



**UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL**

**CURSO DE GRADUAÇÃO EM ENGENHARIA CIVIL
TRABALHO DE CONCLUSÃO DE CURSO**

**AVALIAÇÃO DO DESEMPENHO DE REDES NEURAIS ARTIFICIAIS NA
PREVISÃO DE VAZÕES CRÍTICAS A PARTIR DE SÉRIES CURTAS DE DADOS**

ELIAS ALVES DO SANTOS JÚNIOR

João Pessoa – PB

Setembro de 2019

ELIAS ALVES DO SANTOS JÚNIOR

**AVALIAÇÃO DO DESEMPENHO DE REDES NEURAIS ARTIFICIAIS NA
PREVISÃO DE VAZÕES CRÍTICAS A PARTIR DE SÉRIES CURTAS DE DADOS**

Trabalho de Conclusão de Curso apresentado
à Universidade Federal da Paraíba como parte
dos requisitos para obtenção do título de
Bacharel em Engenharia Civil.

Orientador: Prof. Gustavo Barbosa Lima da
Silva

João Pessoa – PB

Setembro de 2019

Catálogo na publicação
Seção de Catalogação e Classificação

J95a Júnior, Elias Alves Dos Santos.

Avaliação do Desempenho de Redes Neurais Artificiais na
Previsão de Vazões Críticas a Partir de Séries Curtas
de Dados / Elias Alves Dos Santos Júnior. - João
Pessoa, 2019.

87 f. : il.

Orientação: Gustavo Barbosa Lima da Silva.
Monografia (Graduação) - UFPB/CT.

1. Redes neurais artificiais. 2. Previsão de vazões
críticas. 3. Séries curtas de dados. I. Silva, Gustavo
Barbosa Lima da. II. Título.

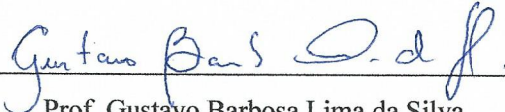
UFPB/BC

FOLHA DE APROVAÇÃO

ELIAS ALVES DOS SANTOS JÚNIOR

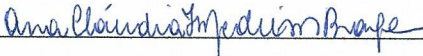
AVALIAÇÃO DO DESEMPENHO DE REDES NEURAIS ARTIFICIAIS NA PREVISÃO DE VAZÕES CRÍTICAS A PARTIR DE SÉRIES CURTAS DE DADOS

Trabalho de Conclusão de Curso em 26/09/2019 perante a seguinte Comissão Julgadora:



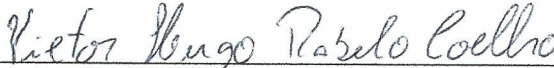
Prof. Gustavo Barbosa Lima da Silva
Departamento de Engenharia Civil e Ambiental do CT/UFPB

Aprovado



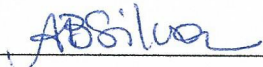
Prof.ª Ana Cláudia Fernandes Medeiros Braga
Departamento de Engenharia Civil e Ambiental do CT/UFPB

Aprovado



Prof. Victor Hugo Rabelo Coelho
Departamento de Engenharia Civil e Ambiental do CT/UFPB

Aprovado



Prof.ª Andrea Brasiliano Silva
Matrícula Siape: 1549557
Coordenadora do Curso de Graduação em Engenharia Civil

Dedico este trabalho aos meus pais e minhas duas irmãs, que durante todos esses anos de graduação sempre se fizeram presentes em minha jornada oferecendo o melhor que um ser humano pode receber, o amor.

Agradecimentos

Muitas vezes movido pela curiosidade, fui em busca dos agradecimentos dos TCCs, teses e dissertações que li durante a graduação, e ficava a imaginar o que escrever quando chegasse a minha vez. Esse momento chegou, a dúvida ainda persiste, mas as palavras fluem aos poucos. Certa vez, em um desses trabalhos, li que a folha dos agradecimentos sempre é a última a ser escrita, e nesse trabalho não foi diferente. Mas ela não é só a última, como também a mais difícil.

Como é de costume, começo os agradecimentos a quem se fez presente em todos os momentos, dos mais fáceis aos mais difíceis, nas alegrias e nas tristezas, Deus. Sua presença em mim é maior a cada dia.

Agradeço secundamente aos meus pais, que foram e continuam sendo as minhas referências e pelos quais tenho inestimado apreço.

Agradeço às minhas irmãs que contribuíram de forma ímpar para que eu chegasse aqui e a todos os meus familiares, em especial a minha avó, Maria Vieira, por todo amor recebido.

Agradeço também aos amigos que tive ao longo do curso pelos bons momentos e pelas maratonas de estudo, pois sem eles esse momento não teria se concretizado.

E a todos os professores que tive ao longo da vida, meu muito obrigado, em especial ao meu orientador, Gustavo Barbosa, por ter aceito a tarefa de orientar esse trabalho e por todo o aprendizado a mim proporcionado nesse último semestre da minha vida acadêmica.

“Concordamos convosco em que a experimentação é necessária; que a pesquisa intelectual é o ponto de partida dos grandes empreendimentos evolutivos; que a curiosidade respeitável é mãe da ciência realizadora; que todo e qualquer processo de conhecimento exige campo de observação e trabalho, como é imprescindível o material didático, nas escolas mais simples.”

Chico Xavier

RESUMO

A operação de reservatórios de usinas hidrelétricas necessita da utilização de modelos que busquem prever as vazões afluentes aos reservatórios. Diversos modelos hidrológicos vem sendo utilizados, contudo eles necessitam de muitas variáveis e de séries históricas longas para sua calibração. As Redes Neurais Artificiais (RNAs) surgem como uma alternativa a esses modelos. O presente estudo tem como objetivo avaliar o desempenho de redes neurais que foram calibradas com séries curtas de dados quando aplicadas na previsão de vazões críticas. A série histórica utilizada possui dados de vazões diárias afluentes ao reservatório da Usina de Chavantes de 1931 até 2007, totalizando 77 anos de medições. Foram treinadas um total de 15 RNAs, variando a porcentagem dos grupos de treinamento e os horizontes de previsão. As porcentagens da série utilizadas na calibração das redes foram de 10, 20, 30, 40 e 50% e foram feitas previsões para três horizontes futuros distintos, 1, 3 e 7 dias. A performance dos modelos criados foram avaliados por meio de 3 parâmetros estatísticos: o erro quadrático médio (RMSE), o erro médio absoluto relativo (MARE) e o coeficiente de eficiência (CE). Ao final foi possível perceber que os melhores resultados ocorreram para os menores horizontes de previsões futuros, contudo nenhuma das redes foi capaz de determinar de forma satisfatória as vazões críticas, visto que as aproximações obtidas apresentaram grandes oscilações em relação aos dados observados.

Palavras-chave: redes neurais artificiais, previsão de vazões críticas, séries curtas de dados.

ABSTRACT

Hydroelectric facilities operation requires the use of models to predict streamflow. Several phenomenological models have been used, however they require many inputs and long historical series to be calibrated. Artificial Neural Networks (ANN) are an alternative to these models. The present study aims to evaluate the performance of neural networks which were calibrated using short time series in order to predict streamflow peaks. It was used a daily historical streamflow data from Chavantes Dam ranging from 1931 to 2007, totaling 77 years. A total of 15 ANNs were trained, varying the percentage of the training groups and forecasting horizons. The percentage of the data used in the calibration of the networks were 10, 20, 30, 40 and 50% and predictions were made for three distinct forecasting horizons, 1, 3 and 7 days. The performance of the ANN models were evaluated using 3 performance metrics: the root mean square error (RMSE), the mean absolute relative absolute error (MARE) and the coefficient of efficiency (CE). It was possible to realize that the best results were obtained when using smaller time horizons, however none of the networks were able to satisfactorily determine the streamflow peaks.

Keywords: artificial neural networks, peak flow forecast, short time series.

LISTA DE FIGURAS

Figura 1 - Matriz elétrica brasileira do ano de 2018	15
Figura 2 - Elementos de um neurônio biológico	20
Figura 3 – Entradas externas chegando em um neurônio.....	22
Figura 4 - Neurônio com terminal de saída	22
Figura 5 - Ativação do neurônio	24
Figura 6 – Neurônio artificial emitindo saída	24
Figura 7 - Função de ativação degrau.....	25
Figura 8- Função de ativação degrau bipolar.....	26
Figura 9 - Função de ativação rampa simétrica	26
Figura 10 - Função de ativação logística.....	27
Figura 11 - Função de ativação tangente hiperbólica.....	28
Figura 12- Função de ativação gaussiana	28
Figura 13 - Função de ativação linear	29
Figura 14 - Simbologia de grafos e vértices	30
Figura 15 – Exemplo de rede feedforward com apenas uma camada	31
Figura 16 - Exemplo de rede MLP.....	32
Figura 17 - Exemplo de rede com retroalimentação.....	33
Figura 18 - Exemplo de rede perceptron multicamadas.....	37
Figura 19 - Exemplo de simulação sem “overfitting” e com “overfitting”	39
Figura 20 - Unidade de gestão de recursos hídricos Paranapanema	46
Figura 21 - Série histórica de vazões utilizada	47
Figura 22 - Grupos de dados para RNA-10.....	49
Figura 23 - Exemplo de vetor fornecido a rede	51
Figura 24 - Treinamento RNA-10 _{1D}	58
Figura 25 - Treinamento RNA-20 _{1D}	59
Figura 26 - Treinamento RNA-30 _{1D}	59
Figura 27 - Treinamento RNA-40 _{1D}	60
Figura 28 - Treinamento RNA-50 _{1D}	60
Figura 29 - Simulação RNA-10 _{1D}	62
Figura 30 - Simulação RNA-20 _{1D}	63

Figura 31 - Simulação RNA-30 _{1D}	63
Figura 32 - Simulação RNA-40 _{1D}	64
Figura 33 - Simulação RNA-50 _{1D}	64
Figura 34 - Treinamento RNA-10 _{3D}	67
Figura 35 - Treinamento RNA-20 _{3D}	67
Figura 36 - Treinamento RNA-30 _{3D}	68
Figura 37 - Treinamento RNA-40 _{3D}	68
Figura 38 - Treinamento RNA-50 _{3D}	69
Figura 39 - Simulação RNA-10 _{3D}	70
Figura 40 - Simulação RNA-20 _{3D}	70
Figura 41 - Simulação RNA-30 _{3D}	71
Figura 42 - Simulação RNA-40 _{3D}	71
Figura 43 - Simulação RNA-50 _{3D}	72
Figura 44 - Treinamento RNA-10 _{7D}	74
Figura 45 - Treinamento RNA-20 _{7D}	74
Figura 46 - Treinamento RNA-30 _{7D}	75
Figura 47 - Treinamento RNA-40 _{7D}	75
Figura 48 - Treinamento RNA-50 _{7D}	76
Figura 49 - Simulação RNA-10 _{7D}	77
Figura 50 - Simulação RNA-20 _{7D}	77
Figura 51 - Simulação RNA-30 _{7D}	78
Figura 52 - Simulação RNA-40 _{7D}	78
Figura 53 - Simulação RNA-50 _{7D}	79

LISTA DE TABELAS

Tabela 1 - Parâmetros estatísticos.....	55
Tabela 2 - Quantidade de vazões superiores a uma determinada referência	56
Tabela 3 - Melhores arquiteturas e parâmetros obtidos para 1 dia de previsão futura	57
Tabela 4 - Comparação entre duas arquiteturas da RNA-50 _{1D}	61
Tabela 5 - Melhores arquiteturas e parâmetros obtidos para 3 dias de previsão futura	65
Tabela 6 - Melhores arquiteturas e parâmetros obtidos para 7 dias de previsão futura	73

LISTA DE ABREVIATURAS E SIGLAS

ARMA	Modelos Autorregressivos de Médias Móveis
ANA	Agência Nacional de Água
LM	Algoritmo de Levenberg-Marquardt
MARE	Erro Médio Absoluto Relativo
MCP	Neurônio de McCulloch e Pitts
MLP	Perceptron de Multicamadas – Multilayer Perceptron
MRLM	Modelos de Regressão Linear Múltipla
NOS	Operador Nacional do Sistema
ONS	Operador Nacional do Sistema Elétrico
RMSE	Raiz do Erro Quadrático Médio
RNA	Rede Neural Artificial

SUMÁRIO

1) INTRODUÇÃO	15
1.1. Contextualização	15
1.2. Objetivos	16
1.3. Justificativa	17
2) FUNDAMENTAÇÃO TEÓRICA	18
2.1. Séries temporais de vazões	18
2.2. Redes Neurais Artificiais	19
2.2.1. Abordagem histórica.....	19
2.2.2. Neurônio biológico.....	20
2.2.3. Neurônio artificial.....	21
2.2.4. Topologia.....	30
2.2.5. Tipos de arquitetura.....	30
2.2.5.1. Redes diretas (“ <i>feedforward</i> ”)	31
2.2.5.1.1. “ <i>Feedforward</i> ” de camada simples	31
2.2.5.1.2. “ <i>Feedforward</i> ” de camadas múltiplas.....	31
2.2.5.2. Rede com ciclos, recorrentes ou realimentadas	32
2.2.6. Aprendizado e treinamento	33
2.2.6.1. Aprendizado supervisionado.....	34
2.2.6.2. Aprendizado não supervisionado.....	35
2.2.6.3. Treinamento com reforço	36
2.2.6.4. Aprendizado por competição	36
2.2.7. Modelo de rede perceptron	36
2.2.7.1. Perceptron de uma única camada	36
2.2.7.2. Redes Perceptron de Múltiplas Camadas.....	37
2.2.7.3. Algoritmo Levenberg-Marquardt	38
2.2.7.4. Algoritmo de Levenberg-Marquardt com Regularização Bayesiana	39
3) METODOLOGIA	44
3.1. Descrição geral.....	44
3.2. Caracterização da área	45
3.3. Base de dados.....	47
3.3.1. Vazões críticas	48

3.4. Criação dos grupos	48
3.5. Modelo de Previsão	49
3.5.1. Consideração inicial	49
3.5.2. Rede adotada	49
3.5.3. Arquitetura	50
3.5.4. Função de ativação adotada	50
3.5.5. Aprendizado e treinamento da rede	51
3.5.6. Normalização dos dados	52
3.5.7. Critérios do treinamento	52
3.5.8. Avaliação das aproximações	53
4) RESULTADOS	55
4.1. Determinação das vazões críticas	55
4.2. Previsões de vazão	56
4.2.1. Previsões para 1 dia a frente	57
4.2.1.1. Treinamento	57
4.2.1.2. Simulações	62
4.2.2. Previsões para 3 dias a frente	65
4.2.2.1. Treinamento	65
4.2.2.2. Simulações	69
4.2.3. Previsões para 7 dias a frente	72
4.2.3.1. Treinamento	72
4.2.3.2. Simulações	76
5) CONCLUSÕES	79
5.1. Trabalhos futuros	81
6) REFERÊNCIAS	83

1) INTRODUÇÃO

1.1. Contextualização

A energia elétrica é uma necessidade constante na vida da humanidade estando presente desde as atividades mais ordinárias, até as atividades mais complexas. Basta olhar ao nosso redor para perceber o papel primordial de tal energia para a manutenção da sociedade atual, possibilitando o contínuo desenvolvimento. Vê, assim, que a necessidade por energia é contínua e crescente ao longo do tempo.

A energia elétrica, contudo, não está disponível de forma fácil na natureza. Sua presença pode ser observada por meio das descargas elétricas provenientes da atmosfera, mas até o momento não dispomos de tecnologias viáveis para captá-las e armazená-las quando oriundas dessa fonte. Sendo assim, a humanidade faz o uso de técnicas de conversão de energia para obtê-la, ou seja, outras fontes de energia existentes são utilizadas para a obtenção de energia cinética, a qual é convertida em energia elétrica. Diversas são as fontes utilizadas, dentre elas podemos citar: sol, movimento das ondas, ventos, fissão nuclear, combustíveis fósseis, a água, etc.

O Brasil, devido a suas características naturais, faz uso das águas dos rios como principal fonte geradora de energia elétrica (EPE, [20--]). Dessa forma, segundo BEN (2019), a matriz elétrica brasileira no ano de 2018 apresentou o seguinte aspecto, onde é possível perceber a grande participação das usinas hidrelétricas.

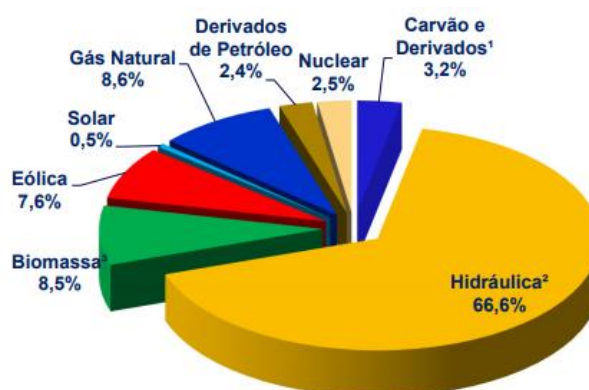


Figura 1 - Matriz elétrica brasileira do ano de 2018

Fonte: BEN (2019).

Devido à grande importância dessa fonte em nossa matriz, diversas técnicas são utilizadas com o intuito de obter maior eficiência no gerenciamento dos reservatórios de hidrelétricas. Um dos controles realizados é abertura e fechamento das comportas dos reservatórios, que necessita, de forma prévia, de dados de vazões afluentes. Assim, para obter tais dados, atualmente faz-se o uso de modelos hidrológicos com o intuito de prever as vazões futuras. Segundo Adamowski e Sun (2010), o uso de tais modelos dá-se em razão do rápido desenvolvimento e mínima necessidade de informações para a obtenção das estimativas de vazões. Ainda segundo Adamowski e Sun (2010), os modelos mais utilizados são os Modelos de Regressão Linear Múltipla (MRLM) e Modelos Autorregressivos de Médias Móveis (ARMA) e mais recentemente as Redes Neurais Artificiais (RNA) que têm sido utilizadas com esse intuito.

O potencial dessa técnica tem sido bastante explorado para prever vazões. Porém destaca-se que as aplicações de RNA no gerenciamento de reservatórios são uma pequena parcela de sua utilização.

As RNAs vem sendo empregadas em diversas outras áreas, sendo bastante útil em diversas abordagens e vale ressaltar que sua utilização não está limitada unicamente a previsão de dados, o campo de aplicação é bastante vasto e a cada dia novas utilizações são estudadas. Demuth e Beale (2004) citam algumas áreas de utilização, dentre as quais: indústria aeroespacial, na simulação de voos, detecção de falhas em espaçonaves e melhoramento do piloto automático; na indústria automotiva, nos sistemas de carros autônomos; nos bancos e operadoras de cartões de crédito, para a análise de crédito dos clientes e verificação de compras suspeitas; na defesa, para reconhecimento facial, compressão de dados, identificação de sinais e imagens; nos eletrônicos, no controle de processos; no entretenimento, para animações e efeitos especiais; dentre outros.

1.2. Objetivos

O objetivo principal desse trabalho consiste em avaliar a influência do tamanho do conjunto de dados de treinamento na previsão de vazões por meio de modelos baseados em redes neurais. Como objetivos específicos tem-se:

- Verificar a influência de diferentes horizontes de previsões para diferentes conjuntos de treinamento nos resultados da rede;
- Comparar e analisar o desempenho das simulações das vazões críticas e dos demais dados simulados em relação aos dados observados.

1.3. Justificativa

Com o desenvolvimento exponencial tecnológico que estamos vivenciando na atualidade e com o aumento da capacidade de processamento dos nossos computadores, novas tecnologias têm sido aplicadas na modelagem hidrológica aumentando a precisão dos dados simulados. As RNAs têm se mostrado como uma possibilidade promissora, necessitando de esforços nesse campo para que seja feita a avaliação do seu desempenho em diferentes cenários.

Como dito anteriormente, o Brasil tem uma grande dependência de sua rede fluvial para a geração de energia elétrica. Assim, a busca pelo aprimoramento no gerenciamento dos reservatórios das usinas hidrelétricas é constante, uma vez que isso está alinhado com o aumento da eficiência e conseqüentemente redução de custos.

Apesar de já existirem modelos que buscam prever as vazões, eles necessitam de séries históricas longas de vazões com o intuito de realizar as previsões de forma satisfatória. E levando em consideração que a disponibilidade de longos períodos de medições nem sempre está disponível, é possível perceber a importância da aplicação de RNAs com esse fim.

Assim, estudar o comportamento de uma RNA na previsão de dados de uma série histórica, a partir de um treinamento com séries curtas de dados é um cenário novo e que tem muito a contribuir na ampliação do conhecimento dessa técnica e, possivelmente, reforçar sua utilidade no gerenciamento de reservatórios.

A série histórica de vazões utilizadas nesse estudo corresponde as vazões afluentes do reservatório de Chavantes, o qual é de grande importância para a produção de energia da região Centro-Oeste e Sudeste. A usina hidroelétrica que utiliza esse reservatório é a Usina de Chavantes, a qual possui uma capacidade

instalada de 414 MW (CTG BRASIL, [20--]a), sendo de grande importância no desenvolvimento dessas regiões abastecidas.

2) FUNDAMENTAÇÃO TEÓRICA

2.1. Séries temporais de vazões

Séries temporais são um conjunto de dados correspondentes a uma ou várias características observadas de forma ordenada ao longo do tempo, com o intuito de observar correlações entre esses dados (EHLERS, 2005). Ou seja, o fator tempo é de grande importância para o estudo de uma determinada série. Esse conjunto de dados podem ser dos mais diversos tipos. Na área de recursos hídricos, é bastante comum a utilização de séries temporais de vazões.

Assim, o estudo de séries temporais pode ser empregado para alcançar diversos objetivos, um deles é a predição de uma determinada característica futura com base nos valores dessa mesma característica reportado no passado. Isso é possível devido a capacidade de observar tendências e variações sazonais ao longo do tempo, por meio de gráficos, modelos probabilísticos, análise espectral, etc (EHLERS, 2005). Nesse estudo, o objetivo de sua utilização está na predição de vazões afluentes futuras com base nas vazões afluentes registradas anteriormente.

A análise desses dados, porém, pode ser bastante complicada. Segundo Ehlers (2005), as análises estatísticas, em sua grande parte, não levam em consideração a correlação entre os dados reportados. Ou seja, para que essa ligação entre os elementos seja considerada, faz-se necessário a utilização de técnicas específicas. Bressan (2004), cita uma grande quantidade de estudos que foram desenvolvidos com o intuito de avaliar a performance de modelos aplicados na tomada de decisões construídos com base em séries temporais. Isso mostra a importância que dados históricos de operação tem no gerenciamento, uma vez que eles são capazes de fornecer informações para a construção de modelos, dentre eles os que fazem uso de redes neurais, que podem ser aplicados em diversas áreas e com diversos objetivos.

2.2. Redes Neurais Artificiais

2.2.1. Abordagem histórica

As redes neurais artificiais são modelos matemáticos que possuem uma grande capacidade de processamento e que se baseiam na estrutura fisiológica do cérebro humano tentando reproduzir suas funções. Para entender como as redes neurais funcionam, é importante conhecer os estudos que levaram ao surgimento dessa tecnologia.

O estudo dessa metodologia iniciou-se com Warren McCulloch e Walter Pitts que, em 1943, publicaram um trabalho intitulado “A Logical Calculus of the Ideas Immanent in Nervous Activity”, no qual foi descrito o funcionamento de um neurônio artificial mostrando suas capacidades para a solução de problemas. Assim, surgiram uma série de estudos com o intuito simular o comportamento do cérebro humano, por meio da utilização de neurônios artificiais (BRAGA; CARVALHO; LUDERMIR, 2007).

Esses estudos deram origem a diversos tipos de regras e modelos de redes neurais que ainda são utilizadas até os dias atuais. Podemos citar Donald Hebb, Widrow e Hoff e Frank Rosenblatt, os quais foram responsáveis pela regra de Hebb, regra delta e o modelo perceptron, respectivamente (BRAGA; CARVALHO; LUDERMIR, 2007).

Contudo a aplicação de técnicas de redes neurais naquela época ainda apresentava algumas limitações, impossibilitando sua utilização na solução de certos problemas, devido a uma necessidade maior de espaço e de tempo para problemas mais complexos. Assim, em 1969 Minsky e Papert publicaram um estudo pessimista abordando esses problemas em relação ao modelo perceptron, o que resultou em uma queda nas pesquisas envolvendo redes neurais durante toda a década seguinte (BRAGA; CARVALHO; LUDERMIR, 2007; SILVA; SPATTI; FLAUZINO, 2010).

A retomada de estudos no campo das redes neurais deu-se após a publicação do artigo “Neural networks and physical systems with emergent collective computational abilities” por John Hopfield. Nesse estudo, Hopfield (1982) analisou as propriedades de redes constituídas por neurônios artificiais e mostrou a relação entre redes recorrentes auto-associativas e sistemas físicos.

Tal estudo estimulou uma volta considerável nas pesquisas sobre redes neurais em toda a comunidade científica (BRAGA; CARVALHO; LUDERMIR, 2007; SILVA; SPATTI; FLAUZINO, 2010) e desde então a Rede Neural de Hopfield tem sido bastante presente em diversas aplicações, tais como: otimização combinatória, restauração de imagem e automações na engenharia (UYKAN, 2013). Assim, junto com o desenvolvimento das tecnologias, as redes neurais vêm mostrando um grande potencial.

2.2.2. Neurônio biológico

A forma como as redes neurais se estruturam podem variar entre os tipos de redes existentes, contudo elas podem ser descritas de forma geral pelos seguintes elementos: unidades de processamento, camadas, conexões e pesos.

As unidades de processamento, que são os neurônios, se organizam em camadas, podendo existir apenas uma ou então várias, as quais realizam conexões entre si. A essas conexões são atribuídos pesos, que são determinados por meio de técnicas de aprendizado até que seja atingida uma solução desejada (BRAGA; CARVALHO; LUDERMIR, 2007).

É possível perceber a semelhança entre a estrutura apresentada acima e a forma como os neurônios biológicos se associam.

Um neurônio biológico é composto de corpo, dendritos e axônio. Uma imagem representativa pode ser vista na Figura 2.

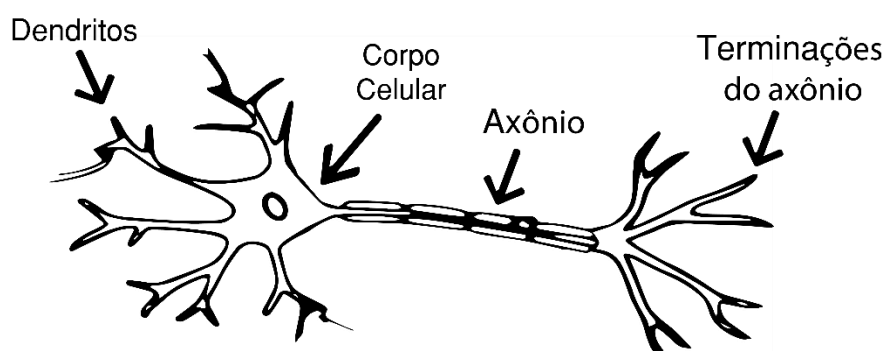


Figura 2 - Elementos de um neurônio biológico

Fonte: Adaptado de PIXABAY (2019).

Os dendritos recebem os impulsos emitidos por outros neurônios até que os mesmos cheguem ao corpo celular do neurônio receptor para que seja processado. Após o processamento, o neurônio receptor emite ou não novos impulsos por meio do seu axônio, dando continuidade ao impulso nervoso. Contudo, os impulsos não são transmitidos de forma deliberada, existem um controle da informação que é realizado por meio das sinapses, as quais fazem a conexão entre os neurônios. As sinapses possuem a função de realizar a transmissão das informações que irão prosseguir para os neurônios seguintes de forma controlada.

Esse controle é feito por meio da excitação e inibição do neurônio, ou seja, se o conjunto das sinapses excitatórias e sinapses inibitórias oriundas dos neurônios pré-sinápticos for suficiente para estimular de forma suficiente o corpo do neurônio que as recebeu, dentro de um intervalo de tempo pequeno, um impulso será emitido desse neurônio para o dendrito do neurônio seguinte. Esse tipo de ativação realizado pelas sinapses conferem a possibilidade de adaptação da rede (BARRETO, 2002; BRAGA; CARVALHO; LUDERMIR, 2007; SILVA; SPATTI; FLAUZINO, 2010).

2.2.3. Neurônio artificial

O primeiro neurônio artificial foi proposto no estudo de McCulloch e Pitts em seu modelo MCP (BRAGA; CARVALHO; LUDERMIR, 2007), o qual tinha o objetivo de imitar a realidade biológica. Esse neurônio consiste em uma simplificação do neurônio biológico apresentado acima. Ou seja, apesar do comportamento semelhante entre estímulo, processamento e resposta, algumas simplificações foram adotadas.

O modelo do neurônio de McCulloch e Pitts (MCP) é o modelo mais simples, porém é capaz de englobar as principais características de um neurônio biológico, dessa forma ele é um dos modelos mais utilizados nas redes neurais (BARRETO, 2002; BRAGA; CARVALHO; LUDERMIR, 2007; SILVA; SPATTI; FLAUZINO, 2010).

Segundo Silva, Spatti e Flauzino (2010), os neurônios artificiais, de uma forma geral, são compostos por 7 elementos básicos, são eles: sinais de entrada, pesos sinápticos, combinação linear, limiar de ativação, potencial de ativação, função de ativação e sinal de saída.

Assim como o neurônio biológico, os neurônios artificiais possuem várias entradas, chamadas de terminais de entrada ou sinais de entrada, os quais são representados pela seguinte notação: x_1, x_2, \dots, x_n .

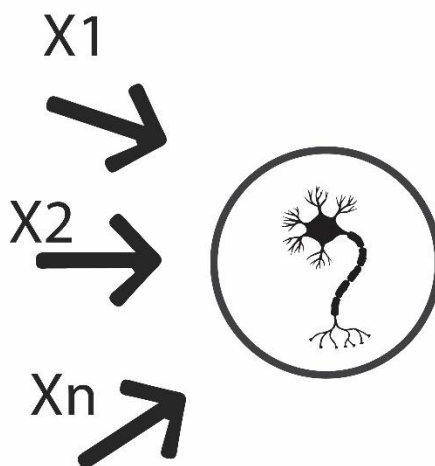


Figura 3 – Entradas externas chegando em um neurônio
Fonte: Adaptado de Hassan (2019).

E como saída, os neurônios apresentam apenas um terminal, também chamado de sinal de saída. Assim, é possível perceber que os terminais de entrada são análogos aos dendritos, enquanto o terminal de saída é análogo ao axônio.

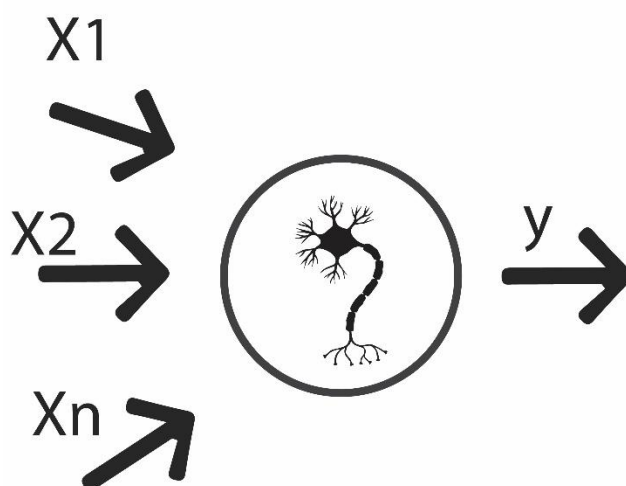


Figura 4 - Neurônio com terminal de saída
Fonte: Adaptado de Hassan (2019).

Com o objetivo de aumentar a eficiência computacional durante a fase de aprendizagem, a qual será detalhada mais adiante, as entradas fornecidas ao neurônio podem ser normalizadas (SILVA; SPATTI; FLAUZINO, 2010). Já em relação a

simulação das sinapses, a influência delas é obtida por meio da adoção de pesos, os quais atuarão sobre as entradas que chegam ao neurônio artificial.

Após a recepção dos estímulos externos, o neurônio artificial irá processar essa informação, tal processamento é realizado por meio de uma função que realiza uma soma ponderada entre as entradas e os pesos, ou seja, faz uma combinação linear das entradas ponderadas. Tal processo é chamado de confluência (BARRETO, 2002; BRAGA; CARVALHO; LUDERMIR, 2007). Para isso, as variáveis de entradas são multiplicadas pelos pesos, os quais destacam ou omitem a importância dessa informação externa para o neurônio que processará esses dados. A equação (2.1) apresentada abaixo mostra essa função utilizada.

O resultado da confluência é chamado de potencial de ativação, que é representado pela letra u . Segundo Barreto (2002), a soma ponderada não é a única possibilidade de processamento de um neurônio artificial, contudo é a mais utilizada.

$$u = \sum_{i=1}^n x_i \cdot w_i \quad (2.1)$$

Até o momento foram apresentados 5 elementos básicos dos neurônios, que são os sinais de entrada, os pesos sinápticos, combinação linear, potencial de ativação e sinal de saída.

O próximo elemento constituinte da estrutura de um neurônio artificial é o limiar de ativação θ . Esse limiar corresponde a uma variável que determina a ativação do neurônio. Assim, após a obtenção do potencial de ativação, compara-se o resultado da equação (2.1) com θ . Logo, se $u \geq \theta$ o neurônio emitirá um potencial excitatório. Caso $u < \theta$, o potencial emitido será inibitório (SILVA; SPATTI; FLAUZINO, 2010).

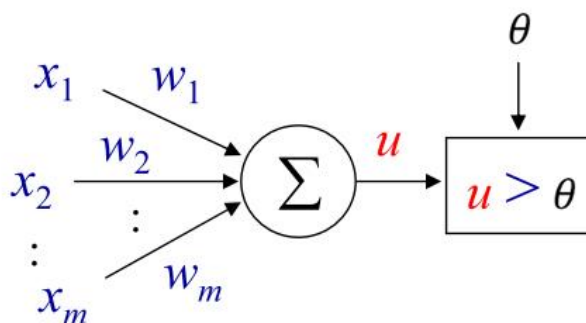


Figura 5 - Ativação do neurônio
 Fonte: Adaptado de Engel ([20--]).

Por fim, aplica-se a o valor de u na função de ativação g .

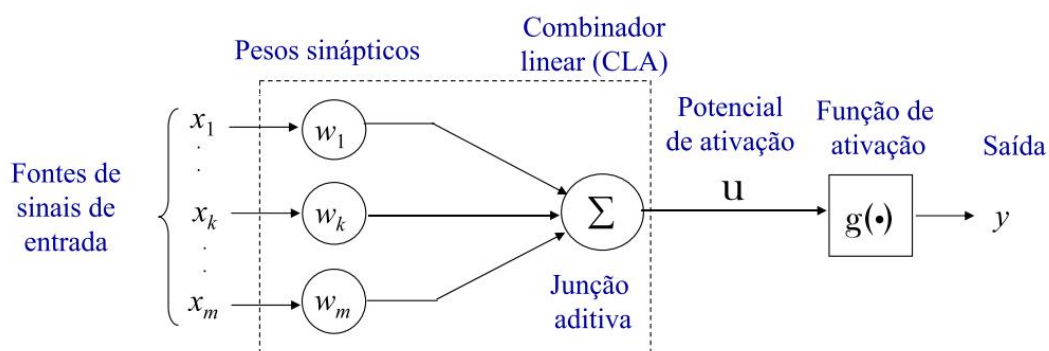


Figura 6 – Neurônio artificial emitindo saída
 Fonte: Adaptado de Engel ([20--]).

Essa função de ativação é responsável por limitar a resposta do neurônio dentro de um determinado intervalo, descrito pela imagem da função utilizada no modelo (SILVA; SPATTI; FLAUZINO, 2010). Existem diversos tipos de função de ativação possíveis. A escolha irá depender do propósito da rede neural. Assim, dependendo da situação, uma função pode ser mais adequada do que outra.

As funções de ativação podem ser classificadas em dois grupos principais, funções parcialmente diferenciáveis e funções totalmente diferenciáveis. O conceito de cada um desses grupos citado por Silva, Spatti e Flauzino (2010), os exemplos de funções pertencentes a eles, suas equações e suas representações gráficas podem ser vistas abaixo.

I. Funções parcialmente diferenciáveis:

São funções que possuem algum(ns) ponto(s), nos quais a derivada de primeira ordem é inexistente. Alguns exemplos são: função degrau, função degrau bipolar e função rampa simétrica.

- a. Função degrau, também conhecida como “*heavyside*”. Sua imagem pode assumir dois valores, 1 ou 0.

$$g(u) = \begin{cases} 1 & \text{se } u \geq 0 \\ 0 & \text{se } u < 0 \end{cases} \quad (2.2)$$

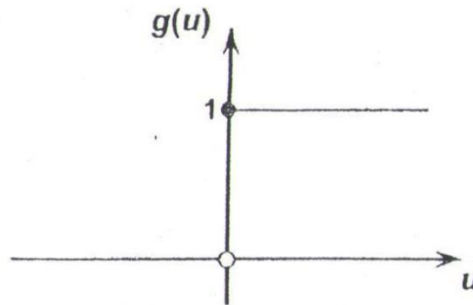


Figura 7 - Função de ativação degrau
Fonte: Silva, Spatti e Flauzino (2010).

- b. Função degrau bipolar ou sinal, tem um gráfico semelhante ao da função degrau. Contudo, pode assumir o valor 1, 0 ou -1 quando o potencial de ativação for maior que zero, igual a zero ou menor que zero, respectivamente.

$$g(u) = \begin{cases} 1, & \text{se } u > 0 \\ 0, & \text{se } u = 0 \\ -1, & \text{se } u < 0 \end{cases} \quad (2.3)$$

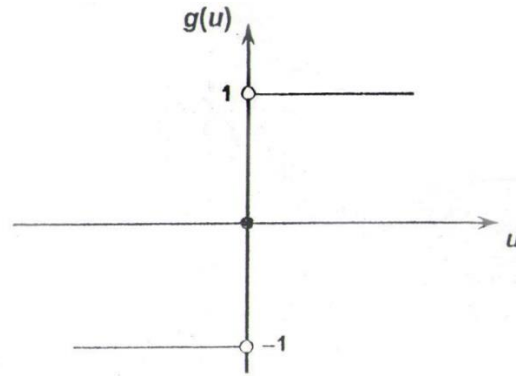


Figura 8- Função de ativação degrau bipolar

Fonte: Silva, Spatti e Flauzino (2010).

- c. Função rampa simétrica, quando definimos valores constantes para faixas específicas. Ou seja, para valores acima de determinados limites, a função retorna o valor desses limites, mas para valores intermediários, a função retorna o próprio valor de entrada, o próprio potencial de ativação.

$$g(u) = \begin{cases} a, & \text{se } u > a \\ u, & \text{se } -a \leq u \leq a \\ -a, & \text{se } u < -a \end{cases} \quad (2.4)$$

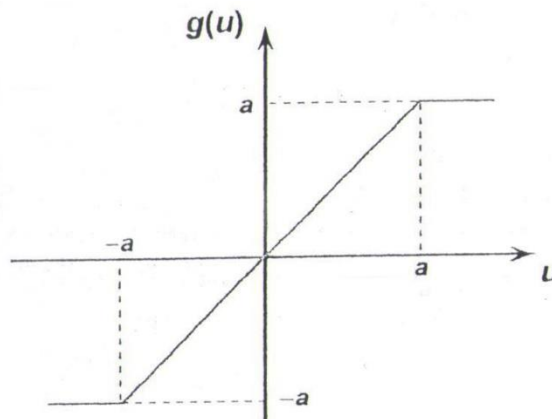


Figura 9 - Função de ativação rampa simétrica

Fonte: Silva, Spatti e Flauzino (2010).

II. Funções totalmente diferenciáveis

Para esse grupo de funções, as derivadas de primeira ordem existem em todo o domínio da função utilizada. Alguns exemplos de função de ativação utilizadas que

pertencem a esse grupo são: função logística, tangente hiperbólica, gaussiana e função linear.

- a. Função logística: uma característica importante dessa função é que a imagem dela corresponde ao intervalo compreendido entre 0 e 1 e sua curva possui um formato semelhante a um S. O seu formato é de certa forma semelhante ao da função degrau, porém a função logística possui derivada de primeira ordem em todo o seu domínio, ao contrário da função degrau. Essa função é do tipo sigmoide. Segundo Braga, Carvalho e Ludermir (2007), essa é uma das funções de ativação mais utilizadas.

$$g(u) = \frac{1}{1 + e^{-\beta \cdot u}} \quad (2.5)$$

Onde,

- β : constante real associada a inclinação da curva.

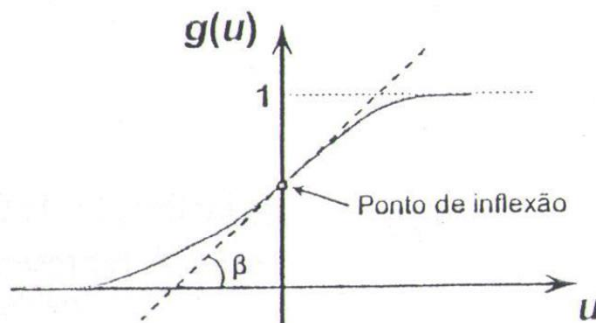


Figura 10 - Função de ativação logística

Fonte: Silva, Spatti e Flauzino (2010).

- b. Função tangente hiperbólica: também é uma função sigmoide e seu gráfico é semelhante ao da função logística. Contudo, sua imagem pode assumir valores reais negativos, uma vez que ela está compreendida entre -1 e 1.

$$g(u) = \frac{1 - e^{-\beta \cdot u}}{1 + e^{-\beta \cdot u}} \quad (2.6)$$

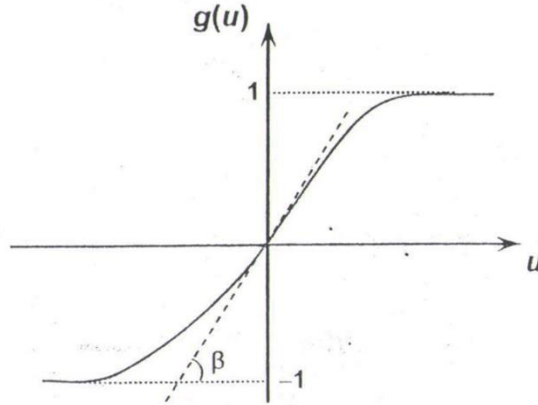


Figura 11 - Função de ativação tangente hiperbólica
Fonte: Silva, Spatti e Flauzino (2010).

- c. Função gaussiana: retorna saídas iguais para valores de potencial de ativação que distem igualmente do centro.

$$g(u) = e^{-\frac{(u-c)^2}{2\sigma^2}} \quad (2.7)$$

Onde,

- c: parâmetro que define o centro da função gaussiana;
- σ : desvio padrão associado.

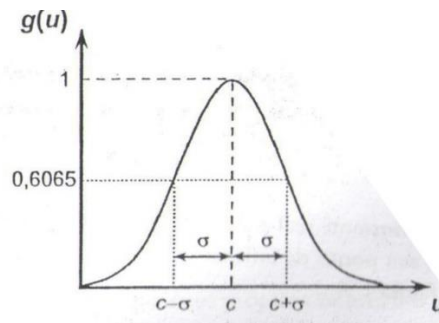


Figura 12- Função de ativação gaussiana
Fonte: Silva, Spatti e Flauzino (2010).

- d. Função linear, também chamada de função identidade. Ao utilizar essa função, a saída do neurônio é a mesma do potencial de ativação.

$$g(u) = u \quad (2.8)$$

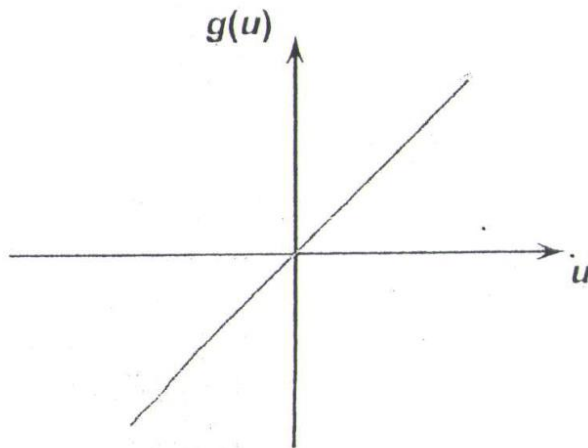


Figura 13 - Função de ativação linear

Fonte: Silva, Spatti e Flauzino (2010).

Assim, podemos resumir os elementos básicos de um neurônio artificial da seguinte forma:

- Sinais de entrada: informações que vem do meio externo;
- Pesos sinápticos: servem para ponderar os sinais de entrada de um neurônio;
- Combinação linear: combina todos os potenciais de entrada já ponderados, obtendo-se o potencial de ativação por meio de uma função linear;
- Limiar de ativação (θ): variável que representa o limiar para ativar o neurônio;
- Potencial de ativação (u): resultado da confluência;
- Função de ativação: limita a saída do neurônio dentro de um determinado intervalo;
- Sinal de saída ($g(u)$): valor final produzido pelo neurônio.

2.2.4. Topologia

Foram apresentados os elementos de processamento de um RNA, ou seja, os neurônios. Agora será apresentada a rede composta por esses elementos.

Uma RNA é composta por neurônios interligados entre si por meio de conexões sinápticas. Nessa rede, alguns neurônios têm comunicação direta com o exterior, recebendo informações externas, são os neurônios de entrada. Alguns neurônios recebem informações de outros neurônios, alterando-as e retornando a saída da rede, são os neurônios de saída.

Por fim, existem neurônios que não são nem de entrada e nem de saída, eles são chamados de neurônios internos ou "*hidden*". Vale salientar a importância desses últimos, uma vez que sem eles é impossível uma RNA resolver problemas não linearmente separáveis (BRAGA; CARVALHO; LUDERMIR, 2007; SILVA; SPATTI; FLAUZINO, 2010).

Para representar a arquitetura de uma rede neural utiliza-se a simbologia de grafos e vértices, em que os vértices são os neurônios e os grafos representam a conexão entre eles, como pode ser visto na Figura 14.

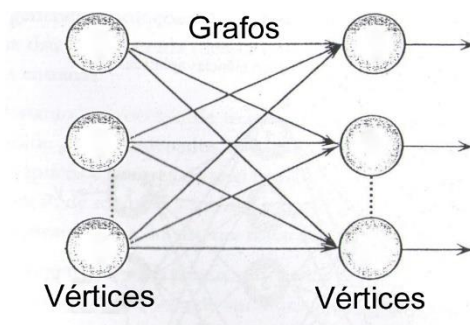


Figura 14 - Simbologia de grafos e vértices

Fonte: Adaptado de Silva, Spatti e Flauzino (2010).

2.2.5. Tipos de arquitetura

Barreto (2002) e Silva, Spatti e Flauzino (2010) comentam os tipos de arquitetura de redes neurais mais utilizados, os quais serão apresentados abaixo.

2.2.5.1. Redes diretas (“*feedforward*”)

Nesse tipo de arquitetura os grafos não têm ciclos, ou seja, vértices posteriores não se conectam a vértices anteriores. Além disso os neurônios são organizados em camadas. Dessa forma, essa arquitetura também é conhecida como redes em camadas.

2.2.5.1.1. “*Feedforward*” de camada simples

Nessa rede, os neurônios que se comunicam com o meio externo ficam localizados na camada de entrada, e os neurônios que retornam a saída da rede ficam localizados na camada de saída. Pelo fato de ser uma rede de camada simples, isso implica que não há nenhuma camada interna, apenas uma camada de entrada e uma camada de saída. É possível perceber que o número de saídas da rede será em função da quantidade de neurônios da camada de saída, ou seja, para cada neurônio uma saída será retornada. Esse tipo de rede é adequado para classificar padrões e filtragem linear. Um exemplo de uma rede dessa pode ser vista na Figura 15.

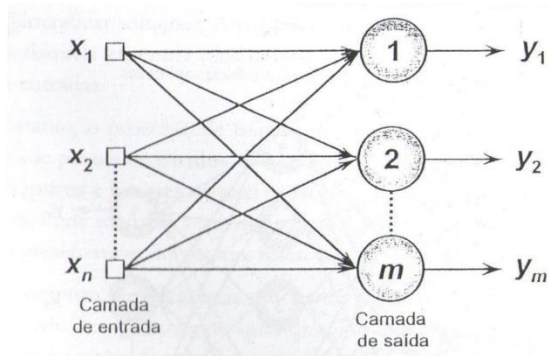


Figura 15 – Exemplo de rede feedforward com apenas uma camada

Fonte: Silva, Spatti e Flauzino (2010).

2.2.5.1.2. “*Feedforward*” de camadas múltiplas

Da mesma forma que as redes com arquitetura “*feedforward*” de camada simples, essa arquitetura não possui ciclos. Contudo, além da camada de entrada e saída existem também no mínimo uma camada interna, em que ficam localizados os neurônios internos responsáveis pela maior parte do processamento da rede. O

número de camadas a ser empregado na rede irá depender do problema a ser resolvido, da quantidade de dados disponíveis como também da qualidade dos mesmos.

Esse tipo de rede, dependendo da quantidade de neurônios existentes, são capazes de aproximar qualquer função não-linear. Além dessa aplicação, existem inúmeros outros campos que a utilização de redes de camadas múltiplas pode ser empregada, por exemplo na identificação de sistemas, otimização, robótica, controle de processos, etc.

Segundo Barreto (2002), essas redes são bem populares em razão de ter métodos de aprendizado bem difundidos e de fácil implementação. Um dos métodos de aprendizado mais utilizados é o “*backpropagation*”. Alguns exemplos de rede que utilizam essa arquitetura são: Perceptron de múltiplas camadas (MLP) e redes de base radial.

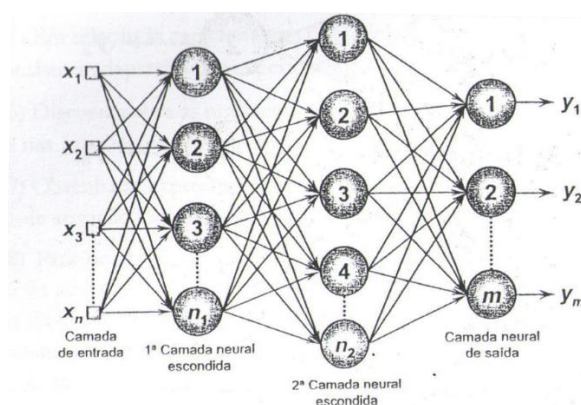


Figura 16 - Exemplo de rede MLP

Fonte: Silva, Spatti e Flauzino (2010).

2.2.5.2. Rede com ciclos, recorrentes ou realimentadas

Nesse tipo de arquitetura, diferentemente das redes diretas, existe pelo menos um ciclo de realimentação dos neurônios, ou seja, os neurônios recebem um “*feedback*” das respostas emitidas anteriormente, aperfeiçoando-as a cada novo ciclo. Dessa forma, a rede é capaz de fazer um processamento dinâmico e armazenar informações que estão atreladas a variável tempo e/ou ordenadas de forma sequencial (VON ZUBEN; ATTUX, 2007). Ou seja, esse tipo de rede é importante para aplicações que utilizam séries temporais. Alguns exemplos de redes que utilizam essa

arquitetura é a rede de Hopfield e a rede Perceptron multicamadas com retroalimentação (HOPFIELD, 1982; SILVA; SPATTI; FLAUZINO, 2010).

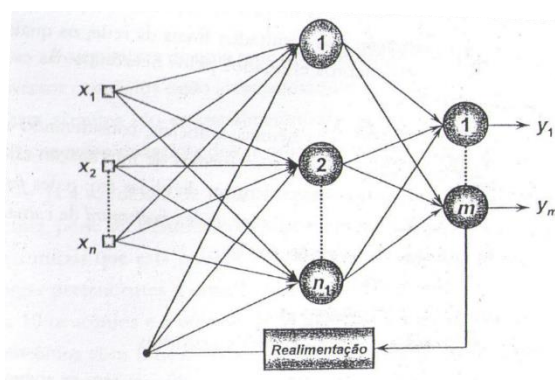


Figura 17 - Exemplo de rede com retroalimentação

Fonte: Silva, Spatti e Flauzino (2010).

2.2.6. Aprendizado e treinamento

O aprendizado de uma rede neural ocorre por meio de passos ordenados, o que é chamado de algoritmo de aprendizagem, em que exemplos são fornecidos a rede com o objetivo de extrair padrões entre as entradas e saídas. Dessa forma, permite-se generalizar as soluções para o problema apresentado e fazer extrapolações. Vale salientar que existem diversos tipos de algoritmos que podem ser empregados e basicamente a diferença entre eles está na forma como os pesos são ajustados ao longo do treinamento (SILVA; SPATTI; FLAUZINO, 2010).

O aprendizado consiste basicamente em duas fases, a primeira, na extração de informações e busca por padrões e a segunda, um processo iterativo de ajuste de pesos (BRAGA; CARVALHO; LUDERMIR, 2007). Dessa forma, divide-se o conjunto dos exemplos em dois grupos, um grupo de treinamento e um grupo de teste.

O grupo de treinamento é utilizado para definir os pesos enquanto o conjunto de teste é utilizado para verificar o desempenho da rede, possibilitando determinar se o grau de precisão obtido por uma dada solução está satisfatório ou não (SILVA; SPATTI; FLAUZINO, 2010).

Silva, Spatti e Flauzino (2010) afirmam que o conjunto de treinamento geralmente compreende de 60 a 90% dos dados disponíveis. Assim, o segundo grupo, o que tem os dados que serão utilizados na simulação, tem cerca de 10 a 40% da

amostra. Além desses percentuais citados como referência, outro aspecto que deve ser levado em consideração na definição do tamanho dos grupos é a caracterização estatísticas dos dados.

Braga, Carvalho e Ludermir (2007) e Silva, Spatti e Flauzino (2010) citam e comentam os tipos de aprendizado existentes, que serão mostrados a seguir. Os tipos de aprendizados podem ser divididos em dois grupos: aprendizado supervisionado e aprendizado não supervisionado. Ainda existem dois casos particulares desses dois grupos, que são respectivamente, aprendizado por reforço e aprendizado por competição.

2.2.6.1. Aprendizado supervisionado

Nesse tipo de aprendizado existe uma figura simbólica de um “professor”, que é responsável por direcionar o aprendizado da rede. Dessa forma, caracteriza-se como um caso de interferência indutiva pura. Esse tipo de aprendizado iniciou-se com Donald Hebb em 1949, que se inspirou em observações neurofisiológicas em sua formulação (SILVA; SPATTI; FLAUZINO, 2010).

Para que esse método possa ser empregado, necessita-se ter conhecimento das entradas e das saídas desejadas, formando pares entrada/saída. Dessa forma, esses pares são fornecidos à rede, gerando as primeiras estimativas daquela. À medida que os resultados das previsões são gerados, comparações são feitas entre a resposta obtida e a resposta desejada, ajustando os pesos e limiares.

A solução será dada como alcançada quando a defasagem entre a saída obtida e a correta estiver dentro de um intervalo de erro aceitável ou então algum outro parâmetro de parada, definido pelo operador da rede, tenha sido atingido. Para calcular esse erro utiliza-se geralmente a função da soma dos erros quadráticos de todas as saídas, e busca minimizar essa função interpretando-a como uma função de custo. Alguns dos algoritmos de aprendizado existentes pertencentes a esse grupo são: a regra delta e o “*backpropagation*”.

O primeiro modelo de rede que utilizou esse tipo de aprendizado foi o “*perceptron*” e Braga, Carvalho e Ludermir (2007) mencionam que a desvantagem

desse tipo de aprendizado está na dependência do professor para o treinamento, uma vez que a rede é incapaz de aprender novas estratégias sozinha.

Além disso, o aprendizado supervisionado pode ser de dois tipos, off-line e on-line, que corresponde a uma característica de como o aprendizado pode ocorrer.

No aprendizado off-line só ocorrerão ajustes nos pesos da rede após a apresentação de todo o conjunto de dados de treinamento. Ou seja, após todas as entradas serem fornecidas a rede, calcula-se o desvio entre os valores obtidos e os valores desejados. Assim, o ajuste dos pesos só é realizado no final de cada época. Nessa abordagem os dados não mudam durante o treinamento, logo a solução obtida é fixa. Portanto, se novos dados forem inseridos, um novo treinamento deve ser feito.

No aprendizado on-line, por sua vez, os ajustes são realizados após a apresentação de cada amostra e o conjunto de dados muda de forma constante, o que faz com que a rede esteja em contínua adaptação. Esse tipo de aprendizado é adequado para problemas nos quais o comportamento que se deseja generalizar varia rapidamente ao longo do tempo.

2.2.6.2. Aprendizado não supervisionado

Para esse tipo de aprendizagem a figura do professor inexistente, ou seja, as saídas desejadas não são conhecidas durante a fase de treinamento.

Braga, Carvalho e Ludermir (2007) afirmam que essa forma de aprender é semelhante aos sistemas biológicos. Tal semelhança está na capacidade de identificar subgrupos (“*clusters*”) nos dados fornecidos e nesses subgrupos detectar padrões. Vale salientar que somente as entradas estão disponíveis, e que esse tipo de aprendizado só pode ser implementado em dados que possuem redundância, caso contrário será impossível chegar a uma solução.

Para facilitar o aprendizado da rede, o operador, quando tem um certo conhecimento dos dados que estão sendo utilizados, pode limitar a quantidade de clusters a serem criados pela rede.

2.2.6.3. Treinamento com reforço

Esse tipo de aprendizado pode ser visto como um caso particular do aprendizado supervisionado. Nessa variação, a única resposta que a rede recebe é se a aproximação alcançada por ela é satisfatória ou não, sem qualquer informação de como melhorar o desempenho. Ou seja, o treinamento é realizado por tentativa e erro com um mapeamento das entradas e das saídas. Caso a aproximação da rede esteja sendo satisfatória, ocorrerão incrementos nos pesos com o intuito de reforçar esse aprendizado detectado, potencializando as saídas e vice-versa. Silva, Spatti e Flauzino (2010) afirmam que diversos algoritmos desse tipo são baseados em métodos estocásticos.

2.2.6.4. Aprendizado por competição

Esse algoritmo é um caso particular do aprendizado não supervisionado, no qual os pesos da rede são atualizados por meio de uma competição entre as unidades de saída. Assim, durante o treinamento as unidades que se mostrarem menos ativas serão inibidas e as unidades que se mostrarem mais ativas ficarão cada vez mais fortes, até que sobre somente uma única unidade de saída no final, que será a vencedora da competição.

2.2.7. Modelo de rede perceptron

2.2.7.1. Perceptron de uma única camada

O modelo perceptron foi proposto por Rosenblatt em 1958. Tal modelo consiste em uma estrutura de rede composta de neurônios do tipo MCP, que são as unidades básicas de processamento. Além da estrutura apresentada, tal modelo foi o primeiro a introduzir o conceito de aprendizado de redes neurais (BRAGA; CARVALHO; LUDERMIR, 2007).

O perceptron é um modelo de rede com arquitetura “*feedforward*” com aprendizado supervisionado, ou seja, durante o treinamento o valor das respostas desejadas já é conhecido pela rede e não existem ciclos de realimentação.

O poder desse modelo foi comprovado alguns anos depois, em 1962, quando o seu criador conseguiu provar que o perceptron de uma única camada é capaz de solucionar qualquer problema linearmente separável (BRAGA; CARVALHO; LUDERMIR, 2007).

2.2.7.2. Redes Perceptron de Múltiplas Camadas

Com o passar do tempo o modelo foi aperfeiçoado, surgindo o perceptron de múltiplas camadas (MLP), o qual é capaz de solucionar problemas mais complexos.

Esse novo modelo proposto possui uma arquitetura composta de pelo menos uma camada intermediária de neurônios. Assim, em uma rede desse tipo estão presentes no mínimo duas camadas, uma intermediária e uma de saída. Devido a presença da camada intermediária, esse modelo é capaz de aproximar qualquer função matemática (BRAGA; CARVALHO; LUDERMIR, 2007).

E igual a seu modelo anterior, o MLP também possui uma arquitetura “*feedforward*” com aprendizado supervisionado.

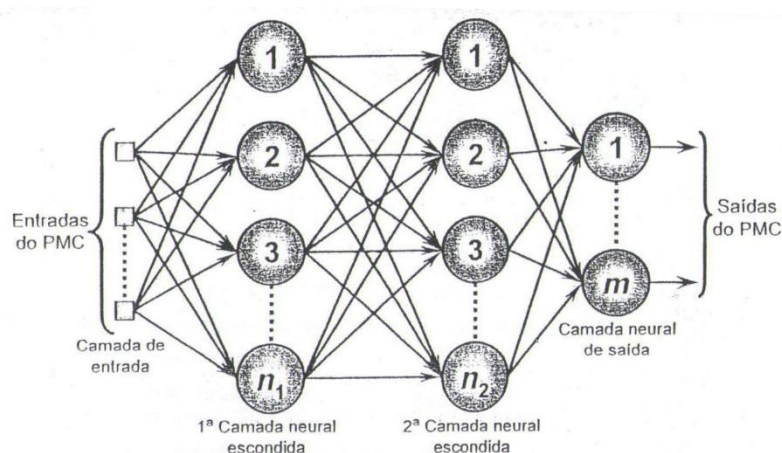


Figura 18 - Exemplo de rede perceptron multicamadas

Fonte: Silva, Spatti e Flauzino (2010).

2.2.7.3. Algoritmo Levenberg-Marquardt

O algoritmo clássico utilizado nas redes MLP é o “*backpropagation*”, que é um algoritmo que faz uso da função erro quadrático, minimizando-a através da redução do seu gradiente. Contudo, esse algoritmo possui uma taxa de conversão muito lenta, precisando, portanto, de uma grande capacidade de processamento. Dessa forma, técnicas de otimização foram criadas, buscando melhorar o seu desempenho através da redução do tempo de processamento e do esforço computacional necessário (SILVA; SPATTI; FLAUZINO, 2010).

Assim, o algoritmo de Levenberg-Marquardt (LM) surgiu como uma variação do *backpropagation*. Esse algoritmo é um método gradiente de segunda ordem com base no método dos mínimos quadrados para modelos não-lineares, sendo uma aproximação do método de Newton. Uma característica dele é o uso de uma taxa de aprendizado variável (BRAGA; CARVALHO; LUDERMIR, 2007; SILVA; SPATTI; FLAUZINO, 2010). A minimização do erro e ajuste dos pesos é realizado pela seguinte equação:

$$\Delta w_{ji}(t) = -[\nabla^2 E(w_{ij}(t)) + \mu I]^{-1} \nabla E(w_{ji}(t)) \quad (2.9)$$

Onde,

- $\nabla^2 E(w_{ij}(t))$: matriz Hessiana (matriz de derivadas de segunda ordem) da função que representa o erro E;
- $\nabla E(w_{ij}(t))$: matriz Jacobiana (matriz de derivadas de primeira ordem) da função que representa o erro E.

Contudo, Braga, Carvalho e Ludermir (2007) afirmam que esse algoritmo necessita de elevada complexidade computacional, sendo pouco eficiente para redes muito extensas. Apesar disso, Silva, Spatti e Flauzino (2010) dizem que o algoritmo LM consegue ser de 10 a 100 vezes mais rápido que o “*backpropagation*”.

2.2.7.4. Algoritmo de Levenberg-Marquardt com Regularização Bayesiana

Um dos problemas existentes durante o treinamento de uma rede neural, principalmente de RNAs com séries curtas de dados, é o “*overfitting*”, que é o sobreajuste dos pesos da rede devido a um treinamento excessivo.

Quando esse problema ocorre, a rede chega a valores de erros muito pequenos durante o treinamento, contudo ao tentar simular dados que não fazem parte do grupo utilizado a performance da rede é ruim, ou seja, é como se a rede memorizasse as nuances do conjunto de treinamento, perdendo a capacidade de generalizar o restante dos dados. Um exemplo de “*overfitting*” pode ser visto na Figura 19.

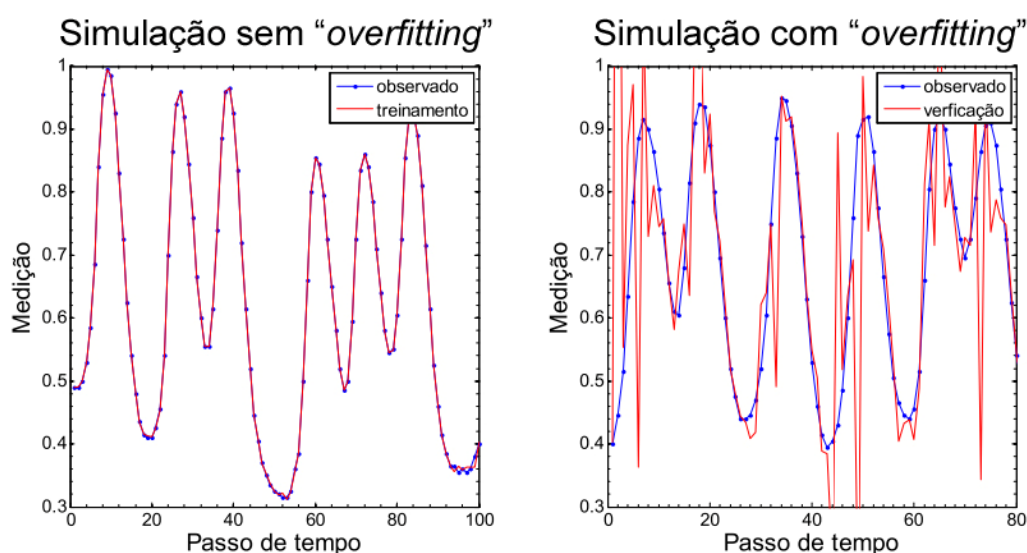


Figura 19 - Exemplo de simulação sem “*overfitting*” e com “*overfitting*”

Fonte: Adaptado de Honorato (2017).

Para solucionar esse problema algumas técnicas podem ser aplicadas durante o treinamento. A primeira delas é a Parada Antecipada do Treinamento, por meio da qual divide-se o conjunto de dados da série histórica em três grupos: conjunto de treinamento, conjunto de validação e conjunto de simulação.

Durante a etapa de ajuste dos pesos, é utilizado o grupo de treinamento. À medida que os pesos são atualizados, é verificado a capacidade da rede em simular os dados do conjunto de validação. A tendência é que tanto os erros do conjunto de

treinamento como os erros do conjunto de validação diminuam com o ajuste dos pesos.

Caso o erro do conjunto de treinamento continue a diminuir e o erro do conjunto de validação comece a aumentar, o treinamento é interrompido, pois isso indica que a rede está começando a memorizar os dados, perdendo a capacidade de generalizar as respostas (MATHWORKS, [20--]).

Outra técnica que pode ser utilizada é a Regularização Bayesiana. A abordagem Bayesiana foi proposta por MacKay em 1992 (CINTRA, 2003) e ela considera os pesos e os limiares da rede como variáveis aleatórias com distribuições específicas que são atualizados ao longo do treinamento por meio da regra de Bayes (CINTRA, 2003; FORESEE; HAGAN, 1997).

Foresee e Hagan (1997) afirmam que esse algoritmo apresenta capacidade excelente de generalização. Outra vantagem em utilizar essa técnica está na ausência da necessidade de um grupo de validação. Sendo, portanto, a melhor opção para redes com séries curtas de dados (CINTRA, 2003; MATHWORKS, [20--]), já que o grupo utilizado no treinamento da rede é menor.

O algoritmo tem como objetivo minimizar uma função de custo e conseqüentemente aproximar as respostas geradas pela rede dos valores desejados. A função de custo leva em consideração o somatório do erro quadrático E_D e o somatório dos pesos da rede neural E_w (CINTRA, 2003; NICOLAU, 2018).

A regularização bayesiana introduz um termo que penaliza pesos sinápticos muito grandes que podem aparecer durante o treinamento, evitando que ocorra “*overfitting*” (KAYRI, 2016). A equação que descreve essa função é a ((2.10).

$$E(w) = \beta E_D + \alpha E_w = \beta \frac{1}{N} \sum_{i=1}^N (d_i - y_i)^2 + \alpha \frac{1}{n} \sum_{j=1}^n w_j^2 \quad (2.10)$$

Onde,

- $E(w)$: função objetivo/custo a ser minimizada;
- β e α : são hiperparâmetros que servem para ponderar o somatório dos erros e dos pesos, respectivamente;

- E_D : somatório do erro quadrático;
- E_w : somatório dos pesos da rede neural;
- N : número de pares observados/estimados;
- n : número de pesos da rede;
- d_i : dados observados;
- y_i : dados estimados;
- w_j : pesos da rede.

Por meio da equação (2.10) é possível perceber que para valores de $\alpha \ll \beta$, o algoritmo irá levar a erros menores, caso $\alpha \gg \beta$ o algoritmo irá priorizar a redução dos pesos da rede em detrimento dos erros (FORESEE; HAGAN, 1997).

Para atualizar os pesos é utilizada a regra de Bayes. Todo o procedimento foi apresentado por Foresee e Hagan (1997) e está descrito a seguir:

$$P(w|D, \alpha, \beta, M) = \frac{P(D|w, \beta, M)P(w|\alpha, M)}{P(D|\alpha, \beta, M)} \quad (2.11)$$

Onde,

- D : é o conjunto de dados;
- M : é o modelo de rede neural utilizado;
- w : vetor dos pesos;
- $P(w|\alpha, M) = \frac{1}{Z_D(\beta)} \exp(-\beta E_D)$, e corresponde a densidade inicial, que representa o conhecimento dos pesos antes de seu ajuste;
- $P(D|w, \beta, M) = \frac{1}{Z_w(\alpha)} \exp(-\alpha E_w)$, que é a função que define a probabilidade dos dados ocorrerem;
- $P(D|\alpha, \beta, M)$: é o fator de normalização.

Sabendo-se que $Z_D(\beta) = \left(\frac{\pi}{\beta}\right)^{n/2}$ e $Z_w(\alpha) = \left(\frac{\pi}{\alpha}\right)^{\frac{n}{2}}$ podemos desenvolver as expressões apresentadas acima, obtendo:

$$P(w|D, \alpha, \beta, M) = \frac{1}{Z_F(\alpha, \beta)} \exp(-F(w)) \quad (2.12)$$

Assim, os pesos ótimos serão os pesos que maximizem a densidade posterior $P(w|D, \alpha, \beta, M)$, o que conseqüentemente resulta em uma minimização da função objetivo apresentada anteriormente.

Em seguida, deve-se determinar os valores ótimos para α e β que permitam que a rede não sofra com o sobreajuste dos pesos e ao mesmo tempo permita uma boa capacidade de generalização do problema desejado. Para que esses valores sejam obtidos, pode-se aplicar novamente a regra de Bayes para determinar os valores ótimos para α e β chegando na seguinte equação:

$$P(\alpha, \beta|D|M) = \frac{P(D|\alpha, \beta, M)P(\alpha, \beta|M)}{P(D|M)} \quad (2.13)$$

Para desenvolver a expressão apresentada acima, devemos supor uma densidade inicial $P(\alpha, \beta|M)$ uniforme para α e β . Em seguida é maximizada a função probabilidade $P(D|\alpha, \beta, M)$, que é justamente o fator de normalização da equação (2.11) apresentado anteriormente.

Contanto que todas as probabilidades tenham uma distribuição gaussiana, a sequência abaixo, que determina $P(D|\alpha, \beta, M)$, é válida.

$$P(w|D, \alpha, \beta, M) = \frac{P(D|w, \beta, M)P(w|\alpha, M)}{P(D|\alpha, \beta, M)} \Rightarrow P(D|\alpha, \beta, M) = \frac{P(D|w, \beta, M)P(w|\alpha, M)}{P(w|D, \alpha, \beta, M)}$$

Substituindo os termos conhecidos chega-se a:

$$P(D|\alpha, \beta, M) = \frac{Z_F(\alpha, \beta)}{Z_D(\beta)Z_W(\alpha)} \quad (2.14)$$

Para determinar o valor $Z_F(\alpha, \beta)$, a função objetivo do algoritmo $E(w)$ deve ter forma quadrática em torno do ponto mínimo w^{PM} . Sendo essa condição satisfeita,

determina-se a série de Taylor de $E(w)$ ao redor de w^{PM} , chegando-se a seguinte equação:

$$Z_F \cong (2\pi)^{\frac{N}{2}} (\det ((H^{PM})^{-1}))^{\frac{1}{2}} \exp(-E(w^{PM})) \quad (2.15)$$

Onde,

- w^{PM} : é o ponto mínimo da densidade posterior;
- H : é a matriz Hessiana da função objetivo, dada por $H = \beta \nabla^{2E_D} + \alpha \nabla^2 E_W$.

Substituindo Z_F na equação (2.14), chega-se as expressões para os valores ótimos de α e β no ponto mínimo, que podem ser vistas na equação (2.16).

$$\alpha^{PM} = \frac{\gamma}{2E_W(w^{PM})} \text{ e } \beta^{PM} = \frac{n - \gamma}{2E_D(w^{PM})} \quad (2.16)$$

Onde,

- $\gamma = N - 2\alpha^{PM} \text{tr}(H^{-1})$, e corresponde ao número efetivo de parâmetros da RNA que foram usados na redução de $E(w)$;
- N : número total de parâmetros da rede neural.

Para simplificar o cálculo de γ , Foresee e Hagan (1997) propuseram utilizar a aproximação de Gauss-Newton no lugar de H .

Por fim, todo o processo descrito acima, fazendo uso aproximação de Gauss-Newton, pode ser resumido nos seguintes passos (FORESEE; HAGAN, 1997):

1. Os parâmetros são iniciados com os valores iniciais $\beta=1$ e $\alpha=0$;
2. São atribuídos valores aleatórios aos pesos;
3. Utiliza-se um passo do algoritmo Levenberg-Marquardt para minimizar a função objetivo;
4. Calcula-se o número efetivo de parâmetros pela equação $\gamma = N - 2\alpha^{PM} \text{tr}(H^{-1})$, utilizando a aproximação de Gauss-Newton, como dito anteriormente;
5. Calcula-se os valores de α e β para o ponto w ;

6. Repete-se os passos de 3 a 4 até que ocorra a convergência do algoritmo.

Vale salientar que para um melhor desempenho do algoritmo, devido a utilização da aproximação Gauss-Newton na regularização bayesiana, as entradas e a matriz com os valores observados, que são fornecidos durante o treinamento, devem ser normalizados no intervalo $[-1, 1]$.

3) METODOLOGIA

3.1. Descrição geral

O presente trabalho buscou verificar o desempenho de cinco grupos de RNAs, treinadas com séries curtas de dados, em prever eventos críticos. Para isso, foram utilizadas apenas pequenas parcelas da série histórica de vazões diárias do reservatório de Chavantes para a sua calibração. Esses dados foram fornecidos à rede neural criada durante o treinamento e, em seguida, fez-se a extrapolação das previsões para dados que não tinham sido apresentados ainda a RNA treinada. Ou seja, a rede foi configurada com uma pequena quantidade de dados e após isso foi testada sua capacidade de generalização para todo o restante da série.

Como dito anteriormente, foram estabelecidos cinco grupos de RNAs com os mesmos dados de vazões, a diferença entre elas foi justamente na porcentagem da série histórica que foi usada no treinamento. Cinco porcentagens de dados da série histórica foram usadas, foram elas: 10, 20, 30, 40 e 50%. Ou seja, para a primeira configuração citada, retirou-se 90% das vazões presentes no banco de dados, nesses dados foram utilizadas tanto vazões consideradas normais como vazões consideradas extremas. Em seguida, usou-se esses 10% da série histórica para treinar a rede. Os 90% dos dados restantes foram utilizadas nas simulações. Logo, é possível perceber que para essa RNA foi usada uma menor quantidade de exemplos possíveis no treinamento. Essa rede foi chamada de RNA-10. O mesmo foi feito para as demais porcentagens citadas anteriormente, criando as redes RNA-20, RNA-30, RNA-40 e RNA-50. E em seguida foi realizado o treinamento das redes para três horizontes de previsões, 1 dia, 3 dias e 7 dias. Dessa forma, chegou-se a um total de 15 RNAs treinadas.

Por fim, a performance de cada uma delas foi determinada com base em uma análise visual e por meio de três parâmetros estatísticos de erro, a raiz do erro quadrático médio, o erro médio absoluto relativo e o coeficiente de eficiência, possibilitando a comparação das RNAS entre si. Tais análises e os parâmetros utilizados serão detalhados mais à frente.

3.2. Caracterização da área

Como dito anteriormente, o reservatório utilizado como objeto de estudo foi o reservatório de Chavantes, que ocupa uma área de 428,34 km². Esse reservatório pertence a Usina Hidrelétrica de Chavantes, que está localizada na cidade de mesmo nome e encontra-se entre as usinas de Jurumirim e Salto Grande.

A usina utiliza o fluxo de água do rio Paranapanema para geração de energia elétrica tendo uma potência instalada de 414 MW. Dessa forma, é uma das mais importantes do estado de São Paulo e do Brasil, uma vez que é responsável pelo abastecimento elétrico do interior de São Paulo, como também da capital, além de suprir demandas do estado do Paraná.

Sua construção foi iniciada em 1959, iniciando parte de suas operações em 1970. O reservatório tem um comprimento de 500 metros e 89 metros de altura, possuindo uma capacidade de armazenamento de 9,4 bilhões de m³, sendo responsável também pela regulação da vazão da região do médio Paranapanema. A operação da usina de Chavantes é coordenada pelo Operador Nacional do Sistema Elétrico (ONS) (CTG BRASIL, [20--]b, ITAPONEWS, 2017).

Um mapa representativo da unidade de gestão de recursos hídricos Paranapanema pode ser visto na Figura 20. Para a sua elaboração foram utilizados arquivos “*shapefile*” disponíveis no site da Agência Nacional de Águas (ANA, 2014).

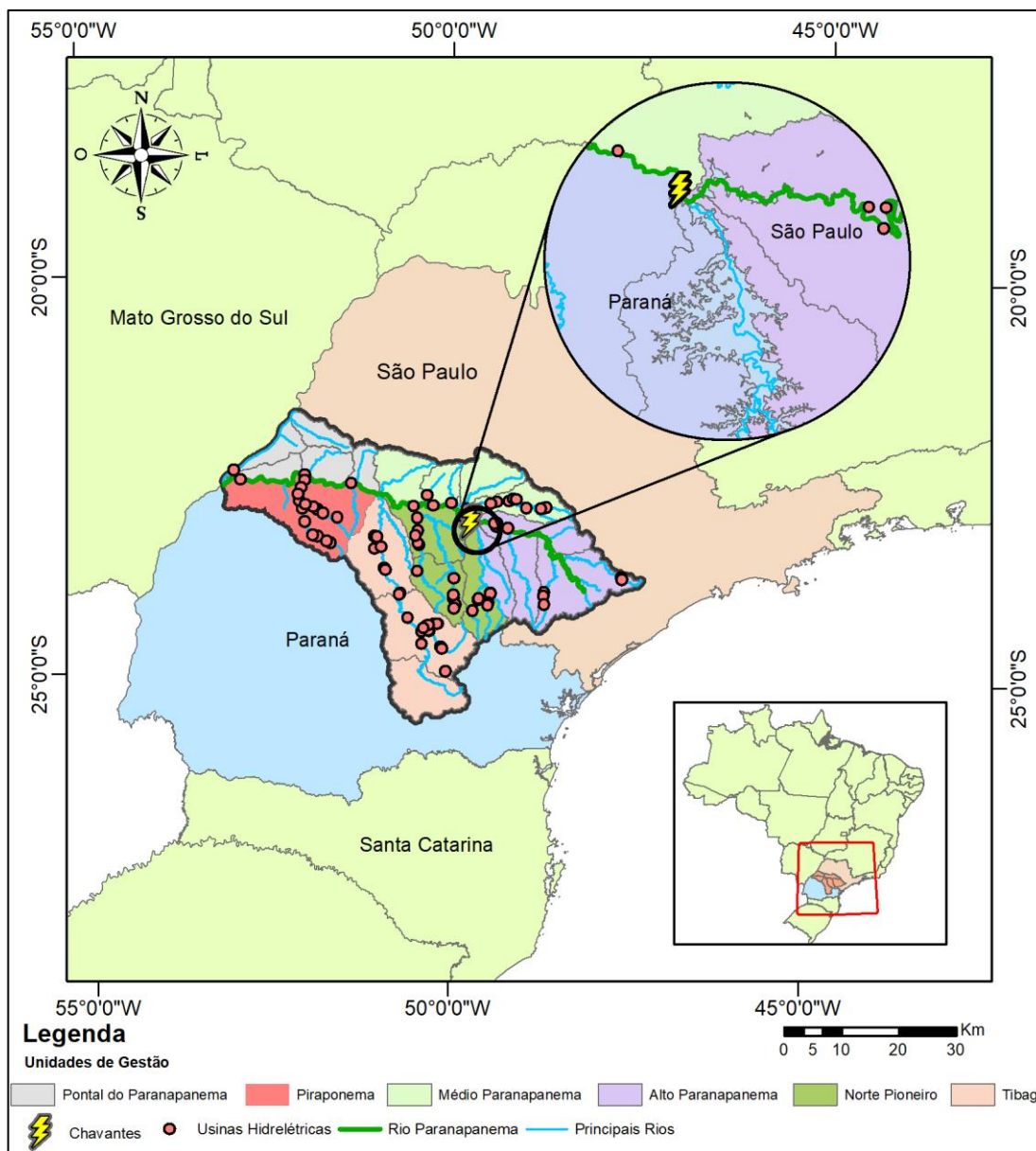


Figura 20 - Unidade de gestão de recursos hídricos Paranapanema

Fonte: Própria (2019).

O rio Paranapanema faz divisa entre o estado do Paraná e São Paulo, sendo um dos rios mais importantes desse estado. Ele nasce na serra Augudos Grandes, no município de Capão Bonito, localizado no estado de São Paulo e desagua no rio Paraná, totalizando uma extensão de 930 quilômetros.

Ao longo do seu percurso a bacia desse rio drena uma área de 247 municípios, tanto do estado do Paraná como de São Paulo. Segundo a ANA (2016), essa área abrange 2,3% da população brasileira, o que mostra a importância social como também econômica desse rio. Devido a sua grande extensão, essa bacia é dividida

em 6 unidades de gestão, Alto Paranapanema, Médio Paranapanema, Pontal do Paranapanema, Norte Pioneiro, Tibagi e Piraponema (ANA, 2016). Além disso, ao longo do seu traçado existe um total de 11 usinas hidrelétricas, as quais se sucedem em cascata.

Essa grande quantidade de pontos de geração de energia é devido as suas características geomorfológicas, as quais são propícias a instalação de reservatórios para a geração de energia elétrica. Algumas dessas características são a declividade e a grande quantidade de quedas d'água no seu percurso, principalmente na região do Médio Paranapanema. Das 11 usinas existentes, a 5 maiores e suas respectivas capacidades instaladas são: Capivara com 619 MW, Taquaruçu com 525 MW, Chavantes com 414 MW, Rosana com 354 MW e Jurumirim com 100,9 MW. A bacia de Paranapanema é responsável pela geração de 5,95% de energia do Subsistema Sudeste/Centro-Oeste, e a usina de Chavantes, que pertence a essa bacia, é responsável por 1,70% do subsistema (CTG BRASIL, [20--]a; MAGNONI, 2009, ONS, 2019b).

3.3. Base de dados

Os dados que foram utilizados para o treinamento e aplicação da rede neural foram dados de vazões diárias afluentes do rio Paranapanema ao reservatório de Chavantes. A escolha desse reservatório deu-se por meio da análise de um conjunto de dados de vários reservatórios sob a coordenação do ONS, disponíveis no site desse órgão (ONS, 2019a). A série histórica escolhida tem um total de 28.124 dias, iniciando no dia 01/01/1931 até 31/12/2007, ou seja, um total de 77 anos de medição. O gráfico da Figura 21 apresenta a variação da vazão diária ao longo desses 77 anos.

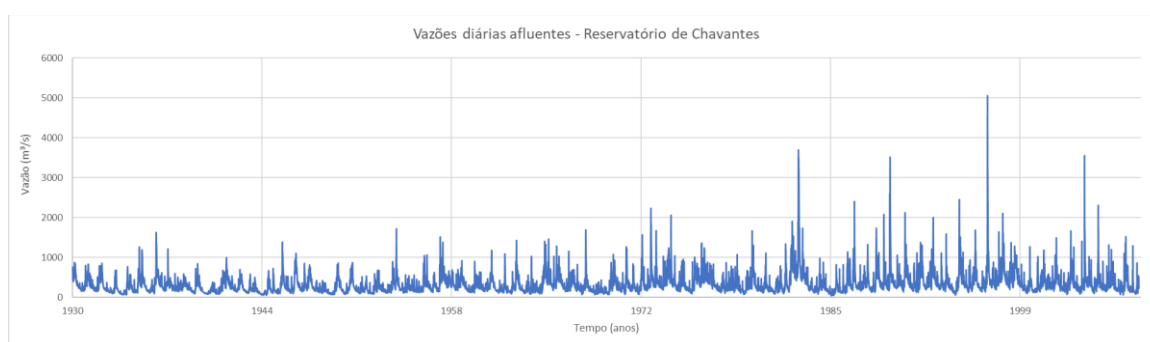


Figura 21 - Série histórica de vazões utilizada

Fonte: Própria (2019).

A escolha desse reservatório foi devido à natureza dos dados. Como é possível perceber por meio do gráfico da Figura 21, de 1931 até 1982 as vazões estavam compreendidas na faixa de 0 a 2200 m³/s. Em 1983 houve o primeiro pico, que ultrapassou 3000 m³/s. O que se repetiu em 1990, 1997 e 2004. Ou seja, na maior parte da série histórica os dados reportados estavam todos compreendidos dentro de uma mesma faixa e apresentou alguns picos de forma pontual que destoaram bastante das vazões usuais. Dessa forma, foi possível verificar o desempenho das RNAs tanto para vazões consideradas normais como vazões consideradas atípicas.

3.3.1. Vazões críticas

Pode-se entender como vazões críticas aquelas que possuem poucas ocorrências ao longo da série histórica, mas que apresentam valores elevados, sendo de grande importância dentro do conjunto de dados.

Para a caracterização de uma vazão como sendo crítica, foi calculada a média de todas as vazões da série histórica e o desvio padrão delas. As vazões superiores à média mais duas vezes o desvio padrão foram consideradas vazões críticas.

3.4. Criação dos grupos

Para realizar o treinamento da rede foram criados grupos que deveriam ser removidos e grupos de treinamento com os dados restantes.

A criação dos grupos obedeceu ao seguinte procedimento: calculou-se a porcentagem dos dados da série que deveriam ser utilizados no treinamento, considerou-se o dia 01/01/1931 como o primeiro elemento desse grupo, englobando os demais dados que compreendiam o intervalo definido do grupo. Em seguida, removeram-se os dados posteriores, ou seja, os dados restantes. Os dados removidos formaram os grupos que foram utilizados como referência das simulações.

A Figura 22 apresenta um exemplo, de forma visual, da localização dos grupos de treinamento e simulação ao longo da série histórica para a RNA-10.

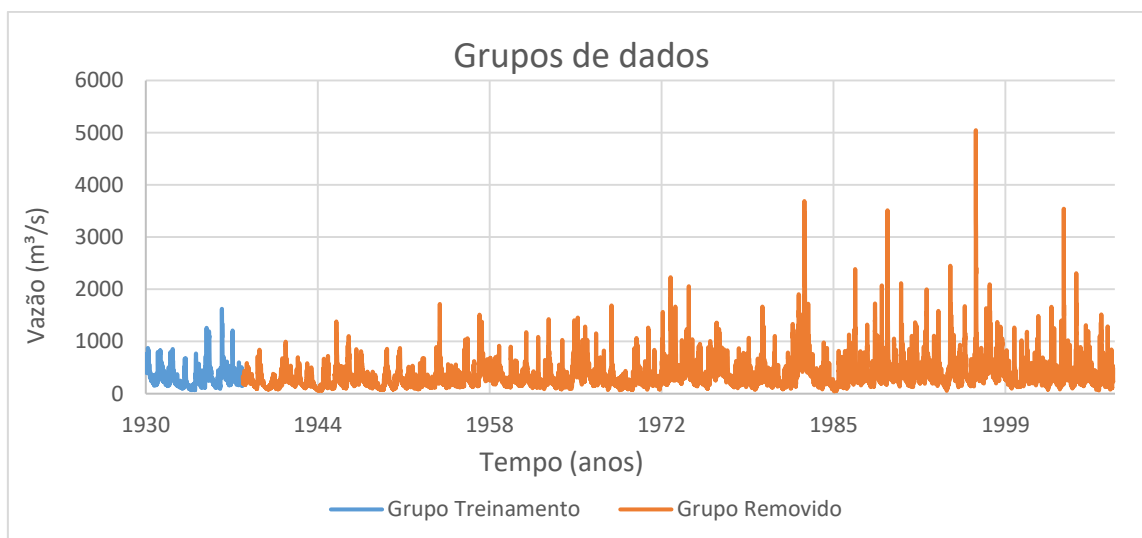


Figura 22 - Grupos de dados para RNA-10
Fonte: Própria (2019).

A mesma abordagem foi feita para as demais RNAs: RNA-20, RNA-30, RNA-40 e RNA-50.

3.5. Modelo de Previsão

3.5.1. Consideração inicial

Para a construção do modelo de previsão utilizou-se uma RNA criada por meio da toolbox de redes neurais do software MATLAB, a qual foi alimentada com os dados do reservatório de Chavantes apresentados acima.

3.5.2. Rede adotada

O modelo de rede neural escolhido para ser usado no estudo foi uma rede do tipo Perceptron de Múltiplas Camadas. Portanto, além da camada de entrada e de saída essa rede possui uma camada intermediária. A escolha de uma única camada intermediária deu-se devido a busca pela adoção de uma rede mais simples possível, já que uma rede MLP com uma camada intermediária já é capaz de aproximar qualquer função matemática, seja ela linear ou não. Além disso Braga, Carvalho e Ludermir (2007) afirmam que o uso de muitas camadas intermediárias não é

recomendado, pois diminui a precisão da propagação do erro durante o treinamento. Outro fator levado em consideração foi a adoção de apenas uma camada intermediária por diversos estudos na área de recurso hídricos, os quais obtiveram sucesso na resolução de diversos problemas, como pode ser visto nos estudos de Adamowski e Sun (2010), Gomes, Montenegro e Valença (2010), Honorato, Silva e Santos (2018) e Sarmiento (1996).

3.5.3. Arquitetura

Pelo fato de ser uma rede MLP, a arquitetura utilizada foi a “*feedforward*” de múltiplas camadas com treinamento supervisionado utilizando o algoritmo de Levenberg-Marquardt com Regularização Bayesiana. O número de neurônios na camada de entrada e na camada intermediária foi determinado por tentativa e erro, adotando a configuração que deu melhores resultados. Para isso variou a quantidade de neurônios de 1 até 10 de forma interativa nessas duas camadas. Já na camada de saída, como o objetivo da rede era retornar uma única vazão futura, a última camada da rede utilizou um único neurônio.

3.5.4. Função de ativação adotada

Como dito anteriormente, o algoritmo de ajuste dos pesos utiliza o método do gradiente descendente durante o seu treinamento para atualizar os pesos. Portanto, as funções de ativação precisam ser contínuas, diferenciáveis e de preferência não-decrescentes. As funções que mais se adequam a essas características são as do tipo sigmoide. Portanto, essa foi uma das funções testadas. Além dela, outra função utilizada nos testes foi a função linear. Devido ao número de camadas escolhido foi necessária a escolha de duas funções de ativação, dessa forma optou-se por testar duas configurações, foram elas: tangente hiperbólica aplicada duas vezes e tangente hiperbólica seguida da função linear. A opção que deu melhores resultados foi a escolhida.

3.5.5. Aprendizado e treinamento da rede

O algoritmo de aprendizado utilizado foi o de Levenberg-Marquardt com Regularização Bayesiana. Para o treinamento das redes, o conjunto de dados fornecido foi a série histórica menos os grupos removidos, que variaram dependendo da rede que estava sendo treinada.

Como o objetivo foi utilizar valores de vazões diárias para prever a vazões futuras, e o treinamento de redes MLP é supervisionado, foi preciso fornecer a rede o valor das vazões que a rede deveria utilizar na previsão e o valor de vazão que ela deveria obter.

Vale salientar que a quantidade de dias utilizados para prever as vazões futuras foi igual ao número de neurônios da camada de entrada. Essas informações foram fornecidas a rede na forma de vetores. Um exemplo esquemático, considerando uma rede com a camada de entrada contendo 5 neurônios pode ser visto na Figura 23. Por meio dessa arquitetura exemplo apresentada abaixo a rede necessitou de 5 dias para prever 1 semana a frente.

Dia 1	Dia 2	Dia 3	Dia 4	Dia 5	Dia 6	Dia 7	Dia 8	Dia 9	Dia 10	Dia 11	Dia 12	Dia 13
736	763	749	710	652	599	568	512	460	454	491	534	492

Dados usado para prever	Valor desejado
736	534
763	
749	
710	
652	

Figura 23 - Exemplo de vetor fornecido a rede

Fonte: Própria (2019).

Após o fornecimento dos dados necessários para a rede e finalizado o treinamento, foi obtida uma primeira resposta do modelo. Contudo, a forma como o treinamento é iniciado pode influenciar na resposta obtida, pois os pesos iniciais utilizados são atribuídos de forma aleatória. Dependendo dos pesos atribuídos a rede pode demorar mais a chegar na solução ótima como também pode acabar encontrado uma solução ótima local em vez da global. Assim, para verificar se a resposta obtida foi a melhor possível, repetiu-se o treinamento pelo menos 10 vezes para cada arquitetura, observando se os dados destoavam de forma significativa entre as tentativas ou não.

3.5.6. Normalização dos dados

Buscando facilitar o aprendizado da rede e melhorar o desempenho do modelo, antes de realizar o treinamento os dados fornecidos a ela foram normalizados. A normalização dos dados além de aumentar a eficiência computacional tem o objetivo de simplificar o modelo, evitar que números muito grandes presentes nos dados sobressaiam números menores e previne a saturação dos pesos da camada intermediária durante o processo de treinamento (BASHEER; HAJMEER, 2000; OLIVEIRA, 2014; SILVA; SPATTI; FLAUZINO, 2010).

Basheer e Hajmeer (2000) afirmam que não existe uma forma padrão de normalizar os dados, contudo uma das estratégias utilizadas é converter os dados para o mesmo intervalo que a imagem da função de ativação está compreendida.

Como foi adotado o algoritmo de Levenberg-Marquardt com Regularização Bayesiana que faz uso da aproximação Gauss-Newton, os dados da série temporal foram normalizados no intervalo $[-1, 1]$.

3.5.7. Critérios do treinamento

Durante o treinamento da rede neural alguns critérios foram levados em consideração para interromper o processo de aprendizagem, esses limites foram definidos com o intuito de evitar que a rede continue buscando uma solução teoricamente melhor, porém de forma ineficiente. Alguns parâmetros que foram definidos antes do treinamento com esse objetivo foram: o alcance da performance desejada, que fica a critério do operador da rede, o número máximo de interações (épocas) e o valor mínimo de gradiente a ser atingido. Vale salientar a importância de utilizar tais parâmetros, pois por meio deles evita-se que a rede tenha um tempo de processamento exagerado sem ganho de performance.

Dessa forma, para as RNAs treinadas o limite da performance desejado foi alterado do valor padrão que é 0 para $1e-5$. Já o número máximo de interações, que por padrão é de 1000 épocas de ajuste, foi mantido, uma vez que o erro na fase de treinamento reduziu com o aumento das interações, ficando praticamente constante para quantidades de interações superiores a 1000. Por fim, o valor mínimo do

gradiente não foi alterado, deixou-se o valor padrão de $1e-7$, pois tal valor mostrou-se suficiente nos treinamentos. Além disso, Honorato (2017) afirma que valores muito baixos desse parâmetro não proporcionam um ganho significativo de performance.

3.5.8. Avaliação das aproximações

É esperado que os resultados obtidos pela rede não sejam exatamente iguais aos desejados, pois apesar dos modelos de predições algumas vezes apresentarem uma performance excelente, eles não são perfeitos para simular os efeitos da natureza. Dessa forma, para determinar o quão perto dos valores reais o modelo foi capaz de chegar, utiliza-se equações estatísticas para a medição da taxa do erro cometido durante as aproximações (OLIVEIRA, 2014). O uso de tais parâmetros de qualidade é de grande valia pois eles são os fatores mais importantes na determinação de qual arquitetura utilizar, quais funções de ativação adotar, se a quantidade de dados do conjunto de treinamento foi suficiente e assim por diante. Ou seja, eles são responsáveis por direcionar os ajustes necessários para alcançar os objetivos desejados.

No presente estudo, como já citado anteriormente, para avaliar os resultados obtidos foram utilizados 3 parâmetros estatísticos aliados com a análise visual de gráficos comparativos. Para a análise visual foram elaborados gráficos sobrepondo os valores obtidos versus os valores desejados, observando o comportamento das duas curvas. Já os parâmetros estatísticos utilizados foram: a raiz do erro quadrático médio (RMSE), o erro médio absoluto relativo (MARE) e o coeficiente de eficiência (CE). As equações para cada um desses parâmetros podem ser vistas a seguir:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3.1)$$

$$MARE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i} \quad (3.2)$$

$$CE = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)}{\sum_{i=1}^n (y_i - \bar{y}_i)} \quad (3.3)$$

Onde,

- y_i : valores observados
- \bar{y}_i : média dos valores observados
- \hat{y}_i : valores das aproximações
- n : número de pares da amostra

O parâmetro RMSE é o mais utilizado nos estudos para determinação das melhores arquiteturas. Ele fornece uma indicação do desvio médio entre o observado e o previsto, sendo uma medida de desempenho que depende da escala, não sendo útil para modelos que utilizam mais de uma série temporal com escalas temporais diferentes. Além disso, é um parâmetro que penaliza grandes desvios, uma vez que a diferença entre valores observados e previstos é elevada ao quadrado (NEUSSER, 2015; OLIVEIRA et al., 2013). Quanto menor for o seu valor, melhor a RNA.

Da mesma forma que o RMSE o MARE indica da precisão absoluta do modelo, e analogamente o quanto menor, melhor. É um parâmetro adimensional, que indica se há tendências nas estimativas, que ocorrem caso os valores sejam crescentes (FERREIRA, 2017).

E por fim, o CE vai de $-\infty$, para o pior caso, até 1, que seria a melhor situação. Ele dá uma ideia de quão bom o modelo consegue prever valores que estão fora da média. Quanto mais perto de 1 o valor desse parâmetro, melhor é a concordância entre o modelo e os dados reais (DAWSON et al., 2006; HONORATO; SILVA; SANTOS, 2018).

4) RESULTADOS

4.1. Determinação das vazões críticas

Após os cálculos dos parâmetros estatísticos, foi possível chegar ao valor de vazão acima do qual as vazões são consideradas críticas. A Tabela 1 mostra os parâmetros calculados.

Tabela 1 - Parâmetros estatísticos

Parâmetros	Valores	Unidades
Média	338,7	m ³ /s
Moda	202	m ³ /s
Mediana	267	m ³ /s
Primeiro Quartil	185	m ³ /s
Terceiro Quartil	412	m ³ /s
Mínimo	44	m ³ /s
Máximo	5048	m ³ /s
Desvio Padrão	253,7	m ³ /s

Fonte: Própria (2019).

Como a vazão crítica foi caracterizada como sendo a média mais duas vezes o desvio padrão, o valor de vazão crítica obtido foi de 850 m³/s.

A Tabela 2 mostra a porcentagem das vazões que são superiores a uma determinada referência especificada.

Tabela 2 - Quantidade de vazões superiores a uma determinada referência

Referência (m ³ /s)	Quantidade de vazões diárias superiores a referência (m ³ /s)	Porcentagem do total (%)
100	27308	97,10
200	19587	69,65
300	11867	42,20
400	7398	26,30
500	4657	16,56
600	2964	10,54
700	1985	7,06
800	1302	4,63
900	899	3,20
1000	646	2,30
2000	61	0,22
3000	21	0,07
4000	3	0,01
5000	1	0,004

Fonte: própria (2019).

Como pode ser visto na Tabela 2, a quantidade de vazões diárias superiores a 800 m³/s correspondem a 4,63% dos dados da série histórica. Para 850 m³/s a porcentagem é ainda menor, um total de 3,85%. Ou seja, tais vazões podem ser consideradas críticas, uma vez que ocorrência ao longo da série é muito pequena e seus valores são bem superiores à média histórica.

4.2. Previsões de vazão

Após o treinamento das redes neurais e feitas as simulações para os períodos normais e críticos da rede, foi possível elaborar gráficos comparativos e calcular os parâmetros estatísticos que serão apresentados a seguir.

4.2.1. Previsões para 1 dia a frente

4.2.1.1. Treinamento

Após a definição das redes neurais e obtidos os resultados do treinamento, as melhores arquiteturas foram determinada por meio do parâmetro RMSE como também por meio de análise gráfica.

Dentre todas as arquiteturas testadas para a rede RNA-10_{1D}, a configuração da rede que obteve melhor desempenho foi a 10-10-1, ou seja, 10 neurônios na camada de entrada, 10 neurônios na camada intermediária e 1 neurônio na camada de saída.

A melhor arquitetura para as demais redes e os parâmetros estatísticos obtidos durante o treinamento podem ser vistas de forma resumida na Tabela 3.

Tabela 3 - Melhores arquiteturas e parâmetros obtidos para 1 dia de previsão futura

Rede	Porcentagem usada no treinamento (%)	Arquitetura			Treinamento			Simulação			
		Dias a frente	nne	nnint	nns	RMSE (m ³ /s)	CE	MARE	RMSE (m ³ /s)	CE	MARE
RNA-10 _{1D}	10	1	10	10	1	18,6000	0,9912	0,0377	119,1800	0,7881	0,0772
RNA-20 _{1D}	20	1	10	10	1	18,9390	0,9893	0,0404	164,8400	0,6167	0,0862
RNA-30 _{1D}	30	1	10	10	1	21,0320	0,9857	0,0487	123,1600	0,7997	0,0796
RNA-40 _{1D}	40	1	10	10	1	26,4720	0,9783	0,0528	153,2100	0,7151	0,0791
RNA-50 _{1D}	50	1	10	9	1	26,3780	0,9800	0,0543	171,6300	0,6687	0,0800

*nne - número de neurônios de entrada; nnint - número de neurônios intermediários e nns - número de neurônios de saída.

Fonte: Própria (2019).

Nos gráficos comparativos da Figura 24 até a Figura 28 podem ser visualizadas as vazões observadas e as vazões estimadas durante o treinamento, da melhor arquitetura obtida para cada uma das RNAs, considerando um dia de previsão futuro.

Observando os “detalhes A” dos gráficos, é possível perceber que as estimativas da rede durante o treinamento para as vazões de até 400 m³/s foram bem próximos aos observados, com erros na ordem de 10 a 20 m³/s, o que corresponde a um erro percentual de 5 a 10%.

Para as vazões superiores a 400 m³/s até 850 m³/s, que podem ser vistos nos “detalhes C”, as simulações conseguiram acompanhar o comportamento de

crescimento e decrescimento das vazões, mas os erros foram mais significativos, na ordem de 10 a 100 m³/s, com erros percentuais variando de 3 a 17%, para mais e para menos

Já para os períodos críticos, ou seja, para os períodos que ocorreram picos de vazão, as estimativas obtidas foram boas, ou seja, os dados gerados foram próximos aos dados reais. Essa comparação pode ser vista destacada nos “detalhes B”.

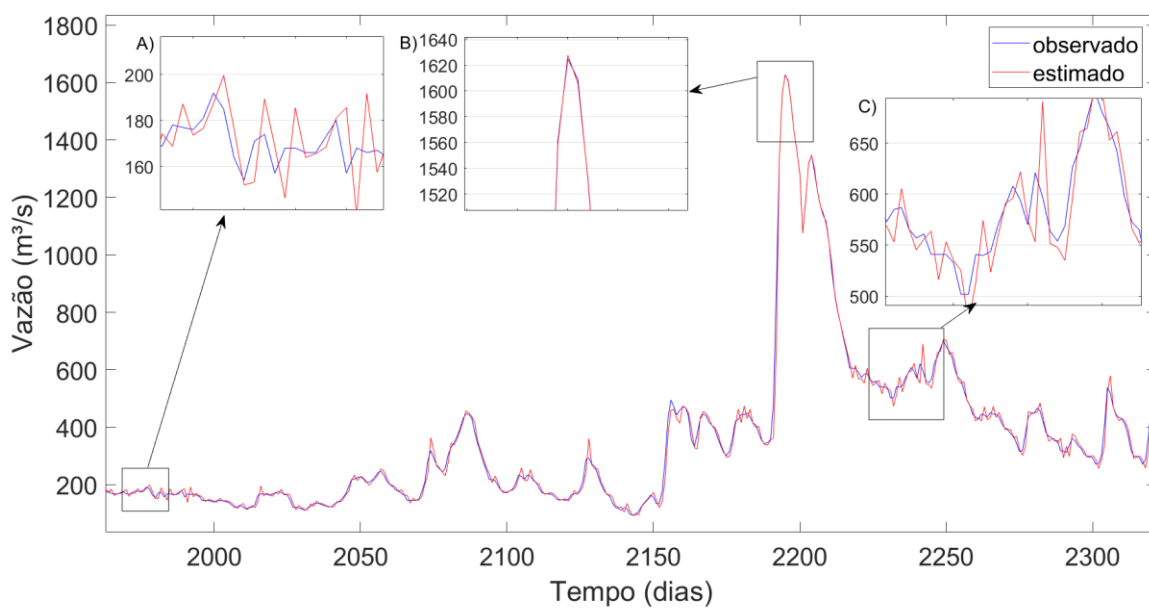


Figura 24 - Treinamento RNA-10_{1D}

Fonte: Própria (2019).

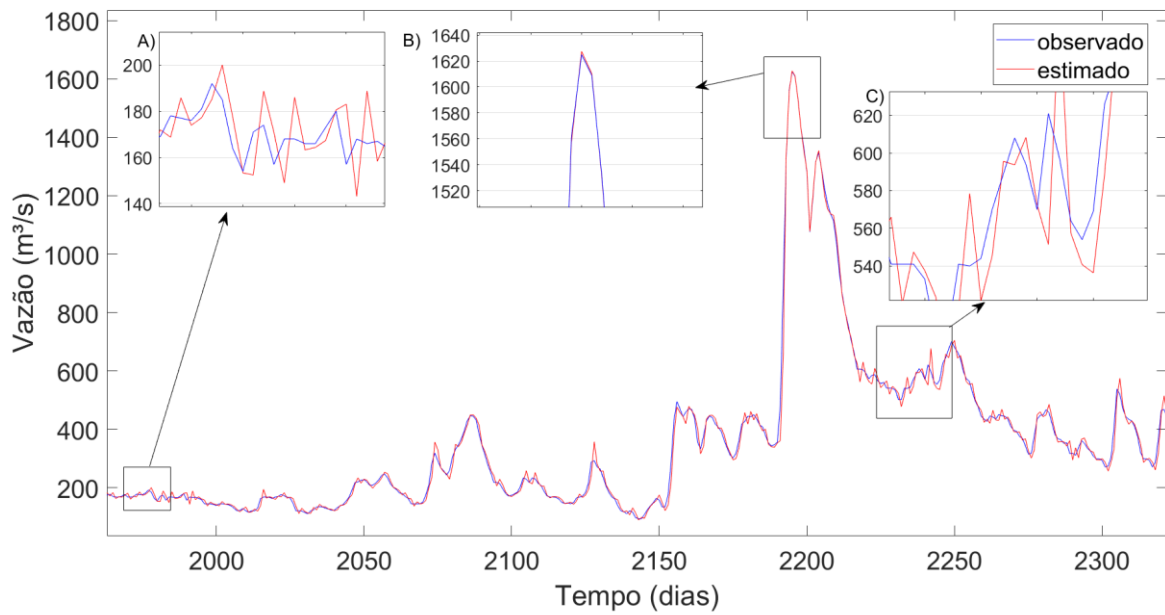


Figura 25 - Treinamento RNA-20_{1D}

Fonte: Própria (2019).

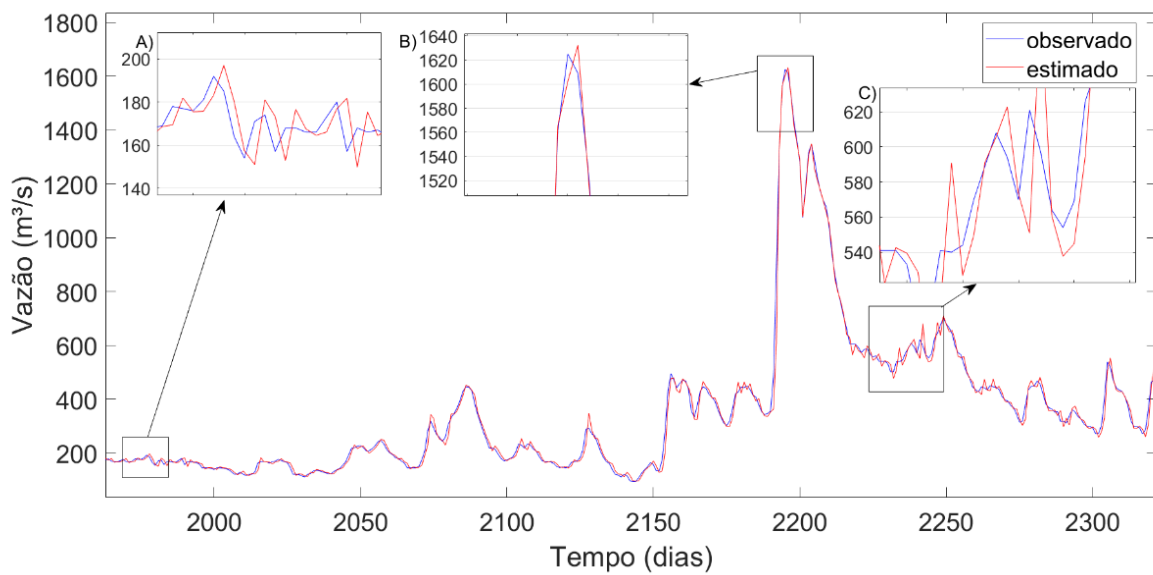


Figura 26 - Treinamento RNA-30_{1D}

Fonte: Própria (2019).

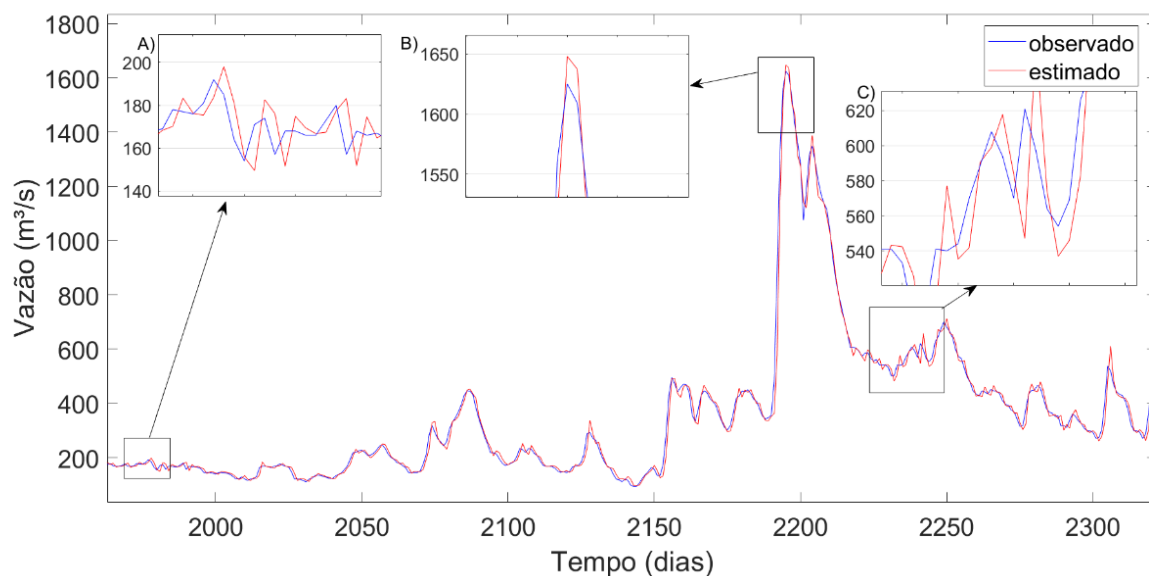


Figura 27 - Treinamento RNA-40_{1D}

Fonte: Própria (2019).

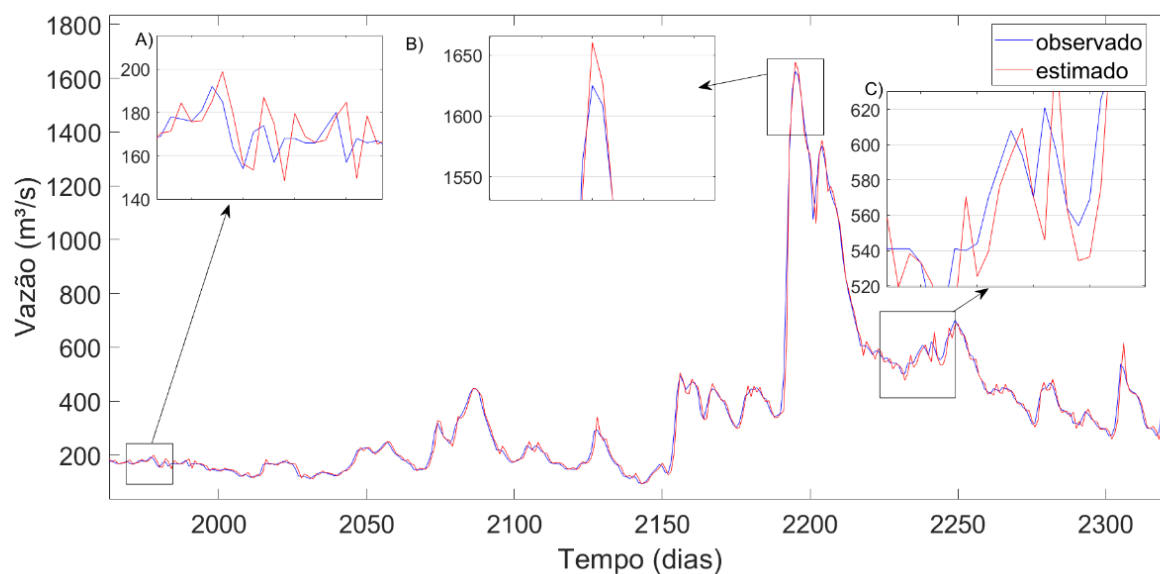


Figura 28 - Treinamento RNA-50_{1D}

Fonte: Própria (2019).

Além disso, foi possível perceber que os melhores resultados foram para as redes treinadas com menor porcentagem (RNA-10_{1D}, RNA-20_{1D} e RNA-30_{1D}) com erros na ordem de 10 m³/s e com erros na ordem de 20 a 40 m³/s para as de maior porcentagem (RNA-40_{1D} e RNA-50_{1D}), o que equivale a um erro percentual de 0,6% e 1,2 a 2,5% respectivamente.

Aliando a análise dos parâmetros RMSE, MARE e CE, com análise gráfica feita acima, é possível confirmar o melhor desempenho das redes treinadas com menor porcentagem, visto que elas apresentaram menores valores de RMSE e MARE e maiores valores de CE durante o treinamento e uma boa proximidade entre as curvas dos gráficos.

Em relação as arquiteturas, por meio da análise da Tabela 3, é possível perceber que as melhores configurações foram justamente as que tiveram uma maior quantidade de neurônios nas camadas, com exceção da RNA-50_{1D}. Isso é justificado pelo o aumento da capacidade de processamento, à medida que aumenta o número de neurônios, que são justamente as unidades de processamento da rede.

A exceção encontrada pode ser justificada pela aleatoriedade dos pesos iniciais que são atribuídos durante o treinamento. Contudo, se compararmos a performance da RNA-50_{1D} com arquitetura 10-9-1 com a performance da mesma rede mudando a arquitetura para 10-10-1, que também foi obtida durante os testes, é possível perceber que a diferença entre os parâmetros não é muito significativa. Essa comparação pode ser vista na tabela abaixo.

Tabela 4 - Comparação entre duas arquiteturas da RNA-50_{1D}

Rede	Porcentagem usada no treinamento (%)	Dias a frente	Arquitetura			Treinamento			Simulação		
			nne	nnint	nns	RMSE (m ³ /s)	CE	MARE	RMSE (m ³ /s)	CE	MARE
RNA-50 _{1D}	50	1	10	9	1	26,3780	0,9800	0,0543	171,6300	0,6687	0,0800
RNA-50 _{1D}	50	1	10	10	1	26,4700	0,9799	0,0542	206,2400	0,5216	0,0837

*nne - número de neurônios de entrada; nnint - número de neurônios intermediários e nns - número de neurônios de saída.

Fonte: Própria (2019).

Ou seja, observando de uma forma global, a tendência é que com o aumento do número de neurônios nas camadas, a performance resultante é melhor, porém, devido a aleatoriedade que os pesos iniciais são gerados durante o treinamento, algumas exceções podem surgir, como no exemplo acima. Além disso, as redes treinadas com menores porcentagem de dados durante a sua calibração tendem a apresentar melhor capacidade de aprender as relações entre os dados.

4.2.1.2. Simulações

Em relação ao desempenho das simulações, ou seja, das previsões feitas com dados de vazões que não foram utilizados no treinamento, os gráficos comparativos entre vazões estimadas e vazões observadas podem ser vistos da Figura 29 até a Figura 33.

Por meio da análise dos gráficos percebe-se que os menores erros foram justamente para as vazões inferiores a 400 m³/s, com uma diferença entre valores observados e simulados na ordem 30 m³/s, correspondendo a erros percentuais de até 19%. Para as vazões maiores que 400 m³/s até 1000 m³/s os erros foram mais consideráveis, chegando a diferenças de 200 a 400 m³/s, com erros percentuais da ordem de 25%. E para os períodos mais críticos, apesar da boa performance durante o treinamento, as simulações não foram boas. É possível perceber que a rede não conseguiu prever as vazões de forma aceitável, inclusive retornando valores de vazões negativos em algumas das redes.

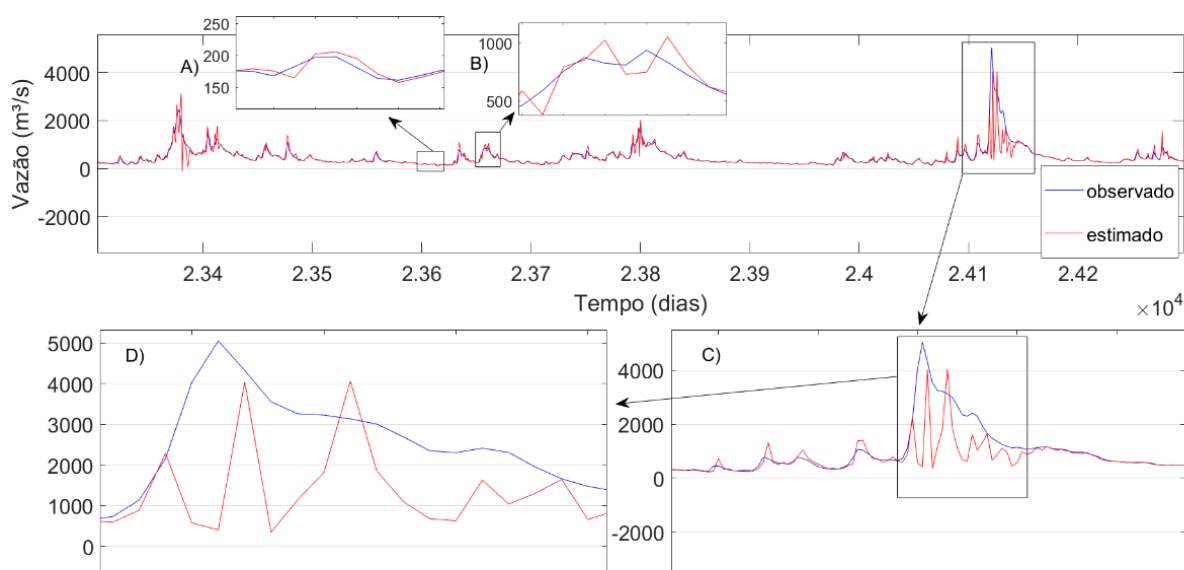


Figura 29 - Simulação RNA-10_{1D}

Fonte: Própria (2019).

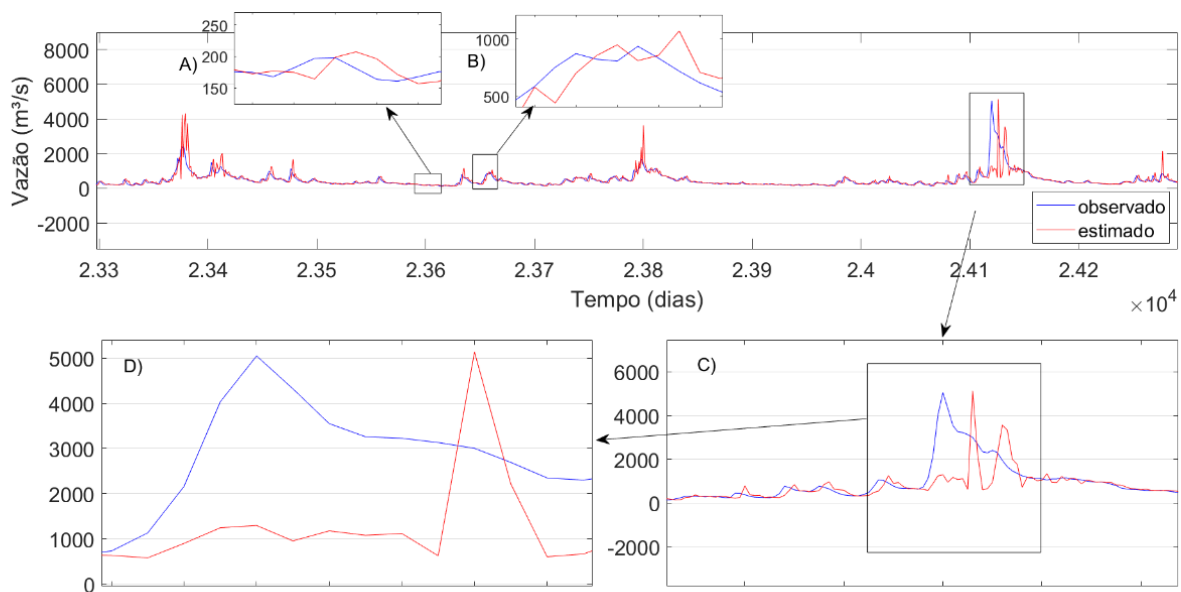


Figura 30 - Simulação RNA-20_{1D}

Fonte: Própria (2019).

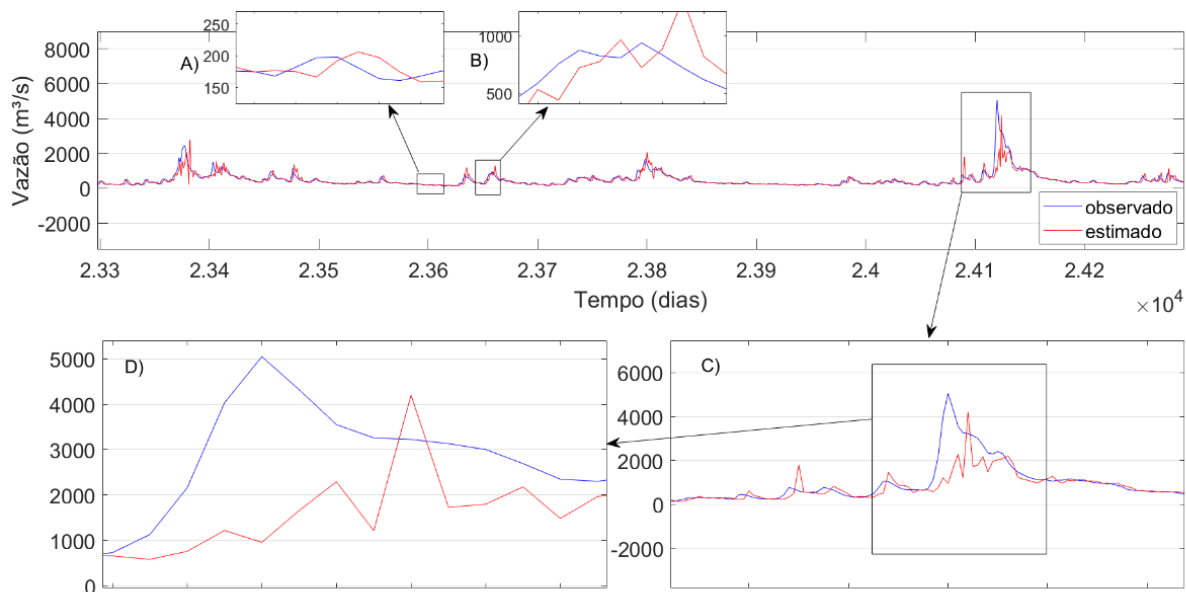


Figura 31 - Simulação RNA-30_{1D}

Fonte: Própria (2019).

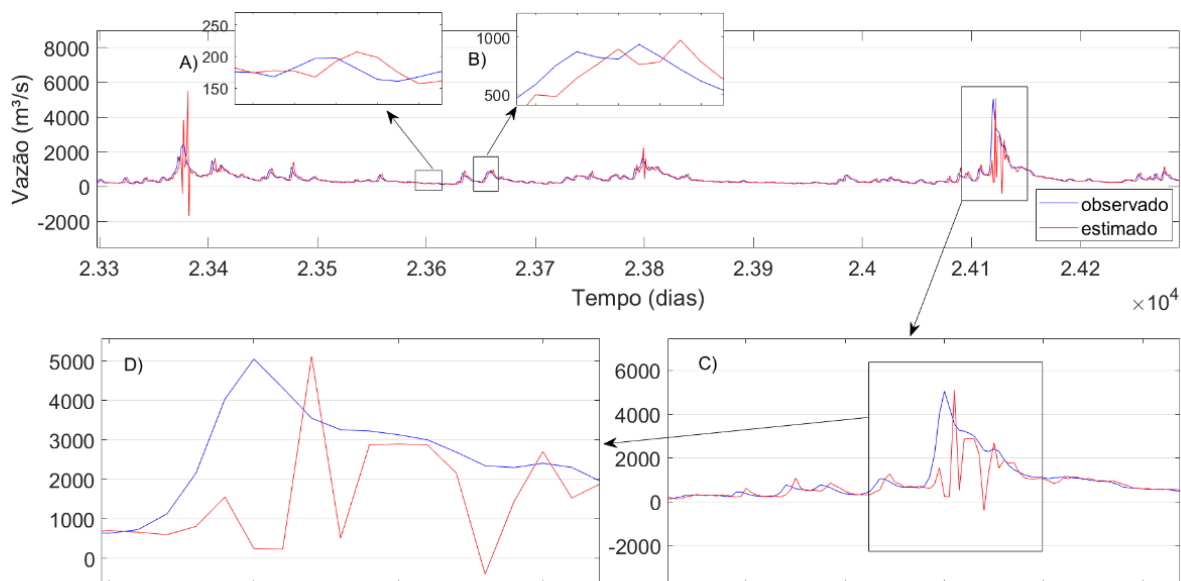


Figura 32 - Simulação RNA-40_{1D}

Fonte: Própria (2019).

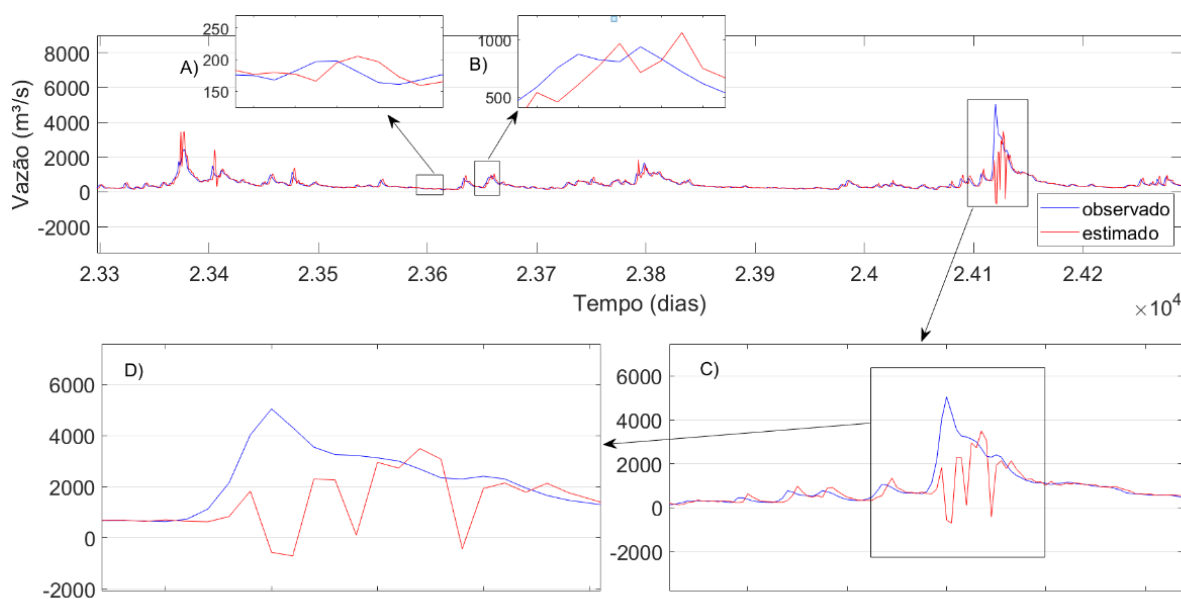


Figura 33 - Simulação RNA-50_{1D}

Fonte: Própria (2019).

Observando a Tabela 3, é possível perceber que a performance das redes durante a simulação foi inferior a performance do treinamento. Da mesma forma que ocorreu no treinamento, as redes calibradas com menor quantidade de dados obtiveram melhor desempenho, com exceção da RNA-20_{1D}, que só ficou na frente da RNA-50_{1D}. A RNA-10_{1D} foi a que obteve melhor desempenho, uma vez que apresentou

menor valor de RMSE e foi a única RNA que não apresentou defasagem significativa entre a curva dos dados observados e dos dados estimados.

4.2.2. Previsões para 3 dias a frente

4.2.2.1. Treinamento

Por meio da análise do parâmetro RMSE e por meio da análise dos gráficos obtidos no treinamento, as melhores arquiteturas para cada uma das RNAs e os parâmetros estatísticos obtidos durante o treinamento podem ser vistas de forma resumida na Tabela 5.

Tabela 5 - Melhores arquiteturas e parâmetros obtidos para 3 dias de previsão futura

Rede	Porcentagem usada no treinamento (%)	Dias a frente	Arquitetura			Treinamento			Simulação		
			nne	nnint	nns	RMSE (m ³ /s)	CE	MARE	RMSE (m ³ /s)	CE	MARE
RNA-10 _{3D}	10	3	10	10	1	65.68	0.89045	0.12327	190.79	0.45703	0.20364
RNA-20 _{3D}	20	3	10	10	1	59.209	0.89541	0.1202	219.03	0.3233	0.21609
RNA-30 _{3D}	30	3	10	10	1	60.857	0.88049	0.12935	208.98	0.42321	0.20657
RNA-40 _{3D}	40	3	10	10	1	67.948	0.85682	0.13587	197.13	0.52823	0.20715
RNA-50 _{3D}	50	3	10	10	1	72.894	0.84718	0.14615	203	0.53649	0.2084

*nne - número de neurônios de entrada; nnint - número de neurônios intermediários e nns - número de neurônios de saída.

Fonte: Própria (2019).

Os gráficos comparativos entre as vazões observadas e as vazões estimadas durante o treinamento, da melhor arquitetura obtida para cada uma das RNAs, considerando três dias de previsão no futuro constam da Figura 34 até a Figura 38.

Da mesma forma que ocorreu na análise das RNAs para previsão de um dia, as redes neurais que apresentaram melhor desempenho foram aquelas com a maior quantidade de neurônios nas camadas de entrada e camada intermediária. Todas as RNAs escolhidas como melhor performance possuem 10 neurônios em cada uma dessas camadas.

Na análise dos gráficos comparativos entre simulações e dados observados para o período de treinamento para as vazões inferiores a 400 m³/s, destacas nos

“detalhes A”, em todas as configurações de RNAs testadas os dados simulados divergiram dos dados reais de forma razoável, hora sendo subestimados, hora sendo superestimados. Os erros variaram de 10 a 50 m³/s, o que equivale a erros percentuais da ordem de 5 a 30%, tanto para mais como para menos

Já em relação as vazões acima de 400 m³/s e inferiores a 850 m³/s, destacadas nos “detalhes C”, em todas as RNAs treinadas houve uma correlação ruim entre dados simulados e observados. Foram constatados erros da ordem de 100 a 200 m³/s, tanto subestimando como superestimando as vazões. Essas diferenças corresponderam a erros percentuais de 20 a 40%, tanto para mais como para menos.

E para os picos de vazões críticos, com vazões de 850 a 1600 m³/s, destacados nos “detalhes B”, é possível perceber que as RNAs treinadas com 10, 20 e 30% apresentaram uma boa aproximação entre os dados da rede e os observados, com erros da ordem de 80 m³/s, o que corresponde aproximadamente a um erro percentual de 5 a 8%. Contudo os erros foram maiores para as RNAs treinadas com 40 e 50%, sendo possível detectar uma subestimação considerável desses picos extremos, chegando a erros percentuais de até 60%.

Assim, foi possível perceber, por meio dos detalhes A, B e C, erros tanto para vazões consideradas normais como vazões consideradas críticas. Apesar dos erros na determinação dos valores, as redes neurais foram capazes de acompanhar a tendência de crescimento e decrescimento das vazões. Graficamente foi possível perceber uma defasagem horizontal entre a curva observada em relação a curva estimada.

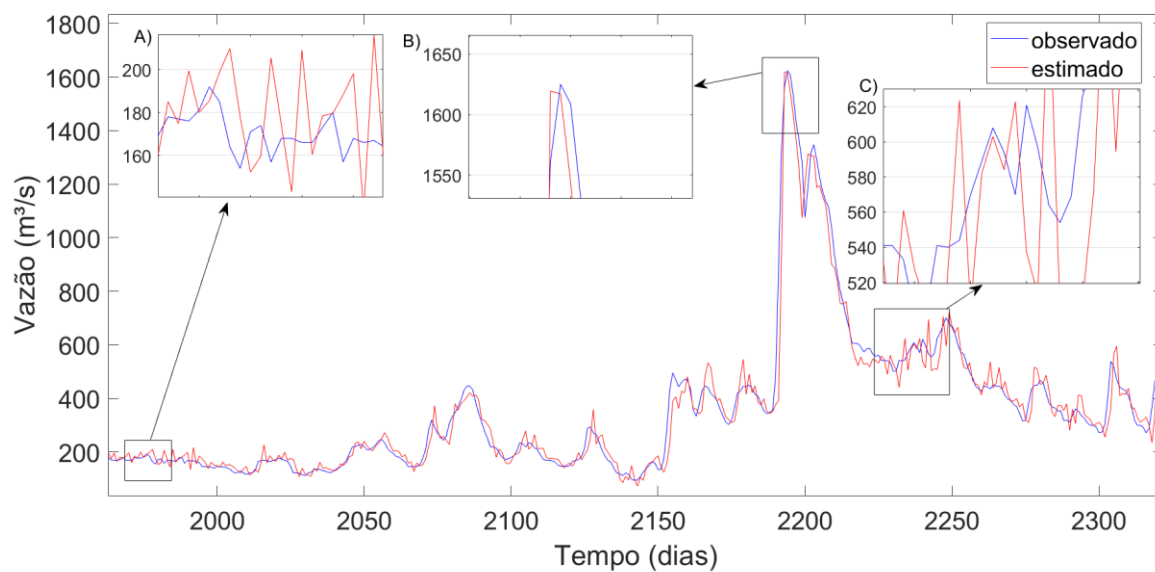


Figura 34 - Treinamento RNA-10_{3D}

Fonte: Própria (2019).

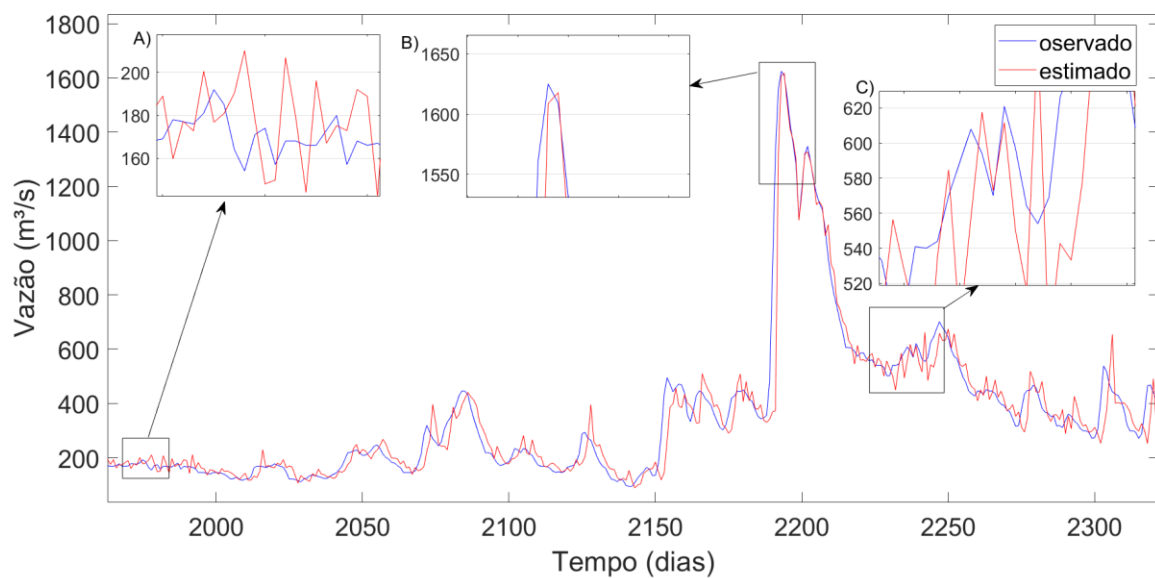
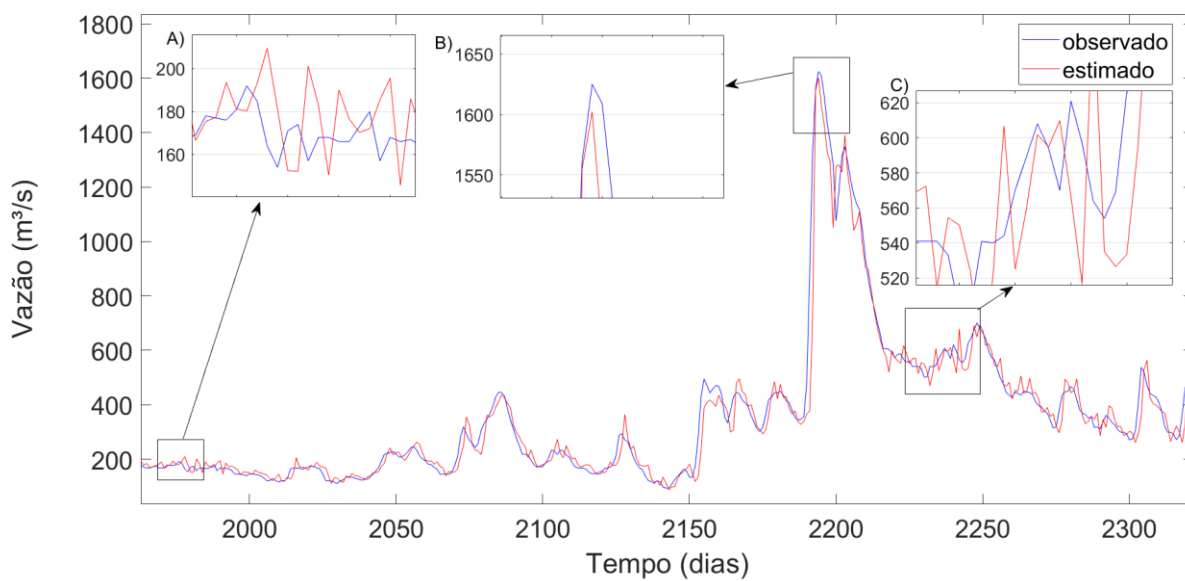
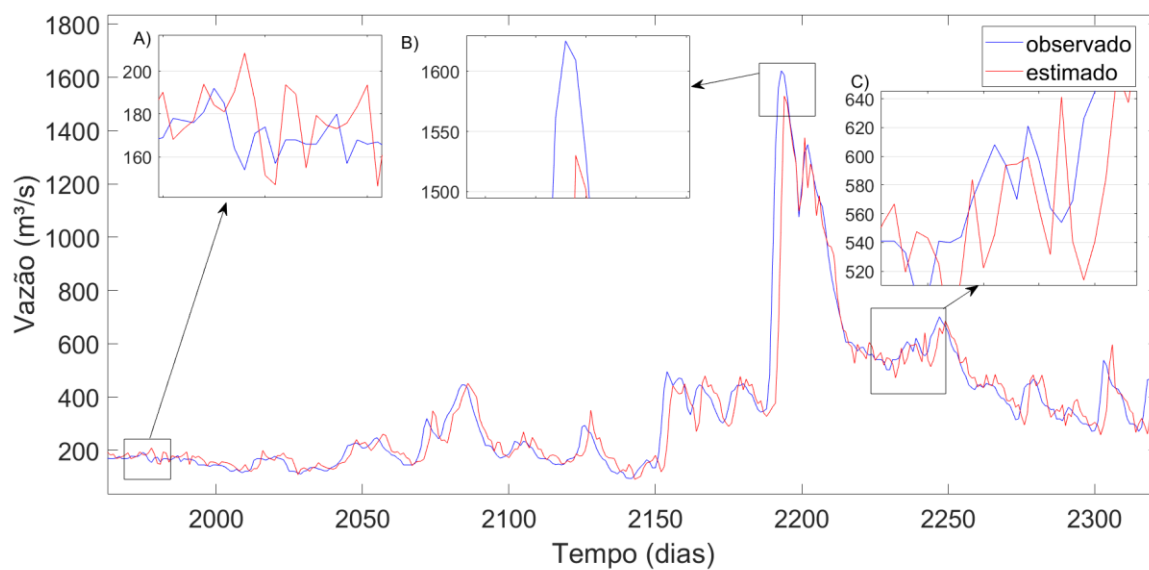


Figura 35 - Treinamento RNA-20_{3D}

Fonte: Própria (2019).

Figura 36 - Treinamento RNA-30_{3D}

Fonte: Própria (2019).

Figura 37 - Treinamento RNA-40_{3D}

Fonte: Própria (2019).

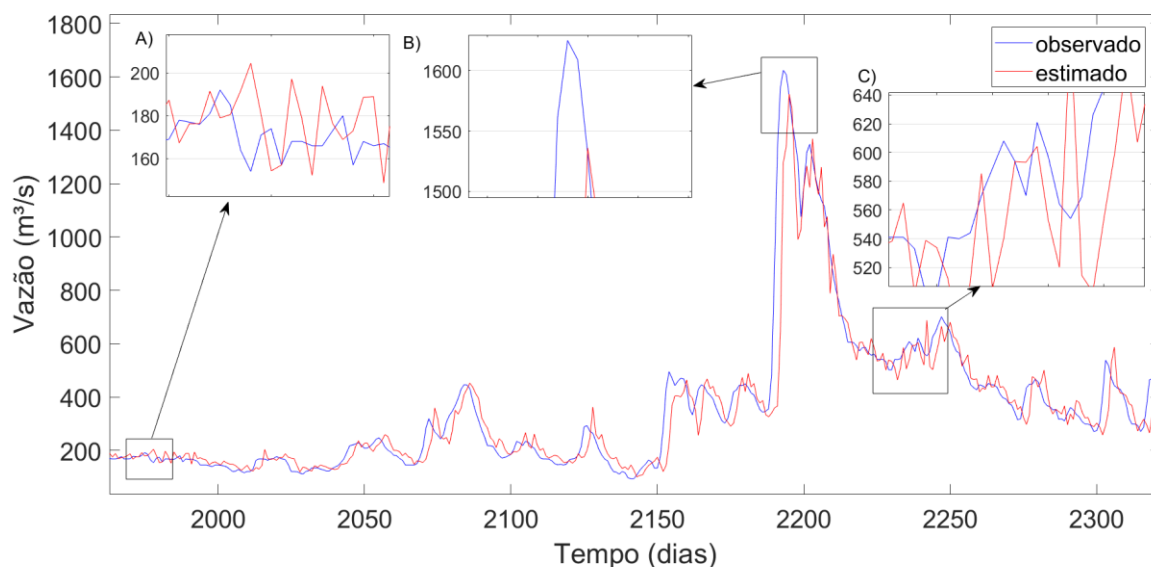


Figura 38 - Treinamento RNA-50_{3D}

Fonte: Própria (2019).

Além disso, por meio da análise do RMSE, percebe-se que a melhor RNA foi a RNA-20_{3D}, que apresentou o menor valor desse parâmetro, sendo seguida pela RNA-30_{3D} e RNA-10_{3D}. Ou seja, da mesma forma que aconteceu nas previsões de um dia a frente, as RNAs que utilizaram menor porcentagem de dados na calibração de seus pesos foram as que demonstraram melhor performance.

4.2.2.2. Simulações

Em relação ao conjunto de simulação, os gráficos que mostram a comparação das simulações com os dados reais podem ser vistos da Figura 39 até a Figura 43.

É possível perceber que tanto para as vazões abaixo de 400 m³/s como para as vazões entre 400 e 1000 m³/s, os erros foram da mesma ordem dos constatados durante o treinamento.

Contudo, em relação as vazões de 1000 m³/s até 5000 m³/s, que são consideradas vazões críticas, os erros aumentaram bastante. Foram constatados erros de até 50%, sendo bem superiores aos erros observados durante o treinamento. Isso pode ser justificado pela ausência de dados de vazões superiores a 1600 m³/s durante a calibração, mostrando que a rede foi incapaz de simular vazões extremas

de forma satisfatória para amplitudes de dados que não foram apresentadas a ela anteriormente. E da mesma forma que ocorreu no treinamento, foi possível perceber uma defasagem das curvas que representam os dados observados e os simulados.

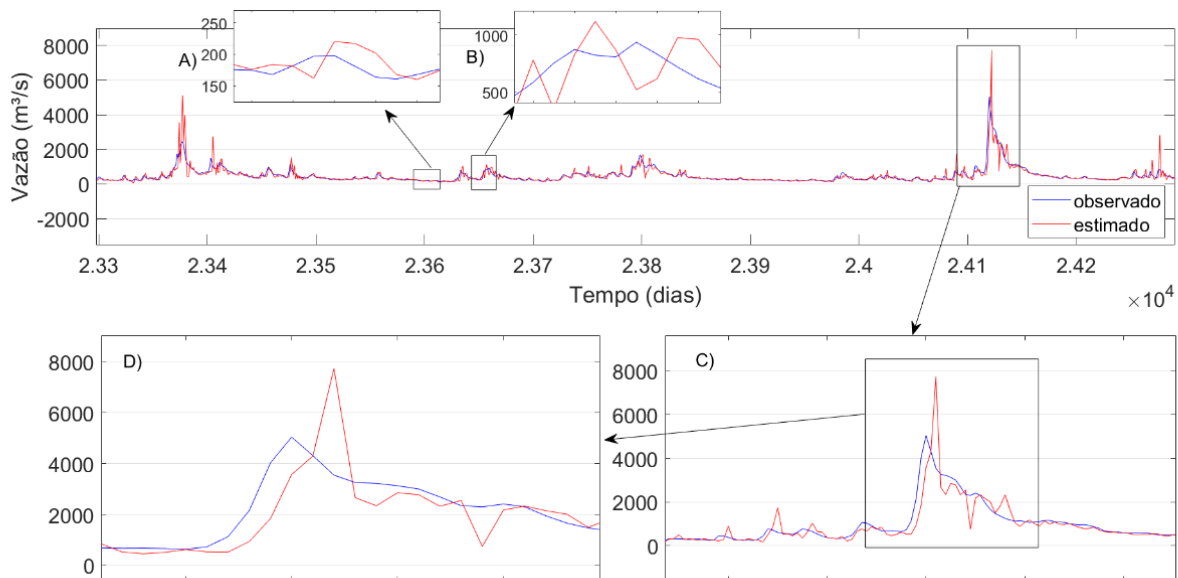


Figura 39 - Simulação RNA-10_{3D}

Fonte: Própria (2019).

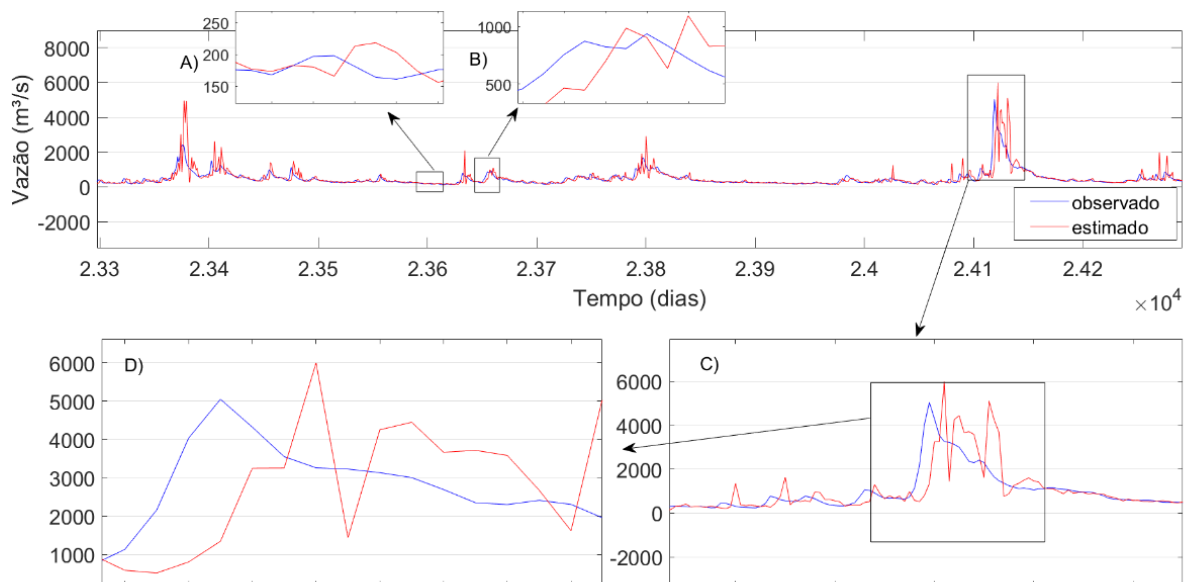


Figura 40 - Simulação RNA-20_{3D}

Fonte: Própria (2019).

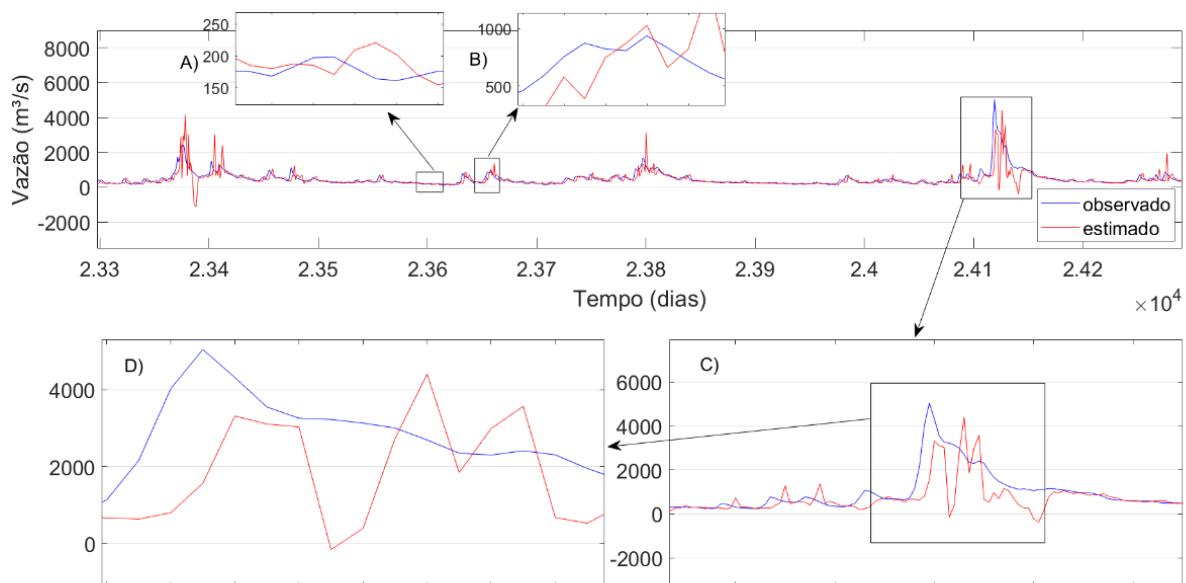


Figura 41 - Simulação RNA-30_{3D}

Fonte: Própria (2019).

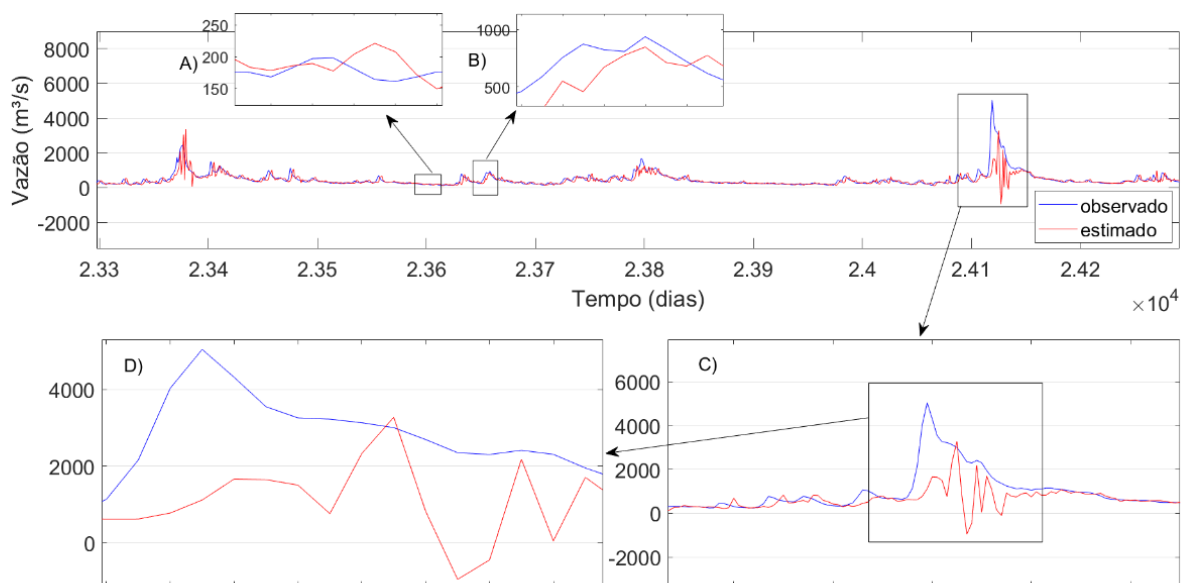


Figura 42 - Simulação RNA-40_{3D}

Fonte: Própria (2019).

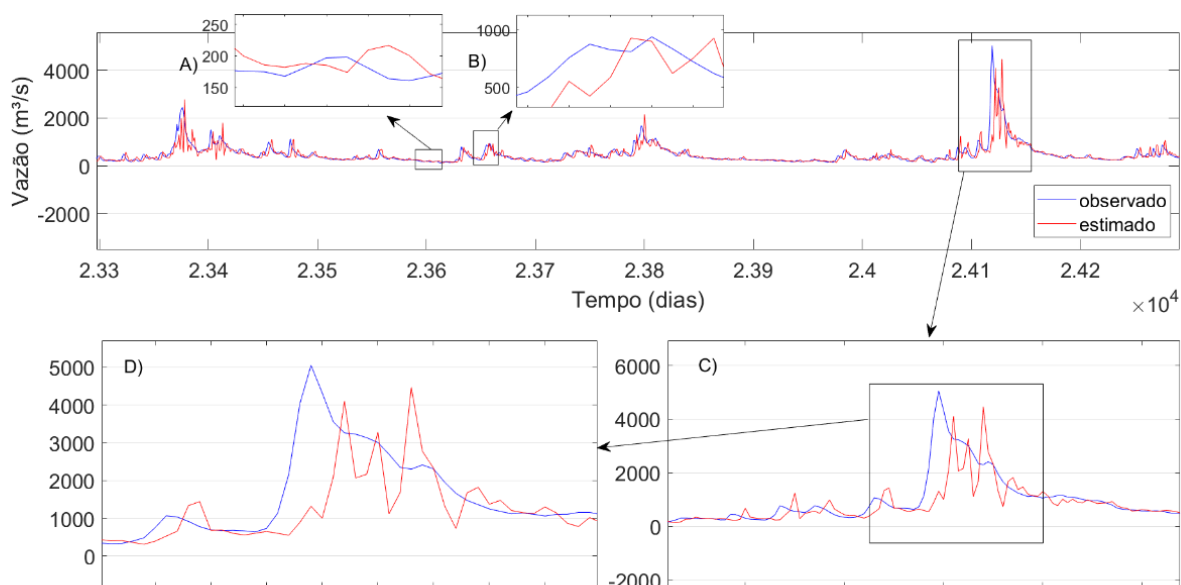


Figura 43 - Simulação RNA-50_{3D}

Fonte: Própria (2019).

Por fim, realizando uma comparação entre as redes por meio da análise do RMSE, é possível perceber que a rede com melhor performance foi a RNA-10_{3D}, contudo as redes RNA-20_{3D} e RNA-30_{3D} obtiveram uma performance inferior às redes RNA-40_{3D} e RNA-50_{3D}. Isso contrariou a tendência constatada até o momento de que quanto menor a porcentagem durante o treinamento, melhor a performance nas simulações. Uma possível justificativa disso são os erros de simulação exagerados que ocorreram para as vazões acima de 1000 m³/s nas redes RNA-20_{3D} e RNA-30_{3D}. Como no parâmetro RMSE a diferença entre os valores observados e os previstos é elevada ao quadrado, as grandes diferenças individuais resultaram em um menor valor de RMSE para essas duas redes.

4.2.3. Previsões para 7 dias a frente

4.2.3.1. Treinamento

Por meio da análise do parâmetro RMSE e por meio da análise dos gráficos obtidos no treinamento, as melhores arquiteturas para cada uma das RNAs e os

parâmetros estatísticos obtidos durante o treinamento podem ser vistas de forma resumida na Tabela 6.

Tabela 6 - Melhores arquiteturas e parâmetros obtidos para 7 dias de previsão futura

Rede	Porcentagem usada no treinamento (%)	Dias a frente	Arquitetura			Treinamento			Simulação		
			nne	nnint	nns	RMSE (m ³ /s)	CE	MARE	RMSE (m ³ /s)	CE	MARE
RNA-10 _{7D}	10	7	7	10	1	116.21	0.65703	0.2405	210.17	0.3411	0.31177
RNA-20 _{7D}	20	7	10	9	1	105.32	0.66888	0.23152	207.47	0.39284	0.29359
RNA-30 _{7D}	30	7	10	10	1	101.83	0.66524	0.239	218.16	0.37137	0.28372
RNA-40 _{7D}	40	7	10	9	1	110.18	0.6235	0.24516	250.71	0.2369	0.3094
RNA-50 _{7D}	50	7	10	10	1	117.95	0.59983	0.25794	261.22	0.23239	0.31265

*nne - número de neurônios de entrada; nnint - número de neurônios intermediários e nns - número de neurônios de saída.

Fonte: Própria (2019).

Os gráficos comparativos entre as vazões observadas e as vazões estimadas durante o treinamento, da melhor arquitetura obtida para cada uma das RNAs, considerando sete dias de previsão no futuro, podem ser vistos da Figura 44 até a Figura 48.

Da mesma forma que ocorreu na análise das RNAs para previsão de um e três dias, foi possível perceber uma tendência na melhora da performance com o aumento da quantidade de neurônios das camadas de entrada e intermediária. Contudo, nem todas as melhores arquiteturas foram as que possuíram o número máximo de 10 neurônios. O exemplo mais relevante disso foi a RNA-10_{7D}, que obteve a melhor arquitetura com sete neurônios em sua camada de entrada.

É possível perceber também, por meio do RMSE, que as RNAs treinadas com menores porcentagem de dados mostraram melhor desempenho, com exceção da RNA-10_{7D} que ficou na frente apenas da RNA-50_{7D}.

Em relação as vazões abaixo de 400 m³/s, os erros foram da ordem 20%, tanto para mais como para menos. Já para as vazões entre 400 e 850 m²/s, os erros foram menores, sendo aproximadamente de 10%. E para as vazões superiores a 850 m³/s, os erros foram de cerca de 20%, com tendência a subestimação dos dados previstos.

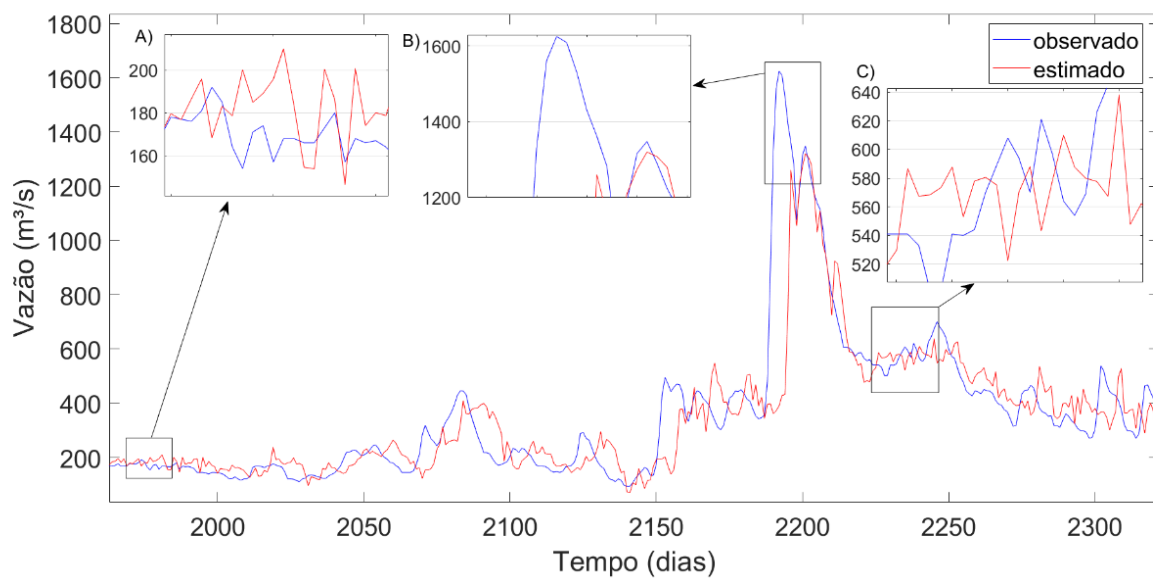


Figura 44 - Treinamento RNA-10_{7D}

Fonte: Própria (2019).

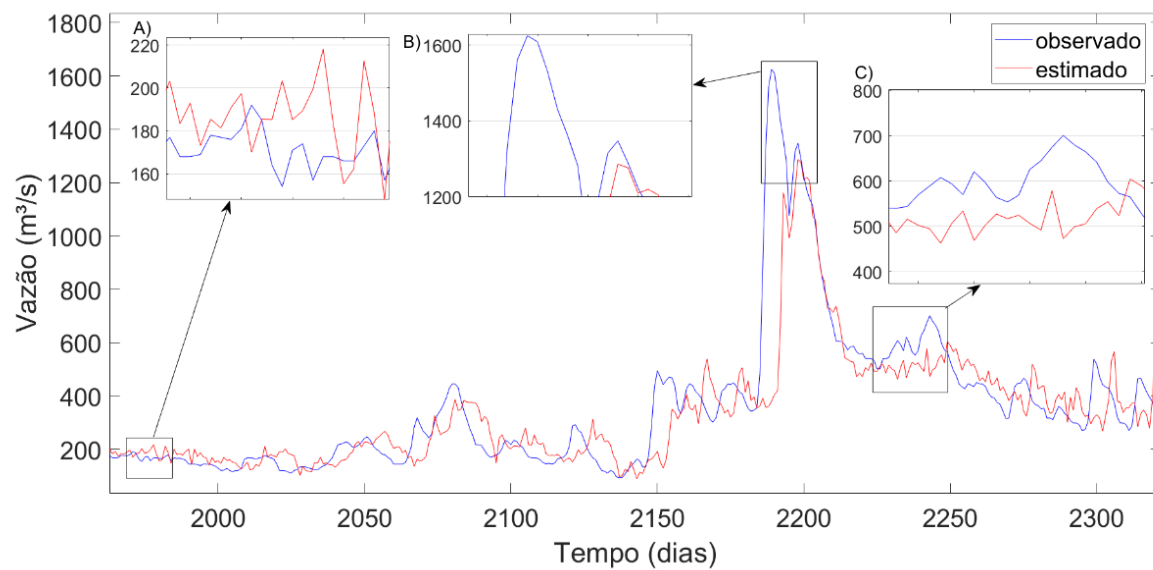


Figura 45 - Treinamento RNA-20_{7D}

Fonte: Própria (2019).

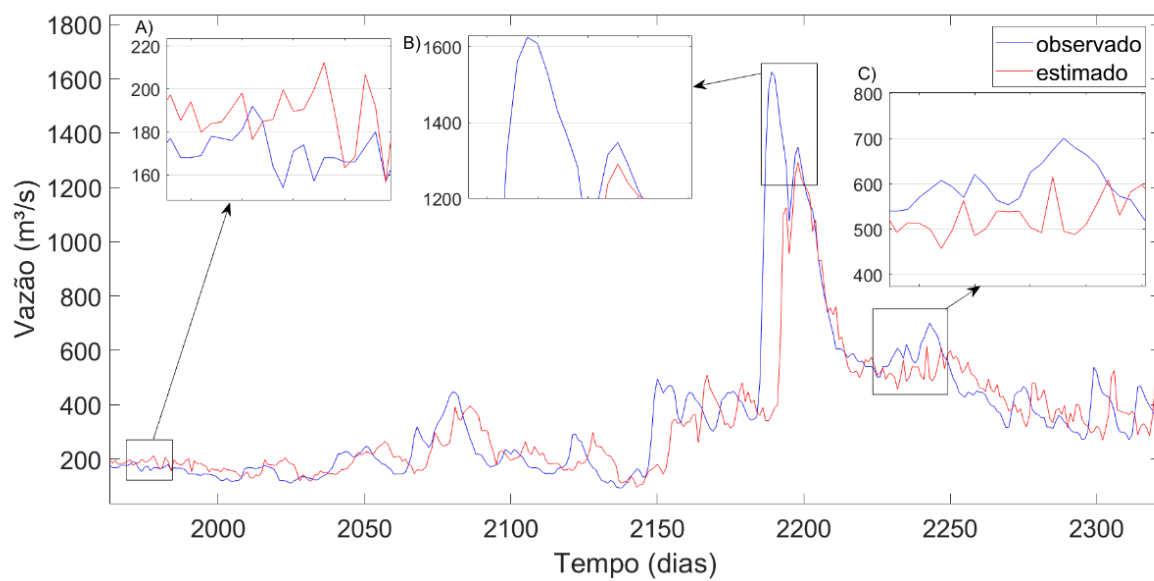


Figura 46 - Treinamento RNA-30_{7D}

Fonte: Própria (2019).

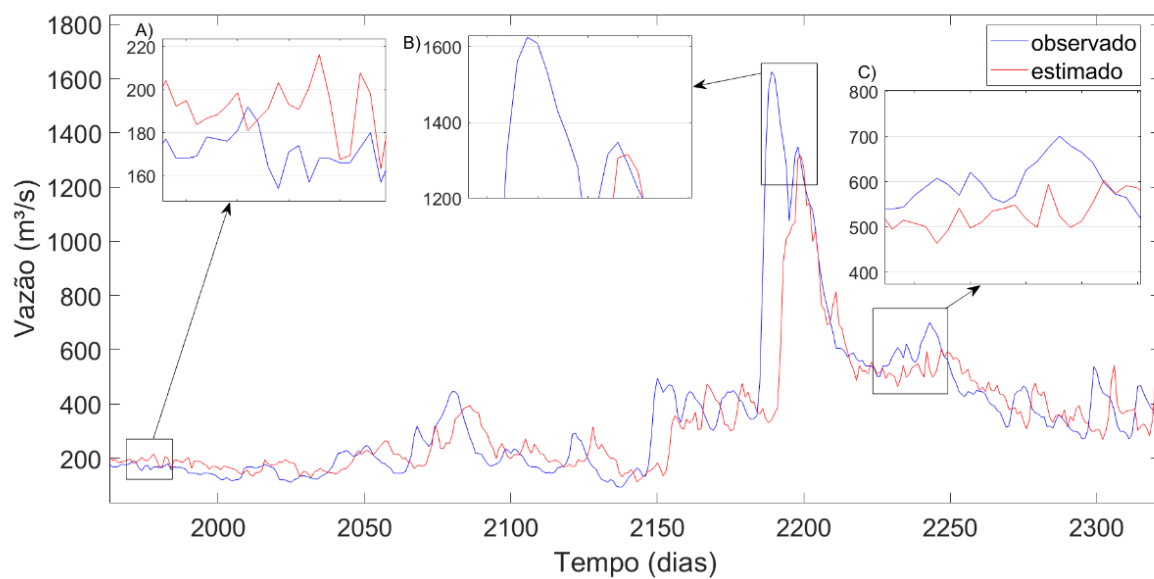


Figura 47 - Treinamento RNA-40_{7D}

Fonte: Própria (2019).

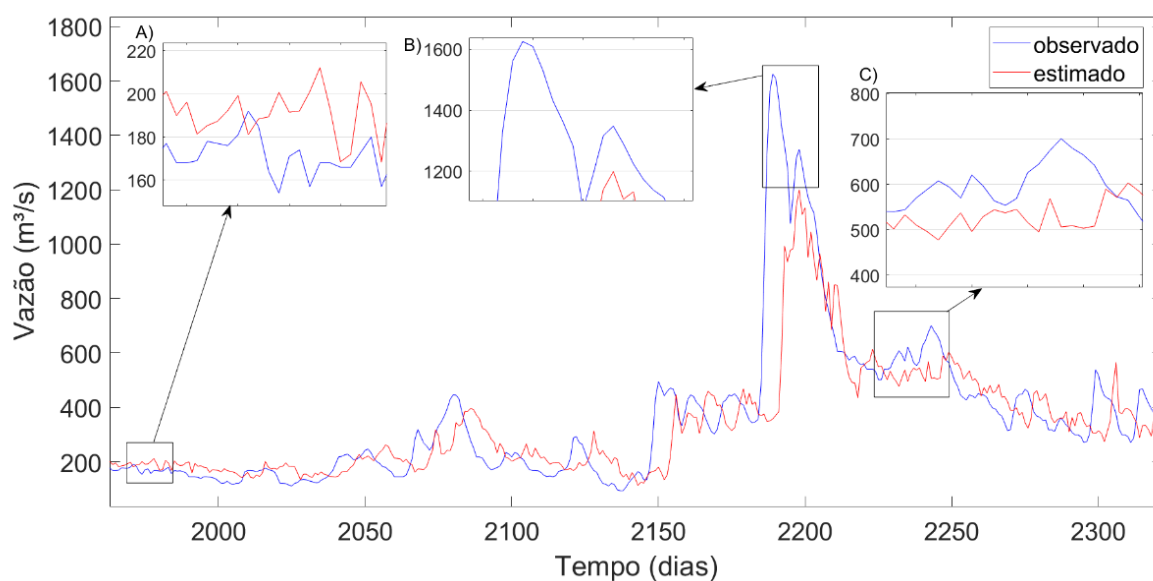


Figura 48 - Treinamento RNA-50_{7D}

Fonte: Própria (2019).

4.2.3.2. Simulações

Os gráficos que mostram a comparação entre as vazões simuladas e as observadas correspondem da Figura 49 até a Figura 53.

Por meio de uma análise visual, é possível perceber que a rede não conseguiu generalizar as simulações, visto que tanto para vazões consideradas normais como para vazões críticas existem erros consideráveis entre dados reais e simulados. Além disso, a defasagem entre a curva que representa os dados estimados e a curva dos dados observados ficou ainda maior.

Para as vazões até 400 m³/s, os erros percentuais chegaram a valores de até 50%, para mais como também para menos. Para as vazões de 400 até 1000 m³/s não foi diferente. Foram constatados erros percentuais de até 100% entre os valores reais e os obtidos pela simulação. Da mesma forma, para as vazões superiores a 1000 m³/s é possível perceber uma grande oscilação nos valores simulados, hora superestimando, hora subestimando, mostrando que a rede não conseguiu captar a relação entre vazões passadas e futuras de forma eficiente.

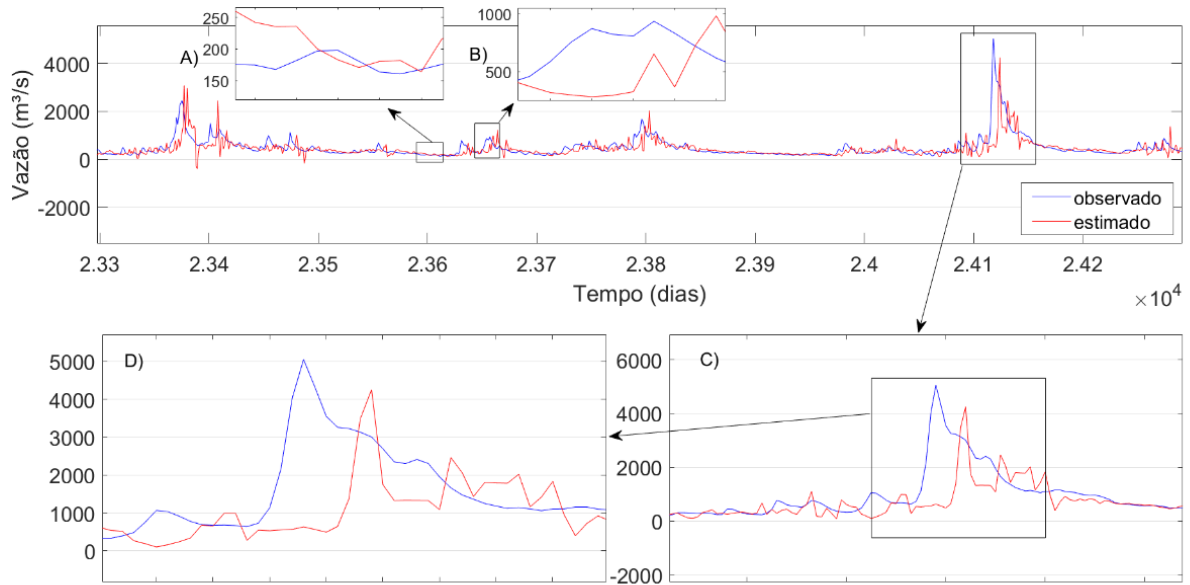


Figura 49 - Simulação RNA-107D

Fonte: Própria (2019).

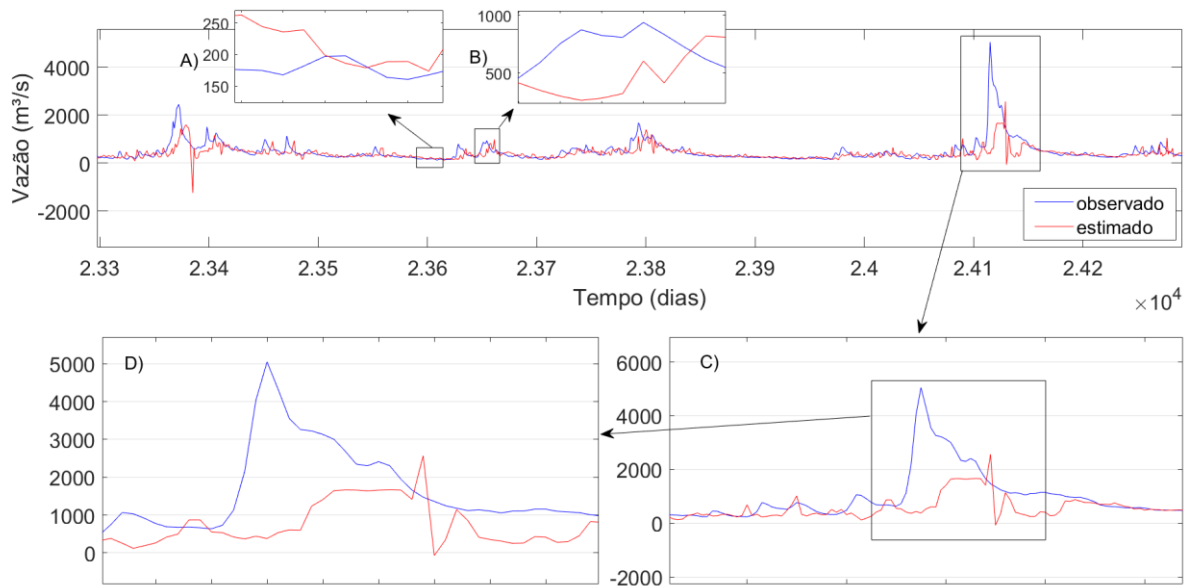


Figura 50 - Simulação RNA-207D

Fonte: Própria (2019).

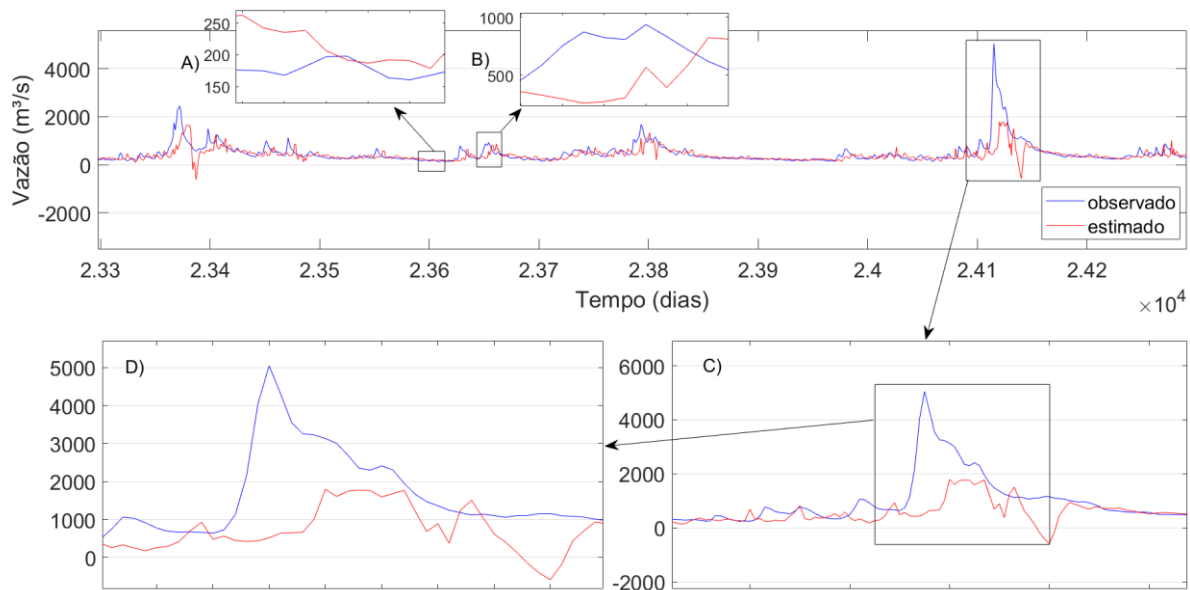


Figura 51 - Simulação RNA-307D

Fonte: Própria (2019).

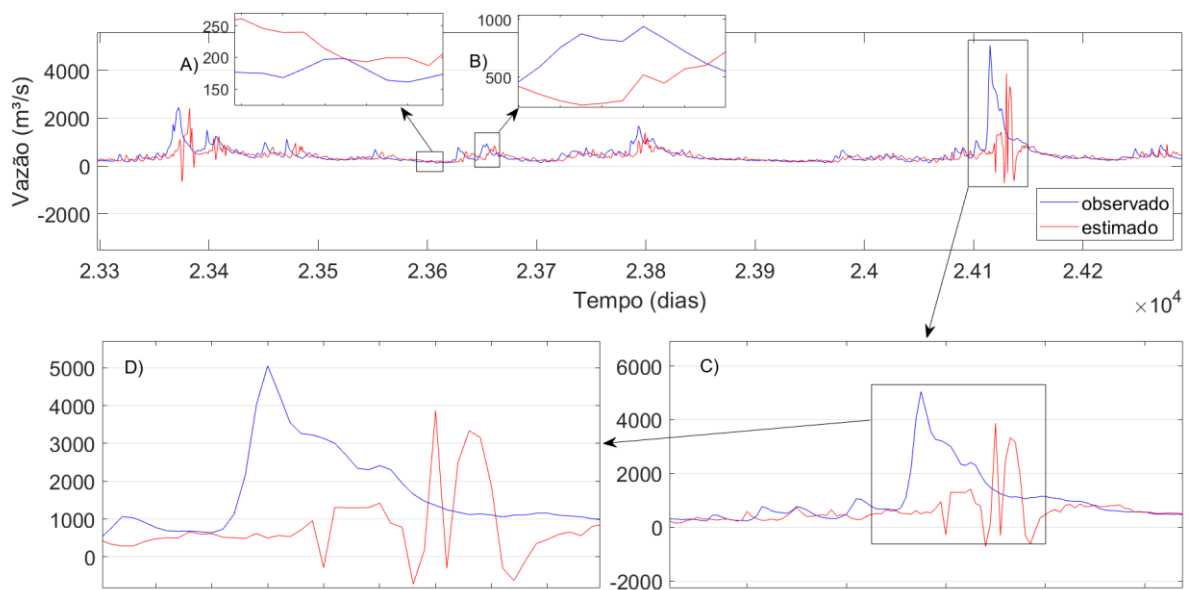


Figura 52 - Simulação RNA-407D

Fonte: Própria (2019).

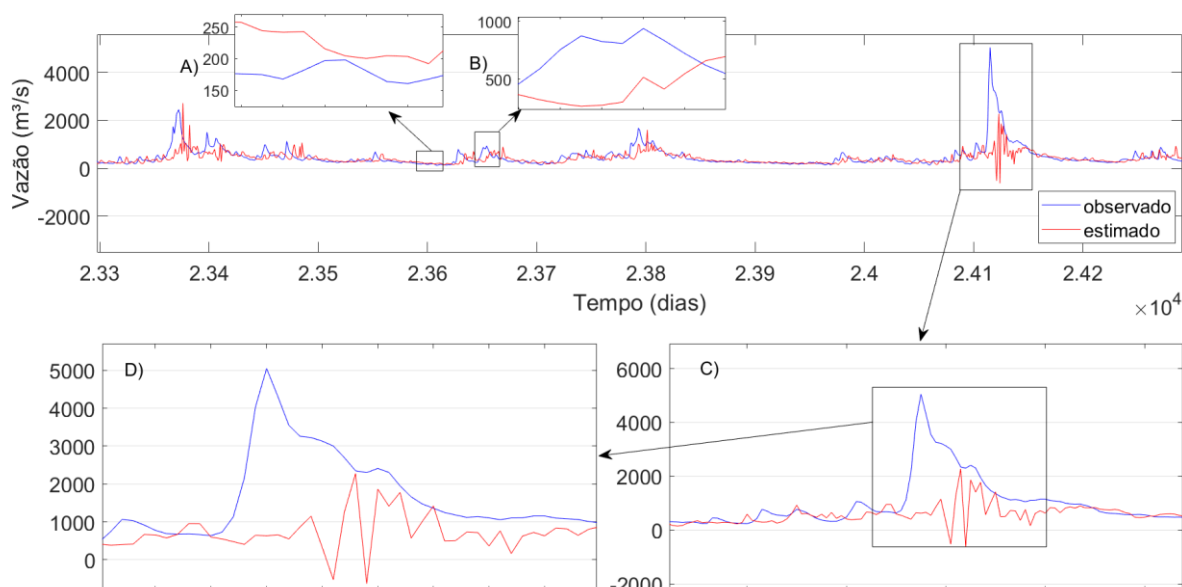


Figura 53 - Simulação RNA-50_{7D}

Fonte: Própria (2019).

Comparando as diferentes RNAs, é possível perceber que o comportamento para as simulações até 1000 m³/s foi relativamente semelhante para todas elas, ou seja, variar a porcentagem dos dados durante o treinamento não pareceu impactar de forma significativa nessas previsões. Já para as vazões superiores a 1000 m³/s, as simulações das redes neurais divergiram bastante, alguma delas tenderam a superestimar, outras a subestimar, contudo não foi possível estabelecer uma relação clara entre as porcentagens do conjunto de treinamento e essas tendências.

Apesar disso, tomando o parâmetro RMSE como base, é possível concluir que as redes treinadas com menor porcentagem obtiveram melhor performance, visto que a RNA-10_{7D}, RNA-20_{7D} e RNA-30_{7D} apresentaram valores menores para esse indicador, como também valores maiores para o CE e menores para o MARE.

5) CONCLUSÕES

Após as análise estatística feita, foi possível concluir que a caracterização das vazões críticas adotada para esse trabalho foi adequada, pois o valor de referência acima do qual as vazões foram consideradas críticas de 850 m³/s realmente ocorreu poucas vezes ao longo da série histórica como também teve um valor bem significativo

acima da média das vazões, como pode ser visto respectivamente por meio da Tabela 2 e da Tabela 1.

A análise de diferentes RNAs, com arquiteturas variadas, com porcentagens de dados diferentes durante o treinamento e com intervalos de previsões futuros distintos possibilitou a detecção de uma tendência para as melhores arquiteturas, tanto em relação a quantidade de neurônios que otimizam as simulações como a porcentagem de dados do treinamento que resultam em melhores performances.

Foi possível perceber que, de um modo geral, à medida que aumentou a quantidade de neurônios nas camadas de entrada e nas camadas intermediárias, a performance da rede melhorou. Da mesma forma, dentre as diferentes porcentagens de dados utilizados no treinamento, a tendência observada foi que para menores porcentagens da série histórica utilizadas na calibração, melhores foram os desempenhos das RNAs, o que pode ser observado pela análise do RMSE, que geralmente foi menor para as RNAs que utilizaram 10, 20 e 30% quando comparadas as RNAs que utilizaram 40 e 50%. Não obstante, algumas exceções foram percebidas. Isso foi um resultado inesperado, visto que essas redes foram treinadas com menos exemplos.

Outra característica observada foi a perda da correlação entre vazões passadas e futuras à medida que aumentou o horizonte das previsões. Para as vazões de até 1000 m³/s, as aproximações foram boas para previsões de um dia a frente. Piorando consideravelmente à medida que ampliou para três dias e posteriormente para sete dias. Já para as vazões acima de 1000 m³/s, nenhuma das RNAs apresentou resultados satisfatórios para nenhum dos intervalos futuros testados.

Dessa forma, é possível concluir que a utilização de séries curtas de dados no treinamento das redes neurais com o objetivo de prever vazões críticas não se mostrou eficiente.

Contudo, foi possível concluir que a busca por melhores aproximações deve priorizar o aumento do número de neurônios nas camadas de entrada e intermediária limitando-se a períodos de previsões futuras não muito distantes.

Além disso, é possível afirmar que a utilização de rede neurais treinadas com séries curtas de dados na previsão de vazões consideradas típicas para períodos

próximos de previsões futuras é uma alternativa vantajosa, visto que a única informação fornecida a rede foi dados de vazões, enquanto que modelos fenomenológicos necessitam de mais variáveis e apresentam grau de complexidade elevado e as vezes são de difícil solução para a computação convencional (FRANCISCO, 2000; GOMES; MONTENEGRO; VALENÇA, 2010; MORAIS; BORGES, 2010).

Por fim, vale salientar que a melhor performance para as RNAs treinadas com as porcentagens de 10, 20 e 30%, o que aconteceu na maioria dos casos, foi um resultado inesperado, visto que essas redes foram treinadas com menos exemplos. Uma possível justificativa pode ser a variação sazonal. Como essas redes foram expostas a menos anos, o conjunto de dados apresentados apresentaram menores variações, facilitando o ajuste dos pesos. Contudo, um estudo mais profundo é necessário para que conclusões mais sólidas sejam obtidas.

5.1. Trabalhos futuros

Para trabalhos futuros recomenda-se ampliar o número de neurônios nas camadas para valores superiores a dez. Ou seja, já que os neurônios são as unidades de processamento, recomenda-se treinar redes mais robustas.

Também devem ser testadas a utilização de outros algoritmos de aprendizagem, pois algoritmos diferentes podem levar a melhores ajustes de pesos e conseqüentemente melhores resultados das simulações.

Outra sugestão é a utilização de modelos híbridos, por exemplo transformadas wavelet e redes neurais artificiais como também testar outros tipos de redes neurais, como por exemplo redes neurais artificiais em cascata, já que nesse trabalho só foi analisada um único tipo de rede.

Em relação ao conjunto de dados, sugere-se utilizar conjuntos de treinamento que embora curtos sejam mais representativos das vazões críticas da série histórica, visto que esse aspecto dos dados não foi levado em consideração nesse estudo.

Por fim, sugere-se realizar estudos que busquem verificar se menores conjuntos de dados irão tender a redes com melhores performances para outros reservatórios,

uma vez que no presente trabalho foi observada uma tendência de melhor performance para as redes treinadas com menores porcentagens de dados.

6) REFERÊNCIAS

ADAMOWSKI, Jan; SUN, Karen. Development of a coupled wavelet transform and neural network method for flow forecasting of non-per. **Journal Of Hydrology**. [S. l.], p. 85-91. 24 jun. 2010.

ANA - AGÊNCIA NACIONAL DE ÁGUAS. **Base Hidrográfica Ottocodificada da Bacia do Rio Paranapanema**. 2014. Disponível em: <<https://metadados.ana.gov.br/geonetwork/srv/pt/metadata.show?id=259&currTab=simple>>. Acesso em: 13 set. 2019.

ANA - AGÊNCIA NACIONAL DE ÁGUAS. **Plano Integrado de Recursos Hídricos da Unidade de Gestão de Recursos Hídricos Paranapanema**. [S. l.]: Ana, 2016. Disponível em: <<https://paranapanema.org/plano/admin/upload/documento/61.pdf>>. Acesso em: 22 ago. 2019.

BARRETO, Jorge M.. **Introdução às Redes Neurais Artificiais**. 2002. Laboratório de Conexionismo e Ciências Cognitivas UFSC - Departamento de Informática e de Estatística. Disponível em: <<http://www.inf.ufsc.br/~j.barreto/tutoriais/Survey.pdf>>. Acesso em: 11 ago. 2019.

BASHEER, I. A.; HAJMEER, M.. Artificial neural networks: fundamentals, computing, design, and application. **Journal of Microbiological Methods**, [S.l.], v. 43, n. 1, p.3-31, 2000.

BEN. **Relatório Síntese / Ano Base 2018**. Rio de Janeiro: EPE, 2019. Disponível em: <<http://www.epe.gov.br/sites-pt/publicacoes-dados-abertos/publicacoes/PublicacoesArquivos/publicacao->>>. Acesso em: 31 maio 2019.

BRAGA, Antônio de Pádua; CARVALHO, André Ponce de Leon F. de; LUDERMIR, Teresa Berbarda. **Redes Neurais Artificiais: Teoria e Aplicações**. 2. ed. Rio de Janeiro: LTC, 2007.

BRESSAN, Aureliano Angel. Tomada de decisão em futuros agropecuários com modelos de previsão de séries temporais. **Rae - Eletrônica**, São Paulo, v. 3, p.1-20, 09 jan./jun. 2004. Trimestral.

CINTRA, Evandro Cardoso. **Aplicação de Redes Neurais no Controle de Teores de Cobre e Ouro do Depósito de Chapada (GO)**. 2003. 155 f. Tese (Doutorado) - Curso de Geologia, Universidade Estadual Paulista, Rio Claro, 2003.

CTG BRASIL - CHINA THREE GORGES CORPORATION BRASIL. **Energia Hidrelétrica**. [20--]a. Disponível em: <<https://www.ctgbr.com.br/portfolio-item/energia-hidreletrica/>>. Acesso em: 22 ago. 2019.

CTG BRASIL - CHINA THREE GORGES CORPORATION BRASIL. **Usinas**. [20--]b. Disponível em: <<http://www.paranapanemaenergia.com.br/usinas/Paginas/Usinas.aspx>>. Acesso em: 21 ago. 2019.

DAWSON, C. W. et al. Flood estimation at ungauged sites using artificial neural networks. **Journal of Hydrology**, [S.l.], v. 319, n. 1-4, p.391-409, mar. 2006. Elsevier BV. <http://dx.doi.org/10.1016/j.jhydrol.2005.07.032>.

DEMUTH, Howard; BEALE, Mark. Neural Network Toolbox: For Use with MATLAB. 4. Natick, MA: The MathWorks, Inc., October 2004. Neural Network Toolbox User's Guide. Disponível em: http://cda.psych.uiuc.edu/matlab_pdf/nnet.pdf. Acesso em: 1 ago. 2019.

EHLERS, R.S. (2005) Análise de Séries Temporais. Departamento de Estatística, UFPR. Disponível em <http://leg.est.ufpr.br/~ehlers/notas>. Acesso em: 08 ago. 2019.

ENGEL, Paulo Martins. **Redes Neurais**: Porto Alegre: UFRGS, [20--]. Color. Disponível em: <<http://www.inf.ufrgs.br/~engel/data/media/file/cmp121/modelos.pdf>>. Acesso em: 26 ago. 2019.

EPE. **Matriz Energética e Elétrica**. [20--]. Disponível em: <<http://epe.gov.br/pt/abcdenergia/matriz-energetica-e-eletrica>>. Acesso em: 31 jul. 2019.

FERREIRA, Paloma Mara de Lima. **Influência da Seleção das Funções Objetivo na Calibração do Modelo Hidrológico Distribuído MGB-IPH**. 2017. 144 f. Dissertação (Mestrado) - Curso de Engenharia Civil e Ambiental, Centro de Tecnologia, Universidade Federal da Paraíba, João Pessoa, 2017.

FORESEE, F. Dan; HAGAN, Martin T.. Gauss-Newton approximation to Bayesian learning. In: PROCEEDINGS OF INTERNATIONAL CONFERENCE ON NEURAL NETWORKS (ICNN'97), 7., 1997, Houston. **Anais...** . Houston: IEEE, 1997. p. 1930 - 1935.

FRANCISCO, Claudio de Oliveira. **Modelagem e Simulação de um Secador Industrial de Gelatina Através de Redes Neurais Artificiais**. 2000. 100 f. Dissertação (Mestrado) - Curso de Engenharia Química, Universidade Estadual de Campinas, Campinas, 2000.

GOMES, Luana Ferreira Costa; MONTENEGRO, Suzana Maria Gico Lima; VALENÇA, Mêuser Jorge da Silva. Modelo Baseado na Técnica de Redes Neurais para Previsão de Vazões na Bacia do Rio São Francisco. **RBRH - Revista Brasileira de Recursos Hídricos**, Porto Alegre, v. 15, n. 1, p.05-15, jan./mar. 2010.

HONORATO, André Gustavo da Silva Melo. **Previsão de Vazões Diárias e Mensais Utilizando Modelos Híbridos de Transformadas Wavelet e Redes Neurais Artificiais**. 2017. 95 f. TCC (Graduação) - Curso de Engenharia Ambiental, Centro de Tecnologia, Universidade Federal da Paraíba, João Pessoa, 2017.

HONORATO, André Gustavo da Silva Melo; SILVA, Gustavo Barbosa Lima da; SANTOS, Celso Augusto Guimarães. Monthly streamflow forecasting using neuro-wavelet techniques and input analysis. **Hydrological Sciences Journal**, [S.l.], v. 63, n. 15-16, p.2060-2075, 10 dez. 2018.

HOPFIELD, J. J.. Neural networks and physical systems with emergent collective computational abilities. **Proceedings of The National Academy of Sciences**, Nova Jersey, v. 79, n. 8, p.2554-2558, 1 abr. 1982. <http://dx.doi.org/10.1073/pnas.79.8.2554>.

ITAPONNEWS. **Agora da chinesa STG, usina Chavantes completa 47 anos de operação**. 2017. Disponível em: <<https://www.itaponews.com.br/agora-da-chinesa-stg-usina-chavantes-completa-47-anos-de-operacao.html>>. Acesso em: 21 ago. 2019.

KAYRI, Murat. Predictive Abilities of Bayesian Regularization and Levenberg-Marquardt Algorithms in Artificial Neural Networks: A Comparative Empirical Study on

Social Data. **Mathematical And Computational Applications**, [S. l.], v. 21, n. 2, p.1-20, may 2016.

MAGNONI, Ana Paula Vidotto. **Ecologia trófica das assembléias de peixes do reservatório de Chavantes (Médio rio Paranapanema, SP/PR)**. 2009. 115 f. Tese (Doutorado) - Curso de Ciências Biológicas, Universidade Estadual Paulista, Botucatu, 2009.

MATHWORKS. **Improve Shallow Neural Network Generalization and Avoid Overfitting**. [20--]. Disponível em: <<https://www.mathworks.com/help/deeplearning/ug/improve-neural-network-generalization-and-avoid-overfitting.html>>. Acesso em: 03 sep. 2019.

MORAIS, Willy Ank de; BORGES, Herbert Christian. Uso de Inteligência Computacional no Desenvolvimento, Adequação e Controle de Qualidade de Aços Laminados a Quente na Usiminas Cubatão. **Tecnologia em Metalurgia e Materiais e Mineração**, São Paulo, v. 6, n. 4, p.230-236, abr./jun. 2010. Editora Cubo Multimidia. <http://dx.doi.org/10.4322/tmm.00604009>.

NEUSSER, Lukas. **Combinação Seletiva de Métodos para Previsão de Demanda a Curtíssimo Prazo em Tempo Real**. 2015. 131 f. Tese (Doutorado) - Curso de Engenharia Elétrica, Centro de Tecnologia, Universidade Federal de Santa Maria, Santa Maria, 2015.

NICOLAU, Lucas Feitosa. **Identificação e Previsão Neural de Atuadores com Memória de Forma na Presença de Deformação Residual Significativa**. 2018. 70 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Centro de Energias Alternativas e Renováveis, Universidade Federal da Paraíba, João Pessoa, 2018.

OLIVEIRA, Julio Cesar de et al. Avaliação de métodos para Reconstrução de Pixel com baixa qualidade em séries temporais de NDVI. In: XVI SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO - SBSR, 16., 2013, Foz do Iguaçu. **Anais...** . Foz do Iguaçu: INPE, 2013. p. 1727 - 1734.

OLIVEIRA, Tiago Prado. **Predição de Tráfego, Usando Redes Neurais Artificiais, para Gerenciamento Adaptativo de Largura de Banda em Roteadores**. 2014. 209 f. Dissertação (Mestrado) - Curso de Ciência da Computação, Universidade Federal de Uberlândia, Uberlândia, 2014.

ONS - OPERADOR NACIONAL DO SISTEMA. **Histórico de Operação**. 2019a. Disponível em: <<http://www.ons.org.br/paginas/resultados-da-operacao/historico-da-operacao>>. Acesso em: 22 ago. 2019.

ONS - OPERADOR NACIONAL DO SISTEMA. **Reservatórios**. 2019b. Disponível em: <<http://www.ons.org.br/paginas/energia-agora/reservatorios>>. Acesso em: 22 ago. 2019.

PIXABAY (Comp.). **Cérebro-neurônio-nervos-celular**. Disponível em: <<https://pixabay.com/pt/vectors/cérebro-neurônio-nervos-celular-2022398/>>. Acesso em: 08 set. 2019.

SARMENTO, Francisco Jácome. Modelagem de séries hidrológicas através de redes de neurônios. **Revista Brasileira de Recursos Hídricos**, Porto Alegre, v. 1, n. 2, p.19-31, jul./dez. 1996.

SILVA, Ivan Nunes da; SPATTI, Danilo Hernane; FLAUZINO, Rogerio Andrade. **Redes Neurais Artificiais para Engenharia e Ciências Aplicadas: Curso Prático**. Campinas: Artliber, 2010.

UYKAN, Zekeriya. Discrete Pseudo-SINR-Balancing Nonlinear Recurrent System. **Discrete Dynamics in Nature and Society**, [S.l.], v. 2013, n. 2013, p.1-19, 05 mar. 2013. Hindawi Limited. <http://dx.doi.org/10.1155/2013/480560>.

VON ZUBEN, Fernando J.; ATTUX, Romis R. F.. **Processamento Dinâmico em Redes Neurais Artificiais: Redes Neurais Recorrentes**. Campinas: Unicamp, 2007. Color. Departamento de Engenharia de Computação e Automação Industrial/Faculdade de Engenharia Elétrica e de Computação.