

**Universidade Federal de Minas Gerais
Escola de Engenharia
Programa de Pós-Graduação em Engenharia Elétrica**

José Hélio de Souza

**Prognóstico de Falhas Utilizando Neo-Fuzzy-Neuron
com Aprendizado on-line**

**Belo Horizonte – MG
Novembro de 2020**

José Hélio de Souza

**Prognóstico de Falhas Utilizando Neo-Fuzzy-Neuron
com Aprendizado on-line**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais, como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica.

Área de Concentração: Sistemas de Computação e Telecomunicações

Linha de Pesquisa: Inteligência Computacional

Orientador: Prof. Dr. Walmir Matos Caminhas

**Belo Horizonte – MG
Novembro de 2020**

S729p

Souza, José Hélio de.

Prognóstico de falhas utilizando Neo-Fuzzy-Neuron com
aprendizado on-line [recurso eletrônico] / José Hélio de Souza. – 2020.
1 recurso online (83 f. : il., color.) : pdf.

Orientador: Walmir Matos Caminhas.

Dissertação (mestrado) Universidade Federal de Minas Gerais,
Escola de Engenharia.

Bibliografia: f. 81-83.

Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia elétrica - Teses. 2. Inteligência computacional - Teses.
3. Análise de séries temporais - Teses. 4. Falha de sistema (Engenharia) -
Teses. I. Caminhas, Walmir Matos. II. Universidade Federal de Minas
Gerais. Escola de Engenharia.
III. Título.

CDU: 621.3(043)

**"Prognóstico de Falhas Utilizando Neo-fuzzy-neuron Com
Aprendizado On-line"**

JOSÉ HÉLIO DE SOUZA

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Mestre em Engenharia Elétrica.

Aprovada em 12 de novembro de 2020.

Por:



**Prof. Dr. Walmir Matos Caminhas
DELT (UFMG) – Orientador**



**Prof. Dr. Marcos Flávio Silveira Vasconcelos D Angelo
DCC (UNIMONTES)**



**Prof. Dr. Fernando Antônio Campos Gomide
DCA (Unicamp)**



**Dr. Marlon Rosa Gouvêa
(Gerdau-Açominas)**

*Dedico este trabalho aos meus pais e aos meus irmãos,
pelo apoio e incentivo constante.*

AGRADECIMENTOS

Agradeço, primeiramente, a Deus pelo o dom da vida, sabedoria, proteção e capacitação.

À minha família, que sempre acreditou em mim e me deu força para alcançar os meus objetivos. Um agradecimento especial a minha irmã, Marilene, que teve papel fundamental em toda a minha trajetória acadêmica.

Ao meu orientador, professor Walmir Matos Caminhas, por todo o apoio, amizade, competência e conhecimentos transmitidos.

Aos professores do Programa de Pós-Graduação em Engenharia Elétrica e do Programa de Pós-graduação em Ciência da Computação, da Universidade Federal de Minas Gerais, pelo aprendizado durante as disciplinas.

Aos amigos do Programa de Pós-Graduação em Engenharia Elétrica, por todo o conhecimento e experiências compartilhadas.

À Fundação de Amparo à Pesquisa do Estado de Minas Gerais, FAPEMIG, pelo suporte financeiro.

*“Lembre-se que as pessoas podem tirar tudo de você,
menos o seu conhecimento.”*
(Albert Einstein)

RESUMO

Os sistemas inteligentes de prognóstico de falhas nas indústrias têm trazido importantes contribuições em termos de segurança e economia, tornando-os indispensáveis, e motivado cada vez mais pesquisas na área. Em problemas de prognóstico o que se busca é prever o tempo de Vida Útil Remanescente (RUL). Em geral, utiliza-se dados históricos e a técnica de previsão de séries temporais, de parâmetros indicativos de degradação, vários passos à frente. Nesse contexto, este trabalho aborda a utilização de uma estrutura *Neurofuzzy* (*Neo-Fuzzy-Neuron*), com aprendizado *online*, para a estimação do RUL. Foi proposta a utilização de pesos intervalares na da rede *Neo-Fuzzy-Neuron*, para realizar a predição de RUL de forma intervalar, e assim obter um RUL com um valor conservador e outro otimista. As metodologias são aplicadas em três bases de dados, bem conhecidas na literatura, para avaliação de problemas de prognóstico, são elas: as baterias de íons de lítio; o desgaste de ferramentas de corte da Máquina de Comando Numérico (CNC); e o desgaste de rolamentos, base de dados PRONOSTIA. Os resultados obtidos mostraram que a metodologia proposta é bastante promissora para prognóstico de falhas em processos/equipamentos.

Palavras-chave: *Neo-Fuzzy-Neuron*, Prognóstico de Falhas, Vida Útil Remanescente, Séries Temporais.

ABSTRACT

Intelligent systems for fault prognostics in industries have brought important contributions in terms of safety and economy, making them indispensable, and motivating more and more research in the area. In prognostics problems, what is sought is to predict the Remaining Useful Life (RUL). In general, historical data and the technique of forecasting time series, of parameters indicative of degradation, multi-step ahead. In this context, this work addresses the use of a Neurofuzzy structure (Neo-Fuzzy-Neuron), with online learning, to estimate the RUL. It was proposed to use interval weights in the Neo-Fuzzy-Neuron network, to perform the prediction of RUL in an interval way, and thus obtain a RUL with a conservative and an optimistic value. The methodologies are applied in three databases, well known in the literature, to assess prognosis problems, they are: lithium ion batteries; wear of cutting tools on the Computer Numeric Control (CNC); and wear of bearings, database PRONOSTIA. The results obtained showed that the proposed methodology is very promising for the fault prognostics in processes/equipment.

Keywords: Neo-Fuzzy-Neuron, Fault Prognostics, Remaining Useful Life, Time Series.

LISTA DE ILUSTRAÇÕES

| | |
|---|----|
| Figura 1.1 – Prognóstico de Falha | 28 |
| Figura 2.1 – Estrutura da <i>Neo-Fuzzy-Neuron</i> | 32 |
| Figura 2.2 – Funções de pertinência triangulares igualmente espaçadas | 33 |
| Figura 2.3 – Funções de pertinência triangulares não igualmente espaçadas | 34 |
| Figura 3.1 – Rede de <i>Kohonen</i> para determinar os centros das funções de pertinência | 37 |
| Figura 3.2 – Conjunto de variáveis | 39 |
| Figura 3.3 – Histograma variável de entrada x_i | 39 |
| Figura 3.4 – Aplicação do ajuste do centro de funções de pertinência com a rede <i>Kohonen</i> | 40 |
| Figura 3.5 – Funções de pertinência geradas para a variável de entrada $x(t)$ | 41 |
| Figura 3.6 – Histograma variável de entrada $x(t)$ | 42 |
| Figura 3.7 – Previsão da série temporal de <i>Mackey-Glass</i> com partição uniforme | 42 |
| Figura 3.8 – Previsão da série temporal de <i>Mackey-Glass</i> com partição ajustada pela rede de <i>Kohonen</i> | 43 |
| Figura 3.9 – Funções de pertinência para a variável de entrada $u(k-4)$ | 44 |
| Figura 3.10 – Funções de pertinência para a variável de entrada $y(k-1)$ | 44 |
| Figura 3.11 – Variáveis de entrada para previsão de CO_2 ($y(k)$) | 45 |
| Figura 3.12 – Histograma para as variáveis de entrada para previsão de CO_2 ($y(k)$) | 45 |
| Figura 3.13 – Previsão da concentração de CO_2 forno a gás de <i>Box-Jenkins</i> com partição uniforme | 46 |
| Figura 3.14 – Previsão da concentração de CO_2 forno a gás de <i>Box-Jenkins</i> com partição ajustada pela rede de <i>Kohonen</i> | 46 |
| Figura 4.1 – Processo de predição da RUL | 49 |
| Figura 4.2 – Plataforma PRONOSTIA | 52 |
| Figura 4.3 – Trajetória cumulativa (<i>atan</i>) dos rolamentos 1, 3 e 5 | 52 |
| Figura 4.4 – Predição da degradação do rolamento 3 da plataforma PRONOSTIA com predição recursiva utilizando os dados brutos da série | 53 |
| Figura 4.5 – Predição da degradação do rolamento 5 da plataforma PRONOSTIA com predição recursiva utilizando os dados brutos da série | 54 |
| Figura 4.6 – Predição da degradação do rolamento 3 da plataforma PRONOSTIA com predição recursiva utilizando a variação em relação ao valor atual | 55 |
| Figura 4.7 – Predição da degradação do rolamento 5 da plataforma PRONOSTIA com predição recursiva utilizando a variação em relação ao valor atual | 55 |
| Figura 4.8 – Máquina CNC de alta velocidade | 56 |

| | |
|---|----|
| Figura 4.9 – Trajetória de desgaste das ferramentas <i>cutter 1</i> , <i>cutter 4</i> e <i>cutter 6</i> | 57 |
| Figura 4.10–Predição de desgaste da ferramenta <i>cutter 1</i> com predição recursiva utilizando os dados brutos da série | 58 |
| Figura 4.11–Predição de desgaste da ferramenta <i>cutter 6</i> com predição recursiva utilizando os dados brutos da série | 58 |
| Figura 4.12–Predição de desgaste da ferramenta <i>cutter 1</i> com predição recursiva utilizando a variação em relação ao valor atual | 59 |
| Figura 4.13–Predição de desgaste da ferramenta <i>cutter 6</i> com predição recursiva utilizando a variação em relação ao valor atual | 60 |
| Figura 4.14–Trajetória de degeneração das bateria B0005, B0006, B0007 e B0018. | 61 |
| Figura 4.15–Predição da degeneração da bateria B0005 com a predição recursiva utilizando os dados brutos da série | 62 |
| Figura 4.16–Predição da degeneração da bateria B0007 com a predição recursiva utilizando os dados brutos da série | 62 |
| Figura 4.17–Predição da degeneração da bateria B0005 com a predição recursiva utilizando a variação em relação ao valor atual | 63 |
| Figura 4.18–Predição da degeneração da bateria B0007 com a predição recursiva utilizando a variação em relação ao valor atual | 64 |
| Figura 5.1 – <i>Neo-Fuzzy-Neuron</i> com peso intervalar | 67 |
| Figura 5.2 – Funções de pertinência triangular com duas funções ativas | 68 |
| Figura 5.3 – Funções de pertinência triangular com três funções ativas | 68 |
| Figura 5.4 – Funções de pertinência triangular com quatro funções ativas | 69 |
| Figura 5.5 – Fluxograma do processo de predição de RUL intervalar com a rede <i>Neo-Fuzzy-Neuron</i> | 71 |
| Figura 5.6 – Predição intervalar da série de degradação $\sigma(atan)$ cumulativa do rolamento 3 | 73 |
| Figura 5.7 – Predição intervalar da série de degradação $\sigma(atan)$ cumulativa do rolamento 5 | 73 |
| Figura 5.8 – Predição intervalar da série de degradação da ferramenta <i>Cutter 1</i> | 75 |
| Figura 5.9 – Predição intervalar da série de degradação da ferramenta <i>Cutter 6</i> | 75 |
| Figura 5.10–Predição intervalar da série de degeneração da bateria B0005 | 77 |
| Figura 5.11–Predição intervalar da série de degeneração da bateria B0007 | 77 |

LISTA DE ALGORITMOS

| | |
|---|----|
| Algoritmo 1 – Rede <i>Neo-Fuzzy-Neuron</i> | 35 |
| Algoritmo 2 – Rede de <i>Kohonen</i> para geração de funções de pertinência | 38 |

LISTA DE TABELAS

| | |
|--|----|
| Tabela 3.1 – Desempenho da rede NFN para a série temporal de <i>Mackey-Glass</i> | 43 |
| Tabela 3.2 – Desempenho da rede na previsão da concentração de CO_2 forno a gás de <i>Box-Jenkins</i> | 47 |
| Tabela 4.1 – Resultados do prognóstico de falhas para a plataforma PRONOSTIA dos rolamentos 3 e 5 com predição recursiva com os dados brutos da série | 54 |
| Tabela 4.2 – Resultados do prognóstico de falhas para a plataforma PRONOSTIA dos rolamentos 3 e 5 com predição recursiva com variação em relação ao valor atual. | 56 |
| Tabela 4.3 – Resultados do prognóstico de falhas para as ferramentas de corte da CNC com predição recursiva utilizando os dados brutos da série | 59 |
| Tabela 4.4 – Resultados do prognóstico de falhas para as ferramentas de corte da CNC com predição recursiva utilizando a variação em relação ao valor atual | 60 |
| Tabela 4.5 – Resultados dos prognóstico de falhas nas baterias B0005 e B007 com predição recursiva utilizando os dados brutos da série | 63 |
| Tabela 4.6 – Resultados dos prognóstico de falhas nas baterias B0005 e B007 com predição recursiva utilizando a variação em relação ao valor atual | 64 |
| Tabela 5.1 – Resultados experimentais da predição de RUL intervalar dos rolamentos 3 e 5 | 74 |
| Tabela 5.2 – Resultados experimentais da predição de RUL intervalar nas ferramentas <i>cutter 1 e cutter 6</i> | 76 |
| Tabela 5.3 – Resultados experimentais da predição de RUL intervalar das baterias B0005 e B0007 | 78 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|--------|---|
| ANFIS | <i>Adaptive Network based Fuzzy Inference System</i> |
| CNC | <i>Máquina de Comando Numérico</i> |
| DE | <i>Diferencial Evolution</i> |
| EBeTS | <i>Error Based Evolving Takagi–Sugeno Fuzzy Model</i> |
| eNFN | <i>Neo-Fuzzy-Neuron evolutiva</i> |
| FALCON | <i>Adaptive Learning Control Network</i> |
| GARIC | <i>Generalized Approximate Reasoning based Intelligence Control</i> |
| IMM | <i>Interacting Multiple Model</i> |
| LSTM | <i>Long short term memory</i> |
| MAPE | <i>Mean Absolute Percentage Error</i> |
| NEFCON | <i>Neuronal Fuzzy Controller</i> |
| NFN | <i>Neo-Fuzzy-Neuron</i> |
| ONFC | <i>Online Neurofuzzy Controller</i> |
| PHM | <i>Prognostics and Health Management</i> |
| PSO | <i>Particle swarm optimization</i> |
| RMSE | <i>Root Mean Square Error</i> |
| RUL | <i>Remaining Useful Life</i> |
| SOH | <i>State of Health</i> |
| SOM | <i>Self-Organizing Maps</i> |
| SW-ELM | <i>Summation Wavelet-Extreme</i> |

LISTA DE SÍMBOLOS

m — Número de funções de pertinência

n — Número de entradas

i — Índice das variáveis de entrada

j — Índice da função de pertinência

μ_A — Função de pertinência

s_i — Saída de um modelo individual da NFN

x_i — Variável de entrada

q_{ij} — Pesos

k_i — Índice para a função de pertinência ativa

y — Saída Real

\hat{y} — Saída estimada

max — Valor máximo

min — Valor mínimo

a_{ij} — Centro das funções de pertinência

Δ — Distância entre os centros das funções de pertinência

r — Índice do centro da função de pertinência

α — Taxa de aprendizagem

β — Valor entre 0 e 1 que define um valor para o α adaptativo

e_t — Erro calculado entre a saída real e saída estimada

w — Vetor de pesos

d_j — Distância do *cluster* j em relação as entradas

j^* — Índice do *Cluster* vencedor

Id_i — Índice de desempenho do neurônio

N — Limiar inteiro para eliminação de funções de pertinência

t_i — Início da predição

t_f — Predição final ao atingir o limiar de falha

p — Quantidade de predições até atingir o limiar de falha

\underline{x}_i — Variável de entrada inferior

\bar{x}_i — Variável de entrada superior

\underline{s}_i — Saída individual para a entrada inferior

\bar{s}_i — Saída individual para a entrada superior

\hat{y} — Saída inferior da NFN com pesos intervalares

$\bar{\hat{y}}$ — Saída superior da NFN com pesos intervalares

\hat{y}^m — Saída média da NFN com pesos intervalares

e_m — Erro médio

α_m — Taxa de aprendizagem média

ρ — Define o tamanho do intervalo na saída da NFN com pesos intervalares

SUMÁRIO

| | | |
|-------|---|----|
| 1 | INTRODUÇÃO | 27 |
| 1.1 | Contexto e Motivação | 27 |
| 1.2 | Objetivos | 30 |
| 1.2.1 | <i>Objetivo Geral</i> | 30 |
| 1.2.2 | <i>Objetivos Específicos</i> | 30 |
| 1.3 | Organização do texto | 30 |
| 2 | REVISITANDO A REDE NEO-FUZZY-NEURON | 31 |
| 2.1 | Estrutura da <i>Neo-Fuzzy-Neuron</i> | 31 |
| 2.2 | Funções de Pertinência | 33 |
| 2.2.1 | <i>Partição Uniforme</i> | 33 |
| 2.2.2 | <i>Partição não Uniforme</i> | 34 |
| 2.3 | Aprendizagem da rede Neo-Fuzzy-Neuron | 34 |
| 2.4 | Resumo | 35 |
| 3 | AJUSTE DE FUNÇÕES DE PERTINÊNCIA COM REDE DE KOHONEN | 37 |
| 3.1 | Rede de Kohonen para geração das funções de pertinência | 37 |
| 3.2 | Resultados de Simulações em Séries Temporais | 40 |
| 3.2.1 | <i>Previsão da série temporal caótica de Mackey-Glass</i> | 40 |
| 3.2.2 | <i>Forno a gás de Box-Jenkins</i> | 43 |
| 3.3 | Resumo | 47 |
| 4 | PROGNÓSTICO DE FALHAS COM A REDE NEO-FUZZY-NEURON | 49 |
| 4.1 | Processo de Estimação da RUL | 49 |
| 4.2 | Método de predição recursiva com os dados brutos | 50 |
| 4.3 | Método de predição recursiva com a variação em relação ao valor atual | 51 |
| 4.4 | Resultados do prognóstico de falhas para a plataforma PRONOSTIA | 51 |
| 4.4.1 | <i>Conjunto de Dados</i> | 51 |
| 4.4.2 | <i>Estimação de RUL do desgaste de rolamentos da plataforma PRONOSTIA utilizando os dados brutos</i> | 53 |
| 4.4.3 | <i>Estimação de RUL do desgaste de rolamentos da plataforma PRONOSTIA utilizando o valor atual e suas variações</i> | 54 |

| | | |
|-------|--|----|
| 4.5 | Resultados do prognóstico de falhas para as ferramentas de Corte da CNC | 56 |
| 4.5.1 | <i>Conjunto de dados</i> | 56 |
| 4.5.2 | <i>Estimação de RUL em ferramentas de corte de uma CNC utilizando os dados brutos</i> | 57 |
| 4.5.3 | <i>Estimação de RUL em ferramentas de corte de uma CNC utilizando o valor atual e suas variações</i> | 59 |
| 4.6 | Resultados do prognóstico de falhas nas baterias de íons de lítio . . | 60 |
| 4.6.1 | <i>Estimação de RUL em baterias de íons de lítio utilizando os dados brutos</i> | 61 |
| 4.6.2 | <i>Estimação de RUL em baterias de íons de lítio utilizando o valor atual e suas variações</i> | 63 |
| 4.7 | Resumo | 64 |
| 5 | PROPOSTA DE PROGNÓSTICO DE FALHA INTERVALAR COM A NEO-FUZZY-NEURON | 67 |
| 5.1 | <i>Neo-Fuzzy-Neuron com Pesos Intervalares</i> | 67 |
| 5.2 | Atualização dos pesos intervalares | 69 |
| 5.2.1 | <i>Predição de RUL intervalar com os pesos intervalares da rede</i> . . . | 70 |
| 5.3 | Resultados Experimentais | 72 |
| 5.3.1 | <i>Predição de RUL intervalar para a plataforma PRONOSTIA</i> | 72 |
| 5.3.2 | <i>Predição de RUL intervalar para a Máquina de Comando Numérico</i> | 74 |
| 5.3.3 | <i>Predição de RUL intervalar para para as baterias de íons de lítio</i> . . | 76 |
| 5.4 | Resumo | 78 |
| 6 | CONCLUSÃO | 79 |
| 6.1 | Trabalhos Futuros | 80 |
| | REFERÊNCIAS | 81 |

INTRODUÇÃO

1.1 Contexto e Motivação

O estudo e desenvolvimento de novas metodologias de sistemas *fuzzy* e redes neurais têm apresentado importantes contribuições para o desenvolvimento da área de inteligência artificial, o que tornou aplicações de sistemas inteligentes cada vez mais comum e indispensáveis. Um avanço significativo foi a combinação da lógica *fuzzy* com as redes neurais, onde se deu origem aos sistemas *Neurofuzzy*. Esta foi uma forma que os pesquisadores encontraram de unir o melhor dos dois métodos. Muitas são as redes *Neurofuzzy*, propostas na literatura, como a *Adaptive Learning Control Network* (FALCON) (LIN; LEE, 1991); a *Adaptive Network based Fuzzy Inference System* (ANFIS) (JANG, 1992); a *Generalized Approximate Reasoning based Intelligence Control* (GARIC) (BERENJI; KHEDKAR, 1992); a *Neuronal Fuzzy Controller* (NEFCON) (NAUCK; KRUSE, 1999).

Uma rede *Neurofuzzy* que tem demonstrado bons resultados em aplicações com aprendizado online é a *Neo-Fuzzy-Neuron* (NFN) (YAMAKAWA *et al.*, 1992). Nessa linha podemos citar os trabalhos: o de Caminhas, Tavares e Gomide (1996) que usam a *Neo-Fuzzy-Neuron* com aprendizagem *online* em detecção de falhas de sistemas dinâmicos; já Landim *et al.* (1998) apresentam um procedimento baseado na NFN para a observação do fluxo do rotor em máquinas de indução com treinamento em tempo real; ainda com o uso da NFN, Gouvêa (2005) propôs o *Online Neurofuzzy Controller* (ONFC), um controlador nebuloso adaptativo de baixo custo computacional; e Silva *et al.* (2014) propuseram uma estrutura evolutiva para a NFN, *Neo-Fuzzy-Neuron* evolutiva (eNFN), capaz de aprender e adaptar sua estrutura de acordo com o fluxo de dados.

Outras importantes aplicações das redes *Neurofuzzy* são previsão de séries temporais e prognóstico de falhas. As previsões de séries temporais envolvem problemas aos quais se deseja prever valores futuros com a observação de dados passados. E o prognóstico de falhas pode ser

tratado como um problema de previsão de série temporal, pois, o objetivo é prever uma possível falha futura ou identificar o tempo de vida restante de um equipamento, de acordo com o seu estado de degradação, que normalmente se comporta como uma série crescente ou decrescente, conforme qual problema está sendo tratado.

O prognóstico de falhas é uma importante ferramenta para as indústrias, tanto em economia como na segurança, pois com o mesmo é possível, por exemplo, prever paradas de máquinas e realizar uma manutenção preventivamente antes que ocorra uma falha em um equipamento, ou seja, a partir da condição atual e dados anteriores é possível estimar quanto tempo de vida tem um certo equipamento. Na literatura esse tipo de prognóstico é denominado vida útil remanescente (do inglês *Remaining Useful Life* - RUL) (JARDINE; LIN; BANJEVIC, 2006).

Os métodos de prognóstico de falhas podem ser divididos em três tipos (LIAO; KÖTTIG, 2014; VACHTSEVANOS *et al.*, 2006), sendo: o prognóstico baseado em modelos (LUO *et al.*, 2003), que se utiliza de modelos matemáticos para prever as condições futuras do equipamento; o prognóstico baseado em experiência (BIAGETTI; SCIUBBA, 2004), que realiza a previsão a partir de modelos probabilístico e experiências passadas; e por último o prognóstico baseado em dados que são métodos que em sua maioria utilizam-se de inteligência computacional.

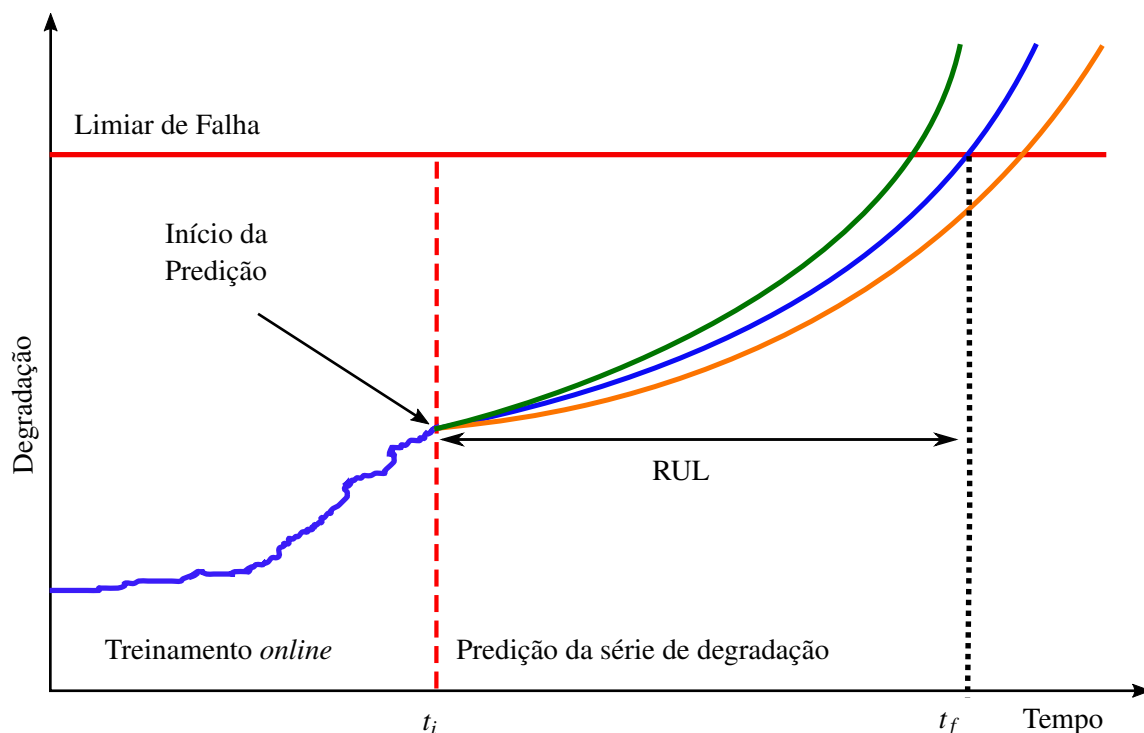


Figura 1.1 – Prognóstico de Falha

Fonte: Adaptada de Soualhi *et al.* (2020).

A Figura 1.1 ilustra um processo de estimativa de RUL baseado em dados, em que a parte inicial da série até um tempo (t_i) é usada para ajustar o modelo que melhor representa

a tendência dos dados, para então iniciar a previsão de quanto tempo resta para a degradação atingir o limiar de falha (y_{t+p}). A diferença entre o tempo em que a série atinge o limiar (t_f) de falha e o tempo inicial da previsão (t_i) é o valor de RUL. No esquema apresentado na figura os dados iniciais são submetidos à aprendizagem *online*, que diferente da aprendizagem *offline* onde os dados de treinamento passam por várias épocas para ajuste do modelo, no aprendizado *online* os dados são apresentados somente uma vez à rede.

O prognóstico baseado em dados e alguns híbridos têm ganhado grande atenção com a realização de diversos estudos e aplicações em problemas comuns da indústria. O trabalho de [Qu et al. \(2019\)](#) propõem uma arquitetura de rede neural de memória de longo prazo (*long short term memory* - LSTM) em conjunto com o algoritmo evolucionário PSO (*Particle swarm optimization*), para a previsão do tempo de vida útil remanescente (RUL) e monitoramento do estado de saúde (*State of Health* - SOH) das baterias de íons de lítio. [Javed et al. \(2018\)](#) usam uma abordagem conexionista orientada a dados (*Summation Wavelet-Extreme* - SW-ELM) com aprendizagem incremental no monitoramento do desgaste da ferramenta de corte de uma Máquina de Comando Numérico (CNC).

Na previsão da degradação de rolamentos [Li, Zhang e Ding \(2019\)](#) propõem uma rede neural profunda e [Cosme et al. \(2019\)](#) usam uma abordagem baseada no filtro IMM (*Interacting Multiple Model*) para o prognóstico de falha. E ainda nas bases de dados de desgaste da ferramenta de corte e rolamentos, [Cosme et al. \(2018\)](#) apresentam uma combinação do filtro de partículas com o algoritmo de evolução diferencial (*Diferencial Evolution* - DE).

[Camargos et al. \(2020\)](#) também apresentam uma abordagem para previsão de RUL, aplicado na base de dados PRONOSTIA. No qual é proposto um modelo *Fuzzy Takagi-Sugeno* evolutivo baseado no erro (EBeTS). O modelo é aplicado para a estimativa de RUL em três cenários: no primeiro é realizado um treinamento prévio do modelo e os dados de testes são utilizados para adaptá-lo em modo *online*. No segundo cenário ocorre somente o treinamento prévio do modelo. E por último apenas os dados de teste estão disponíveis, assim, a estimativa de RUL ocorre sem o treinamento prévio.

Tendo em vista os bons desempenhos relatados com o uso da NFN, este trabalho propõe utilizá-la em problemas de prognóstico de falhas para a previsão de RUL em processos/equipamentos. Onde será realizado a estimação dos parâmetros da NFN em um conjunto de dados com o treinamento *offline*. E com os dados disponíveis do processo/equipamento, referente ao início da degradação, a rede NFN será submetida ao treinamento *online*, para que ela consiga aprender e corrigir a previsão da trajetória da degradação e obter uma melhor estimativa da RUL.

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo deste trabalho é desenvolver sistemas com a rede *Neo-Fuzzy-Neuron* (NFN) para aplicações em Prognóstico de Falhas.

1.2.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Avaliar o desempenho da *Neo-Fuzzy-Neuron* (NFN) com funções de pertinência triangulares ajustas por *Kohonen* em relação as igualmente espaçadas, com aplicações práticas em séries temporais.
- Com o uso da rede *Neo-Fuzzy-Neuron* (NFN), propor e avaliar métodos de análise de dados históricos para estimar o tempo de vida útil remanescente (*Remaining Useful Life - RUL*) em problemas de prognóstico de falhas.
- Desenvolver uma estrutura para a *Neo-Fuzzy-Neuron* com pesos intervalares, para realizar uma estimativa de RUL intervalar com um valor inferior (conservador) e um valor superior (otimista).

1.3 Organização do texto

Os capítulos seguintes da dissertação foram organizados da seguinte forma:

- **Capítulo 2:** Apresenta uma revisão da rede *Neo-Fuzzy-Neuron* (NFN), sua estrutura, funções de pertinência e método aprendizagem.
- **Capítulo 3:** Realiza a aplicação da rede de *Kohonen* na atualização das funções de pertinência triangulares da *Neo-Fuzzy-Neuron* (NFN). Também são apresentados os resultados numéricos de simulações em dois conjunto de dados de séries temporais.
- **Capítulo 4:** Descreve e apresenta resultados de experimentos da NFN em três bases de dados de problemas de prognóstico: a degeneração de baterias de íons de lítio; o desgaste das ferramentas de corte de uma máquina de comando numérico (CNC) e o desgaste de rolamentos extraídos da plataforma PRONOSTIA.
- **Capítulo 5:** Apresenta a *Neo-Fuzzy-Neuron* com pesos intervalares proposta para estimação do tempo vida útil remanescente intervalar. São realizados experimentos e simulações com os mesmos problemas apresentados no capítulo 4.
- **Capítulo 6:** Apresenta as considerações finais e as propostas de trabalhos futuros.

REVISITANDO A REDE NEO-FUZZY-NEURON

Este capítulo apresenta uma revisão bibliográfica da estrutura e método de aprendizado da rede *Neo-Fuzzy-Neuron* (NFN), além disso, demonstra a geração das funções de pertinência do tipo triangulares com espaçamento uniforme e não uniforme utilizadas na NFN.

2.1 Estrutura da *Neo-Fuzzy-Neuron*

A *Neo-Fuzzy-Neuron* (NFN) é uma estrutura de rede *Neurofuzzy* proposta por [Yamakawa et al. \(1992\)](#) que é conhecida pela sua simplicidade de implementação e seu baixo custo computacional. A rede é composta por n modelos desacoplados, tipo *Takagi-Sugeno* de ordem zero, sendo um para cada entrada, e a saída da NFN é a soma das saídas individuais dos modelos ([SILVA et al., 2014](#)).

A Figura 2.1 ilustra a estrutura da NFN com n entradas, $[x_1, x_2, \dots, x_i, \dots, x_n]$, indexadas por i e uma saída \hat{y} . Para cada entrada tem-se um modelo, tipo *Takagi-Sugeno* de ordem zero, composto por m funções de pertinência ($\mu_{A_{ij}}$) e um peso (q_{ij}) associado a ela, em que j é o índice da função de pertinência, tendo como saída de cada modelo:

$$s_i = \mu_{A_{i1}}(x_i)q_{i1} + \dots + \mu_{A_{ij}}(x_i)q_{ij} + \dots + \mu_{A_{im}}(x_i)q_{im}, \quad (2.1)$$

simplificando:

$$s_i = \sum_{j=1}^m \mu_{A_{ij}}(x_i)q_{ij}. \quad (2.2)$$

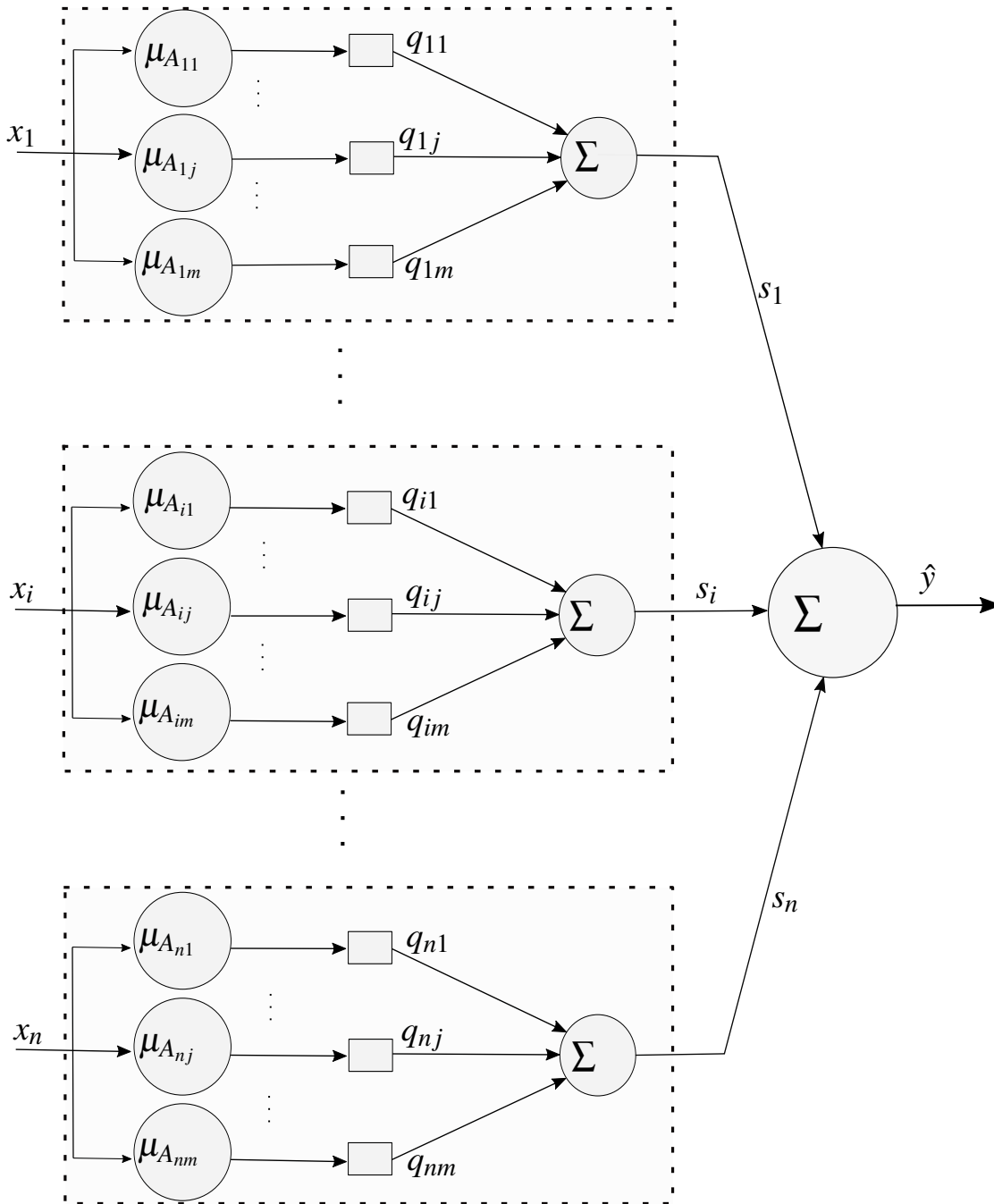


Figura 2.1 – Estrutura da Neo-Fuzzy-Neuron

Fonte: Adaptada de [Silva et al. \(2014\)](#).

As funções de pertinência mais comumente usadas na estrutura da NFN são funções de pertinência triangulares complementares ([YAMAKAWA et al., 1993](#); [CAMINHAS; GOMIDE, 2000](#); [SILVA et al., 2014](#)), desta forma, terá no máximo duas funções ativas para cada entrada x_i , indexadas por $(k_i$ e $k_i + 1)$:

$$s_i = \mu_{A_{ik_i}}(x_i)q_{ik_i} + \mu_{A_{ik_i+1}}(x_i)q_{ik_i+1}. \quad (2.3)$$

A saída da rede (\hat{y}) é o somatório das saídas individuais dos modelos:

$$\hat{y} = \sum_{i=1}^n s_i. \quad (2.4)$$

2.2 Funções de Pertinência

2.2.1 Partição Uniforme

As funções de pertinência triangular uniforme são geradas a partir da quantidade de regras m_i desejada e do valor mínimo e máximo de x_i , sendo $\min(x_i)$ o primeiro valor modal (a_{i1}) e os demais são definidos por:

$$\Delta_i = \frac{\max(x_i) - \min(x_i)}{m - 1}, \quad (2.5)$$

$$a_{ij} = \min(x_i) + (j - 1)\Delta_i, \quad (2.6)$$

onde j é o índice da função pertinência e Δ_i o espaço entre elas, que as mantêm igualmente espaçadas, como ilustra a Figura 2.2.

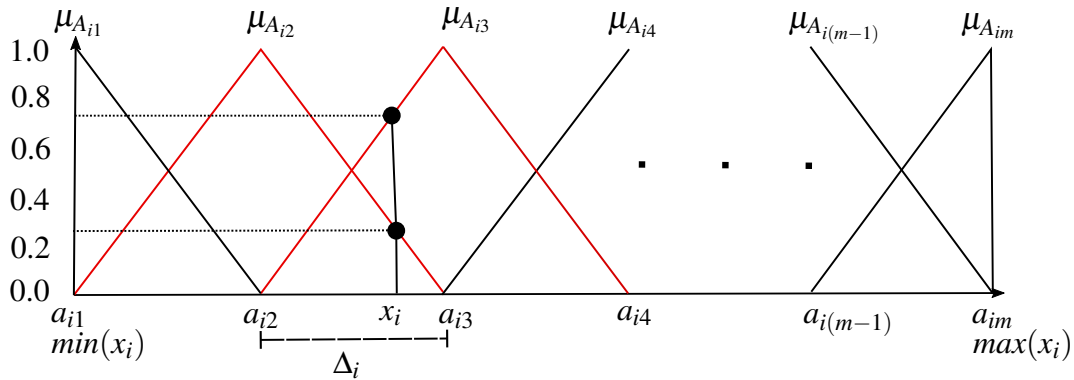


Figura 2.2 – Funções de pertinência triangulares igualmente espaçadas

A equação para encontrar o valor da função de pertinência da entrada x_i é dada por:

$$\mu_{A_{ij}(x_i)} = \begin{cases} \frac{x_i - a_{ir}}{\Delta_i} + 1, & a_{i(r-1)} \leq x_i \leq a_{ir} \\ -\frac{x_i - a_{ir}}{\Delta_i} + 1, & a_{ir} < x_i \leq a_{i(r+1)} \\ 0, & \text{Outro caso} \end{cases} \quad (2.7)$$

em que r é o índice da função de pertinência que está sendo calculada, $r - 1$ e $r + 1$, são as funções anterior e posterior respectivamente.

2.2.2 Partição não Uniforme

As funções de pertinência triangulares também podem ser complementares sem necessariamente está igualmente espaçadas como ilustra a Figura 2.3, nesse caso, seus valores modais são gerados a partir de seu conjunto de dados, normalmente, verifica-se onde se encontra as maiores concentrações de dados para defini-los como o valor modal das funções, um exemplo, é o uso de algoritmos não supervisionados como demonstrado em (CAMINHAS *et al.*, 1999) que será tema do próximo capítulo.

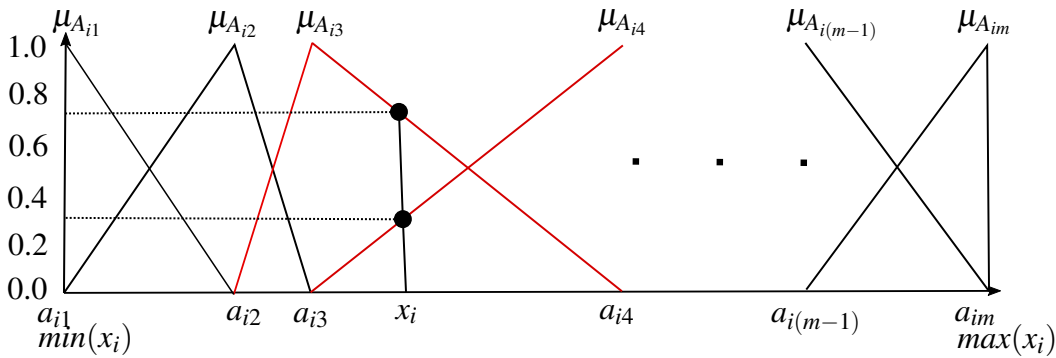


Figura 2.3 – Funções de pertinência triangulares não igualmente espaçadas

A equação para encontrar o valor da função de pertinência em relação a cada entrada x_i é dada por:

$$\mu_{A_{ij}}(x_i) = \begin{cases} \frac{x_i - a_{ir}}{a_{ir} - a_{ir-1}} + 1, & a_{ir-1} \leq x_i \leq a_{ir} \\ \frac{x_i - a_{ir}}{a_{ir} - a_{ir+1}} + 1, & a_{ir} < x_i \leq a_{ir+1} \\ 0, & \text{Outro caso} \end{cases} \quad (2.8)$$

2.3 Aprendizagem da rede Neo-Fuzzy-Neuron

O aprendizado da rede NFN é realizado a partir da atualização dos pesos (q_{ij}), de forma a encontrar os valores deste parâmetro que melhor representa o conjunto de dados.

$$E_t = \frac{1}{2}(\hat{y}_t - y_t)^2 \quad (2.9)$$

Para atualizar os pesos é realizado a minimização da função de erro, Equação 2.9, utilizando o método do gradiente, desta forma, tem-se que:

$$q_{ij}^{Atual} = q_{ij}^{Anterior} + \Delta q_{ij}, \quad (2.10)$$

$$\Delta q_{ij} = \alpha \frac{\partial E_t}{\partial q_{ij}} = -\alpha(\hat{y}_t - y_t)\mu_{A_{ij}}(x_{ti}), \quad (2.11)$$

onde α é a taxa de aprendizagem.

Como visto na seção 2.2, usando funções de pertinência triangular a cada dado de entrada somente duas funções são ativadas, desta forma apenas dois pesos são atualizados por vez:

$$\begin{aligned} q_{ik_i}^{Atual} &= q_{ik_i}^{Anterior} - \alpha(\hat{y}_t - y_t)\mu_{A_{ik_i}}(x_{ti}), \\ q_{ik_{i+1}}^{Atual} &= q_{ik_{i+1}}^{Anterior} - \alpha(\hat{y}_t - y_t)\mu_{A_{ik_{i+1}}}(x_{ti}). \end{aligned} \quad (2.12)$$

O valor da taxa de aprendizagem (α) pode ser um valor fixo ou adaptativo como proposto por (CAMINHAS; GOMIDE, 2000):

$$\alpha = \frac{\beta}{\sum_{i=1}^n \mu_{A_{ik}}(x_{ti})^2 + \mu_{A_{ik+1}}(x_{ti})^2}, \quad (2.13)$$

quando $\beta = 1$, a taxa de aprendizagem proporciona erro nulo para cada amostra no instante t calculado, $e_t = (\hat{y}_t - y_t) = 0$.

O Algoritmo 1 detalha o processo de estimação da saída e atualização dos pesos da NFN.

Algoritmo 1 – Rede Neo-Fuzzy-Neuron

- 1: **Entradas:** x_t, y_t, n ;
 - 2: **Saída:** \hat{y}_t ;
 - 3: Inicializar a_{ij} ; // centro das funções de pertinência
 - 4: **para** $t = 1, 2, 3, \dots$ **faça**
 - 5: Ler x_t, y_t ;
 - 6: **para** $i = 1 : n$ **faça**
 - 7: Calcula $\mu_{A_{ik}}(x_{ti}), \mu_{A_{ik+1}}(x_{ti})$;
 - 8: Calcula s_{ti} ;
 - 9: **fim para**
 - 10: Calcula \hat{y}_t ;
 - 11: Calcula α ;
 - 12: Atualiza q_{ik}, q_{ik+1} ;
 - 13: **fim para**
-

2.4 Resumo

Foi realizada uma revisão da NFN onde foi apresentada sua estrutura, os dois tipos de funções de pertinência triangulares, as uniforme e não uniforme, e como elas são geradas, além de mostrar as partições com bordas trapezoidais, para evitar um valor nulo quando algum dado de entrada cair fora da área factível. Também foi apresentado o método de aprendizagem e como é calculado o α adaptativo, e por último é detalhado o algoritmo da NFN.

As características de baixa complexidade e uma taxa de aprendizagem que proporciona um erro nulo, faz da NFN uma ótima opção para aplicações *online*.

AJUSTE DE FUNÇÕES DE PERTINÊNCIA COM REDE DE KOHONEN

Neste capítulo será demonstrado a utilização da rede de *Kohonen* no ajuste das funções de pertinência triangulares com centros não uniformemente espaçados. E será feita uma comparação do desempenho dessas funções ajustadas por *Kohonen*, na rede *Neo-Fuzzy-Neuron* (NFN), das funções de pertinência triangulares uniforme. O desempenho é avaliado em dois problemas de séries temporais pelas métricas de RMSE e MAPE.

3.1 Rede de Kohonen para geração das funções de pertinência

A rede neural auto-organizada de *Kohonen* é utilizada para geração das funções de pertinência não uniformemente espaçadas como proposto em [Caminhas et al. \(1997\)](#).

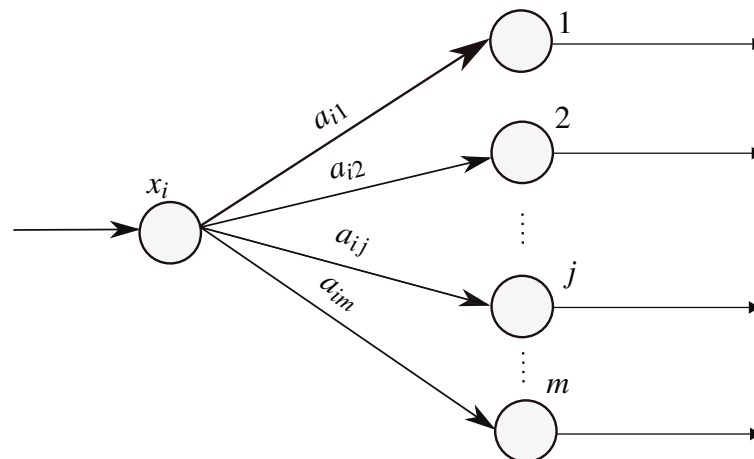


Figura 3.1 – Rede de *Kohonen* para determinar os centros das funções de pertinência

Fonte: Adaptada de [Caminhas et al. \(1997\)](#).

Para cada padrão de entrada x_i é utilizado uma estrutura conforme a Figura 3.1, no qual os pesos da rede, $[a_{i1}, a_{i2}, \dots, a_{ij}, \dots, a_{im}]$, correspondem aos centros das funções de pertinência e j é o índice de um *cluster*. O processo de treinamento da rede é por competição, e o *cluster* (j) com a menor distância do padrão x_i , Equação 3.1, é escolhido como vencedor para atualização, conforme a Equação 3.2.

$$j^* = \arg\{\min_j |x_i - a_{ij}|\}. \quad (3.1)$$

$$a_{ij^*}^{Atual} = a_{ij^*}^{Anterior} + \alpha(x_i - a_{ij^*}^{Anterior}). \quad (3.2)$$

Os centros das funções de pertinência inicializadas de forma igualmente espaçadas são usados como centros dos *clusters* da rede de *Kohonen* que atualiza os vencedores e elimina aqueles que tiverem um baixo índice de desempenho ($Id_i(j)$), ou seja, que venceram poucas vezes, menor que um limiar inteiro (N) (CAMINHAS *et al.*, 1997).

O Algoritmo 2 ilustra o processo de geração das funções de pertinência com a rede de *Kohonen* para um padrão de entrada x_i .

Algoritmo 2 – Rede de *Kohonen* para geração de funções de pertinência

- 1: **Entradas:** x_t, m ;
 - 2: $a_{i1} = \min(x_i)$;
 - 3: $\Delta_i = \frac{\max(x_i) - \min(x_i)}{m-1}$;
 - 4: $a_{ij} = a_{i(j-1)} + \Delta_i$; para $j=2,3,\dots,m$;
 - 5: $Id_i(j) = 0$; para $j=1,2,\dots,m$
 - 6: **para** $t = 1, 2, 3, \dots$ **faça**
 - 7: Ler x_t ;
 - 8: $j^* = \arg\{\min_j |x_i - a_{ij}|\}$
 - 9: $a_{ij^*}^{Atual} = a_{ij^*}^{Anterior} + \alpha(x_i - a_{ij^*}^{Anterior})$;
 - 10: $Id_i(j^*) = Id_i(j^*) + 1$;
 - 11: **fim para**
 - 12: **para** $j = 1 : m$ **faça**
 - 13: **if** $Id_i(j) \leq N$
 - 14: Elimina Neurônio j ;
 - 15: **fim para**
-

Um exemplo do processo de eliminação e ajuste dos centros das funções de pertinência, é aplicado em um conjunto de dados visivelmente espaçados em dois *clusters* (Figura 3.2), criados com centro em $[-1, -1]$ e o outro com centro em $[1, 1]$.

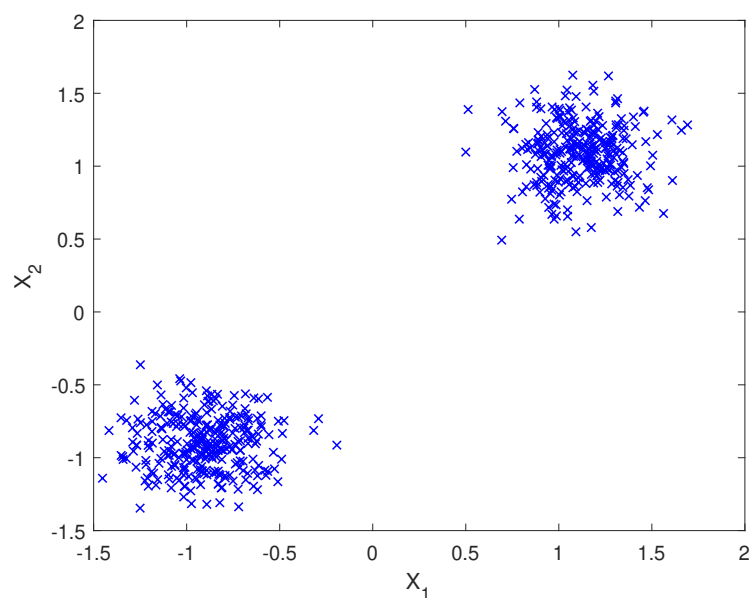
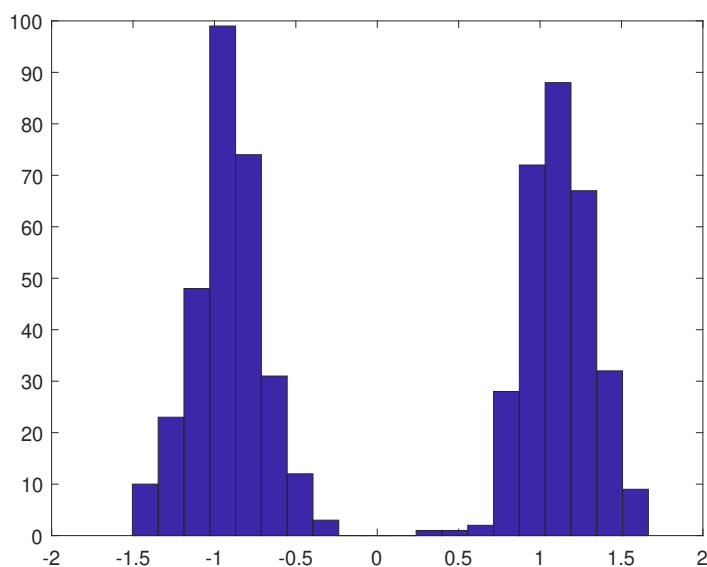


Figura 3.2 – Conjunto de variáveis

Na Figura 3.3 é possível observar a distribuição da variável x_i e a alta concentração de dados em torno dos valores -1 e 1.

Figura 3.3 – Histograma variável de entrada x_i

Ao aplicar a Rede de Kohonen para geração das funções de pertinência para a variável x_i , onde na Figura 3.4a temos os centros iniciais uniforme das funções de pertinência e na Figura 3.4b os centros resultantes após o fim do processo de eliminação e ajuste dos centros. Nota-se que o algoritmo conseguiu identificar os dois centros que melhor representa o conjunto de dados para ser os centros das funções de pertinência, desta forma das 10 funções iniciais restaram

apenas quatro, sendo as duas das extremidades com centros no valor mínimo e no valor máximo e as duas com valor modal nos centros dos *clusters*.

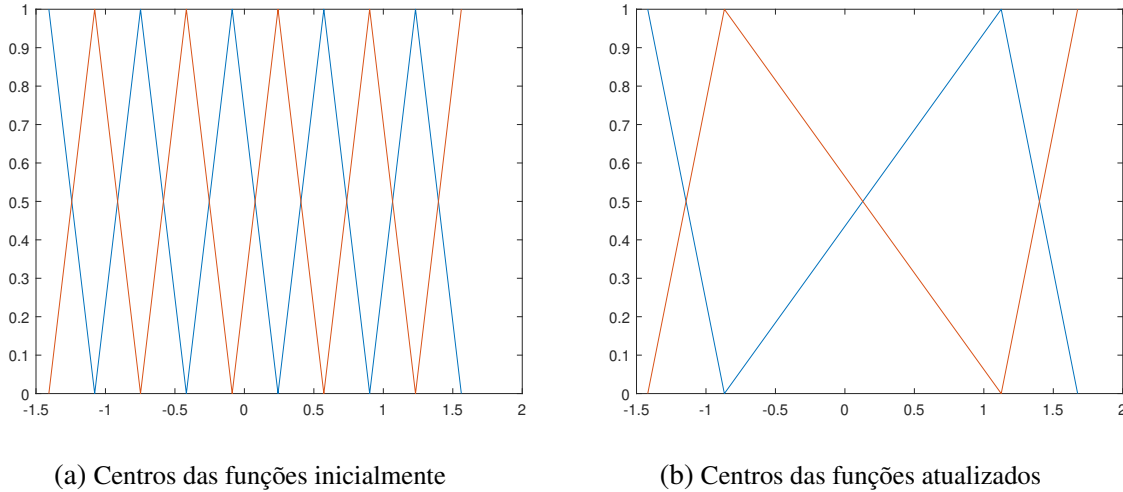


Figura 3.4 – Aplicação do ajuste do centro de funções de pertinência com a rede *Kohonen*

3.2 Resultados de Simulações em Séries Temporais

Para avaliação do desempenho da rede NFN com e sem o ajuste das funções de pertinência por *kohonen*, foi utilizado dois problemas clássicos, previsão da série temporal caótica de *Mackey-Glass* e identificação do forno a gás de *Box-Jenkins*.

As métricas de desempenhos utilizadas foram a raiz quadrática do erro médio (*Root Mean Square Error* - RMSE) e a média percentual absoluta do erro (*Mean Absolute Percentage Error* - MAPE):

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \hat{y}_t)^2}, \quad (3.3)$$

$$\text{MAPE} = \frac{1}{N} \sum_{t=1}^N \frac{|y_t - \hat{y}_t|}{y_t} \times 100, \quad (3.4)$$

onde N é o número de amostra de dados utilizado, y_t é a saída estimada e \hat{y}_t é a saída desejada.

3.2.1 Previsão da série temporal caótica de *Mackey-Glass*

A série temporal de *Mackey-Glass* é gerada pela seguinte expressão:

$$\frac{dx(t)}{dt} = \frac{0.2x(t - \tau)}{1 + x(t - \tau)^{10}} - 0.1x(t) \quad (3.5)$$

com $x(0) = 1.2$ e $\tau = 17$, foi constituído o conjunto de dados para este experimento. O objetivo deste problema é prevê 6 unidade de tempo a frente em função do valor atual, 6, 12 e 18 unidades tempo passado:

$$x(t + 6) = f(x(t - 18), x(t - 12), x(t - 6), x(t)). \quad (3.6)$$

Um total de 1171 amostras foram utilizadas, sendo 781 separas para treinamento da rede e 390 para validação. A quantidade de funções de pertinência utilizada foi a quantidade retornada pelo algoritmo com atualização e eliminação de funções. Sendo assim, a comparação dos dois resultados da NFN, com e sem o ajuste, utilizaram o mesmo número de funções de pertinência no treinamento.

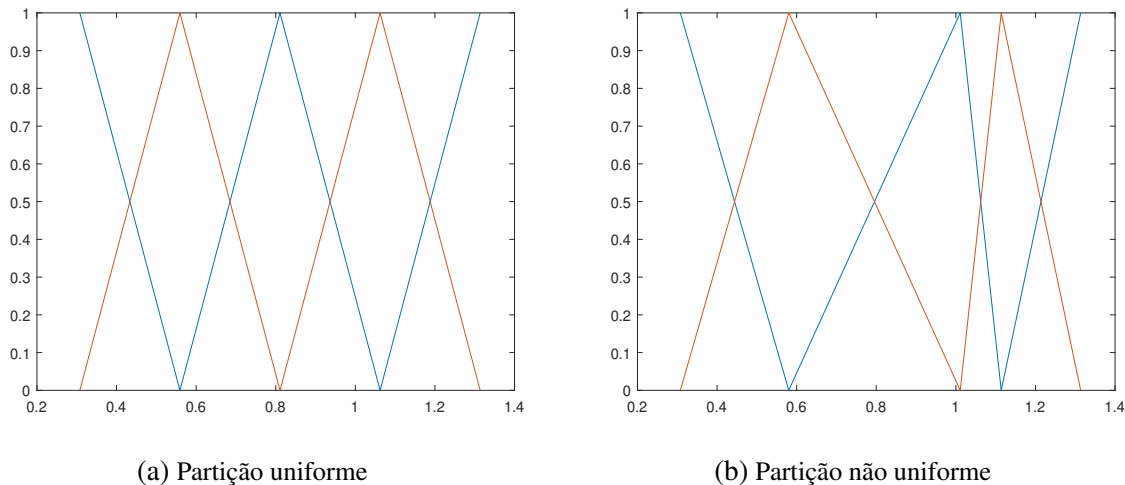
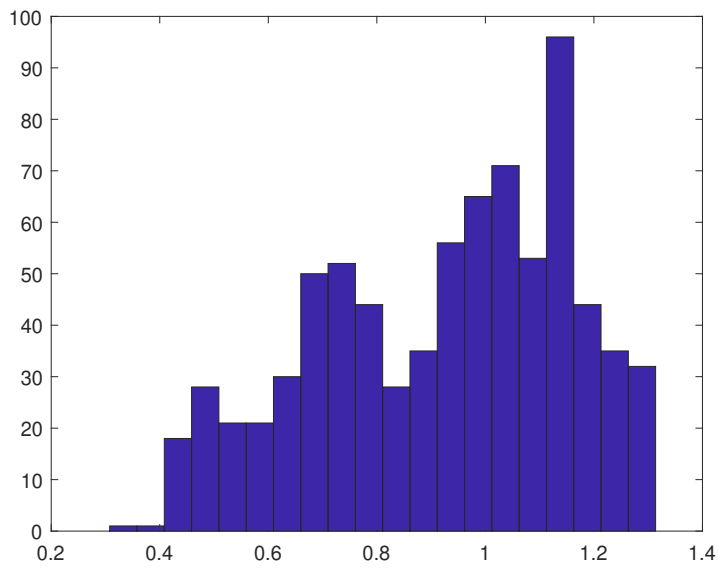
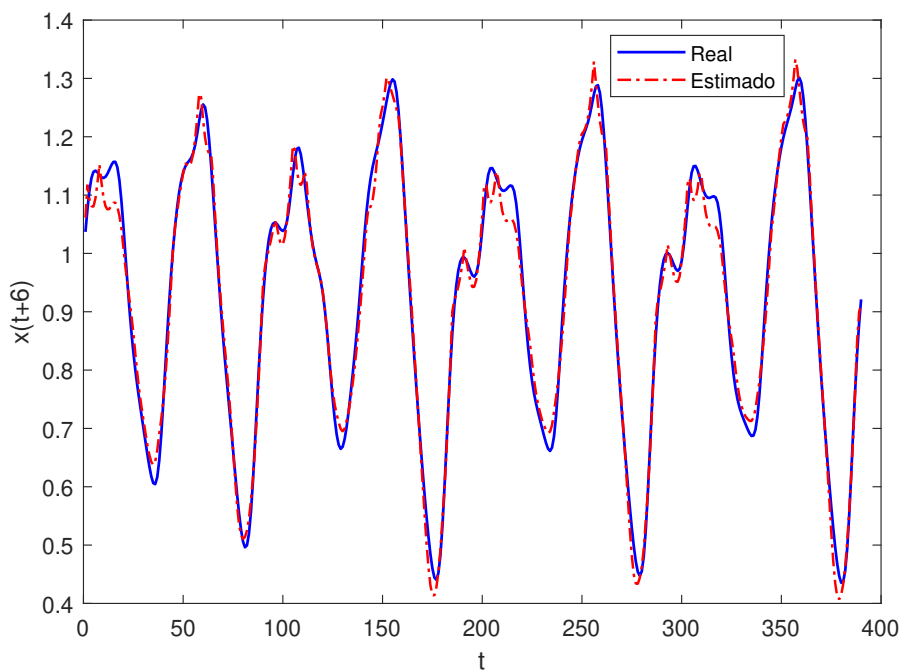


Figura 3.5 – Funções de pertinência geradas para a variável de entrada $x(t)$

Analisando a Figura 3.5, percebe-se que, diferente da partição uniforme, Figura 3.5a, as funções de pertinência geradas pelo algoritmo de *Kohonen*, Figura 3.5b, o centro da terceira função de pertinência ficou distante da segunda e mais próximo da quarta, o que indica uma maior concentração de dados naquele ponto, que pode ser confirmado pelo histograma apresentado na Figura 3.6.

Figura 3.6 – Histograma variável de entrada $x(t)$

Ao analisar os gráficos das Figuras 3.7 e 3.8, percebe-se que, utilizando as funções geradas por *Kohonen*, teve uma melhora na previsão de $x(t+6)$ com valores próximos de 1.1.

Figura 3.7 – Previsão da série temporal de *Mackey-Glass* com partição uniforme

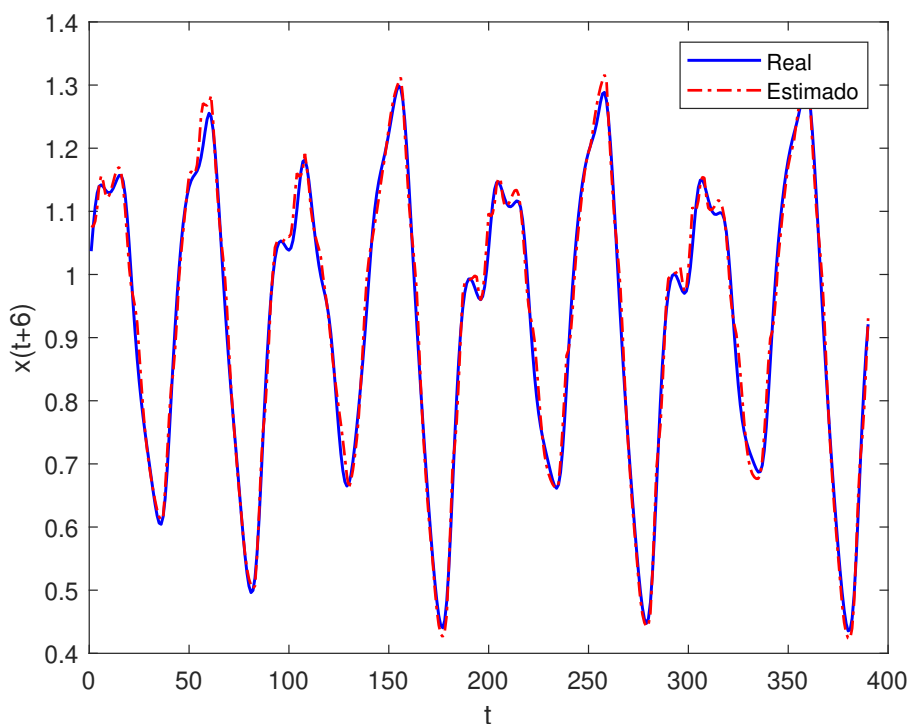


Figura 3.8 – Previsão da série temporal de *Mackey-Glass* com partição ajustada pela rede de *Kohonen*

Na Tabela 3.1 são apresentados os resultados de RMSE e MAPE da previsão, onde são apresentados o melhor resultado de 10 execuções e o valor médio de todas.

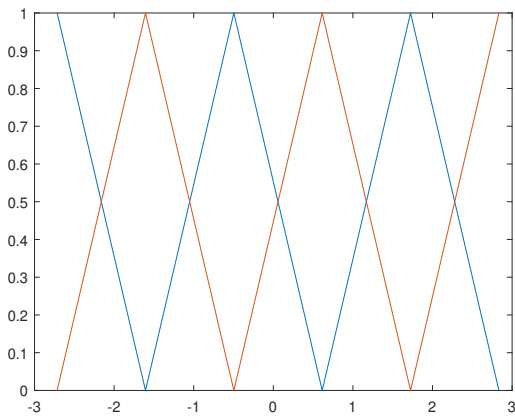
Tabela 3.1 – Desempenho da rede NFN para a série temporal de *Mackey-Glass*

| Modelo da NFN | Número de funções | Melhor RMSE | RMSE Médio | Melhor MAPE(%) | MAPE(%) Médio |
|---|-------------------|-------------|------------|----------------|---------------|
| Funções de pertinência partição uniforme | 5 | 0.0215 | 0.0235 | 2.5025 | 2.5317 |
| Funções de pertinência gerada pela rede de <i>Kohonen</i> | 5 | 0.0155 | 0.01975 | 1.7745 | 1.9987 |

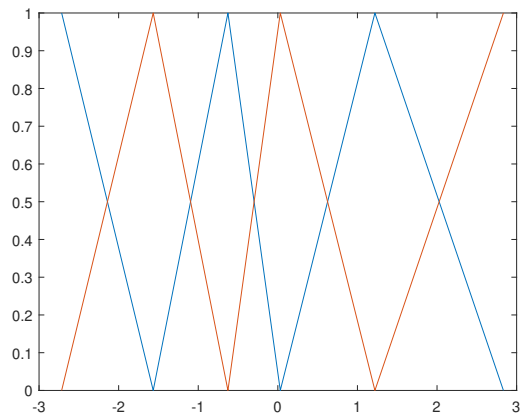
3.2.2 Forno a gás de *Box-Jenkins*

O conjunto de dados do forno a gás de *Box-Jenkins* é composto de 290 pares. O objetivo é prever a concentração de CO_2 ($y(k)$) em função da vazão de gás ($u(k)$). [Box et al. \(2015\)](#) representam esse problema como sendo:

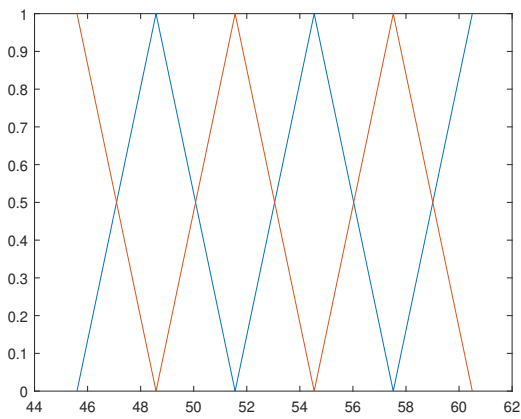
$$y(k) = f(u(k-4), y(k-1)). \quad (3.7)$$



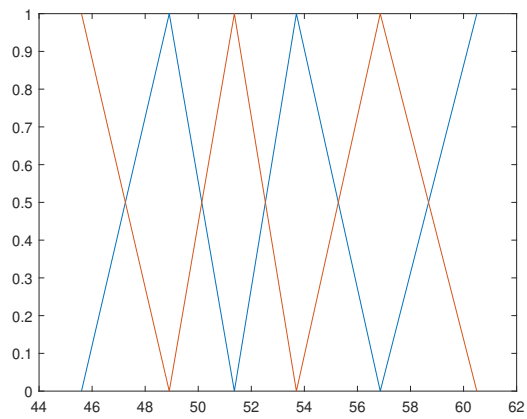
(a) Partição uniforme



(b) Partição não uniforme

Figura 3.9 – Funções de pertinência para a variável de entrada $u(k-4)$ 

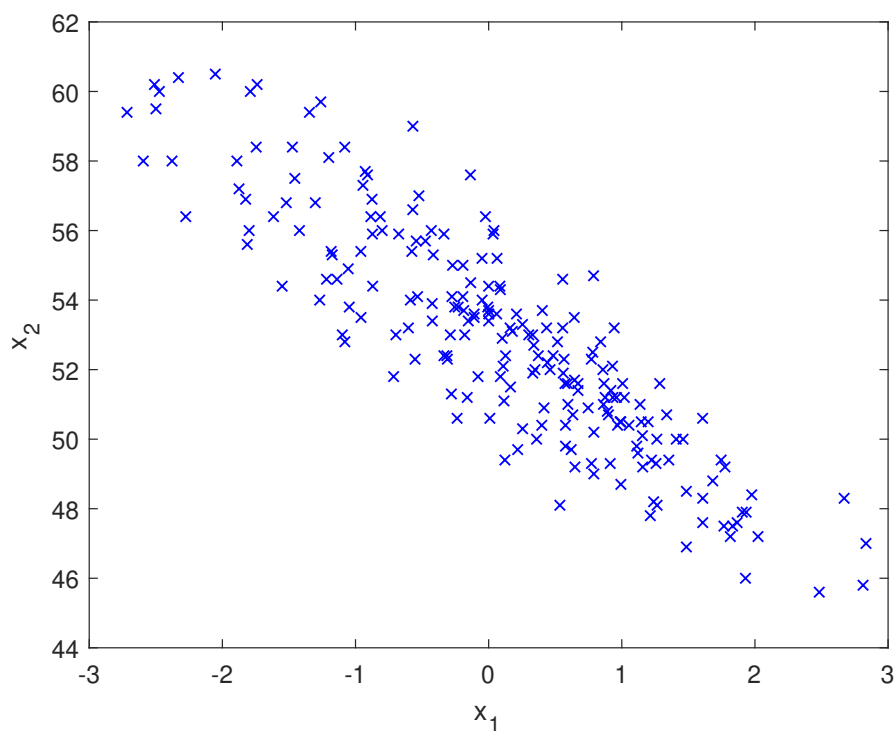
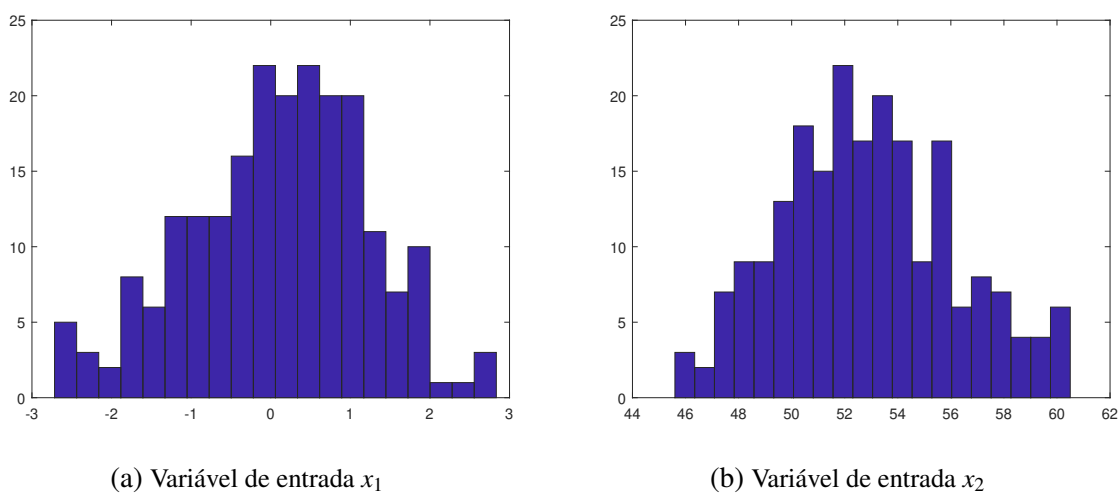
(a) Partição uniforme



(b) Partição não uniforme

Figura 3.10 – Funções de pertinência para a variável de entrada $y(k-1)$

As funções pertinência geradas pelo algoritmo de *Kohonen* (Figura 3.9b e 3.10b) teve pouca variação nos centros, em relação às igualmente espaçadas (Figura 3.9a e 3.10a) isso acontece devido a distribuição das variáveis de entrada, onde possui concentração de dados ao longo de toda área factível, como pode ser notado no gráfico da Figura 3.11 e nos histogramas das variáveis na Figura 3.12.

Figura 3.11 – Variáveis de entrada para previsão de CO_2 ($y(k)$)(a) Variável de entrada x_1 (b) Variável de entrada x_2 Figura 3.12 – Histograma para as variáveis de entrada para previsão de CO_2 ($y(k)$)

Assim como foi realizado no problema anterior, nas Figuras 3.13 e 3.14 são apresentados os gráficos dos valores previstos e na Tabela 3.2 os resultados de RMSE e MAPE, no qual temos o melhor resultado de 10 execuções e o valor médio delas.

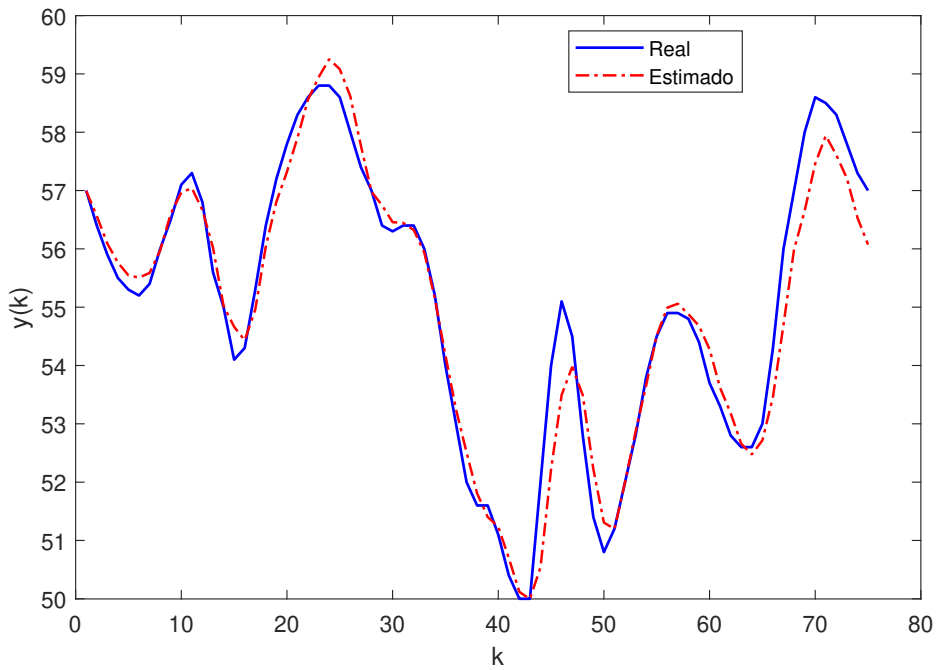


Figura 3.13 – Previsão da concentração de CO_2 forno a gás de *Box-Jenkins* com partição uniforme

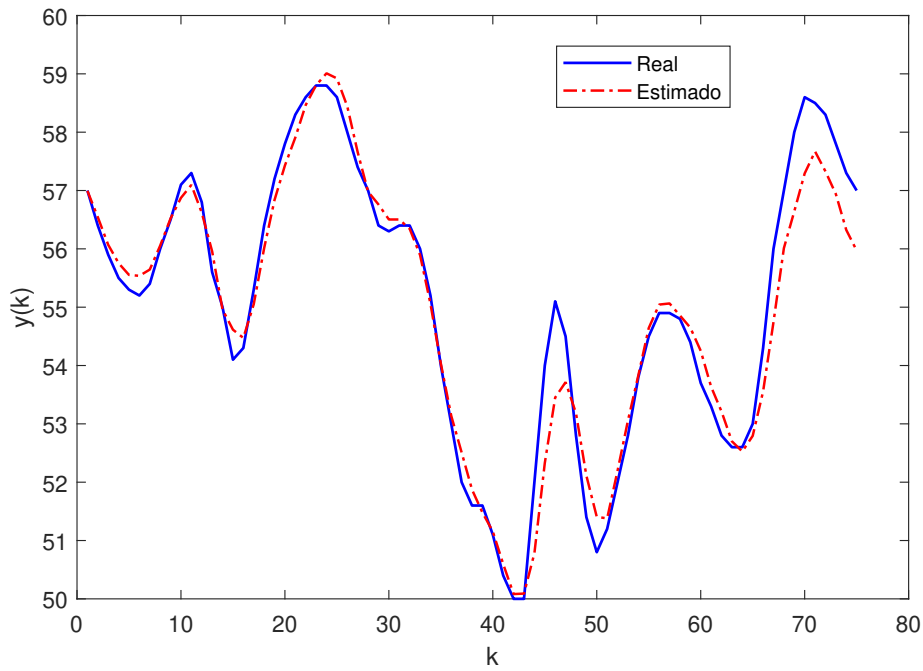


Figura 3.14 – Previsão da concentração de CO_2 forno a gás de *Box-Jenkins* com partição ajustada pela rede de Kohonen

Tabela 3.2 – Desempenho da rede na previsão da concentração de CO_2 forno a gás de *Box-Jenkins*

| Modelo da NFN | Número de funções | Melhor RMSE | RMSE Médio | Melhor MAPE(%) | MAPE(%) Médio |
|--|-------------------|-------------|------------|----------------|---------------|
| Funções de pertinência partição uniforme | 6 | 0.35 | 0.37 | 0.60 | 0.65 |
| Funções de pertinência atualizadas pela rede de <i>Kohonen</i> | 6 | 0.30 | 0.34 | 0.56 | 0.62 |

3.3 Resumo

O algoritmo de *Kohonen* para atualização e eliminação de funções de pertinência da NFN, apresentou resultados satisfatórios conseguindo encontrar os centros principais do conjunto de dados para serem os valores modais das funções de pertinência.

Analisando os experimentos e seus resultados gráficos de previsão foi possível observar um melhor resultado na previsão da série caótica de *Mackey-Glass*. Já na série do forno a gás de *Box-Jenkins* os resultados gráficos foi quase imperceptível, entanto, ao analisar os valores de RMSE e MAPE, nota-se que o resultado melhorou utilizando as funções geradas com a *Kohonen*.

PROGNÓSTICO DE FALHAS COM A REDE NEO-FUZZY-NEURON

Neste capítulo, serão abordadas duas técnicas para a previsão de vários passos a frente recursiva com a rede *Neo-Fuzzy-Neuron* (NFN), para estimar o tempo de vida útil remanescente, uma com os dados brutos da série e outra com a variação em relação ao valor atual. Também serão apresentados três conjuntos de dados, de problemas de prognóstico de falhas, onde são realizados dois experimentos para cada método.

4.1 Processo de Estimação da RUL

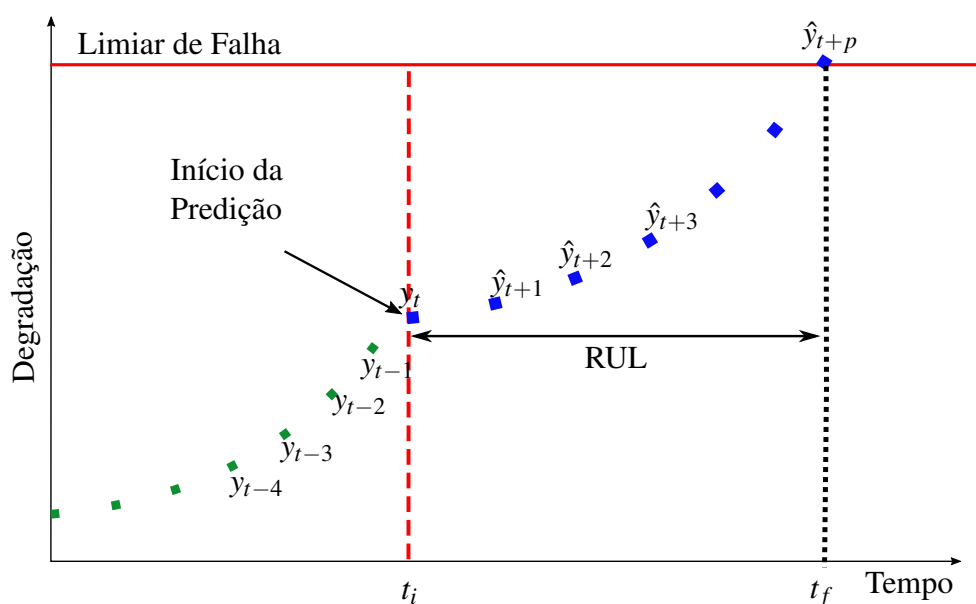


Figura 4.1 – Processo de predição da RUL

Fonte: Adaptada de Soualhi *et al.* (2020).

O processo de estimação do tempo de vida útil remanescente (*Remaining Useful Life - RUL*) apresentado neste trabalho acontece em três fases: treinamento *offline*; treinamento *online* e predição da RUL. A Figura 4.1 ilustra como foi realizado o processo de predição da RUL, a parte inicial da série até o instante (t_i) é usada para ajustar o modelo que melhor representa a tendência dos dados (treinamento *online*), e então, inicia-se a previsão de quanto tempo resta para o estado de degradação atingir o limiar de falha (\hat{y}_{t+p}). A diferença entre o tempo em que a série atinge o limiar de falha (t_f) e o tempo inicial da predição (t_i) é o RUL previsto.

4.2 Método de predição recursiva com os dados brutos

Uma característica das séries de degradação ou desgaste é ser crescente ou decrescente, a depender dos dados a serem analisados. E uma das formas de estimar a RUL, é usar a estratégia recursiva (SORJAMAA; LENDASSE, 2006) na predição dos novos valores da série, observando os valores previstos em instantes anteriores, quando estes não estiverem disponíveis, de forma a estimar essa evolução da degradação com o uso dos valores previstos, até chegar no instante de tempo em que atinge o limiar de falha. A equação:

$$\hat{y}_t = f(y_{t-1}, y_{t-2}, y_{t-3}, \dots, y_{t-w}), \quad (4.1)$$

demonstra a predição de um valor futuro \hat{y}_t , utilizando os dados passados como sendo as *features* da rede, a quantidade de *features* utilizadas é determinada pelo tamanho da janela (w). A função f representa a rede que realiza o mapeamento de entrada e saída dos dados.

A Equação 4.2 detalha o processo de estimação dos valores até atingir o limiar de falha (\hat{y}_{t+p}).

$$\begin{aligned} \hat{y}_{t+1} &= f(y_t, y_{t-1}, y_{t-2}, y_{t-3}, \dots, y_{t-(w-1)}), \\ \hat{y}_{t+2} &= f(\hat{y}_{t+1}, y_t, y_{t-1}, y_{t-2}, \dots, y_{t-(w-2)}), \\ \hat{y}_{t+3} &= f(\hat{y}_{t+2}, \hat{y}_{t+1}, y_t, y_{t-1}, \dots, y_{t-(w-3)}), \\ &\quad \cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\ &\quad \cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\ &\quad \cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\ \hat{y}_{t+p} &= f(\hat{y}_{(t+p)-1}, \hat{y}_{(t+p)-2}, \dots, \hat{y}_{(t+p)-w}), \end{aligned} \quad (4.2)$$

em que p é o número de passos realizados até a degradação atingir o limiar de falha, que consequentemente, é o valor de RUL previsto.

4.3 Método de predição recursiva com a variação em relação ao valor atual

Outra forma de extrair as características da série é usar a variação dos valores, ou seja, ao invés de usar os dados brutos da série, utiliza-se o valor mais recente disponível e suas variações em relação aos valores nos instantes anteriores como entradas da rede:

$$\hat{y}_t = f(y_{t-1}, y_{t-1} - y_{t-2}, y_{t-1} - y_{t-3}, \dots, y_{t-1} - y_{t-w}). \quad (4.3)$$

O processo de estimação do valor de RUL é realizado conforme segue:

$$\begin{aligned} \hat{y}_{t+1} &= f(y_t, y_t - y_{t-1}, y_t - y_{t-2}, \dots, y_t - y_{t-(w-1)}), \\ \hat{y}_{t+2} &= f(\hat{y}_{t+1}, \hat{y}_{t+1} - y_t, \hat{y}_{t+1} - y_{t-1}, \dots, \hat{y}_{t+1} - y_{t-(w-2)}), \\ \hat{y}_{t+3} &= f(\hat{y}_{t+2}, \hat{y}_{t+2} - \hat{y}_{t+1}, \hat{y}_{t+2} - y_t, \dots, \hat{y}_{t+2} - y_{t-(w-3)}), \\ &\cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\ &\cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\ &\cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\ \hat{y}_{t+p} &= f(\hat{y}_{t+(p-1)}, \hat{y}_{t+(p-1)} - \hat{y}_{t+(p-2)}, \dots, \hat{y}_{t+(p-1)} - y_{t-(w-p)}), \end{aligned} \quad (4.4)$$

devido as características das séries de degradação possuir uma tendência crescente ou decrescente, esta técnica tenta capturar essas variações de crescimento e decrescimento do valor atual em relação aos dados passados.

4.4 Resultados do prognóstico de falhas para a plataforma PRONOSTIA

4.4.1 Conjunto de Dados

Este conjunto de dados é gerado a partir da plataforma PRONOSTIA (Figura 4.2), que realiza uma degradação acelerada de rolamentos (*bearings*) (NECTOUX *et al.*, 2012).

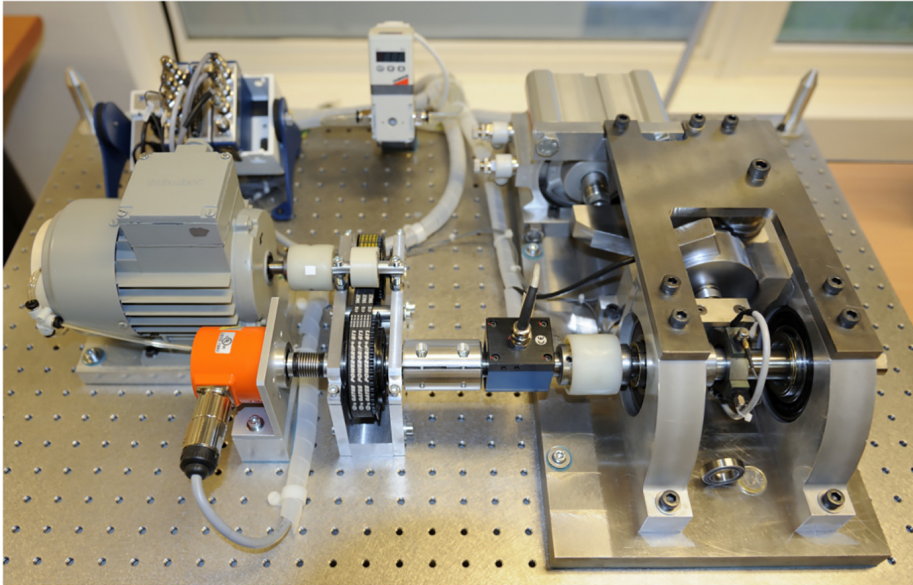
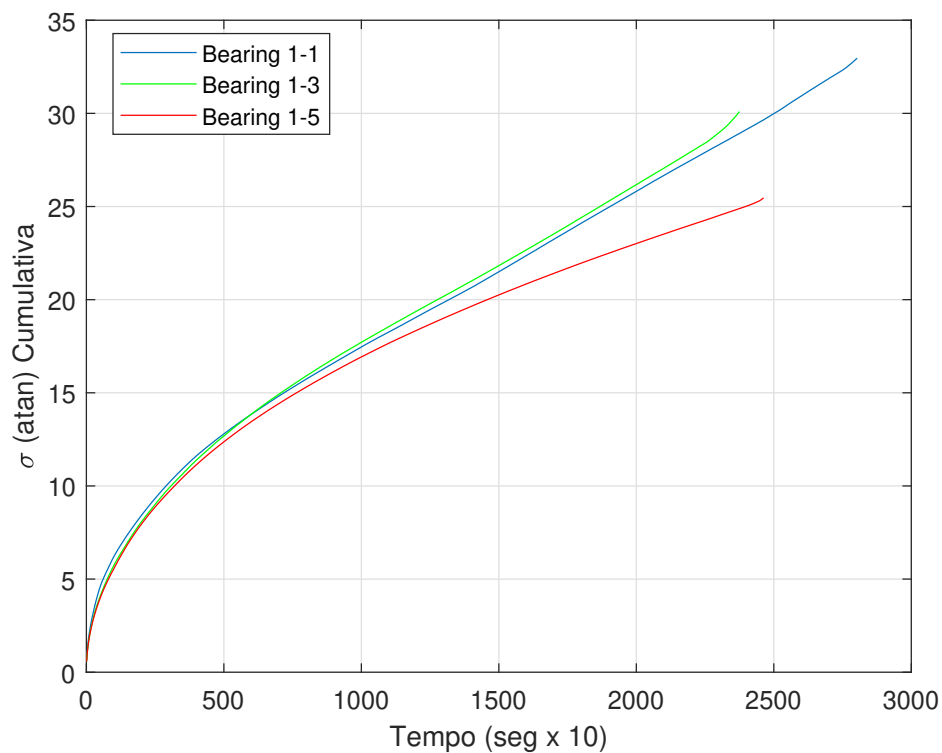


Figura 4.2 – Plataforma PRONOSTIA

Fonte: Pan *et al.* (2020).

Os dados usados foram transformados utilizando os procedimentos descritos por Javed *et al.* (2015), no qual se aplica a função trigonométrica (*atan*), o desvio padrão, a média móvel exponencial para suavização e logo após é realizada a soma cumulativa de forma a obter uma série monotônica da degradação, como ilustra a Figura 4.3.

Figura 4.3 – Trajetória cumulativa (*atan*) dos rolamentos 1, 3 e 5

Este *dataset* é composto de dados de 17 rolamentos em três diferentes condições de operação, porém, para o experimento foram extraídos somente dados de 3 rolamentos (*Bearing* 1-1, *Bearing* 1-3 e *Bearing* 1-5), que foram submetidos as condições operacionais de 1800 rpm e 4000 N. O Rolamento 1 foi utilizado para treino e os rolamentos 3 e 5 para teste.

4.4.2 Estimação de RUL do desgaste de rolamentos da plataforma PRONOSTIA utilizando os dados brutos

Estes experimentos foram realizados utilizando os dados brutos conforme foi apresentado na Seção 4.2 e com a NFN configurada com cinco funções de pertinência, uma janela de tamanho 4 e 50 épocas no processo de treinamento *offline* utilizando o rolamento 1. Com os rolamentos 3 e 5 são executados os testes, onde os primeiros 500 dados foram utilizados no treinamento *online*.

O limiar de falha foi definido como sendo um valor de σ (*atan*) cumulativo de 20. O gráfico da Figura 4.4 ilustra o comportamento da degradação de predição para o rolamento 3, e na Figura 4.5 do rolamento 5. Na Tabela 4.1 são apresentados os resultados de RUL desses experimentos.

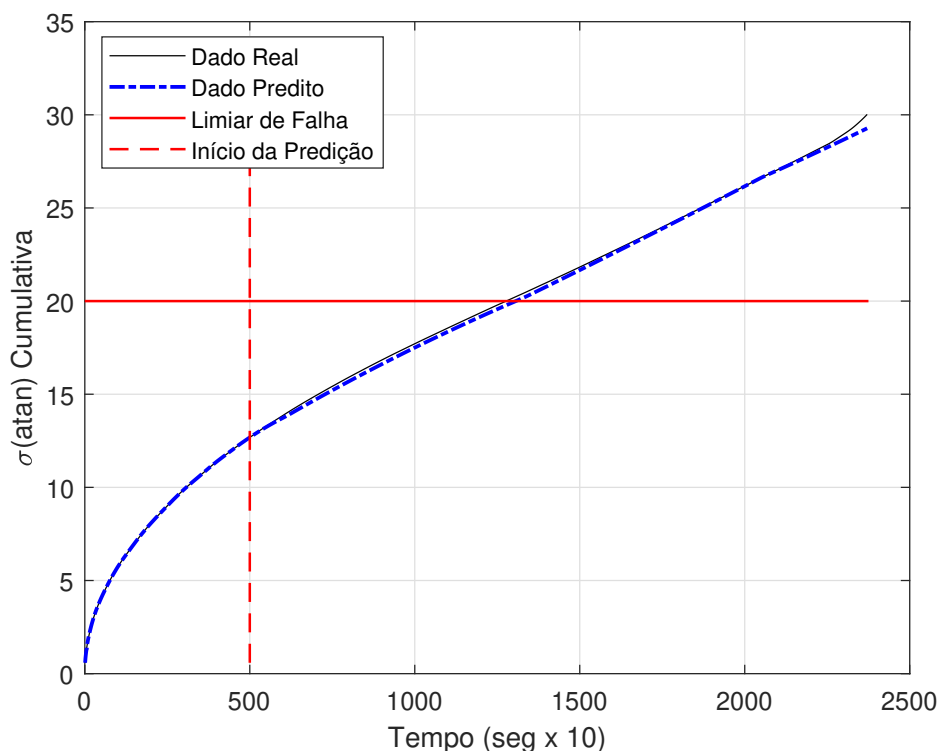


Figura 4.4 – Predição da degradação do rolamento 3 da plataforma PRONOSTIA com predição recursiva utilizando os dados brutos da série

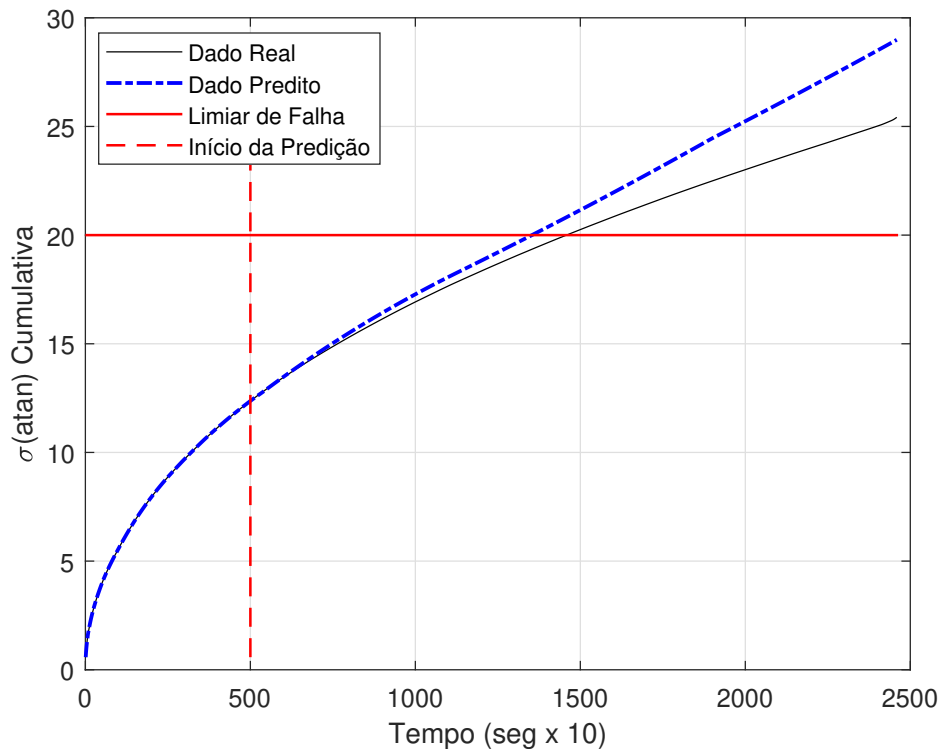


Figura 4.5 – Predição da degradação do rolamento 5 da plataforma PRONOSTIA com predição recursiva utilizando os dados brutos da série

Tabela 4.1 – Resultados do prognóstico de falhas para a plataforma PRONOSTIA dos rolamentos 3 e 5 com predição recursiva com os dados brutos da série

| Rolamento | RUL real | RUL predito |
|--------------------|----------|-------------|
| <i>Bearing 1.3</i> | 776 | 806 |
| <i>Bearing 1.5</i> | 957 | 853 |

4.4.3 Estimação de RUL do desgaste de rolamentos da plataforma PRONOSTIA utilizando o valor atual e suas variações

Assim como no experimento anterior, neste foi realizado o treinamento *offline* com o rolamento 1, submetido a 150 épocas, e os testes são feitos nos rolamentos 3 e 5, onde se utiliza dos primeiros 500 dados para treinamento *online*. As demais configurações foram quatro funções de pertinência e uma janela de tamanho 5.

Nas Figuras 4.6 e 4.7, têm-se, respectivamente, os gráficos do progresso da predição de degradação dos rolamentos 3 e 5. Os resultados de RUL para os dois testes são apresentados na Tabela 4.2.

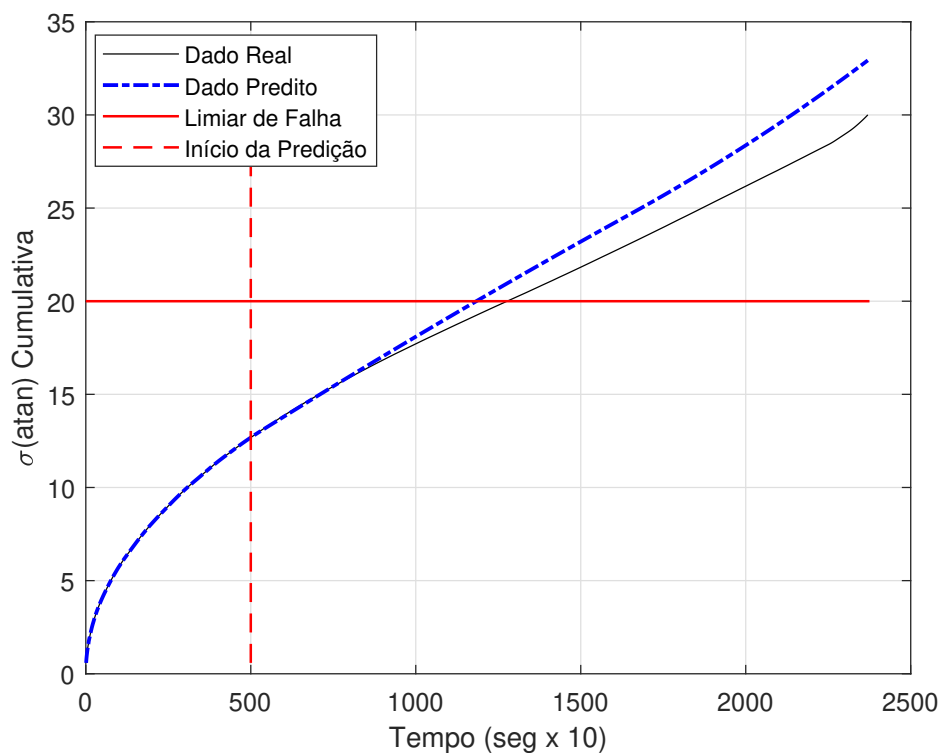


Figura 4.6 – Predição da degradação do rolamento 3 da plataforma PRONOSTIA com predição recursiva utilizando a variação em relação ao valor atual

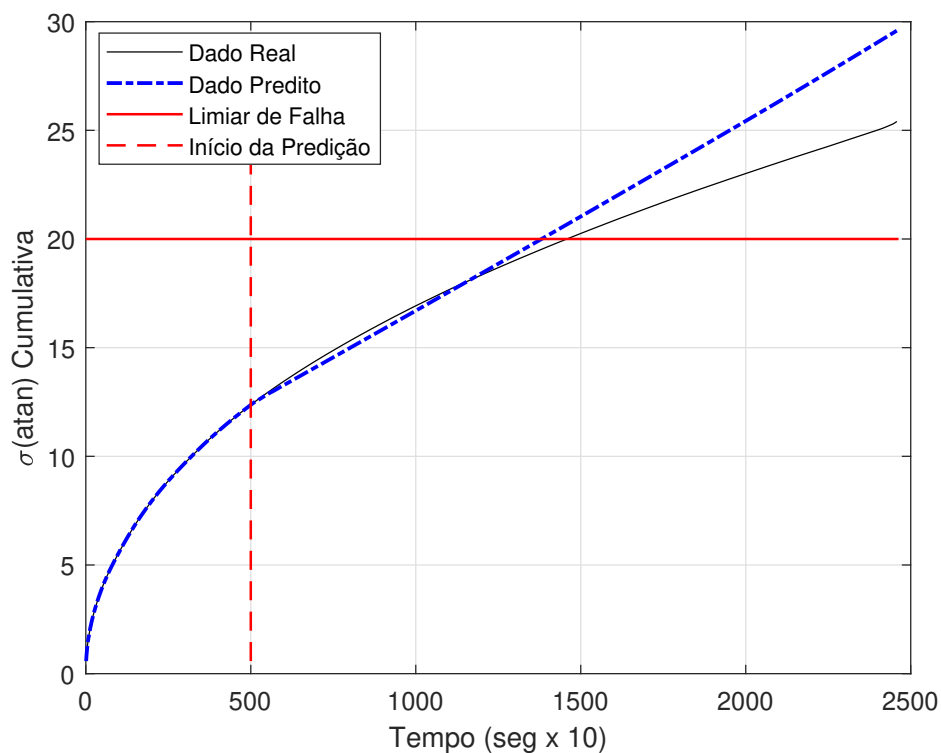


Figura 4.7 – Predição da degradação do rolamento 5 da plataforma PRONOSTIA com predição recursiva utilizando a variação em relação ao valor atual

Tabela 4.2 – Resultados do prognóstico de falhas para a plataforma PRONOSTIA dos rolamentos 3 e 5 com predição recursiva com variação em relação ao valor atual.

| Rolamento | RUL real | RUL predito |
|--------------------|----------|-------------|
| <i>Bearing 1.3</i> | 776 | 683 |
| <i>Bearing 1.5</i> | 957 | 881 |

4.5 Resultados do prognóstico de falhas para as ferramentas de Corte da CNC

4.5.1 Conjunto de dados

Os dados de desgaste das ferramentas de corte da máquina de comando numérico (*Computer Numeric Control - CNC*) (Figura 4.8) foram retirados do desafio “*prognostic data challenge 2010*”, disponibilizado pela Sociedade de Prognóstico e Gestão da Saúde (*Prognostics and Health Management Society-PHM*) ([PHM Society, 2010](#)). [Li et al. \(2009\)](#) apresentam detalhes da extração desses dados obtidos.

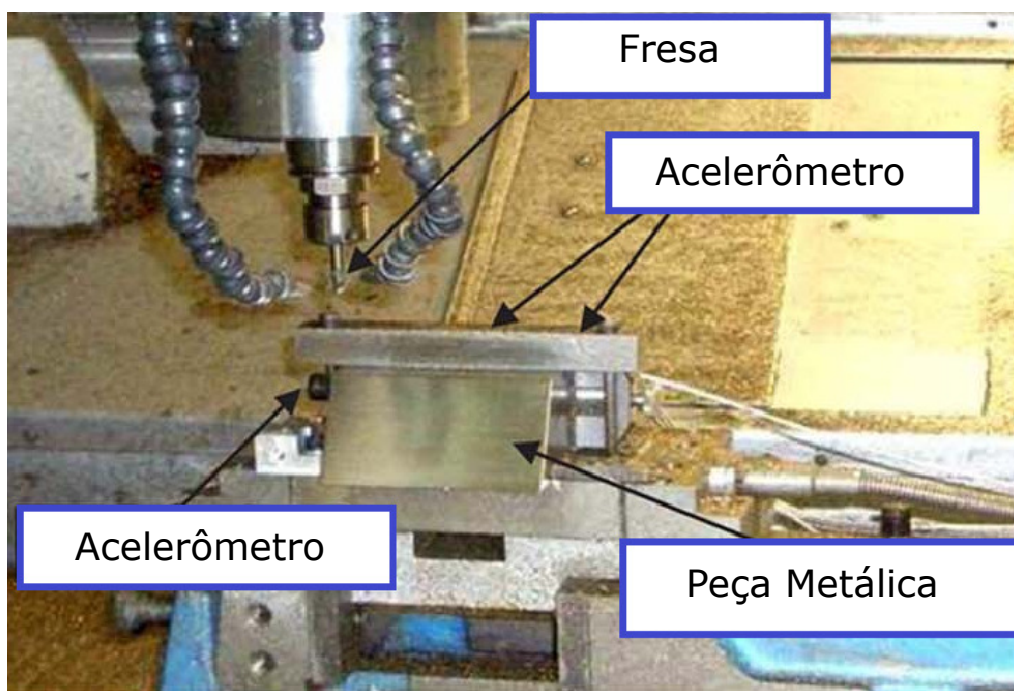


Figura 4.8 – Máquina CNC de alta velocidade

Fonte: [Li et al. \(2009\)](#).

O conjunto de dados possui características de seis ferramentas de corte numeradas de 1 a 6, do início do uso ao final de sua vida útil. Em todos os casos, são feitos 315 cortes com cada ferramenta, esse é o limite máximo de cortes estipulado para sua troca. Dessas seis ferramentas,

foram armazenados dados referentes ao desgaste do diâmetro da ponta das ferramenta 1, 4 e 6. A Figura 4.9 apresenta o gráfico com as medidas de desgastes em milésimos de milímetros (10^{-3} mm) em relação ao número do corte dessas três ferramentas.

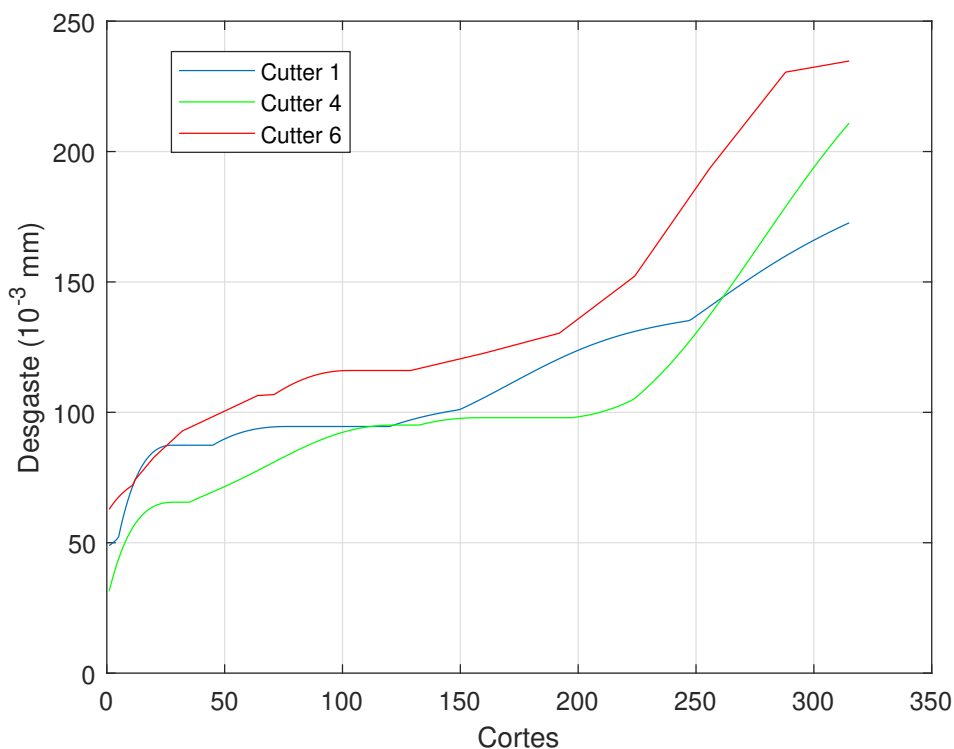


Figura 4.9 – Trajetória de desgaste das ferramentas *cutter 1*, *cutter 4* e *cutter 6*

4.5.2 Estimação de RUL em ferramentas de corte de uma CNC utilizando os dados brutos

Com o modelo de predição recursiva e utilizando os dados brutos da série foi realizado a estimação do desgaste da ferramenta *cutter 1* e *6*, onde foi utilizado os dados da *cutter 4* para treinamento *offline* do modelo, com quatro funções de pertinência, uma janela de tamanho 3 e 500 épocas. Neste experimento, os primeiros 150 valores, foram submetidos ao treinamento *online*, os resultados das predições das séries de desgaste das ferramentas *cutter 1* e *cutter 6* são apresentados nos gráficos das Figuras 4.10 e 4.11 respectivamente.

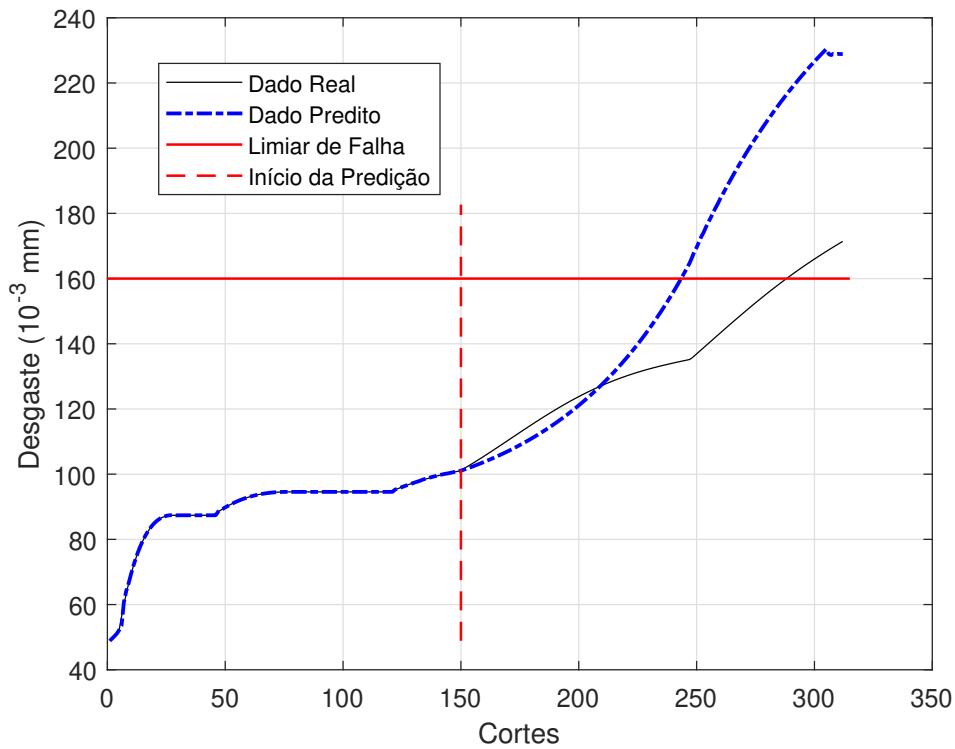


Figura 4.10 – Predição de desgaste da ferramenta *cutter* 1 com predição recursiva utilizando os dados brutos da série

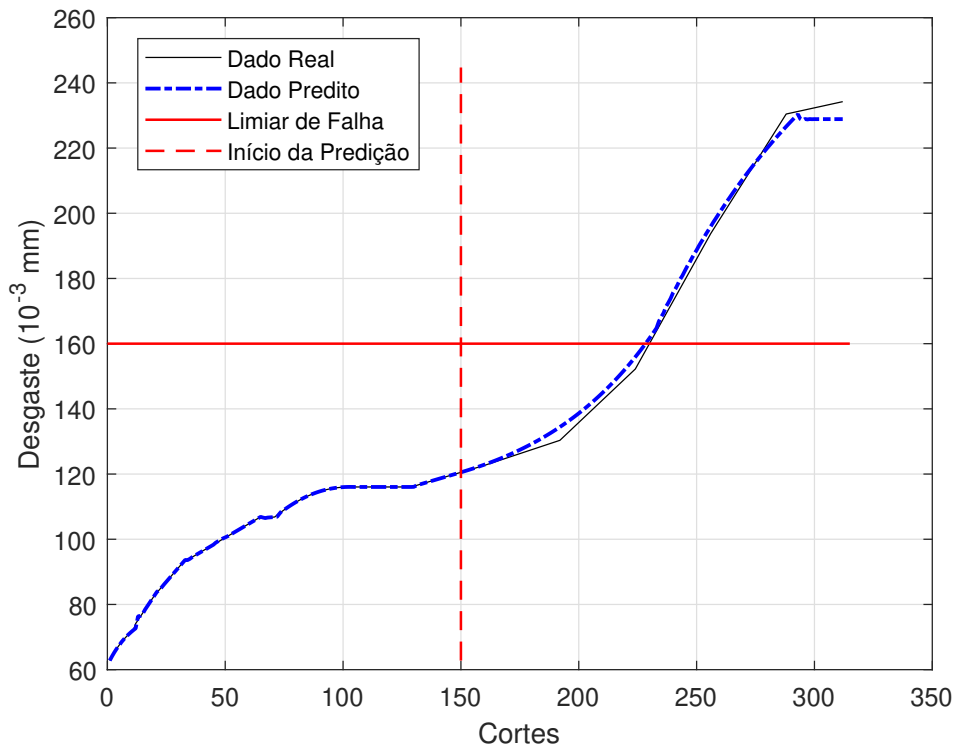


Figura 4.11 – Predição de desgaste da ferramenta *cutter* 6 com predição recursiva utilizando os dados brutos da série

Na Tabela 4.3 são apresentados os resultados de tempo de vida útil encontrado, que é quando o desgaste atinge o valor $160 \times 10^{-3} \text{ mm}$, que foi o valor de limiar de falha definido.

Tabela 4.3 – Resultados do prognóstico de falhas para as ferramentas de corte da CNC com predição recursiva utilizando os dados brutos da série

| Ferramenta | RUL real | RUL predito |
|-----------------|----------|-------------|
| <i>Cutter 1</i> | 138 | 93 |
| <i>Cutter 6</i> | 79 | 78 |

4.5.3 Estimação de RUL em ferramentas de corte de uma CNC utilizando o valor atual e suas variações

Neste experimento, os primeiros 150 valores também foram submetidos ao treinamento *online* e foi utilizado os dados da *cutter 4* para treinamento *offline* do modelo, com oito funções de pertinência, uma janela de tamanho 4 e 500 épocas. Os resultados das predições das séries de desgaste das ferramentas *cutter 1* e *cutter 6* são apresentados nos gráficos das Figuras 4.12 e 4.13 respectivamente. Na Tabela 4.4 são apresentados os resultados da RUL.

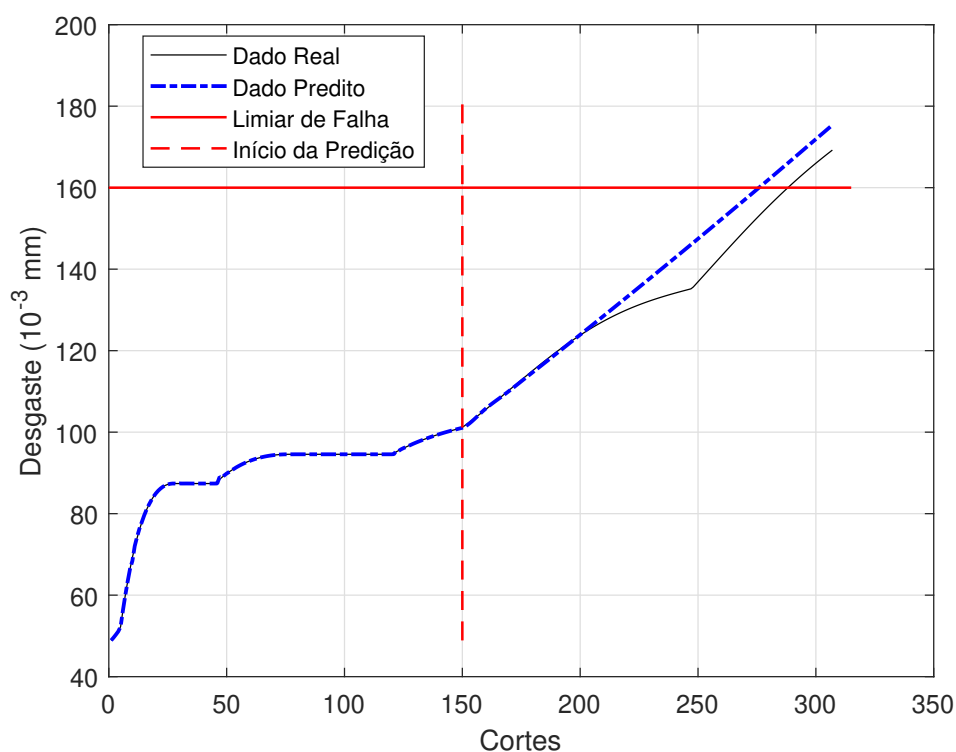


Figura 4.12 – Predição de desgaste da ferramenta *cutter 1* com predição recursiva utilizando a variação em relação ao valor atual

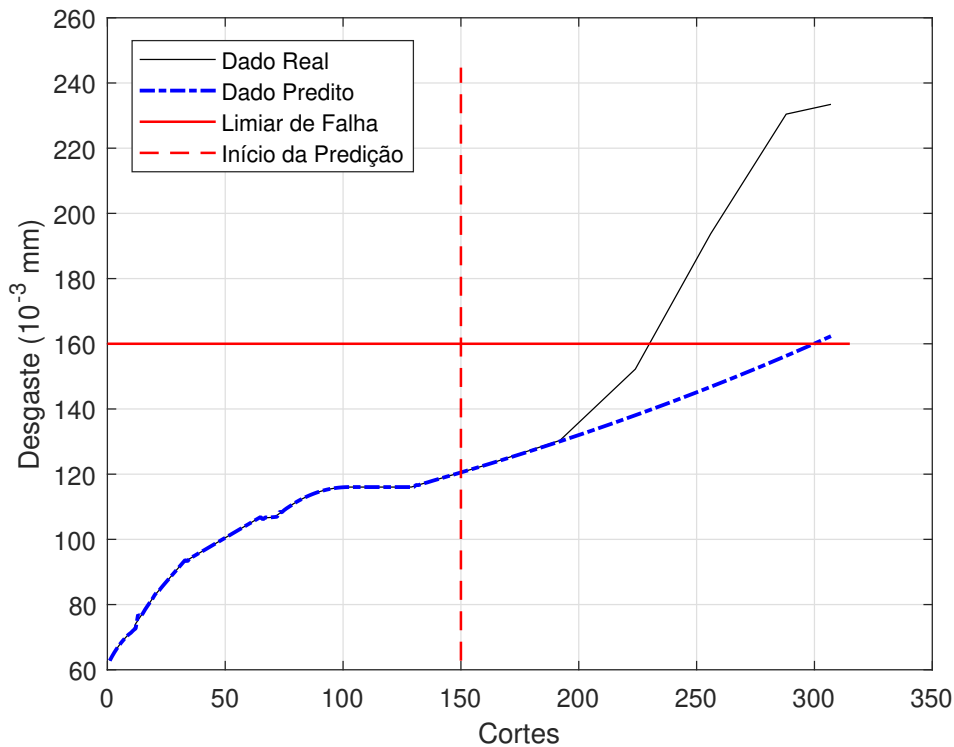


Figura 4.13 – Predição de desgaste da ferramenta *cutter 6* com predição recursiva utilizando a variação em relação ao valor atual

Tabela 4.4 – Resultados do prognóstico de falhas para as ferramentas de corte da CNC com predição recursiva utilizando a variação em relação ao valor atual

| Ferramenta | RUL real | RUL predito |
|-----------------|----------|-------------|
| <i>Cutter 1</i> | 79 | 125 |
| <i>Cutter 6</i> | 138 | 149 |

4.6 Resultados do prognóstico de falhas nas baterias de íons de lítio

Nesta seção, é apresentado experimentos da estimativa de RUL no conjunto de dados de baterias comerciais de íons de lítio tipo 1850, do Centro de Excelência em Prognósticos da NASA (SAHA; GOEBEL, 2007). Essas baterias são numeradas em B0005, B0006, B0007 e B0018, respectivamente, e suas classificações de capacidade são todas de $2.0Ah$. Os dados obtidos dessas baterias foram extraídos a partir de um experimento, onde elas foram submetidas repetidamente a medidas de carga, descarga e impedância na temperatura de 25° , os valores de capacitância (Ah) por ciclo de descarga que serão utilizados, neste trabalho, estão representados na Figura 4.14.

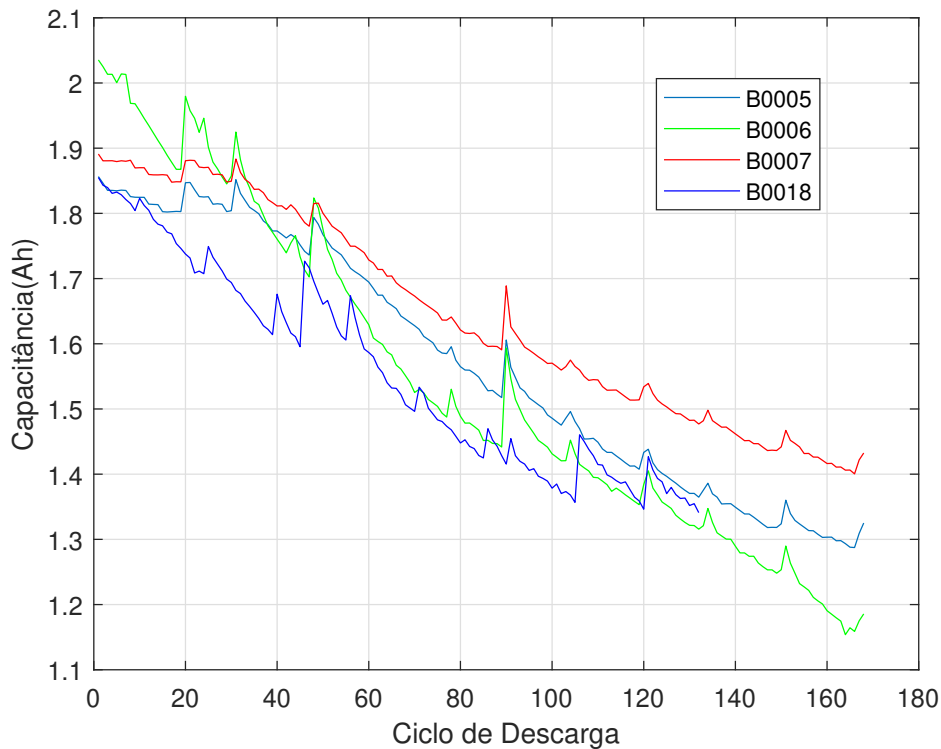


Figura 4.14 – Trajetória de degeneração das bateria B0005, B0006, B0007 e B0018.

O limiar de falha para essas baterias, normalmente, é atingido em 70% de sua capacidade total, sendo assim, para uma capacidade de 2.0Ahr o limiar é definido em:

$$2 \times 0.7 = 1.4Ahr. \quad (4.5)$$

4.6.1 Estimação de RUL em baterias de íons de lítio utilizando os dados brutos

A configuração do modelo neste experimento com as baterias de íons de lítio foi realizado com duas funções de pertinência, uma janela de tamanho 4 e 250 épocas no treinamento *offline*. O modelo treinado é submetido ao treinamento *online* nos primeiros 70 dados de teste e, a partir daí, é realizada a predição dos dados subsequentes para a estimativa da RUL. A bateria B0006 foi separada para o treinamento *offline* da rede, e os testes foram realizados nas baterias B0005 como pode ser visto na Figura 4.15 e B0007 na Figura 4.16, os resultados de RUL a partir do ponto inicial de predição 70 e limiar em 1.4 são apresentados na Tabela 4.5.

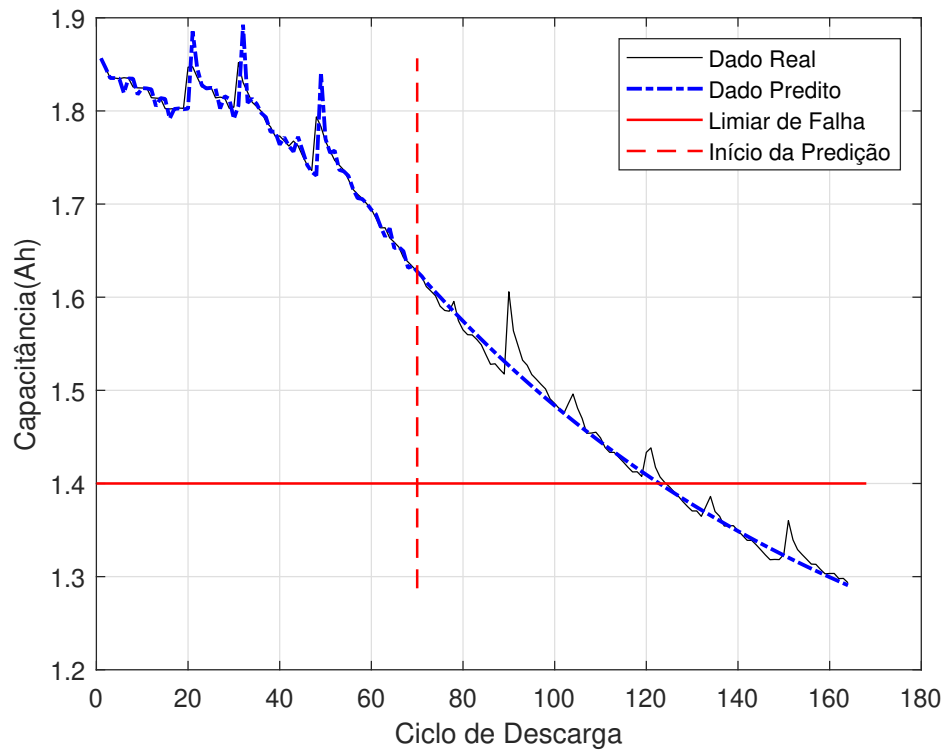


Figura 4.15 – Predição da degeneração da bateria B0005 com a predição recursiva utilizando os dados brutos da série

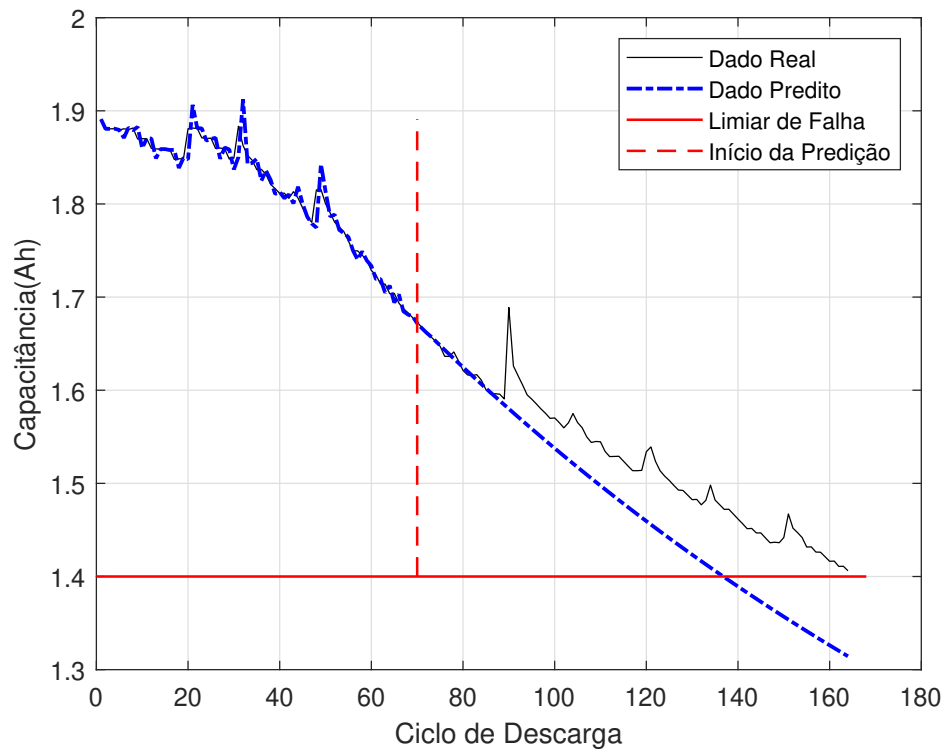


Figura 4.16 – Predição da degeneração da bateria B0007 com a predição recursiva utilizando os dados brutos da série

Tabela 4.5 – Resultados dos prognóstico de falhas nas baterias B0005 e B007 com predição recursiva utilizando os dados brutos da série

| Bateria | RUL real | RUL predito |
|---------|----------|-------------|
| B0005 | 54 | 52 |
| B0007 | 94 | 66 |

4.6.2 Estimação de RUL em baterias de íons de lítio utilizando o valor atual e suas variações

As configurações do modelo neste experimento foram as mesmas apresentadas no experimento anterior, duas funções de pertinência, janela de tamanho 4 e 250 épocas no treinamento *offline* com a bateria B0006 e treinamento *online* nos primeiros 70 dados das baterias B0005 e B007, que foram utilizadas para os testes. Nas Figura 4.17 e 4.18 têm-se os gráficos de predição da degeneração das baterias. Os resultados de RUL a partir do ponto inicial de predição 70 e limiar em 1.4 são apresentados na Tabela 4.6.

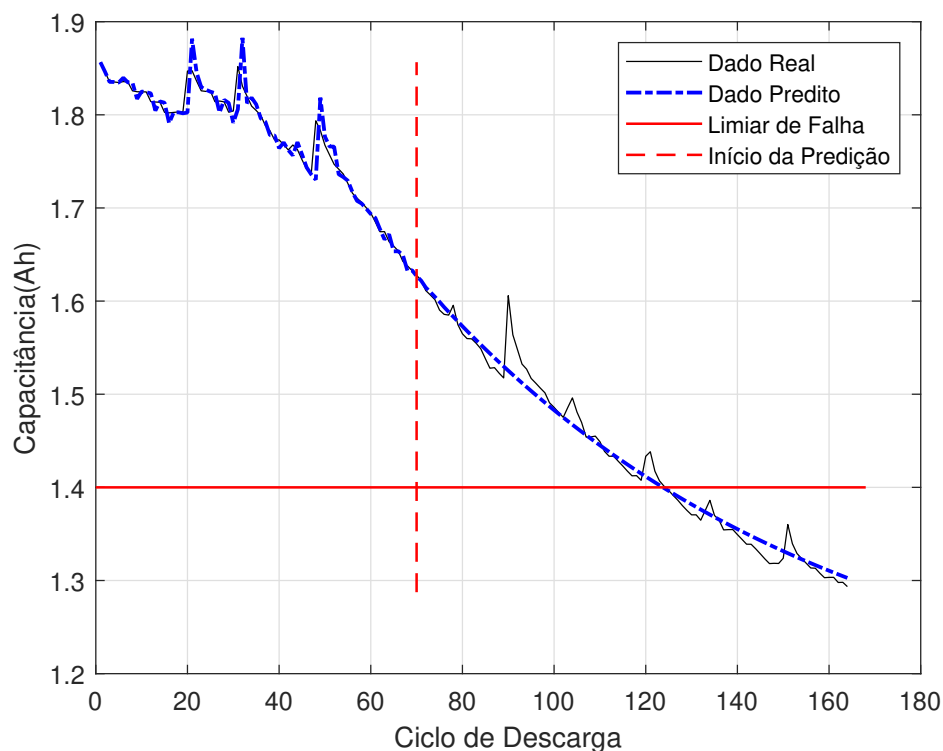


Figura 4.17 – Predição da degeneração da bateria B0005 com a predição recursiva utilizando a variação em relação ao valor atual

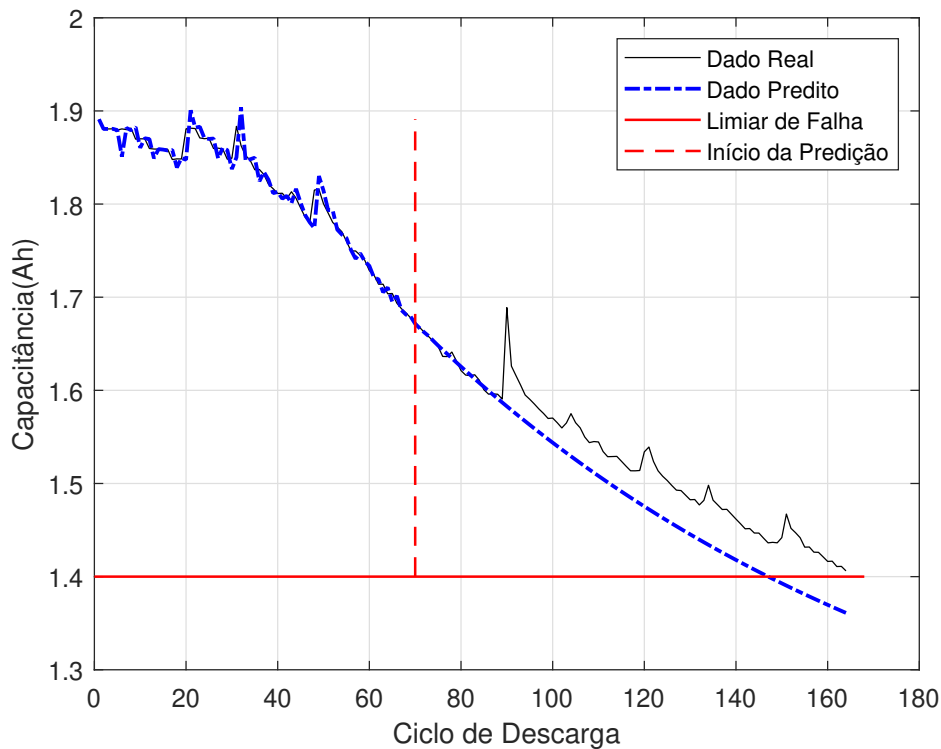


Figura 4.18 – Predição da degeneração da bateria B0007 com a predição recursiva utilizando a variação em relação ao valor atual

Tabela 4.6 – Resultados dos prognóstico de falhas nas baterias B0005 e B007 com predição recursiva utilizando a variação em relação ao valor atual

| Bateria | RUL real | RUL predito |
|---------|----------|-------------|
| B0005 | 55 | 53 |
| B0007 | 94 | 77 |

4.7 Resumo

Percebe-se, pelos experimentos apresentados, que a NFN consegue obter resultados melhores e mais consistentes, quando a série de degradação não apresenta mudanças bruscas e também possui pouca variação local. Como foi o caso da base de dados PRONOSTIA onde os dois métodos apresentaram bons resultados de previsão da RUL.

No prognóstico de falha das baterias, devido a série que representa a degradação de capacitância, medida utilizada para determinar o tempo de vida útil da bateria, possui poucos ciclos de carga e descarga. A NFN não conseguiu capturar as pequenas variações na periodicidade da série estimada. No entanto, ela consegue capturar a tendência para estimar o valor de RUL como foi verificado com os testes realizados utilizando os dois métodos apresentados.

Já nos experimentos com as ferramentas de corte da CNC, pode ser observado que ao utilizar os dados brutos da série, como as características para a previsão, o resultado de RUL encontrado para a ferramenta *cutter* 6 foi melhor em relação a *cutter* 1. Já com o método utilizando o valor atual e suas variações, a ferramenta *cutter* 1 teve melhor resultado. O que indica que esses dois métodos podem ser tratados de forma complementar para um melhor desempenho nos resultados da NFN.

PROPOSTA DE PROGNÓSTICO DE FALHA INTERVALAR COM A NEO-FUZZY-NEURON

Neste capítulo, será apresentado a *Neo-Fuzzy-Neuron* com pesos intervalares e como é feita a atualização desses pesos no processo de estimação da RUL. Também serão realizados experimentos nos problemas de prognóstico de falhas das bases de dados descritas no capítulo anterior.

5.1 *Neo-Fuzzy-Neuron* com Pesos Intervalares

No Capítulo 4 foram apresentados métodos de prognóstico de falhas com previsão vários passos à frente utilizando a rede *Neo-Fuzzy-Neuron*. Um problema que temos ao realizar previsões vários passos à frente são a propagação do erro para as próximas predições, com isso o valor de RUL estimado pode ficar abaixo ou acima do RUL real, para realizar esta estimativa com valor de RUL inferior e superior é proposto a *Neo-Fuzzy-Neuron* com pesos intervalares, como ilustrado na Figura 5.1.

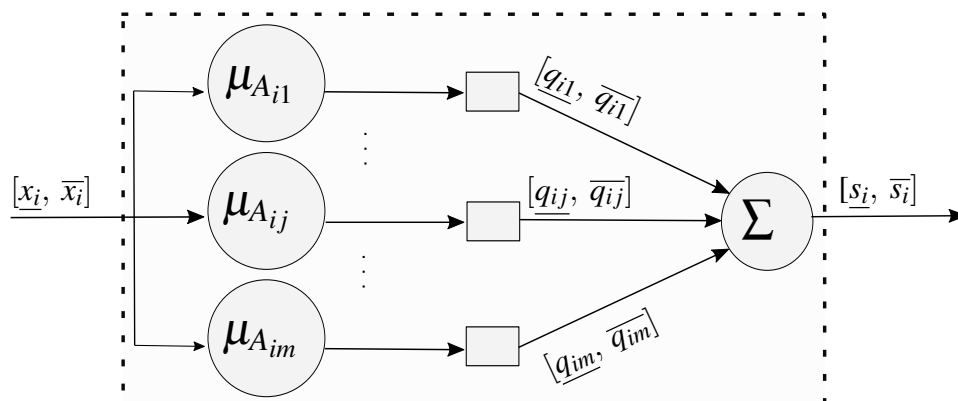


Figura 5.1 – *Neo-Fuzzy-Neuron* com peso intervalar

No processo de treinamento onde temos as informações a estrutura utiliza não é intervalar, os pesos intervalares são adicionas na fase de estimação, em que é utilizado um erro estimado para atualização dos pesos intervalares. A *Neo-Fuzzy-Neuron* com pesos intervalares diferencia da clássica, porque ao adicionar os pesos como sendo um intervalo, a saída para um dado padrão de entrada resultará em dois valores na saída \underline{s}_i e \bar{s}_i .

Ao tratar o intervalo como duas entradas independentes, serão ativadas no mínimo duas e no máximo quatro funções de pertinência diferentes, onde podem ser listados os seguintes casos:

- **Caso 1:** o intervalo está entre os centros das funções k_i^1 e $k_i^1 + 1$, Figura 5.2, neste caso as mesmas funções de pertinências são ativadas para ambos os valores de entrada, inferior e superior.

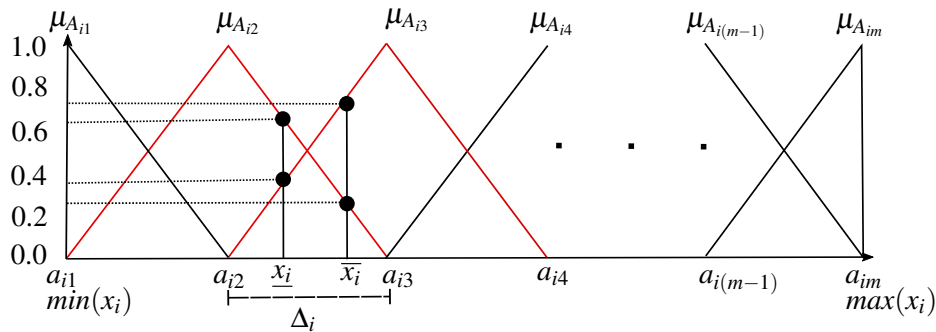


Figura 5.2 – Funções de pertinência triangular com duas funções ativas

- **Caso 2:** o centro da função $k_i^1 + 1$ está dentro do intervalo e os demais centros estão fora, Figura 5.3, neste caso três funções de pertinências são ativadas, onde as funções $k_i^1 + 1$ e k_i^2 são as mesmas para ambos os valores inferior e superior.

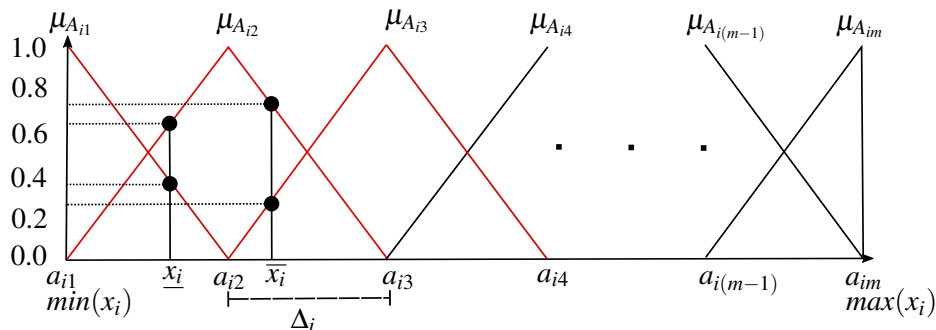


Figura 5.3 – Funções de pertinência triangular com três funções ativas

- **Caso 3:** há mais de um centro dentro do intervalo $[\underline{x}_i, \bar{x}_i]$, Figura 5.4, neste caso, são ativadas quatro funções de pertinências.

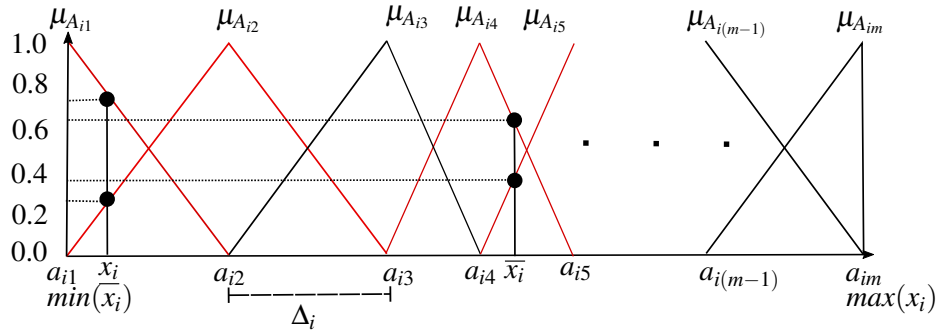


Figura 5.4 – Funções de pertinência triangular com quatro funções ativas

Sendo assim, as funções ativas podem ser indexadas por k_i^1 e $k_i^1 + 1$, para entrada inferior; k_i^2 e $k_i^2 + 1$ para a superior, de forma que elas sejam complementares. Assim, as saídas individuais podem ser calculadas como segue:

$$\begin{aligned} \underline{s}_i &= \mu_{A_{ik_i^1}}(\underline{x}_i) \underline{q}_{ik_i^1} + \mu_{A_{ik_i^1+1}}(\underline{x}_i) \underline{q}_{ik_i^1+1}, \\ \bar{s}_i &= \mu_{A_{ik_i^2}}(\bar{x}_i) \bar{q}_{ik_i^2} + \mu_{A_{ik_i^2+1}}(\bar{x}_i) \bar{q}_{ik_i^2+1}. \end{aligned} \quad (5.1)$$

O somatório das saídas \underline{s}_i e \bar{s}_i resulta na saída intervalar da rede, sendo o somatório de \underline{s}_i a saída inferior e o somatório de \bar{s}_i a saída superior:

$$\begin{aligned} \underline{\hat{y}} &= \sum_{i=1}^n \underline{s}_i, \\ \bar{\hat{y}} &= \sum_{i=1}^n \bar{s}_i. \end{aligned} \quad (5.2)$$

Para obter um valor estimado pontual é extraído o valor médio da saída inferior $\underline{\hat{y}}$ com a superior $\bar{\hat{y}}$:

$$\hat{y}^m = \frac{(\underline{\hat{y}} + \bar{\hat{y}})}{2}. \quad (5.3)$$

5.2 Atualização dos pesos intervalares

Quando o processo de aprendizagem *online* atinge a amostra t_i , que é onde se inicia o processo de predição de RUL, para cada novo valor estimado ocorre um erro de predição $e_{t_i+1} = (\hat{y}_{t_i+1} - y_{t_i+1})$, e este erro é propagado para as próximas predições. Com o intuito de estabelecer um intervalo para o valor de predição de RUL, é calculado o erro absoluto médio (e_m) e o α médio (α_m) das primeiras t_i amostras do aprendizado *online*:

$$e_m = \frac{1}{t_i} \sum_{t=1}^{t_i} |\hat{y}_t - y_t|, \quad (5.4)$$

$$\alpha_m = \frac{1}{t_i} \sum_{t=1}^{t_i} \alpha_t, \quad (5.5)$$

Ao multiplicar o erro absoluto médio por uma constante (ρ) positiva e outra negativa é obtido um erro intervalar:

$$[\underline{e}_m, \overline{e}_m] = [-\rho e_m, \rho e_m] \quad (5.6)$$

onde o intervalo entre os valores de RUL inferior e superior cresce de acordo com o valor de ρ , ou seja, um valor de ρ igual a zero ($\rho = 0$) ocasiona um intervalo nulo, que é quando os valores inferior e superior são iguais.

Para a predição intervalar os pesos da NFN são substituídos por pesos intervalares e os pesos referente as funções ativas para entrada inferior, indexadas por k_i^1 e $k_i^1 + 1$, são atualizados:

$$\begin{aligned} \underline{q}_{ik_i^1}^{Atual} &= \underline{q}_{ik_i^1}^{Anterior} - \alpha_m \overline{e}_m \mu_{A_{ik_i^1}}(\underline{x}_i), \\ \underline{q}_{ik_i^1+1}^{Atual} &= \underline{q}_{ik_i^1+1}^{Anterior} - \alpha_m \overline{e}_m \mu_{A_{ik_i^1+1}}(\underline{x}_i), \end{aligned} \quad (5.7)$$

e também os pesos das funções ativas para a entrada superior, indexadas por k_i^2 e $k_i^2 + 1$:

$$\begin{aligned} \overline{q}_{ik_i^2}^{Atual} &= \overline{q}_{ik_i^2}^{Anterior} - \alpha_m \underline{e}_m \mu_{A_{ik_i^2}}(\overline{x}_i), \\ \overline{q}_{ik_i^2+1}^{Atual} &= \overline{q}_{ik_i^2+1}^{Anterior} - \alpha_m \underline{e}_m \mu_{A_{ik_i^2+1}}(\overline{x}_i). \end{aligned} \quad (5.8)$$

Desta forma, a cada novo valor previsto os pesos da rede são atualizados com base no α_m e no erro absoluto médio encontrado no treinamento *online* das primeiras amostras.

5.2.1 Predição de RUL intervalar com os pesos intervalares da rede

Com a estratégia recursiva, apresentada no Capítulo 4, em que a predição dos novos valores é obtido a partir de dados anteriormente previstos, foi utilizada para predição intervalar com os dados brutos da série. O processo recursivo é repetido até atingir os limiares de falha inferior (\underline{y}_{t+p}) e superior (\overline{y}_{t+p}).

A Expressão 5.9 exemplifica o processo de estimação intervalar, onde f representa o

mapeamento de entrada e saída da rede *Neo-Fuzzy-Neuron* com os pesos intervalares.

$$\begin{aligned}
 [\hat{y}_{t+1}, \overline{\hat{y}_{t+1}}] &= f(y_t, y_{t-1}, y_{t-2}, y_{t-3}, \dots, y_{t-(w-1)}), \\
 [\hat{y}_{t+2}, \overline{\hat{y}_{t+2}}] &= f([\hat{y}_{t+1}, \overline{\hat{y}_{t+1}}], y_t, y_{t-1}, y_{t-2}, \dots, y_{t-(w-2)}), \\
 [\hat{y}_{t+3}, \overline{\hat{y}_{t+3}}] &= f([\hat{y}_{t+2}, \overline{\hat{y}_{t+2}}], [\hat{y}_{t+1}, \overline{\hat{y}_{t+1}}], y_t, y_{t-1}, \dots, y_{t-(w-3)}), \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 [\hat{y}_{t+p}, \overline{\hat{y}_{t+p}}] &= f([\hat{y}_{(t+p)-1}, \overline{\hat{y}_{(t+p)-1}}], [\hat{y}_{(t+p)-2}, \overline{\hat{y}_{(t+p)-2}}], \dots, \\
 &\quad \dots, [\hat{y}_{(t+p)-(w-p)}, \overline{\hat{y}_{(t+p)-(w-p)}}]).
 \end{aligned} \tag{5.9}$$

A Figura 5.5 apresenta um fluxograma dos passos para treinamento e estimação de RUL intervalar com a NFN.

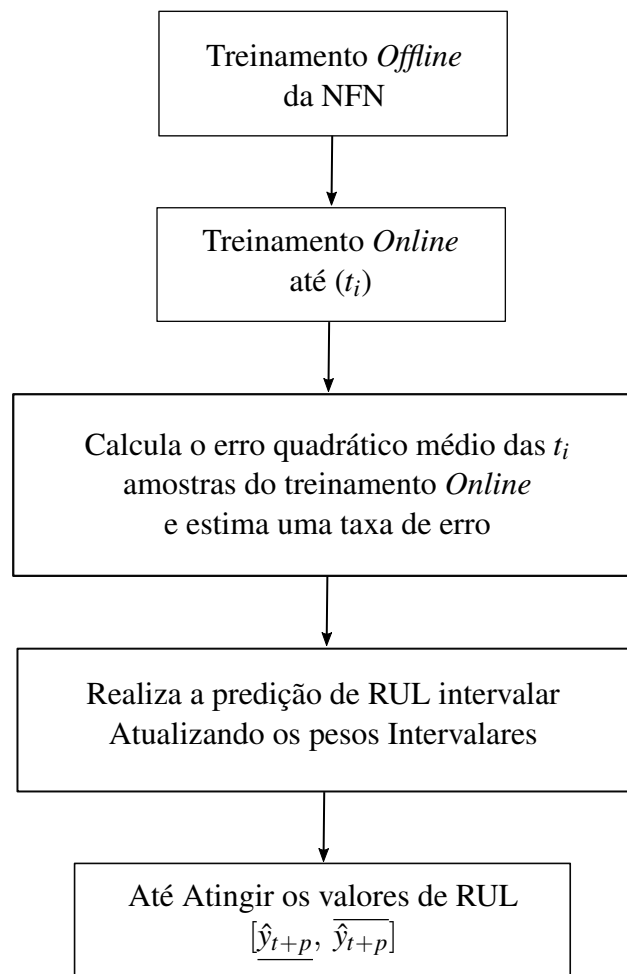


Figura 5.5 – Fluxograma do processo de previsão de RUL intervalar com a rede *Neo-Fuzzy-Neuron*

5.3 Resultados Experimentais

Nesta seção são apresentados os resultados da NFN com pesos intervalares nos problemas de prognóstico listados no Capítulo 4. São utilizados dados de três equipamentos diferentes para cada problema: um para treinamento da rede; o outro para validação, em que é utilizado para a definição dos parâmetros (número de funções de pertinência, tamanho da janela de dados e o valor de ρ que define o tamanho do intervalo); e com o terceiro é realizado o teste do modelo treinado. Os resultados de validação e teste são apresentados graficamente e em forma de tabela para cada experimento.

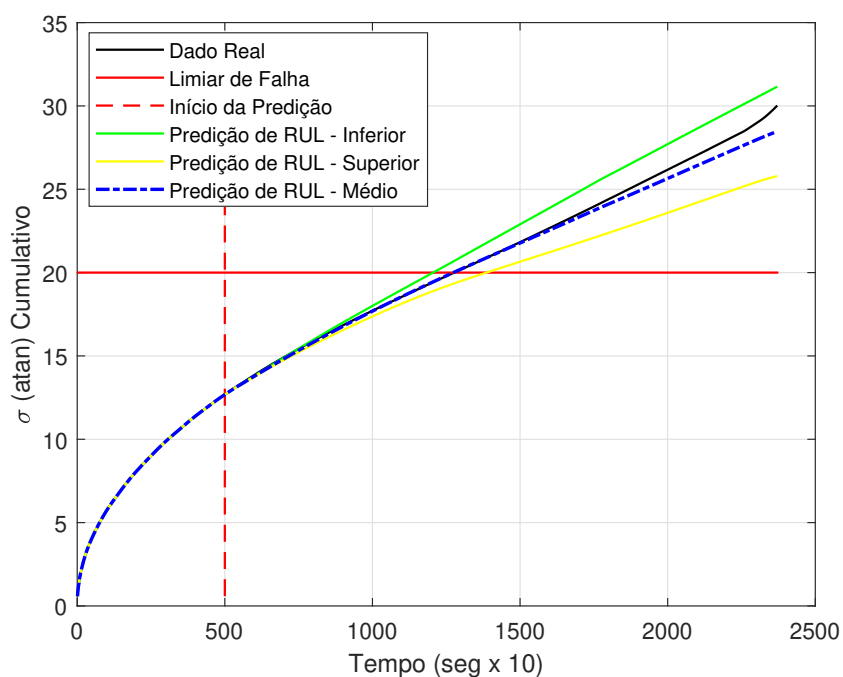
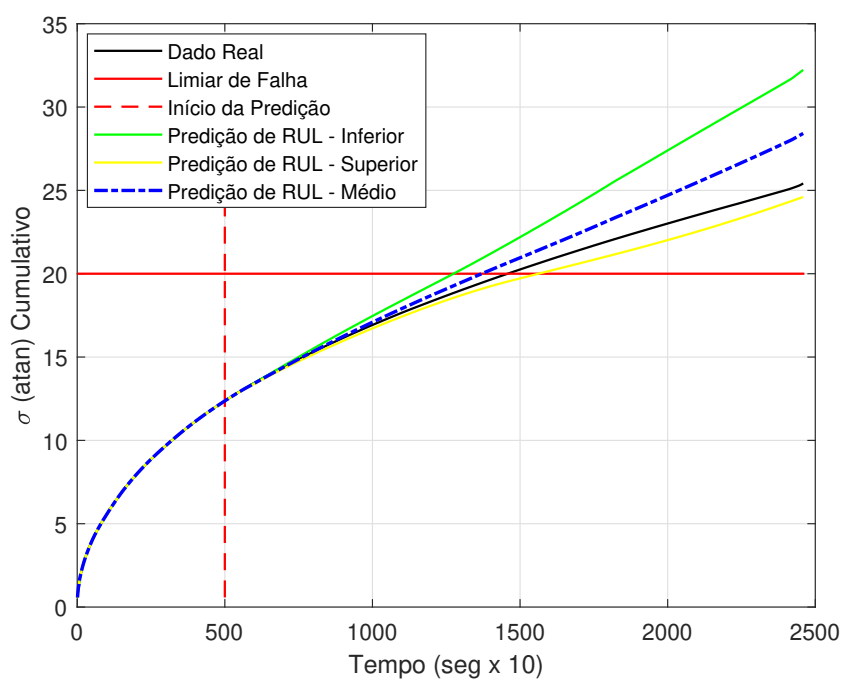
5.3.1 Predição de RUL intervalar para a plataforma PRONOSTIA

Neste o experimento, a rede foi configurada com cinco funções de pertinências, onde foram utilizados uma janela de dados de tamanho 5 e 50 épocas no treinamento *offline* da NFN com os dados do rolamento 1. Já os rolamentos 3 e 5 foram usados para validação e teste, respectivamente, sendo que os 500 primeiros dados desses rolamentos são submetidos ao treinamento *online* da rede, com o α ótimo encontrado a partir da Equação 2.13.

Considerando o erro absoluto médio das primeiras 500 amostras apresentadas para o treinamento *online*, sendo o erro intervalar (Equação 5.6) ponderado por $\rho = 0.015$ e utilizando o α médio, são atualizados os pesos intervalares a cada novo valor previsto da série, até atingir o RUL intervalar conforme mostra a Expressão 5.10:

$$\begin{aligned}
 [\hat{y}_{501}, \overline{\hat{y}_{501}}] &= f(y_{500}, y_{499}, y_{498}, y_{497}, y_{496}), \\
 [\hat{y}_{502}, \overline{\hat{y}_{502}}] &= f([\hat{y}_{501}, \overline{\hat{y}_{501}}], y_{500}, y_{499}, y_{498}, y_{497}), \\
 [\hat{y}_{503}, \overline{\hat{y}_{503}}] &= f([\hat{y}_{502}, \overline{\hat{y}_{502}}], [\hat{y}_{501}, \overline{\hat{y}_{501}}], y_{500}, y_{499}, y_{498}), \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 [\hat{y}_{500+p}, \overline{\hat{y}_{500+p}}] &= f([\hat{y}_{499+p}, \overline{\hat{y}_{499+p}}], [\hat{y}_{498+p}, \overline{\hat{y}_{498+p}}], [\hat{y}_{497+p}, \overline{\hat{y}_{497+p}}], \\
 &\quad [\hat{y}_{496+p}, \overline{\hat{y}_{496+p}}], [\hat{y}_{495+p}, \overline{\hat{y}_{495+p}}]).
 \end{aligned} \tag{5.10}$$

No gráfico da Figura 5.6 mostra o resultado de predição do rolamento 3. Nota-se que o valor de RUL Real ficou dentro do intervalo de predição, ao qual está bem próximo do valor da RUL Inferior. Já na predição do rolamento 5, apresentado no gráfico da Figura 5.7, o valor de RUL Real ficou mais próximo do RUL Superior, no entanto, também ficou dentro do intervalo de predição.

Figura 5.6 – Predição intervalar da série de degradação $\sigma(atan)$ cumulativa do rolamento 3Figura 5.7 – Predição intervalar da série de degradação $\sigma(atan)$ cumulativa do rolamento 5

Na Tabela 5.1 são apresentados os resultados de previsão de RUL intervalar, em que são mostrados os valores obtidos de RUL Inferior, RUL Superior e RUL Médio dos rolamentos.

Tabela 5.1 – Resultados experimentais da predição de RUL intervalar dos rolamentos 3 e 5

| Rolamento | Início Predição | Limiar de Falha | RUL Inferior | RUL Superior | RUL Médio | RUL Real |
|----------------------|-----------------|-----------------|--------------|--------------|-----------|----------|
| <i>Bearing</i> (1.3) | 500 | 20 | 705 | 886 | 773 | 777 |
| <i>Bearing</i> (1.5) | 500 | 20 | 774 | 1062 | 870 | 958 |

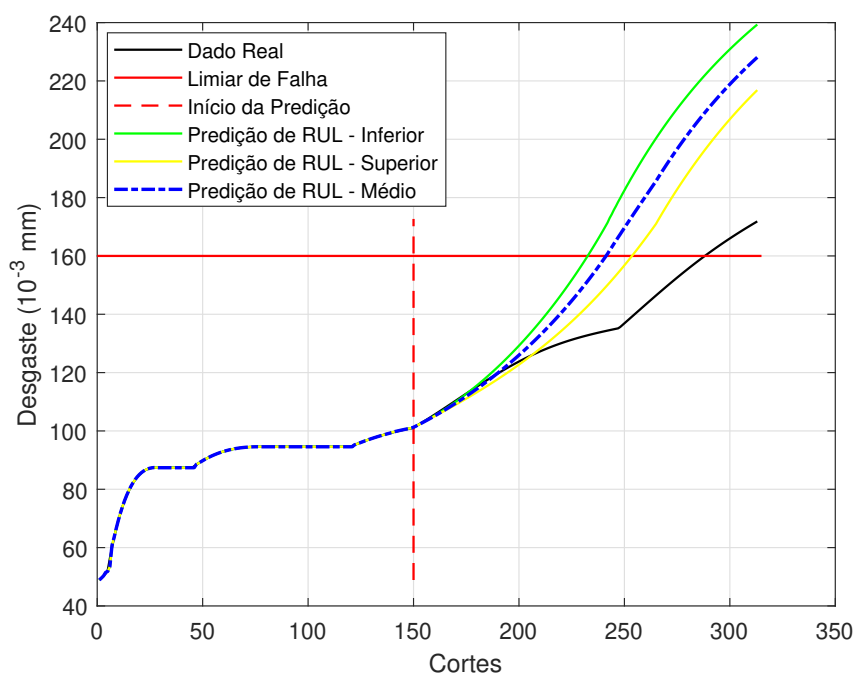
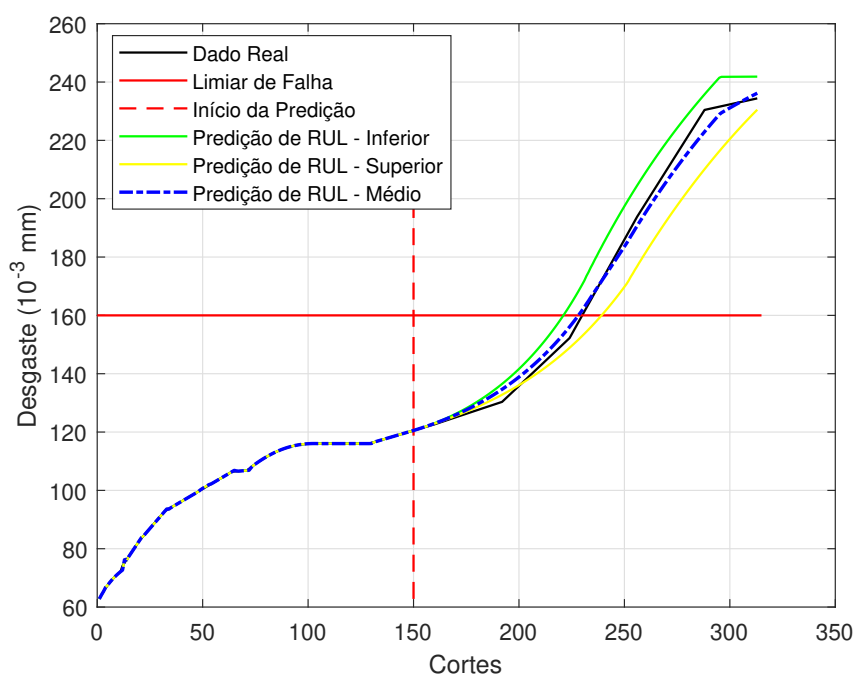
5.3.2 Predição de RUL intervalar para a Máquina de Comando Numérico

O modelo foi configurado com quatro funções de pertinências, uma janela de dados de tamanho 2 e 500 épocas no treinamento *offline*. A ferramenta *cutter* 4 foi utilizada para treinamento *offline* e as *cutter* 1 e 6 foram utilizadas para validação e teste, sendo que os 150 primeiros dados de cortes são utilizados para treinamento *online* da rede.

Considerando o erro absoluto médio das primeiras amostras apresentadas para o treinamento *online*, sendo o erro intervalar (Equação 5.6) ponderado por $\rho = 0.075$ e com o α médio, são atualizados os pesos intervalares a cada novo valor previsto da série, até atingir o RUL intervalar conforme mostra a Expressão 5.11:

$$\begin{aligned}
 [\hat{y}_{151}, \overline{\hat{y}_{151}}] &= f(y_{150}, y_{149}), \\
 [\hat{y}_{152}, \overline{\hat{y}_{152}}] &= f([\hat{y}_{151}, \overline{\hat{y}_{151}}]), \\
 [\hat{y}_{153}, \overline{\hat{y}_{153}}] &= f([\hat{y}_{152}, \overline{\hat{y}_{152}}], [\hat{y}_{151}, \overline{\hat{y}_{151}}], y_{150}), \\
 &\cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\
 &\cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\
 &\cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\
 [\hat{y}_{150+p}, \overline{\hat{y}_{150+p}}] &= f([\hat{y}_{149+p}, \overline{\hat{y}_{149+p}}], [\hat{y}_{148+p}, \overline{\hat{y}_{148+p}}]).
 \end{aligned} \tag{5.11}$$

No gráfico da Figura 5.8 é mostrado o resultado de predição da ferramenta *cutter* 1 e é possível observar que o valor de RUL Real ficou fora do intervalo de predição, para que capturasse o valor de RUL Real seria necessário aumentando o valor de ρ , porém teria um intervalo muito grande. Já na predição da ferramenta *cutter* 6, apresentado no gráfico da Figura 5.9, percebemos que o valor de RUL Real ficou dentro do intervalo de predição.

Figura 5.8 – Predição intervalar da série de degradação da ferramenta *Cutter 1*Figura 5.9 – Predição intervalar da série de degradação da ferramenta *Cutter 6*

A Tabela 5.2 apresenta os resultados de previsão de RUL intervalar para as ferramentas de cortes da CNC.

Tabela 5.2 – Resultados experimentais da predição de RUL intervalar nas ferramentas *cutter* 1 e *cutter* 6

| Ferramenta | Início Predição | Limiar de Falha | RUL Inferior | RUL Superior | RUL Médio | RUL Real |
|-------------------|-----------------|-----------------|--------------|--------------|-----------|----------|
| <i>Cutter</i> (1) | 150 | 160 | 72 | 93 | 81 | 138 |
| <i>Cutter</i> (6) | 150 | 160 | 61 | 79 | 68 | 79 |

5.3.3 Predição de RUL intervalar para para as baterias de íons de lítio

O modelo foi configurado com duas funções de pertinências, com uma janela de dados de tamanho 4 e 250 épocas no treinamento *offline*. A bateria B0006 foi utilizada para treinamento *offline* e as B0005 e B0007 foram utilizadas para validação e teste, sendo que os 60 primeiros dados dessas baterias são utilizados no treinamento *online* da rede.

Considerando o erro absoluto médio das primeiras 60 amostras apresentadas para o treinamento *online*, o erro intervalar ponderado por $\rho = 0.008$ e utilizando o α médio, são atualizados os pesos intervalares a cada novo valor previsto da série, até atingir o RUL intervalar conforme mostra a Expressão 5.12:

$$\begin{aligned}
[\hat{y}_{61}, \overline{\hat{y}_{61}}] &= f(y_{60}, y_{59}, y_{58}, y_{57}), \\
[\hat{y}_{62}, \overline{\hat{y}_{62}}] &= f([\hat{y}_{61}, \overline{\hat{y}_{61}}], y_{60}, y_{59}, y_{58}), \\
[\hat{y}_{63}, \overline{\hat{y}_{63}}] &= f([\hat{y}_{62}, \overline{\hat{y}_{62}}], [\hat{y}_{61}, \overline{\hat{y}_{61}}], y_{60}, y_{59}), \\
&\cdot \quad \cdot \quad \cdot \\
&\cdot \quad \cdot \quad \cdot \\
&\cdot \quad \cdot \quad \cdot \\
[\hat{y}_{60+p}, \overline{\hat{y}_{60+p}}] &= f([\hat{y}_{59+p}, \overline{\hat{y}_{59+p}}], [\hat{y}_{58+p}, \overline{\hat{y}_{58+p}}], [\hat{y}_{57+p}, \overline{\hat{y}_{57+p}}], [\hat{y}_{56+p}, \overline{\hat{y}_{56+p}}]).
\end{aligned} \tag{5.12}$$

No gráfico da Figura 5.10 é mostrado o resultado de predição da bateria B0005, onde o intervalo captura o valor de RUL Real ficando próximo ao RUL Médio. Já no gráfico da Figura 5.11 temos a predição da bateria B0007, no qual o valor de RUL Real fica próximo do RUL Superior.

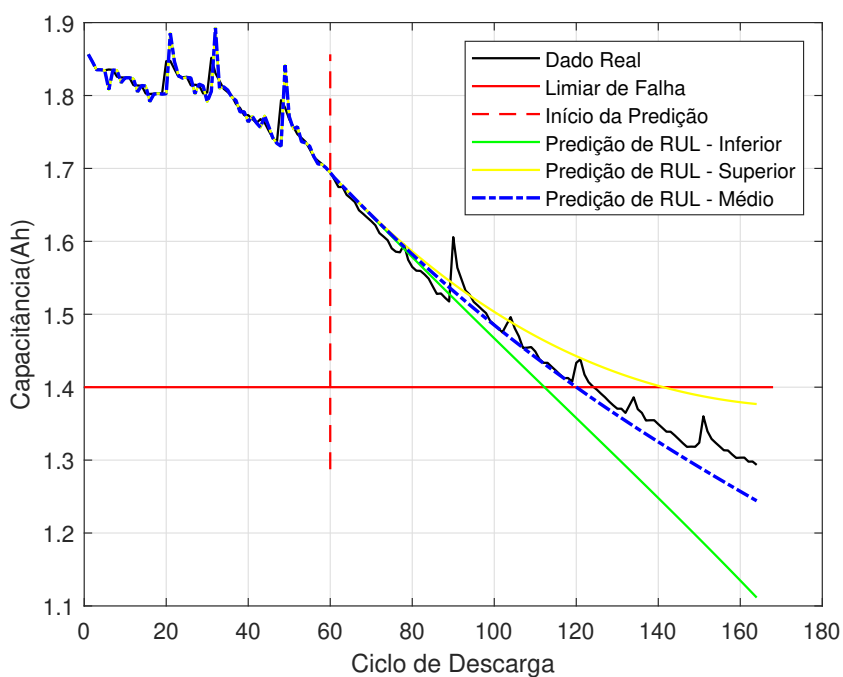


Figura 5.10 – Predição intervalar da série de degeneração da bateria B0005

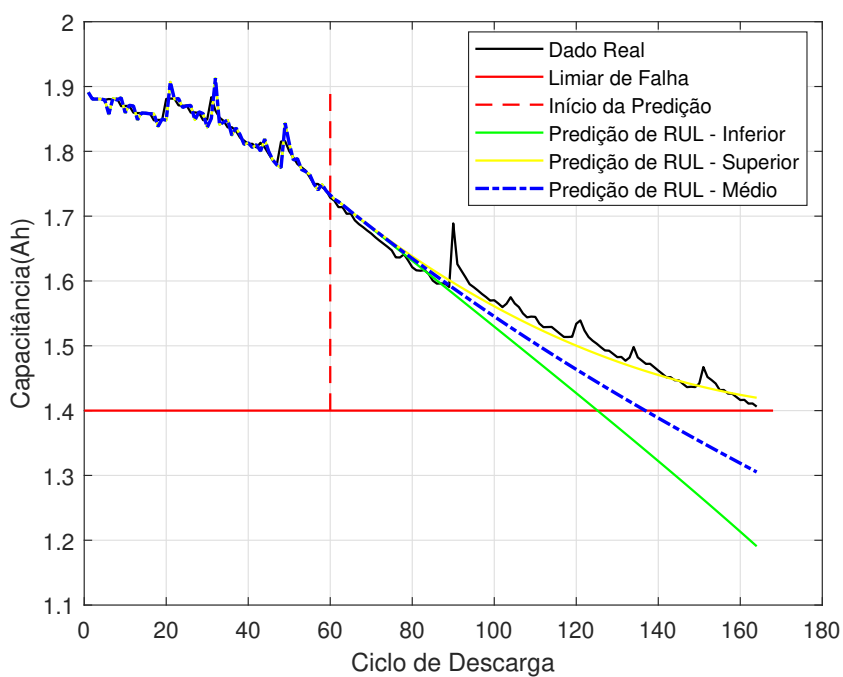


Figura 5.11 – Predição intervalar da série de degeneração da bateria B0007

A Tabela 5.3 apresenta os resultados de previsão de RUL intervalar para as baterias de íons de lítio.

Tabela 5.3 – Resultados experimentais da predição de RUL intervalar das baterias B0005 e B0007

| Bateria | Início Predição | Limiar de Falha | RUL Inferior | RUL Superior | RUL Médio | RUL Real |
|-----------------|-----------------|-----------------|--------------|--------------|-----------|----------|
| Bateria (B0005) | 60 | 1.4 | 52 | 81 | 60 | 64 |
| Bateria (B0007) | 60 | 1.4 | 65 | 76 | 104 | 104 |

5.4 Resumo

Com a NFN intervalar foi possível definir um intervalo para o valor de previsão da RUL, que cresce conforme vai aumentando a quantidade de passos necessários para estimar seu valor. Assim, ao final do processo, a NFN retorna três valores de estimativa da RUL, sendo um valor inferior, um superior e o médio.

Cada um dos três *datasets* têm suas características próprias como escala, variação e trajetória de degradação. O que fez com que fosse utilizado um valor de ρ , que define o tamanho do intervalo, diferente para cada *dataset*.

CONCLUSÃO

Na parte inicial deste trabalho foi feita a apresentação da rede *Neo-Fuzzy-Neuron* e foi avaliado o seu desempenho com as funções de pertinência triangular uniforme e não uniforme em problemas de séries temporais. Para geração das funções de pertinência não uniforme, foi utilizado o algoritmo não supervisionado de *kohonen*, assim foi possível escolher as funções de pertinência que melhor representa o conjunto de dados de entrada para aprendizado do modelo. Nos experimentos numéricos executados de previsão da série temporal, de *Mackey Glass* e do forno a gás de *Box-Jenkins*, as funções de pertinência ajustadas pela rede de *kohonen* demonstraram um melhor desempenho que as uniformes.

Logo em seguida, o trabalho aborda e avalia técnicas de predição em problemas de prognóstico de falhas para estimar o tempo de Vida Útil Remanescente (RUL) utilizando a rede NFN. Em que, o modelo da NFN é ajustado em modo *offline* e para estimar a RUL os dados iniciais da série de degradação são submetidos ao aprendizado *online*. As metodologias aplicadas usam a técnica de previsão vários passos à frente recursiva. Na primeira metodologia a extração de características foram utilizados os dados brutos da série e na segunda foi usado o valor atual e suas variações. As técnicas apresentadas foram avaliadas nos conjuntos de dados de problemas de prognósticos das baterias de íons de lítio, das ferramentas de corte de uma máquina de comando numérico, e de rolamentos extraídos da plataforma PRONOSTIA.

Nos experimentos realizados, a NFN teve melhor desempenho no problema da plataforma PRONOSTIA, isso se dá pelo fato de os dados apresentarem pouco ruído e não possuírem variações bruscas, uma vez que eles foram transformados em uma série com característica monotônica. Tendo isso em questão, a NFN se apresenta como uma ótima opção para aplicações em problemas de estimação do tempo de vida útil remanescente.

Problemas de previsão muitos à frente, como é o caso dos prognósticos de falhas apresentados para estimativa do tempo de vida útil remanescente, são problemas de difícil estimação, devido ao erro propagado das predições iniciais, o que se faz necessário utilizar de um intervalo

de previsão. Por isso, foi proposto pesos intervalares para a NFN, e assim poder estimar um intervalo de previsão da RUL, e então ter um valor conservador e outro otimista para o decisor na saída do modelo.

6.1 Trabalhos Futuros

As metodologias apresentadas para a previsão de RUL apresentaram bons resultados, no entanto, propõe-se apresentar experimentos para diferentes pontos de início da predição e verificar a precisão do resultado de RUL conforme o valor da série se aproxima do limiar de falha.

Com relação a *Neo-Fuzzy-Neuron* Intervalar, sugere-se encontrar formas para representar as funções de pertinência como sendo um intervalo, para que seja possível utilizar operação intervalar, uma vez que, neste trabalho, foi considerado que as entradas inferior e superior eram independentes, para garantir a característica complementar das funções triangulares.

E ainda, realizar outros testes com as metodologias apresentadas neste trabalho, em todo o conjunto da base de dados PRONOSTIA, com a técnica de validação cruzada, e também avaliar as metodologias em outros *datasets* como o da turbina de *turbofan*.

REFERÊNCIAS

- BERENJI, H. R.; KHEDKAR, P. Learning and tuning fuzzy logic controllers through reinforcements. **IEEE Transactions on Neural Networks**, v. 3, n. 5, p. 724–740, 1992. Citado na página 27.
- BIAGETTI, T.; SCIUBBA, E. Automatic diagnostics and prognostics of energy conversion processes via knowledge-based systems. **Energy**, v. 29, n. 12, p. 2553 – 2572, 2004. Citado na página 28.
- BOX, G. E.; JENKINS, G. M.; REINSEL, G. C.; LJUNG, G. M. **Time series analysis: forecasting and control**. [S.l.]: John Wiley & Sons, 2015. Citado na página 43.
- CAMARGOS, M. O.; BESSA, I.; D’ANGELO, M. F. S. V.; COSME, L. B.; PALHARES, R. M. Data-driven prognostics of rolling element bearings using a novel error based evolving takagi–sugeno fuzzy model. **Applied Soft Computing**, v. 96, p. 106628, 2020. ISSN 1568-4946. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1568494620305664>>. Citado na página 29.
- CAMINHAS, W.; GOMIDE, F. A fast learning algorithm for neofuzzy networks. In: **Proc. Information Processing and Management of Uncertainty in Knowledge Based Systems**. [S.l.: s.n.], 2000. v. 1, n. 1, p. 1784–1790. Citado nas páginas 32 e 35.
- CAMINHAS, W.; TAVARES, H.; GOMIDE, F. Rede lógica neurofuzzy: Aplicação em diagnóstico de falhas em sistemas dinâmicos. In: **Anais do XI Congresso Brasileiro de Automática**. [S.l.: s.n.], 1996. v. 1, p. 495–464. Citado na página 27.
- CAMINHAS, W. M. *et al.* **Estratégias de detecção e diagnóstico de falhas em sistemas dinâmicos**. Tese (Doutorado) — Universidade Estadual de Campinas, 1997. Citado nas páginas 37 e 38.
- CAMINHAS, W. M.; TAVARES, H.; GOMIDE, F. A.; PEDRYCZ, W. Fuzzy set based neural networks: Structure, learning and application. **JACIII**, v. 3, n. 3, p. 151–157, 1999. Citado na página 34.
- COSME, L. B.; CAMINHAS, W. M.; D’ANGELO, M. F. S. V.; PALHARES, R. M. A novel fault-prognostic approach based on interacting multiple model filters and fuzzy systems. **IEEE Transactions on Industrial Electronics**, v. 66, n. 1, p. 519–528, 2019. Citado na página 29.
- COSME, L. B.; D’ANGELO, M. F.; CAMINHAS, W. M.; YIN, S.; PALHARES, R. M. A novel fault prognostic approach based on particle filters and differential evolution. **Applied Intelligence**, Springer, v. 48, n. 4, p. 834–853, 2018. Citado na página 29.
- GOUVÊA, M. R. de. **Controle Neurofuzzy de motor de indução com estimação de parâmetros e fluxo de estator**. Tese (Doutorado) — Universidade Federal de Minas Gerais, 2005. Citado na página 27.

JANG, J.-S. **Neuro-fuzzy modeling: architectures, analyses, and applications**. [S.l.]: University of California, Berkeley, 1992. Citado na página 27.

JARDINE, A.; LIN, D.; BANJEVIC, D. A review on machinery diagnostics and prognostics implementing condition-based maintenance. **Mechanical Systems and Signal Processing**, v. 20, p. 1483–1510, 10 2006. Citado na página 28.

JAVED, K.; GOURIVEAU, R.; LI, X.; ZERHOUNI, N. Tool wear monitoring and prognostics challenges: a comparison of connectionist methods toward an adaptive ensemble model. **Journal of Intelligent Manufacturing**, Springer, v. 29, n. 8, p. 1873–1890, 2018. Citado na página 29.

JAVED, K.; GOURIVEAU, R.; ZERHOUNI, N.; NECTOUX, P. Enabling health monitoring approach based on vibration data for accurate prognostics. **IEEE Transactions on Industrial Electronics**, v. 62, n. 1, p. 647–656, 2015. Citado na página 52.

LANDIM, R. P.; RODRIGUES, B.; SILVA, S. R.; CAMINHAS, W. M. A neo-fuzzy-neuron with real time training applied to flux observer for an induction motor. In: **Proceedings 5th Brazilian Symposium on Neural Networks (Cat. No.98EX209)**. [S.l.: s.n.], 1998. p. 67–72. Citado na página 27.

LI, X.; LIM, B.; ZHOU, J.; HUANG, S.; PHUA, S.; SHAW, K.; ER, M. Fuzzy neural network modelling for tool wear estimation in dry milling operation. In: **Annual conference of the prognostics and health management society**. [S.l.: s.n.], 2009. p. 1–11. Citado na página 56.

LI, X.; ZHANG, W.; DING, Q. Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction. **Reliability Engineering & System Safety**, v. 182, p. 208 – 218, 2019. Citado na página 29.

LIAO, L.; KÖTTIG, F. Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction. **IEEE Transactions on Reliability**, v. 63, n. 1, p. 191–207, 2014. Citado na página 28.

LIN, C.; LEE, C. S. G. Neural-network-based fuzzy logic control and decision system. **IEEE Transactions on Computers**, v. 40, n. 12, p. 1320–1336, 1991. Citado na página 27.

LUO, J.; NAMBURU, M.; PATTIPATI, K.; QIAO, L.; KAWAMOTO, M.; CHIGUSA, S. Model-based prognostic techniques [maintenance applications]. In: IEEE. **Proceedings AUTOTEST-CON 2003. IEEE Systems Readiness Technology Conference**. [S.l.], 2003. p. 330–340. Citado na página 28.

NAUCK, D.; KRUSE, R. Neuro-fuzzy systems for function approximation. **Fuzzy sets and systems**, Elsevier, v. 101, n. 2, p. 261–271, 1999. Citado na página 27.

NECTOUX, P.; GOURIVEAU, R.; MEDJAHAR, K.; RAMASSO, E.; CHEBEL-MORELLO, B.; ZERHOUNI, N.; VARNIER, C. PRONOSTIA : An experimental platform for bearings accelerated degradation tests. In: **IEEE International Conference on Prognostics and Health Management, PHM'12**. Denver, Colorado, United States: IEEE Catalog Number : CPF12PHM-CDR, 2012. p. 1–8. Citado na página 51.

PAN, Z.; MENG, Z.; CHEN, Z.; GAO, W.; SHI, Y. A two-stage method based on extreme learning machine for predicting the remaining useful life of rolling-element bearings. **Mechanical Systems and Signal Processing**, Elsevier, v. 144, p. 106899, 2020. Citado na página 52.

PHM Society. **PHM data challenge 2010**. 2010. <<https://www.phmsociety.org/competition/phm/10>>. Accessed: 2020-04-30. Citado na página 56.

QU, J.; LIU, F.; MA, Y.; FAN, J. A neural-network-based method for rul prediction and soh monitoring of lithium-ion battery. **IEEE Access**, v. 7, p. 87178–87191, 2019. Citado na página 29.

SAHA, B.; GOEBEL, K. Battery data set. **NASA Ames Prognostics Data Repository**, NASA Ames Research Center, Moffett Field, CA, 2007. Disponível em: <<https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository>>. Citado na página 60.

SILVA, A. M.; CAMINHAS, W.; LEMOS, A.; GOMIDE, F. A fast learning algorithm for evolving neo-fuzzy neuron. **Applied Soft Computing**, Elsevier, v. 14, p. 194–209, 2014. Citado nas páginas 27, 31 e 32.

SORJAMAA, A.; LENDASSE, A. Time series prediction using dirrec strategy. In: **Esann**. [S.l.: s.n.], 2006. v. 6, p. 143–148. Citado na página 50.

SOUALHI, A.; MEDJAHAR, K.; CELRC, G.; RAZIK, H. Prediction of bearing failures by the analysis of the time series. **Mechanical Systems and Signal Processing**, Elsevier, v. 139, p. 106607, 2020. Citado nas páginas 28 e 49.

VACHTSEVANOS, G.; LEWIS, F.; ROEMER, M.; HESS, A.; WU, B. Intelligent fault diagnosis and prognosis for engineering systems: Methods and case studies. v. 13, 01 2006. Citado na página 28.

YAMAKAWA, T.; KUSANAGI, H.; UCHINO, E.; MIKI, T. A new effective learning algorithm for a neo fuzzy neuron model. In: KOREAN INSTITUTE OF INTELLIGENT SYSTEMS. **Proceedings of the Korean Institute of Intelligent Systems Conference**. [S.l.], 1993. p. 1017–1020. Citado na página 32.

YAMAKAWA, T.; UCHINO, E.; MIKI, T.; KUSANAGI, H. A neo fuzzy neuron and its applications to system identification and predictions to system behavior. **Proceedings of the International Conference on Fuzzy Logic and Neural Networks**, v. 1, p. 477–483, 1992. Citado nas páginas 27 e 31.