

Problemas para a unificação lógica de redes de algoritmos não convencionais

Carlos Alberto Lungarzo

Como citar: LUNGARZO, C. A. Problemas para a unificação lógica de redes de algoritmos não convencionais. *In:* GONZALEZ, M. E. Q.; DEL-MASSO, M. C. S.; PIQUEIRA, J. R. C. (org.). **Encontro com as Ciências Cognitivas - volume 3**. Marília: Oficina Universitária; São Paulo: Cultura Acadêmica, 2001. p. 177-208. DOI: <https://doi.org/10.36311/2001.85-86738-19-0.p177-208>



All the contents of this work, except where otherwise noted, is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 (CC BY-NC-ND 4.0).

Todo o conteúdo deste trabalho, exceto quando houver ressalva, é publicado sob a licença Creative Commons Atribuição-NãoComercial-SemDerivações 4.0 (CC BY-NC-ND 4.0).

Todo el contenido de esta obra, excepto donde se indique lo contrario, está bajo licencia de la licencia Creative Commons Reconocimiento-No comercial-Sin derivados 4.0 (CC BY-NC-ND 4.0).

PROBLEMAS PARA A UNIFICAÇÃO LÓGICA DE REDES E ALGORITMOS NÃO CONVENCIONAIS

Carlos Alberto LUNGARZO¹

Introdução

As redes neurais artificiais (doravante indicadas pela sigla **RN**, mesmo em singular) são utilizadas em diversos campos da ciência cognitiva e, mais especificamente, em *inteligência artificial* (doravante indicada **IA**), em, pelo menos, dois sentidos:

Primeiro, como modelos formais, sendo que este uso percorre um espectro bastante amplo, que vai desde a modelagem quase puramente icônica dos começos da teoria, até a modelagem matemática como é apresentada nos textos mais modernos (CAUDILL; BUTLER, 1992; KOVÁCS, RUMELHART; MCCLELLAND, 1987)

Segundo, como programas baseados nesses modelos.

A formação do modelo lógico-matemático de rede segue o percurso usual da criação de modelos em qualquer ciência factual. Numa primeira aproximação, o modelo é uma estrutura simples, que tenta abstrair as características mais externas do objeto a ser modelizado, e preserva, tanto no caso das **RN** como em casos mais clássicos, uma certa analogia iconográfica com o objeto. Nessa forma foram produzidos os primeiros *modelos* de sinapse, que reproduziam geometricamente as relações entre os neurônios reais. Posteriormente, essas estruturas foram tornando-se mais complexas, ao ponto que sua descrição começou a exigir uma teoria matemática específica.

Este primeiro sentido é o mais relevante para a análise metodológica e epistemológica da teoria das **RN**.

Já o segundo sentido é mais familiar. Como acontece na maior parte das ciências, especialmente em suas áreas mais desenvolvidas, muitos

¹ Instituto de Filosofia e Ciências Humanas/Universidade do Estado do Rio de Janeiro - Área Multidisciplinar/Conselho Nacional de Pesquisa e Desenvolvimento - Lungarzo@hotmail.com
Quero agradecer o apoio recebido pelo Comitê Multidisciplinar do CNPq, sem o qual, minha pesquisa teria sido inviável. Este trabalho foi revisado pelos professores José Roberto Castillo Piqueira e Luiz Henrique Alves Monteiro, da Escola Politécnica da USP aos quais agradeço observações inteligentes e úteis que ajudaram a corrigir muitos defeitos do trabalho original. No entanto, a supervivência de alguns desses defeitos, ou aparição de outros, é apenas de minha própria responsabilidade.

processos computacionais são preparados e executados sem necessidade de uma análise de sua fundamentação. Esta confiança em fundamentos que não são perfeitamente conhecidos, permite o desenvolvimento operacional da disciplina, mesmo que os conceitos básicos permaneçam indefinidos. Existem exemplos conspícuos dessa evolução de teorias totalmente transparentes apoiadas em bases opacas. O mais famoso é o cálculo diferencial e integral dos séculos XVIII e XIX, que foi ricamente desenvolvido mesmo sem possuir uma definição de número real, muito menos de número natural. Outro exemplo é o eletromagnetismo clássico que avançou antes que as propriedades da radiação fossem adequadamente entendidas, o que só aconteceu a partir de 1900.

Eventualmente, a possibilidade de construir teorias complexas, conceitualmente fundamentadas em teorias mais básicas, pode ser um sintoma de que as relações lógicas entre conceitos não são sempre essenciais para estabelecer uma ordem cognitiva, mas os detalhes desta conjectura não estão, por enquanto, devidamente investigados.

Portanto, mesmo aceitando a diversidade de significados do conceito de **RN**, e da falta de clareza sobre sua natureza matemática, tem sido possível desenvolver programas que, convencionalmente, consideramos simulações de redes neurais *reais*. Estas redes seriam, por sua vez, objetos que se comportariam, em seus aspectos principais, como as populações de neurônios biológicos. A eficiência desses programas, rodados em computadores convencionais sob o *hardware* seqüencial, algorítmico e digital (todo o qual constitui o que a teoria de **RN** intuía substituir) é notória em problemas usualmente considerados como pertencentes a **IA**, por exemplo, reconhecimento ótico de caracteres, controle de robôs, calibragem *instantânea* de objetos de precisão, etc.

Um terceiro significado, menos relevante para nossos objetivos, é o de **RN** como uma forma específica de *hardware*. Foram feitos alguns avanços na construção de robôs direcionados apenas por sensores que transmitem informação a seu sistema central sem programação previa, emulando em alguma medida o comportamento *espontâneo* de organismos vivos, e aproveitando os métodos da computação analógica. De fato, a construção real de uma **RN** suporia uma interconexão complexa de numerosos aparelhos que se comportariam como microscópicos computadores analógicos. Além das dificuldades técnicas para substituir um *software* tradicional por todos esses computadores analógicos, subsiste a dificuldade teórica de reproduzir processos que não possuem um análogo físico.

É possível imaginar um robô que realize diversas ações no plano físico, por analogia com o processo real, como acontece no caso de

evitar obstáculos ou manipular objetos; no entanto, não parece ter sentido um tratamento analógico do processo de pensamento ou, mais modestamente, da análise da linguagem natural.

Problemas considerados *hard* para os algoritmos usuais, como a análise da linguagem natural e, portanto, mais aproximados da inteligência biológica que dos procedimentos eletrônicos, sugerem que podem esperar-se soluções de métodos que simulem de maneira mais próxima os processos dessa inteligência natural.

Os métodos tradicionais em inteligência artificial já são mais sofisticados do que os algoritmos *não inteligentes*, pelo fato de possuir *bancos* de conhecimento, que permitem fazer inferências.

Mas a teoria que consiste em criar algoritmos que imitem algumas das propriedades biológicas básicas é bastante recente: é a teoria dos *algoritmos genéticos (AG)*. Estes algoritmos utilizam conceitos tirados da teoria biológica de evolução e pretendem fazer justiça à idéia de *memória histórica* que os processos evolutivos possuem. Junto com as **RN**, os **AG** são a maior emulação dos processos biológicos criados com fins computacionais.

Este parentesco, pelo menos na origem, sugere que ambos conceitos possam ter uma estrutura básica comum. Nos casos simples, isso de fato acontece. Mas o caso geral será examinado com maior detalhe depois.

Como sabemos informalmente, os algoritmos e as **RN**, são definidos como métodos que executam processos *deterministas*. Também se sabe que uma modificação na definição de algoritmo pode transformá-lo em *probabilista*. No caso das **RN**, as propostas de introduzir mecanismos aleatórios são mais diversificadas, porém menos precisas.

Uma possibilidade para introduzir probabilidades em **RN** seria o uso de variáveis aleatórias na definição dos *pesos*, que chamaremos **W** neste trabalho. [Vide a seção seguinte]. Mas, durante os últimos anos tem sido bastante investigada uma estrutura que pretende representar o conhecimento randômico e/ou nossa *crença* na ocorrência de certos fatos: as redes *bayesianas (RB)*. Apesar de conexão dos componentes de uma **RB** em forma de rede, esta estrutura e o sistema real que representa não é uma sub-variedade direta das **RN**.

Com efeito, no primeiro sentido, como estrutura formal e projeto de um sistema real, a **RN** é modelo de um sistema que computaria informação de maneira *conexionista*. Um fato que nem sempre é enfatizado na literatura de metodologia e fundamentação das **RN**, é que esse conexionismo é *forte*,

no sentido de exigir que a informação seja transmitida de maneira simultânea pelas diversas conexões. Este sentido é o clássico nas teorias apoiadas em modelos neurais, pois os primeiros projetos de redes tentaram capturar a maior capacidade possível do conexionismo biológico. Já, o conceito de conexionismo que se tornou popular entre os computólogos depois da tese de Hillis [vide especialmente p. 18 et seq.], é mais fraco e conceitualmente mais convencional.

Já um algoritmo em geral pode ser pensado como um objeto formal independente de qualquer *hardware*, ou como a implementação real desse objeto. Um caso particular de algoritmo é o **AG**.

Tanto as **RN** como os algoritmos, em particular os **AG**, são estruturas que, quando implantadas, *resolvem* problemas formalmente bem definidos (embora deixem sem resolver *mais* problemas dos que podem resolver).

Já as **RB** são também estruturas, mas sua finalidade é fornecer informação, analogamente a qualquer outra teoria científica cujos componentes podem ser interpretados como enunciados. Por outro lado, seu caráter probabilístico, e, em especial, o uso sistemático de probabilidades condicionais, faz plausível pensar-las como métodos para estimar, entre outras informações, o grau de aprendizagem das próprias **RN**.

Resolver problemas de **IA** pela aplicação de probabilidades é relativamente freqüente. Esse método foi proposto por mim (vide LUNGARZO, 1993) para decidir entre informações contraditórias fornecidas por um sistema especialista. Isto é mais simples que o uso das **RB**, e consiste apenas no cálculo da probabilidade não condicional de cada uma das sentenças contraditórias α e $\neg \alpha$, que transportam essa informação.

Este uso *ingênuo* das probabilidades em sistemas especialistas pode ser deslocado para o caso mais complexo de **RN**.

Com efeito, seja um problema π , para o qual existe pelo menos uma solução σ , tal que, se C é a condição de π , então (**d**) satisfaz **C**, para qualquer elemento $d \in D$, sendo **D** o domínio do problema π . (Esta formulação segue a teoria de Veloso e Lopes que não podemos desenvolver aqui; veja, por exemplo, Lungarzo, 2002?).

Embora existam conexões parciais entre todos estes conceitos e algumas delas muito simples, o problema geral é mais complicado:

Será que existe uma família de estruturas suficientemente "genéricas", (**A** _{\wedge}), tal que **RN**, **AG** e **RB**, pensados como estruturas, são casos particulares das **A**?

É fácil reconhecer alguns vínculos entre estes diversos métodos, como veremos em seguida. No entanto, não possuímos um tratamento formal unificado destas três classes de sistemas: **RN**, **RB** e **AG**.

Em geral, há uma tendência a pensar que as **RN** são *mais* do que algorítmicas, como fazem notar Caudill e Butler, ao afirmar que as redes são não digitais, não algorítmicas e fortemente paralelas.

De fato, estas propriedades estão relacionadas. Em particular, a não digitalidade implica, por si mesma, o caráter não algorítmico.

Com efeito, um algoritmo pode ser considerado um sistema formal, mesmo que consista de atividades tão ostensivamente empíricas como a execução de uma receita de bolo. Se diferentes formulações podem ser reconhecidas como "expressões diferentes" de uma mesma receita, isso acontece porque existe um sistema formal subjacente do qual todas essas formulações são casos particulares, que se apresentam com graus diversos de "perturbação" a respeito do original, no mesmo sentido em que reconhecemos todas as letras 'a' deste texto como instâncias da mesma letra "ideal", independentemente da fonte e corpo em que esteja escrita (cf. HAUGELAND, p. 50 et seq.).

Que não ocorra a recíproca (ou seja, que um sistema formal não seja necessariamente um algoritmo) não afeta nosso raciocínio: a questão sobre a solubilidade de problemas formais vai além de nosso objetivo nesta nota.

Portanto, se \mathfrak{R} é um sistema formal, certamente é digital, o que mostra que nenhum objeto não digital pode ser um algoritmo.

O paralelismo forte exigido das **RN** é algo mais complicado. Aqui, a inspiração biológica é adotada da maneira mais difusa e menos explícita. Assumir que um processo informacional seja paralelo, num sentido que vai além do paralelismo convencional consistente na execução simultânea de diversos *fluxos* seqüenciais de informação, como na computação ordinária baseada no modelo de Turing, parece motivado em nossa experiência biológica. Em alguns dos casos dessa experiência, por exemplo, no caso da percepção, recebemos informação *fortemente paralela* de diversas fontes, e a integramos por meio de nossa mente sem que seja possível reconhecer a existência de *bits*, nem, muito menos, a organização dos mesmos.

Isto parece induzir a idéia de que, caso existisse uma definição rigorosa de *paralelismo* (pelo menos, tão rigorosa como é possível tornar a definição de algoritmo e de digitalidade), a não digitalidade dos sistemas de **RN** implicaria também o caráter paralelo, ou seja, a não necessidade de seqüencialidade. Com efeito, se um sistema não é digital, então não pode

ser transmitido como seqüência de dígitos binários, uma vez que a alternativa sim-não (verdadeiro-falso, **0-1**, etc.) permite formalizar digitalmente uma informação. Assim sendo, a informação desses sistemas poderia eventualmente ser transmitida em forma seqüencial (embora não como seqüência de bits), mas talvez não estaria obrigada a isso, chegando até os receptores da mesma maneira *paralela* em que chega aos nossos sentidos a informação contida, por exemplo, num filme que estamos assistindo, incluindo os numerosos estímulos visuais e os diversos sinais auditivos.

Isso não impede a reconstrução de fragmentos dessa informação pela via digital. De fato, nossa percepção tem componentes digitais que são os que fazem possível a captura e estocagem de símbolos. O que usualmente pretendem dizer os especialistas em **RN** é que a transmissão neural é *mais do que* simbólica.

Apesar destas considerações, o que os teóricos das **RN** oferecem é apenas uma *intuição* de paralelismo forte, e não uma definição rigorosa. O problema se encontra em estado ainda especulativo, talvez porque sua elucidação não tenha sido necessária na aplicação das redes. A idéia parece suscetível de ser formalizada considerando um paralelismo que não possa ser fatorizado em feixes paralelos de seqüências, o que sim acontece no uso habitual do conceito de paralelismo em computação.

Mas, para poder comparar **RN** com **AG**, por exemplo, deveremos considerar os algoritmos subjacentes às redes.

Ora, podemos pensar uma **RN** como um objeto real, cujo modelo formal é a **RN** entendida como estrutura. Este procedimento é similar ao de considerar um computador convencional como o objeto real (físico, eletrônico) cujo modelo é a máquina de Turing. Aliás, na época em que Turing propôs aquela estrutura, o computador físico ainda devia ser realizado como aconteceu pouco depois, pois tampouco tinha existência concreta, pelo menos, não na forma em que atualmente se entende o conceito de computador.

A analogia entre ambas situações, possui, no entanto, uma assimetria, que faz nossa situação atual mais confortável. Tendo programas de **RN** (ou seja, redes neurais no segundo sentido enunciado ao começo desta nota) que rodam em computadores convencionais, podemos simular as eventuais redes físicas que se construam no futuro, enquanto que, antes dos primeiros modelos reais da arquitetura Turing/von Neumann, a única maneira de simular uma máquina de Turing era através de um *experimento mental*.

Simulações de **RN** estão claramente explicadas, com software demonstrativo, no volume 2 de Caudill e Butler (1992).

Ao rodar um programa de **RN**, podemos comparar suas propriedades *computacionais* com as de programas considerados convencionais. A diferença entre ambos não é, como às vezes se pretende, *ontológica*, como seria o caso da diferença entre a máquina (qualquer que seja) e a mente humana. Os processos *representados* pelo programa são, sim, diferentes, no caso convencional e no caso *neural*, mas a diferença entre os processos realmente executados é apenas sua versatilidade. Neste sentido, a melhor ilustração desta diferença é a que existe entre programas ditos *convencionais* e os programas de **AI**. Os programas de **RN** teriam uma flexibilidade mais próxima a dos programas de AI que aos programas convencionais.

No entanto, a estrutura de **RN**, pensada como modelo lógico-matemático, pode ser vista teoricamente como uma simulação das conexões neurais reais e não como um programa. Nada impede isto: afinal, o próprio cérebro humano e outros organismos vitais podem ser simulados em computador, pelo menos, na medida em que seu funcionamento é conhecido.

Em síntese: há uma estrutura formal correspondente a **RN**, e, como sabemos, há uma estrutura formal correspondente aos algoritmos convencionais, apesar de que a clareza desta noção dispensa, na prática, a análise da mesma.

Isto continua nossa comparação com o cálculo: com efeito, podemos demonstrar teoremas sobre equações diferenciais, mesmo que não saibamos se existe um critério recursivo para determinar sua condição de teoremas, antes de conhecer a definição de número real! Sabia-se, por exemplo, que equações com coeficientes analíticos possuíam soluções analíticas, antes de conhecer o caráter localmente compacto do conjunto dos números reais.

Um assunto motivado pelo estudo dos fundamentos e métodos de redes e de algoritmos, e a comparação entre as estruturas implícitas subjacentes a ambos, é a interrogação pela possibilidade de possuir uma família geral de estruturas, onde a estrutura de algoritmo fosse um caso particular da estrutura de rede.

Esta interrogação é provocada, em princípio, por um certo *preconceito lógico*, herdado das ciências exatas: o de supor que métodos que geram processos reais afins, devem possuir um cenário teórico comum. De fato, isto acontece de maneira progressiva no caso da física, apesar de deixar fora desta sistematização algumas teorias. A primeira unificação é a que foi feita entre a mecânica celeste e terrestre por Newton, e, posteriormente, entre teorias macroscópicas e cosmológicas, e entre teorias sobre fenômenos observáveis e teorias quânticas.

Sendo que o problema de *computar* resultados, seja via redes, seja via algoritmos convencionais, é mais próximo da matemática que da física, a expectativa de ter uma unificação bem sucedida é maior.

[Um problema que pode vir a complicar este projeto, é o reconhecimento de que certas definições aparentemente formais, no campo da computação, podem precisar de referências ao hardware. Para simplificar nosso enfoque, vamos ignorar aqui essas considerações. No me posso deter neste assunto que foi tratado em outra parte (Cf. LUNGARZO, 1990)].

Mas, será que existe uma estrutura suficientemente geral, para incorporar, da mesma maneira, as redes neurais, os algoritmos genéticos e as redes bayesianas?

Ou seja, existe uma estrutura **A**, tal que, particularizando alguns de seus componentes, possamos obter uma estrutura correspondente ao gênero dos **AG**, das **RN** ou das **RB**?

Caso exista, uma propriedade exigível a **A** é que seja não trivial. Com efeito, a estrutura formada, por exemplo, pelo universo de von Neumann **V**, com a relação de pertinência \in , certamente atende estas exigências, mas não possui interesse lógico-computacional: é grande demais!

Uma maneira de evitar a trivialidade seria garantir uma condição de minimalidade. Entendo que a seguinte seria uma condição desse estilo: **A** resolve todo problema que seja resolúvel por uma rede neural, mas para qualquer subestrutura própria \mathfrak{R} de **A**, deverá existir um problema, resolúvel por **RN** que \mathfrak{R} não resolve.

A colocação pode ser paradoxal, uma vez que não temos uma formulação precisa para *resolúvel por rede neural*, como temos para algoritmo (vide a seção seguinte). Esta proposta pode parecer afetada do mesmo caráter *tautológico* que teve a tese de Church. No entanto, acredito que é possível nos aproximarmos em algum sentido desse conceito.

Ainda, se aceitarmos um conceito intuitivo de não trivialidade, o passo seguinte deve ser a investigação de condições para **A**. Como os métodos **RN**, **AG** e **RB** são basicamente os centrais nas atuais teorias computacionais da cognição (deixando fora os algoritmos quânticos, dos quais não nos ocupamos neste artigo), os problemas seriam:

Determinar condições necessárias e suficientes para a existência de uma estrutura **A** tal que é modelo (no sentido da lógica de ordem superior) das teorias de **RN**, **AG** e **RB**.

Ou seja: $\mathbf{A} \models (\mathbf{RN} \cup \mathbf{AG} \cup \mathbf{RB})$

Garantir que \mathbf{A} seja não trivial. Se $\mathbf{B} \triangleleft \mathbf{A}$ [subestrutura própria], então \mathbf{A} não é modelo dessas teorias.

Encontrar a resposta não parece, neste momento, muito fácil, mas o problema, possui, segundo acredito, um interesse lógico fundamentacional análogo ao que teve, no passado, a teoria dos graus recursivos de solubilidade.

Entretanto, é possível pelo menos fazer uma análise dos componentes dos objetos destas teorias (**RN**, **RG** e **RB**) para dar um passo na direção do entendimento da solução.

Algoritmos deterministas e probabilísticos

Suponhamos \mathfrak{J} um algoritmo no sentido usual em teoria da computação. O conceito de algoritmo é também daqueles que são aceitos com base na clareza do critério para diferenciá-los intuitivamente, e para os quais não se pretende uma definição até que apareçam propriedades que exigem distinções muito finas. [Esse foi o caso da conjectura sobre a relação entre algoritmos deterministas e não deterministas.]

Inclusive nas fontes sobre fundamentação e análise de algoritmos se reconhece o caráter implícito do conceito exceto nos casos críticos, como o do problema **NP ? P**. (Vide, por exemplo, os comentários de MANBER, 1984, no capítulo 11).

No entanto, se desejarmos comparar sua estrutura lógica com a das redes e outros métodos computacionais, uma tentativa mais consistente de definição se faz necessária (HUTH; RYAN, p.176 et seq.).

Considere, por exemplo, um *método* para dividir um segmento de reta em 2^n sub-segmentos do mesmo comprimento.

Este método deve gerar um processo que é a divisão do segmento.

Se imaginarmos o método como um sistema de regras ou instruções, cada aplicação do método (ou seja, de uma das regras do método), seja *nova*, ou seja uma reiteração de uma aplicação anterior, gerará uma nova *etapa* em nosso processo. Ou seja, a resolução do problema com base no método, possui vários casos.

O esquema do método não deve depender de n , nem do comprimento do segmento dado. Serve para dividir qualquer segmento não

nulo em qualquer número de segmentos congruentes que seja potência de 2. No entanto, cada caso específico do método dependerá de n , pois seu número de passos será função deste número.

Observe um *quase programa*, ou seja uma descrição em linguagem natural da aplicação do algoritmo.

Passo	Regra aplicada	Entrada	Saída
1	Dividir o segmento pelo ponto meio, unindo os pontos de interseção de duas circunferências com raio maior ou igual que o segmento e com centro em cada um dos extremos	O segmento inicial AB, de comprimento c	Dois subsegmentos AD e DB de comprimentos $c/2$.
2	Escolher o subsegmento AD, e dividir pelo ponto meio com a mesma técnica de 1	Segmento AD de comprimento $(c/2)$	Segmentos AD' e D'D ambos de comprimento $\frac{1}{2}(c/2)$
3	Escolher o subsegmento DB do original AB, e dividir pelo ponto meio com a mesma técnica de 1	Segmento DB de comprimento $c/2$	Segmentos DB' e B'D ambos de comprimento $\frac{1}{2}(c/2)$
4	continua, dependendo do valor n		

Admitindo que este exemplo faz jus a nossa noção intuitiva de algoritmo, podemos destacar algumas propriedades que serão incorporadas à definição formal.

Primeiro, o algoritmo é um processo seqüencial. Embora sejam possíveis combinações de várias destas seqüências, cada uma delas pode ser isolada como a representação específica de um algoritmo. Esta *fatorização* não acontece nos processos típicos da inteligência natural, onde não há maneira de descompor a totalidade do processo em seqüências. (Isto não impede que partes desse processo possam ser representadas como seqüências, como de fato acontece quando se formaliza um raciocínio ou um cálculo).

Segundo, o algoritmo aplica um conjunto de *instruções*, ou seja, sentenças em forma de regras. Essas regras são digitais, no sentido de que variações em suas formulação podem ser inessenciais, sendo que diversas instâncias das mesmas devem ser reconhecidas como idênticas.

Terceiro, essas regras permitem *alimentar* certas operações fornecendo os dados para suas entradas e colocando novamente em seqüência suas saídas (se houver). Essas operações devem ser *calculáveis*, ou seja, estar representadas por operações *recursivas*.²

No exemplo acima, há basicamente duas regras.

² Vou ignorar a diferença entre *calculável* e *recursivo*, estabelecida na assim chamada 'tese de Church'.

Uma que ordena aplicar a operação de dividir em dois (que é uma operação calculável no sentido tradicional).

A outra é uma regra complexa que permite escolher o sub-segmento no qual será aplicada a operação de divisão. A regra pode ser definida construtivamente por indução no ordinal dos sub-segmentos obtidos numa ordem fixa (por exemplo, a lexicográfica, de esquerda à direita). Estando definida para todo n , com maior razão o estará para todo $n \leq 2^m$, para a m - sima aplicação.

Estas propriedades podem ser capturadas de maneira geral.

Formalmente, um algoritmo é uma seqüência de regras $\langle R_1, \dots, R_n \rangle$ tal que: para todo $i \leq n$, R_i é uma instrução que permite introduzir dados numa operação recursiva $O_{m(n)}$, para toda seqüência de dados $\Delta = \langle D_1, \dots, D_m \rangle$.

Uma aplicação de ρ a Δ , é uma seqüência de "passos": $\langle P_1, \dots, P_q \rangle$ tal que, para cada $1 \leq j \leq p$, ocorre que: $P_j \equiv D_h$, para algum $h \leq m$, ou existe uma subsequência $P_{k_1}, \dots, P_{k_{s(j)}}$ onde os $k_h \leq j$, e uma regra $R_{t \in \rho}$ tais que: $R_t(P_{k_1}, \dots, P_{k_{s(j)}}) = P_j$

Em geral, esta definição de algoritmo supõe o determinismo, uma vez que não consideramos que as regras recursivas possam ser aleatórias. Vamos nos aproximarmos do caso randômico por um caminho algo diferente do convencional (MANBER, 1989, p. 159), já que, apesar da funcionalidade das definições clássicas, elas não permitem uma caracterização lógica precisa.

Seja agora uma estrutura aleatória, ou seja, a formalização (aproximada) da idéia física de um mecanismo regido pelo acaso: $\langle U, \Omega, \mu \rangle$ onde U é um conjunto (o universo), Ω uma σ -álgebra sobre U e μ uma medida de probabilidade sobre os elementos de Ω . [Ou seja, como sempre, μ é σ -aditiva, regular, monotônica, e satisfaz $\mu(U) = 1$]

Agora podemos definir com maior precisão o conceito de algoritmo probabilístico.

Na definição de algoritmo, acima, aceitamos que alguns dos *passos*, possa ser obtido como saída de um sistema aleatório S . Mas formalmente: \mathfrak{S} é um algoritmo **probabilístico** se e somente se existe um sistema aleatório S , tal que, ao menos um passo P_k , pode ser obtido como saída de S para entradas fornecidas por passos anteriores (mesmos que estes tivessem sido deterministas).

Ou seja, a condição de ser aleatório é forte no sentido de eliminar determinismo. Basta que um algoritmo tenha um único uso randômico de uma regra durante uma aplicação, para ser considerado aleatório.

[No caso em que $\mu \equiv 1$, temos o caso determinista. Não entanto, para não banalizar a relação entre os dois casos, usaremos 'probabilista' para nos referirmos ao caso em que $\mu \neq 1$].

Observemos este exemplo: Ordenar os seguintes números de menor a maior **123, 45, 7, 8, 19, 1, 33, 27, 0**. Um algoritmo determinista constrói a seqüência: $0 < 1 < 7 < 8 < 19 < 27 < 33 < 45 < 123$.

Suponha agora que o problema exige: Escolha, entre os números assim ordenados, um que seja > 30 .

Um algoritmo igualmente simples encontra os três números maiores que **30**: são **33, 45, 123**.

O problema é como escolher um deles. A escolha é, no caso do operador humano, um ato voluntário: escolho, por exemplo, **45**. Se não for, deve ser executada por um mecanismo que seja fisicamente randômico. Por exemplo, podemos colocar três plaquinhas com os números **33, 45 e 123**, e, usando uma engenhoca mecânica adequada (por exemplo, um expulsor de objetos que arremessa um deles depois de fazer girar um disco onde estão colocados), selecionar um.

Mas isso não tem equivalente nos hardwares usuais. Os comandos dos algoritmos, traduzidos em mensagens de baixo nível produziram a abertura e fechamento de determinadas portas lógicas gerando um único resultado.

Isto não impede, obviamente, formular um algoritmo probabilístico. De fato, o uso de algoritmos probabilísticos é rotina na teoria da computação e, como outras entidades do mundo real, podem ser adequadamente simulados por estruturas formais que capturam as características que deveria possuir o objeto real.

Em nosso caso, podemos considerar que a estrutura de probabilidade $\langle U, \Omega, \mu \rangle$ é o modelo teórico do componente aleatório de nosso algoritmo. Assim como o algoritmo formal é um projeto de um método real para a solução de um problema, a estrutura de probabilidade é o projeto de um mecanismo real que forneceria saídas aleatórias, mesmo para entradas deterministas.

Sua implementação prática pode ser feita, como de hábito, por programas que produzem eventos (por exemplo, ocorrência de números) de maneira que não é conhecida para o operador, o que os transforma cognitivamente em aleatórios.

Redes neurais e algoritmos convencionais

O conceito de **RN** convencional é bastante familiar ao público de ciências cognitivas ou **IA**.

Para fixar idéias, considere uma **RN**, como uma estrutura **A** da forma: $\mathbf{A} = \langle \{u_\lambda\}; a(t); \{f_\eta(a_i(t))\}; W_j; \mathfrak{R}; \mathbf{E} \rangle$ (*) onde os **u's** são os processadores interconectados considerados como nós (neurônios) da rede; $a(\mathbf{t})$ é a função vetorial de ativação; as f 's são funções que transformam o valor de cada coordenada da função de ativação num sinal de saída e cada W_h é uma matriz que caracteriza às conexões.

\mathfrak{R} é o conjunto das regras que governam o funcionamento da rede, e **E** é o ambiente. [Uma visão sistemática e muito transparente da evolução do conceito de rede pode ser vista em Kovács, 1996, cap. 1-6].

Considerada como estrutura, a rede é de ordem superior a 1a., mas pode ser definida numa teoria de conjuntos que contenha o cálculo diferencial real.

Funcionalmente, é importante perceber que tanto $a(\mathbf{t})$ como **W**, podem ser considerados como codificadores do conhecimento adquirido pela **RN** durante sua fase de treino (cf. ÁVILA GARCEZ; GABRAY, 2001, p. 157). Isto permitirá extrair "conhecimento" simbólico da rede, mas esse é um assunto que vai bem além das pretensões deste trabalho.³

No segundo sentido no qual usamos a expressão *rede neural*, uma **RN** é um programa que simula o funcionamento do *objeto* virtualmente existente, que consistiria na realização da **RN** no primeiro sentido, ou seja, como a estrutura descrita acima.

Em ambos sentidos deve ser clara a relação entre a teoria de **RN** e a de algoritmos. No sentido de programa, uma **RN** roda num computador digital convencional. Como todo programa, deve possuir um conjunto de algoritmos que permite sua execução. O uso de algoritmos para simulação de fenômenos (reais ou virtuais) não algorítmicos não é nada surpreendente e acontece com frequência em computação. Temos programas, por exemplo, para simular fenômenos naturais, como terremotos e tempestades, ou psicológicos, como a conversa entre um psicanalista e

³ O original deste trabalho foi escrito em 1999, e submetido a revisão em outubro de 2001. A relação entre **RN** e métodos simbólicos evoluiu de maneira dramática nesse período e, apesar de meu intuito de atualizar o texto, alguns tópicos podem parecer arcaicos. No entanto, o problema central da estrutura lógica das **RN's** não tem merecido quase nenhuma exploração, o que ainda o mantém atual.

seu paciente. Muitos devem lembrar o famoso Dr. Biatso produzido para o antigo **DOS**, que divertiu aos jovens da década de 80, mostrando como o algoritmo podia ser mais inteligente que a conversa real do analista.

No primeiro sentido, uma **RN**, vista como estrutura, não é um objeto algorítmico. Com efeito, a estrutura (*) captura a *forma* de uma rede neural real, em seus aspectos mais simples; é uma representação *esquemática* de uma teia de neurodos. Mas, por sua vez, é um projeto de processador não pode ser reduzido a um algoritmo. [Ou seja, teoricamente, deveria existir um problema formal bem fundado P, tal que existe uma rede neural que resolve P, mas nenhum algoritmo resolve P].

Na clássica apresentação de McClelland, Rumelhart e Hinton, (RUMELHART; MCCLELLAND, 1987, p. 10 et seg.) as estruturas de processamento paralelo a cujo gênero pertencem as **RN**, são vistas, fundamentalmente, como modelos que "dão conta" (desde uma perspectiva computacional e psicológica) de muitos mecanismos de nossa intuição que não são perfeitamente capturáveis pela computação seqüencial. Mesmo admitindo que o pensamento humano (e a maioria dos processos estudados pela IA, para não falar da ciência cognitiva em geral), se apresenta como uma seqüência de estados, não obstante, é impossível capturar a estrutura interna desse pensamento usando apenas recursos seriais. (RUMELHART; MCCLELLAND, 1987, p.11-12). Ou seja, a vivência de nosso pensamento pode ser vista serialmente (linearmente) ordenada, de tal maneira que cada um de seus componentes é um *estado mental* e que, dados dois estados quaisquer **E** e **E'**, um de ambos precede ao outro no tempo. Contudo, a decomposição de qualquer um desde estados E's não poderá ser feita apenas por processos seriais.

De acordo com este enfoque, a teoria da aprendizagem, ponto relevante no desenvolvimento das teorias de processamento paralelo, não assume como objetivo do aprendizado de uma **RN**, a formulação de regras explícitas, mas a intensificação das conexões numa rede de modo que esta atue *como se* (na expressão dos autores) conhecesse as regras (cf. p. 32).

Esta formulação transparente da epistemologia subjacente à teoria das **RN** mostra que essas estruturas capturam, de maneira relativamente precisa, as características próprias da cognição biológica, especialmente a humana. De fato, seres vivos agem em função de estímulos e aprendem diversas destrezas mesmo sem conhecer as regras desse aprendizado. Além disso, nem sempre precisam uma formulação simbólica do conteúdo do mesmo.

Esta ênfase no caráter "mais que simbólico" e, em particular, "mais" que digital e que algorítmico das **RN**, se manifesta com maior

intensidade nos modelos aplicados a psicologia cognitiva. Isto cria uma espécie de círculo vicioso. Por um lado, o caráter "mais que simbólico" da mente real é capturado nas **RN**, mas, por outro, se justifica esse caráter "mais que simbólico" das **RN** por seu aparente desempenho na explicação de fatos da psicologia cognitiva, especialmente em seu aspectos evolutivos. Em particular, o enfoque paralelo é defendido por alguns autores [vide REIJMAKERS et al., 1996, p. 102 et. seg.] como base da teoria epigenética e contraposto ao caráter "rígido" dos sistemas simbólicos.⁴

De fato, a maior flexibilidade das estruturas de **RN** é conseqüência óbvia da flexibilidade da cognição real, e isso se reflete nos programas de **RN**, cujos algoritmos possuem maior versatilidade que os dos programas convencionais.

Essa maior 'força' das **RN**, em relação com os algoritmos não deterministas, pode ser utilizada para considerar as estruturas de algoritmo com um *caso particular* das redes neurais.

Com efeito, lembremos a estrutura apresentada em (*):

$$\mathbf{A} = \langle \{u_\lambda\}; a(t); \{f_\eta(a_i(t))\}; W_j; \mathfrak{R}; E \rangle$$

Podemos considerar o seguinte:

- 1) As unidades $\{u_n\}$, agora isoladas, são os *dados* iniciais do problema. Isto é compatível com a definição de *neurodo*,⁵ uma vez que ele é pontual e, no caso geral, pode ser concebido como transmissor e armazenador de dados. Agora, ao considerar os neurodos isolados, eles não podem receber nem transmitir informação nova, mas carregam eles mesmos, *toda a informação* que foi depositada em cada um dos dados. É o caso dos algoritmos convencionais que não possuem *história* da informação nem são suscetíveis de aprendizado.
- 2) Estando as unidades isoladas, o conjunto de suas conexões é vazio ou seja, $W = \emptyset$.
- 3) Sendo os algoritmos independentes do tempo, a função vetorial de ativação é uma constante, cujos componentes representam as coordenadas numéricas de seus dados (que agora são os \mathbf{u}). Isto é

⁴ Este exemplo é puramente ilustrativo, e não visa apoiar a teoria psicológica em apreço. Em torno destas aplicações algo espetaculares do paralelismo jaz a idéia de justificar algumas teorias psicológicas empiricamente não bem fundadas, como é o caso da psicologia dita genética.

⁵ O neologismo 'neurodo' é utilizado para designar os "neurônios" artificiais das **RN**, e são exatamente as unidades \mathbf{u} .

compatível com o fato de que $\mathbf{a}(\mathbf{t})$ é uma função real, uma vez que as coordenadas serão números reais recursivos, portanto, reais.

- 4) Nesse caso, as funções f podem ser consideradas como transformações dos dados em saídas, o que é compatível com o sentido original da definição da estrutura de **RN**, mas também com o conceito de algoritmo. Neste caso, as funções devem ser *recursivas* (elas são aplicadas a valores de $\mathbf{a}(\mathbf{t})$ que são números recursivos e produzem saídas também recursivas).
- 5) O ambiente **E** é inessencial, uma vez que um algoritmo convencional não está modificado por ele, como acontece nos algoritmos quânticos. Portanto **E** por ser tomado arbitrariamente grande (pode supor-se que é igual a \mathbf{R}^3).
- 6) As regras para *alimentar* as funções f , ou seja, as *receitas* que explicam como relacionar os dados u , com suas representações e seus resultados depois de aplicar as f , devem estar no conjunto \mathfrak{R} . Este tipo de regras não estão previstas nas RN, mas podem ser consideradas um caso trivial das regras de Hebb. Ou seja, uma regra de aprendizado modifica a capacidade do sistema de adquirir informação. Uma regra de *instrução* pode ser pensada como uma regra que fornece informação com um grau de modificação *zero*.

A situação parece simples na relação **RN** convencional (não probabilística) e algoritmo determinista. Ao introduzir sistemas aleatórios em ambos (**RN** e algoritmos) esta relação se mantém.

As dificuldades maiores, no entanto, está na unificação desta estrutura com as **RB** e os **AG**.

Redes Bayesianas

Nossa conjectura do começo é que pode existir uma estrutura não trivial que unifique os principais métodos computacionais, como **RN**, **RB** e **AG**.

Apesar de ser uma rede no sentido conjuntista (um conjunto dirigido), as redes bayesianas não são propriamente redes neurais, já que sua estrutura não exige a existência de componentes como funções de ativação, regras de aprendizado, e assim em diante.

Para fixar idéias, definimos uma rede bayesiana (**RB**), como uma estrutura formal, que também neste caso é de ordem superior ao 1o, tal que: $\mathbf{B} = \langle D, R \rangle \dots (**)$ sendo:

- 1) D uma estrutura de tipo menor, cujo domínio é B , um conjunto finito de unidades u_n (os nós), e uma relação \prec não necessariamente total. Os nós representam as variáveis nas quais *colocamos* nosso conhecimento, e a relação estabelece uma conexão que não é pensada como "dinâmica", como no caso das **RN**.
- 2) R é um componente representacional, que armazena a informação de tipo estocástico correspondente a cada estado das variáveis. Do ponto de vista lógico, este componente pode ser enquadrado em diferentes categorias, por exemplo, como subconjunto do produto cartesiano do conjunto de estados vezes o conjunto de probabilidades codificadas, ou como um σ -anel que representa conjuntos borelianos que correspondem às medidas estocásticas dos estados, etc. A forma matemática específica não é essencial para estas considerações.

Intuitivamente, o *estado* de uma variável é o dado específico depositado sobre ela em cada *momento*, ou, em termos mais precisos, qualquer uma de suas medidas.

Por exemplo, se a rede está associada a um sistema especialista para diagnóstico cardíaco, a variável pode ser o ritmo do coração do paciente. O estado da variável será o valor numérico em cada caso, com independência de que cada um desses valores seja ou não correlacionado com o momento em que foi medido. (Normalmente, a idéia de *estado*, em qualquer área da informática, é emprestada da física, onde a variável tempo é essencial. No entanto, em muitos problemas reais só interessam aproximações globais. No exemplo dado, pode interessar apenas a parte do dia em que foi medida a frequência cardíaca).

Como em qualquer outro problema tratável por métodos computacionais, os estados de uma mesma entidade (neste caso, de uma variável) aparecem em número finito. De acordo com a idéia física do estado, dois estados quaisquer E e E' de uma mesma variável x devem ser disjuntos, e a união de todos os estados deve ter probabilidade **1**. (Excluímos variáveis cuja atribuição de valores seja *impossível*. Fora deste caso, o conjunto de estados, mesmo sendo finito, é exaustivo).

Isto permitirá, de maneira natural, introduzir probabilidades de que, para um certo número natural n , a variável corresponda X a esse valor, ou seja, é possível definir de maneira precisa a função: **Prob(X=n)**.

Uma estrutura assim pode ser considerada probabilística, desde que a relação seja interpretada de maneira relevante aos fins estocásticos. No entanto, como as probabilidades introduzidas até aqui não são necessariamente condicionais, será necessário acrescentar novas propriedades para que esta relação coincida com o sentido habitual de "bayesiano".

Mais formalmente:

Dado o domínio \mathbf{B} como conjunto de nós (unidades), e uma unidade $\mathbf{u}_k \in \mathbf{B}$, considera-se uma variável \mathbf{x}_k como símbolo para a informação conduzida por \mathbf{u}_k .

A imagem de \mathbf{B} pela função $\mathbf{u}_k \longrightarrow \mathbf{x}_k$ é o conjunto das variáveis \mathbf{Var} , contido no universo de símbolos da linguagem (não necessariamente de 1ª ordem) da teoria das \mathbf{RB} .

Para duas unidades $\mathbf{u}, \mathbf{u}' \in \mathbf{B}$, a relação $\mathbf{u} \prec \mathbf{u}'$ pode ser considerada como um pré-ordem.

Intuitivamente, é uma relação de ancestralidade, o que sugere alguma analogia com os algoritmos genéticos. O \mathbf{u} que precede ao \mathbf{u}' na pré-ordem pode ser pensado como o "gerador" desse \mathbf{u}' .

Essas relações ficam determinadas, como é de esperar, pelo teorema de Bayes aplicado a probabilidades condicionais. Assim, por exemplo, se $\mathbf{u} \prec \mathbf{u}' \prec \mathbf{u}''$, sendo \mathbf{x}, \mathbf{x}' e \mathbf{x}'' as correspondentes variáveis, então a crença que "merece" \mathbf{x}' , sob a hipótese de que se conhecem dados relativos a \mathbf{x} e \mathbf{x}'' , é dada por:⁶ $\text{Crença}(\mathbf{x}') = \text{Prob}(\mathbf{x}' | \mathbf{x} \& \mathbf{x}'')$ ($\#$) onde a barra vertical indica, como é usual, a probabilidade de \mathbf{x}' respeito da conjunção ($\mathbf{x} \& \mathbf{x}''$). A expressão "Prob" é uma especificação da probabilidade μ , já mencionada, ao caso clássico (probabilidades freqüenciais ou obtidas por convergência).

Exceto no caso da igualdade ($\#$), ou seja, no caso da probabilidade de uma variável em função das variáveis dos nós contíguos, assume-se que os valores das probabilidades das variáveis em quaisquer nós são independentes dos valores assumidos pelas variáveis em outros nós.

Então, uma rede bayesiana é uma estrutura ($\mathbf{**}$) com as condições especificadas em (1) e (2), tal que três variáveis que correspondem a nós pré-ordenados satisfazem a igualdade ($\#$).⁷

Em princípio, pode parecer que as estruturas correspondentes a \mathbf{RN} e \mathbf{RB} são de um grau diferente de riqueza, até porque o *suporte* das \mathbf{RB} pode ser definido em primeira ordem. No entanto, as conexões entre os

⁶ A teoria matemática de probabilidades condicionais é assumida aqui como conhecida. Para o estado da arte em redes bayesianas especificamente, pode ver-se Zhang, cujos resultados chegam até 1996. Resultados mais recentes aparecem de maneira misturada no volume de *Artificial Intelligence* de 2001.

⁷ Para uma introdução amigável às redes bayesianas relativamente recente, vide KWOH.

nós das **RB** estão regidas por probabilidades, o que faz que a atribuição de valores às mesmas consista na atribuição de números reais. Isso implica, pelo menos, uma definição de número real análoga à necessária na atribuição de valores aos pesos do conjunto **W** da estrutura **A** acima, que regem as relações entre os nós (unidades ou neurodos) das **RN**. Considerando a formulação de probabilidades em sua forma intuitiva, que exige a existência de um universo U e de uma σ -álgebra, a estrutura subjacente às **RN** também será de ordem superior.

Por tanto, tudo parece indicar que tanto as **RN** como as **RB**, consideradas em sua plenitude, são de ordem superior à 1a. Mesmo que suas ordens lógicas não fossem equivalentes, em ambos casos o problema seria reduzível a 2a. ordem. (Isto não significa que seu nível lógico seja o mesmo, como será comentado em seguida. As **RB** parecem estar no mesmo nível que a teoria de problemas, enquanto que as **RN** e os algoritmos estariam num nível superior.)

Na teoria clássica de **RN** é obscuro, como acontece quase sempre na matemática intuitiva, qual é a "quantidade" de teoria de conjuntos a ser utilizada. No entanto, ela deve ser, pelo menos, a necessária para definir o diferencial de uma função de várias variáveis, já que isto será utilizado no cálculo do gradiente da função erro para alguns casos clássicos, como a rede de retro-propagação (Cf. KOVÁCS, 1996, p. 71).

No caso das **RB** é mais simples, a teoria de conjuntos necessária é exatamente a que permite definir a teoria da medida. No entanto, não há motivo para pensar que uma das *quantidades* de conjuntos necessários seja *menor* do que a outra. Aparentemente, em ambos casos precisamos ordinais contínuos. Mas, modelos contínuos poderiam ser *grandes* demais, já que as teorias computacionais só precisam ordinais enumeráveis. No entanto, não é claro que a matemática necessária para todas estas teorias possa ser desenvolvida num universo menor.

Uma formalização minuciosa de todos os conceitos usados até agora, incluindo **RB**, seria extremamente tediosa e complicada. Não obstante, conjecturamos, em princípio, que o *nível lógico* das **RB** é *menor* que o dos **RN** ou o dos algoritmos, independentemente de que a complexidade matemática possa ser a mesma, e de que todas possam ser reduzidas a 2a. ordem. Os algoritmos podem ser definidos, de maneira geral, num ambiente mais pobre. De acordo com nossa definição anterior de algoritmo \mathfrak{S} , a matemática necessária é a teoria das funções recursivas, bastando, portanto, uma teoria recursiva de números. O fato de que sejam necessários recursos analíticos para o desenvolvimento desta teoria é paralelo a, por exemplo, o

uso de métodos analíticos na teoria de números, mas não é imprescindível para sua definição. Já a teoria de RB não pode ser desenvolvida sem uma teoria da probabilidade, mesmo que nos restrinjamos à sua parte algébrica.

No entanto, o nível lógico das **RB** é *mais baixo*, pois está no mesmo nível que a teoria de problemas. Com efeito, dado um problema **P** adequadamente formalizado, a estrutura que o representa pode ser pensada como argumento de um algoritmo ou de uma **RN**. Por exemplo, à estrutura que representa o problema de extrair a raiz quadrada de um número real positivo pode ser aplicado um caso particular do algoritmo que calcula raízes por aproximação. Isto tem o aspecto de valer em geral: ou seja, se um problema é de nível lógico **n**, então, um algoritmo que o resolva deve ser de nível lógico (**n+1**), mas não tenho uma demonstração rigorosa que apóie esta intuição.

Já uma rede bayesiana é ela mesma um objeto do mesmo nível lógico do problema do qual surgiu. Por exemplo, uma rede bayesiana que modela um certo sistema especialista é da mesma ordem que este.

Portanto, uma **RB** não poderá ser obtida como caso particular uma **RN**. A aplicação metodologicamente mais plausível será *colar* redes bayesianas em alguns segmentos de uma **RN**. Uma de minhas conjeturas é que os pesos **W** podem ser substituídos, sem prejudicar suas propriedades, por RB e não apenas por pesos individuais de tipo probabilístico.

Nas redes bayesianas ordinárias, a crença depositada numa variável pode ser calculada conhecendo os estados das variáveis conectadas com ela. Esta idéia pode ser adequadamente transferida para o caso de redes neurais. Ainda, nesse caso será necessário entender em que sentido a crença influi em outros elementos da rede.

Na teoria de **RN**, o uso de probabilidades condicionais é um recurso *externo* (no caso de considerar a teoria como um conjunto de sentenças, essas probabilidades estariam sendo usadas na metalinguagem) para avaliar o aprendizado de uma rede neural quando ela é treinada. As propriedades bayesianas, portanto, permitem saber *de fora da rede* como ela está funcionando, mas não estão representadas por funções e regras da própria rede.

[O problema mantém alguma analogia que cito apenas por razões heurísticas; é, por exemplo, o caso da demonstração aleatória de teoremas. Se quisermos demonstrar que α é consequência lógica de um conjunto de hipóteses Δ , a prova é *determinista*; não há nela nenhuma alusão a probabilidade. Entretanto, se não tivermos a prova, faz sentido perguntar-se qual é a probabilidade da meta-sentença "De Δ deduz-se α ".]

Inclusive, esta aplicação externa de probabilidades condicionais nem sempre implica lidar com as redes bayesianas em sentido estrito. Na prática, na avaliação de treinamento de redes, o conceito de probabilidade pode ser tomado como equivalente ao de *valor difuso* [nebuloso \equiv *fuzzy*], pois, mais que utilizar as propriedades matemáticas da probabilidade, o que interessa é ter um certo espectro de *graus* de aceitabilidade.

No entanto, da mesma maneira em que a computação seqüencial utiliza algoritmos probabilísticos, também é possível pensar métodos de computação massivamente paralela cujos procedimentos sejam *randômicos*. De fato, a teoria da complexidade probabilística usa probabilidades "embutidas" nas redes, o que nos aproxima da possibilidade de relacionar **RN** com redes que usam componentes aleatórios. (Uma das primeiras apresentações desta teoria foi formulada por Parberry; vide BANERJI, 1990, p. 233 et seq.).

Os esquemas de inferência trabalhados cognitivamente por **RN** que usam teoria da probabilidade para expressar informação, podem codificar relações estatísticas entre algumas das características representacionais em apreço. (Vide SMOLENSKY apud RUMELHART MCCLELLAND, 1987, v.1, p.209). Esta propriedade deveria permitir inferir a probabilidade dos diferentes estados do ambiente **E**. Ora, o tratamento macroscópico da informação, por meio de esquemas, é considerado difícil (p. 210) pois não é simples encontrar uma maneira de representar a versatilidade da cognição biológica. Ao reduzir a análise ao nível micro, focalizando a atenção nos átomos de conhecimento, o problema se simplifica, mas nem sempre a diferença entre tratamentos estatísticos e deterministas é relevante. (Observe o comentário no final da p. 210).

Parece, portanto, que as redes bayesianas possuem uma função específica que não é substituída por artifícios probabilísticos usados em **RN**. Por outro lado, capturar a noção de probabilidade condicional numa estrutura que também inclua as **RN** não deve ser difícil.

Para avançar nessa direção, é necessário ter em conta que os pesos de conexão numa **RN**, poderia receber seus atributos numéricos com base em probabilidades obtidas pelo teorema de Bayes, e não apenas como probabilidades fixas, que é que realmente se faz quando se fala em **RN** probabilísticas. Assim, uma conexão entre dois neurodos de índices **i** e **j**, dada pelo peso w_{ij} , poderia estar definida pelos valores obtidos por uma **RB** que tivesse unidades ancestrais e descendentes, vinculadas com u_i e u_j .

Esta *bayesianização* de uma **RN** não lhe faria perder seu caráter conexionista. Com efeito, a matriz de conexão **W** continuará a respeitar a interatividade recíproca, mas agora com pesos fornecidos pelas **RB**.

O problema parece possuir numerosas nuances técnicas, eventualmente, difíceis de abordar, mas não mostra, em princípio, nenhuma dificuldade teórica.

O desafio está, acredito, em encontrar uma estrutura suficientemente fraca como para nos mantermos fora do caso trivial. O modelo minimal transitivo da teoria de conjuntos de Zermelo-Fraenkel não deveria funcionar pela mesma razão que não funciona para as probabilidades físicas.⁸

Redes neurais e algoritmos genéticos

Os algoritmos genéticos [AG] são métodos (formalmente, estruturas) que possuem um algoritmo convencional como base, mas que estão sujeitos a certas operações que *imitam* processos naturais que acontecem entre organismos (como cruzamento ou mutação). A partir de 1995, tentou-se uma unificação entre AG e RN, especialmente no caso de considerar a rede como uma entidade que pode agir como um AG. A metáfora biológica compartilhada por ambos e a preservação da história do processo parecem favorecer este enfoque.⁹

Vamos tentar analisar, primeiro, a estrutura que captura a forma lógica de um AG, e, depois, a estrutura da combinação de AG com uma rede neural.

No entanto, é necessário diferenciar, como acontece no caso de RN e outras estruturas, estas duas perspectivas: (a) combinar procedimentos para obter novos resultados, por exemplo, RN com AG, de maneira que um deles seja aplicado ao outro; (b) e analisar a conexão lógica e fundamentacional de ambos para determinar se um deles baseia ou não o outro.

É nesse último sentido que se desenvolve esta proposta. Obviamente, probabilidades condicionais, inclusive dispostas em RB, podem ser usadas como método para estudar uma RN, e algoritmos quaisquer podem ser aplicados a quaisquer destas estruturas para simular-las

⁸ Este resultado se deduz de uma demonstração para o universo que serve de portador à matemática necessária para a Mecânica Quântica dada por Paul Benioff, do Argonne National Laboratory em 1977. Minha posterior "navegação" por esse campo, influenciou em propostas similares para o campo das RN. No entanto, a citação completa deste assunto foge dos limites deste artigo.

⁹ Tentativas de misturar redes neurais com algoritmos genéticos são relativamente recentes. Vide, por exemplo, Romaniuk em Chambers, p. 75 ss., mas as motivações já aparecem em Holland.

digitalmente. No caso dos **AG**, o duplo uso do método "intuitivo", por um lado, e do fundamento lógico por outro, é ainda mais notório. Há uma confortável modelagem da forma em que uma **RN** pode ser construída usando **AG**, de modo que o **AG** permita controlar seu "crescimento" (Cf. CHAMBERS, 1995, *passim*).

Apesar de que o problema pode ter desdobramentos epistemológicos complexos que não desejo tratar aqui, esta dualidade em geral pode ser exemplificada com um caso mais evidente: o da dupla relação entre uma teoria eletrônica e uma teoria computacional. A última pode servir de ferramenta para calcular, por métodos numéricos computadorizados, o valor das equações de Maxwell para certos campos. Já uma teoria eletrônica serve de base *lógica*, para deduzir, a partir dela, a teoria do funcionamento de um circuito. Isto acontece, por exemplo, quando provamos que um circuito com dois switches paralelos, se corresponde com a disjunção fraca de duas sentenças.

A aplicação de **AG** no desenvolvimento de redes, faz parte da colaboração geral entre modelos computacionais. Todavia, o que pretendemos ver é qual é a estrutura lógica dos **AG**, como se relaciona, se isso é possível, com as de **RN**, e, avançando na direção geral de nossa proposta, como podem ser capturados os três conceitos (**RN**, **RB**, **AG**).

Começamos observando o papel dos algoritmos nos processos de cognição. Um sistema de cognição de certa complexidade, como é o da mente humana e seus *periféricos* (sistema motor, sistema perceptivo), possui um subsistema central que armazena as funções consideradas *superiores*, como inteligência, memória, atenção, etc.¹⁰ Como é habitual, reduzimos a sua mínima influência os processos emocionais, volitivos e afetivos cujo conteúdo de informação não seja significativo.

Em princípio, conhecer é um processo que parece começar com a percepção e combina dados já armazenados pelo subsistema do sujeito, incluindo, eventualmente, componentes cuja origem não possa ser atribuído aos dados externos.

Operacionalmente, o conhecimento supõe a resolução de problemas. Como sabemos, alguns problemas fogem de qualquer caracterização rigorosa e não pode esperar-se que sejam resolvidos de maneira objetiva. No entanto, há uma certa família **P** de problemas que podem ser expressos de maneira formalmente satisfatória. Considerando estes problemas como abstratos, ou seja, como estruturas, aqueles que são

¹⁰ Existe atualmente um consenso sobre esta maneira de ver a cognição. Tenho me guiado especialmente pelo manual de Posner, mas qualquer outro texto moderno serve.

resolúveis por algoritmos são *menos* que os não resolúveis, embora talvez seja suficiente, para os fins cognitivos usuais, poder resolver apenas os algorítmicos.¹¹

Então, um algoritmo permite aumentar nossa carga cognitiva ao resolver um problema. De fato, a solução, se existe, é uma nova informação.

Como foi comentando ao nos referirmos aos algoritmos em geral, os algoritmos deterministas (e ainda os probabilistas convencionais) constituem uma formalização de uma parte muito pequena de nossa maneira de resolver problemas.

Com efeito, enquanto o sujeito humano possui numerosas estratégias flexíveis para resolver problemas, o algoritmo age de maneira rígida e seqüenciada. Ele aplicará um conjunto de regras inicialmente fixado, sobre uma classe de funções recursivas, algumas das quais (não necessariamente todas) contêm as transformações necessárias para modificar os dados e obter a solução.

Esta rigidez não se modifica por recursos convencionais, e exige utilizar **RN** ou outros recursos de **IA**. No entanto, é possível adicionar um acréscimo aos algoritmos convencionais para permitir que eles capturem a história do processo desde o começo da abordagem do problema.

Estes algoritmos *evolutivos* (de fato, os **AG**) incorporam um parte mais dinâmica de nosso processo biológico de resolução de problemas, que aparece de maneira totalmente estática nos algoritmos convencionais.

	Algoritmos convencionais	Algoritmos genéticos
O que simulam, na resolução de problemas?	O processo de aplicar regras passo a passo, com base no conhecimento de passos anteriores e dados fixos (por exemplo, as tabuadas)	O processo de aplicar regras passo a passo, com base no conhecimento de todo o processo prévio de resolução do problema e de dados que podem ser reformulados durante a própria aplicação do AG
A que operações podem ser submetidos?	Composição, substituição, eventualmente, operações aritméticas se o tipo de algoritmo as comporta	Operações que <i>simulam</i> certos fenômenos biológicos, como <i>reprodução e cruzamento</i>
Aspectos comuns com...	São originários	Relações de ancestralidades similares à RB , e modificações <i>espontâneas</i> , tipo RN

¹¹ Na prática, o problema, considerado como coisa real, está circunscrito ao espaço natural e, portanto, satisfaz as propriedades de separação desse espaço. Então, não podem existir *infinitos* problemas concretos, exceto num sentido potencial. Nesse caso, também os problemas não resolúveis por algoritmos são enumeráveis.

Uma estrutura de *algoritmo genético* deve capturar noções consideradas relevantes para a resolução de problemas, as transformando em conceitos que, mesmo conservando sua motivação biológica, tornam-se *artificiais*, podendo, portanto, serem simulados em computadores ou ainda construir-se em projetos de hardware. Trata-se de um passo ainda mais analítico na antiga tarefa de moldar sob formas lógicas (e depois, em realizações físicas) os métodos biológicos e psíquicos de resolução de problemas.

De acordo com o uso corrente de AG, parece que a seguinte estrutura deveria capturar suas principais propriedades.

$G = \langle \mathfrak{S}, \Xi \rangle$ (***) sendo: \mathfrak{S} um algoritmo convencional a ser aplicado sobre seu conjunto de dados, e eventualmente, sobre a estrutura Ξ que faz parte do AG.

Ξ uma estrutura cujos componentes terão uma complexidade que depende do caso específico, podendo variar de ordem. Assumimos, porém, que o domínio de Ξ é único e será chamado X .

Quando $X = \emptyset$, encontramos a estrutura tradicional do algoritmo convencional. Ou seja, (***) se reduz a \mathfrak{S} :

O domínio X simula uma população de entidades biológicas: organismos, tecidos, células ou, mas freqüentemente *cromossomos*.

O conceito de *cromossomo*, na simulação que os AG fazem dos processos biológicos, é atribuído, dependendo dos autores e das teorias específicas, a entidades diversas. No entanto, nesta apresentação vamos assumir que os cromossomos são os elementos da população que é o domínio da estrutura Ξ . Ou seja, γ é um *cromossomo* se e somente se $\gamma \in X$.

As operações a que estão submetidos os elementos da população por parte das funções e regras que fazem também parte de dependerão, é claro, da especificação da estrutura para cada problema particular.

No entanto, os *cromossomos* computacionais são suscetíveis de gerar, como seu colegas reais, pelo menos os seguintes processos.

1. Mutação.
2. Reprodução.
3. Cruzamento.

Continuando com o paralelismo natural, os *cromossomos* possuem genes, cujos valores (indicados por dígitos binários ou equivalentes) poderão ser *alterados* para as próximas operações. Como, de fato, estes **AG** são processados em computadores convencionais deterministas, a mutação deverá estar pré-programada.

O **AG** também simula a reprodução de organismos, separando população em duas metades que são compatíveis para se *reproduzirem*.

Com base nessa divisão, os **AG** são capazes de gerar *cruzamentos*, fazendo com que "pais" reprodutores possam produzir novos elementos populacionais.

Veja-se, pelo menos superficialmente, que os elementos das populações se reproduzem e são capazes de receber e *transmitir herança*. Então, a informação probabilística poderá ser comunicada também desse modo, incluindo probabilidades condicionais. Parece natural que a relação biológica de *hereditariedade* capturada pelos **AG**, possa ser tratada em termos de **RB**, de maneira análoga a como foi sugerido no caso das conexões ponderadas **W** das **RN**.

O fato de que hereditariedade seja uma relação num único sentido, enquanto que a conectividade das **RN** é uma interação fortemente dinâmica, não seria um empecilho, uma vez que as **RB** são pensadas apenas, neste caso, como fornecedores de informação probabilística com base condicional, calculada pelo teorema de Bayes.

Conclusões

Apesar do aspecto formal com que aparece na literatura, um algoritmo é também uma abstração de uma propriedade operacional de organismos biológicos, pelo menos, de mentes humanas, que consiste na execução de atos que podem ser perfeitamente completados sem nenhum apelo à criatividade, à imaginação ou a qualquer forma de fantasia subjetiva. Em termos cognitivos, um algoritmo é uma seqüência de passos, cada um dos quais é aplicado obedecendo a uma regra que pode realmente ser aplicada de uma maneira única e precisa, e que pode ser compreendida por qualquer mente estatisticamente normal. Sua base é a digitalidade, comum ao processo artificial e, parcialmente, aos padrões de reconhecimento cognitivo de nossa percepção.

O algoritmo convencional também possui, então, inspiração biológica. De fato, as primeiras "receitas" para fazer contas foram extraídas da própria capacidade humana de decorar "tabuadas" e aplicar processos de contagem.

Aplicar um algoritmo é, portanto, uma combinação de obediência a regras claras e objetivas, de uso da memória e de composição de operações (por exemplo, compomos operações de contagem para obter uma soma entre naturais, e compomos somas entre naturais para obter multiplicações).

Esta composição, que é reconhecível experimentalmente em alguns testes psicológicos sobre a execução de operações aritméticas simples, encontra sua formalização nas regras para compor funções numéricas que mantêm a recursividade. (Por exemplo, composição em sentido estrito, substituição, separação, etc.)

A certeza de que os algoritmos convencionais podem ser facilmente formalizados foi sempre tão grande, que mesmo o conceito de algoritmo apareceu tardiamente, e o estudo de sua estrutura lógica é um processo recente. Não obstante, as propostas de receitas para resolver problemas são bem antigas.

Apesar da grande variedade de problemas que podemos resolver por algoritmos convencionais, e sua possibilidade de combina-los com nossa memória ou com a de um computador, as regras do algoritmo só atuam sobre os passos imediatamente anteriores. Um algoritmo não possui, por si mesmo, história. Considerado então como estrutura, ele possui características precisas:

Sua aplicação é seqüencial, exige digitalidade e suas regras não apreendem.

Já os **AG** incorporam, por analogia com a evolução biológica, a idéia de história da informação.

Essa história se transmite através do processo de evolução, imitado pelo **AG** por meio de operações que este pode executar sobre elementos da população e que são mais amplas que as operações restritas que executa um algoritmo convencional, por exemplo, mutação, *reprodução e cruzamento*.¹² O que dá sua particularidade ao **AG** é o fato de que é possível definir, num algoritmo convencional, operações novas, mas flexíveis, acrescentando uma estrutura populacional cujas operações internas permitem que o conceito de cromossomo, gene, organismo, etc., tenham um equivalente no plano computacional.

¹² As regras de mutação, reprodução e análogas nos algoritmos, podem ser também pensadas como operações sobre os mesmos algoritmos. Uma analogia puramente metafórica, mas que pode ser de interesse aos que possuem uma visão lógica do problema, é a que existe entre estas regras especiais dos **AG** e as regras estruturais de uma dedução por seqüentes. Por sua vez, as regras recursivas normais podem ser comparadas com regras dedutivas que dependem das fórmulas. No entanto, este "parecido" não deve ser levado muito longe, pois possui limitações até na própria programação lógica.

A informação transmitida por ancestralidade pode não ser determinista, na medida em que existam mecanismos aleatórios que possam ser simulados computacionalmente. Ora, essa informação probabilística, pelo fato de ser transmitida evolutivamente, não precisa cingir-se a probabilidades absolutas: podem ser usadas probabilidades condicionais no sentido do teorema de Bayes.

O processo de atribuir probabilidades condicionais, tanto a cromossomos de AG como a conexões ponderadas de RN, que são fortemente interativas, não é perturbado pela diferença entre ambos objetos. De fato, o que desejamos obter ao final são valores de medida que interpretamos como informação. Uma RN poderá aprender novos valores se for treinada em colaboração com uma RB, e um AG poderá fazer *berdar* valores já obtidos.

Por sua vez, as RB parecem precisar de uma ancoragem num artefato computacional, já que, em si mesmas, fornecem informação mas não são métodos de resolução de problemas.

Em princípio, uma estrutura de ordem *grande* poderia abranger estas estruturas.

Com efeito, o modo mais natural de gerar uma estrutura comum a **RN**, **RB** e AG consiste em adicionar os componentes das estruturas, assim: $X = [U, \prec, a(t), f_k, W, \mathfrak{J}, \mathfrak{R}, E]$ onde **U** é o domínio da estrutura e, portanto, o universo das unidades ou nós, para cada um dos casos em que isto seja necessário.

Como foi visto ao observar a dedução de algoritmos a partir de **RN**, no caso de não conectividade, **U** pode ser pensado apenas como o domínio dos dados.

\mathfrak{J} é o algoritmo convencional que serve de base à estrutura (***) e **E** pode ser considerado o ambiente, no caso das **RN**, ou a população, no caso dos **AG**.

Os outros componentes são típicos da estrutura (*), exceto \prec que é característico da estrutura (**) de RB. Este pré-ordem, não obstante, poderia ser entendido, intuitivamente, como a relação de ancestralidade, tanto em (*) como em (**). Há algumas dificuldades para tornar essa intuição mais precisa. No caso das **RN**, por exemplo, esta relação de ancestralidade estaria definida entre a informação dos neurodos e não entre eles mesmos, se quisermos fazer jus ao funcionamento dessas redes. Em fim, o assunto precisa elaboração, ainda.

Em princípio, esta mega-estrutura \mathcal{X} pode ser considerada como gerador das outras, por uma adequada *restrição de tipo* (ou seja, *esquecendo* alguns dos componentes) e particularizando outros. Como já vimos ao deduzir os algoritmos convencionais das **RN**, uma particularização é considerar os pesos das interconexões como zero, ou, então, \mathcal{W} como vazio.

A restrição de tipo seria análoga à que é bem conhecida da álgebra. Assim, um grupo ordenado, por exemplo: $[\mathbf{G}, \circ, \mathbf{e}, \prec]$ com a restrição de tipo que omite o último símbolo, se transforma numa estrutura de grupo: $[\mathbf{G}, \circ, \mathbf{e}, \prec]$

Mas isto é possível porque os axiomas de \prec (no caso, reflexividade, antisimetria, transitividade, e compatibilidade com \circ) podem ser *omitidos* da lista geral de axiomas, uma vez que são independentes dos outros. Ao eliminar-los, a teoria se restringe aos axiomas restantes que serão satisfeitos por estruturas *menos exigentes*.

No caso das **RN**, **RB** e **AG**, não é habitual formular axiomas, o que é compatível com o carácter difuso da matemática intuitiva (Mesmo na física, a formulação específica de axiomas ocorre apenas quando é exigido por problemas de fundamentação.) Mas as propriedades que são enunciadas como se fossem axiomas não são de primeira ordem, e sua formulação é, em alguns trechos, impredicativa. Como deve, então, ser eliminado um *axioma* que contenha uma propriedade relativa, por exemplo, a RB, quando se deseja deduzir da estrutura, apenas o caso de **AG**?

Este é um problema complicado. Sua solução equivalerá a dar uma base lógica sólida à matemática usada para redes neurais, como se fez recentemente em outros casos de tratamento logicamente difícil, embora nem tanto como estes.¹³

Uma primeira aproximação deveria ter em conta qual é exatamente a matemática que estas redes precisam. Provavelmente, seria útil um enfoque que utilizasse apenas análise recursivo no sentido feito clássico por Goodstein e outros.¹⁴

¹³ Por exemplo, em 1983, Kopperman encontrou uma formulação de 1ª ordem para a teoria de espaços topológicos separados. O assunto não é de interesse neste artigo, mas serve para mostrar de que maneira seria necessário reduzir uma teoria cuja formulação matemática é apenas "intuitiva" para transformar-la em logicamente transparente.

¹⁴ A análise recursiva é pouco mais que uma curiosidade para lógicos e epistemólogos, mas pode servir de critério para delimitar "quanta análise" vamos utilizar. Uma forma amigável de ver esta teoria está no texto do próprio Goodstein, *Recursive Analysis*, 1964 e nos capítulos sobre recursividade de Bell & Machover, *A course in mathematical logica*, 1976.

Caso seja possível formular uma axiomática consistente com enunciados predicativos finitos, o qual é esperável, deveríamos ter uma lista de sentenças ($\Sigma\xi$) tal que, para cada uma das estruturas típicas de **RN**, **RB** e **AG**, existisse uma sub-seqüência ($\Sigma\xi_i$) cujo modelo fosse exatamente a estrutura prefixada.

Isto ajudaria a determinar um modelo não trivial, que podia ser adotado como modelo padrão.

Talvez a aritmética de 2ª ordem possa servir de inspiração a esta construção.

Referências

- ANDERSON, J. A.; ROSENFELD, E. (Ed.) *Neurocomputing*. Cambridge, Mass: The MIT Press, 1989.
- AVILA GARCEZ, B.; GABBAY. Symbolic knowledge extraction from trained neural networks. *Artificial Intelligence*, n. 125, p. 155-207, 2001.
- BANERJI, R. B. (Ed.). *Formal techniques in artificial intelligence*. Amsterdam: North-Holland, 1990.
- BELL, J. L.; MACHOVER, M. *A course in mathematical logic*. Amsterdam: North Holland, 1976
- CAUDILL, M.; BUTLER, C. *Understanding neural networks: computer explorations*. Cambridge, Mass.: The MIT Press, 1992. v.1.
- CHAMBERS, L. (Ed.). *Genetic algorithms*. Boca Raton, Florida: CRC Press, 1995. v.1.
- GOODSTEIN, R. L., *Recursive analysis*. Amsterdam: North Holland, 1964
- HAUGELAND, J. *Artificial intelligence: the very idea*. Cambridge, Mass: The MIT Press, 1990.
- HILLIS, D. *The connection machine*. Cambridge, Mass: The MIT Press, 1985.
- HUTH; RYAN. *Logic in computer science: modelling and reasoning about systems*. Cambridge: Cambridge University Press, 2000.
- KOVÁCS, Z. L. *Redes neurais artificiais*. São Paulo: Edição Acadêmica, 1996.
- KWOH, C.-K. et al. Using hidden nodes in Bayesian networks. *Artificial Intelligence*, n. 88, p. 1-38, 1996.
- LUNGARZO, C. A face exata da ciência cognitiva. In: VIDAL, V. *Ciências cognitivas: um enfoque multidisciplinar*. Rio de Janeiro: Ed. Fundação Oswaldo Cruz, 200?.
- _____. *Questões lógicas nas redes neurais probabilísticas*. Salvador: Publicações da UFBA, 1995.
- _____. *The notion of linearity in neural networks*. In: INTERNATIONAL COLLOQUIUM ON COGNITIVE SCIENCE, 4, San Sebastián. Espanha: [s.n.], 1995.
- _____. *Logical reasoning under conflictive circumstances*. In: Wittgenstein Symposium, Kirchberg, Austria, 1993.

LUNGARZO, C. *Dificultades para la determinación del status de teorías de la computación*. In: INTERNATIONAL SYMPOSIUM ON STRUCTURES IN MATHEMATICAL THEORIES, San Sebastián. Espanha: [s.n.], 1990.

MANBER, U. *Introduction to algorithms*. Reading: Addison-Wesley, 1989.

RAIJMAKERS, M. E. J. et al. On the validity of simulating development by means of PDP networks. *Cognitive Science*, n. 20, p. 101-136, 1996.

RUMELHART, D. E.; MCCLELLAND, J. L. *Parallel distributed processing*. Cambridge, Mass: MIT Press, 1987.

VAPNIK, V. N. An overview of statistical learning theory. *IEEE Transactions of Neural Networks*, v. 10, n. 5, 1999.

ZHANG, N. L. Irrelevance and parameter learning in Bayesian networks. *Artificial Intelligence*, n. 88, p. 359-373, 1996.