

PercepEduc: Auxiliando na Aprendizagem das Redes Neurais Artificiais

Laurence Rodrigues do Amaral¹, Eder Teixeira Piau²

¹Curso de Ciência da Computação - Universidade Federal de Goiás/Jataí (UFG)
BR364 Km 192, Setor Industrial, Jataí - GO - Brasil

²Mestrado em Educação - Universidade de Uberaba (Uniube), Uberaba - MG - Brasil

laurence.amaral@gmail.com

Abstract. *The Artificial Neural Networks (ANNs) are an important method inside the Artificial Intelligence (AI), and a member of the connectionist approach of the referred area. The ANNs can be considered "black box" methods, i.e., it is impossible to know how the results observed were obtained. AI is a multidisciplinary area, i.e., it is of common interest for scientists, engineers, philosophers, psychologists and biologists. Thus, completely understand the ANNs is a complex challenge faced for researchers of the AI area. Based on this, we proposed an environment, named PercepEduc, to support the teaching of ANNs. The PercepEduc proved to be very effective, making it easy and simple to learn ANNs.*

Resumo. *Uma das abordagens da Inteligência Artificial (IA) é a abordagem conexionista, tendo como principal método as Redes Neurais Artificiais (RNAs). As RNAs são consideradas métodos "caixa-preta", isto é, é impossível saber porque uma RNA chegou a um dado resultado, dificultando assim o seu entendimento e aprendizado. Sendo a área de IA uma área multidisciplinar, sendo objeto de estudo de cientistas da computação, como de linguistas, filósofos, psicólogos e até de biólogos, desenvolvemos um ambiente de aprendizagem de RNAs, chamado PercepEduc, que utiliza a RNA mais simples, o perceptron, como modelo. O PercepEduc se mostrou bastante eficiente, tornando mais simples e fácil o aprendizado das RNAs.*

Palavras-chave: Inteligência Artificial, Redes Neurais Artificiais, Perceptron, Ensino, PercepEduc

1. Introdução

Durante milhares de anos, procuramos entender como pensamos; isto é, como um mero punhado de matéria pode perceber, compreender, prever e manipular um mundo muito maior e mais complicado que ela própria. O campo da inteligência artificial (IA) vai ainda mais além: ele tenta não apenas compreender, mas também construir entidades inteligentes. A IA é uma das ciências mais recentes. O trabalho começou logo após a Segunda Guerra Mundial, e o próprio nome foi cunhado em 1956 [Russel and Norvig 2004].

Atualmente, a IA abrange uma enorme variedade de subcampos, desde áreas de uso geral, como aprendizado e percepção, até tarefas específicas como jogos de xadrez, demonstração de teoremas matemáticos, criação de poesias e diagnóstico de doenças. A IA sistematiza e automatiza tarefas intelectuais e, portanto, é potencialmente relevante

para qualquer esfera da atividade intelectual humana. Nesse sentido, ela é verdadeiramente um campo universal [Russel and Norvig 2004].

A IA busca prover máquinas com a capacidade de realizar algumas atividades mentais do ser humano. As atividades realizadas por estas máquinas podem envolver a sensopercepção (como tato, audição e visão), as capacidades intelectuais (como aprendizado de conceitos e juízos, raciocínio dedutivo e memória), a linguagem (como as verbais e gráficas) e atenção (decisão no sentido de concentrar as atividades sobre um determinado estímulo) [NascimentoJr and Yomeyama 2004].

O estudo da IA contribui não somente para o aperfeiçoamento das máquinas, mas também para a motivação em se conhecer melhor as próprias atividades mentais humanas. Como por exemplo pesquisas na área de linguagem natural, redes neurais artificiais, lógica, robótica, dentre outras. Em termos de tecnologia, a IA permite que máquinas possam realizar tarefas complexas no lugar do operador humano, liberando-o de atividades enfadonhas, insalubres ou inseguras. Além disso, pode-se aumentar a eficiência do humano na sua interação com equipamentos sofisticados. Permite ainda, que conhecimentos possam ser compartilhados por muitas pessoas, sem que haja necessidade de consultas a especialistas [NascimentoJr and Yomeyama 2004].

A IA está a nossa volta. A lógica nebulosa é amplamente utilizada em máquinas de lavar, carros e em mecanismos de controle de elevador. Os agentes inteligentes nos ajudam a resolver problemas enquanto usamos nossos computadores, além de procurarem documentos que podem nos interessar na internet. Sistemas especialistas são utilizados por médicos para ajudar a descobrir sintomas que são difíceis de diagnosticar ou para prescrever tratamentos em casos em que mesmo especialistas humanos tem dificuldade [Coppin 2010].

IA é uma área que tem atraído a atenção de muitos pesquisadores ao longo de sua história e, embora a idéia possa parecer até simples, a criação de máquinas realmente inteligentes continua sendo um grande desafio. De fato, quando se analisa o nível de inteligência de um inseto qualquer capaz de reconhecer outro de sua espécie e, também, possíveis presas e predadores em seu ambiente, controlar o seu voo, localizar água e alimentos, com todo este processamento sendo realizado por um cérebro extremamente pequeno, se constata que os computadores atuais ainda tem sérias limitações em relação aos sistemas biológicos. Porém, as expectativas da área são muito boas, pois nos últimos anos a IA tem se tornando intensamente popular, e as máquinas com diferentes níveis de inteligência já começaram a ser utilizadas em diversas áreas [Artero 2009].

É possível que no futuro estas máquinas inteligentes estejam presentes em praticamente todas as atividades humanas. Certamente, a maior dificuldade da área consiste em determinar o mecanismo utilizado pelos seres inteligentes, para que então, a partir de um modelo bem definido, seja possível implementá-lo com sucesso em um computador [Artero 2009].

Um dos principais métodos da IA são as Redes Neurais Artificiais (RNAs). As RNAs foram desenvolvidas nos anos 40, pelo neurofisiologista McCulloch e pelo matemático Pitts, que sugeriram usar elementos de *hardware* como resistores variáveis conectados a amplificadores, para se comportarem como os neurônios. Em seguida, esperava-se conectá-los, formando uma rede capaz de emular a estrutura e o funciona-

mento do cérebro humano. As RNAs geraram uma grande expectativa, pois se acreditava que elas seriam capazes de emular exatamente o comportamento do cérebro humano, sendo capazes de raciocinar e resolver qualquer tipo de problema. Porém, em 1969, Minsk e Papert, demonstraram sérias limitações desta teoria e as RNAs ficaram adormecidas até a década de 80, quando voltaram a atrair a atenção da comunidade científica, com o surgimento de novos modelos e, desta vez, com um foco mais preciso sobre o tipo de problema que poderiam resolver [Artero 2009].

A RNA é uma estrutura maciçamente paralela com a habilidade de generalização, ou seja, pode produzir saídas adequadas para entradas que não estavam presentes durante o treinamento. Esta capacidade a diferencia dos demais algoritmos, que apenas processam o que lhes foi ensinado. A solução para um dado problema é obtida, decompondo os problemas complexos em um número de tarefas relativamente simples, atribuindo-se às redes neurais artificiais um subconjunto de tarefas que coincidam com as capacidades inerentes ao seu treinamento específico [LudwigJr and Montgomery 2007].

Um dos benefícios das RNAs, diz respeito ao tratamento de um problema clássico em IA, que é a representação de um universo não-estacionário (onde as estatísticas mudam com o tempo). Uma RNA pode ser projetada para modificar seus pesos sinápticos em tempo real. Também deve-se considerar que uma RNA implementada em *hardware* tem o potencial de ser tolerante a falhas, devido à natureza distribuída com que as informações são armazenadas na rede [LudwigJr and Montgomery 2007].

Uma desvantagem das RNAs é o fato delas, normalmente, serem uma "caixa preta". É impossível saber porque uma rede chegou a um resultado ou, no caso de uma rede de múltiplas camadas, saber qual a relevância de um peso sináptico para um determinado resultado ou qual o significado físico de um peso sináptico. Só é possível saber se a rede funciona corretamente pela análise do erro médio quadrático apresentado ao se introduzir os dados para validação, ou seja, não há como gerar um prova formal para os resultados obtidos [LudwigJr and Montgomery 2007].

Utilizando esta desvantagem como norte, apresentamos neste trabalho uma ferramenta que tem por objetivo auxiliar no ensino das RNAs, mais precisamente do Perceptron. Esta ferramenta é chamada de PercepEduc. A seção 2 traz todos os detalhes deste ambiente, explicando passo a passo as funcionalidades e potencialidades da ferramenta.

2. Ambiente PercepEduc

Com o objetivo de simular o comportamento do neurônio biológico, McCulloch e Pitts propuseram o modelo de neurônio artificial ilustrado na Figura 1. Esta figura apresenta o corpo do neurônio, com suas entradas E_i e sua saída Y . O corpo do neurônio artificial é dividido em duas partes, sendo a primeira responsável por somar (somatório), o produto das entradas E_i pelos seus respectivos pesos w_i . Na segunda parte, tem-se a função de ativação ou transferência f , que irá controlar o valor a ser fornecido pela saída Y .

Além das entradas E_i , também se prevê uma entrada extra, chamada de viés (bias), que sempre tem como entrada o valor 1. A primeira parte do processamento é indicada na Equação 1 e tem como saída a variável net .

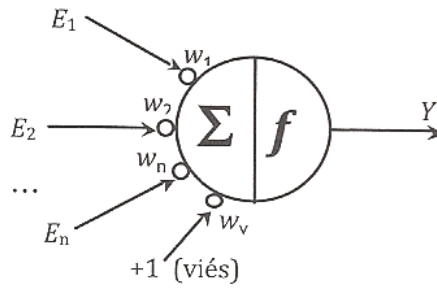


Figure 1. Neurônio artificial [Artero 2009]

$$net = \sum_{i=1}^n E_i * w_i + w_v \quad (1)$$

onde E_i são as entradas da rede, w_i são os pesos associados às entradas E_i e w_v é o peso da entrada de viés (bias). Para a função de transferência ou função de ativação f , existem várias possibilidades, sendo as mais comuns as apresentadas na Figura 2.

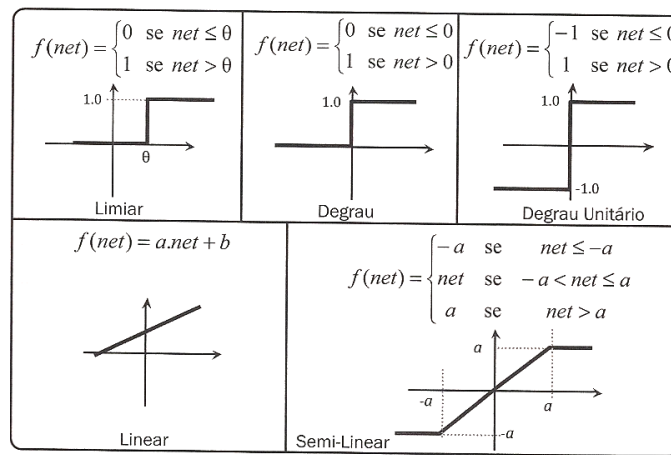


Figure 2. Funções de transferência mais usadas [Artero 2009]

O projeto de RNAs é uma tarefa de tentativa e erro, e quando a rede não apresenta um bom desempenho, é necessário refazer todo o projeto, alterando o número de neurônios, conexões, taxa de aprendizado, funções de progragação, adição de novos parâmetros, dentre outros [Artero 2009].

Na maioria dos problemas de aprendizado, a tarefa é aprender a classificar entradas de acordo com um conjunto finito de classificações. Tipicamente, um sistema de aprendizado é dotado de um conjunto de dados de treinamento, que foram classificados manualmente. O sistema tenta aprender, a partir destes dados de treinamento, a como classificar estes mesmos dados e também como a classificar novos dados ainda não observados [Coppin 2010]. O treinamento das RNAs podem ser feito de forma supervisionada ou não-supervisionada. RNAs de aprendizado supervisionado aprendem ao serem apresentadas a dados de treinamento pré-classificados, utilizando as diferenças entre os valores desejados e os valores obtidos pela RNA para alterar os valores existentes nas

entradas dos neurônios, e assim, a RNA aprenderá o padrão apresentado [Artero 2009]. Desta forma, as RNAs aprendem ao modificarem os pesos das conexões de suas redes, para classificar mais precisamente os dados de treinamento. Deste modo, as RNAs são capazes de generalizar, com grande grau de precisão, a partir de um conjunto de treinamento, chegando ao conjunto completo de entradas possíveis [Coppin 2010].

O ambiente PercepEduc foi construído utilizando um dos tipos de RNA mais simples, o Perceptron. O Perceptron foi criado por Frank Rosenblat em 1957, e estimulou bastante o interesse pelas RNAs [Artero 2009]. Ela possui uma arquitetura bastante simples, apresentando apenas um neurônio [LudwigJr and Montgomery 2007]. Embora não tenha grande potencial, sendo aplicada apenas em problemas linearmente separáveis, o seu estudo se justifica pela sua simplicidade e história [Artero 2009]. Devido a estas características supracitadas, escolhemos o Perceptron no desenvolvimento deste ambiente.

A Figura 3 ilustra a tela principal do PercepEduc. As principais partes do PercepEduc está numerada de 1 a 6, cada uma destacada em uma cor diferente.

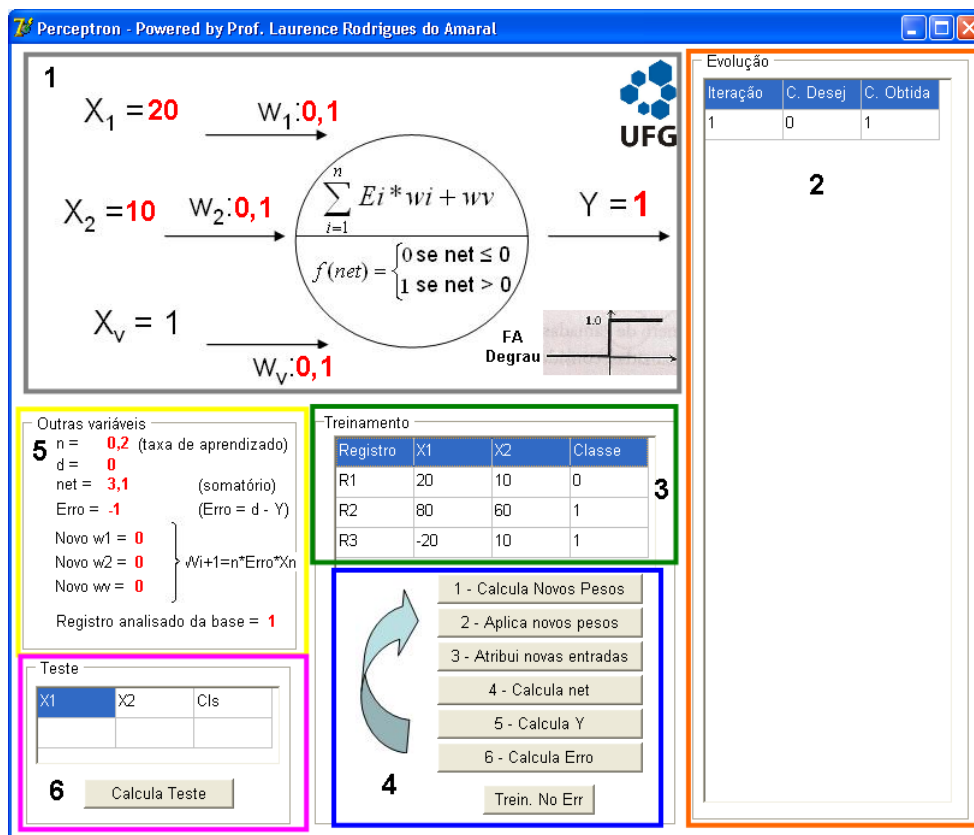


Figure 3. PercepEduc: Tela principal

A parte de número 1 (destacado em cinza), traz as três entradas do perceptron (X_1 , X_2 e X_v), tendo cada uma um peso (w_1 , w_2 e w_v) associado. Além destas informações, ele também traz informações da função de ativação utilizada (função degrau) e a saída do perceptron. Neste ambiente, as amostras podem ser de duas classes distintas, 0 ou 1. Desta forma, a saída Y do perceptron trará como retorno, 0 ou 1, isto é, a amostra analisada em treinamento será atribuído pelo perceptron à classe 0 ou à classe 1.

A parte de número 2 (destacado em laranja) mostra passo a passo o resultado de

cada iteração do perceptron. Neste *grid*, é destacado a classe desejada, isto é, a classe a qual a amostra de treinamento analisada realmente pertence. A terceira coluna representa a classe obtida, isto é, a classe que o perceptron disse que a amostra pertence.

A parte de número 3 (destacado em verde) traz as informações sobre as amostras usadas para o treinamento do perceptron. Como este ambiente tem fins educacionais, utilizamos três registros fixos, sendo o primeiro (R_1) pertencente à classe 0 e os demais (R_2 e R_3) pertencentes à classe 1. Os atributos X_1 e X_2 referem-se às coordenadas x e y em um plano cartesiano. Por exemplo, o registro R_1 refere-se ao par ordenado (20,10) num plano cartesiano.

A parte de número 4 (destacado em azul) são todos os 6 passos que devemos seguir para treinar o perceptron. No primeiro passo são calculados os novos pesos, no segundo aplicamos os pesos calculados no passo 1. O passo 3 aplica às entradas X_1 e X_2 os valores do próximo registro a ser analisado na base de treinamento. De posse das entradas (X_1 e X_2) e dos pesos (w_1 , w_2 e w_v) fazemos o somatório ponderado e obtemos o valor *net*. De posse deste valor, aplicamos o mesmo na função de ativação, onde recebemos como resultado (Y), 0 ou 1. Estes passos deverão ser repetidos até que não seja encontrado nenhum erro para os três registros em treinamento. Ao pressionar o botão "Trein. No Err", ocorrerá o treinamento do perceptron até que não seja encontrado erros de classificação.

A parte de número 5 (destacado em amarelo) traz informações sobre todas as variáveis utilizadas no treinamento, tais como: taxa de aprendizado, valor obtido pela variável *net*, erro (saída obtida - saída desejada), novos valores para os pesos e qual registro da base de treinamento estamos analisando.

Quando estamos construindo um classificador, tão importante quanto o treinamento do mesmo é o desempenho deste em teste. A parte de número 6 (destacado em rosa) permite ao usuário fornecer coordenadas X_1 e X_2 para que o perceptron (utilizando os pesos que foram obtidos em treinamento) possa classificar novas amostras, amostras estas nunca apresentadas ao classificador.

3. Conclusões e Trabalhos Futuros

O ambiente PercepEduc foi utilizado em duas turmas da disciplina de Inteligência Artificial da Universidade Federal de Goiás, Campus Jataí. Nestas duas oportunidades pude perceber que a utilização do PercepEduc aumentou drasticamente o entendimento e interesse da turma com relação às RNAs. Os conceitos de aprendizado supervisionado foram fortalecidos, além do perfeito entendimento da distinção entre registros de treinamento e teste. Vários testes foram feitos (utilizando a região demarcada pelo código 6 e cor rosa), na busca do limiar entre as duas classes. Concluindo, o uso do PercepEduc enriqueceu bastante as aulas da disciplina Inteligência Artificial, tornando mais claro o funcionamento deste tipo de RNA.

Como trabalhos futuros, iremos adicionar uma funcionalidade para podermos selecionar uma função de ativação dentre as funções: limiar, degrau, degrau unitário, linear e semi-linear, descritas na Figura 2. Com esta alteração, conseguiremos comparar cada uma das funções de ativação, podendo estabelecer critérios mais claros e objetivos para a melhor escolha desta função. Além disso, iremos permitir ao usuário inserir mais registros

de treinamento, para que o mesmo possa melhor entender a influência que os registros de treinamento exercem sobre o perceptron. Por fim, iremos reescrever todo o código existente em uma linguagem voltada para web (atualmente ele funciona localmente). Desta forma, o PercepEduc poderá ser usado por toda a comunidade científica, auxiliando no aprendizado das Redes Neurais Artificiais, de forma mais simples e funcional.

References

- Artero, A. O. (2009). *Inteligência Artificial Teórica e Prática*. Livraria da Física, São Paulo.
- Coppin, B. (2010). *Inteligência Artificial*. LTC, Rio de Janeiro.
- LudwigJr, O. and Montgomery, E. (2007). *Redes Neurais: Fundamentos e Aplicações com Programas em C*. Ciência Moderna, Rio de Janeiro.
- NascimentoJr, C. L. and Yomeyama, T. (2004). *Inteligência Artificial em Controle e Automação*. Blucher Fapesp, São Paulo.
- Russel, S. and Norvig, P. (2004). *Inteligência Artificial*. Campus, Rio de Janeiro.