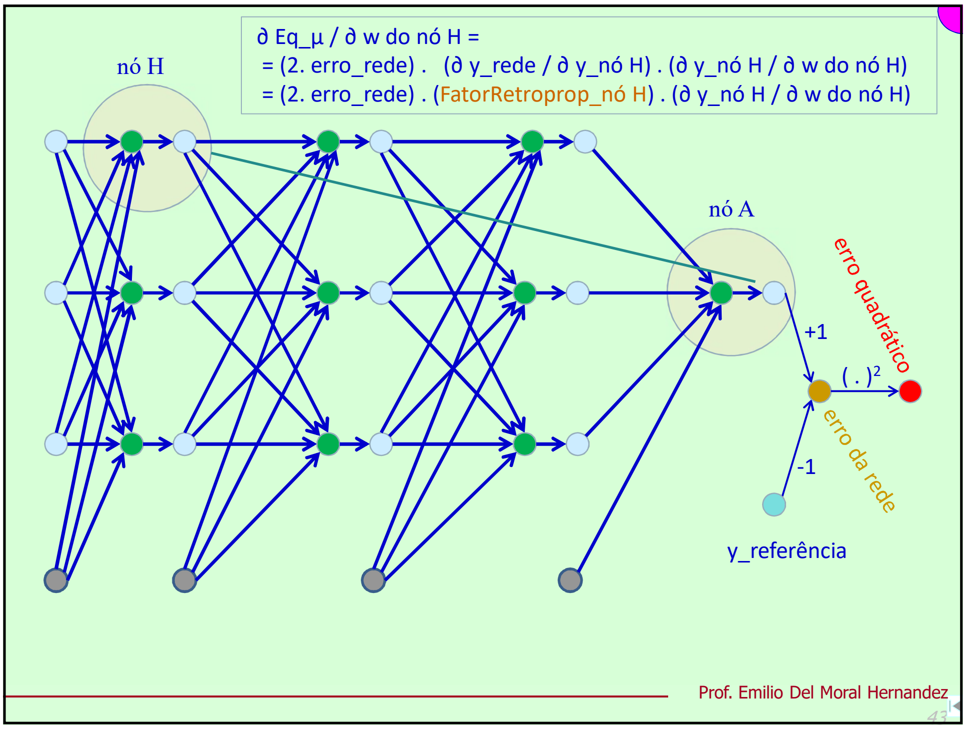
$v = X.W = \Sigma\, w_i\, x_i$

$y = f(v)$

Grafo de cálculos para um único nó neural ...

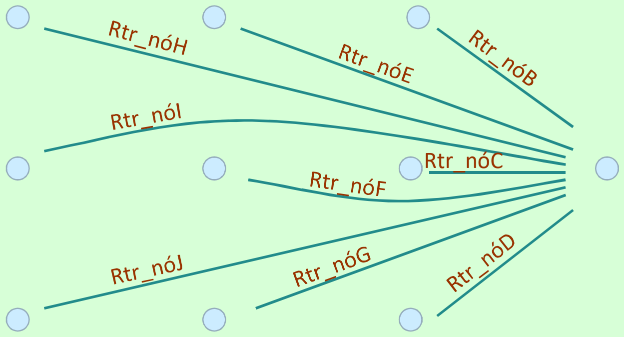

nó H
nó E
nó B
nó I
nó F
nó C
nó A
nó J
nó G
nó D
+1
-1
$(\,.\,)^2$
erro quadrático
erro da rede
y_referência
Prof. Emilio Del Moral Hernandez
42

42


∂ Eq_µ / ∂ w do nó H =
= (2. erro_rede) . (∂ y_rede / ∂ y_nó H) . (∂ y_nó H / ∂ w do nó H)
= (2. erro_rede) . (FatorRetroprop_nó H) . (∂ y_nó H / ∂ w do nó H)
nó H
nó A
+1
-1
$(\,.\,)^2$
erro quadrático
erro da rede
y_referência
Prof. Emilio Del Moral Hernandez
43

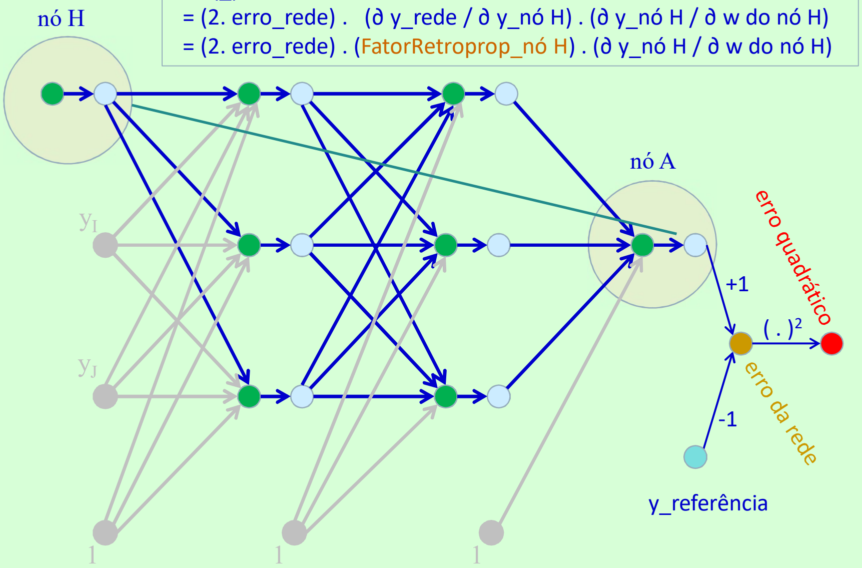
43

Temos um retropropagador para cada no´ da rede, a excessao do no´ de saida "nó A": o nó de saída não necessita do retropropagador, já que o erro da saída da rede é o próprio erro desse nó; aliás, se tentar calcular (∂ y_rede / ∂ y_nó) para ele, chega-se obviamente a Rtr_nóA = 1.

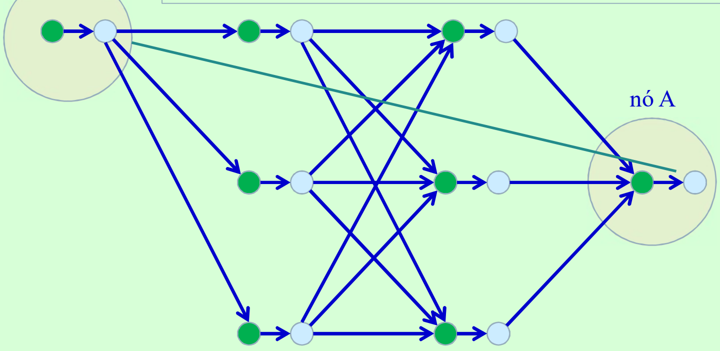


∂ Eq_μ / ∂ w para w nó *KK* =
= (2. erro_rede) . (∂ y_rede / ∂ y_nó *KK*) . (∂ y_nó *KK* / ∂ w)
= (2. erro_rede) . (FatorRetroprop_nó *KK*) . (∂ y_nó *KK* / ∂ w)

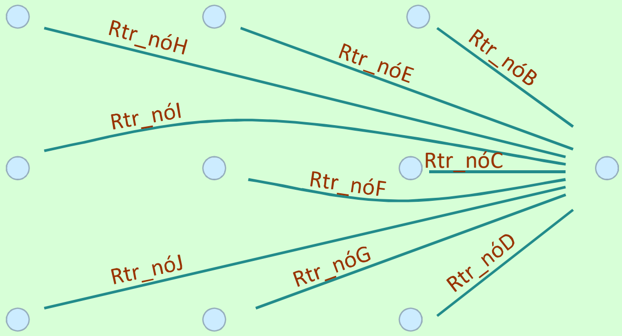
∂ Eq_μ / ∂ w do nó H =
= (2. erro_rede) . (∂ y_rede / ∂ y_nó H) . (∂ y_nó H / ∂ w do nó H)
= (2. erro_rede) . (FatorRetroprop_nó H) . (∂ y_nó H / ∂ w do nó H)



∂ Eq_μ / ∂ w do nó H =
= (2. erro_rede) . (∂ y_rede / ∂ y_nó H) . (∂ y_nó H / ∂ w do nó H)
= (2. erro_rede) . (FatorRetroprop_nó H) . (∂ y_nó H / ∂ w do nó H)



Temos um retropropagador para cada no´ da rede, a excessao do no´ de saida "nó D": o nó de saída não necessita do retropropagador, já que o erro da saída da rede é o próprio errodesse nó; aliás, se tentar calcular (∂ y_rede / ∂ y_nó) para ele, chega-se obviamente a Rtr_nóD = 1.



∂ Eq_μ / ∂ w para w nó NN =
= (2. erro_rede) . (∂ y_rede / ∂ y_nó NN) . (∂ y_nó NN / ∂ w)
= (2. erro_rede) . (FatorRetroprop_nó NN) . (∂ y_nó NN / ∂ w)

Definindo novas variáveis: uma nova variável auxiliar associada a cada nó, chamada erro do nó

... Revisitemos a expressão para $\partial Eq\_\mu / \partial w$, reagrupando termos

$\partial Eq\_\mu / \partial w =$

$= (2.\ erro\_rede).\ (\partial y\_rede / \partial y\_nó).\ (\partial y\_nó / \partial w)$

definição

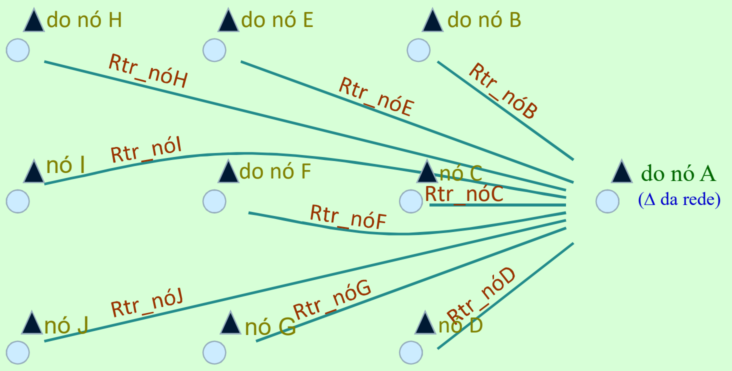
$= (2.\ erro\_rede).\ (FatorRetroprop\_nó).\ (\partial y\_nó / \partial w)$

\ /

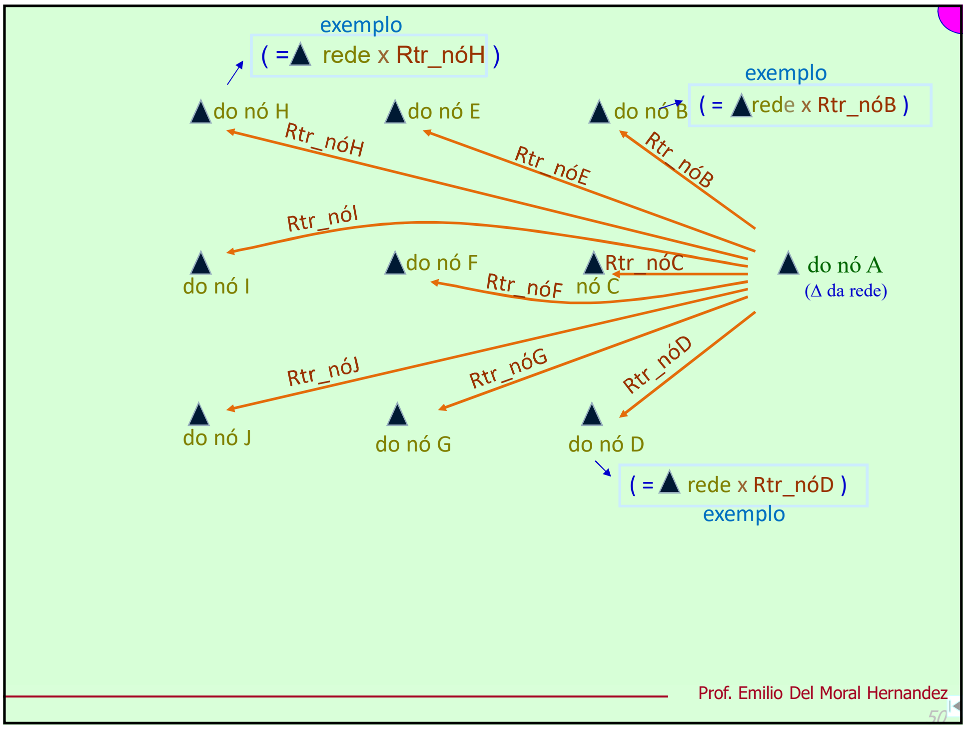
$= (2.\ [\boxed{erro\_rede\ .\ FatorRetroprop\_nó}\ ]\ .\ (\partial y\_nó / \partial w))$

Definição de ***erro de nó***

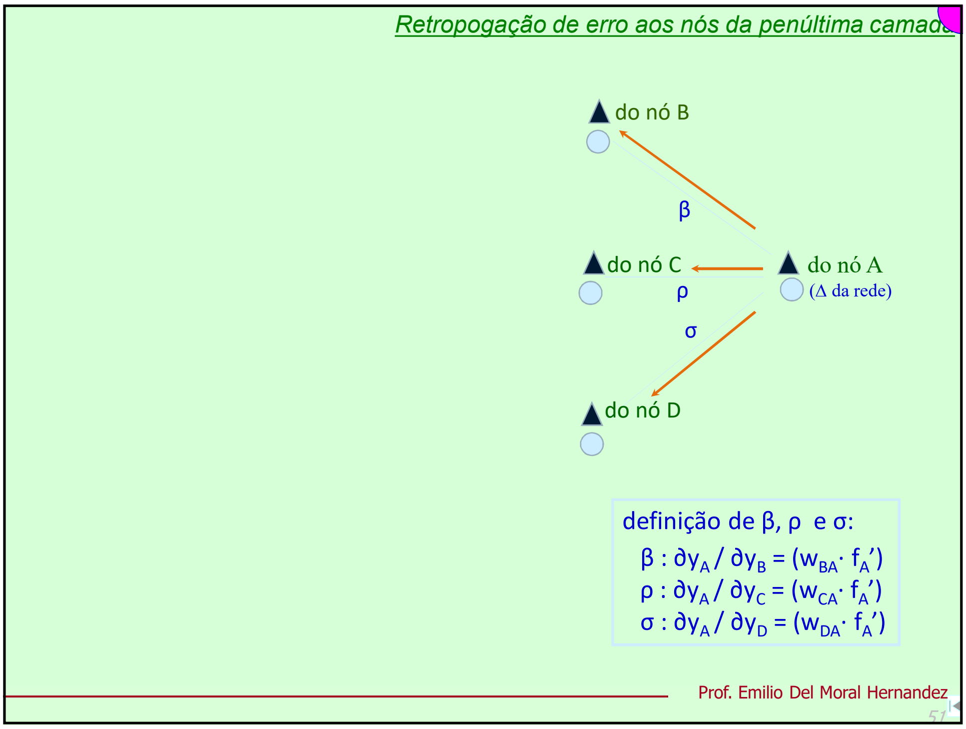
(= erro de rede retroprogado ao nó)



Os triângulos representam os erros associados a cada nó

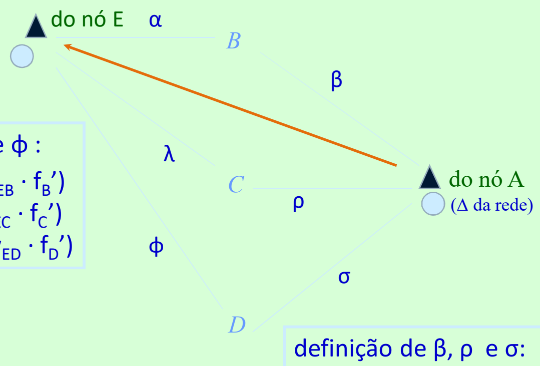

exemplo
(=▲ rede x Rtr_nóH)
▲do nó H
▲do nó E
▲do nó B
exemplo
(= ▲rede x Rtr_nóB)
Rtr_nóH
Rtr_nóE
Rtr_nóB
Rtr_nóI
▲ do nó I
▲do nó F
▲Rtr_nóC
nó C
Rtr_nóF
▲ do nó A
(Δ da rede)
Rtr_nóJ
Rtr_nóG
Rtr_nóD
▲ do nó J
▲ do nó G
▲ do nó D
(= ▲ rede x Rtr_nóD)
exemplo
Prof. Emilio Del Moral Hernandez

50


Retropogação de erro aos nós da penúltima camada
▲do nó B
β
▲do nó C
ρ
▲ do nó A
(Δ da rede)
σ
▲do nó D
definição de β, ρ e σ:
$\beta : \partial y_A / \partial y_B = (w_{BA} \cdot f_A')$
$\rho : \partial y_A / \partial y_C = (w_{CA} \cdot f_A')$
$\sigma : \partial y_A / \partial y_D = (w_{DA} \cdot f_A')$
Prof. Emilio Del Moral Hernandez

51

Retropogação de erro envolvendo cadeia com trifurcação



definição de α, λ e ϕ :

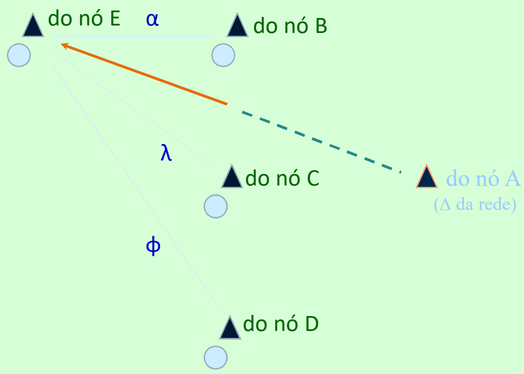
$\alpha : \partial y_B / \partial y_E = (w_{EB} \cdot f_B')$
$\lambda : \partial y_C / \partial y_E = (w_{EC} \cdot f_C')$
$\phi : \partial y_D / \partial y_E = (w_{ED} \cdot f_D')$

definição de β, ρ e σ:

$\beta : \partial y_A / \partial y_B = (w_{BA} \cdot f_A')$
$\rho : \partial y_A / \partial y_C = (w_{CA} \cdot f_A')$
$\sigma : \partial y_A / \partial y_D = (w_{DA} \cdot f_A')$

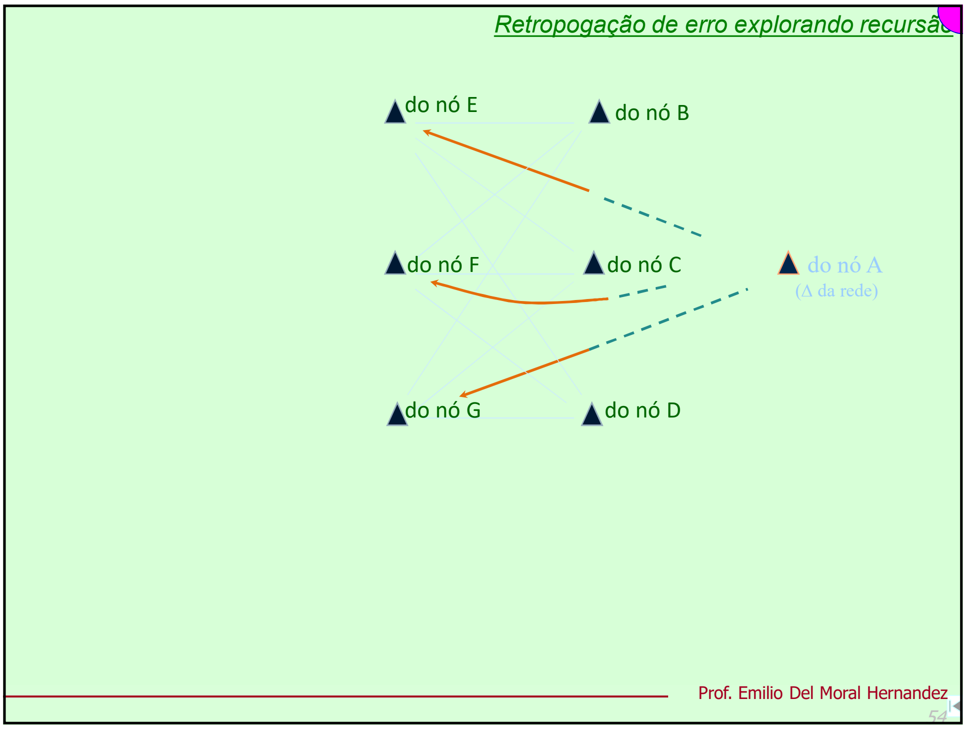
erro E = (α· erro da rede · β) + (λ· erro da rede . ρ) +(ϕ·erro da rede . σ)

Retropogação de erro explorando recursão

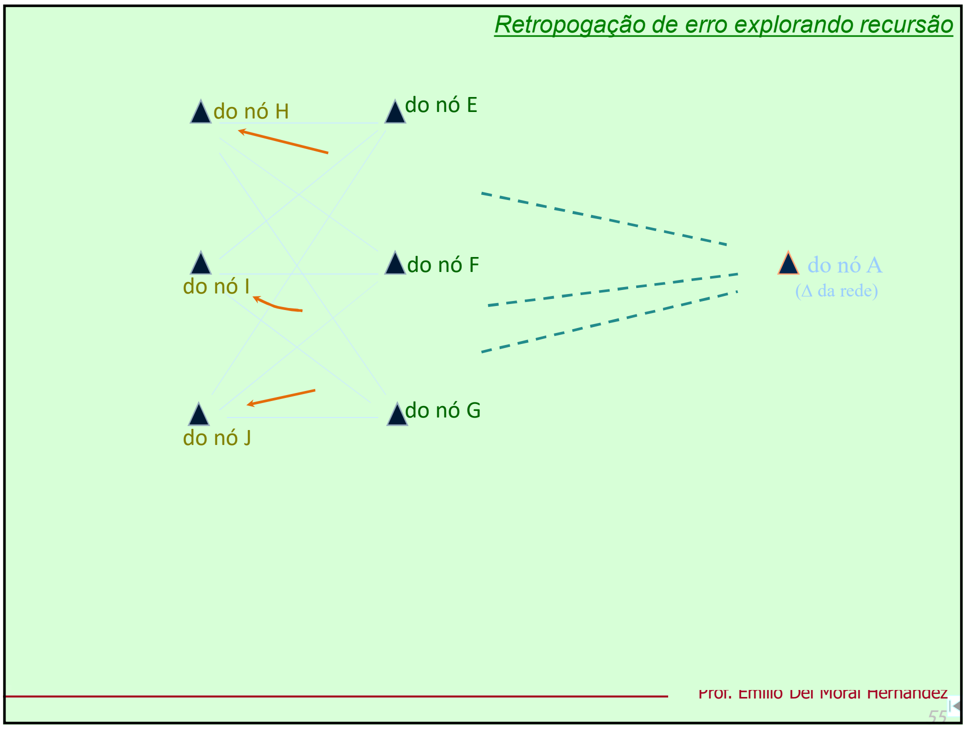


erro E = α·(erro da rede · β) + λ·(erro da rede . ρ) + ϕ·(erro da rede . σ)
= α·(erro B) + λ·(erro C) + ϕ·(erro D)

erro E = (α· erro da rede · β) + (λ· erro da rede . ρ) +(ϕ·erro da rede . σ)

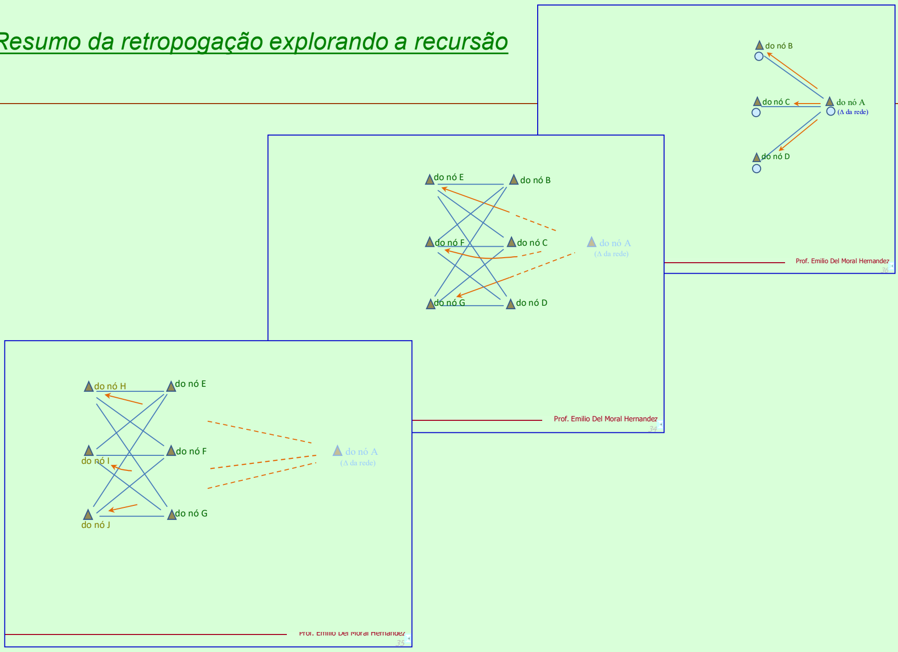

Retropogação de erro explorando recursão
do nó E
do nó B
do nó F
do nó C
do nó A
(Δ da rede)
do nó G
do nó D
Prof. Emilio Del Moral Hernandez

54


Retropogação de erro explorando recursão
do nó H
do nó E
do nó I
do nó F
do nó A
(Δ da rede)
do nó J
do nó G
Prof. Emilio Del Moral Hernandez

55

Resumo da retropogação explorando a recursão



57

Aula 10

(de 15)

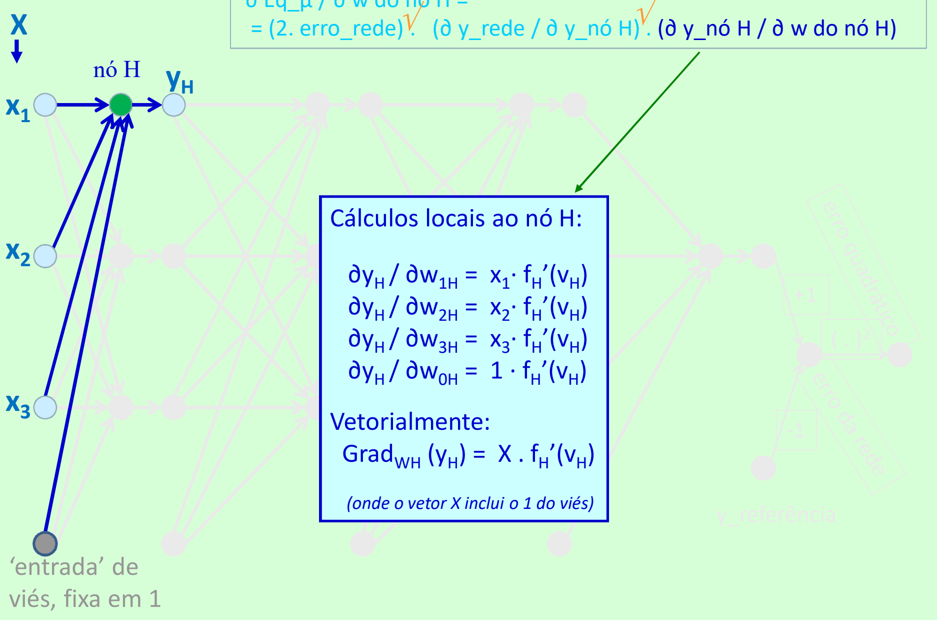
do Prof. Emilio

de PSI3471 de

27-mar-2019

se encerrou aqui

58



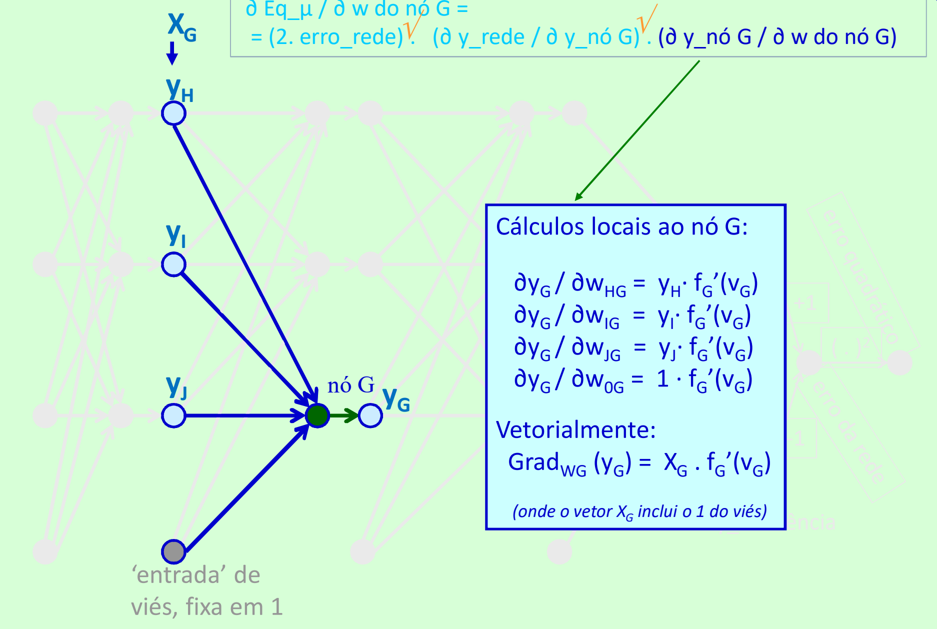
∂ Eq_µ / ∂ w do nó H =
= (2. erro_rede)√. (∂ y_rede / ∂ y_nó H)√. (∂ y_nó H / ∂ w do nó H)

Cálculos locais ao nó H:

$\partial y_H / \partial w_{1H} = x_1 \cdot f_H'(v_H)$
$\partial y_H / \partial w_{2H} = x_2 \cdot f_H'(v_H)$
$\partial y_H / \partial w_{3H} = x_3 \cdot f_H'(v_H)$
$\partial y_H / \partial w_{0H} = 1 \cdot f_H'(v_H)$

Vetorialmente:
$Grad_{WH}(y_H) = X \,.\, f_H'(v_H)$

(onde o vetor X inclui o 1 do viés)



∂ Eq_µ / ∂ w do nó G =
= (2. erro_rede)√. (∂ y_rede / ∂ y_nó G)√. (∂ y_nó G / ∂ w do nó G)

Cálculos locais ao nó G:

$\partial y_G / \partial w_{HG} = y_H \cdot f_G'(v_G)$
$\partial y_G / \partial w_{IG} = y_I \cdot f_G'(v_G)$
$\partial y_G / \partial w_{JG} = y_J \cdot f_G'(v_G)$
$\partial y_G / \partial w_{0G} = 1 \cdot f_G'(v_G)$

Vetorialmente:
$Grad_{WG}(y_G) = X_G \,.\, f_G'(v_G)$

(onde o vetor X_G inclui o 1 do viés)

Método do Gradiente Aplicado aos nossos MLPs: a partir de um W#0, temos aproximações sucessivas ao Eqm mínimo, por repetidos pequenos passos DeltaW, sempre contrários ao gradiente ...

- "Chute" um W inicial para o "Wcorrente", ou "W melhor até agora"
- Em loop até obter Eqm zero, ou baixo o suficiente, ou estável:
  - Determine o vetor gradiente do Eqm, nesse espaço de Ws
  - Em loop varrendo todos os M exemplos $(X^{\mu};y^{\mu})$,
    - Calcule o gradiente de Eq^{μ} associado a um exemplo μ, e vá varrendo μ e somando os gradientes de cada Eq^{μ}, para compor o vetor gradiente de Eqm, assim que sair deste loop em μ ;
    - Cada cálculo como esse, envolve primeiro calcular os argumentos de cada tangente hiperbólica e depois usar esses argumentos na regra da cadeia das derivadas necessárias
  - Tire a média dos M gradientes individuais e dê um passo Delta ΔW nesse espaço, com direção e magnitude dados por $-\eta$*vetor gradiente (Eqm)

Prof. Emilio Del Moral Hernandez

61

Concluindo ... Processo de aproximações sucessivas ao Eqm mínimo:

- Chutamos um vetor W inicial, escolhido randomicamente
- Tipicamente essa escolha randômica leva a um Eqm alto e portanto insatisfatório. Para melhorar o Eqm, queremos calcular Grad (Eqm) com a finalidade de adaptar W, "caminhando" contrariamente ao gradiente do erro: Wnovo = Wcorrente $-\eta$.Grad (Eqm)
- Loop Externo: Loop do Gradiente Descendente
  - Loop Interno: varredura dos M exemplares de treino. Para o cálculo de Grad (Eqm), usamos um Loop de varredura dos M exemplares de entrada / saída disponíveis para o treinamento da rede:
    - Para chegarmos a Grad (Eqm), calculamos M vetores gradiente, um para cada $(X^{\mu};y^{\mu})$. Devemos obter M vetores Grad (Eq^{μ}), um para cada exemplar entrada / saída $(X^{\mu};y^{\mu})$, varrendo pois todos os M exemplares disponíveis para treino da rede neural.
    - O Grad(Eq^{μ}) associado a apenas um exemplo específico $(X^{\mu};y^{\mu})$ é obtido como desenvolvido nas aulas anteriores do módulo, devemos calcular tantas derivadas parciais $\partial Eq^{\mu} / \partial w\_xx$ quantos pesos w_xx existam na rede.
    - Fazemos esses cálculos através, por exemplo, das técnicas de retropropagação e dos passos direto e reverso. Assim calculamos todas as componentes $\partial Eq^{\mu} / \partial w\_xx$ e compomos o gradiente de Eq^{μ}, referente a um μ específico (cada μ define um para entrada / saída de treino).
    - Cumulativamente, somamos esses M gradientes de exemplar, Grad (Eq^{μ}), para ao final da varredura de μ, chegarmos ao vetor médio Grad (Eqm)= Σ Grad (Eq^{μ})/M
  - Ao sairmos do Loop em μ acima (ao finalizarmos a varredura de todos os μ) chegamos ao vetor médio, Grad(Eqm), e estamos prontos para darmos um pequeno passo vetorial ΔW no espaço de pesos, com a direção e magnitude dados por $-\eta$.Grad (Eqm).
  - Assim adaptamos os pesos w's: Wnovo=Wanterior $-\eta$.Grad (Eqm)
  - Com essa mudança, devemos melhorar o Eqm, pois o reduzimos, incrementalmente.
  - Como o gradiente descendente opera por aproximações sucessivas, temos um Loop Externo: várias adaptações de W, Wnovo = Wcorrente $-\eta$.Grad (Eqm), são necessárias em sequência.
  - Critério de parada: seguimos no loop do Gradiente descendente até Eqm ser zero, baixo, ou estável (sem melhoras incrementais observadas), ou até estourarmos um número limíte de adaptações.
- *Ao sairmos do Loop Externo, temos "bons w´s" (que levam a erros Eqm baixos) e podemos usar o modelo neural para estimar novos valores de y, com conhecimento apenas dos x's.*

Prof. Emilio Del Moral Hernandez

62

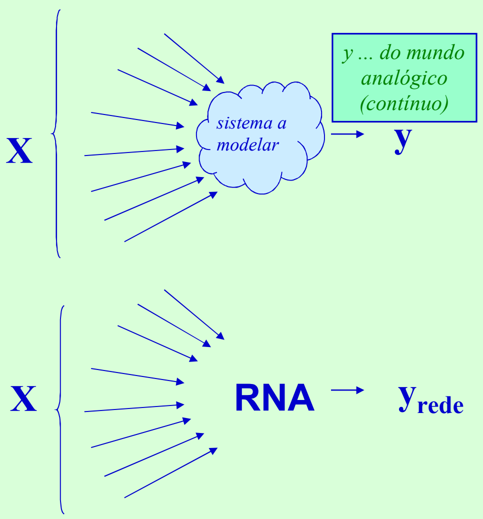
... E relembrando ... Para que serve uma RNA MLP com pesos treinados por EBP como aprendido nas aulas recentes ?? ... Serve por exemplo para modelar sistemas descritos por você nas atividades entregues de #1, #2 e #3.

Nota: lembremos também do poder do MLP conferido pela aproximação universal de Cybenko + considerações práticas que necessariamente relaxam a premissa ideal de um sistema regido por uma função exata ligando X e y, conforme slides que seguem

Prof. Emilio Del Moral Hernandez

63

Modelagem de um sistema por função de mapeamento X -> y (a RNA como regressor contínuo não linear multivariável)

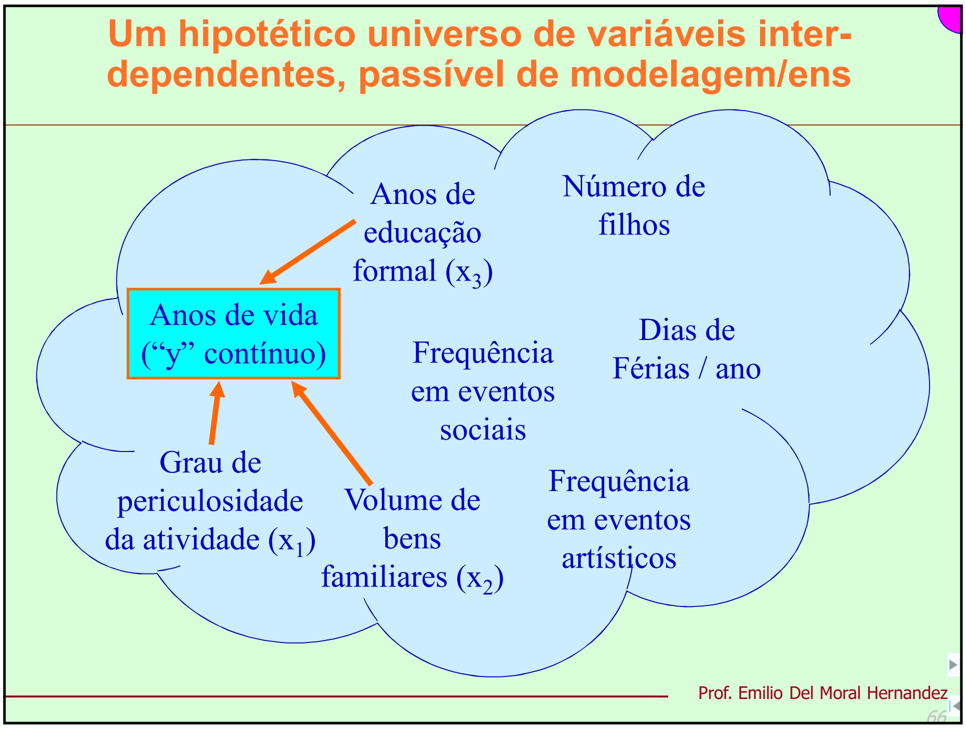


Assumimos que a variável y do sistema a modelar é uma função (normalmente desconhecida e possivelmente não linear) de diversas outras variáveis desse mesmo sistema

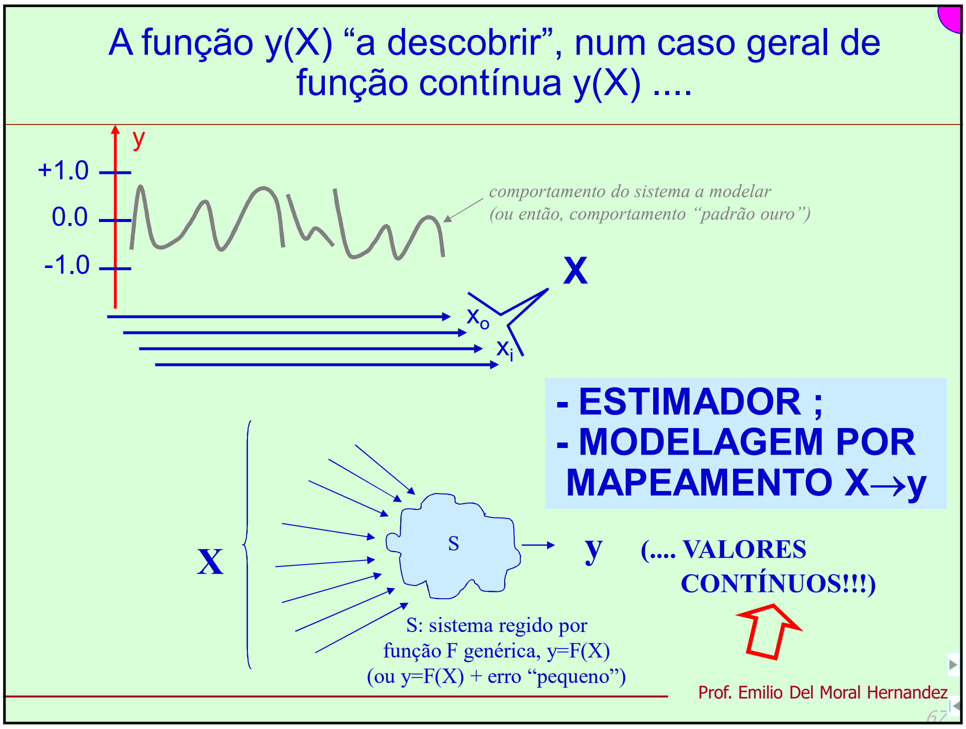
A RNA , para ser um bom modelo do sistema, deve reproduzir essa relação entre X e y, tão bem quanto possível

Prof. Emilio Del Moral Hernandez

65

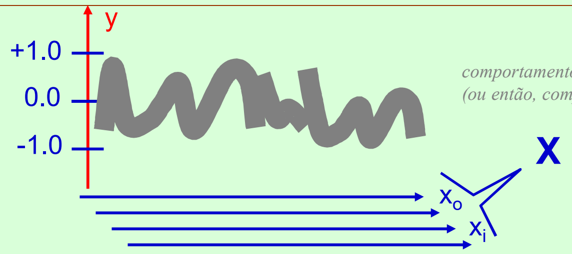

Um hipotético universo de variáveis inter-dependentes, passível de modelagem/ens
Anos de educação formal (x_3)
Número de filhos
Anos de vida (“y” contínuo)
Frequência em eventos sociais
Dias de Férias / ano
Grau de periculosidade da atividade (x_1)
Volume de bens familiares (x_2)
Frequência em eventos artísticos
Prof. Emilio Del Moral Hernandez

66


A função y(X) “a descobrir”, num caso geral de função contínua y(X) ....
y
+1.0
0.0
-1.0
comportamento do sistema a modelar
(ou então, comportamento “padrão ouro”)
X
x_o
x_i
- ESTIMADOR ;
- MODELAGEM POR MAPEAMENTO X→y
X
S
y
(.... VALORES CONTÍNUOS!!!)
S: sistema regido por função F genérica, y=F(X)
(ou y=F(X) + erro “pequeno”)
Prof. Emilio Del Moral Hernandez

67

Cenário mais real: a "função" y(X) do sistema modelado é "difusa": $y=F_{médio}(X)+ flutuação$ ....



comportamento do sistema a modelar
(ou então, comportamento "padrão ouro")

.... em problemas concretos / reais, há sempre alguma ambiguidade no mapeamento que leva valores de X a valores de y. Para decepção de Cybenko, não temos uma função y=F(X) no sentido matemático exato, pois para uma dada ênupla de valores X fixados, temos tipicamente uma faixa de valores que podem ser observados para a variável y: $y=F_{médio}(X)+ flutuação$.

Neste cenário, buscamos que o modelo capture o comportamento médio das relações observadas entre X e y:

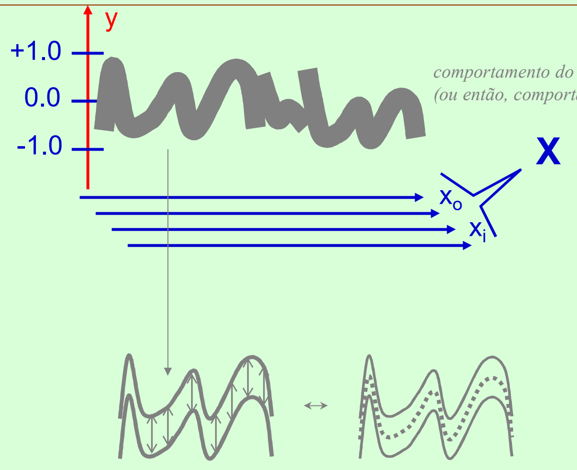
... y_{rede} ~ ymédio esperado para um dado X

Prof. Emilio Del Moral Hernandez

68

68

Cenário mais real: a "função" y(X) do sistema modelado é "difusa": $y=F_{médio}(X)+ flutuação$ ....



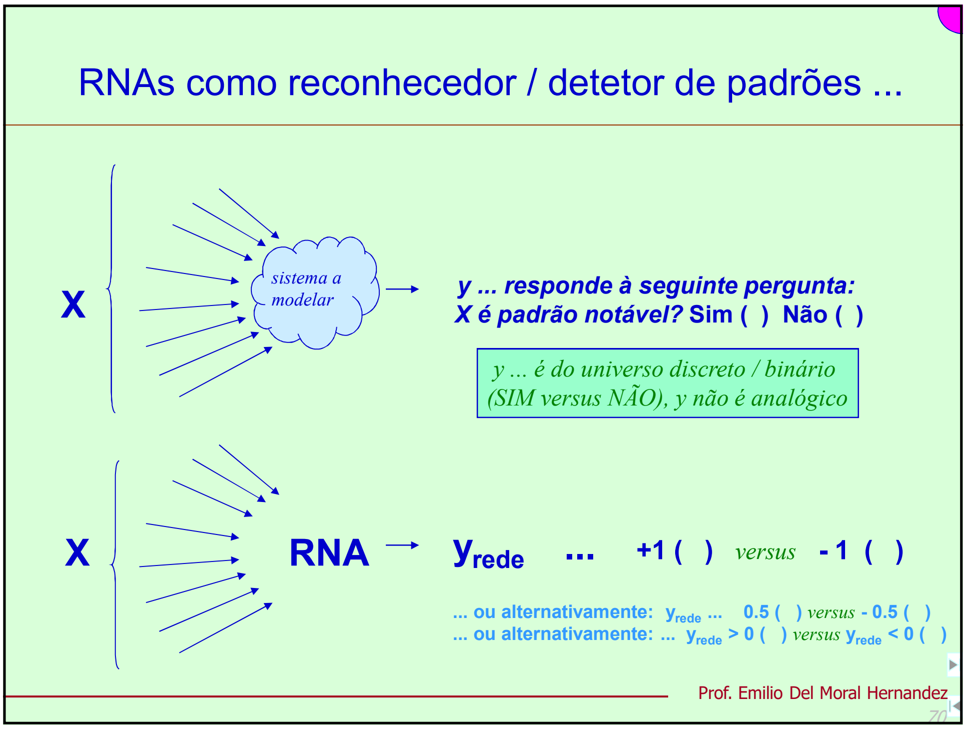
comportamento do sistema a modelar
(ou então, comportamento "padrão ouro")

... neste cenário, podemos olhar para o y observado no sistema que se deseja modelar não mais como um valor específico bem definido, mas como um valor médio esperado (dado valor de X) e uma faixa de valores em torno desse valor médio esperado.

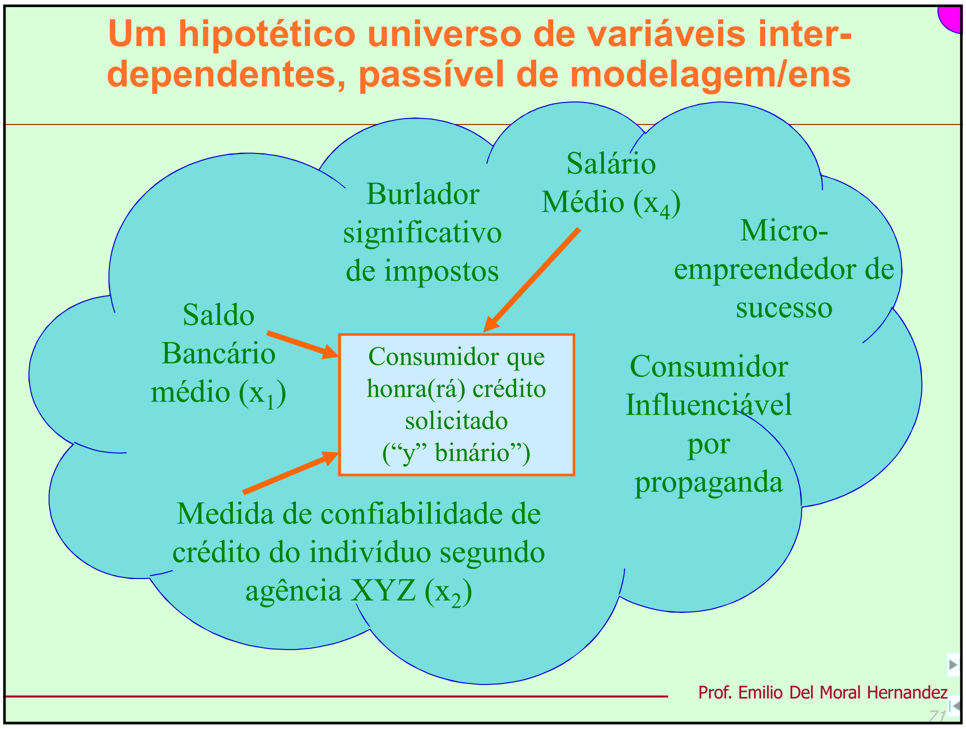
Prof. Emilio Del Moral Hernandez

69

69

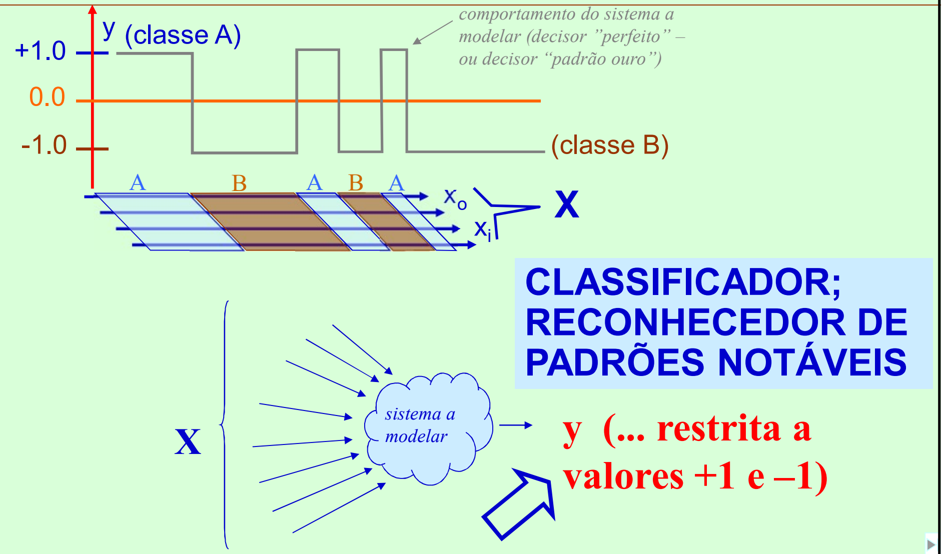

RNAs como reconhecedor / detetor de padrões ...
X
sistema a modelar
y ... responde à seguinte pergunta:
X é padrão notável? Sim ( ) Não ( )
y ... é do universo discreto / binário (SIM versus NÃO), y não é analógico
X
RNA
y rede ... +1 ( ) versus - 1 ( )
... ou alternativamente: y rede ... 0.5 ( ) versus - 0.5 ( )
... ou alternativamente: ... y rede > 0 ( ) versus y rede < 0 ( )
Prof. Emilio Del Moral Hernandez

70


Um hipotético universo de variáveis inter-dependentes, passível de modelagem/ens
Burlador significativo de impostos
Salário Médio (x4)
Micro-empreendedor de sucesso
Saldo Bancário médio (x1)
Consumidor que honra(rá) crédito solicitado ("y" binário")
Consumidor Influenciável por propaganda
Medida de confiabilidade de crédito do indivíduo segundo agência XYZ (x2)
Prof. Emilio Del Moral Hernandez

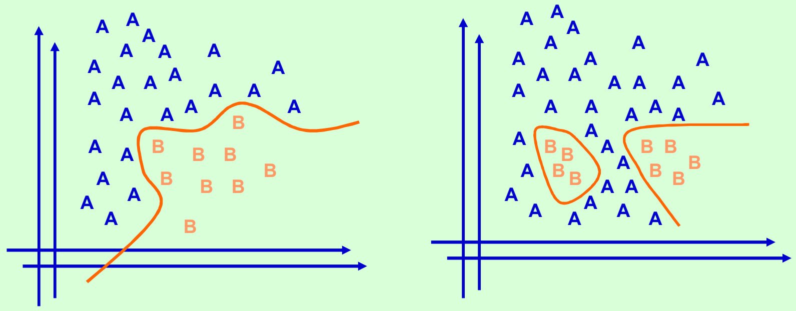
71

Caso de classificação binária / reconhecimento de padrões, será do tipo ...

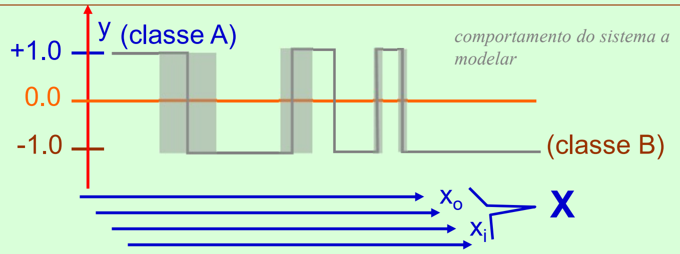


Capacidade de reconhecimento de padrões em casos complexos NÃO LINEARES

Com as RNAs, a hipersuperfície de separação entre classes vai muito além dos hiperplanos



Cenário mais real: a separação entre regiões do espaço de X não é perfeitamente definida ....



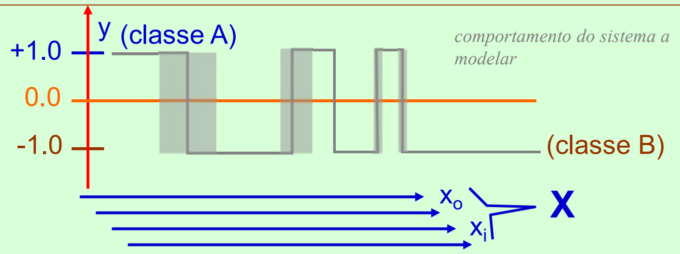
.... em problemas concretos / reais, há sempre alguma ambiguidade no mapeamento que leva valores de X aos valores discretos de y. Não temos uma função y=F(X) no sentido matemático exato, pois para uma dada ênupla de valores X fixado temos em alguns casos de fronteira a possibilidade de observar no y empírico tanto a classe A quanto a classe B: y=A ou B, com maior ou menor probabilidade para cada classe de acordo com o X. Neste desejamos que o modelo capture o comportamento médio das relações observadas entre X e y:

... y_{rede} ~ classe 'mais esperada' para um dado X

Prof. Emilio Del Moral Hernandez

74

Cenário mais real: a separação entre regiões do espaço de X não é perfeitamente definida ....



... podemos olhar para o y (classe A ou B) observado no sistema que se deseja modelar não mais como uma classe sempre bem definida e com fronteiras de separação entre A e B bem definidas no espaço de valores de X, mas como sendo delineadas na modelagem através de fronteiras com eventuais faixas de tolerância e com sobreposição parcial das classes no espaço de X

Prof. Emilio Del Moral Hernandez

75

Situações de classes com sobreposição parcial no espaço de atributos X ; situações de fronteiras de separação difusas ...