



MINISTÉRIO DA EDUCAÇÃO
Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Especialização em Ciência de Dados



PREDIÇÃO DO VOLUME DE OXIGÊNIO E PESO DE SÍNER EM CONVERTEDORES DO TIPO LD UTILIZANDO REDES NEURAI

Davison Patrocínio Cardoso

Eder Neves da Rocha

Vinicius de Castro Toledo

João Monlevade, MG

2023

Davison Patrocínio Cardoso

Eder Neves da Rocha

Vinicius de Castro Toledo

**PREDIÇÃO DO VOLUME DE OXIGÊNIO E PESO DE SÍNER EM
CONVERTEDORES DO TIPO LD UTILIZANDO REDES NEURAIAS**

Trabalho de conclusão de curso apresentado ao curso de Especialização em Ciência dos Dados do Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto, como parte dos requisitos necessários para a obtenção do título de Especialista em Ciência dos Dados.

Orientadora: Profa. Dra. Sarah Negreiros de Carvalho Leite

João Monlevade, MG

2023

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

R672p Rocha, Eder Neves da.
Predição do volume de oxigênio e peso de sínter em convertedores do tipo LD utilizando redes neurais. [manuscrito] / Eder Neves da Rocha. Davison Patrocinio Cardoso. Vinicius de Castro Toledo. - 2023.
101 f.: il.: color., gráf., tab..

Orientadora: Profa. Dra. Sarah Negreiros de Carvalho Leite.
Produção Científica (Especialização). Universidade Federal de Ouro Preto. Departamento de Engenharia de Produção.

1. Aço - Fabricação. 2. Aço - Indústria. 3. Redes neurais (Computação). 4. Usinas siderúrgicas. I. Cardoso, Davison Patrocinio. II. Toledo, Vinicius de Castro. III. Leite, Sarah Negreiros de Carvalho. IV. Universidade Federal de Ouro Preto. V. Título.

CDU 658.5:004

Bibliotecário(a) Responsável: Flavia Reis - CRB6-2431



FOLHA DE APROVAÇÃO

Davison Patrocínio Cardoso
Eder Neves da Rocha
Vinicius de Castro Toledo

Predição do volume de oxigênio e peso de sínter em convertedores do tipo LD utilizando redes neurais

Monografia apresentada ao Curso de Ciência dos Dados da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Especialista em Ciência dos Dados.

Aprovada em 28 de fevereiro de 2023.

Membros da banca

Profa. Dra. Sarah Negreiros de Carvalho Leite - Orientadora - Universidade Federal de Ouro Preto
Prof. Dr. Harlei Miguel de Arruda Leite - Universidade Federal de Ouro Preto
Me. João Domingos Guimarães de Athayde Júnior - USIMINAS

Profa. Dra. Sarah Negreiros de Carvalho Leite, orientadora do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 24/07/2023.



Documento assinado eletronicamente por **Sarah Negreiros de Carvalho Leite, PROFESSOR DE MAGISTERIO SUPERIOR**, em 24/07/2023, às 18:11, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0562093** e o código CRC **5348FEAD**.

Agradecimentos

Gostaria de expressar nossos sinceros agradecimentos a todas as pessoas que contribuíram para a conclusão deste trabalho de conclusão de curso. Primeiramente, agradecemos às famílias e amigos pelo constante apoio e encorajamento durante todo o processo.

Gostaríamos de agradecer especialmente à nossa orientadora, professora Dra. Sarah Negreiros de Carvalho Leite pela orientação dedicada e pelos valiosos conselhos que nos ajudaram a desenvolver nossa pesquisa e aperfeiçoar nossas habilidades acadêmicas.

Também agradecemos à Usiminas e a toda equipe técnica da Aciaria, em especial aos profissionais João Domingos e Beatriz Rocha que dedicaram seu tempo e esforço para responder às nossas perguntas e compartilhar suas experiências, contribuindo de forma significativa para o desenvolvimento deste trabalho.

Por fim, agradecemos a nossa instituição de ensino por nos proporcionar a oportunidade de realizar este trabalho de conclusão de curso e por nos equipar com as habilidades e conhecimentos necessários para enfrentar os desafios do mundo acadêmico e profissional.

Mais uma vez, obrigado a todos que contribuíram para este trabalho e por tornarem esta jornada educacional tão gratificante e inesquecível.

Resumo

O presente trabalho apresenta a aplicação de redes neurais artificiais (RNA) do tipo Multilayer Perceptron (MLP) ao processo de produção de aço líquido em convertedores do tipo LD, com o objetivo de elaborar dois modelos de predição, sendo o primeiro para predição do volume de oxigênio que deverá ser soprado e o peso de sínter que deverá ser adicionado no processo, e o segundo para predição da temperatura e carbono ao final do processo de produção de aço líquido. Para o desenvolvimento dos modelos de RNA, utilizou-se uma base de dados extraída do banco de dados da Aciaria 1 da Usiminas de Ipatinga com um total de 24.095 observações. Após tratamento da base, utilizou-se 6.482 corridas para o treinamento e validação dos modelos de RNA. A primeira MLP desenvolvida, para predição do volume de oxigênio e peso de sínter, possui oito variáveis de entrada e duas variáveis de saída, e a melhor topologia encontrada foi 8x97x50x20x2. A segunda MLP desenvolvida, para predição da temperatura e carbono fim de sopro, também empregou oito variáveis de entrada e duas variáveis de saída, sendo que destas, duas variáveis de entrada eram o volume de oxigênio e o peso sínter preditos pela primeira RNA. A melhor topologia encontrada para essa segunda rede MLP foi 8x72x60x110x2. Utilizou-se o MAE (*Mean Absolute Error*) como métrica para mensurar o erro do modelo e o MSE (*Mean Square Error*) como função de perda. Para realizar o teste dos modelos, utilizou-se uma nova base de dados, com um total de 93 corridas. Após a aplicação dos modelos de RNA na base de teste, obteve-se um acerto simultâneo da temperatura e carbono fim de sopro de 91%, contra 33% do resultado observado na área. O modelo de RNA obteve um MAE de 10 na variável de Temperatura Fim de Sopro, contra 24 para os valores observados na área. Com relação à variável de Carbono Fim de Sopro, o modelo de RNA obteve um MAE de 1 contra um MAE de 2,1 para os valores observados na área. Concluiu-se que, diante da melhora do acerto simultâneo de temperatura e carbono fim de sopro, os modelos de RNA desenvolvidos se mostraram ferramentas úteis para o ajuste do volume de oxigênio soprado e o peso de sínter adicionado ao processo de produção de aço líquido.

Abstract

This study focuses on the use of Multilayer Perceptron (MLP) Artificial Neural Networks (ANNs) to predict key parameters in the production process of liquid steel in LD converters. Specifically, two prediction models were developed: one for predicting the volume of oxygen and weight of sinters required, and the other for predicting the end-of-blow temperature and carbon levels. The ANN models were developed using a database of 24,095 observations from Aciaria 1/Usiminas in Ipatinga. After cleaning the database, 6,482 runs were used for training and validation of the ANN models. The first MLP developed to predict the volume of oxygen and weight of sinters required utilized eight inputs and two outputs, with the optimal topology being 8x97x50x20x2. The second MLP also utilized eight inputs and two outputs, with two input variables being the predicted volume of oxygen and weight of sinters from the first ANN. The best topology found for this second MLP was 8x72x60x110x2. The MAE (Mean Absolute Error) was used as a metric to measure the error of the model, and the MSE (Mean Square Error) was used as the loss function. To test the models, a new database of 93 runs was extracted from the same database. The ANN models were applied to this database, and a simultaneous accuracy of 91% was achieved for the end-of-blow temperature and carbon, compared to 33% for the observed result in the area. The MLP model achieved a MAE of 10 in the end-of-blow temperature variable, compared to 24 for the observed values in the area. For the end-of-blow carbon variable, the MLP model achieved a MAE of 1 compared to a MAE of 2.1 for the observed values in the area. In conclusion, the developed ANN models proved to be useful tools for adjusting the volume of blown oxygen and the weight of sinter added to the liquid steel production process. The improvement in the simultaneous accuracy of end-of-blow temperature and carbon suggests that these models have the potential to enhance the efficiency and accuracy of liquid steel production.

LISTA DE FIGURAS

Figura 1 - Fluxograma de processos de uma siderurgia.....	11
Figura 2 - Fluxograma das etapas do estudo.	14
Figura 3 - Fluxo simplificado da produção do aço.	18
Figura 4 - Convertedor Bessemer.....	19
Figura 5 - Forno Siemens Martin.	21
Figura 6 - Convertedor Thomas.	22
Figura 7 - Convertedor LD.....	23
Figura 8 - Etapas do processo de produção do aço em convertedores LD.....	26
Figura 9 - Modelo de um neurônio artificial.	31
Figura 10 - Rede neural artificial com uma camada intermediária.	35
Figura 11 - Neurônio artificial.	36
Figura 12 - Rede feedforward com uma camada oculta e uma camada de saída.	39
Figura 13 - Rede recorrente.....	40
Figura 14 - Fluxo de processamento do algoritmo de treinamento Backpropagation.	45
Figura 15 - Estrutura da RNA para predição do volume de oxigênio e peso de sínter.....	54
Figura 16 - Estrutura da RNA para predição da temperatura e carbono fim de sopro.	55
Figura 17 - Heatmap das variáveis do modelo de rede neural.	56
Figura 18 - Heatmap das variáveis da RNA de temperatura e carbono fim de sopro.	58
Figura 19 - Variáveis com dados na escala normal.	60
Figura 20 - Dados das variáveis na escala normalizada [-1;1].....	61
Figura 21 - Resultado do treinamento e validação da RNA de volume de oxigênio e peso de sínter, considerando apenas a saída Volume de Oxigênio.....	73
Figura 22 - Análise do valor real x predito da variável Volume de Oxigênio.	74
Figura 23 - Histograma do número de amostras da variável Volume de Oxigênio.....	75
Figura 24 - Resultado do treinamento e validação da RNA de volume de oxigênio e peso de sínter, considerando apenas a saída Peso de Sínter.	76
Figura 25 - Análise do valor real x predito da variável Peso de Sínter.....	77
Figura 26 - Histograma do número de amostras da variável Peso de Sínter.....	77
Figura 27 - Resultado do treinamento e validação da RNA de temperatura e carbono fim de sopro, considerando apenas a saída Temperatura Fim de Sopro.	81
Figura 28 - Análise do valor real x predito da variável Temperatura Fim de Sopro.	82
Figura 29 - Histograma do número de amostras da variável Temperatura Fim de Sopro.	82
Figura 30 - Resultado do treinamento e validação da RNA de temperatura e carbono fim de sopro, considerando apenas a variável de saída Carbono Fim de Sopro.....	83
Figura 31 - Histograma do número de amostras da variável Carbono Fim de Sopro.....	84
Figura 32 - Comparativo entre a taxa de acerto observada x taxa de acerto da RNA.	85
Figura 33 - Erro da variável Temperatura Fim de Sopro.	86
Figura 34 - Erro da variável Carbono Fim de Sopro.	86

LISTA DE TABELAS

Tabela 1 - Melhor configuração para as RNAs desenvolvidas.	67
Tabela 2 - Análise de sensibilidade da topologia da RNA de volume de oxigênio e peso de sínter.....	71
Tabela 3 - Análise de sensibilidade da topologia da RNA de temperatura e carbono fim de sopro.	79
Tabela 4 - Comparativo entre os valores médios da RNA x Observados x Objetivados.....	89
Tabela 5 - Comparativo entre os valores de MAE da RNA x Observados.	89

LISTA DE QUADROS

Quadro 1 - Dicionário das variáveis do modelo de rede neural.....	52
Quadro 2 - Regras de filtragem das variáveis.	53
Quadro 3 - Configuração dos parâmetros do Keras Tuner.	63
Quadro 4 - Faixa de acerto da variável Carbono Fim de Sopro.....	84
Quadro 5 - Comparativo entre os resultados da RNA x resultados observados.....	88

SUMÁRIO

1. Introdução	11
1.1 Objetivo geral	13
1.1.1 Objetivos específicos	13
1.2 Organização do Trabalho	15
2. Contextualização Teórica	16
2.1 Produção de aço	16
2.1.1 Etapas da produção	16
2.1.2 Tipos de siderurgia	17
2.1.3 Tipos de Convertedores	18
2.2 Processo de fabricação do aço no convertedor LD	23
2.2.1 Controle do processo	26
2.2.1.1 Modelo Estático	27
2.2.1.2 Modelo Dinâmico	28
2.3 Modelagem de Processos	29
2.4 Redes Neurais Artificiais	30
2.4.1 Surgimento e evolução das redes neurais artificiais	30
2.4.2 Inteligência artificial e definição de RNA	33
2.4.3 Estrutura de uma RNA	35
2.4.4 Arquitetura da RNA	37
2.4.5 Treinamento da RNA	40
2.4.5.1 Treinamento Supervisionado	41
2.4.5.2 Treinamento não supervisionado	42
2.4.5.3 Algoritmos de Treinamento	43
2.4.5.3.1 Algoritmo <i>Backpropagation</i>	44
2.4.6 Tarefas de aprendizagem	46
3 Metodologia	51
3.1 Instrumentos e formas de coleta	51
3.2 Análise dos Dados	52
3.3 Variáveis de entrada e saída	54
3.4 RNA para predição do volume de oxigênio e peso de sinter	55
3.5 RNA para predição da temperatura e carbono fim de sopro	57
3.6 Normalização dos Dados	59
3.7 Divisão da base de dados	61

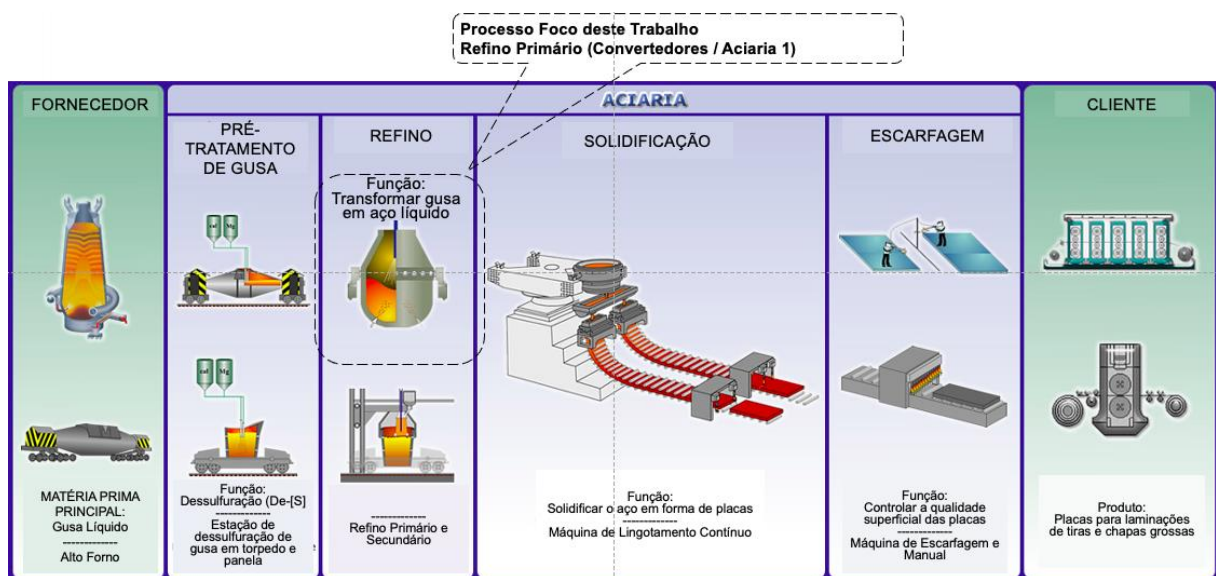
3.8	Topologia das RNAs	62
3.9	Otimizador e compilação dos modelos de RNAs	67
3.10	Treinamento e Validação das RNAs.....	68
3.10.1	Épocas de Treinamento	68
3.10.2	<i>Batch Size</i>	69
4	Resultados e Discussão	70
4.1	Resultados do treinamento e validação da RNA de volume de oxigênio e peso de sínter.....	70
4.1.1	Resultados da variável volume de oxigênio.....	73
4.1.2	Resultados da variável peso de sínter	75
4.2	Resultados do treinamento e validação da RNA de temperatura e carbono fim de sopra	78
4.2.1	Resultados da variável de temperatura fim de sopra.....	80
4.2.2	Resultados da variável carbono fim de sopra.....	83
4.3	Resultado consolidado e taxas de acerto da RNA.....	84
4.3.1	Comparação de valores médios e MAE da RNA x valores observados e objetivados ..	89
5	Conclusão	91
5.1	Sugestões de trabalhos futuros.....	92
	REFERÊNCIAS	93
	ANEXO	98

1. Introdução

Em uma usina siderúrgica integrada, o aço líquido é produzido nas aciarias em um equipamento chamado convertedor. A aciaria recebe o ferro gusa produzido nos altos fornos, e esse ferro gusa é dessulfurado (processo que consiste em reduzir o teor de enxofre) para então ser carregado no convertedor. Juntamente com o ferro gusa, são adicionadas sucatas de aço no convertedor, que compõem a carga metálica carregada no equipamento. Após os carregamentos de sucata e gusa, é iniciado o processo de sopro de oxigênio, que tem como principal objetivo promover a descarburização (processo que consiste em reduzir o teor de carbono) da carga metálica e conseqüentemente, um aumento de temperatura por se tratar de uma reação exotérmica.

Em resumo, o principal papel do convertedor no processo de produção do aço líquido é garantir que, ao final do processo, a temperatura e carbono estejam dentro da faixa desejável para prosseguir para os processos posteriores (refino secundário e lingotamento). Na Figura 1 é representado o fluxograma de processo de uma Siderurgia, dando destaque ao Refino Primário (Aciaria), foco deste trabalho.

Figura 1 - Fluxograma de processos de uma siderurgia.



Fonte: Autor (2023).

Para o controle da temperatura e teor de carbono ao final do sopro de oxigênio é necessário definir basicamente dois parâmetros antes de iniciar o sopro:

- I. Volume de oxigênio que será soprado;
- II. Quantidade de material refrigerante (normalmente síter) que será adicionado ao processo produtivo.

Existem modelos termodinâmicos que recebem as informações dos processos anteriores e fazem um cálculo para definir estes dois parâmetros, com o objetivo de alcançar um valor determinado de temperatura e carbono ao final do sopro. Entretanto, devido a diversos fatores que causam variabilidade nos processos anteriores, erros e/ou falta de medição, podem ocorrer desvios nesse cálculo, de modo que ao final da etapa de sopro, a temperatura e carbono estejam fora da faixa desejada.

O operador do convertedor só consegue saber o resultado da temperatura e carbono ao final do sopro, após retirar uma amostra do aço líquido e enviá-la para o laboratório de análise química. O tempo estimado para ter o resultado da amostra é de aproximadamente 10 a 13 minutos. Durante esse tempo, o aço líquido fica parado dentro do convertedor ocasionando desgaste nos tijolos refratários e perda de produtividade no processo, uma vez que o processo não pode prosseguir enquanto o operador não tiver o resultado da amostra. Após receber o resultado da amostra, é verificado se a temperatura e carbono ficaram dentro dos valores desejados para prosseguir o processo, e caso não estejam, é necessário reiniciar o sopro de oxigênio, processo denominado resopro (retrabalho) para que o aço alcance a composição química desejada.

Uma forma de reduzir a frequência desses dois problemas (ressopro e perda de produtividade) é utilizar modelos mais eficientes que possam ser mais assertivos no cálculo do volume de oxigênio e peso de síter, e a utilização de redes neurais artificiais (RNA) pode ser uma alternativa a esses modelos termodinâmicos, devido ao seu poder de correlacionar diversas variáveis para prever uma ou mais variáveis.

Segundo Chu (2001), o processo de sopro de oxigênio em convertedores caminha para melhorias, associando novas técnicas de modelagem, como redes neurais e de acompanhamento com medições contínuas utilizando microcâmeras na lança de oxigênio.

A utilização de RNA na modelagem de processos não-lineares que possuem características complexas e/ou desconhecidas é bastante favorável se comparado com os métodos

determinísticos tradicionais. Elas possuem processadores, denominados neurônios, que possuem capacidade de armazenar o conhecimento a partir de treinamentos e de se adaptarem, conforme alterações no ambiente em que estão sendo utilizadas (Haykin, 2001).

1.1 Objetivo geral

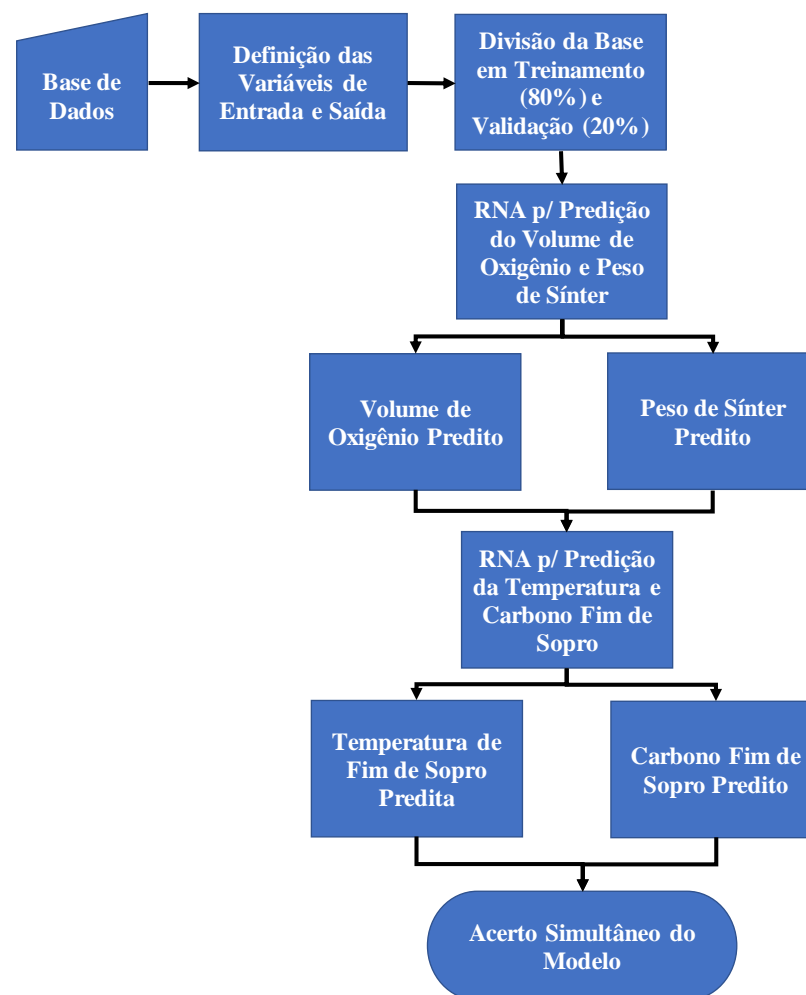
O objetivo geral deste trabalho é desenvolver um modelo empregando redes neurais para prever o volume de oxigênio que será soprado e o peso de sínter que será adicionado no processo de produção do aço, de modo a se garantir a temperatura e o teor de carbono visados ao final do processo de sopro de oxigênio no convertedor.

1.1.1 Objetivos específicos

Para cumprimento do objetivo geral é necessário atender aos seguintes objetivos específicos:

- Identificar as variáveis significativas, a partir de análises estatísticas, que influenciam no peso de sínter, volume de oxigênio, temperatura e carbono de fim de sopro do aço líquido;
- Desenvolver uma RNA para predição do peso de sínter que será adicionado na corrida e do volume de oxigênio que será soprado;
- Desenvolver uma RNA para predição da temperatura e teor de carbono ao fim da etapa de sopro, utilizando como entradas, o peso de sínter e o volume de oxigênio preditos pela RNA desenvolvida no objetivo anterior, de forma a validar o acerto das RNAs.

A Figura 2 apresenta um fluxograma com a ideia proposta neste estudo.

Figura 2 - Fluxograma das etapas do estudo.

Fonte: Autor (2023).

1.2 Organização do Trabalho

No Capítulo 1 é apresentada uma introdução sobre o tema, conceituando a área de aplicação do trabalho, o problema de pesquisa abordado, a justificativa do trabalho e os objetivos. No Capítulo 2 tem-se uma revisão bibliográfica sobre o tema do trabalho, explicitando o histórico e o processo de produção do aço, abordando conceitos de outros autores e explorando as ferramentas relacionadas ao problema em estudo. No Capítulo 3 é apresentada a metodologia do trabalho, indicado o método de pesquisa utilizado, os passos percorridos ao longo da pesquisa, explicitando os parâmetros e técnicas utilizadas para alcançar os objetivos traçados e os parâmetros utilizados para o desenvolvimento dos modelos de predição. No Capítulo 4 são apresentados e discutidos os resultados obtidos. Também foi realizada uma comparação dos resultados deste trabalho com os resultados praticados no sistema atual da empresa. Por fim, no Capítulo 5 é realizada a conclusão do trabalho e sugestões de trabalhos futuros sobre o tema abordado.

2. Contextualização Teórica

Neste Capítulo são apresentados os principais conceitos para entendimento do processo de produção de aço nos Convertedores LD e de Redes Neurais Artificiais utilizados neste estudo.

2.1 Produção de aço

O aço é, dentre os materiais estruturais, um dos principais no mundo moderno. Trata-se de uma liga de ferro e carbono produzida na Siderurgia.

“A fronteira entre o ferro e o aço foi definida na Revolução Industrial, com a invenção de fornos que permitiam não só corrigir as impurezas do ferro, como adicionar-lhes propriedades como resistência ao desgaste, ao impacto, à corrosão etc. Por causa dessas propriedades e do seu baixo custo o aço passou a representar cerca de 90% de todos os metais consumidos pela civilização industrial” (Instituto Aço Brasil - IAB).

2.1.1 Etapas da produção

No processo siderúrgico, o aço é fabricado a partir do minério de ferro e do carvão. Conforme o Instituto Aço Brasil (IAB), a fabricação do aço pode ser dividida em quatro etapas:

- **Preparação da carga:** nesta etapa, o minério de ferro e o carvão são preparados para serem utilizados no Alto Forno. O carvão é destilado na Coqueria, obtendo-se o coque. E o minério de ferro, em sua maior parte, é aglomerado utilizando-se cal e finos de coque, resultando no sinter.
- **Redução:** as matérias-primas preparadas na etapa anterior são carregadas no Alto Forno, onde são expostas ao oxigênio aquecido a uma temperatura de 1000° C, soprado pela parte de baixo deste equipamento. Em contato com o oxigênio, o carvão produz calor suficiente para fundir a carga metálica, reduzindo o minério de ferro em um metal

líquido (ferro gusa). O ferro gusa trata-se de uma liga de ferro e carbono com elevado teor de carbono.

- **Refino:** é nesta etapa que o ferro gusa (líquido ou sólido) transforma-se em aço líquido. Os convertedores (a oxigênio – utilizam ferro gusa líquido, ou elétricos – utilizam ferro gusa sólido), principais equipamentos do Refino Primário das Aciarias, são responsáveis por esta transformação, removendo boa parte do carbono juntamente com impurezas. O aço líquido então é solidificado em equipamentos de lingotamento contínuo, produzindo semiacabados (lingotes e blocos).
- **Laminação:** os semiacabados produzidos na etapa anterior (lingotes e blocos) são então processados mecanicamente, através dos Laminadores, transformando-se em diversos produtos siderúrgicos. Estes são denominados de acordo com sua forma e/ou composição química.

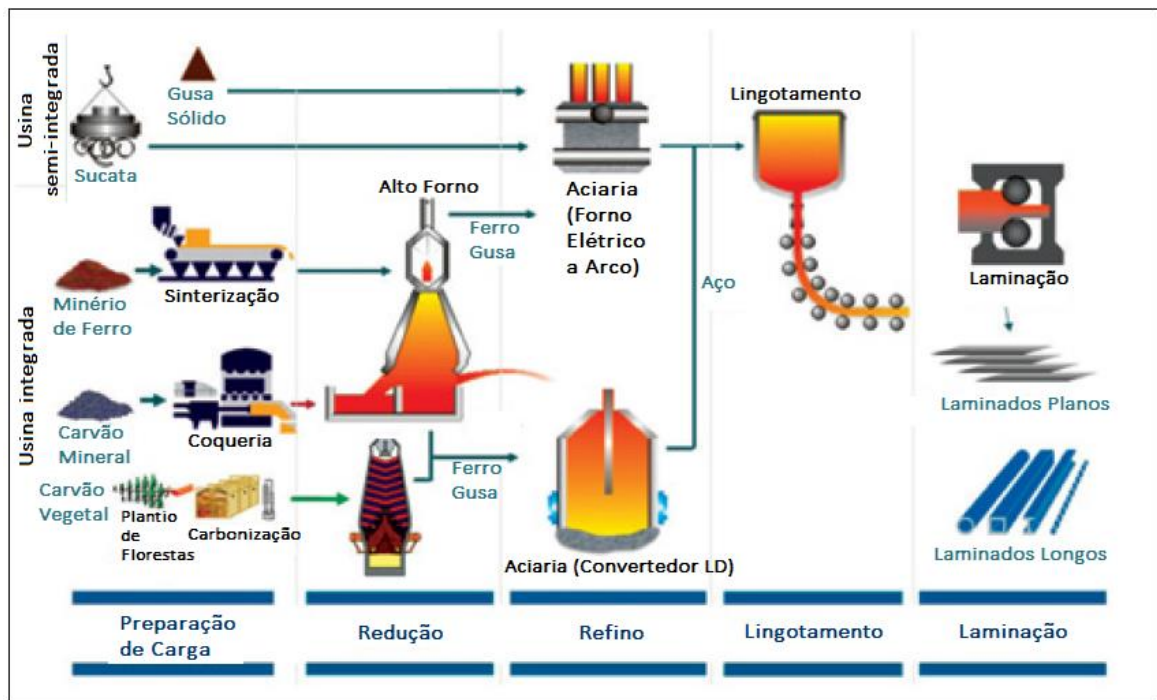
2.1.2 Tipos de siderurgia

Existem basicamente duas rotas tecnológicas para produção de aço nas usinas siderúrgicas (CNI, 2017):

- **Usinas Integradas:** o minério de ferro é reduzido nos Altos Fornos, utilizando o coque (produzido nas Coquerias) obtido a partir do carvão mineral ou do carvão vegetal, produzindo o ferro gusa. O ferro gusa é então refinado em um forno a oxigênio ou elétrico (chamado de Convertedor) na Aciaria, transformando-se em aço líquido. Em uma próxima etapa, o aço líquido é transformado em placas (no lingotamento), seguindo para o processo de laminação.
- **Usinas Semi-integradas:** o processo deste tipo de rota de produção de aço inicia-se diretamente no refino, realizado em fornos elétricos e utilizando como matéria-prima a sucata e o ferro gusa sólido. A partir daí, o aço segue para as etapas de lingotamento e laminação.

A Figura 3 apresenta o fluxo simplificado de produção em usinas integradas e semi-integradas.

Figura 3 - Fluxo simplificado da produção do aço.



Fonte: Adaptado do Instituto Aço Brasil (2012).

A Usiminas é uma usina siderúrgica do tipo integrada, contendo, portanto, as etapas de produção anteriores ao refino, além da própria etapa de refino e da etapa de laminação.

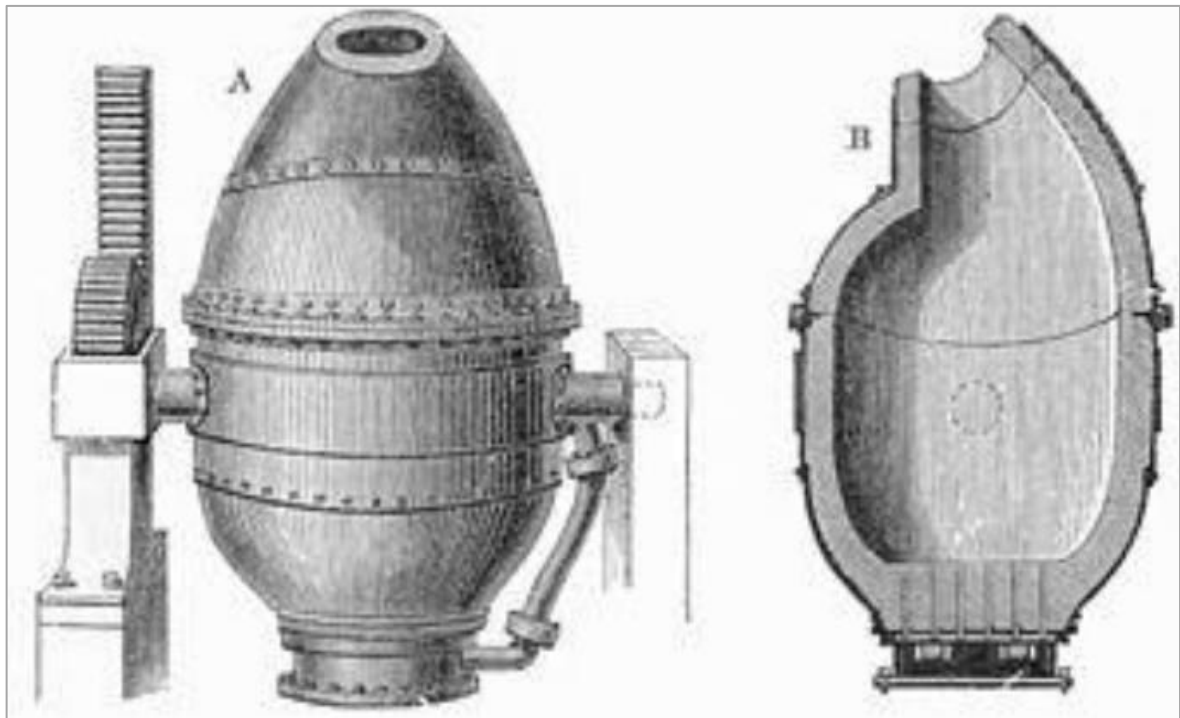
2.1.3 Tipos de Convertedores

Na usina siderúrgica integrada, o refino primário é responsável pela transformação do ferro gusa em aço. Para transformar o ferro gusa em aço, é necessário que ele passe por um processo de oxidação, onde o ferro e as impurezas se combinam com o oxigênio, até que as concentrações de carbono e das impurezas se reduzam aos valores desejados. Este processo ocorre no Convertedor a oxigênio, principal equipamento do refino primário. A seguir são citados os principais tipos de convertedores a oxigênio para produção de aço ao longo da história.

Em 1856, Henry Bessemer criou um processo de produção do aço a partir da remoção de impurezas e carbono do ferro pela oxidação com ar soprado através do ferro gusa líquido. O denominado Convertedor Bessemer, em homenagem ao seu inventor, foi o primeiro construído para a fabricação de aços em larga escala.

O Convertedor Bessemer tratava-se de um recipiente grande de aço de formato ovoide, revestido com argila, com capacidade de 8 a 30 toneladas de ferro fundido. No topo possuía uma abertura inclinada, pela qual o ferro era introduzido e o produto final removido. Enquanto, no fundo existiam perfurações, denominadas *tuyeres* (ou ventaneiras), pelas quais o ar era forçado para dentro do convertedor. Com o auxílio de munhões instalados em seu corpo, o convertedor era rotacionado para receber a carga, girado à posição normal durante o processo de sopragem do ar e rotacionado no outro sentido para descarregar o aço fundido ao final do processo (Dutra, 2017). A Figura 4 ilustra um exemplo do convertedor Bessemer.

Figura 4 - Convertedor Bessemer.



Fonte: Dutra (2017)

O convertedor Bessemer não necessitava de combustível. As temperaturas altas eram mantidas devido às reações químicas que ocorriam dentro do convertedor.

O processo de conversão Bessemer (chamado de sopro) era muito rápido, completando-se em aproximadamente 20 minutos, e seu método reduziu significativamente os custos de produção do aço.

O convertedor Bessemer foi substituído posteriormente, à medida que houve a necessidade de um controle melhor na composição química final do aço. O processo de Bessemer era tão

rápido, que não permitia analisar quimicamente e ajustar os elementos ligantes no aço produzido. Além disso, fatores como a não remoção do fósforo (necessitando utilizar ferro gusa com baixo teor de fósforo, cada vez mais caro ao longo dos anos) e a capacidade de receber uma quantidade limitada de sucata (material cada vez mais barato), fez com que os custos para produção de aço em convertedores Bessemer aumentassem. A utilização de ar (em vez de oxigênio puro) também contribuiu para a substituição deste tipo de convertedor, uma vez que alguns tipos de aços são sensíveis ao nitrogênio (presente no ar soprado).

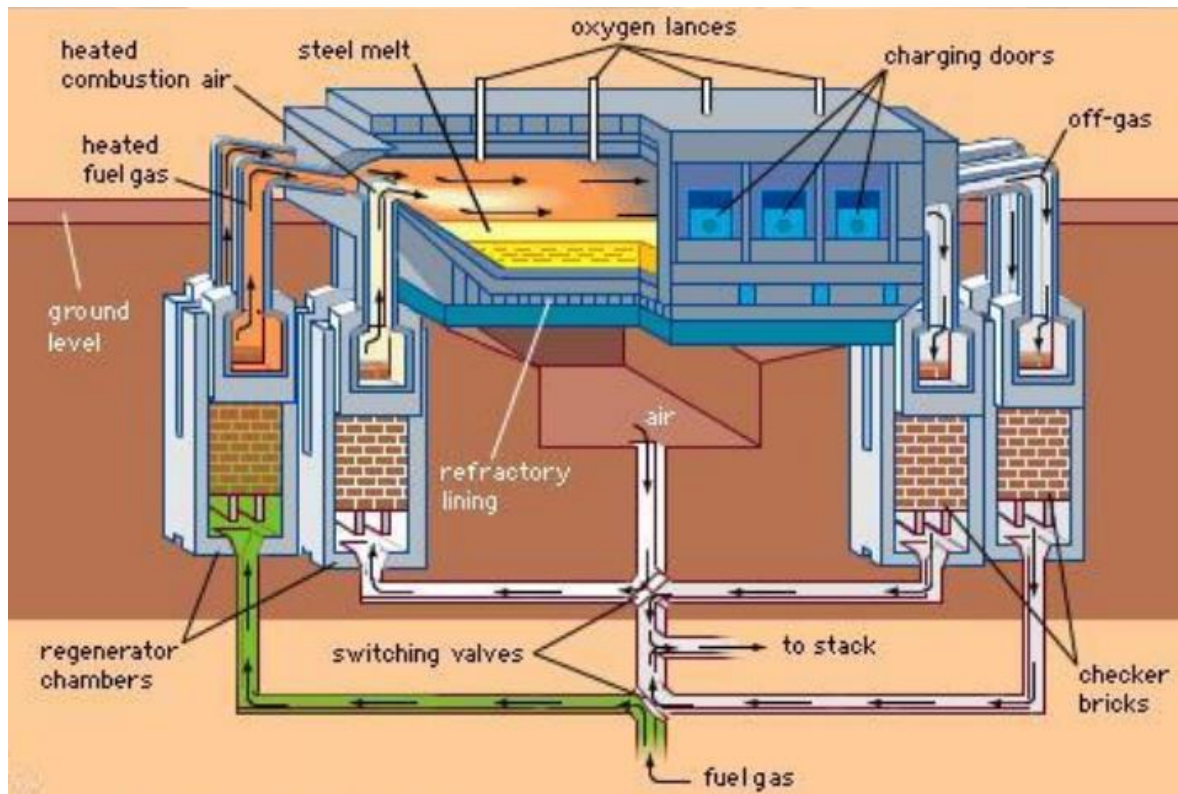
Por volta de 1865, surgiu o Forno Siemens-Martin, também designado forno de soleira aberta, e consistia num processo para a obtenção de aço, idealizado pelo metalurgista francês Pierre Martin e desenvolvido pelo engenheiro e físico Wilhelm Siemens. Martin e Siemens realizaram uma adaptação de um tipo de forno regenerativo a gás, inventado pelo irmão de Wilhelm, o também engenheiro Friedrich Siemens, que era utilizado na fabricação do vidro (Porto Editora, 2022). Na época, o processo Bessemer era encarado com certa desconfiança, o que fez com que o forno Siemens-Martin fosse amplamente utilizado na Grã-Bretanha como base da indústria do aço, em rápida expansão naquele país.

O forno Siemens-Martin utilizava combustíveis, como óleo e gás, para elevar a temperatura do ferro gusa líquido, ocasionando a liberação do oxigênio (através da redução dos óxidos de ferro presentes na sucata) e, conseqüentemente, a oxidação das impurezas. A Figura 5 ilustra um exemplo do forno Siemens Martin.

O processo de fabricação do aço através do forno Siemens-Martin era extremamente demorado, durava em média de 6 a 8 horas, e necessitava utilizar uma enorme quantidade de sucata.

O forno Siemens-Martin foi utilizado até a década de 60 como principal equipamento para transformação do ferro gusa em aço. Ele foi gradativamente sendo substituído (até sua eliminação total), à medida que o custo de energia elétrica foi reduzindo em relação ao aquecimento a gás ou óleo.

Figura 5 - Forno Siemens Martin.

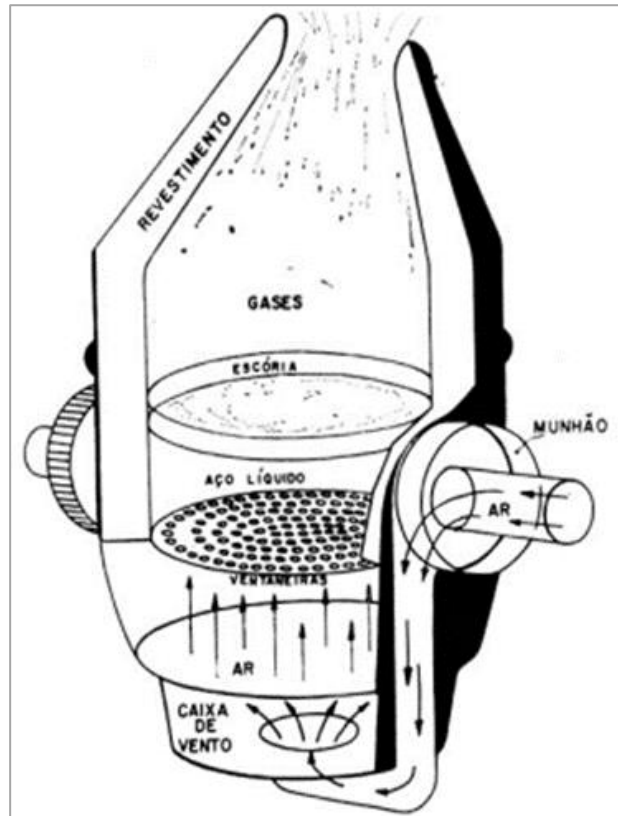


Fonte: Dutra (2017).

Em 1978, Sidney Gilchrist Thomas escreveu em seu artigo “*Elimination of phosphorus in the Bessemer converter*” sua descoberta (feita anos antes, em 1975), que melhorava o funcionamento do convertedor Bessemer. O convertedor Thomas tinha, fisicamente, as mesmas características do convertedor Bessemer. O que o diferenciava era o revestimento interno, permitindo alterar algumas características de processo (Dutra, 2017).

O revestimento, agora feito de dolomita, resistia ao ataque da escória, permitindo assim, trabalhar com ferro gusa com alto teor de fósforo. Esta melhoria fez com que as reações químicas no convertedor Thomas permitissem a remoção do fósforo (através da utilização de cal, a qual é possível porque o revestimento é feito de material refratário de natureza básica) e de enxofre (Dutra, 2017). A Figura 6 apresenta o convertedor Thomas.

Figura 6 - Convertedor Thomas.



Fonte: Dutra (2017).

O convertedor LD, cuja concepção é a mais moderna, foi idealizado em 1948 por Robert Durrer. O nome LD deve-se ao fato de que plantas em escala piloto foram testadas, em 1952, nas cidades austríacas de Linz e Donawitz. Neste mesmo ano, a empresa Voest de Linz passou a comercializar este tipo de convertedor (Moreira e Lana, 2002).

Embora o princípio deste convertedor continue sendo o de injeção de oxigênio, ele diferencia-se pela sopragem de oxigênio puro (e não mais ar) sob pressão na superfície do gusa líquido. O método de sopro também se alterou, ocorrendo agora pelo topo e não mais pela parte inferior do convertedor. Na zona de impacto entre o oxigênio e o metal líquido, pode-se chegar a temperaturas de 2500 a 3500°C. A Figura 7 apresenta um exemplo do Convertedor do tipo LD.

Figura 7 - Convertedor LD.



Fonte: Adaptado de Maia et al. (2018).

O convertedor LD permite o controle adequado da composição química final do aço, por ter um processo mais lento que o dos convertedores Bessemer e Thomas, mas infinitamente mais ágil do que o forno Siemens-Martin. Ele também permite a redução adequada de impurezas e do carbono.

Cerca de 80% da produção de aço no Brasil é feita através de convertedores LD, que apresenta alta produtividade, custos adequados e grande flexibilidade metalúrgica (Chaves, 2006). Os 20% restantes basicamente são produzidos em fornos elétricos a arco (FEA), que se diferencia do convertedor LD pela fonte de energia e pela composição da carga metálica e não metálica.

A Usiminas possui na Aciaria 1 três convertedores do tipo LD.

2.2 Processo de fabricação do aço no convertedor LD

No convertedor LD, a fabricação do aço consiste no processo de refino primário deste, onde as impurezas do ferro gusa (carga metálica líquida, produzida nos Altos Fornos) e,

eventualmente, da sucata (carga metálica sólida, parcialmente gerada por outros processos dentro da usina e complementada por fornecimento externo) são oxidadas, através do sopro de oxigênio por uma lança supersônica (Araújo, 2009; Barão et al.,2011). Estas impurezas oxidadas então incorporam-se à escória, que é formada pela carga não metálica (fundentes e escorificantes) adicionada durante o processo de sopro dentro dos convertedores (Araújo, 2009; Barão et al.,2011). Como consequência do processo no convertedor LD, são gerados gases, que são captados e tratados para utilização em outros processos dentro da usina.

Na oxidação ocorrida dentro dos convertedores, o oxigênio combina-se com elementos, tais como carbono, manganês, silício e fósforo (e acaba se combinando com parte do ferro), formando óxidos, compondo a escória ou formando gases. O elemento enxofre, por sua vez, é removido através da combinação da basicidade da escória com a temperatura atingida no processo (Cunha, 2001).

Na descarburização, a mais importante dentre as reações que ocorrem dentro do convertedor (pois controla a taxa de produção), o oxigênio e o carbono (dissolvidos no meio) e, o carbono do banho e o FeO, reagem, formando em ambas as reações o monóxido de carbono (Cunha, 2001; Herskovic, 1976).

Conforme Borges (2016), as etapas do processo de fabricação do aço no convertedor LD são descritas abaixo. Estas etapas compõem o ciclo de produção, também chamado de corrida, cujo tempo é em torno de 38 a 40 minutos.

Etapa 1 – Carregamento de carga metálica: nesta etapa o convertedor é rotacionado a um ângulo de 45° em relação à vertical e é carregada a carga metálica sólida, que é basicamente a sucata (de ferro, mista – ferro com revestimento de zinco ou estanho, sucata oriunda de corridas anteriores). As dimensões da sucata devem ser tais que não causem danos ao revestimento do convertedor durante o carregamento e permitam a completa fusão durante o sopro. A fim de evitar risco de explosão, a sucata deve estar completamente seca.

Com o convertedor ainda rotacionado a 45°, é carregada a carga metálica líquida, o ferro gusa líquido (geralmente composto por 94,5% de ferro e 4,3% de carbono e, impurezas como fósforo, silício, manganês e enxofre). A proporção de gusa líquido carregado depende de alguns fatores, tais como sua composição e temperatura, qualidade do aço a ser produzido, volume das adições de cal, síter e carepa e, em parte das dimensões do próprio convertedor. Normalmente esta proporção fica em torno de 70 a 85% da carga metálica.

Etapa 2 – Sopros: terminada a etapa de carregamento da carga metálica, o convertedor retorna para a posição vertical e então ocorre a descida da lança de oxigênio e o sopro (responsável pela oxidação de impurezas e carbono). É utilizado um padrão de altura da lança em relação ao banho metálico durante o sopro, denominado “distância lança-banho”, que é determinado de acordo com os diferentes passos do sopro. O sopro se desenvolve basicamente em quatro passos: ignição, formação da escória, descarburagem e oxidação para acerto da temperatura.

Ainda nesta etapa, é realizada a adição de fundentes. Dentre os fundentes, podemos citar a cal, necessária para a escorificação da sílica formada pela oxidação do silício da carga metálica e, para a remoção do fósforo e enxofre. Deve-se também manter na escória uma quantidade de CaO suficiente para diminuir o ataque dos refratários. Outro fundente de cal, CaF₂ - fluorita, é adicionado a fim de aumentar a fluidez de escórias muito viscosas. Dentre os fundentes, pode-se citar também os aluminatos (especialmente bauxita) e os boratos.

É realizada a amostragem do banho e a medição de temperatura e do percentual de carbono (através de uma inferência da temperatura de solidificação da amostra). Caso a temperatura do banho esteja acima da adequada, realiza-se o resopro de oxigênio. E caso ela esteja abaixo da prevista, é adicionado material refrigerante (normalmente sinter) para o resfriamento do banho até a temperatura desejada.

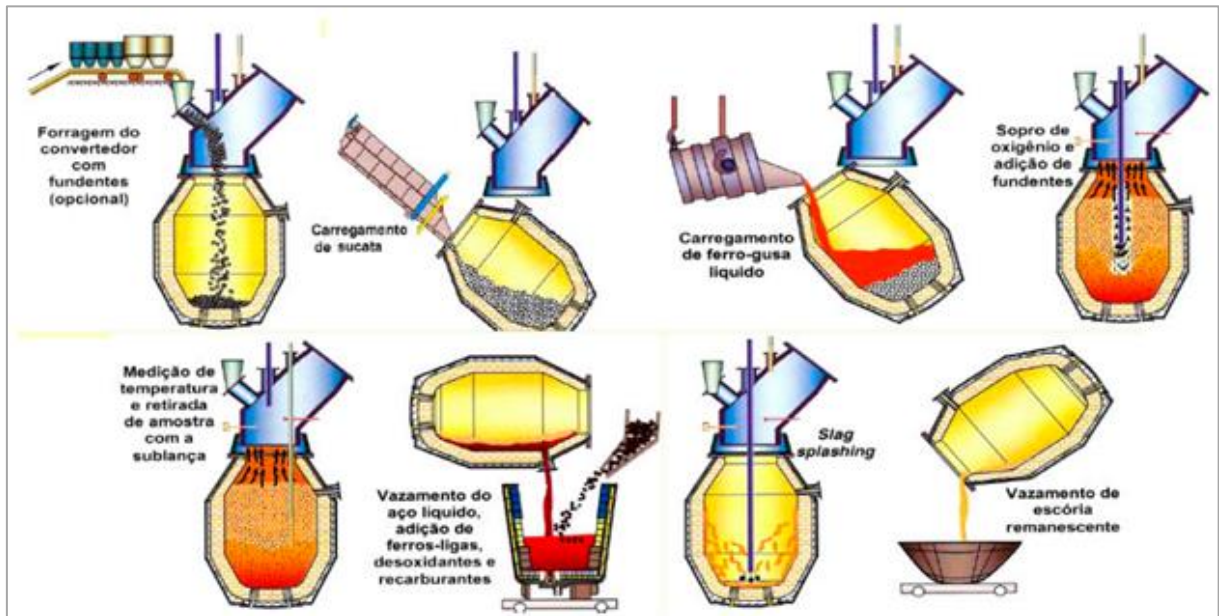
Etapa 3 – Vazamento do aço: o convertedor é rotacionado para realização do vazamento do aço para a panela, momento no qual são adicionados materiais desoxidantes e ligas, de forma a atingir a qualidade desejada para o aço. Dentre os desoxidantes possíveis, pode-se citar o alumínio e o silício. O tempo de vazamento é controlado de forma a minimizar a passagem de escória para a panela, evitar oxidação excessiva do metal pelo ar, bem como perdas significativas de temperatura, garantindo assim uma produtividade adequada.

Após o vazamento do aço, pode-se realizar o *slag splashing* (processo opcional), que se trata de uma forma de reparo do revestimento refratário do convertedor. Neste procedimento, é feita a adição de determinados materiais à escória e realizado um sopro (normalmente de nitrogênio), visando aplicar a escória ao revestimento. A escória remanescente que não aderir ao revestimento é depois retirada através do vazamento de escória.

Caso não exista a etapa do *slag splashing*, após o vazamento do aço, vai-se diretamente para a etapa de vazamento de escória, rotacionando o convertedor para o lado contrário ao de vazamento de aço. O convertedor retorna, então, para o ângulo de 45° em relação à vertical para receber novo carregamento de carga metálica. A Figura 8 apresenta as etapas do processo

de produção do aço no Convertedor LD:

Figura 8 - Etapas do processo de produção do aço em convertedores LD.



Fonte: Silva (2013).

2.2.1 Controle do processo

O controle do processo no convertedor LD tem a finalidade de obter, ao final do sopro, o peso de aço visado, na temperatura e com a composição química previstas. Um dos parâmetros possíveis para avaliar os resultados operacionais de um convertedor LD é justamente o acerto destas características (Herskovic, 1976). Este acerto está diretamente ligado à produtividade, pois o resopro devido ao alto carbono ou à baixa temperatura, além de prejudicarem a qualidade do aço, resultam num maior tempo de processamento. Para a realização do controle do processo, são utilizadas várias técnicas, com distintos graus de refinamento, de acordo com as condições locais e técnico-econômicas.

Dentre os tipos de controle de processo, o mais simples, denominado modelo manual, baseia-se na experiência do soprador para a condução de operação e na observação visual da chama para avaliação do teor de carbono e da temperatura, definindo as medidas a serem adotadas para a interrupção do sopro no momento adequado.

Entretanto, geralmente utiliza-se um modelo matemático, que abrange um determinado número de parâmetros e fornece ao soprador subsídios (mais ou menos completos, de acordo

com a complexidade do modelo matemático) para a condução do sopro. A partir do modelo de controle do processo são definidas as matérias-primas, os insumos e as condições de operação do convertedor. Classificam-se os modelos matemáticos em estáticos, quando os cálculos são realizados a partir de dados levantados do início do sopro, e dinâmicos, quando utilizam informações colhidas durante o sopro e determinam ações corretivas.

Os modelos estáticos baseiam-se em balanços de massa e de energia, termodinâmica e análise estatística de dados operacionais e experimentais. Enquanto os modelos dinâmicos, além dos conhecimentos utilizados nos modelos estáticos, baseiam-se em equações diferenciais de variação da composição química e da temperatura ao longo do tempo (Cunha, 2001).

2.2.1.1 Modelo Estático

Os modelos estáticos de controle de processo baseiam-se em balanços de massa e de energia, acrescidos de relações estatísticas e procedimentos operacionais (Cunha, 2001). Segundo Cunha (2001), existem alguns fatores que limitam o acerto na previsão dos modelos estáticos de controle de processo em convertedores LD, tais como a oscilação química e térmica do gusa e, química e granulométrica das matérias primas sólidas, as limitações de medidas, a diferenciação na atuação humana (cuja influência varia de acordo com o grau de padronização das operações da empresa), as simplificações teóricas dos modelos e procedimentos ainda não equalizados (como tempo entre corridas e de transporte de gusa).

O balanço de massa consiste em determinar as massas de cada elemento (como C, Mn, Si, P, S, O etc.) no início e no final do processo que está sendo considerado. Deste modo, consegue-se calcular a quantidade de oxigênio necessária para a oxidação dos elementos presentes no ferro gusa e a quantidade de oxigênio que ficará dissolvida no banho. Dessas quantidades definidas de oxigênio, deverá ser subtraído o aporte de oxigênio proveniente do sinter ou minério de ferro que será adicionado no final do processo.

O balanço de energia, de acordo com a lei de conservação de energia, é definido pela seguinte equação, onde o total de entalpia que entra no sistema é igual ao total de entalpia que sai do sistema menos as perdas: $\Delta H_{Entrada} = \Delta Saida + \Delta Perdas$. Para calcular o modelo de conservação de energia todas as massas são conhecidas com exceção da massa de sinter (ou minério de ferro). A quantidade calculada de sinter será a necessária para igualar a entalpia de entrada com a entalpia de saída do sistema mais as perdas.

O modelo utilizado para controle do processo dos convertedores LD da Aciaria 1 da Usiminas é do tipo estático. Existem variáveis que influenciam no processo dos convertedores LD da Usiminas, sendo algumas inerentes às características dos equipamentos, outras aos procedimentos operacionais existentes e outras às variações das matérias primas e insumos.

Dentre estas variáveis, pode-se citar: conversor utilizado para produzir o aço (possível erro devido aos diferentes perfis de revestimento refratário), tempo de transporte do ferro gusa até o convertedor (influenciando na perda térmica relacionada à temperatura interna do convertedor), peso de gusa, temperatura do ferro gusa, peso de sucata, composição química do ferro gusa etc.

2.2.1.2 Modelo Dinâmico

Os modelos dinâmicos de controle de processo utilizam os cálculos dos modelos estáticos, acrescidos de dados de sensoriamento, que possibilitam ações corretivas durante o sopro (Cunha, 2001). O acompanhamento da velocidade de descarburização é primordial e pode ser verificada através da análise dos gases gerados, por métodos audiométricos ou pela utilização de sub-lanças.

Segundo Cunha (2001), o controle dinâmico pode ser realizado através dos seguintes sistemas:

Método da sublança: mede-se a temperatura do banho através de um sensor e calcula-se o percentual de carbono de acordo com a temperatura de solidificação do aço;

Método da análise de gás de exaustão: as porcentagens de monóxido e dióxido de carbono são medidas. Deste modo, é conhecida a porcentagem de carbono removida do aço;

Bomb-method: mede-se a temperatura do banho e a temperatura de solidificação do aço.

O modelo de controle dinâmico mais difundido, desenvolvido por Takemura *et al.* (1978), baseia-se na tese de que o processo de refino é altamente dinâmico e, através de sublança, obtém as condições operacionais necessárias para alimentar o modelo. Takemura *et al.* (1978) dividem o refino em duas etapas, chamadas de descarburização direta e indireta.

Na descarburização direta, o refino do banho se dá através de reações químicas, controladas pelas condições iniciais, entre os principais elementos do banho, C, Si e P, e o oxigênio. A descarburização indireta, influenciada por determinados fatores (tais como as condições de

escorificação, as reações intrínsecas, as condições de agitação e as reações dos fundentes), caracteriza-se pelas reações entre o carbono e o oxigênio e, entre o ferro e o oxigênio.

O modelo matemático para o controle dinâmico leva em consideração a taxa de descarburização (reações de descarburização direta e indireta) e a taxa de aumento de temperatura (balanço de energia no convertedor durante o tempo de medida na sublança até o *blow off*) (Cunha, 2001).

2.3 Modelagem de Processos

“A modelagem de situações reais é necessária por poder ser objeto de utilização em ensaios e experimentos de uma forma mais fácil, e com menores custos, do que a utilização dos sistemas reais” (Shaw e Simões, 1999). A modelagem de situações reais pode se basear em três tipos de métodos (Shaw e Simões, 1999):

Método experimental: representa em gráficos os pontos discretos de uma curva característica de entradas e saídas obtidas experimentalmente. É um método mais difícil de ser aplicado, pois geralmente os equipamentos dos quais se constitui o processo não estão disponíveis para experimentos. Assim, não se consegue realizar experimentos para verificar como o processo reage às diversas entradas, não permitindo o projeto de um sistema de controle adequado. Além da indisponibilidade dos equipamentos, o alto custo e a necessidade de avaliação das faixas de valores de entrada e saída cobertas pelos instrumentos de medição são outros limitadores para a utilização do método experimental. De maneira a reduzir a dependência dos instrumentos de medição, pode-se prever os possíveis valores de entrada, sendo necessário para tal uma descrição de como o processo reage de acordo com as várias entradas (descrição esta realizada pelos modelos matemáticos);

Método de modelagem matemática: obtém-se um modelo idealizado do processo a ser controlado, através de equações diferenciais ou algébricas. Aplicam-se certas restrições de forma a simplificar o modelo, tais como assumir que o processo seja linear (variações na entrada produzem variações proporcionais na saída). Desta forma, difícil desenvolver uma descrição matemática realista do processo, uma vez que ao simplificá-lo pode-se ignorar informações importantes e necessitar de reajustes pelo operador;

Método heurístico: realiza tarefas, com base em experiência prévia, utilizando regras práticas (implicações lógicas) e estratégias frequentemente aplicadas. As “regras associam conclusões (ou consequências) com condições (ou antecedentes)” (Shaw e Simões, 1999). Semelhante ao

método experimental, o método heurístico também constrói uma tabela de entradas e saídas (onde a função entrada-saída é descrita ponto-a-ponto), porém não se restringe à linearidade. Através do método heurístico, pode-se construir uma função entrada versus saída, não matemática, porém útil na descrição do processo. “Isso pode ser feito sempre que um modelo matemático for muito difícil ou complexo de se obter” (Shaw e Simões, 1999);

A aplicação de metodologias não-lineares para controle de processos tem sido cada vez mais requisitada, embora em muitos sistemas não-lineares seja extremamente difícil e, conseqüentemente, caro, obter um modelo preciso do processo baseado em princípios anteriores (Hussain, 1999). Uma nova técnica de modelagem, a rede neural artificial, está em evidência, utilizando com habilidade um caminho indefinido para relacionar dados de entrada com dados de saída. Trata-se de uma técnica estatística não-linear utilizada para solucionar problemas não adequados aos métodos estatísticos convencionais (Malinova *et al.*, 2001). A rede neural artificial é bastante útil na modelagem de processos em que não se tem o entendimento completo físico/estatístico do problema, existem variações estatísticas nos dados observados ou um mecanismo não linear é responsável pela geração dos dados.

2.4 Redes Neurais Artificiais

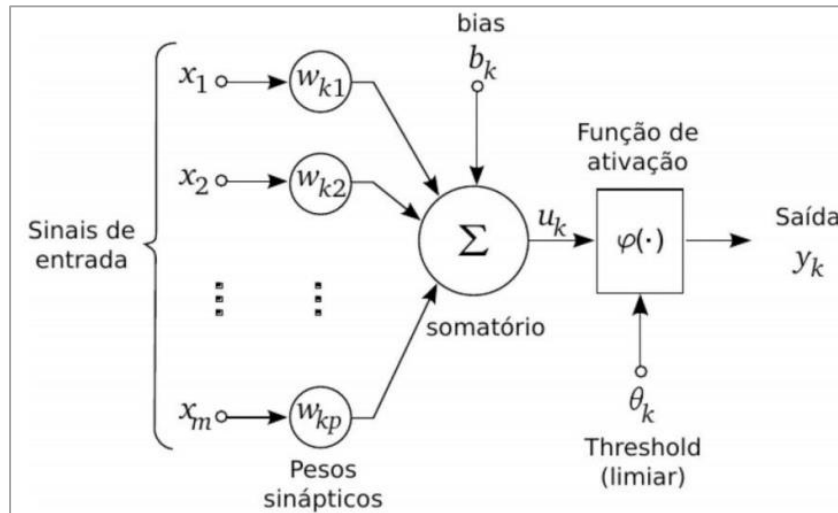
2.4.1 Surgimento e evolução das redes neurais artificiais

A primeira rede neural foi concebida por Warren McCulloch e Walter Pitts em 1943 e, inspirava-se no modelo biológico. McCulloch e Pitts tentaram entender a capacidade do cérebro produzir padrões altamente complexos a partir de muitas células básicas conectadas entre si. Essas células cerebrais básicas são chamadas de neurônios. No artigo “*A logical calculus of the ideas immanent in nervous activity*”, eles descreveram como os neurônios devem funcionar e, então, modelaram estas ideias, criando assim uma rede neural simples com circuitos elétricos (McCulloch e Pitts, 1943). O modelo de McCulloch e Pitts de um neurônio, contribuiu significativamente para o desenvolvimento de redes neurais artificiais que modelam matematicamente as principais características dos neurônios biológicos. Aos neurônios originais de McCulloch e Pitts, na época limitados, foram adicionados recursos que lhes permitiram “aprender”.

Abaixo, na Figura 9, Haykin (2001) representa o modelo de um neurônio artificial (conhecido

como unidade de processamento binária):

Figura 9 - Modelo de um neurônio artificial.



Fonte: Haykin (2001).

Em 1958, dando continuidade aos estudos relacionados às redes neurais artificiais, Frank Rosenblatt cria um algoritmo, denominado *Perceptron*, para reconhecimento de padrões baseado em uma rede neural computacional de duas camadas usando simples adição e subtração. Embora ele tenha proposto camadas adicionais com notações matemáticas, isto somente foi possível em 1975 com a contribuição de Kunihiko Fukushima.

Em 1960, dois estudiosos da Universidade de Stanford, Bernard Widrow e Marcian Hoff, desenvolveram os modelos conhecidos como ADALINE e MADALINE. ADALINE (sigla para *ADaptive LINear Element*) tinha como propósito reconhecer padrões binários de maneira tal que, se ele estivesse lendo bits de transmissão de uma linha telefônica, poderia prever o próximo bit (Widrow e Hoff, 1960). Enquanto MADALINE (sigla para *Multiple ADaptive LINear Elements*), como a própria sigla sugere, era uma rede multicamadas de unidades ADALINE (Widrow e Hoff, 1960). Embora muito antiga, a rede neural MADALINE continua em uso comercial atualmente, sendo considerada a primeira rede neural aplicada a um problema do mundo real (onde usava um filtro adaptativo para eliminar ecos em linhas telefônicas).

Com a publicação de “*Perceptrons: An Introduction to Computational Geometry*”, por Minsky e Papert em 1969, aliada à limitação da eletrônica disponível na época, houve uma redução significativa de pesquisas relacionadas às redes neurais. Minsky e Papert

demonstraram a limitação de redes neurais artificiais construídas com uma única camada, como *Perceptron* e ADALINE, e a impossibilidade de classificar problemas não lineares com as redes neurais artificiais (Minsky e Papert, 1988).

As pesquisas só foram retomadas com maior entusiasmo em 1980. Neste mesmo ano, Kunihiko Fukushima cria a rede neural verdadeiramente de multicamadas. A Neocognitron criada por Fukushima, era uma rede neural de hierarquia, de multicamadas, utilizada para o reconhecimento de caligrafia e solução de outros problemas de reconhecimento de padrões (Fukushima, 1980).

Poucos anos depois, em 1982, Hopfield apresentou um estudo demonstrando matematicamente a capacidade das redes neurais artificiais. Segundo Hopfield (1982), ela poderia ser utilizada não somente para modelar cérebros, mas também para criar dispositivos úteis.

Em 1986, com a contribuição David Rumelhart, Geoffrey Hinton e Ronald Williams, surgem as redes *Backpropagation*, que distribuem erros de reconhecimento de padrões em toda a rede (Rumelhart et al., 1986). Estas redes, de multicamadas, aprendem mais lentamente, pois necessitam de milhares de iterações para aprender, mas geram um resultado muito preciso.

Embora algoritmos que usavam redes neurais de aprendizagem profunda (*Deep Learning*) tenham sido criados em 1989, o uso destes no mundo real tornou-se impraticável devido aos altos tempos de treinamento (medidos em dias). Os hardwares até o momento não haviam evoluído o suficiente. A aprendizagem profunda (*Deep Learning*) só foi ganhar notoriedade em meados dos anos 2000, quando Hinton e Salakhutdinov mostram como uma rede neural de multicamadas poderia ser pré-treinada uma camada por vez (Hinton e Salakhutdinov, 2006).

No ano de 2009, descobre-se que com um conjunto de dados suficientemente grande, as redes neurais não precisam de pré-treinamento e as taxas de erro caem significativamente. Entre os anos de 2012 e 2016, pode-se citar avanços importantes para a história das redes neurais artificiais, tais como: em 2012, algoritmos de reconhecimento de padrões artificiais alcançam desempenho em nível humano em determinadas tarefas; em 2015, algoritmos executam tarefas superiores de reconhecimento facial usando redes profundas que levam em conta 120 milhões de parâmetros; em 2016, o algoritmo do Google *DeepMind, AlphaGo*, mapeia as características do complexo jogo de tabuleiro Go, vencendo o campeão mundial, Lee Sedol, em um torneio em Seul.

Em 2017, o *Deep Learning* passou a ser amplamente utilizado no meio corporativo e em aplicações mobile. Eventos de tecnologia relacionados a *Data Science*, Inteligência Artificial e *Big Data*, entendem *Deep Learning* como sendo a principal tecnologia para criação de sistemas inteligentes.

Inicialmente, as redes neurais tinham como propósito criarem um sistema computacional capaz de resolver problemas como um cérebro humano. À medida que as pesquisas foram avançando, as redes neurais passaram a resolver tarefas específicas, desviando-se de uma abordagem estritamente biológica. Reconhecimento de fala, visão computacional, filtragem de redes sociais, diagnósticos médicos são exemplos das mais diversas tarefas suportadas atualmente pelas redes neurais.

Com o aumento do volume de dados (estruturados ou não) a níveis de *big data*, foram sendo desenvolvidos modelos de *Deep Learning* (redes neurais de multicamadas que permitem captura e mineração de volumes maiores de dados, sejam eles estruturados ou não).

2.4.2 Inteligência artificial e definição de RNA

“A inteligência artificial é uma disciplina que estuda como as pessoas resolvem problemas e como as máquinas podem emular este comportamento humano de ‘solução de problemas’. Em outras palavras, como fazer com que as máquinas sejam mais atribuídas de características da inteligência humana” (Shaw e Simões, 1999).

De acordo com Cunha (2001), pode-se dizer, portanto, que a inteligência artificial reproduz características inteligentes do cérebro humano, de modo a fazer com que as máquinas solucionem problemas e realizem tarefas que até então são mais bem realizadas pelo homem. Para tal, lança-se mão de algumas técnicas, dentre elas as redes neurais. A seguir pode-se verificar, além das redes neurais, outras técnicas listadas por Cunha (2001):

Algoritmos Genéticos: através dos quais possíveis soluções de um problema são combinadas e alteradas, por mecanismos inspirados na seleção natural (de Darwin), no *Crossing-Over* e na mutação genética. Nesta técnica, obtém-se um conjunto de soluções em vez de uma única solução;

Sistemas Especialistas: inspirados na lógica do raciocínio humano, avaliam, encadeiam e definem quando utilizar determinado conhecimento sobre um assunto, expresso em regras;

Redes Neurais: inspiradas nos princípios do processamento neurofisiológico, se caracterizam pelo aprendizado através de exemplos e pela capacidade de generalização. As redes neurais baseiam-se no processamento paralelo e distribuído das informações;

Sistemas Híbridos: combinam uma ou mais técnicas de Inteligência Artificial.

Segundo Haykin (2001) o que motivou o trabalho em redes neurais artificiais foi o modo como o cérebro humano processa informações, que difere inteiramente do modo de processamento do computador digital convencional. Considerado por Haykin (2001) como um sistema de processamento de informação (computador) altamente complexo, não-linear e paralelo, o cérebro é capaz de organizar seus constituintes estruturais, conhecidos por neurônios, de modo a realizar determinados processamentos (tais como o reconhecimento de padrões) muito mais ágil que o mais rápido computador digital hoje existente.

“Uma rede neural é um processador maciçamente e paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. Ela se assemelha ao cérebro em dois aspectos:

- *O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.*
- *Forças de conexão entre neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido” (Haykin, 2001).*

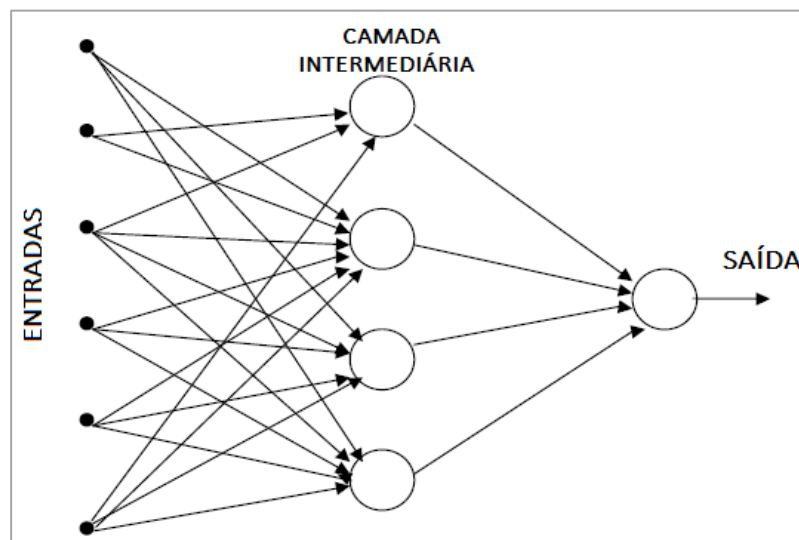
A habilidade de aprender faz com que as redes neurais artificiais sejam aplicadas nos mais variados campos, como modelagem, análise de séries, processamento de sinais e controle (Haykin, 2001).

Além da habilidade de aprender pela experiência, as redes neurais artificiais possuem outras características inerentes ao sistema neural humano, tais como: não seguir programação; reconhecimento de padrão e generalização da resposta (mesmo diante de entradas muito distorcidas). Em contrapartida, elas são mais robustas que o sistema neural humano pois, enquanto o homem perde neurônios da ordem de 103 ao dia, as máquinas não perdem nada (Cunha, 2001).

2.4.3 Estrutura de uma RNA

Uma rede neural artificial é composta por muitos elementos processantes, denominados neurônios, arranjados em camadas. A primeira camada difere-se das demais, sendo composta por nós. Nesta camada não há processamento, ocorrendo apenas o recebimento e distribuição das variáveis de entrada. A última camada fornece as variáveis de saída. As camadas internas, também chamadas de intermediárias ou ocultas, tem como função intervir de maneira útil entre a entrada externa e a saída da rede. Estas camadas intermediárias, que podem ou não existir, tornam a rede capaz de extrair estatísticas de ordem elevada (Haykin, 2001). A Figura 10 apresenta um esquema de uma RNA com uma camada intermediária.

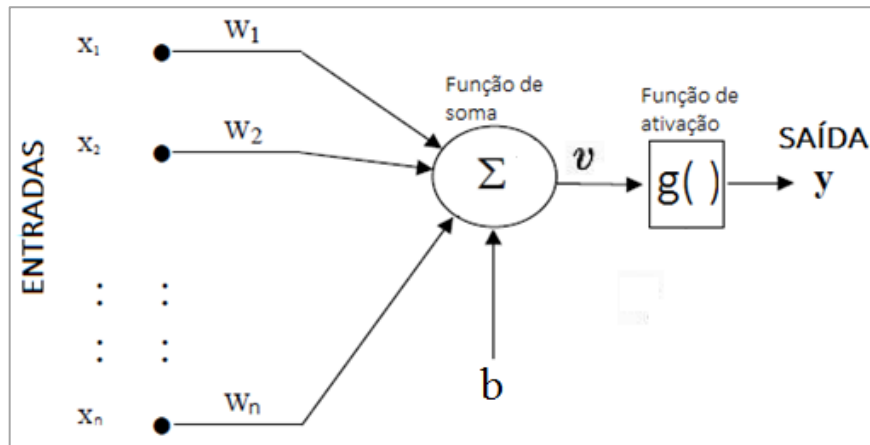
Figura 10 - Rede neural artificial com uma camada intermediária.



Fonte: Haykin, 2001.

O neurônio artificial é um modelo matemático inspirado na análise da geração e propagação de impulsos elétricos pelo sistema neural humano. Os neurônios artificiais conectam entre si através de conexões intercelulares, cada qual com um peso associado. Estes pesos sinápticos, que são fatores de multiplicação, caracterizam a forma de ativação específica de cada neurônio e a sua relevância. Esta ativação, denominada potencial de ativação, serve como argumento para a função de ativação, que é uma função das entradas recebidas (balanceadas pelo bias) e cujo resultado é enviado pelo neurônio, em forma de sinal, para diversos outros neurônios. A Figura 11 apresenta um exemplo de um neurônio artificial.

Figura 11 - Neurônio artificial.



Fonte: Haykin, 2001.

O neurônio artificial constitui-se dos seguintes elementos básicos (Da Silva et al., 2010; Haykin, 2001):

Sinais de entrada (X_1, X_2, \dots, X_n): sinais provenientes do meio externo, geralmente normalizados, incrementando a eficiência computacional dos algoritmos de aprendizagem. Estes sinais representam os valores assumidos pelas variáveis de uma aplicação específica;

Pesos sinápticos (W_1, W_2, \dots, W_n): valores utilizados para ponderar cada uma das variáveis de entrada da rede, quantificando as suas relevâncias em relação à funcionalidade do respectivo neurônio;

Combinador linear (Σ): agrega os sinais de entrada ponderados pelos respectivos pesos sinápticos, produzindo um valor de potencial de ativação;

Resultado do combinador linear (u): é a soma ponderada das entradas;

Limiar de ativação ou bias (b): variável que especifica o patamar apropriado para que o resultado produzido pelo combinador linear possa gerar um valor de disparo em direção à saída do neurônio;

Potencial de ativação (U): também conhecido como campo local induzido, é o resultado da diferença do valor produzido pelo combinador linear e pelo limiar de ativação. Quando o valor do potencial de ativação é positivo, o neurônio produz um potencial excitatório. Quando negativo, o neurônio produz um potencial inibitório. O potencial de ativação é representado pela equação: $U = u + b$;

Função de ativação (ϕ): tem como objetivo limitar a saída do neurônio dentro de um intervalo de valores razoáveis a serem assumidos pela sua própria imagem funcional;

Sinal de saída (y): valor final produzido pelo neurônio, de acordo com um determinado conjunto de sinais de entrada, podendo ser também utilizado por outros neurônios que estão sequencialmente interligados.

Os pesos sinápticos, características de significativa importância no neurônio artificial, são vistos matematicamente como vetores de valores para um neurônio ou uma matriz para um conjunto de neurônios. O valor dos pesos é modificado em função da intensidade do sinal de entrada pelo processo de aprendizagem.

Os sinais de entrada (assemelham-se aos impulsos elétricos externos do neurônio biológico) são multiplicados pelos seus respectivos pesos sinápticos, ponderando-se, portanto, todas as informações externas que chegam ao neurônio (Da Silva et al., 2010). Torna-se possível, então, verificar que o resultado do combinador linear, denominado u , é a soma ponderada das entradas.

Na sequência, este valor é balanceado, através do bias, gerando assim o potencial de ativação. Se este valor for positivo, o sinal é repassado adiante através da saída. Caso contrário, o sinal não será transferido.

Para modelos mais simples de redes neurais artificiais, a função de ativação pode ser resultado somente da soma das entradas ponderadas do neurônio. Para modelos mais complexos, a função de ativação pode ter um processamento atribuído, como, por exemplo, a utilização de um valor prévio de saída como a entrada para o próprio neurônio (auto excitação) (Tafner et al., 1996).

Dentre as diversas formas de função de ativação, existem três tipos básicos: limiar (ou *heaviside*), linear por partes e sigmoide (Haykin, 2001).

2.4.4 Arquitetura da RNA

O algoritmo (regra) de aprendizagem utilizado para o treinamento de uma rede neural está diretamente ligado à forma na qual os neurônios desta rede estão estruturados. A estrutura (denominada arquitetura da rede) utilizada influencia significativamente no desempenho da rede, de modo que a escolha errada acarreta falhas ou até mesmo a incapacidade de resolver dado problema. Conforme Haykin (2001), pode-se identificar duas classes de arquiteturas

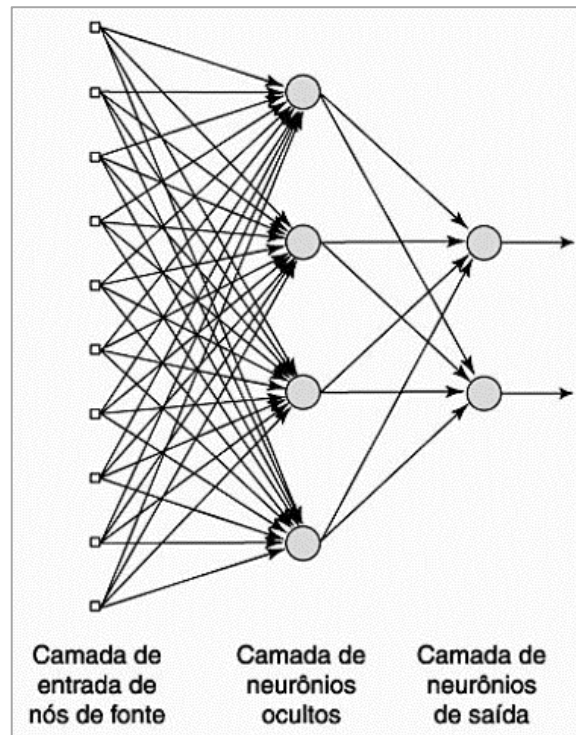
fundamentalmente diferentes: redes *feedforward* (que podem ser com camada única ou múltiplas camadas) e redes recorrentes. Neste trabalho, utilizou-se redes do tipo *feedforward*.

Redes *feedforward*: Nas redes *feedforwards* tem-se uma camada de entrada composta por nós, uma ou mais camadas intermediárias (também conhecidas como ocultas ou camadas de neurônios ocultos). Estas camadas intermediárias não existem no caso das redes *feedforward* com camada única, sendo estas, portanto, compostas apenas por uma camada de entrada e uma de saída) compostas por neurônios e uma camada de saída composta por neurônios (Haykin, 2001). Na camada de entrada não há processamento das informações, ocorrendo apenas o recebimento e distribuição dos estímulos externos. A propagação do sinal se dá em um único sentido, adiante, seguindo camada por camada. Churchland e Sejnowski (1992) afirmam que as redes adquirem uma perspectiva global, apesar de sua conectividade local, devido à dimensão extra de interações neurais e ao acréscimo significativo de conexões sinápticas.

Além da classificação de padrões, as redes *feedforward* com camadas múltiplas podem ser utilizadas para solução de problemas de regressão linear, solução de problemas de aproximação de funções, otimização, robótica, identificação de sistemas e controle de processos.

Como exemplos de redes com este tipo de arquitetura, pode-se citar o perceptron de múltiplas camadas (*multilayer perceptron* - MLP) e a rede de função de base radial (*radial-basis function* - RBF). A Figura 12 apresenta um exemplo de uma rede *feedforward* com uma camada oculta e uma camada de saída.

Figura 12 - Rede *feedforward* com uma camada oculta e uma camada de saída.



Fonte: Haykin, 2001.

Neste tipo de rede, a quantidade de neurônios que compõem a primeira camada escondida normalmente difere do número de sinais que compõem a camada de entrada da rede (Da Silva et al., 2010). A complexidade e o tipo de problema a ser solucionado pela rede define o número de camadas escondidas e seus respectivos neurônios constituintes.

Redes recorrentes ou *feedback*: Uma rede é considerada recorrente, ou *feedback*, quando pelo menos um de seus neurônios apresenta conexões sinápticas que enviam informações para alguma camada anterior à sua (conhecidas como laços de realimentação), de forma que seus impulsos podem realimentar a rede e, possivelmente, suas próprias entradas (Haykin, 2001). Analogamente ao processo biológico, pode-se citar a troca de informações entre neurônios da mesma região do cérebro.

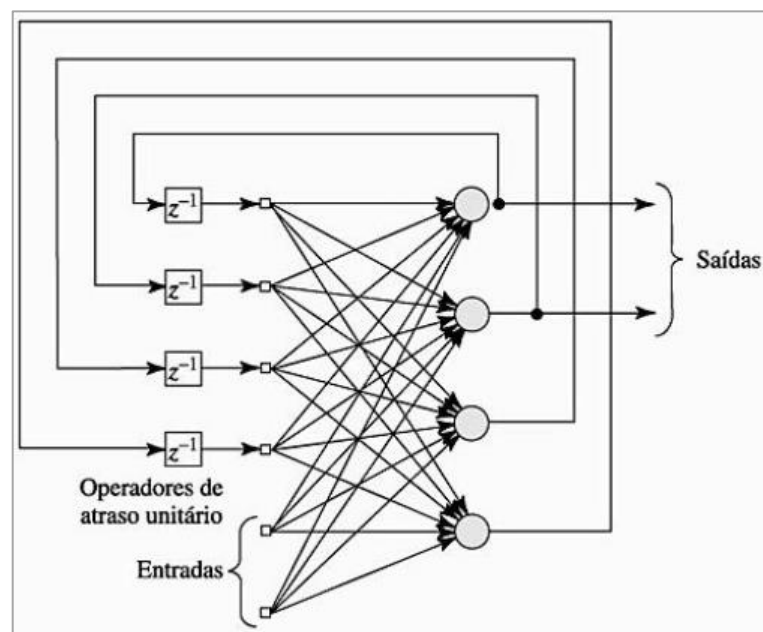
A realimentação faz com que estas redes, conhecidas como redes com memória, respondam dinamicamente aos estímulos, onde a cada recebimento de nova entrada, calcula-se a saída que é realimentada, modificando a entrada (Velasco, 2007). A capacidade de aprendizagem e o desempenho da rede são altamente influenciados pelos laços de realimentação. “Além disso, os laços de realimentação envolvem o uso de ramos particulares compostos de elementos de atraso

unitário (z^{-1}), o que resulta em um comportamento dinâmico não-linear, admitindo-se que a rede neural contenha unidades não-lineares” (Haykin, 2001).

Além da aplicação em controle de processos, otimização e identificação de sistemas, a dinamicidade das redes neurais recorrentes permitem que elas sejam utilizadas em sistemas variantes em relação ao tempo, tais como previsão de séries temporais.

Um modelo de rede neural recorrente totalmente conectada bastante conhecido na literatura é o modelo de Hopfield. A Figura 13 apresenta um exemplo de uma rede recorrente.

Figura 13 - Rede recorrente.



Fonte: Haykin, 2001.

2.4.5 Treinamento da RNA

Segundo Haykin (2001): “*aprendizagem é um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual a rede está inserida. O tipo de aprendizagem é determinado pela maneira pela qual a modificação dos parâmetros ocorre. Esta definição do processo de aprendizagem implica a seguinte sequência de eventos:*

- *A rede neural é estimulada por um ambiente.*
- *A rede neural sofre modificações nos seus parâmetros livres como*

resultado desta estimulação.

- *A rede neural responde de uma maneira nova ao ambiente, devido às modificações ocorridas na sua estrutura interna”.*

Segundo Sabbatini (1992), o aprendizado da rede neural é uma modificação da matriz de pesos desta rede, de forma a otimizar o mapeamento dos padrões de entrada e de saída. Ainda conforme Sabbatini (1992), é desejável que este processo se dê de forma auto-organizada, através de um algoritmo apropriado, onde a rede autoajusta seus pesos durante a progressão do aprendizado.

O processo de aprendizado se dá através do treinamento da rede neural e consiste, de acordo com Da Silva et al. (2010) em aplicar passos ordenados necessários para a sincronização dos pesos sinápticos e limiares de seus neurônios, a fim de que sejam geradas soluções pelas suas saídas. Estas soluções representam o sistema físico mapeado pela rede.

As diversas maneiras de treinamento propostas para o processo de aprendizado das redes neurais artificiais se subdividem em dois paradigmas principais: o treinamento supervisionado e o não supervisionado. Neste trabalho, utilizamos o processo de treinamento supervisionado.

2.4.5.1 Treinamento Supervisionado

No treinamento supervisionado as redes neurais artificiais são treinadas antes de serem utilizadas.

Este tipo de treinamento, utilizado para aprendizado pela maioria das redes neurais artificiais, supervisiona a diferença entre a saída atual da rede e a saída desejada, sendo esta diferença usada no procedimento de ajuste (Da Silva et al., 2010). São informados valores iniciais aleatórios para os pesos, que vão sendo ajustados pela rede a cada ciclo ou iteração, através do algoritmo de aprendizado. O ajuste sináptico é realizado continuamente até que se tenha certa precisão da rede, minimizando o fluxo corrente de erros de todos os elementos em processamento.

É de suma importância que, após o treinamento supervisionado, seja realizada uma análise do que a rede é capaz de realizar com os dados que ainda não foram apresentados à mesma. Se a rede não alcançar um certo nível de performance com este novo conjunto de dados (chamado

conjunto de teste), presume-se que o treinamento da rede ainda não foi suficiente. Esta análise, portanto, assegura que a rede simplesmente não memorizou um dado conjunto de dados, mas sim aprendeu os modelos / padrões gerais envolvidos na aplicação.

Em determinados momentos, o problema de performance da rede neural pode estar relacionado à má qualidade dos dados usados para o treinamento e não a um problema da rede em si. Logo, é necessário um conjunto suficientemente grande de dados, de forma a garantir que contenha as informações necessárias para que a rede aprenda os moldes e as relações importantes.

Atingido o nível de performance adequado, o treinamento é considerado completo e a rede neural artificial pode então ser utilizada.

Quando as redes neurais artificiais são utilizadas na modelagem de processos, é usual utilizar-se o treinamento supervisionado. Dentre os algoritmos baseados no conceito de aprendizagem supervisionada, pode-se citar *Backpropagation*, ADALINE e MADALINE.

2.4.5.2 *Treinamento não supervisionado*

Considerado como a grande promessa para o futuro, o treinamento não supervisionado de redes neurais dispensa a necessidade de um conjunto de dados de treinamento para o processo de aprendizado.

Sem a utilização de saídas desejadas para determinar se o aprendizado converge para o caminho certo, a própria rede se estabelece em relação às individualidades dos elementos que constituem o conjunto total de dados (auto nivelção), identificando os semelhantes. Em outras palavras, “os neurônios individuais da rede aprendem a se especializar em agrupamentos de padrões similares; fazendo isso, eles se tornam detectores de características para classes diferentes de padrões de entrada” (Haykin, 2001).

Uma vez reajustada aos parâmetros estatísticos das informações de entrada, a rede neural amplifica sua habilidade de codificar as características de entrada e formar novas classes.

A propriedade de auto nivelção deve-se às regras definidas no algoritmo de aprendizagem. A regra utilizada pelo treinamento não supervisionado é a competitiva. Nela, as unidades de processamento competem e somente os pesos pertencentes à unidade vencedora são ajustados.

No treinamento não supervisionado, as ações da rede são reforçadas quando melhorarem o desempenho da rede. Caso contrário, ocorrerá uma penalização e a probabilidade de ocorrência de determinadas ações em situações futuras reduz.

2.4.5.3 Algoritmos de Treinamento

Existem diversos algoritmos, também conhecidos como regras de aprendizagem, utilizados para o treinamento das redes neurais artificiais. A maioria das regras de aprendizagem encontra-se em uso e costuma ser variações da mais conhecida, que é a regra de aprendizagem Hebbiana. Dentre as regras de aprendizagem existentes, Haykin (2001) cita as seguintes:

Regra de aprendizagem Hebbiana – contida no livro *The Organization of Behavior*, a primeira regra de aprendizagem, conforme Hebb, parte do seguinte princípio: “*se dois neurônios em ambos os lados de uma sinapse são ativados sincronamente e simultaneamente, então a força daquela sinapse é seletivamente aumentada*” (Hebb, 1949). Neste processo de treinamento os pesos das conexões são ajustados, portanto, baseados nas atividades dos neurônios.

Regra de aprendizagem por correção de erro – também conhecida como regra delta de Widrow e Hoff, esta regra de aprendizagem altera continuamente a intensidade e a importância das conexões de entrada, reduzindo de forma significativa a diferença entre o valor de saída desejado e o valor atual de saída do neurônio. Esta regra altera, portanto, os pesos sinápticos de modo a minimizar o erro da rede. O erro calculado na saída é transformado pela derivação da função de transferência e usado para ajustar os pesos de entrada da camada prévia da rede, ou seja, o erro é retro propagado às camadas anteriores, uma de cada vez. Os pesos sinápticos são ajustados até atingirem um estado estável.

Regra de aprendizagem baseada em memória – a rede procura um padrão aproximado para a entrada, reconhecendo, pela regra do vizinho, padrões a partir dos neurônios mais próximos. Estes padrões são baseados em experiências passadas, armazenadas em uma memória de exemplos de entrada-saída classificados corretamente.

Regra de aprendizagem competitiva – ao ser fornecido um padrão à rede, o neurônio que responder melhor terá os pesos de suas ligações reforçados. Em alguns casos pode ocorrer também o reforço dos pesos dos neurônios vizinhos. Os neurônios competem entre si para se

tornarem ativos. O neurónio que apresentar um campo local induzido maior é excitado e envia um sinal unitário para a saída, enquanto os outros neurónios são inibidos a enviarem sinais.

Regra de aprendizagem de Boltzmann – trata-se de um algoritmo estocástico baseado na mecânica estatística de Boltzmann, caracterizada por uma função de energia. Nesta regra, cada neurónio possui um estado, representado por um valor binário (+1 ligado / -1 desligado), e está incluído em um grupo funcional (visível / oculto). Nela há dois modos de operação (condição livre / condição presa). O ajuste dos pesos das ligações é realizado de acordo com a correlação demonstrada pelos neurónios envolvidos em cada ligação.

2.4.5.3.1 Algoritmo *Backpropagation*

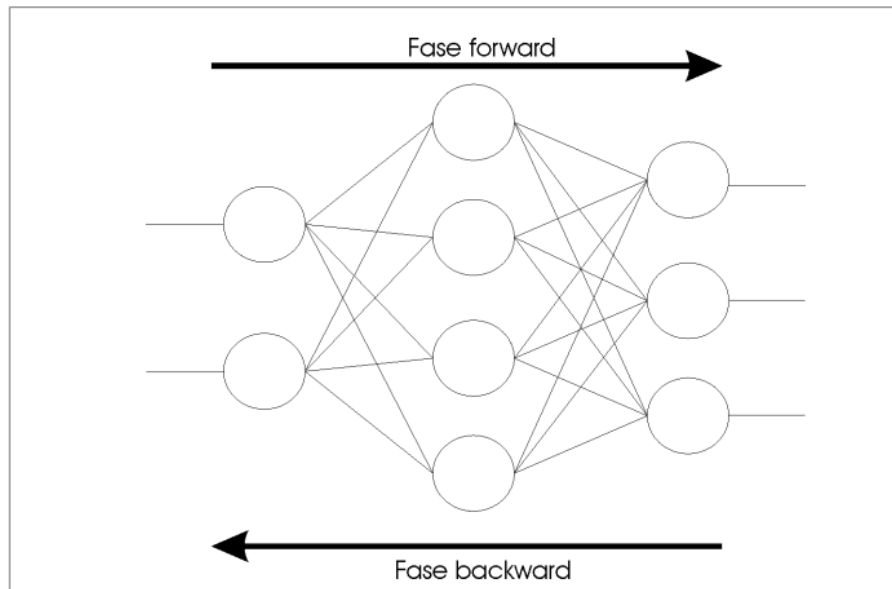
O algoritmo *Backpropagation*, também conhecido como regra de aprendizagem delta generalizada, tem como objetivo ajustar os pesos e os bias de forma a reduzir o erro da rede. Assim como a regra delta, o *Backpropagation* é baseado no método do gradiente descendente.

Durante a etapa de treinamento, o conjunto de dados de entrada (ou padrão) é apresentado à rede e então propagado adiante, camada por camada, até a produção do sinal de saída. Esta saída é então comparada com a saída desejada, definindo o erro da rede. Este erro é retro propagado, camada a camada, ajustando assim os pesos sinápticos de cada camada no sentido oposto à propagação adiante. Os pesos sinápticos são ajustados com o intuito de fazer com que a resposta real da rede se aproxime da resposta desejada.

As fases do processo de treinamento são repetidas até que se tenha realizado um predeterminado número de iterações (épocas), ou até que a diferença entre os erros de duas iterações consecutivas seja próxima de zero (ou fique abaixo de uma constante ϵ) ou, ainda, até que a porcentagem de classificações corretas esteja acima de uma determinada constante (critério mais indicado para saídas binárias).

Pode-se então dizer que o algoritmo de treinamento *Backpropagation* é composto por duas fases, sendo que cada uma percorre um sentido da rede. Na primeira fase, denominada *forward*, a saída é definida para um determinado padrão de dados de entrada. Esta saída é comparada com a saída desejada, determinando assim o erro. Na segunda fase, o erro é retro propagado, ajustando os pesos sinápticos da rede neural. A Figura 14 indica o fluxo de processamento do algoritmo de treinamento *Backpropagation*.

Figura 14 - Fluxo de processamento do algoritmo de treinamento *Backpropagation*.



Fonte: Braga et. al, 1998.

Em resumo, segundo Braga *et al.* (1998) é possível definir as fases do algoritmo de treinamento *Backpropagation* da seguinte maneira:

Fase *forward*

1. Os sinais de entrada são apresentados à primeira camada intermediária (c_i) que compõe a rede;
2. Para cada camada c_i a partir da camada de entrada: os sinais de saída dos neurônios da camada c_i alimentam a entrada da próxima camada (c_{i+1});
3. As saídas produzidas pelos neurônios da última camada são comparadas às saídas desejadas.

Fase *backward*

1. Da última camada da rede até a entrada:
 - 1.1. Os neurônios artificiais da camada atual (c_a) ajustam seus pesos sinápticos de forma a reduzir os erros;
 - 1.2. O erro de um neurônio das camadas intermediárias é calculado utilizando os erros retropropagados dos neurônios que pertencem às camadas seguintes conectados a ele, os quais são ponderados pelos pesos das conexões entre eles.

Desta forma, Braga *et al.* (1998) apresentam o algoritmo de treinamento *Backpropagation*:

1. A rede é inicializada (pesos e parâmetros);
2. É repetido para cada padrão / conjunto de dados de treinamento:
 - 2.1.A saída da rede é calculada através da fase *forward*;
 - 2.2.A saída calculada é comparada com a saída desejada;
 - 2.3.Os pesos sinápticos são atualizados através da fase *backward*;
3. A etapa 2 é repetida por um determinado número de iterações (épocas) ou até o erro ser mínimo.

2.4.6 Tarefas de aprendizagem

A escolha de um algoritmo de treinamento está diretamente ligada à tarefa de aprendizagem que a rede neural artificial deve executar. Haykin (2001) identifica seis tarefas de aprendizagem que se aplicam ao uso de redes neurais artificiais:

Associação de Padrões: A memória associativa é uma memória distribuída inspirada no cérebro humano, que por sua vez tem como característica o aprendizado por associação. Conforme Anderson e Sutton (1995), todos os modelos de cognição utilizam a associação como operação básica. Existem duas formas de associação: auto associação e heteroassociação.

Na auto associação, um conjunto de padrões apresentados repetidamente devem ser armazenados pela rede neural. Na sequência, é apresentada à rede uma descrição parcial ou distorcida de um padrão original armazenado e a rede neural deve recuperar / recordar aquele padrão particular.

A diferença básica entre a auto associação e a heteroassociação é que nesta última um conjunto arbitrário de padrões de entrada é associado a um outro conjunto arbitrário de padrões de saída. Na auto associação o treinamento é não supervisionado, enquanto, na heteroassociação, é utilizado o treinamento supervisionado.

Reconhecimento de Padrões: Assim como acontece com os seres humanos, as redes neurais realizam o reconhecimento de padrões a partir de um processo de aprendizagem.

“O reconhecimento de padrões é formalmente definido como o processo pelo qual um padrão / sinal recebido é atribuído a uma classe dentre um número determinado de classes

(categorias)” (Haykin, 2001). A rede neural é inicialmente treinada utilizando repetidamente um conjunto de padrões juntamente com a categoria à qual ele pertence. Posteriormente, é apresentado à rede um novo padrão que não foi visto antes, mas que pertence à mesma população dos padrões de treinamento da rede. A rede neural identifica a classe daquele padrão devido às informações extraídas no treinamento.

O reconhecimento de padrões das redes neurais artificiais é de natureza estatística, onde os padrões são representados por pontos em um espaço de decisão multidimensional dividido em regiões, cada uma associada a uma classe.

Aproximação de Funções: No caso da aproximação de funções, o objetivo básico da aprendizagem é descobrir a função a ser disponibilizado um conjunto de pares entrada-saída que reproduzem o sistema a ser tratado. E para esta tarefa de aprendizagem, as redes neurais artificiais são bastante úteis, visto que são aproximadores universais, eficientes e podem ser implementadas como máquinas de aprendizagem.

Neste contexto, portanto, as redes neurais multicamadas treinadas podem ser consideradas um veículo prático para a realização de um mapeamento não-linear de entradas-saídas de natureza geral.

Quando se utiliza uma rede neural artificial com uma única camada intermediária, como os neurônios desta camada tendem a interagir entre si globalmente, há uma dificuldade em melhorar a aproximação em um determinado ponto sem piorá-la em outro. Portanto, é usual que se utilize pelo menos duas camadas intermediárias, onde o processo de aproximação torna-se mais gerenciável e podem ser alcançados melhores resultados (Chester, 1990).

A aproximação de funções em redes neurais artificiais de duas camadas intermediárias ocorre da seguinte maneira:

- **Primeira camada intermediária:** nesta camada são extraídas as características locais. Um determinado número de neurônios é utilizado para dividir a entrada em regiões, enquanto outros neurônios aprendem as características locais destas regiões;
- **Segunda camada intermediária:** nesta camada são extraídas as características globais. As saídas de neurônios da primeira camada intermediária são combinadas em um neurônio da segunda camada intermediária, em uma determinada região, fazendo com que este neurônio aprenda as características globais da referida região e forneça “zero” como saída nas outras regiões.

Controle: O controle de uma planta (processo ou parte crítica de um sistema) trata-se de, como o próprio nome sugere, manter esta planta em uma condição controlada.

Em um dado sistema de controle realimentado, a saída da planta é diretamente realimentada para a entrada e subtraída de um sinal de referência fornecido por uma fonte externa. O sinal de erro produzido através desta operação é então aplicado a um controlador neural, de forma a ajustar os seus parâmetros livres. O controlador, portanto, tem como objetivo fornecer entradas apropriadas para a planta, fazendo com que a sua saída siga o sinal de referência. Conforme Haykin (2001), pode-se adotar uma das duas abordagens para controle de uma planta (linha de produção):

- **Aprendizagem indireta:** constrói-se um modelo baseado em rede neural artificial, utilizando medidas de entrada-saída reais da planta, de forma a produzir uma cópia da planta. Utiliza-se este modelo para fornecer uma estimativa da matriz jacobiana e suas derivadas parciais são, posteriormente, utilizadas no algoritmo de aprendizagem por correção de erro, calculando-se assim os ajustes dos parâmetros livres do controlador neural;
- **Aprendizagem direta:** geralmente os sinais das derivadas parciais são conhecidos e se mantêm constantes ao longo do intervalo dinâmico da planta. Desta forma, pode-se aproximar as derivadas parciais pelos seus sinais individuais e seus valores absolutos recebem uma representação distribuída nos parâmetros livres do controlador neural. Assim, o controlador neural torna-se capaz de aprender os ajustes de seus parâmetros livres diretamente da planta.

Filtragem: Filtro é um dispositivo ou algoritmo que extrai informações sobre uma determinada grandeza de interesse a partir de um conjunto de dados ruidosos. As fontes do ruído são as mais variadas, tais como medição por meio de sensores ruidosos, corrompimento do sinal pela transmissão através de um canal de comunicação, corrompimento da informação por um sinal de interferência captado do meio ambiente.

Haykin (2001) esclarece que os filtros podem ser utilizados para a realização de três tarefas de processamento de informação:

- **Filtragem:** a informação é extraída sobre uma quantidade de interesse no tempo discreto n , utilizando dados medidos até este tempo n inclusive;

- **Suavização:** neste caso, não há necessidade de que a informação sobre a grandeza de interesse esteja disponível no tempo n e pode-se obter a informação utilizando os dados medidos após este tempo n . Desta forma, há um atraso na produção do resultado de interesse, bem como uma maior precisão em um sentido estatístico, visto que são utilizados dados não somente até o tempo, mas também após este tempo;
- **Previsão:** o processamento de informação objetiva a predição, derivando a informação sobre como será a grandeza de interesse em um determinado tempo $n + n_0$ futuro, para algum $n_0 > 0$, fazendo-se uso dos dados medidos até o tempo n inclusive.

Formação de feixe: A formação de feixe é uma forma de filtragem espacial e objetiva distinguir as propriedades espaciais de um sinal alvo do ruído de fundo. Ela é, portanto, utilizada para produzir uma seletividade por atenção.

Como exemplo de aplicação da formação de feixe, podem ser citados os sistemas de radar e sonar, onde o objetivo principal é detectar e perseguir um alvo de interesse na presença combinada de ruído do receptor e sinais de interferência ou obstruções. A complexidade desta tarefa deve-se ao fato de que o sinal alvo se origina em uma direção desconhecida e não há a priori informação disponível sobre os sinais de interferência.

Nestas situações pode ser utilizado um cancelador de lóbulo lateral generalizado (CLLG), constituído por:

- Um arranjo de elementos de antenas, permitindo uma amostragem do sinal observado em pontos discretos do espaço;
- Um combinador linear definido por um conjunto de pesos fixos e cuja saída é uma resposta desejada, agindo assim como um filtro espacial caracterizado por um padrão de radiação. O lóbulo principal deste padrão aponta para uma direção predeterminada, para a qual o CLLG deve ser restrito, de forma a produzir uma resposta sem distorções;
- Uma matriz bloqueadora de sinal que tem como função o cancelamento da interferência que escapa dos lóbulos laterais do padrão de radiação do filtro espacial que representa o combinador linear;
- Uma rede neural artificial com parâmetros ajustáveis projetada de forma a acomodar variações estatísticas nos sinais de interferência. Os ajustes dos parâmetros livres da rede neural são realizados através de um algoritmo de aprendizagem por correção de erro.

A diversidade das seis tarefas de aprendizagem mostra a universalidade das redes neurais artificiais como sistemas de processamento de informação. “*Em um sentido fundamental, todas estas tarefas de aprendizagem são problemas relativos a aprender um mapeamento a partir de exemplos (possivelmente ruidosos) de mapeamentos*” (Haykin, 2001).

3 Metodologia

O problema estudado trata-se do baixo acerto das variáveis de temperatura e carbono ao final da etapa de sopro de oxigênio, ocasionando alto tempo de espera por resultado da análise química em laboratório e do índice de resopro (retrabalho), quando os resultados da análise química não são satisfatórios. A partir dos dados existentes, é verificada a correlação entre as variáveis de processo, permitindo inferir os valores futuros de variáveis de interesse, por meio de redes neurais artificiais.

3.1 Instrumentos e formas de coleta

Para desenvolvimento do trabalho em questão são utilizados dados quantitativos, primários, oriundos de sistemas “legados” da empresa (análogo ao Diário de Bordo). Desta forma, como os dados estão em sistemas próprios dos processos, estes devem ser coletados/separados através de uma pesquisa das informações utilizando as ferramentas *Access*, *Excel* e *Power Query* da *Microsoft*®. Para tal pesquisa dos dados, é criada uma “consulta” metodológica ao banco de dados, onde são estabelecidos critérios de seleção dos dados, como período temporal, variáveis a serem coletadas, processos em foco, entre outros.

O período de consulta foi definido entre 25 de agosto/2020 e 30 de novembro/2022. Os dados anteriores a este período foram desconsiderados em função de mudança de processo ocorrida em agosto/2020, o que afeta o processo e torna os dados inviáveis de serem analisados. Os dados coletados representam a população, e não uma amostragem propriamente dita, pois são utilizados todos os dados do período possível. Uma tratativa estatística é utilizada apenas para que sejam retirados valores incoerentes ou discrepantes que poderiam afetar de forma indesejável os resultados.

As variáveis utilizadas no trabalho foram definidas com base em trabalhos do mesmo assunto já publicados na literatura, como o de Cunha (2001) juntamente com indicações de especialistas da área técnica. Também foram utilizadas técnicas estatísticas para verificar a contribuição de cada variável de entrada no resultado da variável de saída. O resultado desta coleta de dados,

por fim, passou por um tratamento de dados visando a eliminação de dados zerados e negativos (quando for o caso de algumas variáveis), “*outliers*” e valores fora de faixa prevista para cada variável. Também foi considerado se os valores pertenciam às faixas de valores aceitáveis para cada variável.

3.2 Análise dos Dados

Diante da coleta dos dados através da ferramenta *Access* no sistema “legado” do processo, o próximo passo foi realizar o tratamento destes dados, transformando-os em informação a ser utilizada na criação da rede neural. Para este tratamento de dados foram utilizadas as ferramentas *Excel* e *Power Query*, com o intuito de realizar um refinamento. Em seguida, a análise dos dados foi realizada através de uma rede neural artificial desenvolvida utilizando a linguagem de programação *Python*. O Quadro 1 apresenta o dicionário das variáveis do modelo, indicando o conceito de cada variável.

Quadro 1 - Dicionário das variáveis do modelo de rede neural.

Nº	Variável	Definição
1.	TEMP_FIM_SOPRO	Temperatura do aço líquido no final da etapa de sopro de oxigênio.
2.	C_FIM_SOPRO	Carbono do aço líquido no final da etapa de sopro de oxigênio.
3.	PERC_SUC	Percentual de sucata carregado.
4.	SI_GUSA	% Silício do gusa líquido carregado.
5.	LD_PARADO	Tempo (em minutos) em que o Convertedor ficou sem produzir.
6.	TEMP_GUSA_LIQ	Temperatura do gusa líquido carregado.
7.	VIDA_REVEST_LD	Número total de corridas produzidas no Convertedor.
8.	BASICIDADE	Relação Binária dos Óxidos de CaO e SiO ₂ Visada na escória de fim de sopro.
9.	PESO_SINTER	Peso de Sínter adicionado na corrida.
10.	VOLUME_O2	Volume de Oxigênio soprado na corrida.

Fonte: Autor (2023).

Com o objetivo de retirar valores indevidos da base de dados (zeros, nulos e *outliers*), criou-se uma regra de filtros na ferramenta *Power Query* de forma a deixar na base, somente os valores que são pertinentes ao processo. O Quadro 2 apresenta as regras de filtragem utilizadas em cada variável. Tais regras foram definidas juntos com os especialistas da área do processo.

Quadro 2 - Regras de filtragem das variáveis.

Variável	Filtro	Observação
ACERTO_SIMULTANEO	= 1	Manter as corridas onde o ACERTO_SIMULTANEO = 1.
MN_GUSA	< > nulo	Manter as corridas com valores de MN_GUSA diferentes de nulo.
RESSOPRO	= 0	Manter as corridas com valores de RESSOPRO = 0.
AÇO_RETORNADO	= 0	Manter as corridas com AÇO_RETORNADO = 0.
VOLUME_O2	>= 3530	Manter as corridas com o VOLUME_O2 maior ou igual a 3520 Nm ³ /h e menor ou igual a 4230 Nm ³ /h.
	<= 4230	
C_FIM_SOPRO	> 0	Manter as corridas com C_FIM_SOPRO maior do que zero pontos e menor ou igual a 50 pontos.
	<= 50	
VIDA_REVEST_LD	> 0	Manter as corridas com VIDA_REVEST_LD maior do que zero corridas.
LD_PARADO	<= 400	Manter as corridas com LD_PARADO menor ou igual a 400 minutos.
SI_GUSA	>= 20	Manter as corridas com o SI_GUSA maior ou igual a 20 pontos.
TEMP_GUSA_LIQ	<= 1400	Manter as corridas com TEMP_GUSA_LIQ menor ou igual a 1400 °C.
SINTER_SOPRO	<= 2700	Manter as corridas com SINTER_SOPRO menor ou igual a 2700 quilos.

Fonte: Autor (2023).

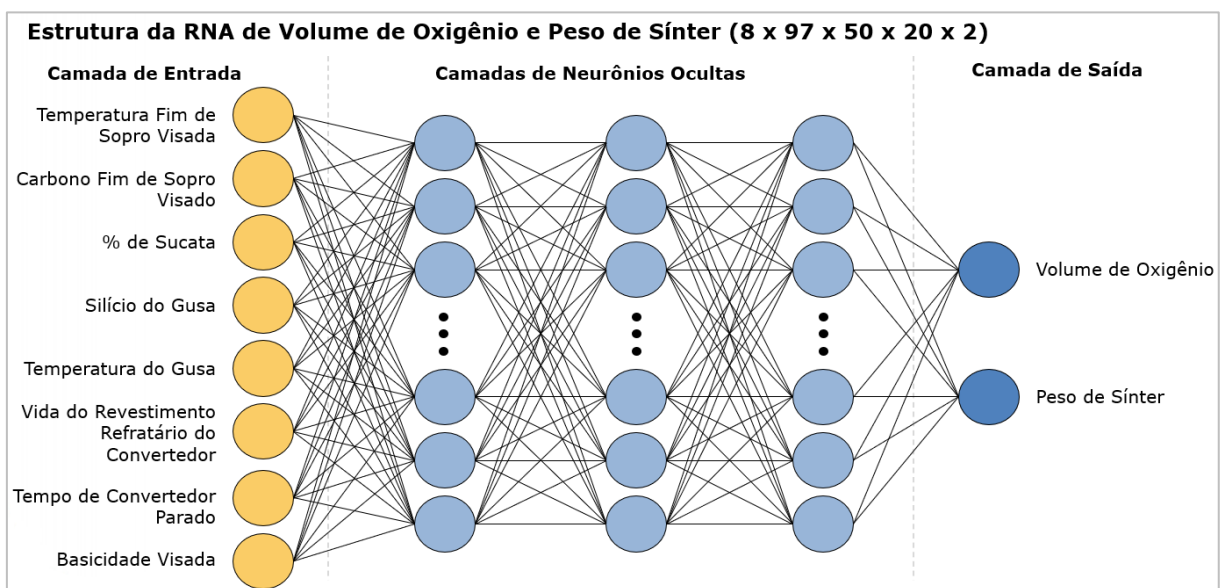
A base de dados possui um total de 24.095 observações, e após o tratamento dos dados, a base final ficou com 6.482 observações.

3.3 Variáveis de entrada e saída

Conforme indicado nos objetivos específicos deste trabalho, foram desenvolvidas duas RNAs diferentes, sendo uma para predição do volume de oxigênio que será soprado e o peso de sínter que será adicionado na corrida, e outra para predição da temperatura e carbono ao fim da etapa de sopro da corrida.

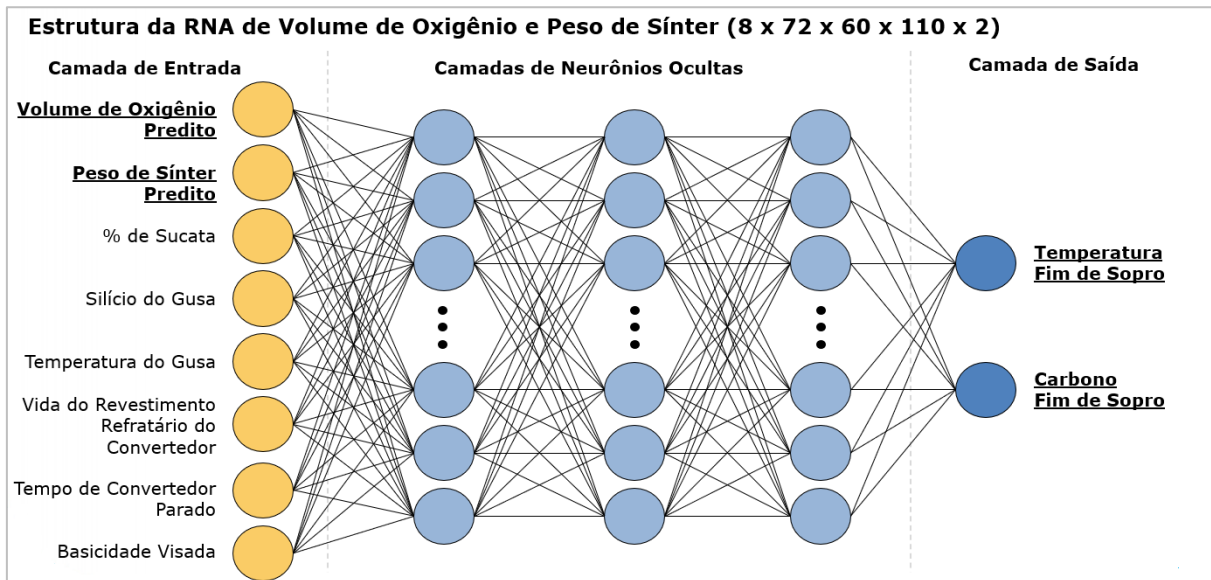
O desenvolvimento de duas RNAs justifica-se pelo fato da segunda RNA validar o resultado da primeira RNA, uma vez que, as variáveis de saída da primeira RNA são utilizadas como variáveis de entrada na segunda RNA para verificar se o volume de oxigênio e peso de sínter preditos são suficientes para atingir a temperatura e o carbono dentro da faixa visada ao final da etapa do sopro de oxigênio. As Figuras 15 e 16 apresentam a estrutura das duas RNAs desenvolvidas.

Figura 15 - Estrutura da RNA para predição do volume de oxigênio e peso de sínter.



Fonte: Autor (2023)

Figura 16 - Estrutura da RNA para predição da temperatura e carbono fim de sopro.



Fonte: Autor (2023).

A base de dados utilizada no treinamento e validação da RNA de predição do Volume de Oxigênio e o Peso de Sínter possui oito variáveis de entrada e duas variáveis de saída. Para a RNA de predição de Temperatura e Carbono de Fim de Sopro, a base de dados também possui oito variáveis de entrada e duas variáveis de saída. Em todos os casos, as bases de dados possuem um total de 6.482 observações.

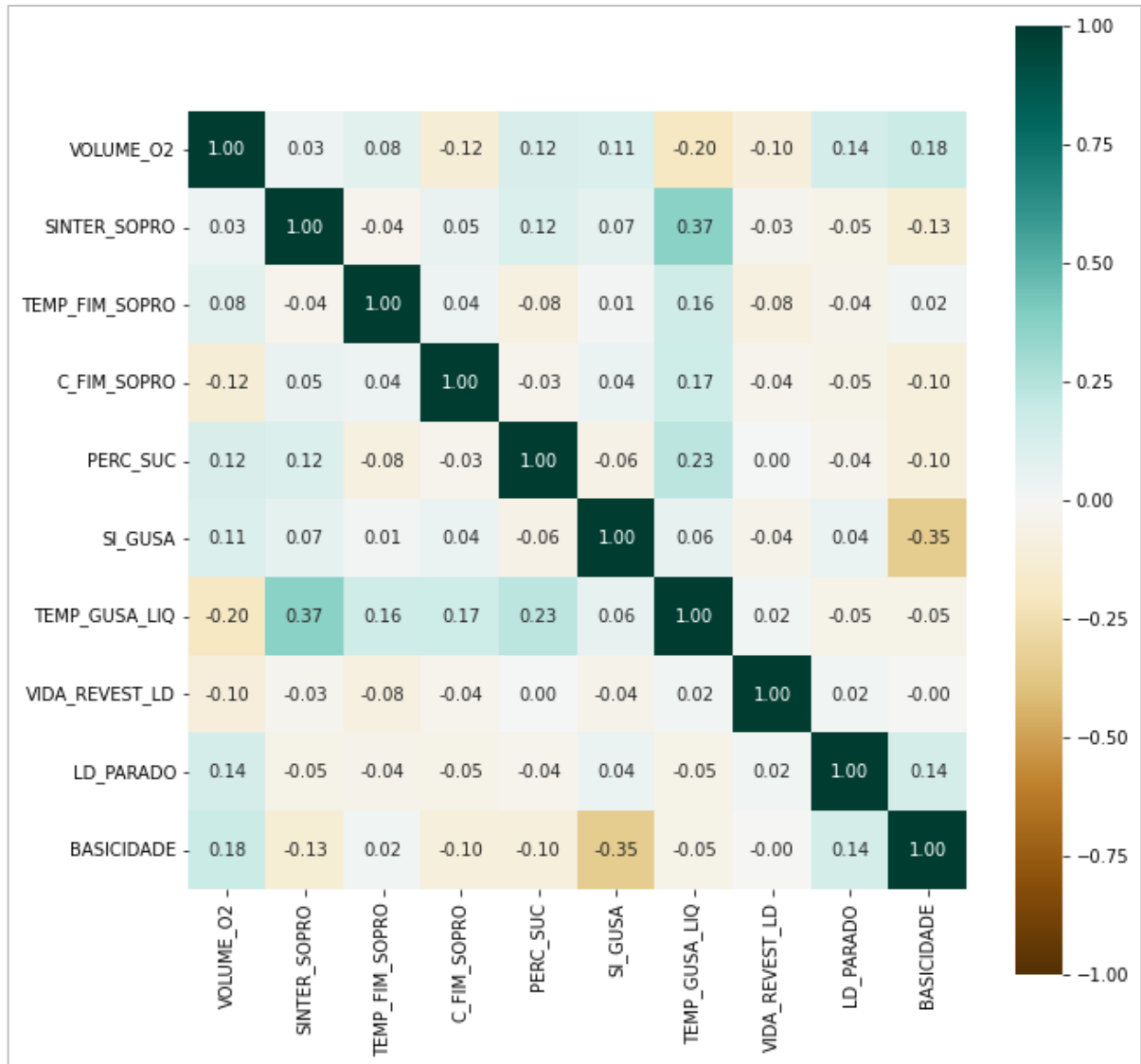
3.4 RNA para predição do volume de oxigênio e peso de sínter

O sínter é um material considerado “refrigerante”, e seu principal objetivo é o controle de temperatura do aço líquido, portanto, dependendo das condições térmicas da corrida, pode-se ou não adicionar o sínter no processo.

Com relação ao Oxigênio, é a partir do sopro de oxigênio puro dentro do convertedor, que todas as reações químicas do processo de produção do aço acontecem, destacando a descarbonação (redução do teor de carbono), dessiliciação (redução do teor de silício) e desfosforação (redução do teor de fósforo), além de promover aumento de temperatura do processo, uma vez que algumas dessas reações químicas são consideradas exotérmicas.

Com as variáveis da RNA definidas, foi feito um *heatmap* para verificar a correlação das variáveis de entrada com as variáveis de saída, conforme é apresentado na Figura 17.

Figura 17 - *Heatmap* das variáveis do modelo de rede neural.



Fonte: Autor (2023).

Pelo gráfico acima, percebe-se que a variável TEMP_GUSA_LIQ (temperatura do ferro gusa medida na aciaria) é a que possui maior correlação com a variável PESO_SINTER (variável de saída), com um coeficiente de correlação de 0,37, seguida pela variável BASICIDADE com um coeficiente de correlação de -0,13.

Com relação à variável de Volume de Oxigênio, a variável TEMP_GUSA_LIQ também é a que possui um maior coeficiente de correlação, sendo de -0,20, seguida da variável BASICIDADE, com um coeficiente de correlação de 0,18. Além da correlação com a variável de saída, é

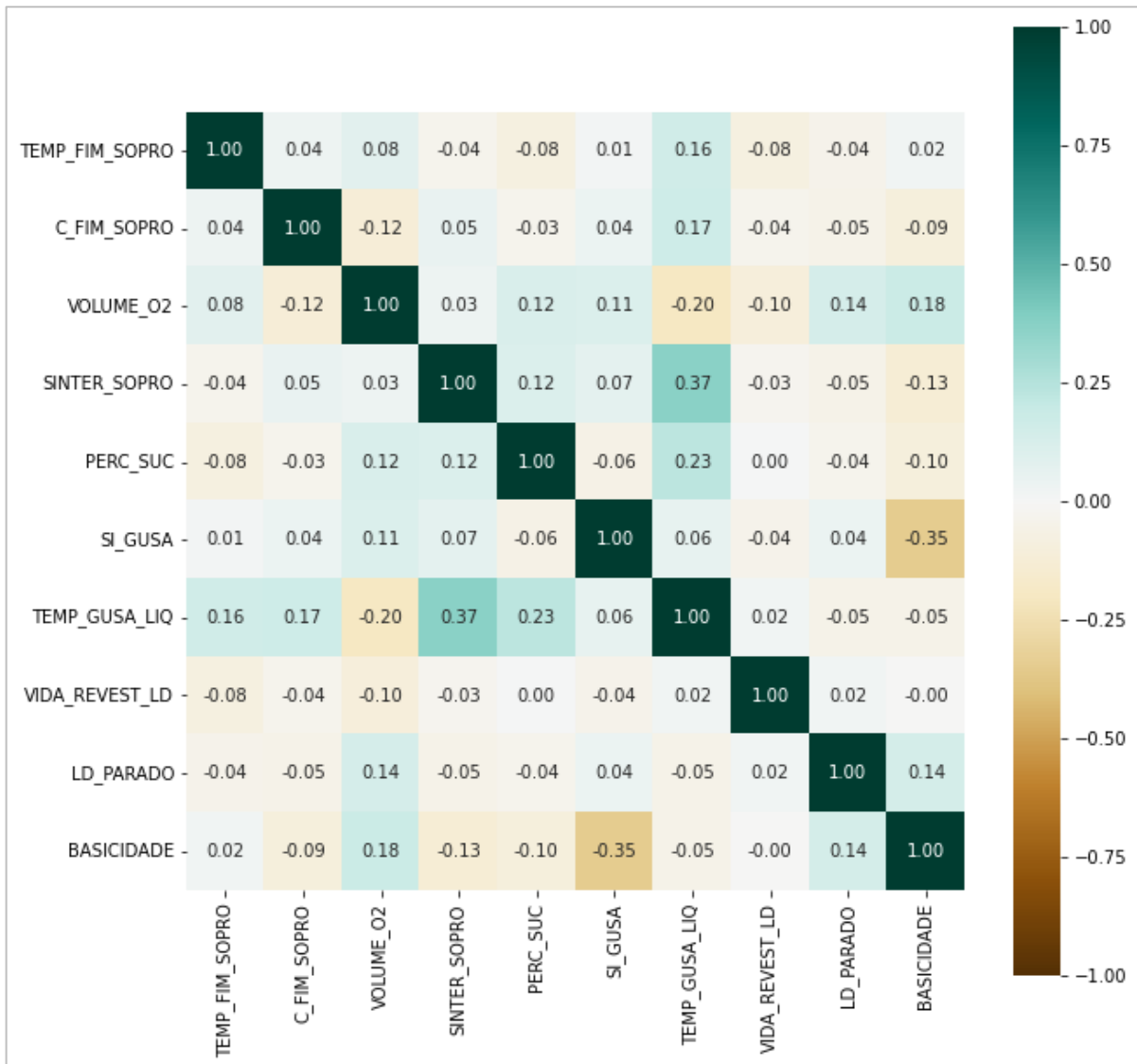
importante verificar também a correlação entre as variáveis de entrada, pois em caso de ter duas variáveis de entrada com alta correlação entre si, é importante analisar se é necessário manter as duas variáveis na base.

3.5 RNA para predição da temperatura e carbono fim de sopro

Um dos principais objetivos do refino do aço em convertedores a oxigênio, é garantir uma temperatura e um teor de carbono ideais para cada tipo de aço que será produzido, portanto, faz-se necessário ter um acerto dessas duas variáveis ao final da etapa de sopro no convertedor. Para o desenvolvimento da RNA de predição de Temperatura e Carbono fim de sopro.

Com as variáveis da RNA definidas, foi feito um *heatmap* para verificar a correlação das variáveis de entrada com a variável de saída, conforme é apresentado na Figura 18.

Figura 18 - Heatmap das variáveis da RNA de temperatura e carbono fim de sopra.



Fonte: Autor (2023)

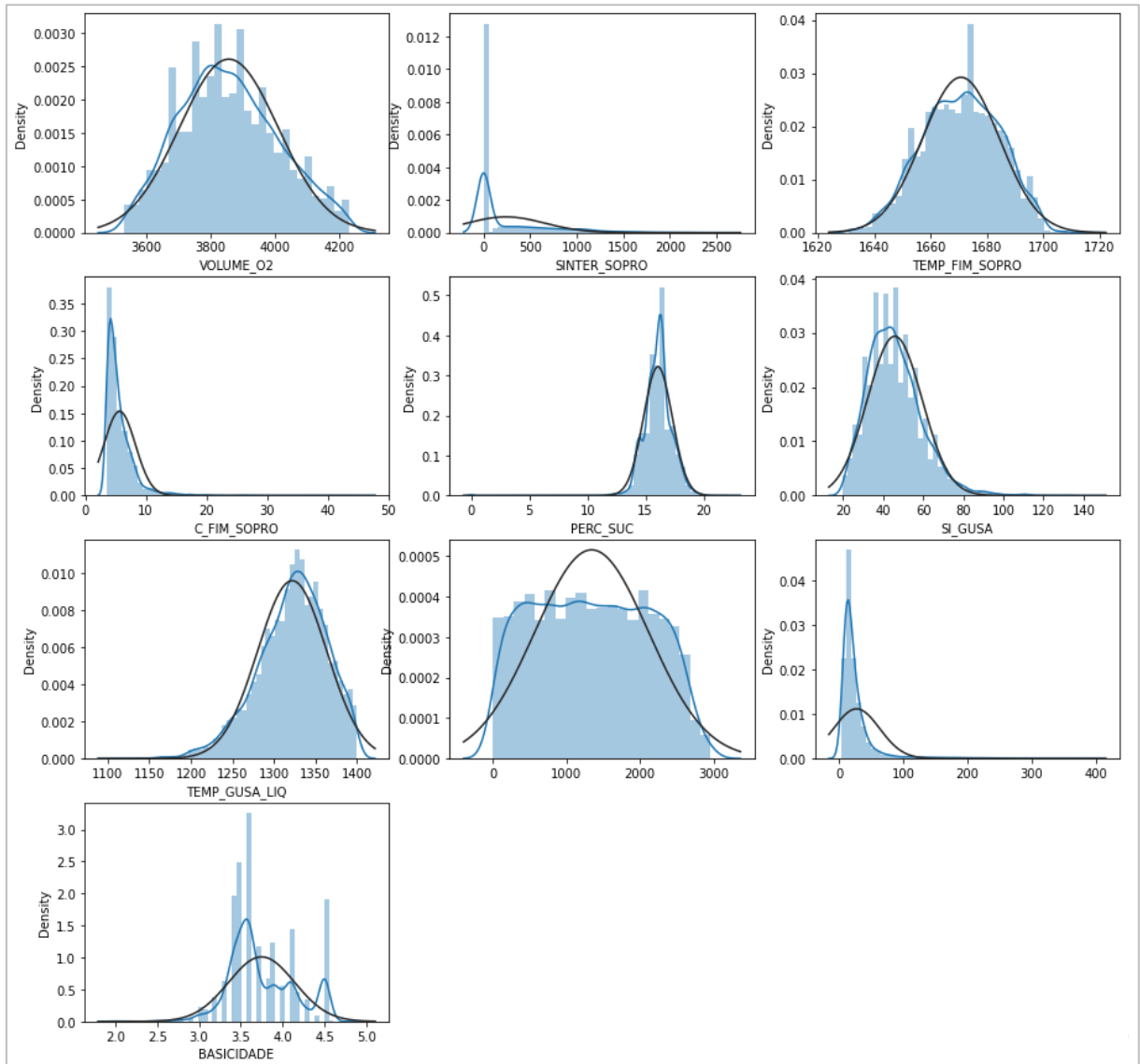
Pelo gráfico acima, percebe-se que a variável com maior correlação com a variável TEMP_FIM_SOPRO também possuem maior correlação com a variável C_FIM_SOPRO, que é a variável TEMP_GUSA_LIQ. Conforme já citado acima, a variável TEMP_GUSA_LIQ se mostrou importante nas duas RNAs, por ser a variável com maior coeficiente de correlação com as variáveis de saída das duas RNAs desenvolvidas.

3.6 Normalização dos Dados

Em projetos de predição de dados, sejam eles utilizando regressões, redes neurais ou outros métodos de predição, é importante normalizar os dados que serão utilizados nos modelos de predição. A normalização elimina qualquer influência que a proporção de valores das variáveis de entrada possa exercer na variável de saída, colocando todos os valores dentro da mesma proporção.

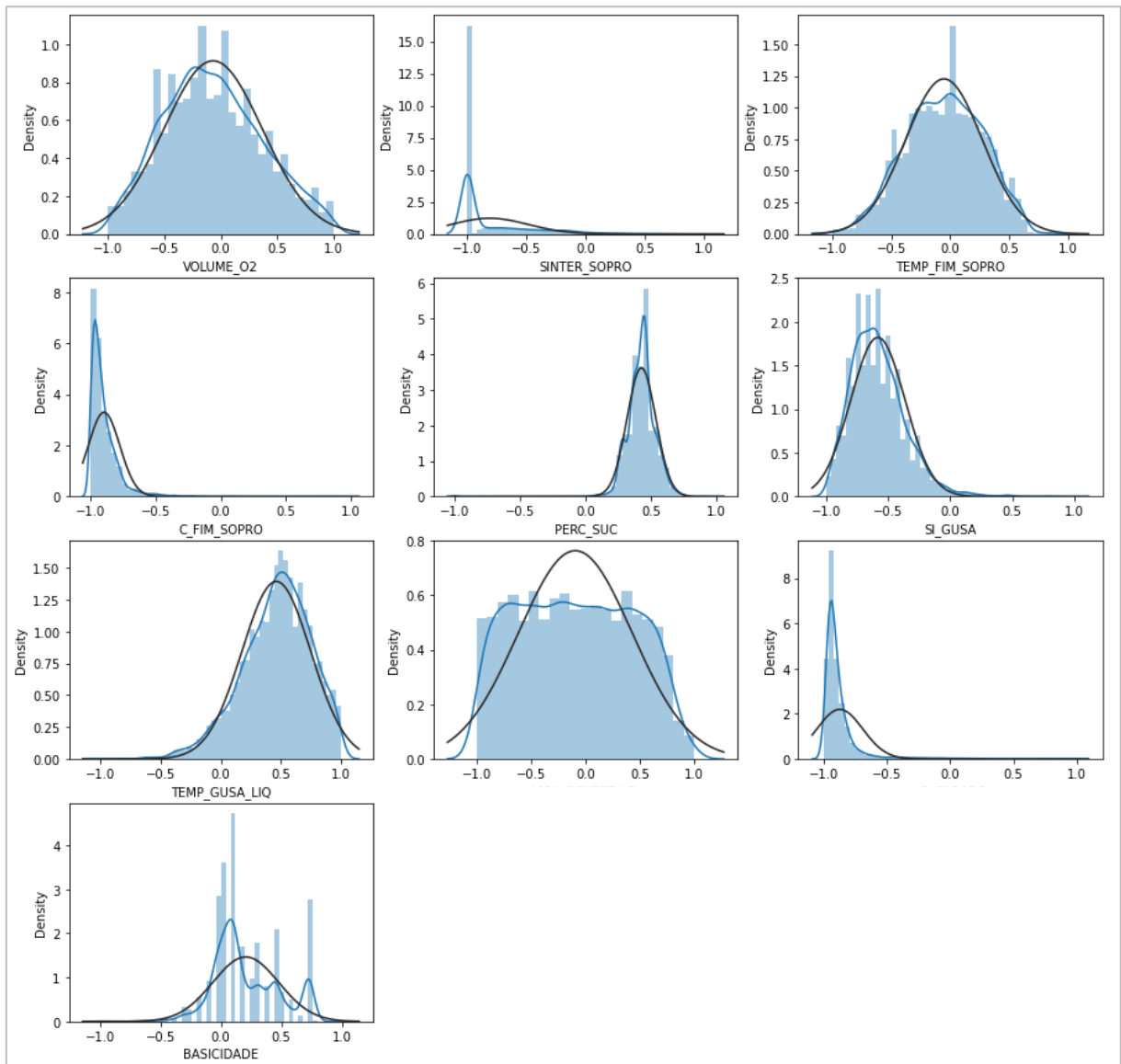
Neste trabalho, utilizou-se um modelo de normalização que colocou os dados no intervalo de $[-1; 1]$ para todas as variáveis de entrada e saída, e um outro modelo no qual a média dos dados de cada variável receba o valor zero e o desvio padrão receba o valor 1, sendo que o primeiro método, com intervalo $[-1; 1]$ se mostrou mais interessante em termos de resultados. As Figuras 19 e 20 apresentam os dados na escala normal e na escala normalizada com o intervalo de $[-1; 1]$.

Figura 19 - Variáveis com dados na escala normal.



Fonte: Autor (2023).

Figura 20 - Dados das variáveis na escala normalizada [-1;1].



Fonte: Autor (2023).

3.7 Divisão da base de dados

Uma parte importante de um projeto de RNA é a divisão da base de dados em treinamento e validação. É nesta etapa que é definido qual será o tamanho da base (número de observações) que a rede terá disponível para fazer o treinamento e qual o tamanho da base que a rede terá disponível para fazer a validação dos resultados. Usualmente, utiliza-se cerca de 70% a 80% para treinamento e 30% a 20% para validação. É importante que a base de treinamento não fique com poucas observações, pois senão pode acontecer da rede não conseguir identificar os

padrões dos dados, e com isso, não aprender a regras para prever a variável de saída. Para este trabalho, foi utilizada a proporção de 80% | 20% para treinamento e validação, respectivamente, resultando em 5.185 observações para o treinamento da rede e 1.297 observações para a validação.

3.8 Topologia das RNAs

Para determinar a topologia mais adequada para cada rede neural desenvolvida, foram feitos diversos testes com diferentes topologia, com o objetivo de encontrar a topologia que melhor se adequa aos dados das variáveis. Não existe, até o momento, uma metodologia que indique qual é o número ótimo de camadas que uma RNA deve ter, bem como o número de neurônios em cada camada. Se essa definição de topologia for feita de forma manual, um caminho a ser seguido seria iniciar a rede com apenas uma camada, e ir variando a quantidade de neurônios dessa camada. Numa segunda etapa, pode-se inserir uma segunda camada, fixando o melhor resultado obtido na fase anterior, e variar apenas os neurônios da segunda camada, mesmo assim, a definição seria bem empírica.

Uma forma de se obter resultados mais assertivos é utilizar um algoritmo para fazer esses testes de diferentes camadas e diferentes quantidades de neurônios, dentro de um intervalo previamente definido. Diante de tantas possibilidades, não há garantia de se ter um resultado ótimo, porém, dependendo da quantidade de topologias diferentes testadas pelo algoritmo, pode-se obter resultados interessantes.

Neste trabalho utilizou-se a ferramenta Keras Tuner, disponível na biblioteca Keras. O Keras Tuner é uma ferramenta de ajuste de hiper parâmetros, na qual é possível parametrizar em intervalos os hiper parâmetros de uma rede neural, e a partir daí, realizar diversos testes, com diferentes estruturas na busca do melhor resultado.

Com relação a quantidade de camadas intermediárias ou camadas densas, foram feitos testes variando de duas a cinco camadas, sendo que os melhores resultados ocorreram com três camadas intermediárias.

Camada densa é um tipo de camada em redes neurais artificiais que realiza uma transformação linear dos dados de entrada e, em seguida, aplica uma função de ativação não-linear. Ela é comumente utilizada em modelos de aprendizado profundo (*deep learning*), como redes neurais

multicamadas, para aprender representações cada vez mais complexas dos dados de entrada. Nas camadas intermediárias, definimos alguns parâmetros que irão auxiliar no treinamento e validação da RNA, como a inserção de camadas de *Dropout*, definição das funções de ativação, inicialização dos pesos etc. (Goodfellow *et al.*, 2016).

Percebeu-se que, quanto mais camadas intermediárias eram adicionadas na rede, o resultado tendia para um *overfitting*. Além de gerenciar a quantidade de camadas intermediárias e quantidade de neurônios, uma outra forma de minimizar o *overfitting* é inserindo camadas de *dropout* entre as camadas ocultas. A camada de *dropout* tem como objetivo eliminar alguns neurônios durante o treinamento da rede neural, de modo que a dependência seja reduzida e a rede consiga generalizar melhor. O principal parâmetro da camada de Dropout é a taxa de redução de neurônios que se deseja aplicar.

Com relação à quantidade de neurônios nas camadas, o Quadro 3 apresenta como foi feita a definição do intervalo mínimo e máximo e o *step* (valor de salto em cada iteração do algoritmo):

Quadro 3 - Configuração dos parâmetros do Keras Tuner.

Camada	Intervalo
Camada Densa 1	Mín: 28 Max: 128 Step: 10
Dropout 1	Min: 0 Max: 0,5 Step: 0,25
Camada Densa 2	Mín: 0 Max: 128 Step: 10
Dropout 2	Min: 0 Max: 0,5 Step: 0,25
Camada Densa 3	Mín: 0 Max: 128 Step: 10
Dropout 3	Min: 0 Max: 0,5 Step: 0,25
Saída	2

Fonte: Autor (2023).

Além de testar a quantidade de camadas ocultas e a quantidade de neurônios nas camadas, também foram testadas quais funções de ativação deveriam ser definidas nas camadas ocultas, sendo disponibilizadas para a ferramenta do Keras Tuner as seguintes funções de ativação: **Relu**, **Sigmoid**, **Tangente hiperbólica (tanh)**, **Elu**, **Selu**. Para as camadas de entrada e saída foram empregadas as funções **Relu** e **Linear**, respectivamente.

Segundo Nwankpa *et al.* (2018) a função de ativação Relu (*Rectified Linear Unit*) é uma função não linear amplamente utilizada em redes neurais profundas. Ela é definida como $f(x) = \max(0, x)$, onde x é a entrada para a função. Em outras palavras, a função retorna a entrada se ela for positiva, e zero se for negativa.

De acordo com Goodfellow *et al.* (2016), a função de ativação sigmoid é uma função não linear que mapeia qualquer valor real em um intervalo entre 0 e 1. Já a função de ativação Tanh, também é uma função não linear, porém ela mapeia qualquer valor real entre o intervalo -1 e 1.

A função de ativação ELU (*Exponential Linear Unit*) é uma função não linear que tem uma suavização no ponto zero e tende a ter um melhor desempenho do que a função Relu. E por fim, a função de ativação SELU (*Scaled Exponential Linear Unit*) é uma variação da função ELU e tem uma propriedade de autorregulação, que pode ajudar a evitar o problema da *vanishing/exploding gradients* em redes neurais profundas (Klambauer *et al.*, 2017).

Já a função de ativação Linear, segundo Goodfellow *et al.* (2016), é uma função simples e linear que transforma uma saída linearmente sem modificar a escala e amplitude da entrada.

Também foram testados diferentes otimizadores para compilação do modelo de RNA, sendo eles: **Adam, Adamax, Nadam, RMSprop e SGD.**

Adam, Adamax, Nadam, RMSprop e SGD são algoritmos de otimização usados em aprendizado de máquina e treinamento de redes neurais. Aqui estão as diferenças entre eles:

- **SGD (Stochastic Gradient Descent):** é um dos algoritmos de otimização mais simples e amplamente utilizados. Ele atualiza os pesos do modelo em cada iteração do treinamento, seguindo a direção do gradiente descendente da função de perda. SGD é eficiente para dados de grande escala e conjuntos de dados com muitas amostras, mas pode ter dificuldade em convergir para mínimos locais e pode ser sensível à escolha da taxa de aprendizado. (Goodfellow *et al.*, 2016).
- **RMSprop (Root Mean Square Propagation):** é um algoritmo de otimização adaptativo que ajusta a taxa de aprendizado de forma diferente para cada parâmetro do modelo com base no gradiente médio dos últimos passos de treinamento. Isso ajuda a suavizar o caminho de atualização do gradiente e pode levar a uma convergência mais rápida. No entanto, pode ser sensível à escolha da taxa de aprendizado e pode oscilar em torno do mínimo global da função de perda (Hinton, 2012).

- **Adam (Adaptive Moment Estimation):** é um algoritmo de otimização adaptativo que calcula uma estimativa adaptativa do momento de primeira e segunda ordem do gradiente para atualizar os pesos do modelo. Ele combina as vantagens do RMSprop e do método do momento estocástico (SGD com momento) para convergir rapidamente e lidar com gradientes esparsos. Adam é amplamente utilizado em muitos modelos de aprendizado de máquina devido à sua eficácia e eficiência em termos de tempo de processamento (Goodfellow *et al.*, 2016).
- **Adamax:** é uma variação do Adam que usa uma estimativa infinita para o segundo momento do gradiente. Isso ajuda a lidar com gradientes esparsos e pode ser mais estável do que o Adam em algumas situações (Goodfellow *et al.*, 2016).
- **Nadam (Nesterov-accelerated Adaptive Moment Estimation):** é uma variação do Adam que incorpora o método de aceleração de Nesterov. Ele usa a estimativa do momento de primeira ordem para calcular o gradiente em vez de usar o gradiente atual. Nadam pode ser mais rápido do que Adam em muitos casos e pode ter melhor convergência em problemas comuns de aprendizado profundo (Dozat, 2016).

Em resumo, SGD é simples e amplamente utilizado, RMSprop é uma variante adaptativa simples que suaviza o caminho de atualização do gradiente, Adam é uma combinação eficaz de RMSprop e SGD com momento, Adamax é uma variante mais estável do Adam, e Nadam é uma variante mais rápida e eficaz do Adam que incorpora o método de aceleração de Nesterov. A escolha do algoritmo de otimização depende do problema em questão e das características do conjunto de dados.

Com relação à métrica de avaliação do modelo e função de custo, foram definidas métricas e funções de custo relacionadas a problemas de predição. A função de custo definida foi a *Mean Square Error* (MSE) e a métrica de avaliação do modelo foi definida a *Mean Absolute Error* (MAE).

Segundo James *et al.* (2013), O MAE é uma métrica de avaliação de modelo comum em problemas de regressão. É calculado como a média das diferenças absolutas entre as previsões do modelo e os valores verdadeiros dos dados de teste.

A Equação 1 apresenta o cálculo do MAE:

$$MAE = \frac{1}{n} \sum |y_{pred} - y_{true}| \quad (1)$$

Onde:

y_{pred} é o valor previsto pelo modelo

y_{true} é o valor verdadeiro dos dados de teste

n é o número total de amostras nos dados de teste

\sum é o somatório de todas as diferenças absolutas entre as previsões e os valores verdadeiros.

De acordo com James *et al.* (2013), o MSE é uma métrica de avaliação de modelo comum em problemas de regressão. É calculado como a média das diferenças ao quadrado entre as previsões do modelo e os valores verdadeiros dos dados de teste.

A Equação 2 apresenta o cálculo do MSE:

$$MSE = \frac{1}{n} \sum (y_{pred} - y_{true})^2 \quad (2)$$

Onde:

y_{pred} é o valor previsto pelo modelo

y_{true} é o valor verdadeiro dos dados de teste

n é o número total de amostras nos dados de teste

\sum é o somatório de todas as diferenças ao quadrado entre as previsões e os valores verdadeiros.

O MSE tem a vantagem de penalizar mais fortemente os erros maiores, mas pode ser sensível a valores discrepantes.

Por fim, utilizou-se o algoritmo *Random Search* para fazer a busca da melhor estrutura da rede, definindo como objetivo do algoritmo, buscar a estrutura que retornasse o melhor valor de MAE dos dados de validação do modelo.

O *Random Search* é um método de otimização que envolve a geração aleatória de soluções em um espaço de busca para encontrar a melhor solução para um determinado problema. Esse método é simples, fácil de implementar e geralmente é usado como um método de referência em comparação com algoritmos de otimização mais avançados, como algoritmos genéticos ou algoritmos de otimização baseados em gradiente (Spall, 1992).

O algoritmo buscou 15 configurações de redes diferentes, sendo que as melhores configurações para a rede de síntese, oxigênio, temperatura e carbono estão indicadas na Tabela 1.

Tabela 1 - Melhor configuração para as RNAs desenvolvidas.

RNA	Quantidade de Camadas	Quantidade de Neurônios	Funções de Ativação	Taxa de Dropout	Otimizador
Volume de Oxigênio e Peso de Síntese	1 Entrada	97	Relu	0,05	Adam
	2 Ocultas	50	Elu	0,30	
	1 Saída	20	Relu	0,00	
Temperatura e Carbono Fim de Sopro	1 Saída	2	Linear	-	RMSprop
	1 Entrada	72	Relu	0,20	
	2 Ocultas	60	Sigmoid	0,15	
	1 Saída	110	Selu	0,25	
	1 Saída	2	Linear	-	

Fonte: Autor (2023).

3.9 Otimizador e compilação dos modelos de RNAs

A definição de um otimizador é necessária para compilar um modelo de rede neural utilizando as ferramentas da biblioteca do Keras. Com o auxílio do Keras Tuner foi definido o otimizador Adam para a RNA de volume de oxigênio e peso de síntese e o otimizador RMSprop para a RNA de temperatura e carbono fim de sopro conforme foi apresentado na Tabela 1.

Um dos principais parâmetros dos otimizadores é o *learning rate*. Segundo Goodfellow *et al.* (2016) o *learning rate* é um hiper parâmetro usado em algoritmos de aprendizado de máquina, como redes neurais, que controla a magnitude da atualização dos pesos durante o treinamento.

Uma taxa de aprendizado alta pode levar a atualizações muito grandes dos pesos, o que pode levar à instabilidade do treinamento, enquanto uma taxa de aprendizado muito baixa pode levar a um treinamento lento e a um risco de ficar preso em mínimos locais. Neste trabalho, utilizou-se o valor para o *learning rate* de 0,001 para cada otimizador, por ser um valor comumente usado como um valor inicial para muitos algoritmos de otimização, sendo um valor seguro e conservador para começar a ajustar outros hiperparâmetros do modelo.

3.10 Treinamento e Validação das RNAs

O treinamento de uma rede neural é a parte final para que se valide a eficiência da rede. É no treinamento que o algoritmo escolhido irá “estudar” a base de dados definida para o treinamento e “aprender” o comportamento desses dados de forma que ele seja capaz de criar regras que vão minimizar o erro na variável de saída do modelo. Neste trabalho, o algoritmo de treinamento utilizado foi o *Backpropagation*.

3.10.1 Épocas de Treinamento

Um dos parâmetros que deve ser definido antes de iniciar o treinamento da rede, é a quantidade de épocas que a rede vai treinar. O processo de treinamento da rede acontece a cada volta ou a cada época da base de dados. Durante a época, o processo de atualização dos pesos nas camadas de neurônios e a minimização de erro na camada de saída são repetidos.

Depois de rodar a quantidade de épocas definidas, se tem um modelo treinado, isto é, que aprendeu as características dos dados apresentados. Portanto, ao exibir uma nova informação que esteja dentro da mesma classe de dados que o modelo foi treinado, espera-se que o modelo seja capaz de prever a saída de maneira precisa.

É importante ressaltar que, definir um número de épocas relativamente alto pode gerar lentidão desnecessária no modelo, uma vez que os pesos já não estão mais atualizados. Para o

treinamento das RNAs desenvolvidas neste trabalho, definiu-se a quantidade de 10.000 épocas para treinamento, e como critério de parada, utilizou-se o algoritmo *EarlyStopping* disponível na biblioteca do Keras. No algoritmo foi definido o monitoramento do valor de MAE da base de validação dos dados, ou seja, caso o algoritmo perceba que o valor de MAE não está diminuindo, ele interromperá o treinamento da rede e armazenará o melhor resultado até o momento da interrupção.

3.10.2 *Batch Size*

O *batch size*, ou, tamanho do lote, é um termo usado em *machine learning* e refere-se ao tamanho lote de dados que será usado no treinamento. O tamanho do lote pode ser definido de três maneiras:

1. ***Batch mode***: onde o tamanho do lote é igual ao conjunto de dados total, tornando os valores de iteração e épocas equivalentes;
2. ***Mini Batch mode***: onde o tamanho do lote é maior do que um, mas menor que o tamanho total do conjunto de dados. Geralmente, um número que pode ser dividido no tamanho total do conjunto de dados;
3. ***Stochastic mode***: onde o tamanho do lote é igual a um. Portanto, o gradiente e os parâmetros da rede neural são atualizados após cada amostra.

O tamanho do lote é um dos hiper parâmetros mais importantes para sintonizar os modelos de *machine learning*. Igual aos demais parâmetros, não existe um número ideal para definir o *batch size*, portanto, deve-se testar diferentes valores e verificar o comportamento do modelo durante o treinamento. O *batch size* utilizado nas RNAs desenvolvidas neste trabalho foi 64.

4 Resultados e Discussão

Este Capítulo apresenta os resultados obtidos com base na Metodologia descrita no Capítulo 3 e uma discussão sobre esses resultados, apresentando os ganhos e os pontos de melhorias identificados. É importante ressaltar que o acerto do modelo de RNA desenvolvido será mensurado pelo percentual de acerto simultâneo, que é quando ocorre o acerto da temperatura fim de sopro e do carbono fim de sopro da corrida.

Conforme descrito na Metodologia, desenvolveu-se neste trabalho duas RNAs, sendo a primeira para prever o volume de oxigênio que será soprado e o peso de síter que será adicionado na corrida. Posteriormente, os resultados obtidos na primeira RNA são usados como dados de entrada na segunda RNA para prever a temperatura e carbono fim de sopro.

Devido ao alto tempo para treinar as redes e validar os resultados, optou-se por testar a segunda RNA somente após alcançar um bom resultado na primeira RNA, utilizando como parâmetro de qualidade da RNA, o MAE encontrado na validação do treinamento.

4.1 Resultados do treinamento e validação da RNA de volume de oxigênio e peso de síter

Conforme já mencionado, a primeira RNA desenvolvida neste trabalho foi uma RNA para prever o volume de oxigênio que será soprado e o peso de síter que será adicionado na corrida. A melhor topologia encontrada foi 8 x 97 x 50 x 20 x 2 (8 variáveis de entrada, 3 camadas intermediárias com 97, 50 e 20 neurônios, respectivamente, e 1 camada com 2 saídas), utilizando as funções de ativação Relu, Elu, Relu e Linear, respectivamente, com o otimizador Adam, e camadas de *Dropout* de 0,05 x 0,30 x 0,00, porém foram testadas diversas topologias diferentes com o auxílio do Keras Tuner, conforme é apresentado na Tabela 2.

6.	80% 20%	Random Search	8 96 110 60 2	relu tanh elu linear	0.1 0.2 0.45	RMSprop	0.0005	MSE	MAE	10000	64	0.1313	0.2737
7.	80% 20%	Random Search	8 56 80 120 2	relu relu tanh linear	0.15 0.05 0.1	RMSprop	0.001	MSE	MAE	10000	64	0.1269	0.2684
8.	80% 20%	Random Search	8 97 50 20 2	relu elu relu linear	0.05 0.30 0.0	Adam	0.001	MSE	MAE	10000	65	0.1094	0.2547

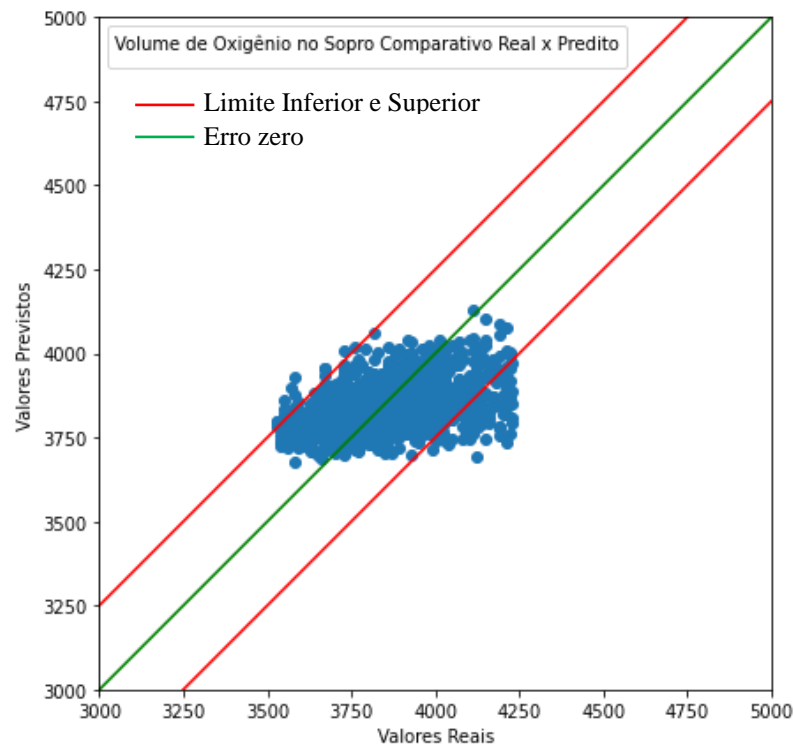
Fonte: Autor (2023).

Conforme resultados da Tabela 2, a simulação “8” foi a que resultou em um menor valor de MAE. É importante destacar que esse valor de MAE é resultante da base normalizada na escala de $[-1;1]$, portanto, é difícil mensurar o erro médio absoluto nessa escala. Para resolver esse problema, é necessário reverter a normalização após o treinamento e validação dos dados, de forma que os dados voltem para o valor original.

4.1.1 Resultados da variável volume de oxigênio

A Figura 21 apresenta o resultado do treinamento e validação da RNA de volume de oxigênio e peso de sínter, considerando apenas a variável de saída Volume de Oxigênio. É possível verificar que a maioria das previsões ficou dentro dos limites superior e inferior da variável Volume de Oxigênio. Os limites definidos foram de $\pm 250 \text{ Nm}^3$, com base no trabalho de Cunha (2001), uma vez que a Aciaria 1 da Usiminas não tem um valor padrão para os limites da variável de volume de oxigênio.

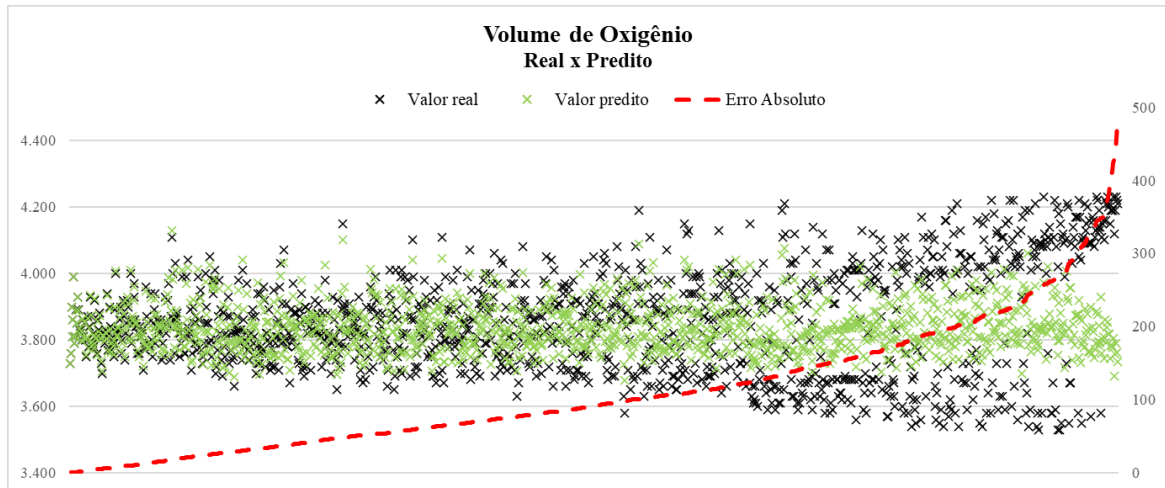
Figura 21 - Resultado do treinamento e validação da RNA de volume de oxigênio e peso de sínter, considerando apenas a saída Volume de Oxigênio.



Fonte: Autor (2023).

Outra forma de analisar o comparativo do Real x Predito da variável de Volume de Oxigênio é ordenar o erro absoluto do menor para o maior valor, para verificar em quais situações que o erro aumenta. A Figura 22 apresenta essa análise.

Figura 22 - Análise do valor real x predito da variável Volume de Oxigênio.

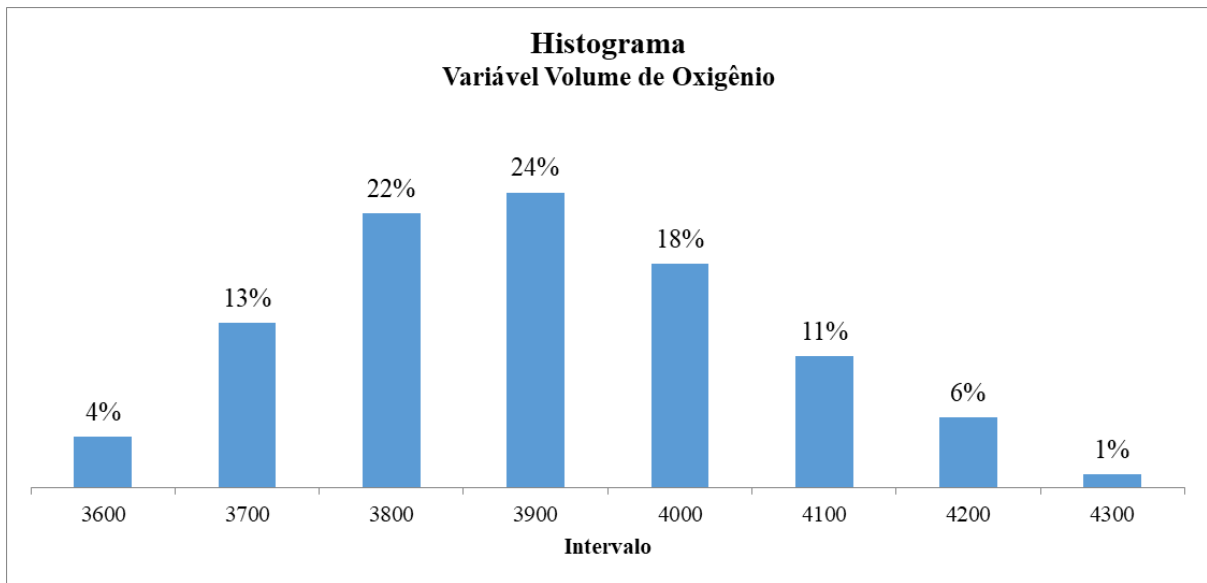


Fonte: Autor (2023).

É possível verificar na Figura 22, que o erro entre o valor real x predito aumenta consideravelmente quando o valor real da variável Volume de Oxigênio é menor do que aproximadamente 3600 Nm³ e maior do que aproximadamente 4100 Nm³.

Tal comportamento do erro absoluto da variável Volume de Oxigênio justifica-se pelo fato de ter poucas observações na faixa de valores onde o erro foi maior. A Figura 23 apresenta um histograma da variável Volume de Oxigênio, mostrando o percentual de amostras em cada faixa de valor.

Figura 23 – Histograma do número de amostras da variável Volume de Oxigênio.



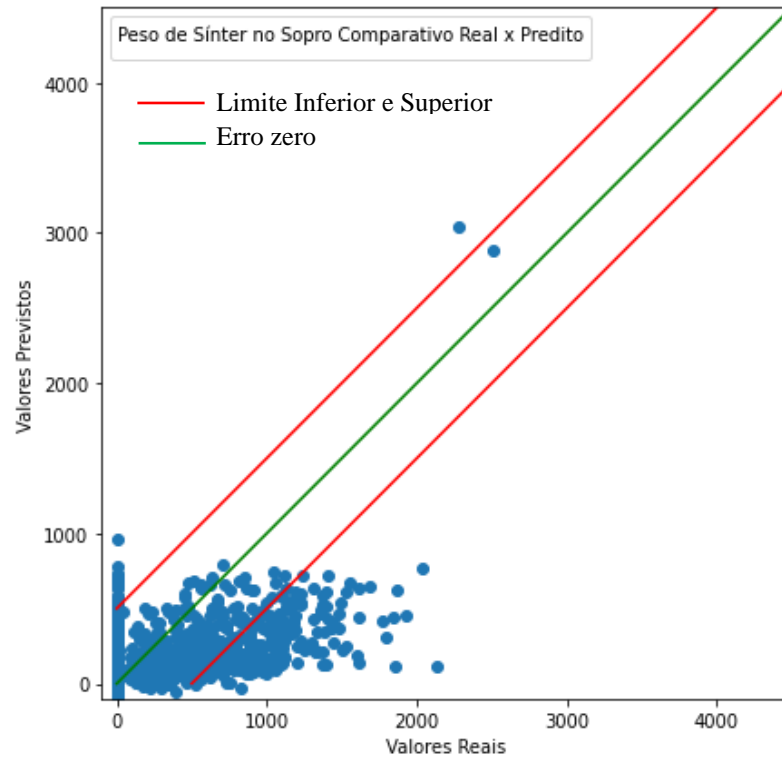
Fonte: Autor (2023).

Percebe que 4% dos valores estão na faixa de até 3600 Nm³ e 7% dos valores estão na faixa acima de 4100 Nm³. Uma sugestão para tentar reduzir o erro nas faixas minoritárias, é utilizar alguma técnica de balanceamento da base de dados na variável de saída, de forma que a distribuição dos valores fique mais uniforme, porém neste trabalho optou-se por manter a originalidade da base de dados, uma vez que esse erro nas faixas minoritárias foi aceitável por ter uma baixa frequência de acontecimentos.

4.1.2 Resultados da variável peso de sínter

A Figura 24 apresenta o resultado do treinamento e validação da RNA de volume de oxigênio e peso de sínter, considerando apenas a saída Peso de Sínter. É possível verificar que a maioria das predições ficou dentro dos limites superior e inferior da variável Peso de Sínter. Os limites definidos foram de ± 500 kg, com base no trabalho de Cunha (2001), uma vez que a Aciaria 1 da Usiminas não tem um valor padrão para os limites da variável peso de sínter.

Figura 24 - Resultado do treinamento e validação da RNA de volume de oxigênio e peso de sínter, considerando apenas a saída Peso de Sínter.

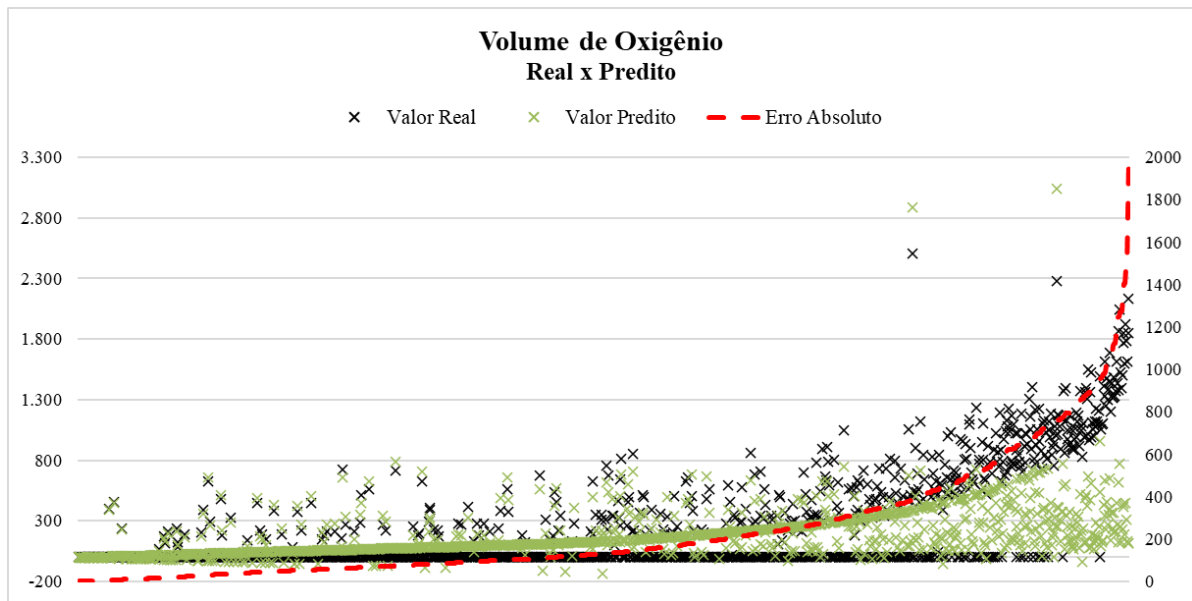


Fonte: Autor (2023).

Analogamente a análise realizada com a variável Volume de Oxigênio, pode-se comparar os valores Real x Predito da variável Peso de Sínter e ordenar o erro absoluto do menor para o maior valor, para verificar em quais situações que o erro aumenta.

A Figura 25 apresenta essa análise. É possível verificar que o erro entre o valor real x predito aumenta consideravelmente quando o valor real da variável Peso de Sínter é maior do que 800 kg.

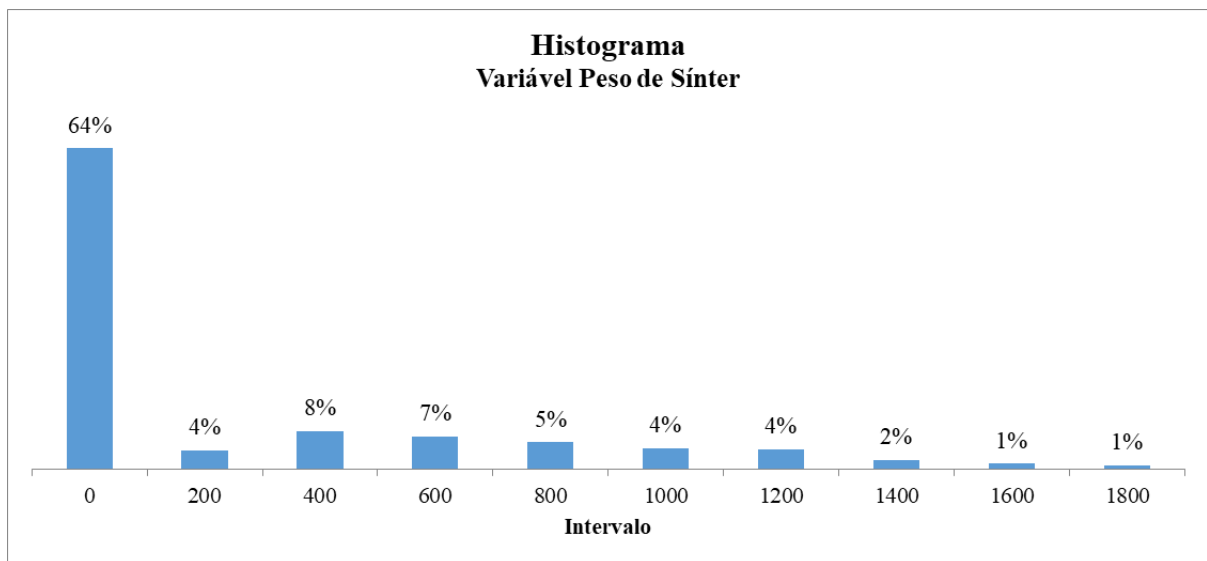
Figura 25 - Análise do valor real x predito da variável Peso de Sínter.



Fonte: Autor (2023).

Tal comportamento do erro absoluto da variável Peso de Sínter justifica-se pelo fato de ter poucas observações na faixa de valores onde o erro foi maior. A Figura 26 apresenta um histograma da variável Volume de Oxigênio, mostrando o percentual de amostras em cada faixa de valor.

Figura 26 - Histograma do número de amostras da variável Peso de Sínter.



Fonte: Autor (2023).

Percebe que 12% dos valores estão acima da faixa de 800 kg. Para a variável do peso de síter já se esperava uma variabilidade maior dos resultados uma vez que a concentração de observações no valor zero (quando não houve adição de síter na corrida) é bem superior às demais faixas.

4.2 Resultados do treinamento e validação da RNA de temperatura e carbono fim de sopro

A segunda RNA desenvolvida neste trabalho foi uma RNA para prever a temperatura e o carbono da corrida ao final da etapa de sopro de oxigênio. A melhor topologia encontrada foi 8 x 72 x 60 x 110 x 2 (8 variáveis de entrada, 3 camadas intermediárias com 72, 60 e 110 neurônios, respectivamente, e 1 camada com 2 saídas), utilizando as funções de ativação Relu, Sigmoid, Selu e Linear, respectivamente, com o otimizador RMSprop, e camadas de *Dropout* de 0,20 x 0,15 x 0,25, porém foram testadas diversas topologias diferentes com o auxílio do Keras Tuner, conforme é apresentado na Tabela 3.

Tabela 3 - Análise de sensibilidade da topologia da RNA de temperatura e carbono fim de sopra.

Nº	Treinamento o Validação	Algoritmo do Keras Tuner	Parâmetros da Rede						Treinamento e Validação da Rede				
			Quantidade de Neurônios	Funções de Ativação	Dropout	Otimize- zador	Learning Rate	Loss	Métrica	Épocas	Batch Size	Val Loss	Val MAE
1.	80% 20%	Random Search	116	relu	0.10	SGD	0.003	MSE	MAE	10000	128	0.0634	0.1840
			110	elu									
			50	relu									
			2	linear									
2.	80% 20%	Random Search	36	relu	0.45	SGD	0.0007	MSE	MAE	10000	64	0.1005	0.2481
			50	relu	0.00								
			50	selu	0.05								
			2	linear									
3.	80% 20%	Random Search	72	relu	0.40	RMSP rop	0.001	MSE	MAE	10000	64	0.1183	0.2165
			40	elu									
			10	tanh									
			2	linear									
4.	80% 20%	Random Search	72	relu	0.20	RMSP rop	0.001	MSE	MAE	10000	64	0.0783	0.1590
			60	sigmoid	0.15								
			110	selu	0.25								
			2	linear									

Fonte: Autor (2023)

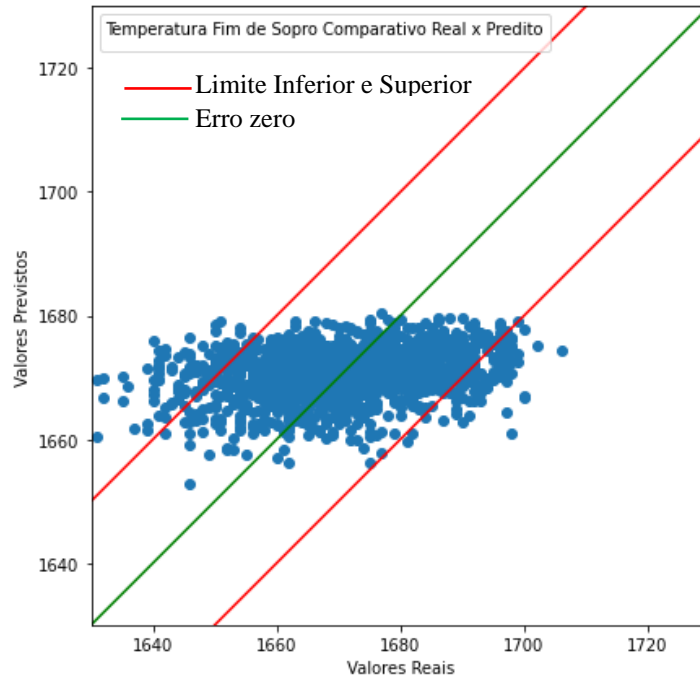
Conforme resultados da Tabela 3, a simulação “4” foi a que resultou em um menor valor de MAE. É importante destacar que esse valor de MAE é resultante da base normalizada na escala de [-1;1], portanto, é difícil mensurar o erro médio absoluto nessa escala. Para resolver esse problema, é necessário reverter a normalização após o treinamento e validação dos dados, de forma que os dados voltem para o valor original.

É difícil definir quando parar de fazer simulações com topologias diferentes de RNAs, porém optou-se por encerrar as simulações de cenários nesses valores, pois os resultados foram considerados satisfatórios, conforme apresentado na Seção 4.3.

4.2.1 Resultados da variável de temperatura fim de sopro

A Figura 27 apresenta o resultado do treinamento e validação da RNA de temperatura e carbono fim de sopro, considerando apenas a saída Temperatura Fim de Sopro. É possível verificar que a maioria das previsões ficou dentro dos limites superior e inferior da variável Temperatura Fim de Sopro. Os limites foram definidos considerando o que é atualmente tolerado no sistema da Aciaria 1, ou seja, se considera que houve acerto da temperatura de fim de sopro se o valor variar $\pm 20^{\circ}\text{C}$ do valor objetivado.

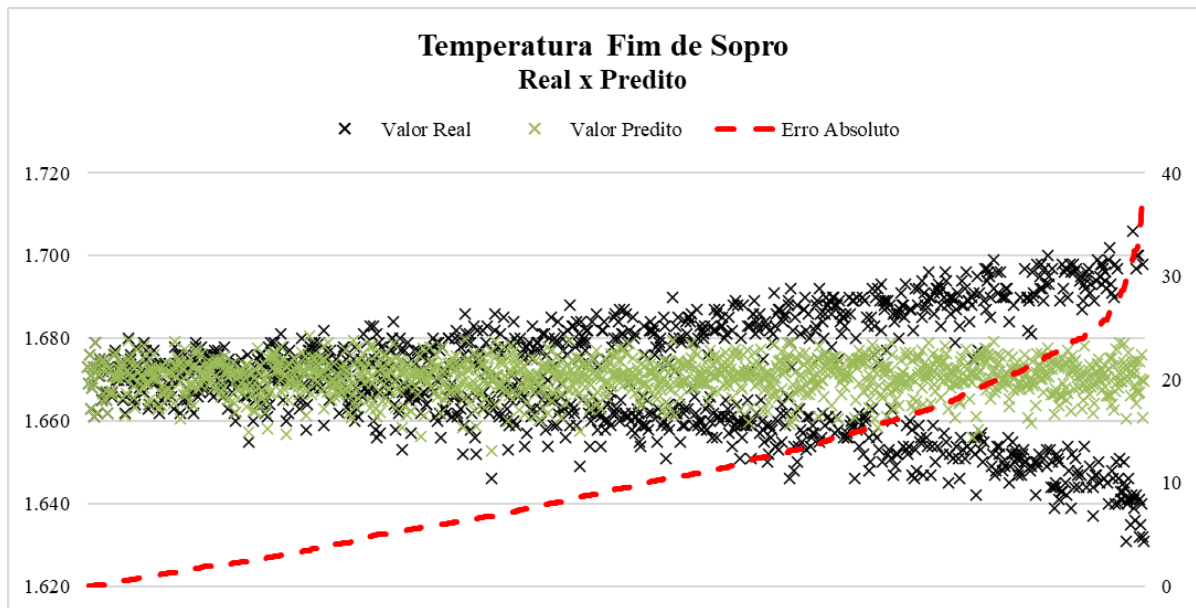
Figura 27 - Resultado do treinamento e validação da RNA de temperatura e carbono fim de sopro, considerando apenas a saída Temperatura Fim de Sopro.



Fonte: Autor (2023).

Complementarmente, analisaram-se quantas previsões ficaram fora dos limites de tolerância. A Figura 28 apresenta a relação entre o valor real x valor predito pela RNA. Verifica-se que o erro entre o valor real x predito aumenta consideravelmente quando o valor real da variável Temperatura Fim de Sopro é menor do que 1650 °C e maior do que 1690 °C.

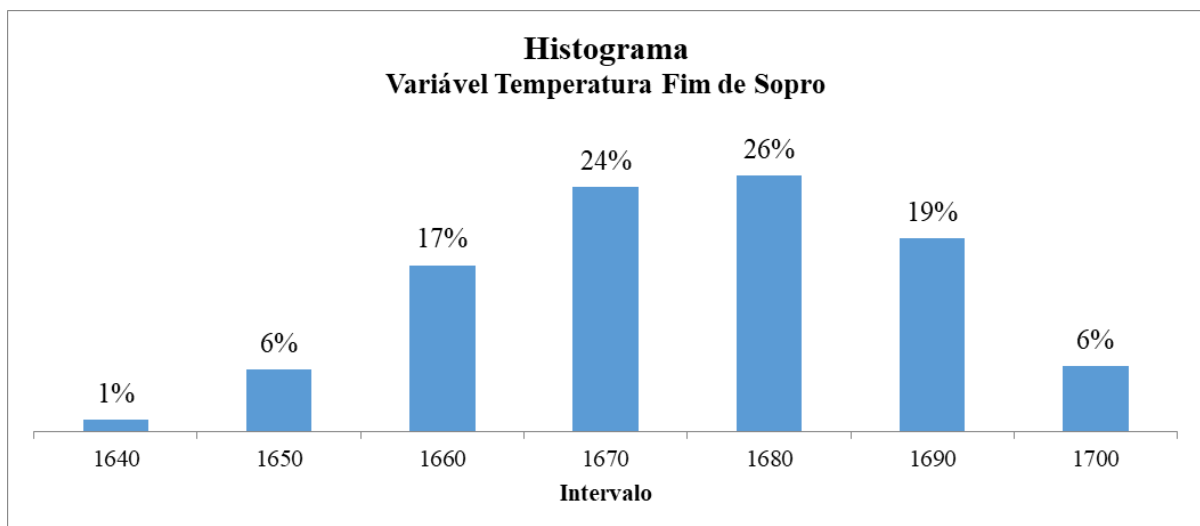
Figura 28 - Análise do valor real x predito da variável Temperatura Fim de Sopro.



Fonte: Autor (2023).

Conforme já verificado nas variáveis de Volume de Oxigênio e Peso de Sínter, o comportamento do erro da variável Temperatura Fim de Sopro nas faixas de valores citadas acima se justifica pela baixa quantidade de dados nessas faixas conforme apresenta a Figura 29.

Figura 29 - Histograma do número de amostras da variável Temperatura Fim de Sopro.



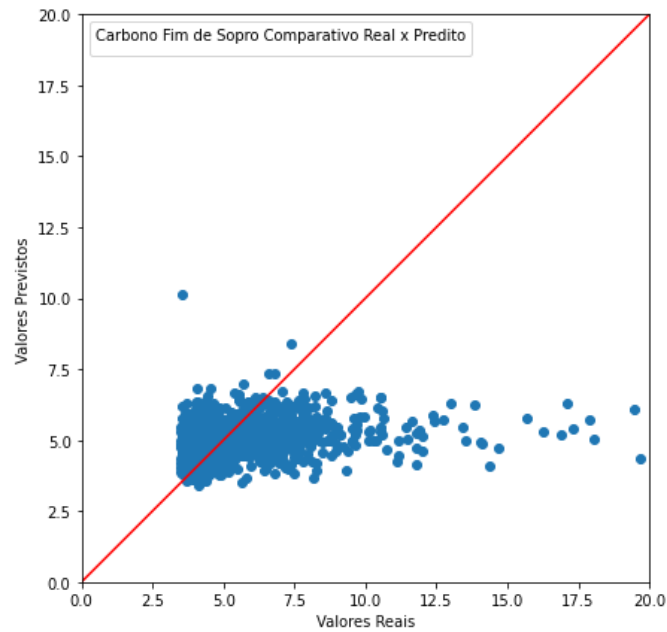
Fonte: Autor (2023).

Conforme citado, é possível verificar na Figura 30 que as faixas de valores minoritários representam apenas 7% da base de treinamento e com isso, a RNA não teve muitas observações para treinar com valores nessas faixas.

4.2.2 Resultados da variável carbono fim de sopro

A Figura 30 apresenta o resultado do treinamento e validação da RNA de temperatura e carbono fim de sopro, considerando apenas a variável de saída Carbono Fim de Sopro. É possível analisar que a RNA não conseguiu aproximar o erro absoluto a zero quando os valores da variável de carbono fim de sopro são maiores. Entretanto, não é necessário que o erro absoluto seja igual à zero, pois nesta variável os limites superior e inferior variam de acordo com o valor objetivado ao final do sopro conforme mostra o Quadro 4.

Figura 30 - Resultado do treinamento e validação da RNA de temperatura e carbono fim de sopro, considerando apenas a variável de saída Carbono Fim de Sopro.



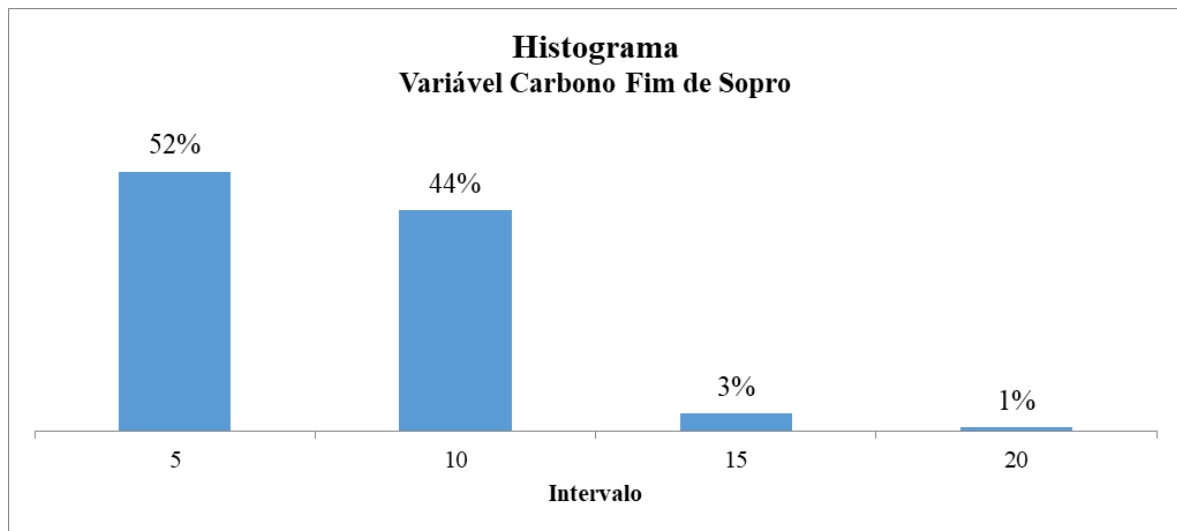
Fonte: Autor (2023).

Quadro 4 - Faixa de acerto da variável Carbono Fim de Sopro.

Faixa de Carbono Objetivado (pts)	Faixa de Acerto (pts)	
≤ 4	-0,5	1,5
$4,1 \leq \text{CFS} \leq 4,4$	-1	3,5
$4,5 \leq \text{CFS} \leq 6,4$	-2	4,5
$6,5 \leq \text{CFS} \leq 8,4$	-3,5	6
$8,5 \leq \text{CFS} \leq 16,4$	-6	8
$\geq 16,5$	-9	9

Fonte: Autor (2023).

Com relação à distribuição de amostras da variável carbono fim de sopro, a Figura 31 apresenta um histograma com as faixas de valores e o percentual de cada faixa. Observa-se que os valores da variável de carbono fim de sopro estão concentrados entre 5 e 10 pontos, o que justifica o maior erro da RNA para valores maiores do que 10 pontos de carbono.

Figura 31 - Histograma do número de amostras da variável Carbono Fim de Sopro.

Fonte: Autor (2023).

4.3 Resultado consolidado e taxas de acerto da RNA

Como conclusão deste Capítulo são apresentados os resultados consolidados dos modelos de RNA desenvolvidos neste trabalho, fazendo um comparativo entre o resultado observado na área de produção da Aciaria 1 com o resultado obtido pelo modelo de RNA.

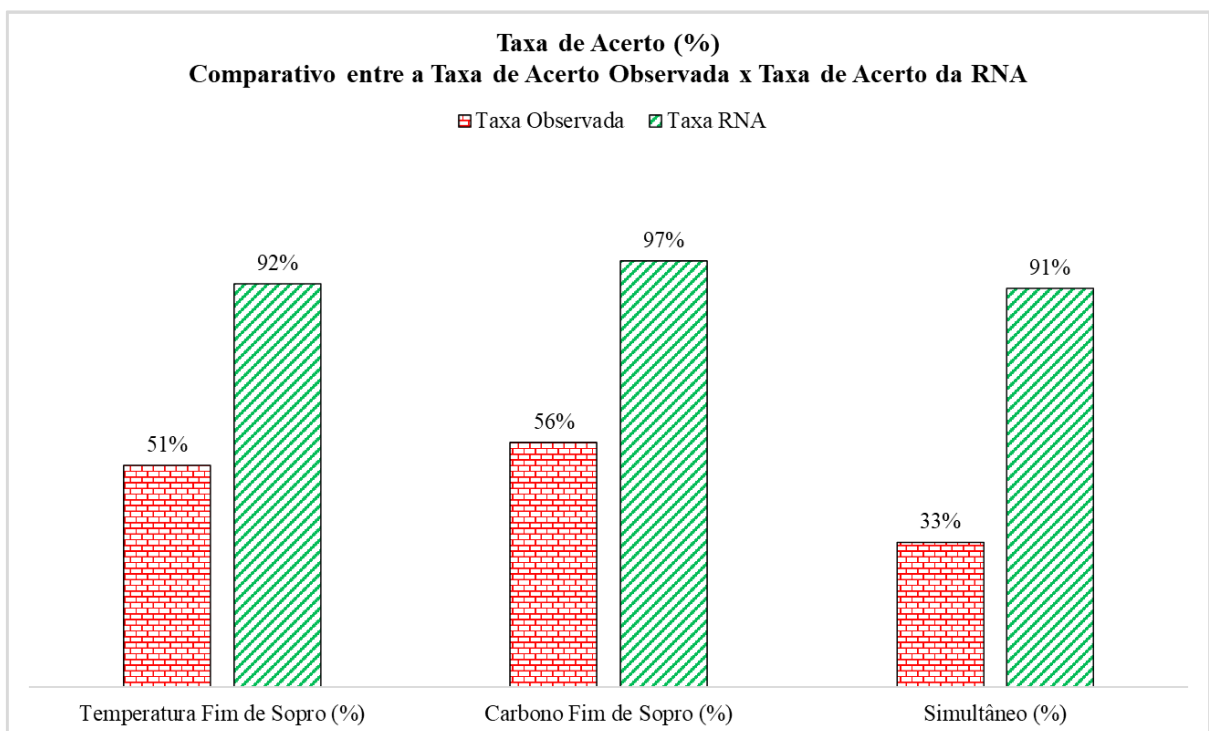
Os resultados a seguir foram gerados a partir de uma base de teste, com observações diferentes das observações utilizadas para o treinamento e validação da RNA. A base de teste possui 93 observações e foi extraída do banco de dados da Aciaria do período de 01/12/2022 a 08/12/2022.

Conforme já mencionado, o objetivo final do tratamento do aço em convertedores LD é garantir que ao final do processo de produção, o aço esteja com a temperatura e carbono dentro da faixa objetivada para que o processo produtivo possa seguir em diante.

Considerando os mesmos parâmetros de acerto de temperatura e carbono usados atualmente na empresa, o modelo de RNA desenvolvido apresentou melhores resultados se comparado com os resultados observados na área de produção. Foi avaliada a taxa de acerto da temperatura, a taxa de acerto do carbono e a taxa de acerto simultâneo das duas variáveis.

A Figura 32 apresenta o comparativo entre as taxas de acerto do modelo e as taxas de acerto observadas.

Figura 32 - Comparativo entre a taxa de acerto observada x taxa de acerto da RNA.



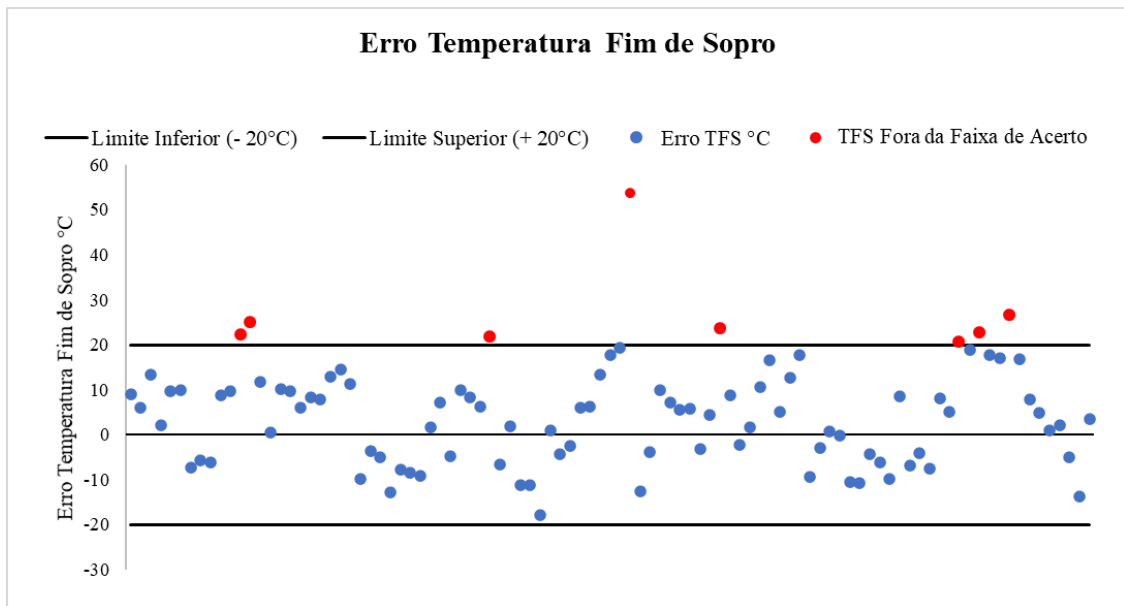
Fonte: Autor (2023).

Conforme pode-se observar na Figura 32, o modelo de RNA obteve um acerto de 92% na variável de temperatura fim de sopros, contra 51% dos resultados observados. Na variável

carbono fim de sopro, a taxa de acerto da RNA foi de 97%, contra 56% dos resultados observados. E por fim, a taxa de acerto simultâneo da RNA, foi de 91%, contra 33% dos valores observados.

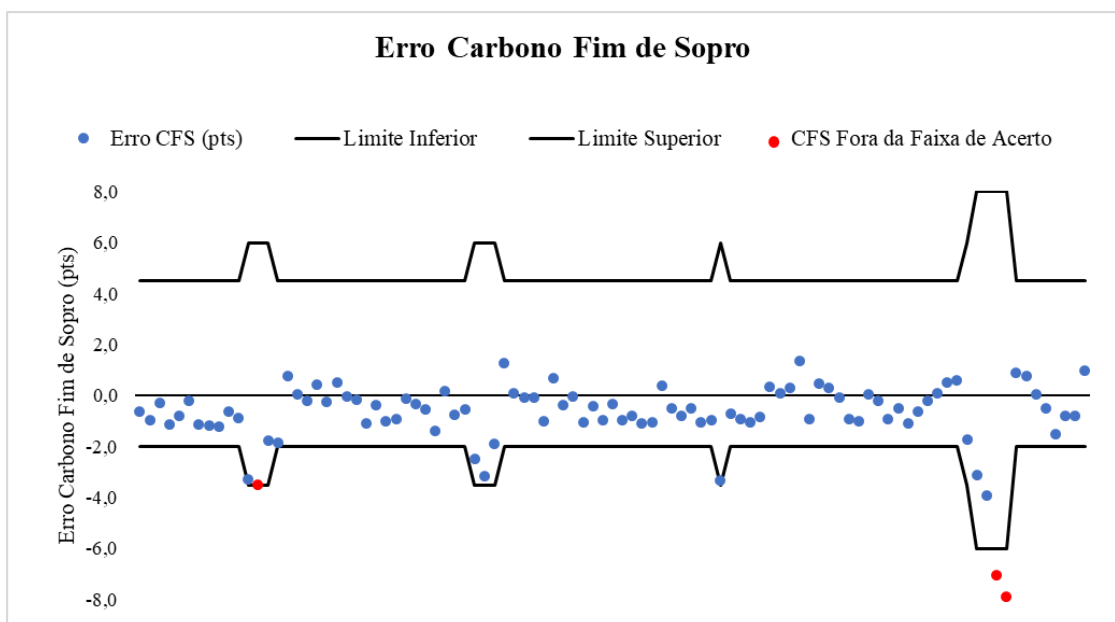
As Figuras 33 e 34 apresentam o erro das variáveis de temperatura fim de sopro e carbono fim de sopro.

Figura 33 - Erro da variável Temperatura Fim de Sopro.



Fonte: Autor (2023).

Figura 34 - Erro da variável Carbono Fim de Sopro.



Fonte: Autor (2023).

A Figura 33 mostra que 8 corridas ficaram fora da faixa de acerto da temperatura fim de sopra. Analisando detalhadamente essas corridas, observou-se que na maioria dos casos a temperatura objetivada era menor que 1650 °C, e conforme identificado nos resultados de validação, a RNA apresentou erros maiores nos valores dessa faixa. É importante ressaltar que, nessas 8 corridas que a RNA errou a predição da temperatura, o resultado observado na área também foi baixo, acertando a temperatura em apenas 3 corridas.

Com relação à Figura 34, observa-se que apenas 3 corridas das 93 que ficaram fora da faixa de acerto do carbono fim de sopra. Analisando os valores da base, observou-se que das 3 corridas, 2 tinham o valor de carbono fim de sopra objetivado acima de 10 pontos, e conforme foi identificado nos resultados de validação, a RNA apresentou erros maiores para os valores dessa faixa. Verificando o resultado observado na área dessas 3 corridas, em apenas uma que houve o acerto do carbono.

Para complementar, foi realizada outra análise em algumas corridas que o valor predito de volume de oxigênio ficou mais distante do valor observado, para verificar se essa variação foi benéfica para o modelo. O Quadro 5 apresenta os resultados. Na primeira corrida do Quadro 5, apesar de ter tido acerto simultâneo tanto da RNA como do resultado observado, o resultado da RNA mostrou que não era necessário soprar um volume alto de oxigênio e adicionar uma quantidade maior de síter para conseguir garantir a temperatura e carbono ao final do sopra. A mesma análise acontece para a penúltima corrida. Na última corrida do Quadro 5, o operador do convertedor soprou um volume muito baixo de oxigênio, e ainda adicionou síter na corrida, isso fez com que a temperatura fim de sopra ficasse muito baixa e o carbono fim de sopra alto, pois o volume de oxigênio não foi suficiente para promover a descarburização necessária. O restante das corridas está disponibilizado como Anexo ao final deste trabalho.

Quadro 5 - Comparativo entre os resultados da RNA x resultados observados

NUM_CORR_ACI	TEMP_FIM_SOPRO_VIS	TEMP_FIM_SOPRO_PRED	TEMP_FIM_SOPRO_REAL	C_FIM_SOPRO_EST	C_FIM_SOPRO_PRED	C_FIM_SOPRO_REAL	VOLUME_O2_PRED	VOLUME_O2_REAL	SINTER_SOPRO_PRED	SINTER_SOPRO_REAL
●● 261745	1660	1673	1650	5	5,5	3,0	3785	4070	432	1159
●● 261750	1670	1660	1709	5	3,9	3,0	3819	4450	7	0
●● 261770	1660	1666	1636	8	4,9	5,9	3777	3490	16	0
●● 360691	1660	1665	1617	5	4,2	5,7	3850	3570	12	290
●● 360709	1672	1663	1657	5	4,1	7,3	3796	4080	-2	161
●● 360730	1650	1668	1634	9	5,1	24,6	3769	3400	20	182

●● Acerto Simultâneo da RNA e Acerto Simultâneo Observado
 ●● Acerto Simultâneo da RNA

Fonte: Autor (2023).

4.3.1 Comparação de valores médios e MAE da RNA x valores observados e objetivados

As Tabelas 4 e 5 apresentam um comparativo entre os valores médios e MAE obtidos na RNA x os valores médios e MAE observados e objetivados no processo produtivo. É possível verificar que os valores médios de temperatura e carbono ficaram próximos entre os modelos, com exceção da variável Peso de Sínter que teve uma diferença maior entre os valores observados e a RNA. Porém, é importante destacar que, apesar dos valores médios estarem próximos, os valores de MAE são mais importantes para o acerto das variáveis de temperatura e carbono fim de sopro, pois, sendo o MAE a média do erro absoluto, este passa a ser um complemento para a análise uma vez que, quanto menor o valor do MAE, melhor será o acerto do modelo. Na Tabela 5, verifica-se que o MAE da RNA foi de 10, ou seja, na média, o erro absoluto do valor objetivado *versus* o valor predito pelo modelo, variou 10°C e o MAE dos valores observados foi de 24, ou seja, na média, o erro absoluto do valor objetivado *versus* o valor observado, variou 24°C sendo aceitável pelos parâmetros operacionais, uma variação de $\pm 20^\circ\text{C}$.

Tabela 4 - Comparativo entre os valores médios da RNA x Observados x Objetivados.

Valores Médios: RNA x Observados x Objetivados				
Modelo	Temperatura Fim de Sopro (°C)	Carbono Fim de Sopro (Pts)	Volume de Oxigênio (Nm³)	Sínter Sopro (kg)
Observado	1.671	5,0	3.857	209,5
RNA	1.670	4,7	3.865	83,1
Objetivado	1.665	5,5	-	-

Fonte: Autor (2023).

Tabela 5 - Comparativo entre os valores de MAE da RNA x Observados.

Erro Médio Absoluto (MAE)		
Modelo	Temperatura Fim de Sopro (°C)	Carbono Fim de Sopro (Pts)
Observado	24	2,1
RNA	10	1,0

Fonte: Autor (2023).

Com relação aos valores de MAE da variável carbono fim de sopro, a diferença entre o MAE da RNA e dos valores observados foi menor, o que justifica o melhor desempenho de acerto desta variável.

5 Conclusão

Neste trabalho empregou-se redes neurais do tipo *Multilayer Perceptron* ao processo produtivo do aço líquido em convertedores LD, no qual foram desenvolvidas duas RNAs, sendo a primeira para prever o volume de oxigênio que será soprado e o peso de sínter que será adicionado no processo de produção do aço, e a segunda RNA foi desenvolvida para validar os resultados da primeira RNA, pois o objetivo dela é verificar se os valores de volume de oxigênio e peso de sínter preditos na primeira RNA serão suficientes para garantir a temperatura e o carbono no final do sopro dentro dos valores objetivados.

Para identificar os melhores modelos, foram testadas diversas topologias com parâmetros diferentes, utilizando a biblioteca do Keras como suporte. Com o auxílio do Keras Tuner foi possível definir um range para vários parâmetros, de forma que o algoritmo teste várias topologias diferentes e armazena aquela que obteve o melhor resultado.

Após a definição da topologia das RNAs e dos parâmetros de configuração, foi feito o treinamento e validação das RNAs com uma base de dados de 6.482 corridas, sendo que 80% da base foi destinada para o treinamento da rede e 20% para validação dos resultados.

Depois de treinadas e validadas, as RNAs foram testadas em uma nova base de dados, com corridas desconhecidas pelos modelos para verificar o comportamento com novos dados. A nova base foi extraída de um período diferente e possui 93 corridas.

Com relação ao primeiro objetivo específico, que era identificar as variáveis significativas a partir de análises estatísticas que influenciam no peso de sínter, volume de oxigênio, temperatura e carbono de fim de sopro do aço líquido, ele foi alcançado utilizando técnicas estatísticas como gráficos de *boxplot*, *heatmap* e regressões para verificar o comportamento das variáveis.

Além das análises estatísticas, foi utilizado como base o trabalho de Cunha (2001) que também utilizou RNA aplicada em convertedores LD para predição do volume de oxigênio e o peso de sínter, e conversas com os especialistas do processo na Aciaria.

Em relação aos outros dois objetivos específicos, que eram desenvolver as duas RNAs para predição do volume de oxigênio e peso de sínter, e predição de temperatura e carbono fim de

sopro, os resultados apresentados no Capítulo 4 comprovaram a eficiência dos modelos de RNA aplicados no processo siderúrgico, mostrando que é uma ferramenta alternativa aos modelos metalúrgicos presentes em diversas siderurgias. O modelo de RNA obteve um acerto de 92% na variável de temperatura fim de sopro, contra 51% dos resultados observados. Na variável carbono fim de sopro, a taxa de acerto da RNA foi de 97%, contra 56% dos resultados observados. E por fim, a taxa de acerto simultâneo da RNA, foi de 91%, contra 33% dos valores observados.

5.1 Sugestões de trabalhos futuros

Tendo em vista os resultados satisfatórios apresentados neste trabalho, foram elencadas novas oportunidades de aplicação dos conceitos desenvolvidos, gerando demandas e sugestões de trabalhos futuros:

- Desenvolver uma RNA para predição do fósforo fim de sopro, sendo o controle deste elemento químico um importante parâmetro para os processos posteriores ao convertedor;
- Estreitar a faixa de acerto da Temperatura Fim de Sopro, de $\pm 20^{\circ}\text{C}$ para -10°C e $+20^{\circ}\text{C}$ de forma a reduzir os desvios dos valores objetivados, utilizando como base a própria RNA desenvolvida neste trabalho, apenas treinando a rede com novos parâmetros.

REFERÊNCIAS

ARAÚJO, L. A. **Manual de siderurgia**. 2ª edição, Brasil: Editora Arte e Ciência, 2009. Vol.1.

ANDERSON, J. A.; SUTTON, J. P. **The network of networks model**. Washington, DC: Erlbaum, Proceedings of the World Congress of Neural Networks, vol 1, pp. 145–152, 1995.

BARÃO, C. D. et. al. **Aciaria a oxigênio**. São Paulo: Associação Brasileira de Metalurgia e Materiais - ABM, Curso de Aciaria a Oxigênio, 2016.

BERTRAND, J. W. M.; FRANSOO, J. C. **Operations management research methodologies using quantitative modeling**. **International Journal of Operations & Production Management**, MCB UP Ltd, v. 22, n. 2, p. 241–264, 2002. ISSN 0144-3577. Disponível em:<<https://doi.org/10.1108/01443570210414338>>. Acesso em: 8 abr. 2022.

BORGES, R. A. A. **Otimização do carregamento dos fornos de produção de aço (convertedores LD) minimizando custo**. Belo Horizonte: Universidade Federal de Minas Gerais UFMG - Departamento de Ciência da Computação, 2016. 92 p.

BRAGA, A. de P.; CARVALHO, A. C. P. de L. F.; LUDERMIR, T. B. **Fundamentos de redes neurais artificiais**. Rio de Janeiro: DCC/IM - COPPE/Sistemas - NCE-UFRJ, 246p., 1998.

CAUCHICK-MIGUEL, P. A.; FLEURY, A.; MELLO, C. H. P.; NAKANO, D. N. **Metodologia de pesquisa em engenharia de produção e gestão de operações**. 3. ed. Rio de Janeiro: Elsevier, 2018. 244 p. ISBN 9788535291346. Disponível em: <<https://integrada.minhabiblioteca.com.br/books/9788595153561>>. Acesso em: 8 abr. 2022.

CHAVES, A. J. M. **Avaliação do desempenho operacional de um convertedor LD através do desenvolvimento do processo de sopro com lança de quatro furos**. Belo Horizonte: Universidade Federal de Minas Gerais UFMG - Departamento de Engenharia Metalúrgica e de Minas, 123 p., 2006.

CHESTER, D.L. **Why two hidden layers are better than one**. Washington, DC: Erlbaum, Proceedings of the World Congress of Neural Networks, vol 1, pp. 265–268, 1990.

CHURCHLAND, P.S.; SEJNOWSKI, T.J. **The computational brain**. Cambridge:

Massachusetts Institute of Technology - MIT Press, 558 p., 1992.

CONFEDERAÇÃO NACIONAL DA INDÚSTRIA (CNI). **A indústria do aço no Brasil**. Brasília, 2017.

CUNHA, A. P. **Redes Neurais em processos siderúrgicos: analisador virtual de propriedades metalúrgicas do sinter e modelo de predição de qualidade do aço**. Campinas: Universidade Estadual de Campinas UNICAMP - Faculdade de Engenharia Química, 250 p., 2001.

DA SILVA, I.N.; SPATTI, D.H.; FLAUZINO, R.A. **Redes neurais artificiais para engenharia e ciências aplicadas**. São Paulo: Artliber Editora, 399 p., 2010.

DOZAT, T. (2016). Incorporating Nesterov momentum into Adam. ICLR Workshop.

DUTRA, K. **Processos metalúrgicos – Fabricação do aço**. Fortaleza: CEPEP, 2017.

FUKUSHIMA, K. **Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position**. Tokyo: Springer-Verlag, Biological Cybernetics 36, pp. 193-202, 1980.

FUNAHASHI, K. **On the approximate realization of continuous mappings by neural networks**. USA: Neural Networks, vol.2, p. 183-192, 1989.

GOODFELLOW, I., BENGIO, Y., & COURVILLE, A. **Deep learning**. MIT press, 2016.

HAYKIN, S. **Redes Neurais- Princípios e Práticas**. Bookman, São Paulo, 2ª ed. 2001.

HERSKOVIC, J. **Elaboração do aço: fusão e refino**. Reading: Mundim, M. J., 1976 Capítulo IV: O Processo LD Associação Brasileira de Metais, ABM; São Paulo, SP; 1976.

HINTON, G. E.; SALAKHUTDINOV, R. R. **Reducing the dimensionality of data with neural networks**. New York: American Association for the Advancement of Science, Science 313, pp. 504-507, 2006.

HOPFIELD, J. J. **Neural networks and physical systems with emergent collective computational abilities**. Washington: Proceedings of the National Academy of Science of the USA, Biophysics, v.79, p. 2554-2558, 1982.

HUSSAIN, M.A. **Review of the applications of neural networks in chemical process control**

- **simulation and online implementation.** Artificial Intelligence in Engineering, 13, p. 55-68, 1999.

INSTITUTO AÇO BRASIL (IAB). **A indústria do aço no Brasil - Encontro da Indústria para a Sustentabilidade.** Brasília, 2012.

INSTITUTO AÇO BRASIL (IAB). **Processo Siderúrgico.** Disponível em: <https://acobrasil.org.br/site/processo-siderurgico/>. Acesso em 21 set. 2022.

HINTON, G. (2012). Neural Networks for Machine Learning. Lecture 6e – RMSprop.

KLAMBAUER, G., UNTERTHINER, T., MAYR, A., & HOCHREITER, S. (2017). Self-normalizing neural networks. In Advances in neural information processing systems (pp. 971-980).

KOVÁCS, Z. L. **Redes neurais artificiais: fundamentos e aplicações.** São Paulo: Collegium Cognito, 2ª edição, 174 p., 1996.

MAIA, B. T. et. al. **Lança de oxigênio – Múltiplas funções durante o tap to tap.** Brasil: Associação Brasileira de Metalurgia e Materiais, 49º Seminário de Aciaria, 2018.

MALINOVA, T.; MALINOV, S.; PANTEV, N. **Simulation of microhardness profiles for nitrocarburized surface layers by neural network.** Surface and Coatings Technology 135 p.258-267, 2001.

MCCULLOCH, W. S.; PITTS, W. **A logical calculus of the ideas immanent in nervous activity.** *Grã-Bretanha: Bulletin of Mathematical Biology.* v.5, p.115-133, 1943. Reimpressão: vol. 52, n°. 1/2. pp. 99-115, 1990.

MENSH, B.; KORDING, K. Ten simple rules for structuring papers. **PLoS Computational Biology**, Public Library of Science, v. 13, n. 9, p. e1005619, 2017.

MINSKY, M. L.; PAPERT, S. A. **Perceptrons: an introduction to computational geometry.** Cambridge, MA: Massachusetts Institute of Technology – MIT Press, Expanded Edition, 311 p., 1988.

MOREIRA, D.B.; LANA, M.F., **A Geração de Escória de Aciaria LD pelo Método de Convertedor com Sopro de Oxigênio.** Material Didático, Companhia Siderúrgica de Tubarão, 2002.

NWANKPA, C., IJOMAH, W., GACHAGAN, A., & MARSHALL, S. (2018). Activation functions: Comparison of trends in practice and research for deep learning. arXiv preprint arXiv:1811.03378.

PORTO EDITORA. **Processo Siemens-Martin.** Disponível em: [https://www.infopedia.pt/\\$processo-siemens-martin](https://www.infopedia.pt/$processo-siemens-martin). Porto: Porto Editora. Acesso em: 10 set. 2022.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Learning representations by back-propagating errors.** New York: Springer, Nature 323, pp. 533-536, 1986.

SABBATINI, R. M. E. **Aplicações biomédicas de redes neurais artificiais.** Campinas: Núcleo de Informática Biomédica da Universidade Estadual de Campinas, 1992.

SHAW, L. S.; SIMÕES, M. G. **Controle e modelagem FUZZY.** São Paulo: Editora Edgard Blücher Ltda; 1ª edição, 1999.

SILVA, A. J. **Estudos dos principais processos metalúrgicos de aciaria (EOF, FEA, LD).** Divinópolis: Faculdade Pitágoras / Unidade Divinópolis, 2013. 54 p.

SILVA, R. M. **Redes Neurais Artificiais aplicadas à Detecção de Intrusão em Redes TCP/IP.** Rio de Janeiro: Pontifícia Universidade Católica do Rio de Janeiro – PUC-RJ, 2005. 14 p.

SPALL, J. C. (1992). Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation. IEEE Transactions on Automatic Control, 37(3), 332–341.

TAFNER, M.; XEREZ, M.; RODRIGUES, I. **Redes neurais artificiais: Introdução e princípios de neurocomputação.** Blumenau: EKO, 202 p., 1996.

TAKEMURA, Y.; SAITO, T.; FUKUDA, S.; KATO, K. **BOF Dynamic control using subblance system.** Nippon Steel Technical Report, No. 11, p. 57-68, 1978.

VELLASCO, M. M. B. R. **Redes Neurais Artificiais.** Rio de Janeiro: PUC Rio - Laboratório de Inteligência Computacional Aplicada, 41p., 2007.

VOSS, C.; TSIKRIKTSIS, N.; FROHLICH, M. Case research in operations management.

International Journal of Operations & Production Management, MCB UP Ltd, v. 22, n. 2, p. 195–219, 01 2002. ISSN 0144-3577. Disponível em: <<https://doi.org/10.1108/01443570210414329>>. Acesso em: 8 abr. 2022.

WAZLAWICK, R. S. **Metodologia de pesquisa para ciência da computação**. Rio de Janeiro: Elsevier, 2009. 124 p. 6ª impressão. ISBN 9788535235227.

WIDROW, B.; HOFF, M. E. **Adaptive switching circuits**. Los Angeles: Ire Wescon Convention Record, pp. 96-104. 1960.

ANEXO

Resultado das corridas realizadas para comparação entre os valores reais e os valores preditos pela rede neural.

NUM_CORR_ACI	TEMP_FIM_SOPRO_VIS	TEMP_FIM_SOPRO_PRED	TEMP_FIM_SOPRO_REAL	C_FIM_SOPRO_EST	C_FIM_SOPRO_PRED	C_FIM_SOPRO_REAL	VOLUME_O2_PRED	VOLUME_O2_REAL	SINTER_SOPRO_PRED	SINTER_SOPRO_REAL
261716	1660	1669	1668	5	4,4	4,2	3954	4150	29	0
261717	1660	1666	1673	5	4,0	3,6	3943	3990	-19	0
261718	1660	1673	1709	5	4,7	2,8	3976	3880	132	0
261720	1660	1662	1674	5	3,9	3,5	3873	4050	9	0
261721	1660	1670	1669	5	4,2	3,6	3943	3860	19	0
261722	1667	1677	1689	5	4,8	4,4	3961	3930	164	109
261724	1669	1662	1658	5	3,9	3,5	4035	3910	-111	26
261725	1669	1663	1682	5	3,8	4,1	3935	3940	-14	0
261726	1669	1663	1654	5	3,8	3,9	3963	3850	-33	0
261730	1660	1669	1646	5	4,4	3,6	3955	3830	20	0
261731	1660	1670	1691	5	4,1	3,2	4050	4090	75	590
261732	1650	1672	1705	8	4,7	3,4	3889	3690	111	445
261734	1640	1665	1580	8	4,5	21,0	3816	3620	22	517
261735	1666	1678	1634	7	5,2	4,5	3956	3920	217	0
261737	1666	1666	1619	6	4,1	3,9	3890	3770	5	0
261739	1665	1675	1710	5	5,8	5,7	3896	3980	216	558
261740	1665	1675	1672	5	5,0	8,1	3965	3820	115	579
261741	1665	1671	1701	5	4,8	4,8	3945	3930	69	350
261743	1665	1673	1632	5	5,4	6,9	3790	3780	285	1272
261744	1665	1673	1681	5	4,7	11,6	3941	4140	225	1061
261745	1660	1673	1650	5	5,5	3,0	3785	4070	432	1159
261746	1660	1675	1700	5	5,0	3,2	3897	3980	254	1970
261748	1660	1671	1693	5	4,8	3,0	3816	3990	100	2338
261750	1670	1660	1709	5	3,9	3,0	3819	4450	7	0
261753	1670	1666	1675	5	4,6	3,4	3774	3860	-10	0
261754	1670	1665	1666	5	4,0	3,8	3884	3950	-26	0
261755	1679	1666	1636	5	4,1	3,6	3899	3970	-22	0
261756	1679	1671	1699	5	4,9	2,8	3778	4000	31	0
261759	1679	1671	1675	5	4,7	3,7	3821	4050	19	0
261760	1679	1670	1671	5	4,5	3,0	3858	4060	-2	0

NUM_CORR_ACI	TEMP_FIM_SOPRO_ VIS	TEMP_FIM_SOPRO_ _PRED	TEMP_FIM_SOPRO_ _REAL	C_FIM_SOPRO_ _EST	C_FIM_SOPRO_ _PRED	C_FIM_SOPRO_ _REAL	VOLUME_O2_ _PRED	VOLUME_O2_ _REAL	SINTER_SOPRO_ _PRED	SINTER_SOPRO_ _REAL
261764	1660	1662	1677	5	3,6	3,4	3991	4160	-5	0
261765	1660	1667	1665	5	5,2	3,2	3725	3870	35	0
261766	1670	1665	1649	5	4,3	5,9	3886	3950	9	0
261767	1660	1670	1678	5	4,5	3,5	3909	4080	62	188
261768	1660	1668	1660	8	5,5	3,6	3735	3650	75	0
261770	1660	1666	1636	8	4,9	5,9	3777	3490	16	0
261771	1650	1672	1689	8	6,1	4,0	3667	3650	240	0
261772	1679	1672	1632	5	6,3	6,9	3688	3820	286	114
261775	1665	1667	1667	5	5,1	3,8	3738	3760	45	124
261776	1679	1668	1679	5	4,9	5,4	3852	3910	61	147
261777	1679	1668	1688	5	4,9	3,2	3857	3820	62	136
261778	1679	1661	1660	5	4,0	3,1	3853	3860	-17	182
261780	1670	1671	1711	5	5,7	3,4	3708	3800	41	0
261781	1670	1666	1667	5	4,6	4,5	3800	3700	6	0
261786	1670	1667	1684	5	5,0	2,2	3779	3760	60	0
261787	1660	1666	1669	5	4,0	2,4	3944	4020	20	254
360680	1670	1676	1599	5	4,6	4,3	3953	3960	151	1617
360681	1650	1663	1643	5	4,0	3,7	3902	4010	-16	0
360682	1650	1668	1707	5	4,7	3,0	3752	3840	2	0
360683	1650	1669	1642	5	4,1	4,6	4034	3990	322	0
360684	1665	1719	1716	5	4,2	3,3	3913	4040	521	0
360685	1679	1666	1646	5	3,9	2,3	3983	3870	-74	0
360687	1670	1666	1678	5	4,0	2,9	3904	3750	11	0
360689	1660	1670	1680	5	5,4	11,5	3806	3530	105	48
360690	1660	1667	1662	5	4,5	4,7	3846	3700	16	128
360691	1660	1665	1617	5	4,2	5,7	3850	3570	12	290
360692	1669	1675	1708	5	4,5	4,4	3952	3970	94	0
360694	1670	1667	1711	5	3,9	9,3	3964	3860	-20	0
360696	1665	1669	1736	5	4,1	3,0	4052	4180	102	60
360697	1650	1673	1649	8	4,7	7,3	3958	4120	115	0

NUM_CORR_ACI	TEMP_FIM_SOPRO_VIS	TEMP_FIM_SOPRO_PRED	TEMP_FIM_SOPRO_REAL	C_FIM_SOPRO_EST	C_FIM_SOPRO_PRED	C_FIM_SOPRO_REAL	VOLUME_O2_PRED	VOLUME_O2_REAL	SINTER_SOPRO_PRED	SINTER_SOPRO_REAL
360699	1670	1668	1671	5	4,1	4,1	3889	3910	10	78
360700	1665	1667	1696	5	4,0	2,3	3914	3870	21	0
360701	1660	1671	1702	5	4,1	3,6	4045	3840	104	742
360702	1660	1677	1682	5	5,3	4,7	3942	3740	244	744
360703	1670	1675	1639	5	5,1	5,0	3971	3760	107	935
360706	1665	1678	1670	5	5,3	8,4	3902	3850	392	1716
360708	1665	1683	1723	5	6,3	2,7	4044	4270	321	1174
360709	1672	1663	1657	5	4,1	7,3	3796	4080	-2	161
360710	1675	1672	1692	5	5,5	4,2	3774	3900	66	159
360711	1670	1671	1691	5	5,3	4,9	3772	3790	39	0
360712	1670	1670	1693	5	4,9	2,7	3778	3850	10	0
360713	1679	1668	1660	5	4,1	4,2	3891	3830	-22	0
360714	1679	1668	1662	5	4,0	4,0	3924	3970	-28	0
360715	1679	1675	1684	5	5,0	3,7	3794	3880	54	0
360716	1679	1673	1711	5	4,8	2,9	3814	3880	26	0
360719	1679	1669	1647	5	4,1	5,4	3912	3940	-25	0
360720	1660	1668	1695	5	4,5	2,0	3785	3650	32	0
360721	1675	1668	1665	5	3,9	2,5	4030	3800	-23	0
360722	1670	1666	1642	5	4,4	3,1	3914	3670	-1	0
360723	1675	1667	1640	5	4,8	3,6	3803	3770	12	0
360724	1660	1668	1658	5	5,1	3,2	3756	3670	28	0
360725	1665	1670	1695	5	5,5	3,9	3732	3650	27	0
360727	1650	1671	1633	5	5,6	4,2	3750	3700	121	0
360728	1650	1669	1674	7	5,3	6,9	3754	3580	65	0
360729	1650	1673	1660	10	6,9	12,2	3695	3490	400	0
360730	1650	1668	1634	9	5,1	24,6	3769	3400	20	182
360731	1650	1667	1650	12	4,9	11,4	3758	3540	11	0
360732	1645	1672	1668	14	6,1	9,2	3669	3420	154	0
360733	1655	1672	1699	5	5,9	3,7	3733	3700	182	0
360734	1665	1673	1680	5	5,8	5,6	3733	3610	130	0
360735	1665	1670	1656	5	5,1	3,9	3779	3600	70	0
360736	1665	1666	1699	5	4,5	3,1	3813	3790	20	0
360737	1660	1662	1572	5	3,5	7,3	4058	3900	-35	11
360739	1669	1664	1697	5	4,2	2,7	3854	3940	7	0
360740	1679	1665	1646	5	4,2	6,7	3845	3850	-38	0
360741	1669	1673	1665	5	6,0	10,5	3727	3570	201	0