

ANÁLISE COMPARATIVA ENTRE REDES NEURAIS ARTIFICIAIS EM ESTUDO DE CASO DE MANUTENÇÃO PREDITIVA

A. G. Alvarenga, andre.guaracy@sga.pucminas.br¹
L. M. Rodrigues, lucas.rodrigues.869763@sga.pucminas.br¹
R. M. da Silva, rafael.marcial@sga.pucminas.br¹
J. Jr. Landre, janes@pucminas.br²

¹Graduação Pontifícia Universidade Católica de Minas Gerais, Rua Dom José Gaspar, 500, Coração Eucarístico, Belo Horizonte, MG, Brasil.

²Pós-Graduação Pontifícia Universidade Católica de Minas Gerais, Rua Dom José Gaspar, 500, Coração Eucarístico, Belo Horizonte, MG, Brasil.

Resumo: Dentre as técnicas de manutenção preditiva, a análise de vibração é a que mais oferece contribuições para a verificação do comportamento de rolamentos, porém é altamente dependente da interpretação humana. A partir do estudo de caso de predição de falha em rolamentos utilizando uma rede Perceptron de Múltiplas Camadas, em que é desenvolvido o estudo de redes neurais no sentido de contribuir com a análise vibracional, foi possível comparar o resultado experimental da rede com a correspondente similar oferecida pelo MATLAB®. Para realizar a comparação foram analisadas dois tipos de redes, uma classificatória e outra de aproximação de função, ambas foram desenvolvidas separadamente pelo programa computacional próprio desenvolvido no MATLAB®, enquanto a outra é oferecida no toolbox de redes neurais do próprio software.

Palavras-chave: Rede Neural, Toolbox, Cross Validation, MATLAB®, Rolamentos.

1. INTRODUÇÃO

A demanda por técnicas de diagnose e predição de falha são necessárias à medida que se deseja aumentar a eficiência produtiva e reduzir os custos de manutenção. O operacional das plantas industriais depende diretamente do funcionamento adequado de alguns equipamentos essenciais, como motores, bombas, ventiladores, entre outros, também conhecidos como ativos. Muitos deles utilizam mancais de rolamento em sua estrutura, sendo o rolamento uma das partes críticas de cada ativo.

Independente da variável controlada os métodos de Inteligência computacional são ótimos auxiliares para as análises das condições de funcionamento de cada máquina. Quanto maior o ritmo produtivo, maiores são as perdas relacionadas a uma parada repentina de um equipamento, sobretudo os que funcionam ininterruptamente.

A manutenção é uma área que tem a capacidade de reduzir custos de produção e de serviço, garantindo que os processos funcionem adequadamente pelo maior tempo possível, para entregar os melhores resultados. A manutenção preditiva pode ser definida como sendo uma técnica de previsão ou presságio da falha, através de parâmetros mensuráveis que possam apontar uma falha antes do colapso (Siqueira, 2012). Dentre os processos de manutenção existentes, a manutenção preditiva utilizando inteligência computacional está sendo cada vez mais desenvolvida por permitir um planejamento mais preciso, atuando no momento adequado, antes que o equipamento falhe.

Para melhorar a eficiência da análise de dados e a assertividade de resultados é preciso encontrar a melhor rede neural para cada situação. Para chegar a essa conclusão é necessário fazer comparações e testes empíricos que proporcionem a tomada de decisões corretas. A proposta do presente trabalho é a realização de uma comparação crítica entre solução em software desenvolvidas no trabalho de fim de curso pelos autores desse artigo (Alvarenga *et al.*, 2017), denominada BearingNetwork, e a solução disponível no MATLAB® (toolbox). A rede escolhida para alcançar o objetivo do trabalho citado anteriormente foi a Perceptron de Múltiplas Camadas com retro propagação do erro.

2. DADOS UTILIZADOS

Para a análise proposta foram utilizados dados coletados por um experimento feito na universidade americana Case Western Reserve University. O experimento consiste no monitoramento de um motor e seus rolamentos, onde foram induzidos, utilizando descargas elétricas controladas, defeitos de três tamanhos diferentes em locais específicos. Os

defeitos foram induzidos na pista interna, na pista externa ou no elemento rolante do rolamento, sempre de forma isolada, ou seja, sem defeitos simultâneos. Os tamanhos dos defeitos foram: 0,1778 mm; 0,3556 mm; 0,5334 mm.

2.1. Pré-processamento dos Dados

Os dados disponíveis no site da Case Western University são colhidos por um acelerômetro localizado próximo ao rolamento onde está presente o defeito. Como a análise de vibrações não é o foco do presente trabalho, será apresentado um breve resumo sobre o caminho percorrido entre os dados obtidos até os dados utilizados como entradas das redes neurais.

Optou-se por utilizar uma abordagem através de análises tanto no domínio do tempo quanto no domínio da frequência. No domínio do tempo, foram utilizados os valores de RMS, Fator de Crista e Fator K, calculados diretamente da forma de onda gerada pelo acelerômetro. Esses valores tiveram seus comportamentos descritos por Mitchell *et al.* (1993) onde foi mostrada a tendência de cada um de acordo com o aumento do defeito.

Após os cálculos dos fatores no domínio do tempo foi aplicado o Envelope conforme mostrado por Haykyn *et al.* (2001), Vasconcelos *et al.* (2007) e Randall *et al.* (2011). Do Envelope de cada sinal foram extraídos os picos de amplitude presente em torno das frequências características de defeito de cada componente do rolamento, pista interna, pista externa e elemento rolante. Tais amplitudes também serão variáveis de entrada das redes neurais e serão os principais indicadores do local do defeito.

Os seis fatores supracitados, RMS, Fator de Crista, Fator K e as amplitudes das frequências características da pista interna, pista externa e elemento rolante, junto da carga aplicada ao rolamento em HP, serão as entradas das redes neurais desenvolvidas.

3. MÉTODOS DE ANÁLISE

Sendo o objetivo a comparação de duas técnicas distintas é preciso que existam parâmetros comuns entre elas para que conclusões mensuráveis sejam criadas. Para tanto, foi necessário construir primeiramente uma solução que atendesse um problema real e, então, poder replicar os parâmetros em outras fontes e poder compará-los. Utilizando as mesmas entradas nas duas técnicas, serão comparadas as eficiências das respostas das saídas.

O problema real analisado é um estudo de caso do desenvolvimento de um aplicativo em MATLAB® para o auxílio de manutenção preditiva em rolamentos em que os melhores resultados encontrados, após testes e validações empíricas, foram com duas redes neurais distintas, uma destinada para classificar o local do defeito e outra para aproximação do tamanho físico do defeito. Os autores denominaram o aplicativo feito de BearingNetwork. A segunda técnica analisada para a comparação foi o aplicativo Neural Net Fitting GUI (Graphic User Interface), parte do toolbox de redes neurais do MATLAB®.

3.1. BearingNetwork

O BearingNetwork consiste em duas redes neurais com mesma quantidade de entradas, sete, que são amplitude no elemento rolante, amplitude na pista externa, amplitude na pista interna, valor RMS, fator de crista, fator K e o valor da carga. Porém, a rede destinada para a classificação do local do defeito do rolamento tem três saídas, cujas respostas são binárias, pois a combinação dessas indica o local do defeito predominantes, de acordo com a Tab. 1.

Para isso, a resposta de saída (y), para cada i , foi pós-processada de acordo com a Eq. (1) para se obter uma combinação binária no conjunto de saídas da rede de classificação do defeito.

$$y_i^{pós} = \begin{cases} 1, & y_i^{saída} \geq 0,5 \\ 0, & y_i^{saída} < 0,5 \end{cases} \quad (1)$$

A outra rede foi usada para prever o tamanho do defeito, por isso tem uma única saída.

Tabela 1. Combinação binária das saídas da rede neural para a classificação do local do defeito.

Saída 1	Saída 2	Saída 3	Local do defeito
1	0	0	Pista Interna
0	1	0	Pista Externa
0	0	1	Elemento Rolante

Para o treinamento das redes neurais, o conjunto amostral foi dividido de forma aleatória em dois grupos, o subconjunto de treinamento e o de teste, que correspondem, respectivamente, a 90% e 10% dos dados disponíveis (total de 80 dados). Depois esses subconjuntos foram normalizados em um intervalo fechados de -1 a 1 para acelerar a fase de treinamento e evitar as regiões de saturação do neurônio.

De acordo com Braga *et al.* (2000), o desempenho da rede neural depende da topologia dessa rede e dos parâmetros de treinamento utilizados, o que afeta a sua velocidade, eficiência de aprendizado, a tolerância a ruídos e a capacidade de generalização. Considerando topologia a estrutura da rede neural, ou seja, o número de neurônios e o número de camadas de neurônios.

A melhor topologia das redes, utilizadas no BearingNetwork, foi determinada pelo método Cross Validation, mais especificamente a Random Subsampling Cross Validation (Validação Cruzada por Amostragem Aleatória), em que são comparadas diversas topologias de rede para que seja encontrada a melhor para a situação. Devido à incerteza da distribuição dos dados no espaço amostral, foram testadas uma e duas camadas intermediárias para solucionar o problema das duas redes propostas.

Para uma camada intermediária foram testadas topologias desde 2 neurônios na camada intermediária até 52, variando de 2 em 2. E para duas camadas intermediárias os testes foram feitos partindo de 2 neurônios até 16 em ambas as camadas intermediárias, variando de 1 em 1, realizando-se todas as combinações possíveis. Os números de neurônios testados foram limitados a 52 e 16 empiricamente, sendo que aumentando os números de neurônios além desses valores não se obtinham variações positivas relevantes na resposta. Em ambos os casos todos os neurônios tiveram a mesma função de ativação, a sigmóide.

Todas as redes treinadas tiveram a mesma taxa de aprendizagem de 0,1 e taxa de momentum de 0,9. O termo momentum foi usado para acelerar o treinamento da rede com o backpropagation. Além disso, cada rede neural foi treinada 5 vezes para, em seguida, se obter a média do desempenho de cada topologia candidata durante a fase de testes.

Os critérios de parada abordados durante a fase de treinamento das redes foram:

- a) Quando o número de épocas ou iterações atingir um valor máximo de 10000, ou;
- b) Quando a diferença do erro quadrático médio entre duas iterações sucessivas for igual ou menor que 10^{-6} .

No geral, a metodologia empregada para a fase de treinamento da rede neural do BearingNetwork foi adaptada da metodologia de treinamento da rede neural proposta por Silva *et al.* (2016).

3.2. Neural Net Fitting GUI

Assim como no BearingNetwork foram feitas duas redes neurais artificiais, uma para a classificação do local com sete entradas e três saídas, enquanto a outra, para a aproximação do tamanho de defeito, também possui sete entradas mas gera apenas uma saída, o Neural Net Fitting GUI teve também essas duas redes com as mesmas entrada e saídas.

Esse aplicativo foi utilizado com seus parâmetros padrões e utilizando o treinamento Levenberg-Marquardt. Além disso, os dados amostrais foram divididos em três subconjuntos, o de treinamento, o de teste e o de validação com, respectivamente, 70%, 15% e 15% da quantidade total de amostras.

Para determinar a topologia da rede foi utilizado o método do Cross Validation, assim como o realizado para o BearingNetwork, porém o Neural Net Fitting GUI limita o número de camadas intermediárias para apenas uma. Portanto, o Cross Validation para ambas as redes foi feito com apenas uma camada intermediária com o número de neurônios indo de 2 a 52, variando de 2 em 2.

O Neural Net Fitting GUI nos informa somente os erros quadráticos médios (MSE) da fase de treinamento, de validação e de teste, não nos informa a taxa de acerto, então foi escolhido como parâmetro de comparação para a seleção da melhor topologia a rede com o menor MSE de teste, pois é esse valor que irá indicar o grau de generalização da rede. Quanto menor o MSE de teste maior o grau de generalização da rede.

4. RESULTADOS E ANÁLISE

A seguir serão apresentados os melhores resultados para o Cross Validation feito tanto para o BearingNetwork quanto para o Neural Net Fitting GUI e posteriormente será feita uma comparação entre a performance obtida por cada rede.

4.1. Cross Validation BearingNetwork

Como dito anteriormente, a escolha da topologia da rede neural no BearingNetwork foi realizado pelo método Cross Validation, utilizando-se uma e duas camadas intermediárias cujos melhores resultados na fase de teste encontram-se na Tab. 2.

Os valores com * são referentes à taxa de acerto da rede neural, portanto valores altos indicam uma melhor generalização. Os valores com ** são referentes ao erro relativo médio da rede, então quanto mais próximo de zero maior é a assertividade da rede.

O erro quadrático médio e a taxa de acerto ou o erro relativo médio durante o teste servem de indicadores da precisão e do desempenho da rede neural frente a novos valores, pois os dados utilizados nessa fase não participaram da fase de treinamento. Caso os dados utilizados para teste fossem os mesmos utilizados no treinamento, a eficiência das respostas obtidas seria duvidosa, pois o software analisaria condições idênticas às do treino.

Com a Tabela 2, observa-se que para a classificação do local do defeito apenas uma camada intermediária foi suficiente para obter resultados bem próximos do valor ideal, com relação ao mesmo tipo de problema treinado na rede com duas camadas intermediárias.

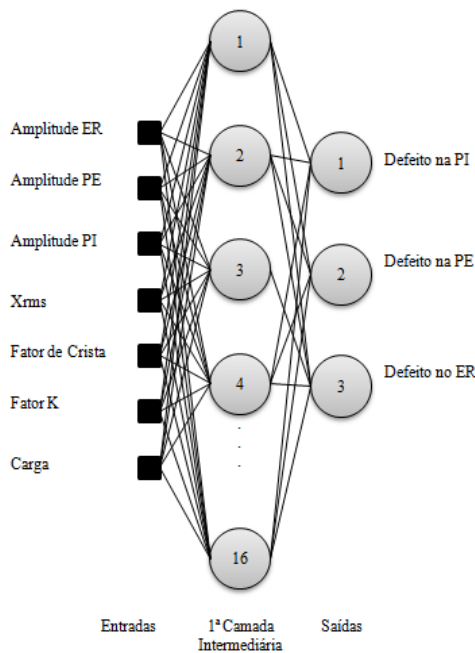
Quanto á aproximação do tamanho do defeito, embora as duas topologias de rede tenham demonstrado taxas de desempenho próximas, o valor para o erro quadrático foi menor com treinamento para duas camadas intermediárias. Portanto ela pode garantir uma melhor precisão na resposta da rede.

Diante disso, as redes escolhidas foram:

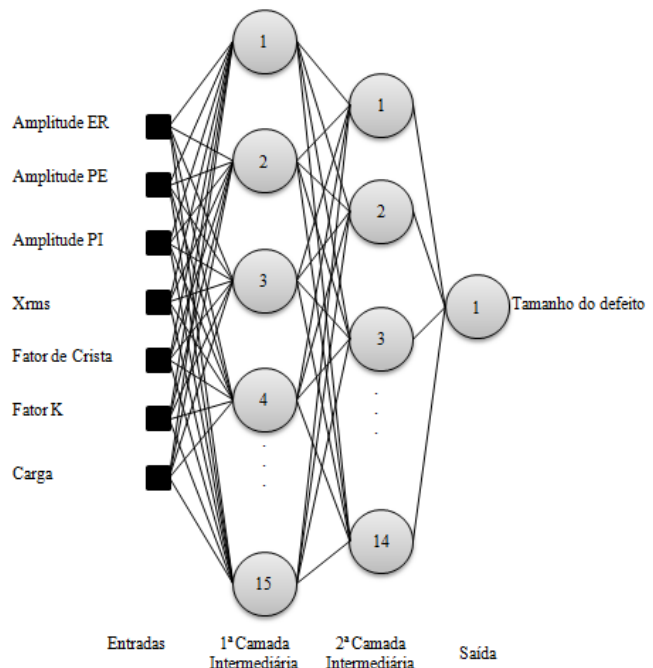
- Classificador do local do defeito: 16 neurônios na camada intermediária, conforme a Fig. 1.a;
- Aproximação do tamanho do defeito: 15 neurônios na primeira camada intermediária e 14 neurônios na segunda camada intermediária, conforme a Fig. 1.b.

Tabela 2. Resultados das melhores topologias da rede neural para o BearingNetwork.

1 camada intermediária					
	Número de neurônios	Número de iterações	Erro quadrático médio	Taxa de desempenho (%)	
Classificação do local do defeito	16	1578	0,0038	100*	
Aproximação do tamanho do defeito	12	2209,4	0,0706	20,9**	
2 camadas intermediárias					
	Número de neurônios na 1ª camada	Número de neurônios na 2ª camada	Número de iterações	Erro quadrático médio	Taxa de desempenho (%)
Classificação do local do defeito	7	6	578,6	0,0305	97,5*
Aproximação do tamanho do defeito	15	14	1098,4	0,0299	21,2**



a. Rede neural artificial do local do defeito



b. Rede neural artificial do tamanho do defeito

Figura 1. Representação das topologias das duas redes neurais do BearingNetwork.

Em termos de simplicidade foram usadas as siglas PI para pista interna, PE para pista externa e ER para elemento rolante.

Vale enfatizar que o método do Cross Validation teve a função de determinar a melhor topologia através da comparação dos melhores desempenhos médios globais durante certo número de treinamentos da rede. Com isso, a rede

pode apresentar, por exemplo, um valor de taxa de acerto individual elevado num dado treinamento, mas apresentar um baixo valor de taxa de acerto médio em n treinamentos.

Com as topologias definidas, a evolução dos erros quadráticos da rede, tanto na fase de treinamento quanto na fase de teste, para a rede de classificação do local do defeito e para a rede de aproximação do tamanho do defeito são representados pela Fig. 2 e Fig. 3, respectivamente.

A Figura 2 representa a rede classificadora do local de defeito com as características na Tab. 3.

Tabela 3. Parâmetros da rede neural para a classificação do local do defeito do BearingNetwork.

Rede para a classificação do local de defeito	
Taxa de acerto (%)	100
Erro quadrático médio do treinamento	0,0007
Erro quadrático médio do teste	0,0005
Épocas	2163

E a Figura 3 representa a rede classificadora do local de defeito com as características na Tab. 4.

Tabela 4. Parâmetros da rede neural para a aproximação do tamanho do defeito do BearingNetwork.

Rede para a aproximação do defeito	
Taxa de erro relativo (%)	6,42
Erro quadrático médio do treinamento	0,0208
Erro quadrático médio do teste	0,0143
Épocas	911

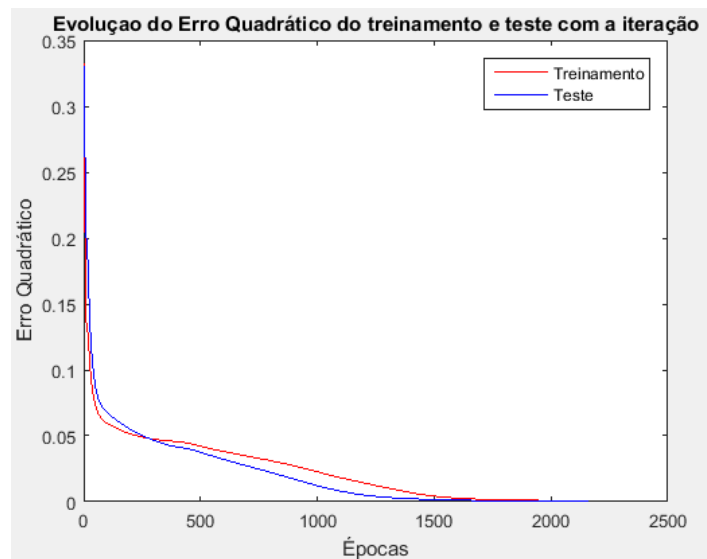


Figura 2. Performance da rede neural para localização do defeito do BearingNetwork.

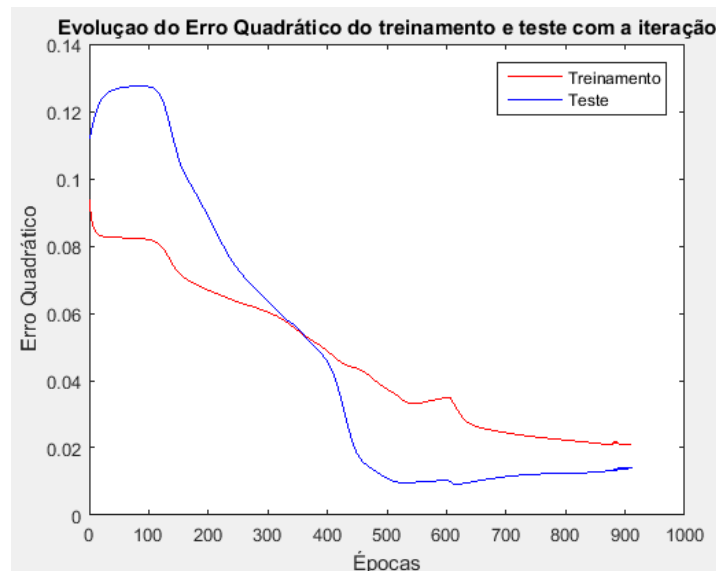


Figura 3. Performance da rede neural para tamanho do defeito do BearingNetwork.

Os resultados obtidos nas Tab. 3 e Tab. 4 são diferentes dos resultados do Cross Validation, pois realizaram vários treinamentos nessas mesmas redes selecionadas até encontrar uma configuração com os melhores desempenhos. Pois ao iniciar o processo iterativo do backpropagation, os valores dos pesos iniciais são escolhidos aleatoriamente contribuindo ou não para encontrar os valores mínimos globais do erro quadrático durante a fase de treinamento da rede.

E na Figura 2 e Figura 3 percebe-se que à medida que foi aumentando o número de iterações (épocas), o erro quadrático médio da fase de treinamento foi diminuindo e, conseqüentemente, o erro quadrático da fase de teste também diminuiu. Esse decaimento do erro é feito pela técnica backpropagation, na qual busca convergir para um erro quadrático mínimo local.

4.2. Cross Validation Neural Net Fitting

Já para o Cross Validation feito para o Neural Net Fitting GUI do Matlab são mostrados a seguir na Tab. 5 os três melhores resultados para a rede que classifica o local do defeito e para a rede que aproxima o tamanho da falha. Esses resultados foram escolhidos os melhores do Cross Validation por possuírem o menor MSE de teste, conforme já mencionado.

Tabela 5. Resultados das melhores topologias da rede neural para o Neural Net Fitting.

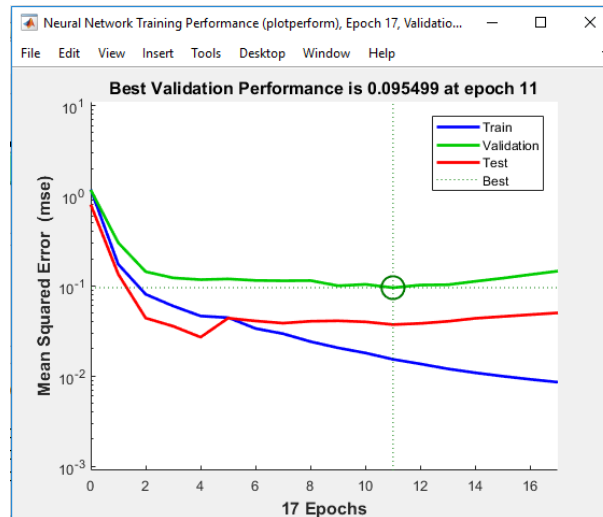
Classificação do local do defeito				
Nº de neurônios	Épocas	MSE Treinamento	MSE Teste	MSE Validação
24	10	0,0415984	0,0538472	0,06850532
22	11,4	0,0456361	0,0605075	0,11166684
18	11	0,03933932	0,0628408	0,11527726
Aproximação do tamanho do defeito				
Nº de neurônios	Épocas	MSE Treinamento	MSE Teste	MSE Validação
40	11,4	0,005763308	0,0163635	0,02499626
30	8,2	0,00988157	0,0190232	0,03032832
38	9,8	0,008036706	0,0195679	0,02419824

Pode-se observar que, para cada rede, o número de neurônios dos melhores resultados foram bem próximos, portanto foram escolhidas as redes com os menores MSE de teste sem se importar com o tempo de processamento, que aumenta com a quantidade de neurônios, que cada rede gasta para processar os resultados. Portanto, a rede neural para a classificação do local defeito será uma rede com uma camada intermediária com 24 neurônios e a rede neural para a aproximação do tamanho do defeito será constituída de uma camada intermediária com 40 neurônios.

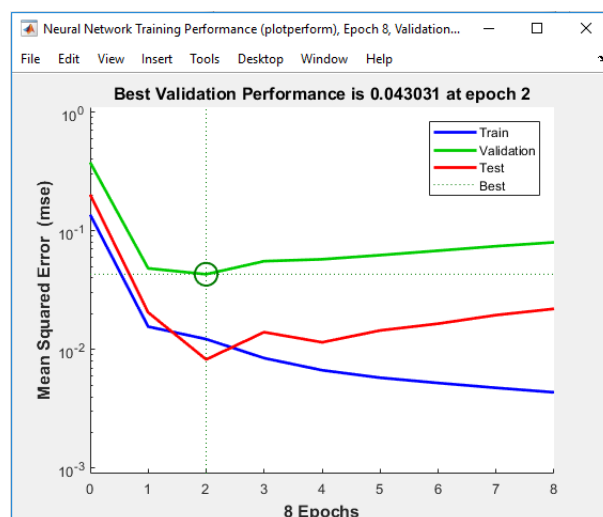
Foram feitos outros treinamentos com a topologia escolhida e os pesos dos melhores resultados foram salvos. Os resultados obtidos no Neural Net Fitting GUI para a classificação do local do defeito estão na Tab. 6 e na Fig. 4 e para a aproximação do tamanho do defeito estão na Tab. 7 e na Fig. 5.

Tabela 6. Parâmetros da rede neural para a classificação do local do defeito do Neural Net Fitting.

Rede para a classificação do local de defeito	
Erro quadrático médio do treinamento	0,0153
Erro quadrático médio da validação	0,0955
Erro quadrático médio do teste	0,0372
Épocas	17

**Figura 4. Performance do rede neural para local do defeito do Neural Net Fitting.****Tabela 7. Parâmetros da rede neural para a aproximação do tamanho do defeito do Neural Net Fitting.**

Rede para a aproximação do defeito	
Erro quadrático médio do treinamento	0,0122
Erro quadrático médio da validação	0,0430
Erro quadrático médio do teste	0,0082
Épocas	8

**Figura 5. Performance do rede neural para tamanho do defeito do Neural Net Fitting.**

Com a Tab. 6 e Tab. 7 nota-se os valores do erro quadrático médio de treinamento e de teste foram maiores que os valores encontrados no BearingNetwork para a rede de classificação do local do defeito. Mas a quantidade de épocas foi menor. Isso pode ter acontecido pelo método de treinamento ser distintos entre os aplicativos, com um utilizando o

Momentum e o outro o Levenberg-Marquardt. Pois de acordo com Hagan *et al.*(1994), o Levenberg-Marquardt tem velocidade de treinamento maior que o algoritmo backpropagation.

Além da diferença dos métodos, os critérios de paradas abordados por cada aplicativo durante a fase de treinamento foram distintos. No Neural Net Fitting GUI a fase de validação, que ocorre após cada treinamento em cada iteração, o resultado da validação é utilizado como um critério de parada de treinamento, conforme visto nas Fig. 4 e Fig. 5, o treinamento para quando observa-se que o MSE da validação cresce 6 vezes seguidas, e é obtido como resultado ótimo a rede treinada antes desses 6 incrementos. Além desse critério de parada, é determinado um máximo de 1000 iterações ou quando o MSE chega a zero.

4.3. Comparação das performances

Para a aplicação estudada, no Neural Net Fitting GUI ganha-se em tempo de treinamento, já que esse requer muito menos iterações para ser finalizado, por outro lado, no BearingNetwork ganha-se na precisão, já que esse demonstrou um MSE muito menor. Deve-se observar a prioridade do caso estudado e tentar encontrar um equilíbrio entre essas duas variáveis.

Em relação à topologia das redes, observa-se que foram necessários um número significativamente menor de neurônios para a rede BearingNetwork do que para a rede do Neural Net Fitting GUI, isso pode reduzir a complexidade do treinamento e trazer vantagens em operações mais complexas.

Para uma comparação mais efetiva entre os dois aplicativos utilizados, criou-se um subconjunto teste opcional para avaliar as respostas dos mesmos. Observa-se nos resultados desse subconjunto, mostrados na Tab. 8 e na Tab. 9, que o BearingNetwork apresentou resultados melhores que o aplicativo disponível no MATLAB®, apresentando uma taxa de acerto maior, para o local do defeito, e um valor de erro quadrático menor, para o tamanho do defeito, que seu concorrente. Mas para isso o BearingNetwork demorou mais em seu treinamento para obter essas respostas.

Tabela 8. Comparação das respostas geradas para o local do defeito.

Local do defeito															
Entradas							Saídas desejadas			Repostas BearingNetwork			Respostas Toolbox Matlab		
PI	PE	ER	X rms	F. crista	F. K	Carga	PI	PE	ER	PI	PE	ER	PI	PE	ER
0,0079	0,0089	0,0093	0,1640	19,1300	0,5147	1	0	0	1	0	0	1	0	0	1
0,0076	0,0031	0,0056	0,1147	17,4509	0,2294	1	0	0	1	0	0	1	0	0	1
0,0043	0,0316	0,0082	0,1299	5,3213	0,0898	2	0	1	0	0	1	0	0	1	0
0,0974	0,0124	0,0047	0,4481	8,0665	1,6201	3	1	0	0	1	0	0	1	0	0
0,1323	0,0051	0,0086	0,5252	7,2121	1,9896	0	1	0	0	1	0	0	1	0	0
0,0177	0,0062	0,0066	0,1536	4,3704	0,1032	3	0	0	1	0	0	1	0	0	1
0,0803	0,0536	0,0100	0,2302	6,7023	0,3550	2	1	0	0	1	0	0	1	0	0
0,0621	0,0329	0,0171	0,1653	12,2838	0,3355	1	1	0	0	1	0	0	0	0	1
0,0402	0,0253	0,0297	0,1328	5,8465	0,1031	2	1	0	0	1	0	0	1	0	0
Taxa de acertos (%)										100			88,89		
MSE										0,0005			0,0997		

Tabela 9. Comparação das respostas geradas para o tamanho do defeito.

Tamanho do defeito									
Entradas							Saídas desejadas	Respostas da BearingNetwork	Respostas Toolbox Matlab
PI	PE	ER	X rms	F. crista	F. K	Carga	Tamanho	Tamanho	Tamanho
0,0079	0,0089	0,0093	0,164	19,13	0,5147	1	0,5334	0,4776	0,3089
0,0076	0,0031	0,0056	0,1147	17,4509	0,2294	1	0,1778	0,3384	0,2554
0,0043	0,0316	0,0082	0,1299	5,3213	0,0898	2	0,5334	0,5046	0,3692
0,0974	0,0124	0,0047	0,4481	8,0665	1,6201	3	0,5334	0,5206	0,4531
0,1323	0,0051	0,0086	0,5252	7,2121	1,9896	0	0,5334	0,5225	0,5377
0,0177	0,0062	0,0066	0,1536	4,3704	0,1032	3	0,1778	0,1804	0,3075
0,0803	0,0536	0,01	0,2302	6,7023	0,355	2	0,1778	0,1793	0,3416
0,0621	0,0329	0,0171	0,1653	12,2838	0,3355	1	0,3556	0,3672	0,3342
0,0402	0,0253	0,0297	0,1328	5,8465	0,1031	2	0,3556	0,3176	0,2773
Erro relativo médio (%)								14,10	36,17
MSE								0,0017	0,0156

5. CONCLUSÃO

Como a principal intenção desse artigo era a comparação entre duas ferramentas, o Neural Net Fitting GUI do MATLAB® e o BearingNetwork, que possuem a mesma função fez-se um estudo para encontrar a melhor configuração para cada ferramenta separadamente e, por fim, analisar os resultados obtidos. Analisando as variáveis mais relevantes para se determinar a performance da Rede Neural, o MSE e a taxa de acerto, nota-se que o BearingNetwork apresentou um desempenho superior ao Neural Net Fitting GUI para o caso aplicado.

Notou-se também uma certa limitação no Neural Net Fitting GUI em relação à praticidade de realização de rotinas repetitivas como o Cross Validation. Quanto à configuração da rede, a ferramenta apresenta poucas opções de treinamento, pode-se observar que o treinamento de Levenberg-Marquardt foi rápido, mas pouco eficiente, além disso a ferramenta limita o número de camadas intermediárias da rede à apenas uma. É possível que com duas ou mais camadas intermediárias os resultados tivessem sido mais precisos utilizando-se uma quantidade menor de neurônios.

Apesar disso, não se deve descartar a opção de utilização do Neural Net Fitting GUI para a solução de problemas na engenharia, é uma ferramenta intuitiva, prática e eficiente que possibilita a utilização de redes neurais para iniciantes que ainda não possuem um conhecimento muito aprofundado da teoria, além disso, as limitações apresentadas podem não ser significativas para outros problemas. Porém, para quem deseja resultados melhores e mais confiáveis é aconselhável a elaboração de um algoritmo de rede neural, que, além de garantir maior precisão, possibilita a liberdade para manipular a topologia da rede e os métodos de treinamento mais adequados para cada situação e necessidade.

6. AGRADECIMENTOS

Gostaríamos de agradecer primeiramente à Pontifícia Universidade Católica de Minas Gerais, à Vale e à SKF por apoiarem a pesquisa fornecendo conhecimentos teóricos e práticos além de nos fornecerem a estrutura necessária para o desenvolvimento do trabalho. Agradecemos, também, aos familiares e aos demais colegas da universidade pelo apoio moral e opiniões sempre construtivas.

7. REFERÊNCIAS

- Alvarenga, A.G., Rodrigues, L.M. and da Silva, R.M., 2017, “Desenvolvimento de um aplicativo em Matlab para o auxílio de manutenção preditiva em rolamentos”, Tese de TCC, Pontifícia Universidade Católica de Minas Gerais, Belo Horizonte.
- Bearing Data Center Case Western University. Disponível em: < <https://csegroups.case.edu/bearingdatacenter/pages/welcome-case-western-reserve-university-bearing-data-center-website>>. Acesso em 04 dez. 2017.
- Braga, A.P., Ludemir, T.B. and Carvalho, A.C.P.L, 2000, “Redes neurais artificiais: teoria e aplicações”, Ed. LTC, Rio de Janeiro, Brasil, 262 p.
- Hagan, M.T. and Menhaj, M.B., 1994, “Training feedforward networks with the Marquardt algorithm”, IEEE Transaction on Neural Networks, Vol. 5, No. 5, pp. 989-993.

- Haykn, S. and Veen, B.V., 2001, “Sinais e Sistemas”, Editora Bookman, Porto Alegre, Brasil.
- Mitchell, J.S., 1993, “Machinery analysis and monitoring”, Penn Well Books, Oklahoma, Estados Unidos.
- Randall, R.B. and Antoni, J., 2011, “Rolling element bearing diagnostic – A tutorial”, Mechanical System and Signal Processing, Vol. 25, pp. 485-520.
- Silva, I.N., Spatti, D.H. and Flauzino, R.A., 2016, “Redes neurais artificiais: para engenharia e ciências aplicadas”, Ed. Artliber Editora, São Paulo, Brasil, 431 p
- Siqueira, P., 2012, “Manutenção centrada na confiabilidade: Manutenção de implementação”, Qualitymark Editora, Rio de Janeiro, Brasil.
- Vasconcelos, N. and Ribeiro, M.P., 2007, “Técnicas de análise de defeito em rolamentos. Técnica tradicional, nova tecnologia e perspectiva de uso na Aço Minas”, AçoMinas, Minas Gerais, Brasil.

8. RESPONSABILIDADE AUTURAL

Os autores são os únicos responsáveis pelo conteúdo deste trabalho.

COMPARATIVE ANALYSIS BETWEEN ARTIFICIAL NEURAL NETWORKS IN A PREDICTIVE MAINTENANCE CASE STUDY

A. G. Alvarenga, andre.guaracy@sga.pucminas.br¹
L. M. Rodrigues, lucas.rodrigues.869763@sga.pucminas.br¹
R. M. da Silva, rafael.marcial@sga.pucminas.br¹
J. L. Junior, janes@pucminas.br²

¹Graduação Pontifícia Universidade Católica de Minas Gerais, Rua Dom José Gaspar, 500, Coração Eucarístico, Belo Horizonte, MG, Brasil.

²Pós-Graduação Pontifícia Universidade Católica de Minas Gerais, Rua Dom José Gaspar, 500, Coração Eucarístico, Belo Horizonte, MG, Brasil.

Abstract. *Between the techniques concerning predictive maintenance, the vibration analysis is the one that shows more contribution when verifying the behavior of a rolling bearing. In the case study of predicting a fail in an bearing utilizing a Perceptron Multilayer Neural Network, where a study on neural networks is developed in order to contribute with the vibration analysis, it was possible to compare the Networks's experimental results with a similar Network offered by MATLAB®. Two types of networks were analyzed, a classificatory network and a function approximation network, both were developed separately by an algorithm developed by the authors in MATLAB® and the other one offered by MATLAB® Neural Net Fitting GUI.*

Keywords: Neural Network, Toolbox, Cross Validation, MATLAB®, Bearings