

## ESTUDO COMPARATIVO DO USO DE REDES NEURAIS ARTIFICIAIS E REGRESSÃO LINEAR MÚLTIPLA PARA A PREVISÃO DA CONCENTRAÇÃO CÁUSTICA EM UMA ETAPA DO PROCESSO DE FABRICAÇÃO DE ALUMINA

### A COMPARATIVE STUDY BETWEEN THE USE OF ARTIFICIAL NEURAL NETWORKS AND MULTIPLE LINEAR REGRESSION FOR CAUSTIC CONCENTRATION PREDICTION IN A STAGE OF ALUMINA PRODUCTION

Giovanni Leopoldo Rozza\* E-mail: [giovanni.rozza@ufpr.br](mailto:giovanni.rozza@ufpr.br)  
Ruy Gomes da Silva\*\* E-mail: [ruyrqs@gmail.com](mailto:ruyrqs@gmail.com)  
Sonia Isoldi Marty Gama Müller\* E-mail: [soniaisoldi@ufpr.br](mailto:soniaisoldi@ufpr.br)  
\*Universidade Federal do Paraná (UFPR), Curitiba, PR  
\*\*Pontifícia Universidade Católica (PUC/PR), Curitiba, PR

**Resumo:** Para manterem-se competitivas as empresas otimizam seus processos continuamente, de maneira sustentável e reaproveitando os recursos naturais. As ferramentas de apoio a tomada de decisão são de extrema importância, principalmente quando tais ferramentas auxiliam na antecipação dos problemas operacionais, evitando custos, perdas de produtividade, acidentes de trabalho e ambientais. Este estudo tem foco no processo produtivo de alumina pelo método Bayer, na mensuração do teor cáustico da mistura de bauxita, que influencia diretamente na qualidade. A empresa obtém este resultado uma vez ao dia, por meio de análise laboratorial, devido aos elevados custos. A medida não é suficiente para tomada de decisão e antecipação das correções de problemas que possam evitar a ineficiência do processo, observado apenas no dia seguinte, quando ocorrer o recebimento do novo resultado. Propõe-se a previsão da concentração cáustica em tempo real, através da modelagem do processo por Regressão Linear Múltipla e Rede Neural Artificial. Tais modelos foram gerados através dos softwares SPSS e MATLAB. Os resultados foram comparados com base no erro entre a previsão da concentração do produto gerada pelos modelos, e a concentração real de saída obtidas de análise laboratorial. Concluiu-se que a técnica de Redes Neurais Artificiais desempenhou melhor que a Regressão Linear Múltipla. Os resultados mostraram a viabilidade do processo de análise imediata por meio da previsão, auxiliando assim a decisão da equipe de trabalho envolvida.

**Palavras-chave:** Redes Neurais Artificiais. Processo Bayer. Regressão Linear Múltipla. Alumina. Evaporação.

**Abstract:** With world becoming each day a global village, enterprises continuously seek to optimize their internal processes to hold or improve their competitiveness and make better use of natural resources. In this context, decision support tools are an underlying requirement. Such tools are helpful on predicting operational issues, avoiding cost risings, loss of productivity, work-related accident leaves or environmental disasters. This paper has its focus on the prediction of spent liquor caustic concentration of Bayer process for alumina production. Caustic concentration measuring is essential to keep it at expected levels, otherwise quality issues might arise. The organization requests caustic concentration by chemical analysis laboratory once a day, such information is not enough to issue preventive actions to handle process inefficiencies that will be known only after new measurement on the next day. Thereby, this paper proposes using Multiple Linear Regression and Artificial Neural Networks techniques a mathematical model to predict the spent liquor's caustic concentration. Hence preventive actions will occur in real time. Such models were built using software tool for numerical computation (MATLAB) and a statistical analysis

software package (SPSS). The models output (predicted caustic concentration) were compared with the real lab data. We found evidence suggesting superior results with use of Artificial Neural Networks over Multiple Linear Regression model. The results demonstrate that replacing laboratorial analysis by the forecasting model to support technical staff on decision making could be feasible.

**Keywords:** Artificial Neural Networks. Multiple Linear Regression. Evaporation. Bayer Process. Alumina.

## 1 INTRODUÇÃO

O conhecimento de processos produtivos é fundamental para a melhoria das etapas operacionais de um determinado setor de produção. Apesar das decisões serem tomadas a partir do nível Estratégico, encaminhadas ao nível Tático e finalmente ao nível Operacional, muitos ambientes de trabalho levam em consideração as informações que partem do nível Tático e Operacional, integrando estes três setores (NANCI et al., 2008).

Slack e Lewis (2009) definem o nível Operacional como a base do sistema que atua diretamente no sistema produtivo, recebendo as informações oriundas do nível Tático. Neste papel de atuação na linha de frente, torna-se claro a necessidade de informações que auxiliem as equipes de operações e controle, quanto a tomada de decisão imediata sem a necessidade de recorrer-se aos demais setores. Destacam ainda que o emprego de tecnologias nos processos é fundamental, uma vez que permitem o aperfeiçoamento da produção e da qualidade do produto final.

A produção industrial de alumina utilizada como insumo para indústria do alumínio dá-se através do purificação da bauxita pelo processo Bayer, que utiliza soda cáustica (NaOH) como elemento dissolvente. Dessa forma, a pureza da alumina e portanto sua qualidade, está intrinsecamente associada a correta concentração da soda cáustica necessária para a solubilidade dos diferentes constituintes que compõem a bauxita (CONSTANTINO et al., 2002).

O monitoramento da concentração cáustica se faz através da leitura e interpretação de diversos parâmetros físicos, é item necessário para o controle da relação de produtividade de uma planta industrial de beneficiamento de bauxita para a produção de alumina pelo processo Bayer. Atualmente o valor da concentração cáustica é obtida através de serviços de laboratório, um processo caro e de longa duração.

Este trabalho propõe a previsão por meio da modelagem matemática baseada no conhecimento de Redes Neurais Artificiais (RNA) e Regressão Linear Múltipla da concentração cáustica no produto, utilizando variáveis de controle da etapa de evaporação do processo Bayer. O desempenho dos modelos de previsão são mensurados através do seu erro médio quadrático de forma a se determinar qual modelo apresenta melhor acurácia na previsão da concentração de soda cáustica.

## 2 A DESCRIÇÃO DO PROCESSO PRODUTIVO

O beneficiamento da bauxita para a produção de alumina foi desenvolvido em 1858 por Le Chatelier e aperfeiçoado posteriormente em 1888 por Karl Bayer, sendo conhecido atualmente como processo Bayer (SILVA FILHO; ALVES; MOTA, 2007; HIND; BHARGAVA; GROCCOTT, 1999). Basicamente a produção de alumina, ou óxido de alumínio ( $Al_2O_3$ ), tem seu início com a extração da bauxita e posterior moagem, seguido pela digestão onde é adicionado uma solução cáustica de hidróxido de sódio (NaOH) sob temperatura variando de 145 à 170 °C e pressão aproximada de 20 kgf, sendo que tais parâmetros dependem do tipo de bauxita.

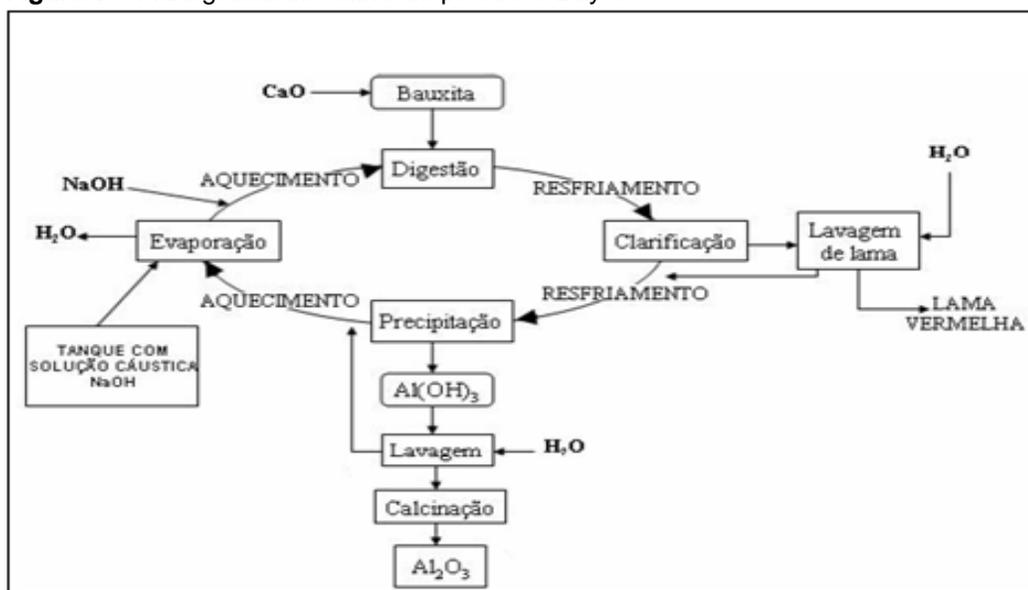
Segundo o *International Aluminium Institute* IAI (2013) aproximadamente 40% da produção de bauxita produzida no mundo provêm da Austrália, destacando-se, ainda como principais produtores: Guiana (14%), Jamaica (11%), Brasil (8%), Índia (5%) China (3%). As reservas mundiais de bauxitas estão distribuídas nas regiões tropicais (57%), mediterrânea (33%) e subtropicais (10%). O Processo *Bayer* para a produção de alumina está dividido em etapas sendo válido para qualquer refinaria que utilize esse processo. Cerca de duas a três toneladas de alumina são necessárias para se produzir uma tonelada de alumínio. A Figura 1 representa o fluxograma das etapas de todo o processo.

A alumina posteriormente é transformada em alumínio através do processo de Hall-Héroult, que se resume na passagem de uma corrente elétrica na mistura da alumina com o minério creolite ( $Na_3AlF_6$ ), utilizado para reduzir a sua temperatura de fusão, causando a redução da alumina. Dessa forma obtém-se o alumínio metálico através de decantação (FURTADO; URIAS, 2013).

O processo que é objeto deste estudo está no sistema de evaporação, que faz parte de uma das etapas do processo Bayer. O sistema de evaporação tem como propósito elevar a concentração de NaOH medida em gramas por litro (g/l), através da retirada de água em forma de vapor por intermédio de tanques de expansão de vapor denominados *flash tank*.

Um sistema de evaporação é composto por vasos pressurizados que fazem a expansão dos líquidos aquecidos por troca térmica a partir do consumo do vapor de caldeira, de forma a remover a água do processo.

**Figura 1** – Fluxograma descritivo do processo Bayer



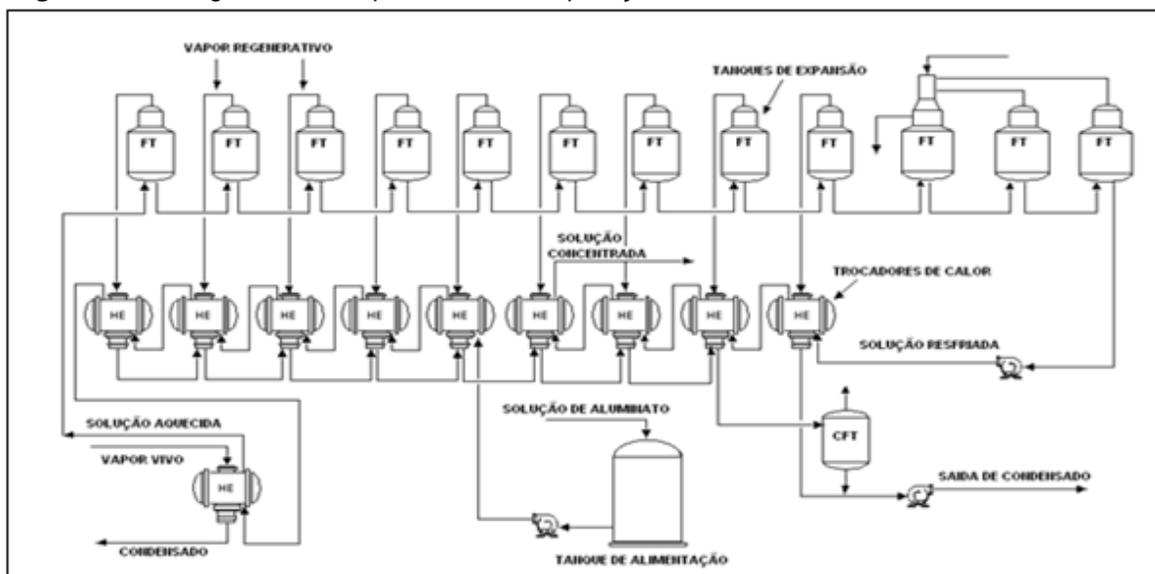
Fonte: IAI (2013)

O *flash tank* é um equipamento de porte industrial onde ocorre o fenômeno da expansão, ou seja, a passagem da molécula de água do estado líquido para o gasoso na forma de vapor d'água. Esta mudança de estado se dá em função da diferença de pressão a que a solução é submetida quando sai das tubulações e entra nos tanques, que possuem uma grande área útil.

Em síntese, o fluido é bombeado para o sistema de evaporação com concentração de soda reduzida, devido à presença excessiva de água. Tal fluido está a uma temperatura de aproximadamente 75° C, em seguida é aquecido com vapor ao passar pelos trocadores de calor. Neste ponto a solução atinge o estado de ebulição e

segue para os tanques evaporadores onde ocorre a expansão do fluido, desprendendo vapor de água ocasionando a melhora na concentração cáustica no produto de saída do sistema. Adicionalmente seu volume diminui, o que contribui para o controle de volume total da planta, onde se incluem as demais etapas. O sistema de evaporação está representado na Figura 2 a seguir:

**Figura 2** – Fluxograma de um processo de evaporação



Fonte: Adaptado de IAI (2013)

## 2.1 A Mensuração da Concentração de Soda Cáustica no Produto da Evaporação

Um dos procedimentos padronizados deste processo consiste em mensurar a concentração cáustica uma vez ao dia do produto da evaporação, por intermédio de coleta e análise do fluido. Este procedimento possui custos adicionais com serviços de laboratório e proporcionam riscos aos funcionários que efetuam a tarefa junto aos equipamentos em operação. Assim, obtém-se uma medição da concentração cáustica que indica se haverá necessidade ou não, de se executar medidas operacionais para elevar a eficiência do sistema evaporador.

Após o resultado do laboratório fornecido para a equipe de operação do processo, medidas podem ser tomadas para manter a concentração cáustica dentro do exigido pelo etapa seguinte, como por exemplo, adicionando mais soda cáustica a

mistura se as leituras apresentarem um resultado menor que o valor mínimo necessário.

Percebe-se portanto que há uma limitação considerável na capacidade de avaliação das condições ideais de operação do processo. As medidas preventivas ou corretivas necessárias para promover a estabilidade do processo frente a possíveis desequilíbrios na mistura sofrem um retardo de ação considerável, o que pode acarretar perda da qualidade da alumina produzida.

## **2.2 Alternativa para Previsão Através da Análise Laboratorial**

A proposta deste trabalho é desenvolver um modelo de previsão baseada nas técnicas de rede neural artificial (RNA) e Regressão Linear Múltipla, que tratarão as variáveis que afetam o processo de evaporação. Adicionalmente se investigará qual metodologia apresenta o menor erro comparado com os dados reais obtidos do processo. Os modelos desenvolvidos poderão prever o valor da concentração cáustica no produto de saída da etapa de evaporação, apresentando-se dessa forma como uma alternativa à monitoração deste parâmetro por meio de análise laboratorial.

A principal vantagem apresentada pelos métodos de previsão é a medição em tempo real, eliminando o retardo de ações preventivas ou corretivas, contribuindo para a melhoria da qualidade da alumina produzida. A substituição do custoso exame laboratorial pelo método de previsão também acarretará vantagem econômica.

## **3 REFERENCIAL TEÓRICO**

### **3.1 A Rede Neural Artificial *Feedforward Multilayer Perceptron***

Redes Neurais artificiais (RNA) são modelos matemáticos de processamento de informações que possuem característica de desempenho semelhantes as redes neurais biológicas. Tal qual o nosso cérebro, a informação é processada em elementos básicos denominados neurônios, que se comunicam através de conexões. A estas conexões estão associadas pesos que multiplicam o sinal transmitido e são somados linearmente

fornecendo assim a informação de entrada para a função de ativação, esta por sua vez irá determinar o sinal de saída (FAUSSET, 1994).

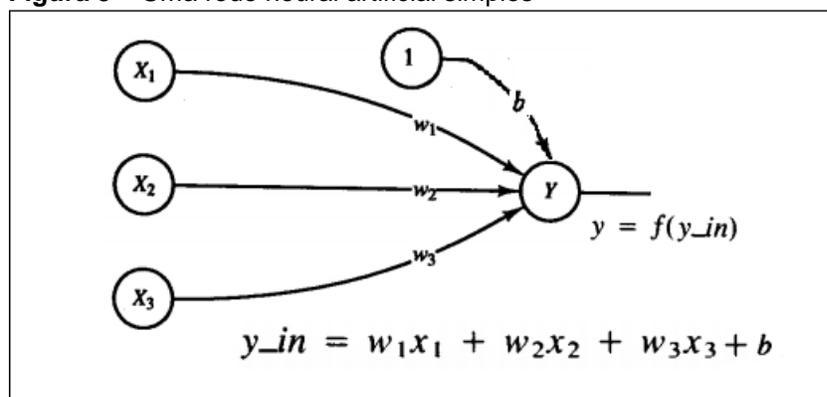
A Figura 3 abaixo representa um modelo simples de rede neural artificial com 3 entradas, uma entrada de valor constante (*bias*) e a função de ativação  $f(y_{in})$  que determina o valor da resposta da saída  $y$ .

Para se iniciar o processo de treinamento ou aprendizagem de uma RNA de forma a ser capaz de identificar padrões, aplica-se a combinação desejada de valores em suas entradas, que serão ponderadas e combinadas linearmente. A RNA soma à esta combinação um valor constante, portanto independente do valor de suas entradas.

Esta soma linear de combinação desejada das entradas e um valor constante ponderadas por pesos ( $y_{in}$ ) torna-se o sinal de entrada da função de ativação  $f(y_{in})$ , o resultado então é utilizado para obter-se o valor correspondente de resposta para as entradas.

Comparada com a resposta desejada, se a diferença estiver acima de um determinado limiar, os pesos da rede neural são atualizados e o processo de treinamento reinicia com subsequente combinação de entradas.

**Figura 3** – Uma rede neural artificial simples



Fonte: Adaptado de FAUSSET (1994)

O processo de treinamento é finalizado quando a diferença entre a resposta desejada e a saída da RNA retorna um valor menor que o limiar configurado (SIVANANDAM; DEEPA, 2006).

A RNA de arquitetura mais simples possível, capaz de identificar padrões simples, como as compostas por duas classes linearmente separáveis, é a rede neural do tipo *Perceptron*. Apresentado por Rosenblatt (1958), a RNA do tipo *Perceptron* consiste de apenas um único neurônio que apresenta entradas e uma constante de valor unitário ponderados por pesos ajustáveis em função do erro da resposta. Sua função de ativação  $f(y_{in})$  é uma função descontínua do tipo degrau de intervalo  $[0,+1]$  ou degrau simétrico de intervalo  $[-1,+1]$  (DEMUTH; BEALE; HAGAN, 2009).

De arquitetura mais complexa que o *Perceptron*, a rede do tipo *feedforward multilayer perceptron* (FFMLP) desenvolvido por Rumelhart *et al* (1985) é um dos modelos existentes de rede neural artificial mais versáteis e populares, com aplicações em campos tão variados quanto a engenharia industrial, finanças, reconhecimento de voz e de imagem (REED; MARKS, 1998; PANDYA; MACY, 1995).

A rede FFMLP é formada por  $k$  camadas (*layers*) de neurônios do tipo *Perceptron*, a primeira camada (*input layer*) representa os  $n$  sinais de entrada  $x$ , a  $k$ -ésima camada representa a camada de saída  $y$  (*output layer*) que contém a resposta ao conjunto dado de entradas. A quantidade de neurônios na camada de saída pode ser unitária ou apresentar múltiplas saídas, neste caso associa-se um neurônio a cada saída.

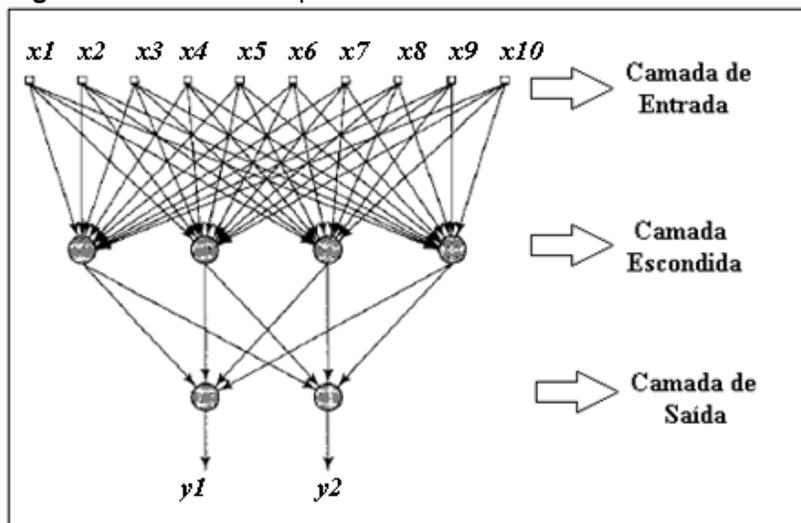
As camadas intermediárias são denominadas de camadas escondidas (*hidden layers*) e sua principal característica é o fato de que seus neurônios possuem funções de ativação não lineares. Porém ao contrário do neurônio *Perceptron*, estas funções não possuem descontinuidades e portanto são deriváveis em qualquer ponto. Dessa forma possibilita-se que a rede consiga tratar problemas cuja solução não é trivial como a classificação de padrões não lineares, aproximação de funções polinomiais complexas e problemas com múltiplas variáveis (ZHANG, 2000).

Apesar de ser possível arbitrar-se um número qualquer de camadas escondidas Hecht- Nielsen (1987) afirmou, baseado no teorema da Superposição de Kolmogorov (1957), que uma rede contendo somente uma única camada escondida com um número suficiente de neurônios pode modelar qualquer função continua univariada. O Teorema da Superposição, por consequência, possibilita estender essa propriedade

para funções multivariadas contínuas (CHARPENTIER; LESNE; KAPITONOVICH, 2007 ;GIROSI; FAUSETT, 1994; POGGIO, 1989).

A Figura 4 a seguir apresenta um exemplo de configuração de rede FFMLP com 10 nós na camada de entrada (entradas  $x1...x10$ ), cada uma delas aplicada aos 3 neurônios, por sua vez cada uma das 3 saídas desses neurônios é utilizado como entrada para os 2 neurônios da camada de saída (saídas  $y1$  e  $y2$ ).

**Figura 4** – RNA FFMLP para um sistema com 10 entradas e 2 saídas



Fonte: Adaptado de HAYKIN (1999)

Para o treinamento das redes FFMLP utiliza-se uma generalização do algoritmo de filtro adaptativo Mínimo Quadrado Médio, do inglês *Least Mean Square (LMS)*, denominado *error back-propagation* Haykin (1999). Neste algoritmo o sinal de entrada propaga-se pela rede de uma camada para a próxima (propagação *feedforward*) de forma que ao atingir a camada de saída o resultado calculado é então subtraído do resultado esperado. Esta diferença, que representa o sinal de erro, por sua vez propaga-se no sentido reverso (propagação *backward*), sendo utilizado para se ajustar os pesos dos neurônios de todas as camadas da rede. O somatório do quadrado do sinal erro para todo o conjunto de entradas pode ser expresso como (YONABA; ANCTIL; FORTIN, 2010; SCHROEDER, 2004):

$$SSE = \sum_{i=1}^n (q_i - \hat{q}_i)^2 \quad (1)$$

$Q_i$  é a saída desejada para dado vetor de entradas  $i$ ,  $n$  é o total de vetores de entradas e  $\hat{Q}_i$  é representado por:

$$\hat{Q}_i = \Phi_2[\sum w \Phi_1(\sum wx + b) + b] \quad (2)$$

$\Phi_1$  e  $\Phi_2$  representam as funções de ativação utilizadas nas camadas escondidas e de saída,  $w$  e  $b$  representam os pesos das entradas e a da constante (*bias*) associados a cada neurônio da rede.

Dessa forma o algoritmo procura ao longo de várias iterações no sentido *backward*, ajustar os pesos na direção e magnitude do gradiente descendente dos pesos  $w$  minimizando o custo representado pela soma dos quadrados do sinal erro. Essa minimização é calculada através das derivadas parciais de 1ª ordem do sinal do erro em relação aos pesos (STEINER, 2012).

Apesar do algoritmo de *back-propagation* tratar-se de um método computacionalmente eficiente para treinar uma rede neural Haykin (1999), a sua convergência na direção do resultado é lenta, além de haver o risco da solução apresentar um mínimo local (YONABA; ANCTIL; FORTIN, 2010; JEON, 2007).

O algoritmo de *Levenberg-Marquardt* (LM) apresentado por Hagan e Menhaj (1994) representa um aperfeiçoamento do algoritmo *back-propagation* e é reconhecido como um dos métodos mais velozes para treinamento de redes FFMLP.

Para sua utilização entretanto, é necessário que atendam-se certas condições, entre as quais conter apenas uma dimensão para a saída, também é preciso que a função erro seja quadrática e que o tamanho da rede não ultrapasse poucas centenas de neurônios. Esta última restrição se dá porque a ordem da matriz de derivadas parciais de 1ª ordem, denominada de matriz Jacobiana, é numericamente igual a soma de todos os pesos da rede FFMLP. Assim, a medida que a rede aumenta de complexidade, o uso de recurso computacional torna-se cada vez mais intensivo (YONABA; ANCTIL; FORTIN, 2010; HIL; LEWICKI, 2006; HAGAN; DEMUTH; BEALE, 1996).

O algoritmo de LM é uma modificação do algoritmo *quasi-Newton*, ambos necessitam computar os valores da matriz de derivadas parciais de segunda ordem,

denominada de matriz Hessiana. Os algoritmos realizam tais cálculos por meio de operações com a matriz Jacobiana, de complexidade computacional menor, obtendo-se uma aproximação satisfatória da matriz Hessiana. O algoritmo *quasi-Newton* apresenta limitações quando o sinal de erro não é pequeno, convergindo muito lentamente ou não convergindo, o que é corrigido pelo algoritmo LM pela introdução de um valor escalar. Os pesos são atualizados de acordo com a seguinte expressão para o algoritmo *quasi-Newton* e LM respectivamente (DEMUTH; BEALE; HAGAN, 2009; JEON, 2007):

$$x_{k+1} = x_k - [J^T J]^{-1} J^T e \quad (3)$$

$$x_{k+1} = x_k - [J^T J + \mu I]^{-1} J^T e \quad (4)$$

Onde tem-se  $J$  como a matriz Jacobiana,  $J^T J$  como uma aproximação da matriz Hessiana,  $I$  é a matriz identidade,  $e$  o sinal de erro e  $J^T e$  sendo o gradiente do sinal erro. Pode-se perceber que quando o valor de  $\mu$  aproxima-se de zero o algoritmo LM se comporta como *quasi-Newton* e quando  $\mu$  aumenta, seu comportamento se aproxima do algoritmo clássico de *back-propagation* pois neste caso temos que  $[J^T J + \mu I] \approx \mu I$  (JEON, 2007).

O algoritmo LM opera sobre o valor de  $\mu$ , diminuindo seu valor ao se aproximar da convergência, uma vez que o método *quasi-Newton* é eficaz e rápido quando o sinal erro torna-se pequeno. Caso o sinal erro cresça o valor de  $\mu$  é aumentado, dessa forma altera-se o comportamento do algoritmo para atuar como *back-propagation* com um valor pequeno de taxa de aprendizagem (DEMUTH; BEALE; HAGAN, 2009).

A determinação dos valores iniciais dos pesos de uma rede FFMLP também tem influência na velocidade para convergência da solução, uma vez que a superfície representada pelo erro é suficientemente complicada para apresentar mínimos locais.

Nguyen e Widrow desenvolveram na década de 90 um algoritmo que consiste em distribuir a região de atuação de cada neurônio de maneira uniforme na região do espaço definido pelas entradas. Assim aumenta-se o seu espalhamento sobre a região

de atuação das funções de ativação e por consequência a velocidade de convergência (BLACKWELL; CHEN, 2009; DUBROVIN; SUBBOTIN, 2002).

### 3.2 Regressão Linear Múltipla

A regressão linear múltipla permite estudar os múltiplos relacionamentos entre a variável dependente (resposta de saída) e as variáveis independentes (entradas) através da combinação linear de coeficientes de regressão, dado pela seguinte expressão (MONTGOMERY; RUNGER, 2009):

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \dots + \beta_k X_k + \epsilon \quad (5)$$

Onde  $X_1, X_2 \dots X_k$  são as  $k$  variáveis de entrada,  $\beta_0, \beta_1, \beta_2 \dots \beta_k$  são os coeficientes de regressão e  $\epsilon$  é a componente aleatória, denominada de resíduo, que não pode ser estimada. Para que o modelo seja válido a distribuição estatística do resíduo deve obedecer uma distribuição gaussiana de média zero e variância  $\sigma$ , além disso os resíduos devem possuir covariância zero, o que significa que devem ser independentes entre si.

O cálculo dos coeficientes de regressão não é trivial, envolvendo operações com matrizes de magnitude do número de amostras, e na prática utiliza-se auxílio computacional através do uso de aplicativos especializados em cálculos estatísticos como por exemplo o SPSS, MINITAB ou STATGRAPHICS.

Igualmente importante ao se desenvolver um modelo de previsão baseado na regressão linear é verificar se no mínimo uma das variáveis de entrada contribui significativamente para o valor da resposta, ou se a componente aleatória tem composição predominante tal que a variação das entradas tem pouco ou nenhum impacto no valor de saída. O teste de significância é dado pelas seguintes hipóteses (MONTGOMERY; RUNGER, 2009):

$$H_0: \beta_1 = \beta_2 = \beta_k = 0 \quad (6)$$

$$H_1: \beta_j \neq 0 \text{ para no mínimo um } j \quad (7)$$

Para se determinar qual hipótese é válida, considera-se  $H_0$  verdadeira e calcula-se a sua probabilidade, denominada de *p-value*. Este valor é comparado com valores críticos que são determinados pelo nível de significância adotado. O nível de significância é definido como a probabilidade de rejeitarmos incorretamente a hipótese  $H_0$ . Valores de *p-value* menores que o nível de significância determinam a rejeição da hipótese  $H_0$ , indicando que ao menos uma das variáveis dependentes contribui para a variabilidade da variável resposta.

Outro parâmetro importante utilizado na análise de regressão linear múltipla é o coeficiente de determinação  $R^2$  e mede a proporção da variabilidade da resposta que pode ser explicada pelo modelo de regressão, o valor de  $R^2$  se encontra entre [0,1]. Quanto mais próximo do valor unitário maior a porcentagem da variabilidade da resposta explicada pelo modelo de regressão e menor a influência da componente aleatória e vice-versa (WALPOLE et al., 2012).

### 3.3 Medidas de Erro

Medidas de erro podem ser utilizadas para se avaliar o ajuste (*goodnes-of-fit*) de determinado modelo matemático, entre eles estão o RMSE, RRMSE e o MRE (DAAMEN; BUISSON; HOOGENDOORN, 2014).

A Raiz Quadrada do Valor Médio Quadrático do inglês *Root Mean Squared Error* (RMSE) mede o grau de dispersão em torno da linha de regressão e é dado pela fórmula (ZIELESNY, 2011; BARRETO; HOWLAND, 2006):

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\sigma_i - t_i)^2}{n}} \quad (8)$$

Onde  $n$  é o número de valores de entrada,  $\sigma_i$  é  $i$ -ésima saída da rede FFMLP e  $t_i$  é o  $i$ -ésimo valor desejado. Note que esse parâmetro não leva em consideração a magnitude da grandeza, dessa forma por exemplo um MSE de 5 é um valor muito

pequeno se a saída é da ordem de milhares de unidades, porém é alto se a saída estiver ao redor das dezenas de unidades.

De forma a relativizar a medida de desempenho, utilizou-se também o Erro Relativo do Valor Médio Quadrático do inglês *Relative Root Mean Square Error* (RRMSE) dado pela fórmula (BARCELÓ, 2010):

$$RRMSE = \sqrt{\frac{\sum_{i=1}^n \left[ \frac{(o_i - t_i)^2}{t_i} \right]}{n}} \quad (9)$$

O valor médio relativo do erro de saída apresentado pela rede neural artificial é um parâmetro de desempenho igualmente importante, para isto calculou-se o Erro Médio Relativo MRE do inglês *Mean Relative Error* (MRE) dado pela seguinte expressão (MASTERS, 1993):

$$MRE = \frac{\sum_{i=1}^n \left| \frac{t_i - o_i}{t_i} \right|}{n} \quad (10)$$

#### 4 METODOLOGIA

Esse estudo foi realizado em uma refinaria de minério de bauxita, localizada no estado do Pará. A base de dados foi obtida por meio dos resultados das análises químicas referentes à concentração cáustica contida no produto de saída do sistema de evaporação, que faz parte do processo Bayer de produção de alumina. Essa medição tem como objetivo elevar a concentração de soda cáustica, medida na relação de controle dessa solução, pela redução da quantidade de água presente.

As variáveis de entrada foram selecionadas com auxílio dos especialistas do processo e tiveram tratamento estatístico para eliminação dos valores denominados *outliers*. Tais variáveis de entrada foram caracterizadas da seguinte forma: Cáustico de entrada (X1), Vazão de alimentação (X2), Retirada de condensado (X3) e a temperatura do vapor vivo (X4). Todas as variáveis combinadas na entrada do processo são usadas para controle e ajustes necessários ao parâmetro desejado (concentração cáustica) na saída da evaporação, definido como Y. O banco de dados utilizado neste trabalho contém um total de 269 amostras.

Utilizou-se o aplicativo MATLAB versão 7 para modelar a rede FFMLP, configurada com uma camada de entrada de 4 nós representando o conjunto de entradas  $X_1...X_4$ , e uma camada de saída com um neurônio (saída  $Y$ ). O modelo inicial da rede FFMLP continha apenas um neurônio escondido, aumentou-se seu número ao longo da pesquisa com o objetivo de se determinar qual arquitetura apresenta a melhor eficiência.

A função hiperbólica tangente (parâmetro MATLAB *tansig*)  $f(x) = \frac{2}{1+e^{-2x}} - 1$  foi selecionada como função de ativação da camada escondida. Funções de ativação sigmoidais antissimétricas como a hiperbólica tangente são mais eficientes do que as não simétricas, pois aumentam a velocidade de aprendizagem e assim diminuem o tempo para convergência da solução da rede FFMLP (HAYKIN, 1999).

Para a camada de saída utilizou-se a função linear (parâmetro MATLAB *purelin*)  $f(x) = x$  pois revela-se mais adequada no uso de redes neurais como aproximadores de função, não havendo melhora de desempenho com a utilização de uma função não linear (YONABA; ANCTIL; FORTIN, 2010; DEMUTH; BEALE, HAGAN, 2009).

Selecionou-se o algoritmo de *Levenberg-Marquardt* (parâmetro MATLAB *trainlm*) para treinar a rede, associado com o algoritmo de inicialização de pesos *Nguyen-Widrow* (parâmetro MATLAB *initnw*) para aumentar a eficiência do treinamento. Por utilizar-se o algoritmo LM os pesos das entradas e do *bias* da rede FFMLP deverão ser modificados em lote (parâmetro MATLAB *batch mode*), significando que somente após todos os dados de entrada do conjunto de treinamento serem apresentados à rede os seus pesos serão atualizados (DEMUTH; BEALE; HAGAN, 2009).

A partir de um tamanho mínimo, várias configurações foram avaliadas com objetivo de se conseguir a arquitetura mais simples de rede FFMLP que generalize adequadamente o espaço de soluções definido para o conjunto de entradas, sem incorrer em *overfitting*, o que significaria um aumento desnecessário da complexidade da rede sem um ganho significativa de eficiência (DEMUTH; BEALE, 1996; MASTERS, 1993; HAGAN).

Não houve necessidade de se normalizar os dados de entrada uma vez que as funções para desenvolver os modelos de rede FFMLP embutidas no software MATLAB automaticamente parametrizam estes valores para a função de ativação escolhida.

No processo de treinamento da rede neural os dados são reordenados de forma aleatória cada vez que todos os dados são apresentados a rede neural. Essa randomização acelera o processo de convergência pois aumenta o espaço de valores assumidos pelos pesos e conseqüentemente o espaço de procura da solução (HAYKIN, 1999).

O fluxograma da Figura 5 a seguir representa a estratégia para a determinação da melhor arquitetura da RNA. O algoritmo, para uma configuração específica de neurônios escondidos, procurou identificar o menor valor possível para o MSE. Assim ao se treinar a rede, se um valor de MSE menor que o valor do treinamento anterior é encontrado, o algoritmo zera o contador de treinamentos que determina quando deve-se aumentar o número de neurônios escondido da RNA. O número máximo de iterações para cada configuração de rede neural é de 10 mil.

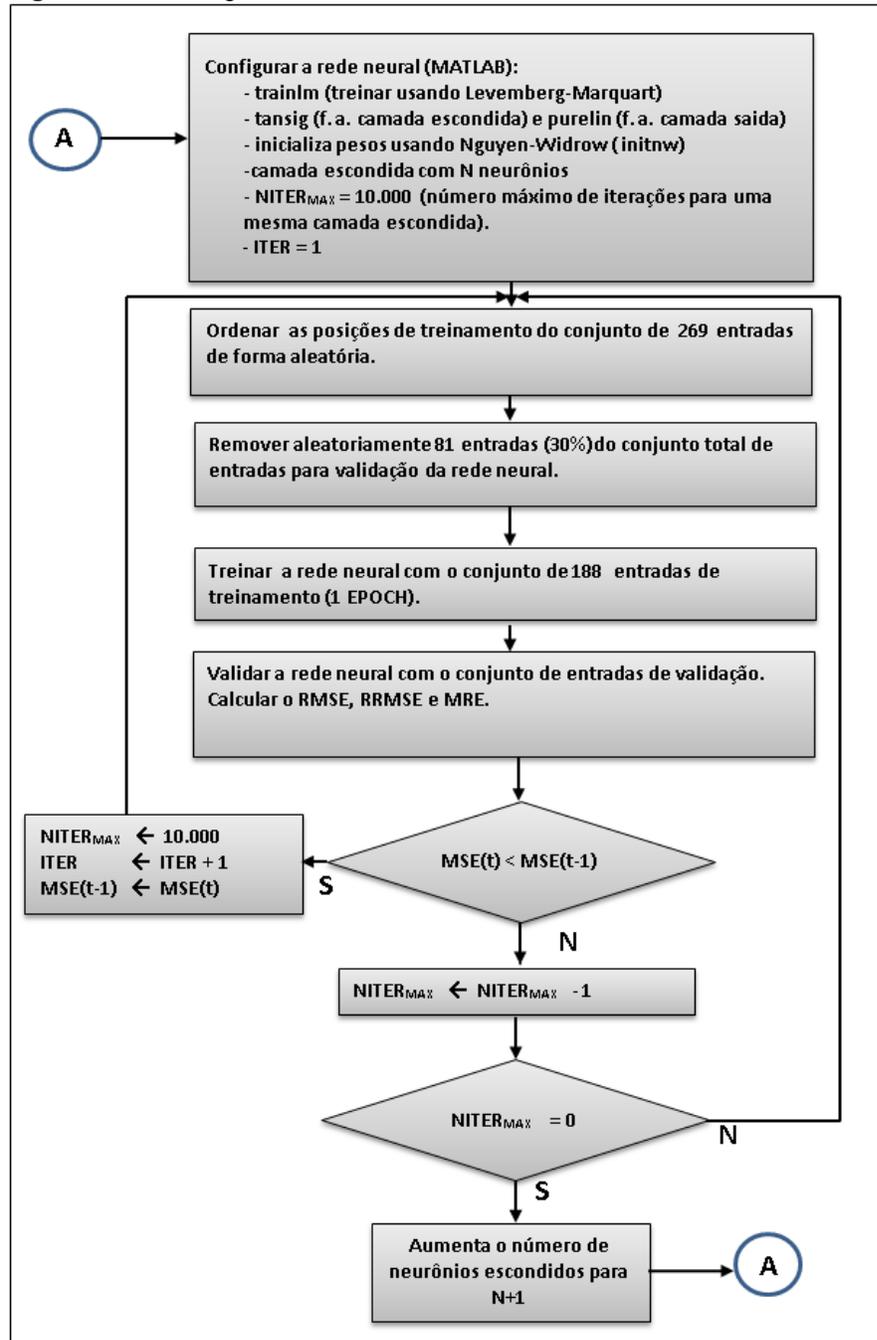
Por exemplo, caso o algoritmo encontre um valor de MSE menor que o valor anterior no 9990<sup>a</sup> treinamento, o contador irá voltar para 1 e o algoritmo tentará novamente por 10 mil vezes encontrar um valor ainda menor para o MSE, antes de se aumentar o número de neurônios escondidos. Em não se encontrando um valor menor de MSE, a RNA é reconfigurado com a adição de um neurônio escondido e o processo é reiniciado.

É importante destacar que repetiu-se a estratégia de treinamento e validação 3 vezes para cada configuração de neurônios escondido, de forma a se minimizar a influência do fator aleatório. Para as amostras obtidas foram calculados o valor médio e o desvio padrão do RMSE, bem como o RRMSE e o MRE.

Uma vez realizado a varredura de diferentes configurações de neurônios escondidos, a configuração de rede FFMLP que apresentou o menor valor para o RMSE repetiu seu processo de treinamento por cerca de 30 vezes. Esta sequência de treinamento seguiu o algoritmo definido na Figura 5, a única diferença é que agora o número de neurônios escondidos não é mais incrementado.

O objetivo desta repetição de treinamentos foi de otimizar os pesos da melhor configuração identificada de forma que se consiga reduzir ainda mais o erro quadrático para o conjunto de dados.

**Figura 5** – Estratégia de treinamento



Fonte: Os Autores

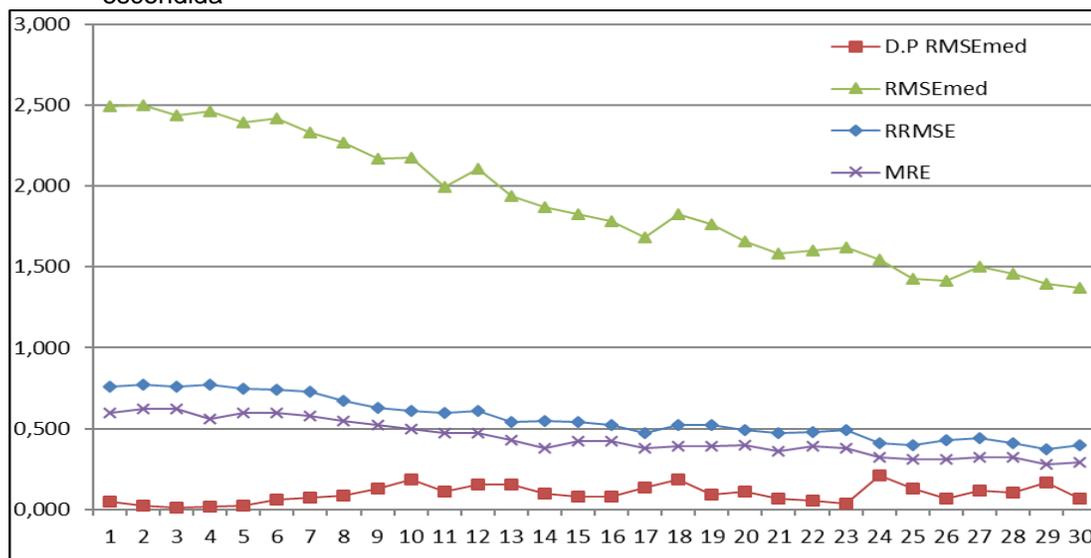
## 5 RESULTADOS E ANÁLISES

### 5.1 Rede Neural Artificial

O gráfico da Figura 6 a seguir descreve a evolução dos valores para o valor médio e o desvio padrão do RMSE, o RRMSE e MRE para cada configuração de neurônios escondidos. Para as medidas de erro RRMSE e MRE, considerou-se apenas valores os associados ao menor RMSE da amostra, pois sua variação foi muito pequena, não incorrendo esta simplificação em erro significativo.

O valor de  $RMSE_{med}$  diminui de maneira quase linear até atingir o valor de 1,5 g/l com 21 neurônios na camada escondida. A partir desta configuração a inclinação da reta se torna mais suave indicando que o número de neurônios escondidos está atingindo a sua saturação em torno deste valor. Os valores de RRMSE e MRE também diminuem a medida que aumenta o número de neurônios na camada escondida, porém apresentam um inclinação menos acentuada se comparada a curva de  $RMSE_{med}$ .

**Figura 6** –  $RMSE_{med}$ , D.P  $RMSE_{med}$ , RRMSE e MRE versus a quantidade de neurônios na camada escondida



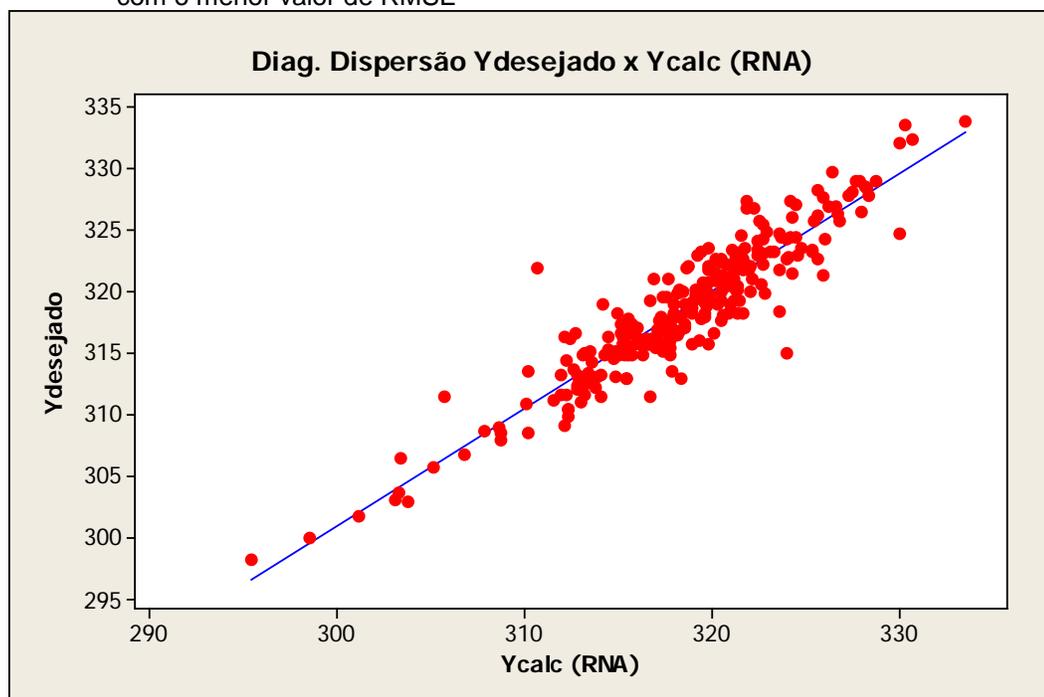
Fonte: Os Autores

Selecionou-se a rede FFMLP com 25 neurônios ( $RMSE \sim 1,42$  g/l) escondidos como a melhor configuração para modelar o problema da concentração cáustica de saída da etapa de vaporização. A escolha foi determinada pelo fato de que a diferença

para os valores médios encontrados para o  $RMSE_{med}$  de configurações com maior número de escondidos não ser representativa. De acordo com Hagan (1996), não é aconselhável utilizar uma rede de maior complexidade quando uma rede mais simples consegue reconhecer os dados de maneira satisfatória.

Repetiu-se 30 vezes o treinamento e validação da rede com 25 escondidos. A Figura 7 representa o diagrama de dispersão entre a resposta desejada e a calculada prevista pela rede FFMLP que apresentou o menor RMSE, com um valor de  $R^2$  de cerca de 86%. Os valores de RMSE, RRMSE e MRE, desta vez levando-se em consideração o conjunto total de entradas, são respectivamente 2,112 g/l, 0,662% e 0,480%.

**Figura 7** – Diagrama de Dispersão Y desejado versus Y calculado pela RNA com 25 escondidos com o menor valor de RMSE



Fonte: Os Autores

## 5.2 Regressão Linear Múltipla

A Tabela 1 abaixo apresenta o resultado da regressão linear múltipla calculado pelo software SPSS para o conjunto de dados desta pesquisa. O valor encontrado para *P-value* próximo de zero indica que as variáveis de entrada tem influência na

variabilidade de saída com alto grau de probabilidade. O valor de  $R^2$  encontrado para esta regressão demonstra que o modelo de regressão consegue explicar cerca de 65% da variabilidade da resposta.

Os valores encontrados para os coeficientes de regressão  $\beta$  resultam na construção da seguinte combinação linear entre as entradas:

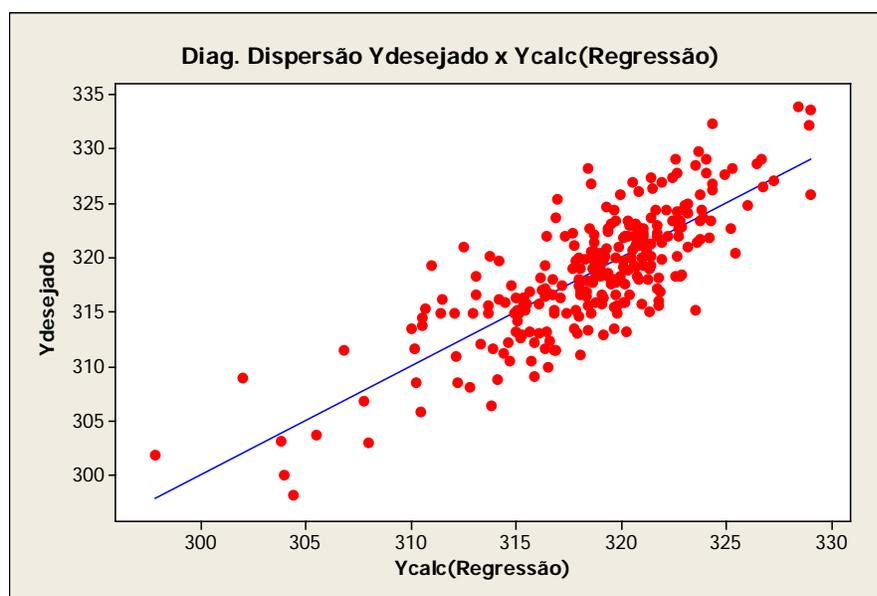
$$Y_{\text{calc}} = 7,706 + 0,948X_1 + 0,001X_2 + 0,027X_3 + 0,347X_4 \quad (11)$$

**Tabela 1** – Resultado Regressão Linear Múltipla

<b>REGRESSÃO LINEAR MÚLTIPLA</b>					
	<b>gl</b>	<b>SQ</b>	<b>MQ</b>	<b>F</b>	<b>P-value</b>
Regressão	4	5541,210354	1385,302589	116,9084178	3,13513E-57
Resíduo	264	3128,259625	11,84946828		
Total	268	8669,46998			

Fonte: Os Autores

**Figura 8** – Diagrama de Dispersão Y desejado versus Y calculado pela regressão linear múltipla



Fonte: Os Autores

A Figura 8 acima apresenta o gráfico de dispersão entre o valor desejado e o calculado pela regressão linear e a reta representado por (11). A análise dos resíduos apresentou distribuição Gaussiana com média aproximadamente zero e desvio-padrão de cerca de 3,5 g/l.

Para os valores de MSE, MRE e RRMSE encontrou-se respectivamente 3,410 g/l, 0,849% e 1,072%.

## 6 CONCLUSÃO

Comparando-se os modelos de regressão linear múltipla e de redes neurais FFMLP na Tabela 2 a seguir e observando-se a dispersão nas Figuras 7 e 8 pode-se concluir que o desempenho é significativamente melhor em favor do modelo de redes neurais, com um valor de RMSE quase 40% vezes menor e igualmente são, nesta ordem de grandeza, menores os valores de MRE e RRMSE.

**Tabela 2** – Comparativo RNA x Regressão

<b>Comparativo RNA x Regressão</b>						
	RMSE	diff %	MRE	diff %	RRME	diff %
RNA	2,112	38%	0,480	43%	0,662	38%
Regressão	3,410		0,849		1,072	

**Fonte:** Os Autores

O desempenho superior da rede neural artificial pode ser atribuído ao fato de que são consideradas aproximadores universais de função, como já foi demonstrado por Hecht-Nielsen (1987). Ao contrário da regressão linear múltipla, redes neurais do tipo FFMLP não dependem de nenhum conhecimento prévio ou premissa do relacionamento funcional entre as entradas e as saídas, é importante entretanto que a amostra selecionada contenham suficiente informação para descrever a variável de resposta, bem como a sua distribuição, para selecionar apropriadamente a função não linear de ativação (MATIGNON, 2005).

Adicionalmente a rede FFMLP, graças a função sigmoideal dos neurônios da camada escondida, consegue representar tanto as relações lineares como as não lineares entre as variáveis, ao passo que a regressão linear múltipla consegue apenas extrair e modelar a componente linear dessas relações (TANG; LEUNG; BAGCHI, 2006).

O trabalho de Silva et al. (2011) propõe a aplicação da lógica *fuzzy* para equacionar o problema do cálculo da concentração cáustica, obtendo após otimização do modelo um RMSE de 4 g/l. Portanto a lógica *fuzzy* apresenta, para esta aplicação,

uma acuracidade inferior em relação ao modelo de rede neural FFMLP proposto nesta pesquisa.

Sugere-se como temas para pesquisas futuras a utilização de outras metodologias de previsão como a Regressão Não Linear Múltipla ou *Support Vector Machine* (SVM) de forma a se comparar o resultados obtidos com o presente trabalho.

## REFERÊNCIAS

- BARCELÓ, J. **Fundamentals of Traffic Simulation**. International Series in Operations Research & Management Science, 2010. <http://dx.doi.org/10.1007/978-1-4419-6142-6>
- BARRETO, H.; HOWLAND, F.. **Introductory econometrics**: using Monte Carlo simulation with Microsoft excel. Cambridge University Press, 2006.
- BLACKWELL, W. J.; CHEN, F. W. **Neural networks in atmospheric remote sensing**. Editora Artech House, 2009.
- CHARPENTIER, E; LESNE, A; KAPITONOVICH, N. **Kolmogorov's heritage in mathematics**. Editora Springer, 2007. <http://dx.doi.org/10.1007/978-3-540-36351-4>
- CONSTANTINO, V. R. L. et al . **Preparação de compostos de alumínio a partir da bauxita**: considerações sobre alguns aspectos envolvidos em um experimento didático. Quím. Nova, São Paulo , v. 25, n. 3, 2002 .
- DAAMEN, W.; BUISSON, C.; HOOGENDOORN, S. P. (Ed.). **Traffic simulation and data: validation methods and applications**. CRC Press, 2014. <http://dx.doi.org/10.1201/b17440>
- DEMUTH, H.; BEALE, M.; HAGAN, M. **Neural network toolbox user's guide. version 6.02** The MathWorks Inc., 2009.
- DEVORE, J. L. **Probability & statistics for engineering and the sciences**. 8a ed., Duxbury Press, 2012.
- DUBROVIN, V.; SUBBOTIN, S. **Neural network method in plant spectral recognition**. From Laboratory Spectroscopy to Remotely Sensed Spectra of Terrestrial Ecosystems, p. 147, 2002. [http://dx.doi.org/10.1007/978-94-017-1620-8\\_7](http://dx.doi.org/10.1007/978-94-017-1620-8_7)
- GIROSI, F.; POGGIO, T. **Representation properties of networks**: kolmogorov's theorem is irrelevant. Neural Computation, v. 1, n. 4, p. 465-469, 1989. <http://dx.doi.org/10.1162/neco.1989.1.4.465>
- FAUSETT L. **Fundamentals of neural networks architectures, algorithms and applications**. Prentice Hall, 1994.
- FURTADO, J.; URIAS, E. **Recursos naturais e desenvolvimento**: estudos sobre o potencial dinamizador da mineração na economia brasileira. IBRAM-Instituto Brasileiro de Mineração, 2013.

HAGAN, M. T.; DEMUTH, H. B.; BEALE, M. H **Neural network design**. Boston: Pws Pub., 1996. <http://dx.doi.org/10.1109/72.329697>

HAGAN, M. T.; MENHAJ, M. B Training feedforward neural networks with the Marquardt algorithm. **IEEE Transactions on Neural Networks**, v. 5, 989-993, 1994.

HAYKIN, S. S. **Neural networks: a comprehensive foundation**. Upper Saddle River: Prentice-Hall, 1999

HECHT-NIELSEN, R. Kolmogorov's mapping neural network existence theorem. **Proceedings IEEE International Conference on Neural Networks**, v. II, p. 11-13, 1987.

HILL, T.; LEWICKI, P. **Statistics: methods and applications: a comprehensive reference for science, industry, and data mining**. StatSoft, Inc., 2006.

HIND, R. A.; BHARGAVA, S. K.; GROCCOTT, S. C. The surface chemistry of Bayer process solids: a review. Colloids and surfaces A. **Physicochemical and engineering aspects**, n.146, p.359-374, 1999. [http://dx.doi.org/10.1016/S0927-7757\(98\)00798-5](http://dx.doi.org/10.1016/S0927-7757(98)00798-5)

IAI – **International Aluminium Institute**. Disponível em : [www.world-aluminium.org](http://www.world-aluminium.org)  
Acesso em: 30 set. 2013.

JEON, J. **Fuzzy and neural network models for analyses of piles**. Editora ProQuest, 2007.

KOLMOGOROV, A.N. On the representation of continuous functions of several variables by a superposition of continuous functions of one variable and addition. In: **Dokl. Akad. Nauk SSSR**. 1957. p. 953-956.

MASTERS, T. **Practical neural network recipes in C++**. Morgan Kaufmann, 1993.

MATIGNON, R. **Neural network modeling using SAS enterprise miner**. AuthorHouse, 2005.

MONTGOMERY, D. C.; RUNGER, G. C. **Estatística aplicada e probabilidade para Engenheiros**. 4. ed., Editora LTC, 2009.

NANCI, L. C. et al. O PCP no controle estratégico. In: **LUSTOSA, L. et al. (Orgs.) Planejamento e controle da produção**. Rio de Janeiro: Elsevier, 2008.

PANDYA, A. S.; MACY, R. B. **Pattern recognition with neural networks in C++**. CRC press, 1995.

REED, R. D.; MARKS, R. J. **Neural smithing: supervised learning in feedforward artificial neural networks**. MIT Press, 1998.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, v. 65, n. 6, p. 386, 1958.  
<http://dx.doi.org/10.1037/h0042519>

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Learning internal representations by error propagation**. California University San Diego La Jolla Institute For Cognitive Science, 1985.

SCHROEDER, M. R. **Computer speech**: recognition, compression, synthesis. Editora Springer, 2004. <http://dx.doi.org/10.1007/978-3-662-06384-2>

SILVA FILHO, E. B.; ALVES, M. C. M. & MOTA, M. Lama vermelha da indústria de beneficiamento de alumina: produção, características, disposição e aplicações alternativas. **Revista Matéria**, v.12, n.2, p.322-338, 2007. <http://dx.doi.org/10.1590/S1517-70762007000200011>

SILVA, R. G et al. Lógica Fuzzy aplicada para a produção de alumina. **Anais do XIV Simpósio de Pesquisa Operacional e Logística da Marinha**, 2011.

SIVANANDAM, S. N.; DEEPA, S. N. **Introduction to neural networks using Matlab 6.0**. Tata McGraw-Hill Education, 2006.

SLACK, N ; LEWIS, M. **Estratégia de operações**. 2 ed. Porto Alegre: Bookman, 2009.

STEINER, M. T. A. **Notas de Aula Disciplina TP703**. Aplicações da Pesquisa Operacional. Universidade Federal do Paraná (UFPR), 2012.

TANG, Z.; LEUNG, C. W.; BAGCHI, K. Improving population estimation with neural network models. **Advances in Neural Networks-ISNN**, p. 1181-1186, 2006. [http://dx.doi.org/10.1007/11760191\\_172](http://dx.doi.org/10.1007/11760191_172)

WALPOLE, R. E et al. **Probability and statistics for engineers and scientists**. 9 ed., Prentice Hall, 2012.

YONABA, H.; ANCTIL, F.; FORTIN, V. Comparing sigmoid transfer functions for neural network multistep ahead stream flow forecasting. **Journal of Hydrologic Engineering**, v. 15, n. 4, p. 275-283, 2010. [http://dx.doi.org/10.1061/\(ASCE\)HE.1943-5584.0000188](http://dx.doi.org/10.1061/(ASCE)HE.1943-5584.0000188)

ZIELESNY, A. **Intelligent systems reference library**: from curve fitting to machine learning: an illustrative guide to scientific data analysis and computational intelligence. Vol. 18, Springer, 2011.

ZHANG, X. S. **Neural networks in optimization**. Editora Springer, 2000. <http://dx.doi.org/10.1007/978-1-4757-3167-5>



Artigo recebido em 07/12/2014 e aceito para publicação em 11/12/2014  
DOI: <http://dx.doi.org/10.14488/1676-1901.v15i3.1941>