Um tema importante no curso:

Sobreapredizado / Sobreajuste

Conceito, entendimento da sua origem e formas de limitá-lo

recordando

Questões intrigantes, p/ esta aula e p/ pensar em casa ...

- *No que impacta escolhermos o "epslon" de Cybenko de alto valor? O que muda na estrutura de Cybenko com isso?*
- *No que impacta escolhermos o "epslon" de Cybenko de baixo valor?*
- *Como definimos o número de nós da primeira camada do MLP ? Isto pode ser definido a priori, antes de testar o seu desempenho? (por exemplo com base no número de entradas da rede e/ou com base no número de exemplares de treino M?)*
- *O que ganhamos e o que perdemos se escolhermos usar POUCOS nós na construção rede neural?*
- *O que ganhamos e o que perdemos se escolhermos usar MUITOS nós na construção da rede neural?*

recordando

A aproximação universal com RNAs do tipo MLP, segundo Cybenko (& Kolmogorov)

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

recordando

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

Mathematics of Control, Signals, and Systems

© 1989 Springer-Verlag New York Inc.

Approximation by Superpositions of a Sigmoidal Function*

G. Cybenko†

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

Key words. Neural networks, Approximation, Completeness.

1. Introduction

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

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

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

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

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

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

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

4. Results for Other Activation Functions

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

stems and control, *IEEE Control*

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

l nets using the Radon transform,

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

us mappings by neural networks,

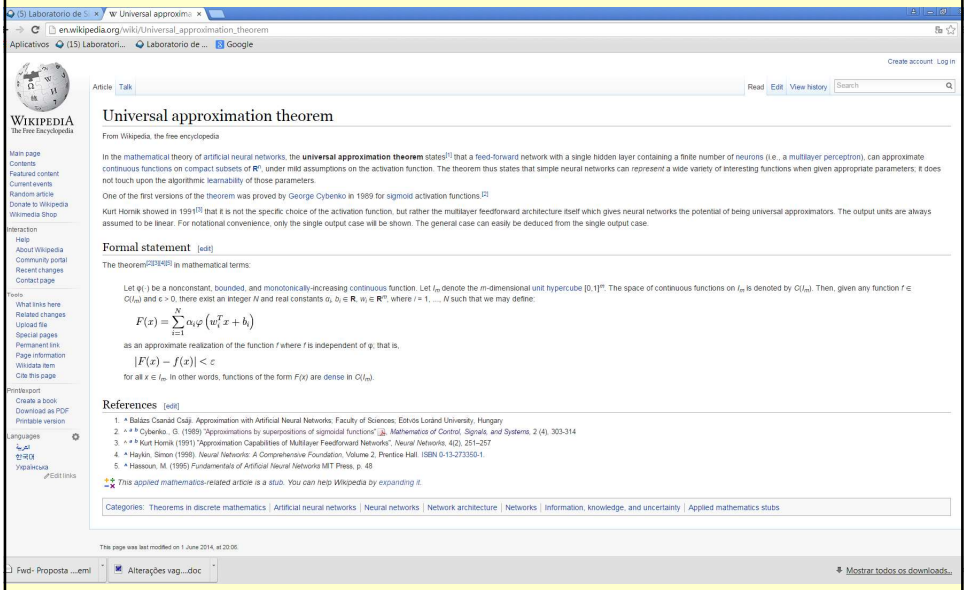
IEE Trans. Acoust. Speech Signal

dforward networks are universal

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

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

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



Universal approximation theorem

From Wikipedia, the free encyclopedia

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

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

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

Formal statement [edit]

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

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

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

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

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

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

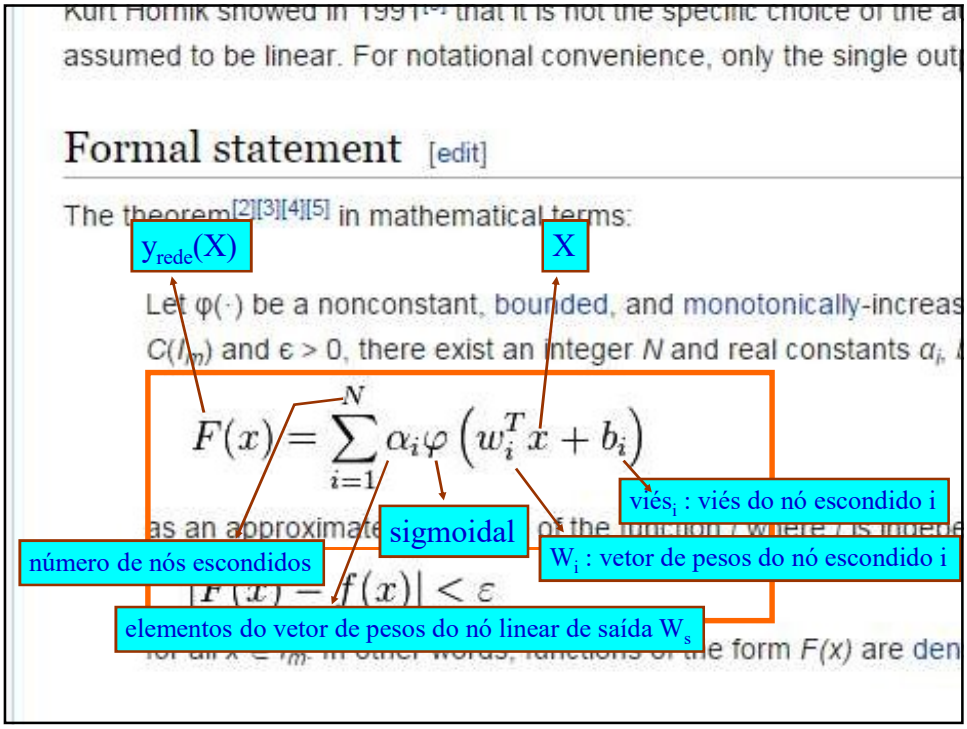
References [edit]

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

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

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

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



Kurt Hornik showed in 1991[6] that it is not the specific choice of the a
assumed to be linear. For notational convenience, only the single out

Formal statement [edit]

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

Let φ(·) be a nonconstant, bounded, and monotonically-increas
$C(I_m)$ and ε > 0, there exist an integer N and real constants α_i,

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

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

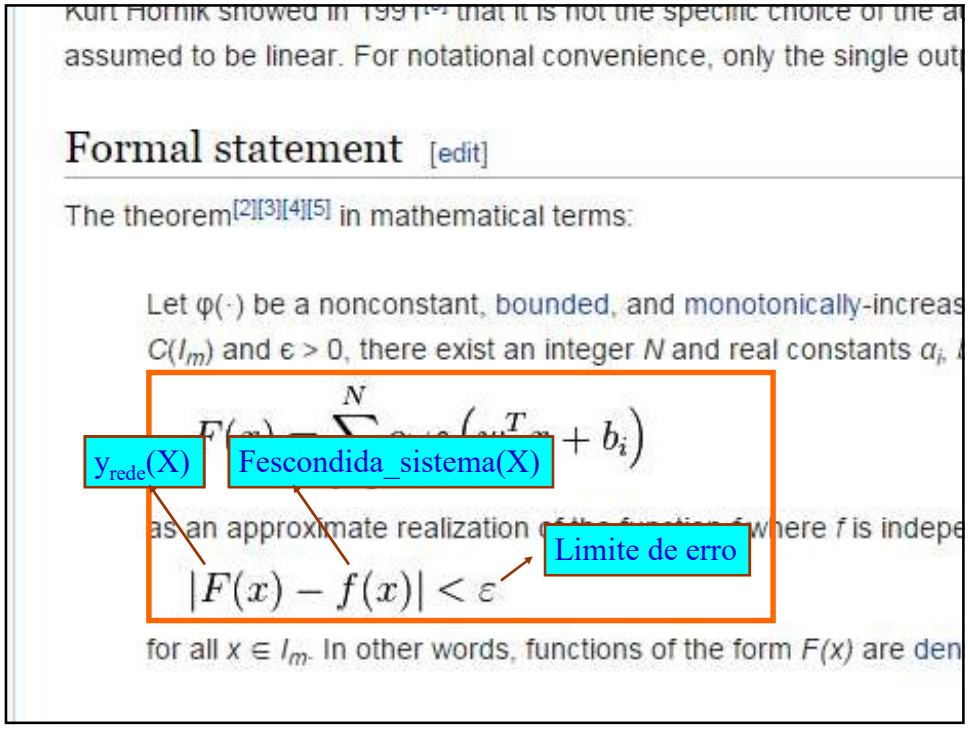
$$|F(x) - f(x)| < \varepsilon$$

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

$y_{rede}(X)$

Fescondida_sistema(X)

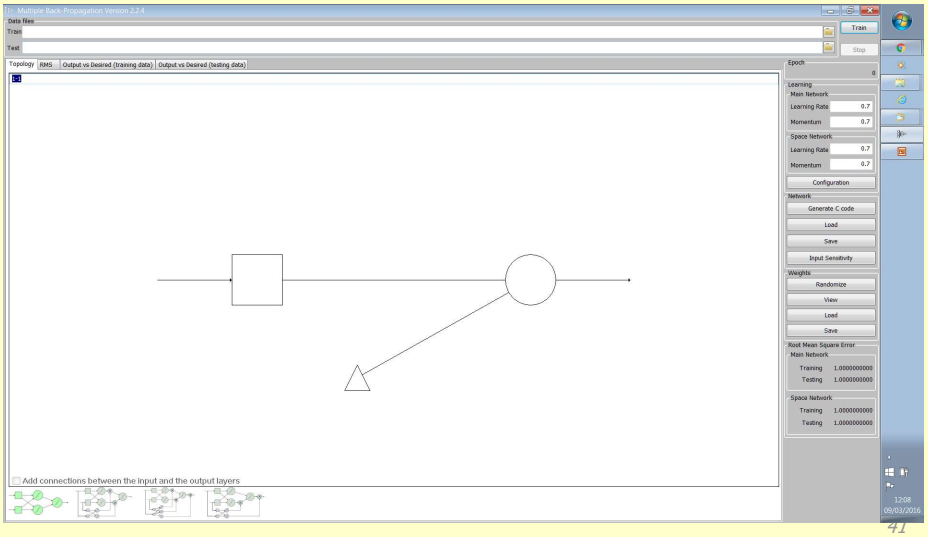
Limite de erro



... Um parênteses em lousa para discutirmos um pouco a aproximação universal com sigmoides / funções em formato de "S", no caso simples e bem particular univariado ... aproximação de uma função y de uma única variável x_1:

$$y(x_1)$$

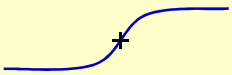
O que conseguimos fazer com um único neurônio sigmoidal, no caso de regressões (“y contínuo”)?



O que conseguimos fazer com um único neurônio sigmoidal $y(w_1.x_1)$ c/ escalamento de x_1 via w_1 e ***VIÉS NULO***

recordando

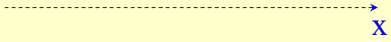
$w_1 = 1$



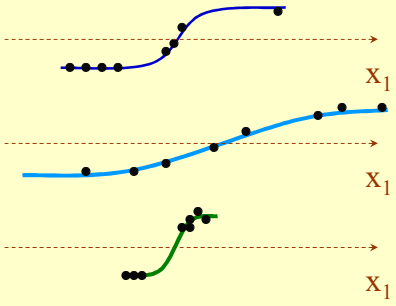
(esta é a tgh “de fábrica”: w_1 unitário e w_0 nulo)

$w_1 << 1$

$w_1 >> 1$



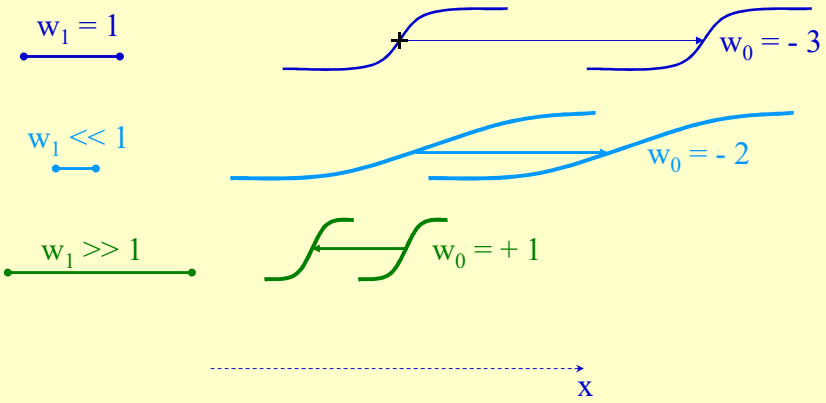
Que tipo de dados empíricos modelamos com um único neurônio sigmoidal em regressões ("y(x_1) contínuo")?



Os pontos pretos são pares empíricos (x^μ,y^μ); As curvas coloridas, são regressões sigmoidais aderentes a tais pares.

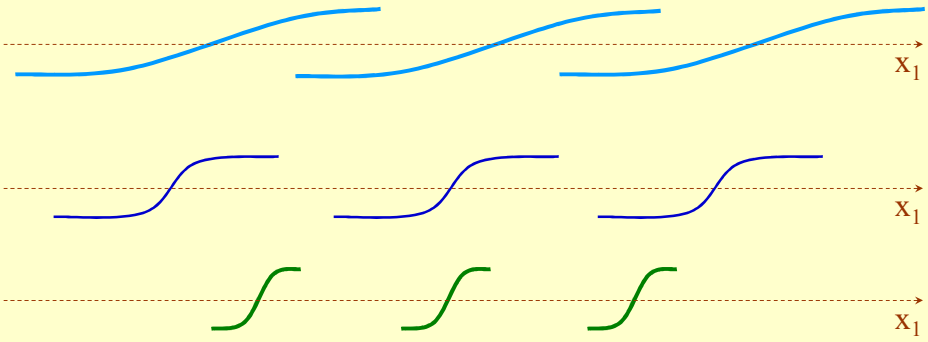
O que conseguimos fazer com um único neurônio sigmoidal y($w_1.x_1 + w_0.1$), c/ escalamento de x_1 via w_1

... e agora também com o viés, via viés w_0



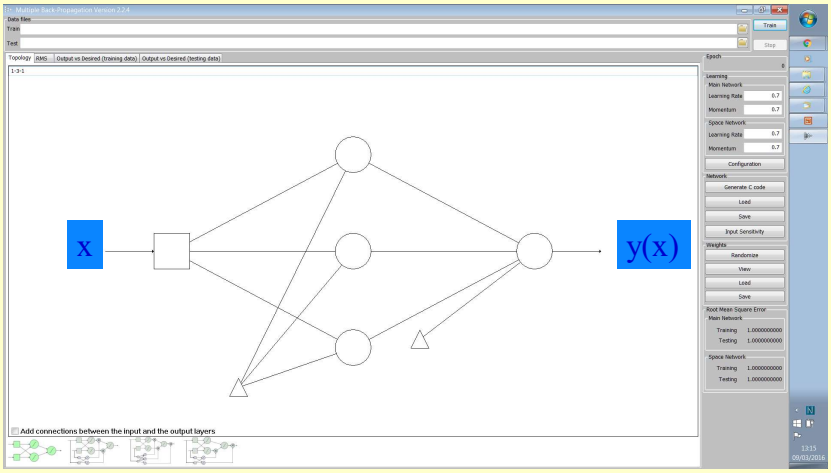
O que conseguimos fazer com um único neurônio sigmoidal, no caso de regressões ("$y(x_1)$ contínuo")?

recordando



$$y = tgh\ (w_1.x_1 + viés)$$

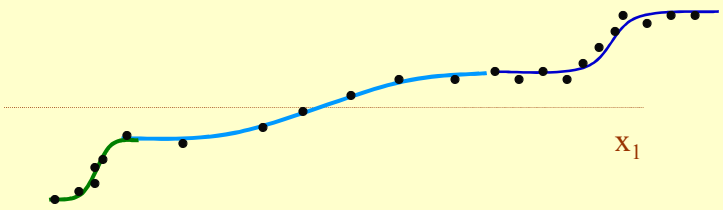
Regressão univariada com Cybenko "café com leite" de 3 nós na primeira camada ...



Cybenko "café com leite" (regressão genérica univariada), para aproximação universal de funções de 1 variável x_1 apenas?

recordando

... superposição de várias sigmóides deslocadas e escaladas



Vocês enxergam acima 3 nós "tgh" na primeira camada, com com 3 viéses distintos e 3 escaladores de x_1 distintos, e mais um 4o nó combinador (somatória simples de 3 entradas) na camada de saída?

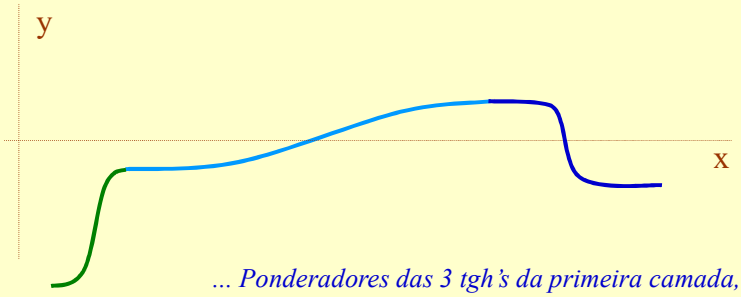
Algumas discussões adicionais sobre o Cybenko "café com leite" da regressão univariada ...

- *Vimos acima como se comporta o regressor univariado de Cybenko "café com leite" quando o nó de saída tem função de ativação identidade, seus pesos ponderadores das saídas da primeira camada são todos unitários positivos e o peso de viés é nulo.*
- *O que ocorre se os esses pesos ponderadores não forem mais unitários? (podem ser agora positivos, negativos, encolhedores (módulo < 1) ou amplificadores (módulo > 1))*
- *E se o seu peso de viés do 4o nó não for mais nulo?*

Cybenko "café com leite", para aproximação universal de funções de 1 variável x apenas?

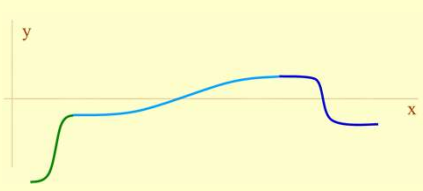
recordando

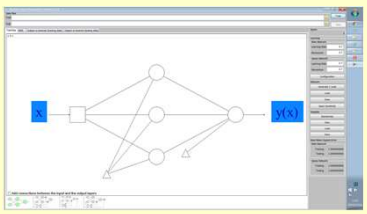
... superposição de várias sigmóides deslocadas e escaladas em x e em y ...



... Ponderadores das 3 tgh's da primeira camada, que são implemantados nos pesos sinápticos do 4o nó, não são mais unitários nem necessariamente positivos

Isto indica claramente que ao menos no caso de funções univariadas no domínio e na imagem (uma única variável x no argumento e uma única variável y na "saída" da função) uma RNA de duas camadas (com vários neurônios na segunda, não apenas 3 como ilustrado) pode aproximar qualquer função contínua univariada com erro bem pequeno se necessário: se desejado, basta usarmos mais e mais nós na segunda camada do MLP, aumentando assim arbitrariamente a precisão da aproximação da função alvo da modelagem.





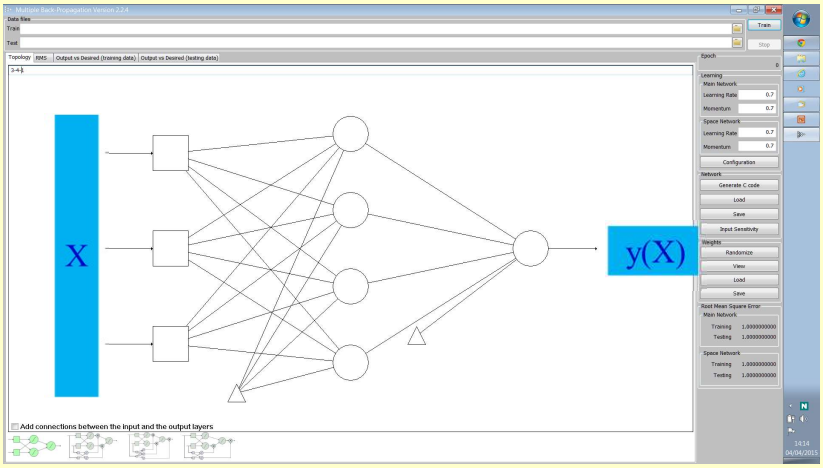
Cybenko foi além de mostrar a viabilidade de aproximação em casos unidimensionais, ele fez a prova de *Aproximação Universal* no âmbito de funções de múltiplas variáveis!

Qualquer Função(X) genérica pode ser aproximada por um MLP – O que é bom para Estimação / Regressão Contínua (um de alvos neste curso) !!!

E ...

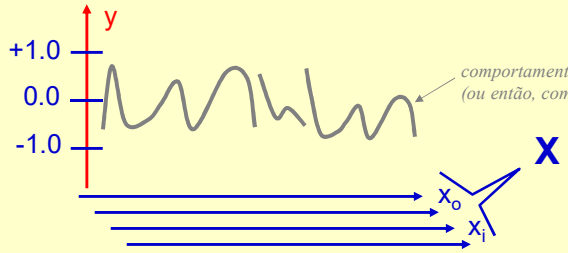
É também bom para o Reconhecimento de Padrões (nosso segundo alvo de modelagem) com o MLP ... Trata-se de um caso específico de função binária na saída !!!

Vamos neste caso para o terreno de vetor de entradas X em lugar de um x unidimensional

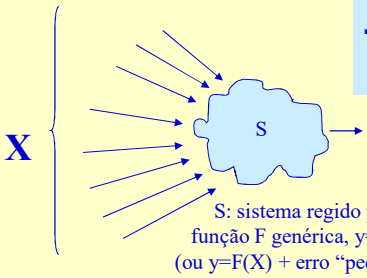


A função y(X) "a descobrir", num caso geral de função analógica y(X) ....

recordando



- ESTIMADOR ;
- MODELAGEM POR MAPEAMENTO X→y

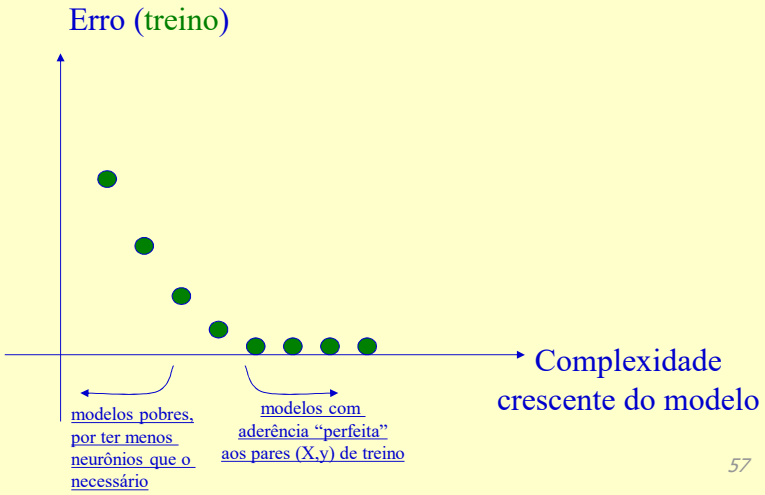


(.... VALORES CONTÍNUOS!!!)

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

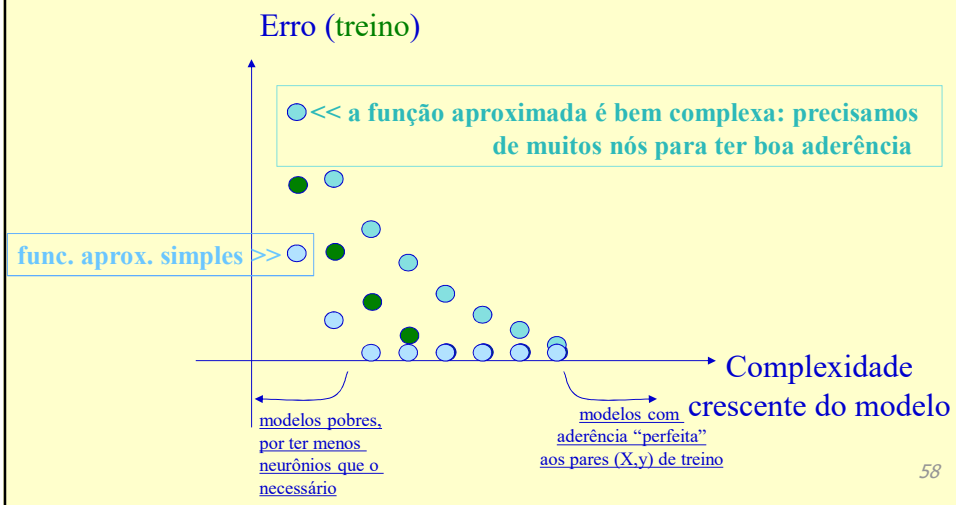
Aumento de aderência aos dados de treino com o aumento de nós da RNA ...

recordando



Aumento de aderência aos dados de treino com o aumento de nós da RNA ...

recordando



58

Isto quer dizer que sempre é melhor termos um modelo com mais nós neurais que um modelo com menos nós neurais?

Afinal, da mesma maneira que a computação de um regressor polinomial de grau seis engloba a computação dos regressores polinomiais de graus menores, os modelos com mais nós neurais englobam os mais simples (em termos de capacidades de computações possíveis) correto?

Sim, correto! Mas há um limite no "lucro" em tal estratégia, dado pelo fenômeno de

Sobreaprendizado e perda de generalização ...

60

O conceito de sobreaprendizado nos dará critérios adicionais para a definição do número de neurônios / grau de complexidade de uma rede neural

61

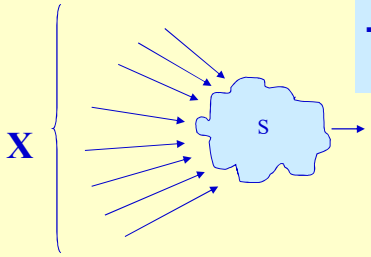
Inicialmente façamos algumas reflexões sobre o conjunto de treino em regressores e sua característica de amostra da totalidade de comportamentos (X;y) possíveis do sistema sendo modelado

63

Hipótese:

$$y = f(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10})$$

$$y = f(X)$$



- ESTIMADOR ;
- MODELAGEM POR MAPEAMENTO X→y

y (.... VALORES CONTÍNUOS!!!)

64

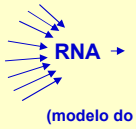
O treinamento mira minimizar o Eqm das amostras (X ; y) de treino. (exclusivamente!)

Relação de amostragem ...
... E lembremos que as amostras sempre são uma representação parcial do comportamento mais geral do sistema que está sendo modelado.

Tabela de amostras (X ; y)

$$\sum_{\mu} [\, y_{RNA}(X^{\mu}) - y_{sistema}{}^{\mu} \,]^2 \;/\; M$$

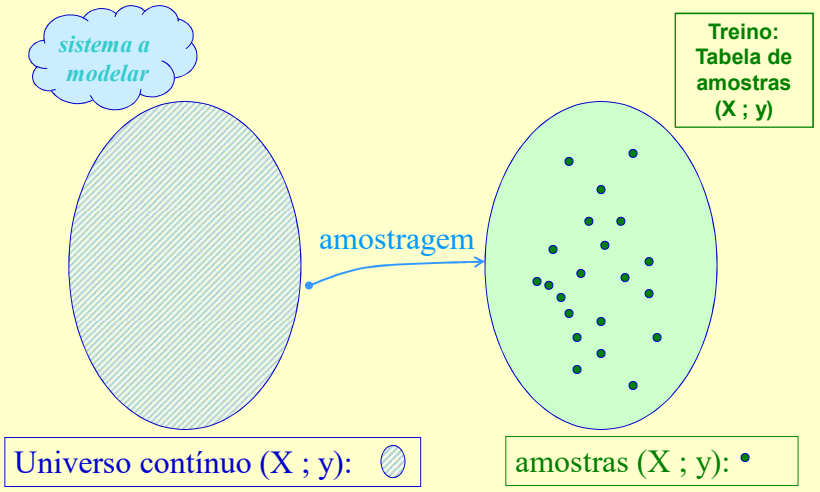
Eqm



(modelo do sistema)

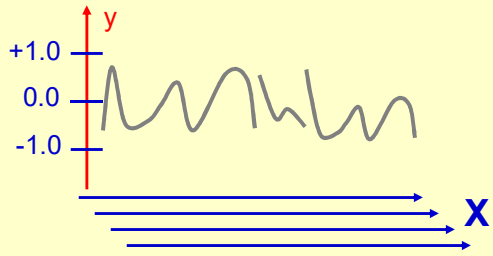
65

Conjunto de treino ... Composto de amostragens discretas do universo contínuo (X ; y)

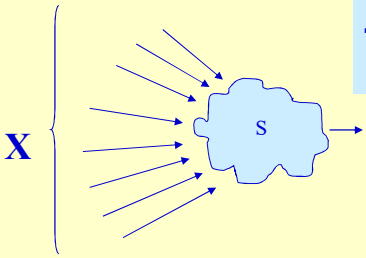


66

A função *f(X)* (interna ao sistema a modelar) é desconhecida e assume valores contínuos ....



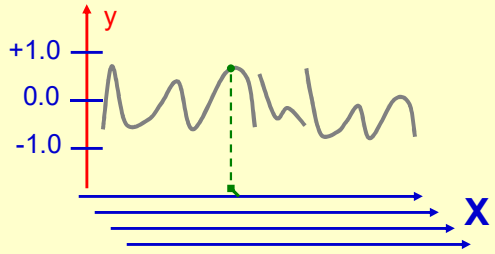
- ESTIMADOR ;
- MODELAGEM POR MAPEAMENTO X→y

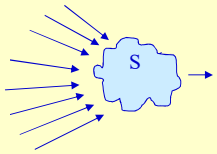


(.... VALORES CONTÍNUOS!!!)

68

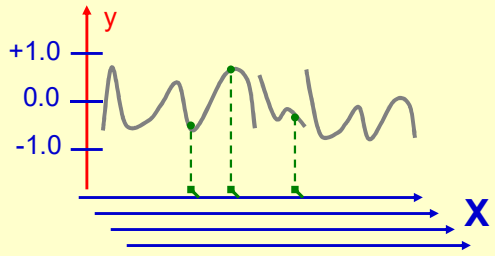
Mas podemos observar o sistema e registrar, por exemplo, um par (X;y) isolado ...

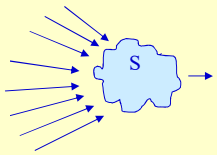




69

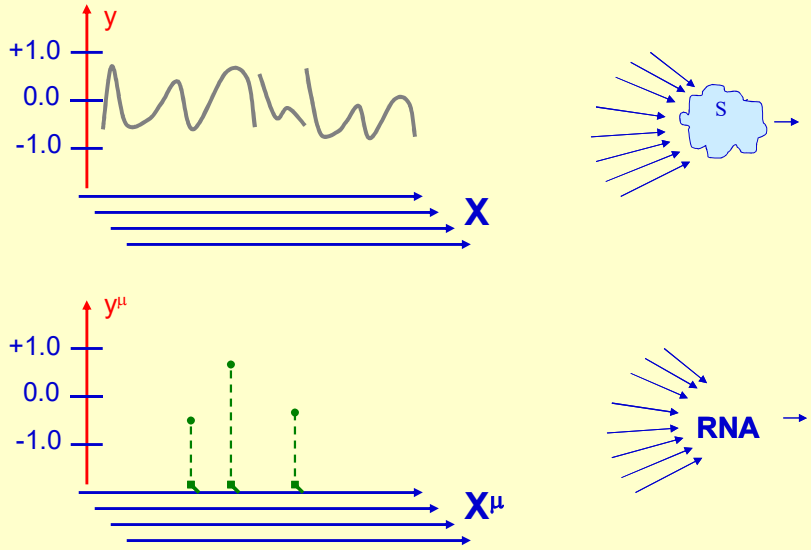
Ou mais pares (X;y) .... 3 pares, para começar





70

E podemos construir com esses 3 pares um conjunto de treino {(X^{μ} ; y^{μ})} para uma RNA ...



71

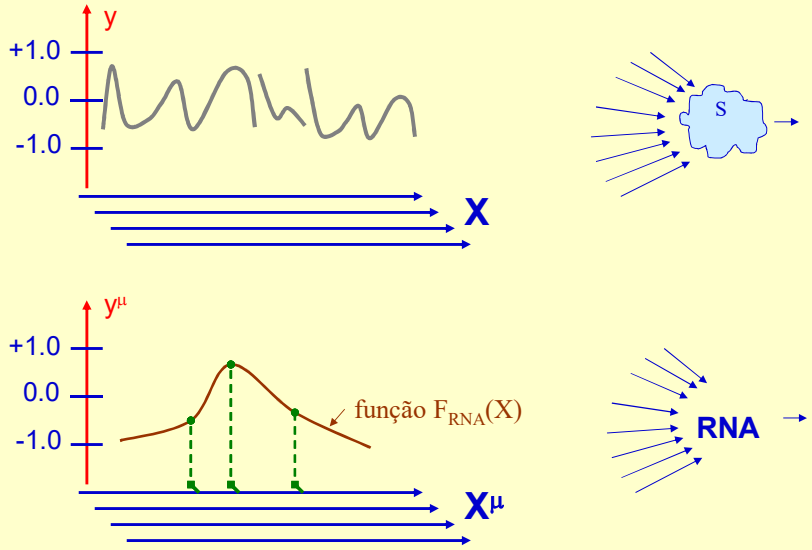
recordando

Gráfico fornecido pelo ambiente MBP da ev… …m com o número de repetidos usos da "bússola do … …endente": isto conecta o MBP com o gráfico apresen… …terior

Nota: o RMS do eixo vertical deste gráfico significa Root Mean Square, e é a raiz quadrada do nosso conhecido Eqm

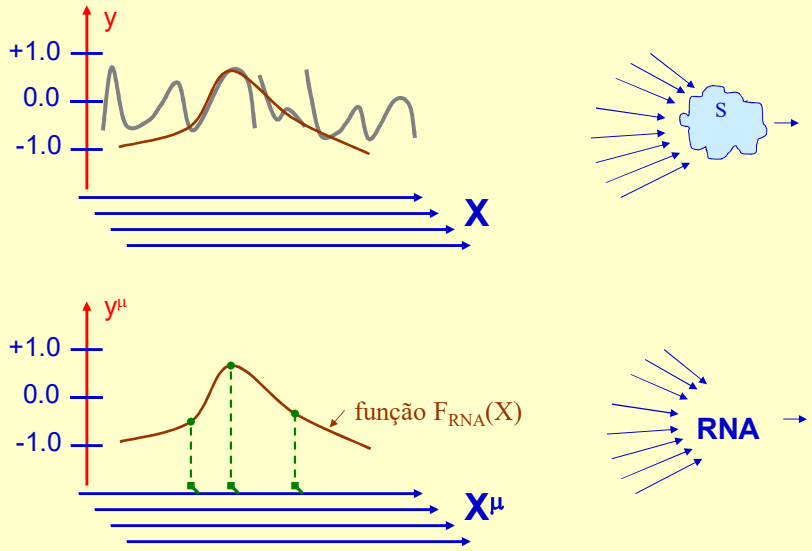
Gráfico mostrando as primeiras 47 iterações do processo de refinamentos sucessivos do modelo ...

Após treino por EBP e miniminação do Eqm, a RNA será um modelo fiel aos 3 pares ....



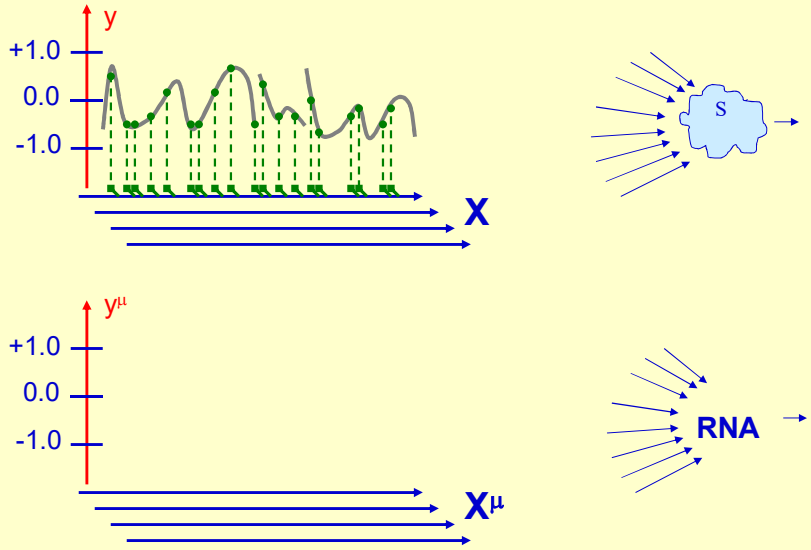
73

Mas não necessariamente fiel ao sistema sendo modelado ....



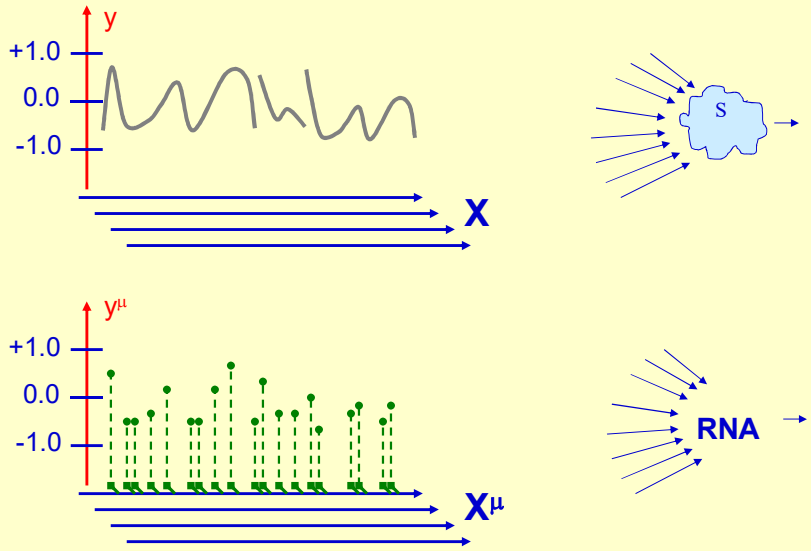
74

Melhor se tivermos mais pares, de forma a retratar melhor o sistema a modelar ....



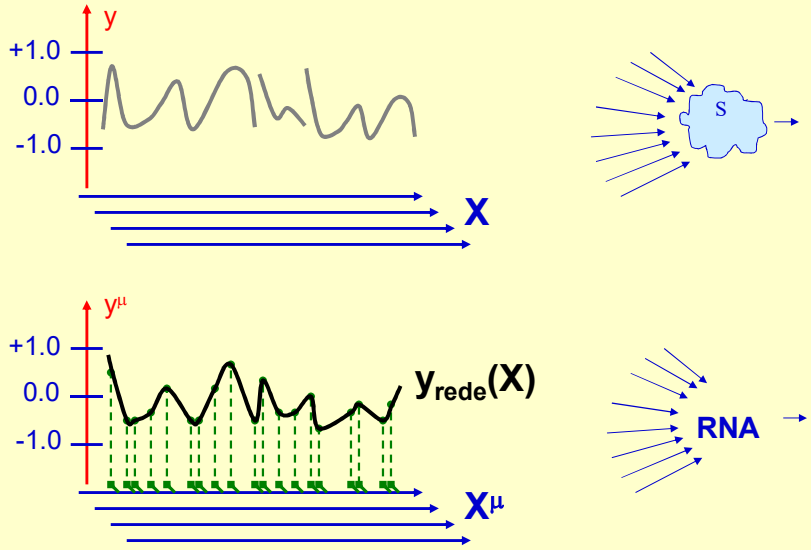
75

O tamanho M do conjunto de treino será maior ....



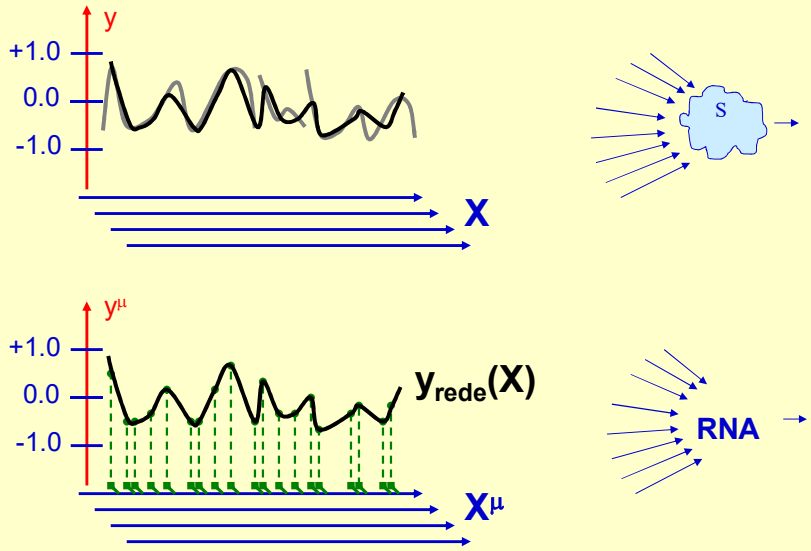
76

... e portanto o modelo será mais bem definido e mais próximo ao sistema modelado



77

... mas claro, nunca será totalmente idêntico ao sistema ....



78

E basta termos muitas amostras (X;y) – ou seja, alto valor de M – , e ter um ajuste bom a elas – ou seja, altonúmero de nós neurais – , e a modelagem será de qualidade?

Não é bem assim!!!! ...

O fenômeno de Sobreaprendizado / Sobreajuste, que ocorre em modelos mais complexos leva à perda de generalização (falha do modelo em novos cenários), mesmo quando o número de observações usadas no treino desse modelo (M) é bastante grande ...

79

Sobreaprendizado em polinômios:

Primeiro entendamos o conceito num universo mais familiar (e mais simples), o de regressão polinomial univariada, usada para representar dados com comportamento linear ou não linear e sujeitos a alguma flutuação em "y"

...

Depois, vocês mesmos podem pensar nos equivalentes dos nossos raciocínios feitos aqui para o universo de polinômios, mas para o universo de RNAs e mesmo de outros tipos de modelos com número de parâmetros variável (complexidade variável) que você conheça ...

82

Falemos em lousa um pouco sobre a reta média para um conjunto de pares (x,y), a parábola média, a cúbica média ... etc

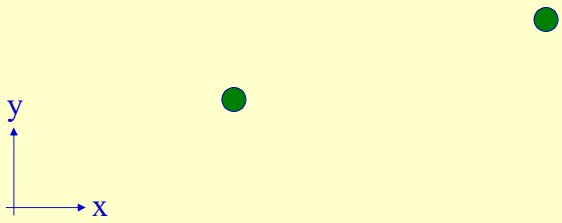
$$y \sim ax+b \; ; \quad y \sim ax^2 +bx +c \; ; \quad y \sim ax^3 +bx^2 +cx +d$$

e mais além, falemos sobre regressão polinomial univariada, com o grau do polinômio aproximador podendo ser 1, 2, 3, ou mesmo graus bastante mais altos como 50, 51 etc.

$$y \sim ax^{51} +bx^{50} +cx^{49} + ...$$

83

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

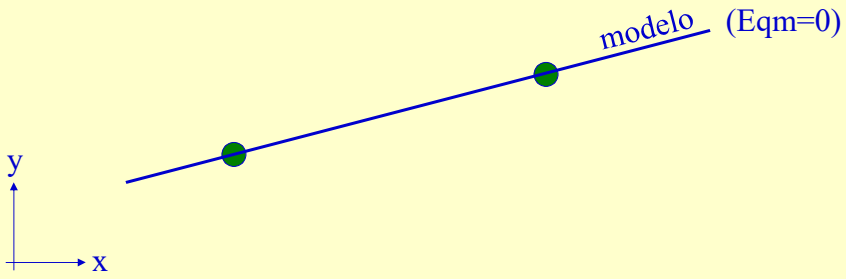


Os dados empíricos (x^{μ},y^{μ}) estão em verde;

84

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

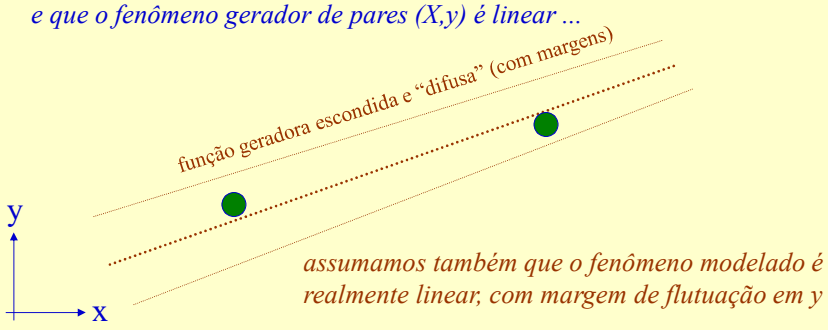
façamos uso de modelagem linear ...



Os dados empíricos (x^{μ},y^{μ}) estão em verde;
O modelo linear gerado a partir dos dados, em azul.

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

e que o fenômeno gerador de pares (X,y) é linear ...

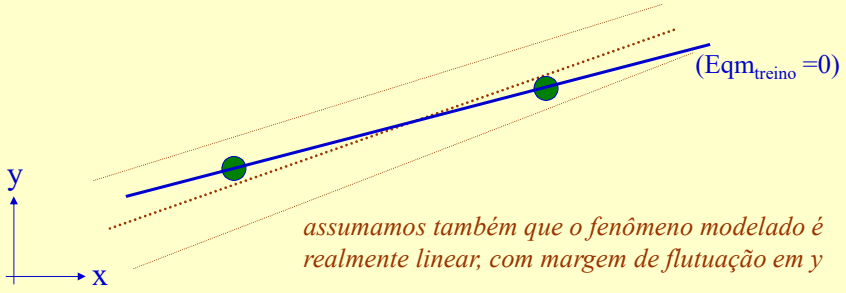


assumamos também que o fenômeno modelado é realmente linear, com margem de flutuação em y

Os dados empíricos (x^{μ},y^{μ}) estão em verde;
O modelo linear gerado a partir dos dados, em azul.
O fenômeno gerador de pares (x,y) é linear em essência, mas tem alguma flutuação randômica em y. A tendência e os limites da flutuação estão representados em marrom

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

façamos uso de modelagem linear ...

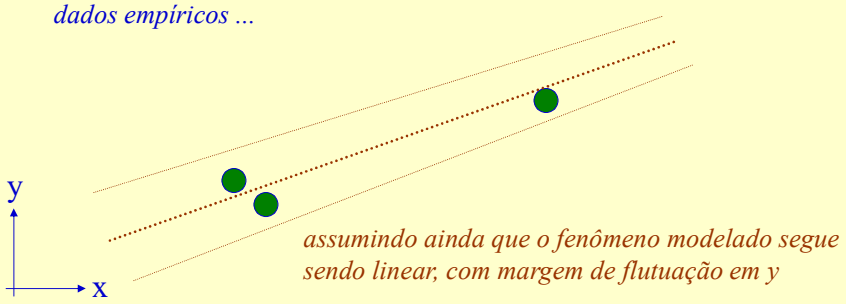


assumamos também que o fenômeno modelado é realmente linear, com margem de flutuação em y

Os dados empíricos (x^{μ},y^{μ}) estão em verde;
O modelo linear gerado a partir dos dados, em azul.
O fenômeno gerador de pares (x,y) é linear em essência, mas tem alguma flutuação randômica em y. A tendência e os limites da flutuação estão representados em marrom

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

Consideremos agora nova situação com mais dados empíricos ...

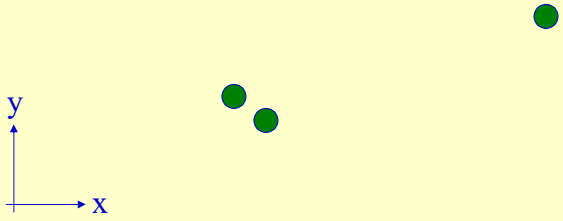


assumindo ainda que o fenômeno modelado segue sendo linear, com margem de flutuação em y

Os dados empíricos (x^{μ},y^{μ}) estão em verde;

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

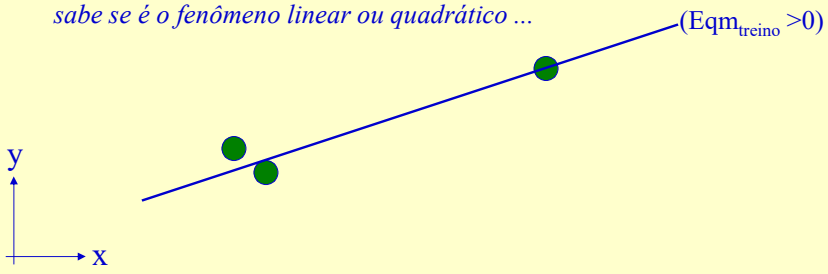
Consideremos agora nova situação com mais dados empíricos ... mas na modelagem não se sabe se é o fenômeno linear ou quadrático ...



Os dados empíricos (x^{μ},y^{μ}) estão em verde;

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

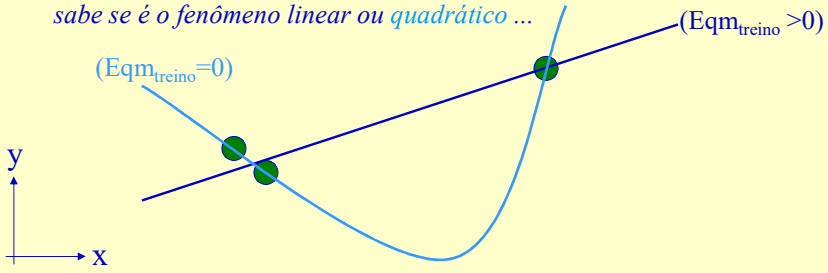
Consideremos agora nova situação com mais dados empíricos ... mas na modelagem não se sabe se é o fenômeno linear ou quadrático ...



Os dados empíricos (x^{μ},y^{μ}) estão em verde;

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

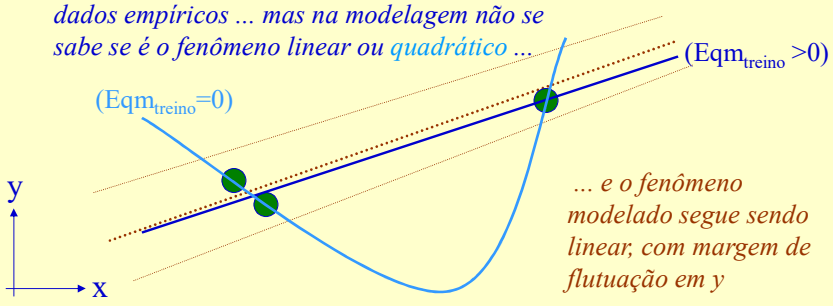
Consideremos agora nova situação com mais dados empíricos ... mas na modelagem não se sabe se é o fenômeno linear ou quadrático ...



Os dados empíricos (x^{μ},y^{μ}) estão em verde;
Dois modelos polinomiais gerados a partir dos dados, em azuis.
O fenômeno gerador de pares (x,y) é linear em essência, mas tem alguma flutuação randômica em y. A tendência e os limites da flutuação estão representados em marrom

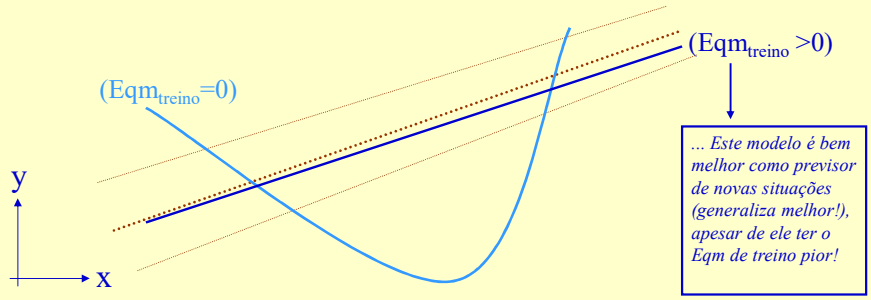
Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

Consideremos agora nova situação com mais dados empíricos ... mas na modelagem não se sabe se é o fenômeno linear ou quadrático ...



Os dados empíricos (x^{μ},y^{μ}) estão em verde;
Dois modelos polinomiais gerados a partir dos dados, em azuis.
O fenômeno gerador de pares (x,y) é linear em essência, mas tem alguma flutuação randômica em y. A tendência e os limites da flutuação estão representados em marrom

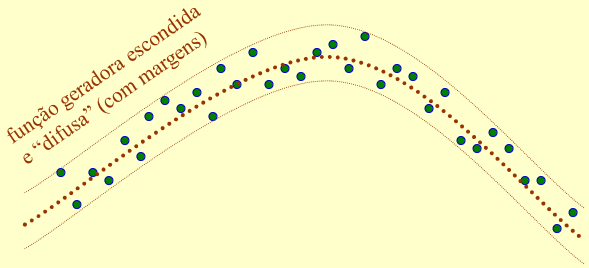
Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada



Os dados empíricos (x^{μ},y^{μ}) estão em verde;
Dois modelos polinomiais gerados a partir dos dados, em azuis. O fenômeno gerador de pares (x,y) é linear em essência, mas tem alguma flutuação randômica em y. A tendência e os limites da flutuação estão representados em marrom

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

um novo exemplo

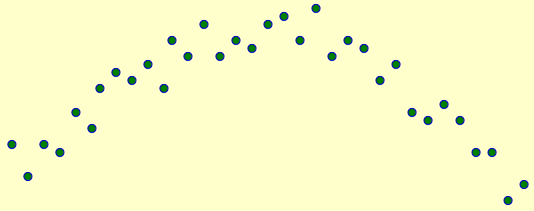


Os dados empíricos (x^{μ},y^{μ}) estão em verde;

O fenômeno gerador de pares (x,y) é quadrático em essência, mas tem alguma flutuação randômica em y. A tendência e os limites da flutuação estão representados em marrom

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

um novo exemplo

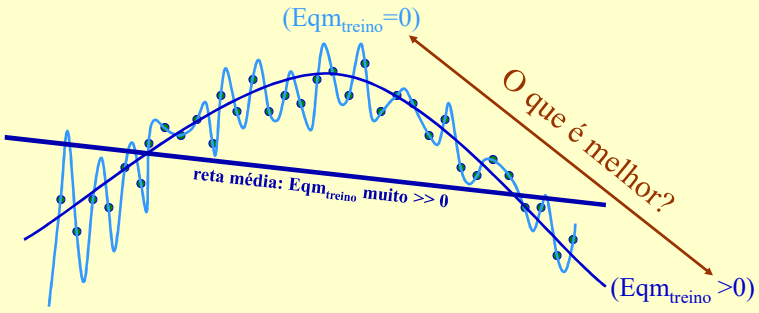


Os dados empíricos (x^{μ},y^{μ}) estão em verde;

O fenômeno gerador de pares (x,y) é quadrático em essência, mas tem alguma flutuação randômica em y. A tendência e os limites da flutuação estão representados em marrom

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

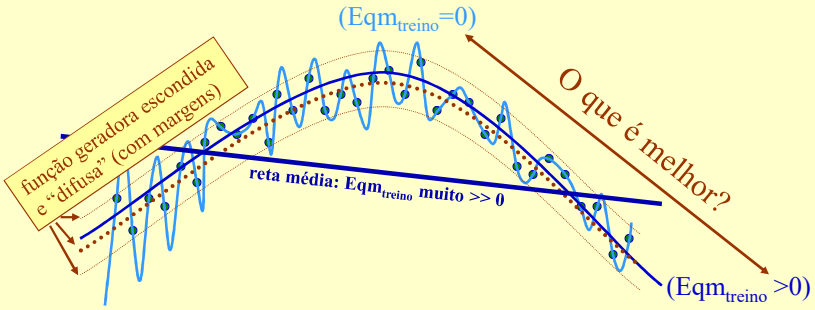
um novo exemplo



Os dados empíricos (x^{μ},y^{μ}) estão em verde;
Três modelos polinomiais gerados a partir dos dados, em azuis.

Sobreaprendizado ilustrado em sua fenomenologia, na regressão polinomial (de vários graus) univariada

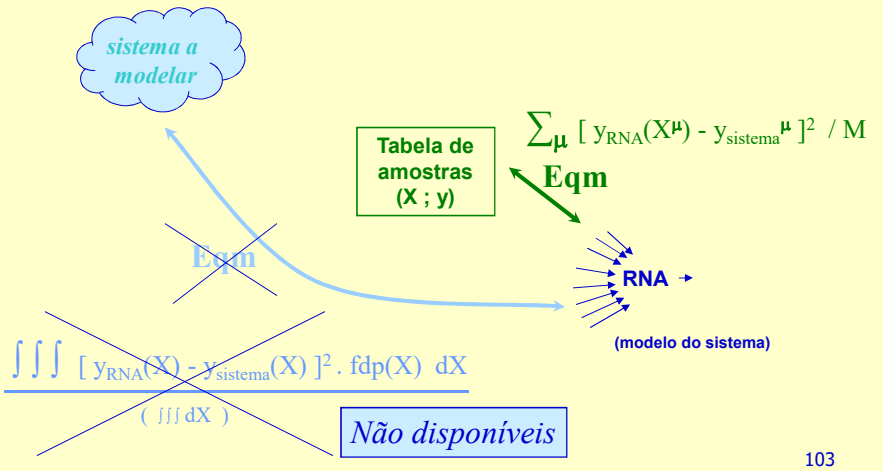
um novo exemplo



Os dados empíricos (x^{μ},y^{μ}) estão em verde;
Três modelos polinomiais gerados a partir dos dados, em azuis.

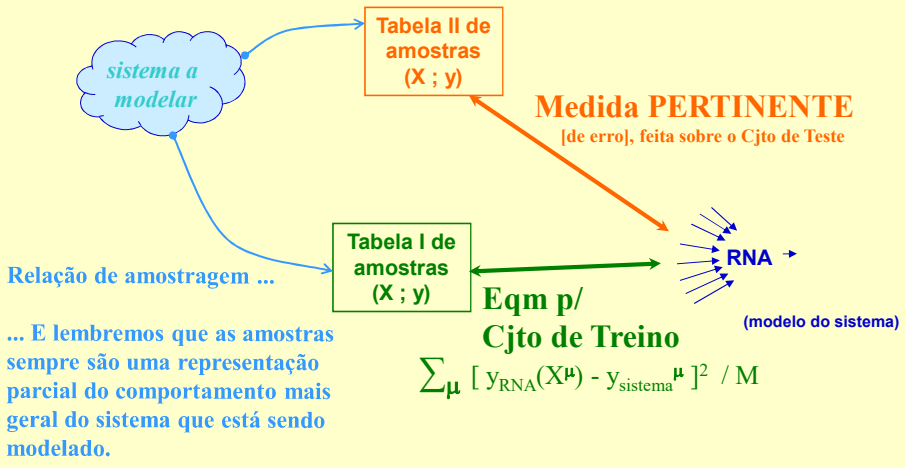
97

Sistema ... Amostras de treino ... RNA ...

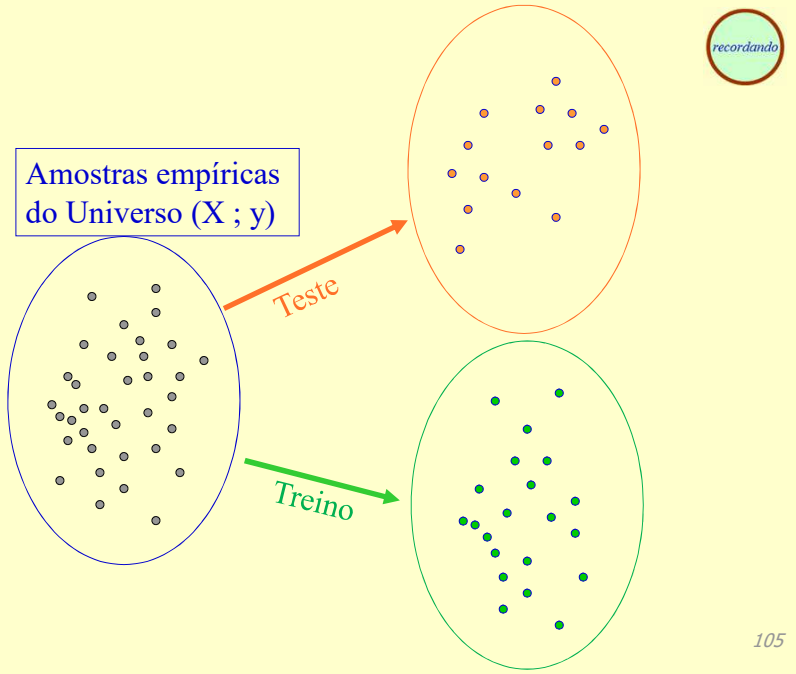


103

O treinamento mira minimizar o Eqm das amostras (X ; y) de treino. A avaliação de qualidade de generalização é feita a partir de Eqm com novas amostras (X ; y) ... Estas são chamadas amostras de teste :

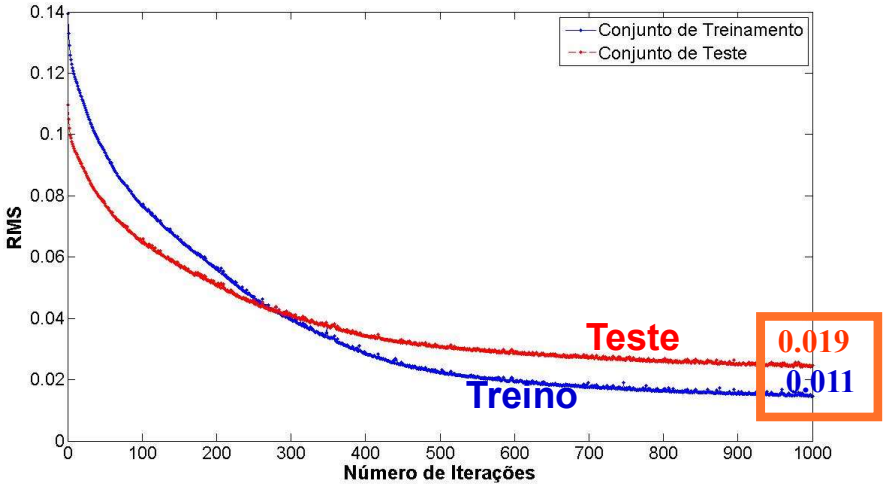


... E lembremos que as amostras sempre são uma representação parcial do comportamento mais geral do sistema que está sendo modelado.



Raiz do Erro Quadrático Médio (RMS) p/ conjuntos de treino e teste ... $RMS_{teste} > RMS_{treino}$

recordando



106

O Ciclo completo da modelagem:

0) Formalização do problema, mapeamento quantitativo em um modelo neural inicial e ... 0b) coleta de pares empíricos (X,y)

1) Fase de TREINO da RNA (MLP): com conhecimento dos X e dos y, que são ambos usados na calibração do modelo

2) Fase de TESTE / Caracterização da qualidade da RNA para generalizar: temos novos pares X e y, com y guardado "na gaveta", usado apenas para avaliação, não para re-calibração. É como um ensaio de uso final do modelo, com possibilidade de medir a sua qualidade com o y que foi guardado na gaveta.

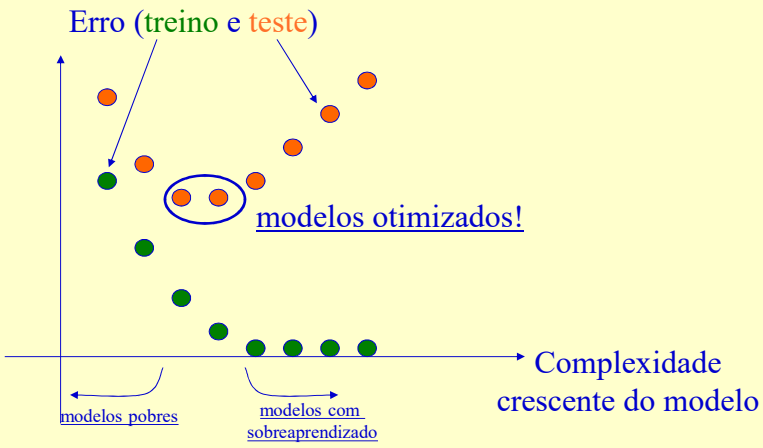
[Fase de refinamentos sucessivos da RNA e/ou dos dados e/ou do modelo, em ciclos diversos, recomeçando desde o passo 0 ou do passo 1]

3) Fase de USO FINAL da RNA, com y efetivamente não conhecido, e estimado com conhecimento dos X + uso do modelo calibrado.

.... Diferenças e semelhanças entre 1, 2 e 3

107

Sobreaprendizado em "sumário executivo"



Mas cuidado ... você experimentou rodar mais de uma vez o MBP sobre os mesmos dados de treino e ver se o resultado final é o mesmo?

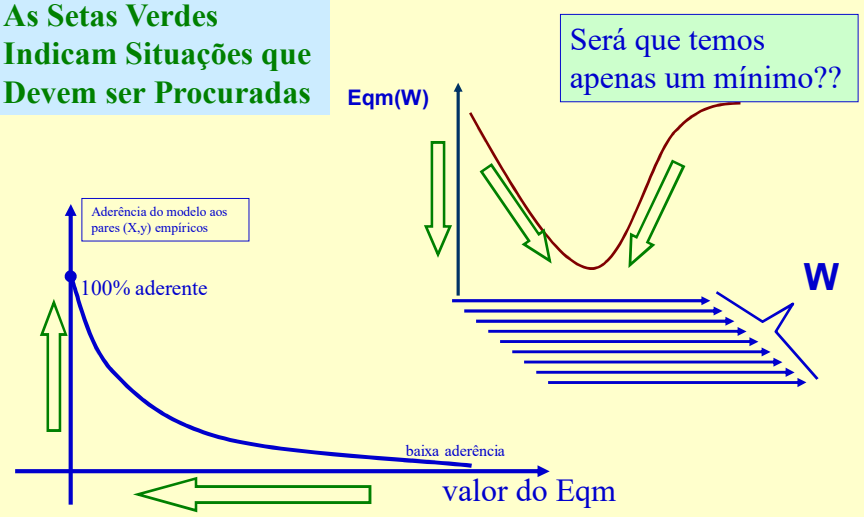
- **Não há garantia de que o W randômico, seguido de Gradiente Descendente leve ao mínimo global quando há vários mínimos, só há garantia de que levará a um dos mínimo locais, que pode não ser o mínimo global**
- **Isto faz necessário rodar várias vezes a otimização de pesos e selecionar a configuração com melhor Eqm**
- **É legítimo ficarmos com a rede otimizada de menos Eqm final? Ou deveríamos ter um ataque de avaliar o Eqm médio de várias tentativas?**

- ***Outro assunto ... e quanto à variação associada ao k-fold Cross Validation? Média ou melhor Eqm?***

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

Devemos buscar Maximização da aderência = Minimo Eqm possível

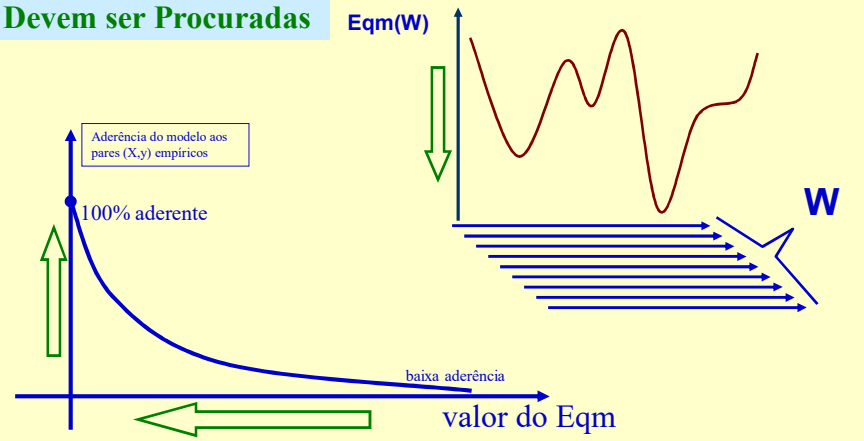
As Setas Verdes Indicam Situações que Devem ser Procuradas



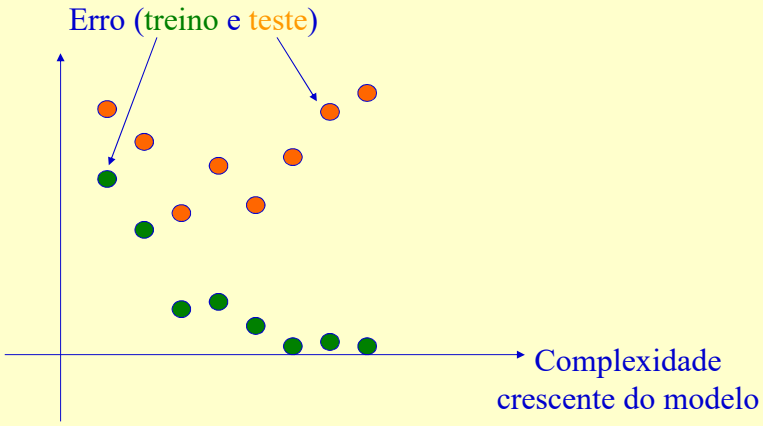
O que devemos mirar quando exploramos o espaço de pesos W buscando que a RNA seja um bom modelo?

Devemos buscar Maximização da aderência = Minimo Eqm possível

As Setas Verdes Indicam Situações que Devem ser Procuradas



Atenção para componentes randômicas que impactam muito quando se faz um único ensaio de medida de erro, para cada tamanho de rede específico (um ensaio apenas, para cada grau de complexidade) ...

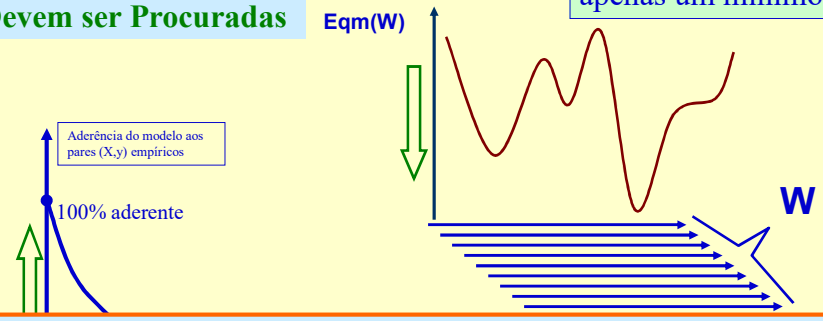


O que devemos mirar quando exploramos o espaço de pesos W buscando que a RNA seja um bom modelo?

Devemos buscar Maximização da aderência = Minimo Eqm possível

As Setas Verdes Indicam Situações que Devem ser Procuradas

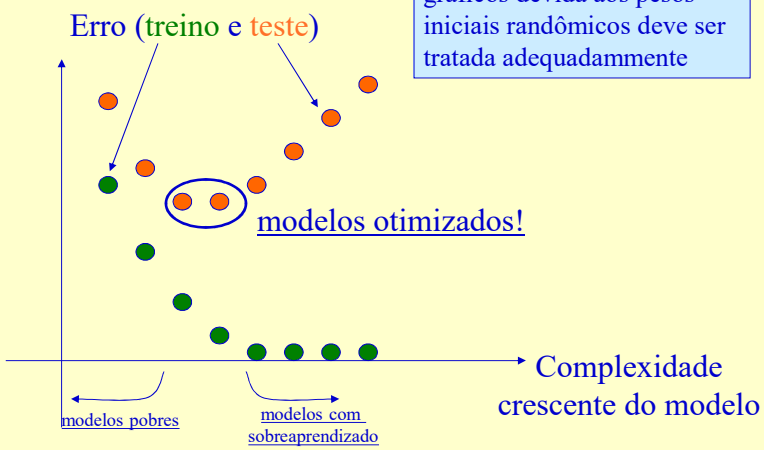
Será que temos apenas um mínimo??



... Para não sermos reféns de mínimos locais com alto Eqm, podemos aplicar o gradiente descendente repetidamente na mesma RNA, com novos pesos iniciais randômicos em cada rodada, mantendo para o modelo final apenas os valores de pesos associados ao ensaio com o melhor dos resultados finais no Eqm!

Com repetidos ensaios em cada grau de complexidade os mínimos locais são evitados e detectamos adequadamente o sobreaprendizado

Ou seja, a flutuação nos gráficos devida aos pesos iniciais randômicos deve ser tratada adequadammente

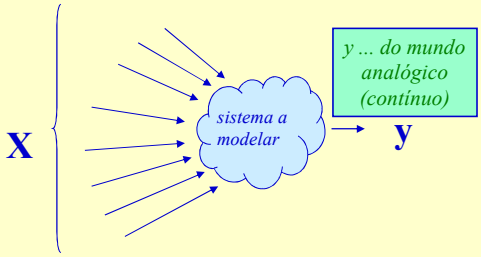


116

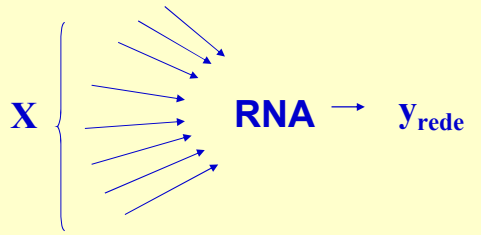
Identificando os ingredientes para o risco de sobreaprendizado nos contextos de regressão multivariada e de reconhecimento de padrões multivariado

121

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



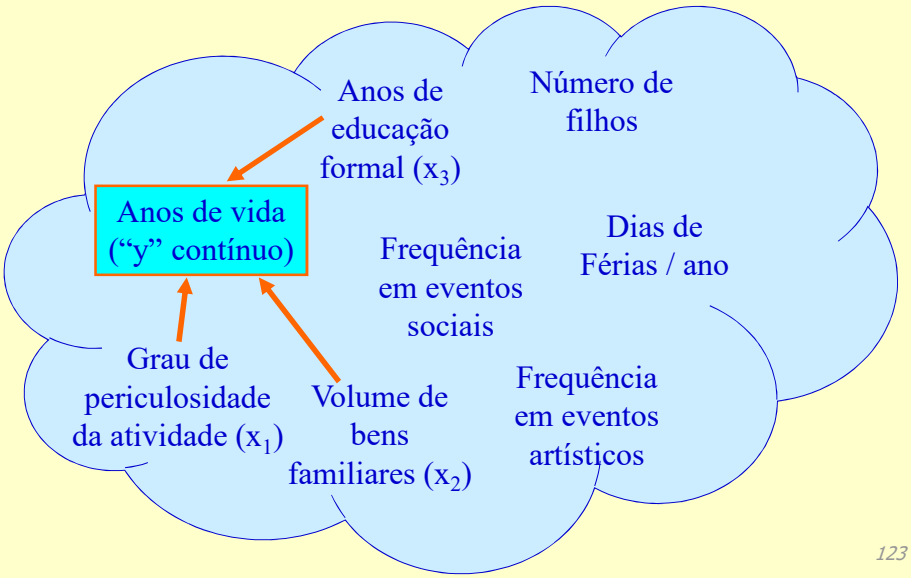
Assumimos que a variável y do sistema a modelar é uma função (normalmente desconhecida e possivelmente não linear) de diversas outras variáveis desse mesmo sistema



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

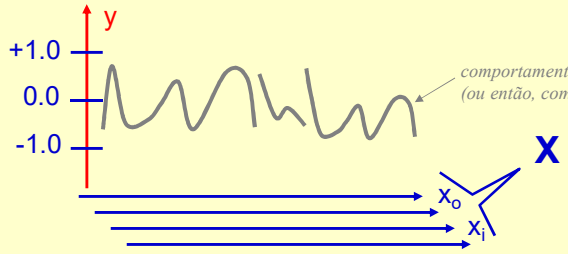
122

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



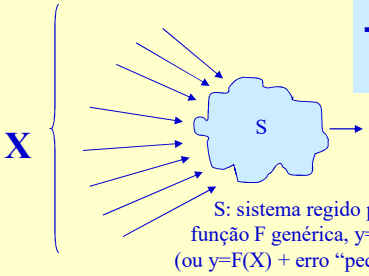
123

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



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

- ESTIMADOR ;
- MODELAGEM POR MAPEAMENTO X→y



(.... VALORES CONTÍNUOS!!!)

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

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

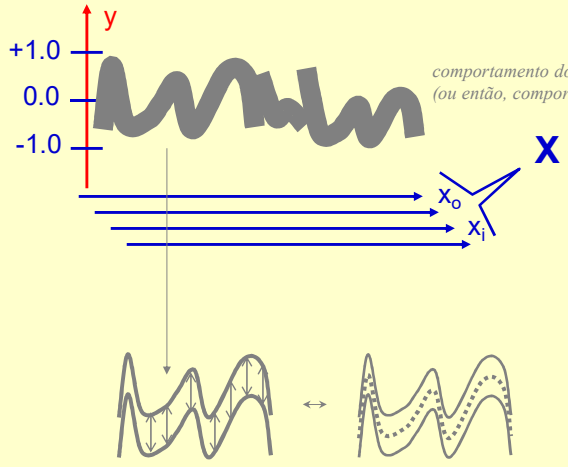
comportamento do sistema a modelar
(ou então, comportamento "padrão ouro")

.... 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

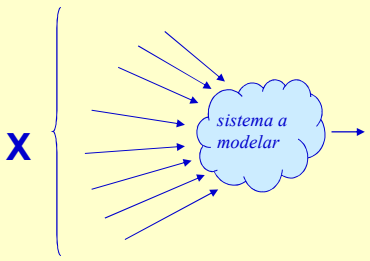
Cenário mais real: a "função" y(X) do sistema modelado é "difusa": $y = F_{médio}(X) + flutuação$ ....



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

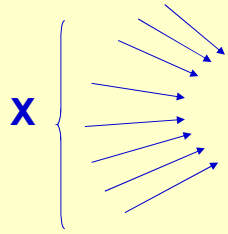
... neste cenário, podemos olhar para o y observado no sistema que se deseja modelar não mais como um valor específico bem definido, mas como um valor médio esperado (dado valor de X) e uma faixa de valores em torno desse valor médio esperado.

RNAs como reconhecedor / detetor de padrões ...



y ... responde à seguinte pergunta:
X é padrão notável? **Sim ( ) Não ( )**

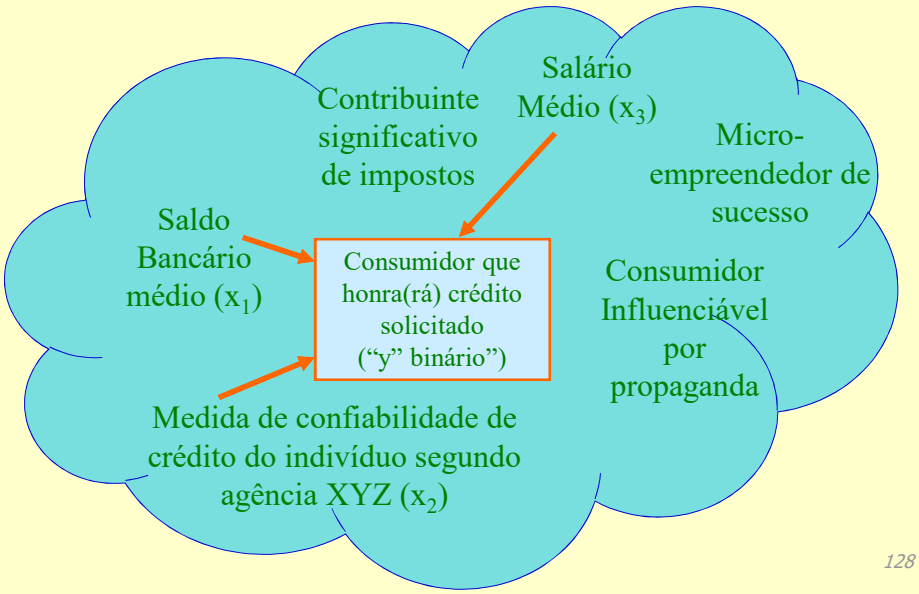
y ... é do universo discreto / binário (SIM versus NÃO), y não é analógico



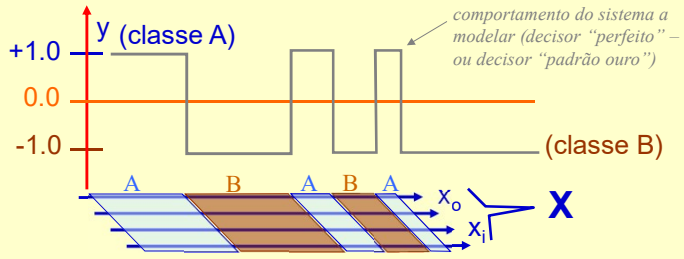
RNA → y_{rede} **... +1 ( )** *versus* **- 1 ( )**

... ou alternativamente: y_{rede} **... 0.5 ( )** *versus* **- 0.5 ( )**
... ou alternativamente: ... y_{rede} **> 0 ( )** *versus* y_{rede} **< 0 ( )**

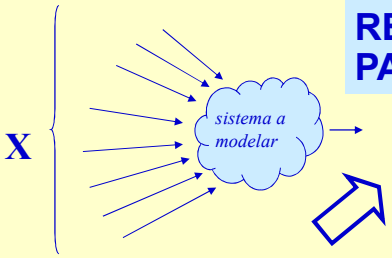
Um hipotético universo de variáveis inter-dependentes, passível de modelagem/ens



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



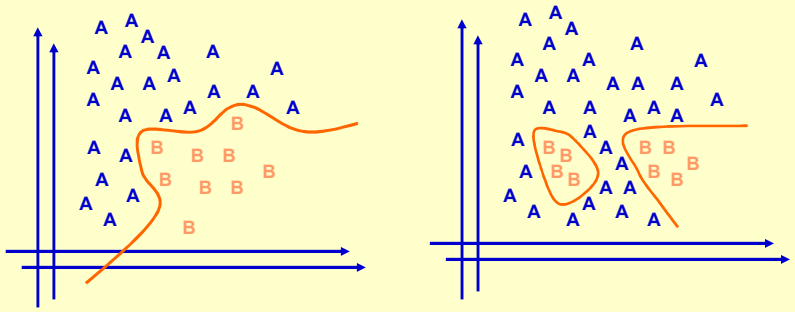
CLASSIFICADOR; RECONHECEDOR DE PADRÕES NOTÁVEIS



y (... restrita a valores +1 e –1)

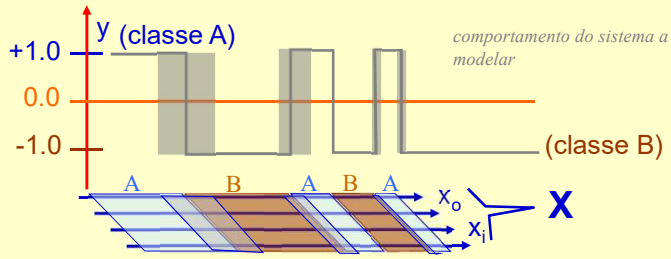
Capacidade de reconhecimento de padrões em casos complexos NÃO LINEARES

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



130

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

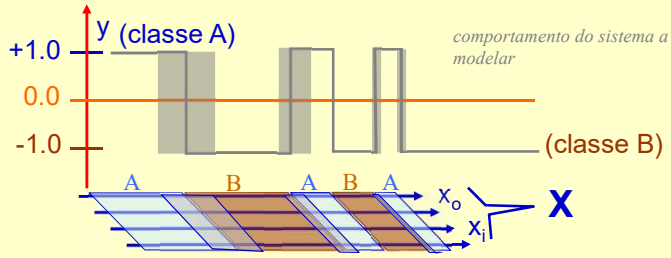


.... em problemas concretos / reais, há sempre alguma ambiguidade no mapeamento que leva valores de X aos valores discretos de y. Não temos uma função y=F(X) no sentido matemático exato, pois para uma dada ênupla de valores X fixado temos em alguns casos de fronteira a possibilidade de observar no y 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

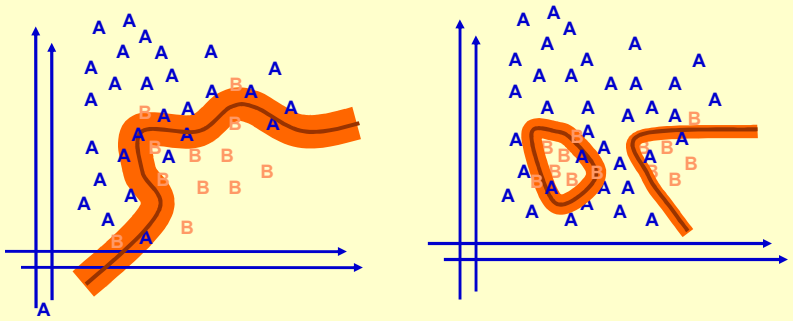
131

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

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



O Ciclo completo da modelagem:

0) Formalização do problema, mapeamento quantitativo em um modelo neural inicial e ... 0b) coleta de pares empíricos (X,y)

1) Fase de TREINO da RNA (MLP): com conhecimento dos X e dos y, que são ambos usados na calibração do modelo

2) Fase de TESTE / Caracterização da qualidade da RNA para generalizar: temos novos pares X e y, com y guardado "na gaveta", usado apenas para avaliação, não para re-calibração. É como um ensaio de uso final do modelo, com possibilidade de medir a sua qualidade com o y que foi guardado na gaveta.

[Fase de refinamentos sucessivos da RNA e/ou dos dados e/ou do modelo, em ciclos diversos, recomeçando desde o passo 0 ou do passo 1]

3) Fase de USO FINAL da RNA, com y efetivamente não conhecido, e estimado com conhecimento dos X + uso do modelo calibrado.

.... Diferenças e semelhanças entre 1, 2 e 3

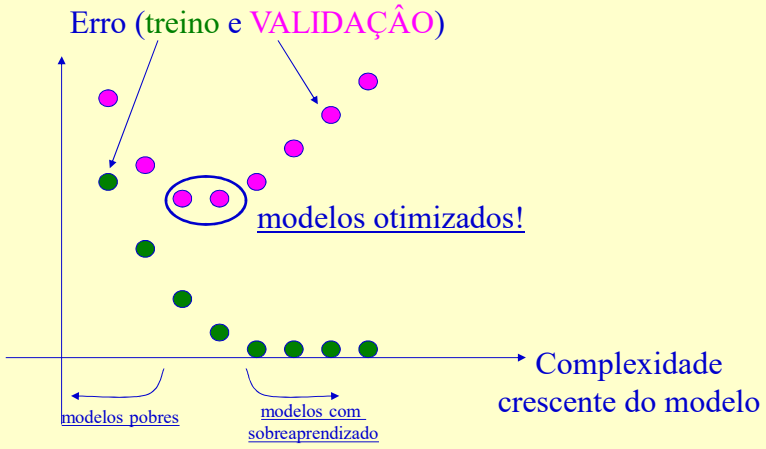
Revisitando os Conjuntos de Dados Empíricos ...

Treino + Teste ....

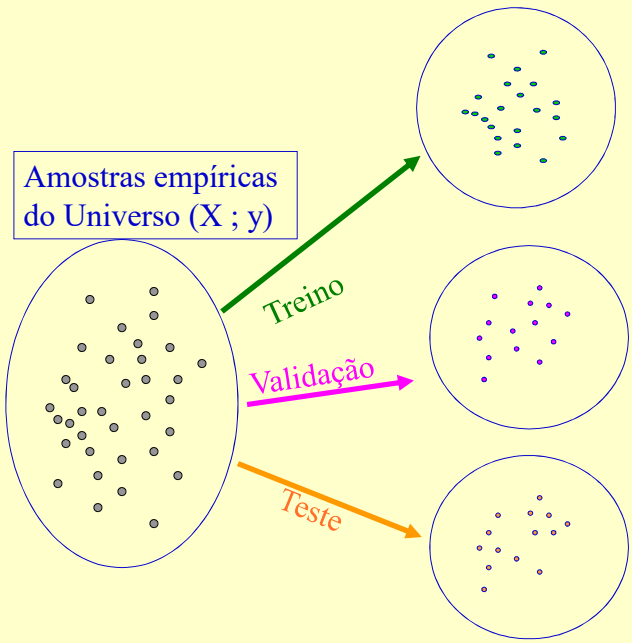
versus

Treino + Validação + Teste

Sobreaprendizado em "sumário executivo"

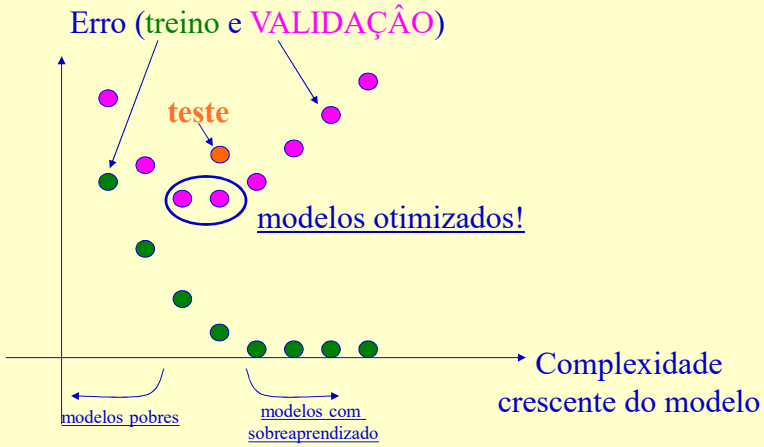


142



143

Sobreaprendizado em "sumário executivo"



144

Alguns dos diversos usos do conceito de conjunto de validação, adicional ao conjunto de treino e de teste:

- *Seleção de complexidade do modelo neural para limitação de sobreaprendizado*
- *Ativação do early stop no aprendizado (Matlab): critério de parada adicional no processo de refinamento de pesos sinápticos*
- *Balizador no processo de seleção de estratégias de pré-processamento / extração de medidas X alternativos*
- *... etc*

145

recordando

Elaborando uma Solução em dois estágios ....