UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

PAULO VITOR DUARTE DE SOUZA

REDE NEURAL ARTIFICIAL PARA PREDIÇÃO DA PRODUTIVIDADE DA CULTURA DO MILHO

SANTA HELENA

2021

PAULO VITOR DUARTE DE SOUZA

REDE NEURAL ARTIFICIAL PARA PREDIÇÃO DA PRODUTIVIDADE DA CULTURA DO MILHO

ARTIFICIAL NEURAL NETWORK FOR FORCASTING CORN YIELD

Trabalho de Conclusão de Curso de Graduação apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação da Coordenação de Ciência da Computação da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientador: Prof. Dra. Leiliane Pereira de Rezende

Co-orientador: Prof. Dr. Glauco Vieira Miranda

SANTA HELENA

2021

PAULO VITOR DUARTE DE SOUZA

REDE NEURAL ARTIFICIAL PARA PREDIÇÃO DA CULTURA DO MILHO

Trabalho de Conclusão de Curso de Graduação apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação da Coordenação de Ciência da Computação da Universidade Tecnológica Federal do Paraná.

Data de Aprovação: 23/Agosto/2021

Leiliane Pereira de Rezende - Doutora em Ciência da Computação

Universidade Tecnológica Federal do Paraná, UTFPR

Glauco Vieira Miranda - D.Sc. Doctor Scientiae

Universidade Tecnológica Federal do Paraná, UTFPR

Thiago França Naves - Doutor em Ciência da Computação

Universidade Tecnológica Federal do Paraná, UTFPR

Gloria Patricia Lopez Sepulveda - Doutora em Engenharia Elétrica

Faculdade de Ensino Superior de São Miguel do Iguaçu, FAESI

SANTA HELENA

2021

Dedico este trabalho à Cristo Jesus, por
meio dele pude me achegar a Deus.

AGRADECIMENTOS

Antes de tudo, sem Deus esse trabalho não existiria, a Ele toda honra e gloria para todo sempre.

Agradeço a todos que participaram dessa jornada, mesmo que não esteja presente nessas breves palavras.

Agradeço ao Aildson Pereira Duarte - Programa Milho e Sorgo IAC/APTA, Campinas - pelos dados disponibilizados, sem dúvidas há uma grande contribuição no desenvolvimento desse trabalho.

A minha família pelo apoio, todos contribuíram para esse trabalho, especialmente aos meus pais Maria José e José Heleno.

Agradeço as meus amigos Marclícia Nilcleany, Lais Paixão, Ewellyn Sousa, Sílvia Contini, Fábio, Pr. Luciano, Nathalia Mota, Edgar Mota, Raíssa Macedo pelo apoio.

Aos meus orientadores, Prof. Dra. Leiliane Pereira de Rezende e Prof. Dr. Glauco Vieira Miranda.

Os céus proclamam a tua gloria e o firmamento anuncia a obra das suas mãos. (Salmos 19:1)

RESUMO

A predição da produtividade da cultura do milho tem muitos benefícios na produção global de alimentos e de pequenos agricultores, por ser possível gerenciar melhor o processo de plantio e otimizar os lucros. Este trabalho objetiva construir modelos de Perceptrons de Múltiplas Camadas para a predição da produtividade do milho no Vale do Paranapanema, São Paulo, considerando parâmetros de desenvolvimento, condições climáticas e balanço hídrico. Foram considerados dados de dois anos agrícolas de diferentes localidades. Dados faltantes foram imputados por meio do *iterative imputation*. A hiperparametrização dos modelos foram obtidas por meio do *GridSearch* e *k-fold cross-validation*. Os modelos foram divididos em bases de dados com e sem a imputação de dados faltantes. A interpretação do modelo foi realizada pelo método SHAP. Os modelos obtiveram um resultado aceitável sendo o melhor modelo com um RMSE de 70,651 $kg \cdot ha^{-1}$ considerando as bases de dados imputadas. O modelo sem imputação na base de dados obteve 190,851 $kg \cdot ha^{-1}$. Em todos os modelos as condições climáticas foram as que tiveram maior peso na predição da produtividade. Conclui-se que os modelos construídos obtiveram um desempenho aceitável e capturaram os efeitos não lineares entre o ambiente e o genótipo da planta de milho.

Palavras-chave: Rede Neural Artificial. Multilayer Perceptron. Produtividade. Milho. Predição.

ABSTRACT

The forecasting corn yield has many advantages in global food production and small farmers. The forecasting can manage better the crop and optimize the profit. The objective of the work is to build models of Multilayer Perceptrons for the forecasting of corn yield in the Vale do Paranapanema, São Paulo, with the crop growing features, weather conditions and water balance. Two year (2018 and 2019) crop of different locations were considered. Missing data were imputed through the iterative imputation. The model hyper parametrization were obtained through of the GridSearch and k-fold cross-validation. The models were divided into datasets with and without data imputation. The interpretation of the model was made through SHAP method. The models obtained satisfactory results, with the better model had RMSE de 70,651 $kg \cdot ha^{-1}$ for the imputed dataset. In the data without imputation obtained 190,851 $kg \cdot ha^{-1}$. In all models, the weather conditions were that had importance in predictions of yield. Thus, the build models had satisfactory performance and they took the nonlinear interactions among the crop genotype and the environment.

Keywords: Artificial Neural Network. Multilayer Percetron. Yield. Corn. Forecasting.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ADAM	Momentos Adaptativos (*Adaptive Moments*)
ARM	Armazenamento
BH1_V8	Armazenamento no Estádio Vegetativo
BH1_R1_2	Armazenamento no Estádio Reprodutivo
BH2_R1_2	Evapotranspiração de Referência no Estádio Reprodutivo
BH3_V8	Deficit no Estádio Vegetativo
BH3_F	Deficit no Estádio de Florescimento
BH4_V4	Excedente no Estádio Vegetativo
BH4_R6	Excedente no Estádio Reprodutivo
ETR	Evapotranspiração Real
ETo	Evapotranspiração de Referência
F	Estádio de Florescimento
GPU	Unidade de Processamento Gráfico (*Graphics Processing Unit*)
INMET	Instituto Nacional de Meteorologia
API	Interface de Programação de Aplicações (*Application Programming Interface*)
LIME	Substituto Local (*Local Surrogate*)
MAE	Erro Médio Absoluto (*Mean Absolute Error*)
Med_Prod_local	Média da Produtividade Local
MLP	Perceptron de Múltiplas camadas (*Multilayer Perceptron*)
MSE	Erro Quadrático Médio (*Mean Squared Error*)
R	Estádio Reprodutivo
RE	Relação entre Altura da Planta e Espiga

ReLU	Unidade Linear Ratificada (*Rectified Linear Unit*)
RMSE	Raiz Quadrada do Erro Médio (*Root Mean Square Error*)
RMSProp	Propagação da Raiz Quadrada Média (*Root Mean Square Propagation*)
RNA	Rede Neural Artificial
SGD	Gradiente Descendente Estocástico (*Stochastic Gradiente Descent*)
SHAP	Explicações Aditivas de SHapley (*SHapley Additive exPlanations*)
Temp Max	Temperatura Máxima
Temp Min	Temperatura Mínima
V	Estádio Vegetativo
VE	Estádio Vegetativo Emergencial
W1_V8	Precipitação Total Diário no Estádio Vegetativo
W1_R6	Precipitação Total Diário no Estádio Reprodutivo
W5_R6	Temperatura Mínima Diária no Estádio Reprodutivo
W7_VE	Umidade Relativa do Ar Média Diária no Estádio Vegetativo Emergencial
W7_F	Umidade Relativa do Ar Média Diária no Estádio de Florescimento
W7_R1_2	Umidade Relativa do Ar Média Diária no Estádio Reprodutivo
W6_V8	Temperatura Mínima Diária no Estádio Vegetativo

LISTA DE SÍMBOLOS

$\alpha(u)$	Função de ativação
$f(x)$	Função que descreve o relacionamento de y e x
J	Função de custo
k	Partição da conjunto de dados segundo *k-fold*
L	L-ésima camada da MLP
n	Número de folha por estádio
N	Quantidade de exemplos de exposto a RNA
P($y\|x$)	Probabilidade condicional de y dado x
ρ	Coeficiente de Correlação de Person
u	Somador linear
v	Saída da função de ativação
x	Variável de entrada
x'	valor de x normalizado
x_{max}	Máximo valor assumido por x segunda avariável do conjunto de dados
x_{min}	Mínimo valor assumido por x segunda a variável do conjunto de dados
X	Vetor de entrada
y	Variável de saída ou alvo

SUMÁRIO

1 INTRODUÇÃO

A produção mundial de alimentos tem encontrado limites em relação à expansão de áreas para as atividades agrícolas. Isto é agravado pelo contínuo crescimento populacional, que exige demandas maiores para o suprimento mundial, e pela expansão das cidades (SAATH; FACHINELLO, 2018). Segundo as perspectivas da FAO (2015), durante os anos de 2014 a 2024, a demanda para países em desenvolvimento tende a ser maior que para países desenvolvidos. Esse problema não pode simplesmente ser resolvido expandindo novas áreas, devido a restrições impostas citadas anteriormente.

O aumento da produtividade é uma solução para suprir essa demanda dado que uma alta produtividade reduz a expansão de novas áreas. No entanto, nesses últimos anos, a produtividade tem crescido lentamente devido ao modelo tecnológico atual (GUIMARÃES, 2019). Em relação a cultura do milho, apesar de ser considerada a maior cultura do mundo, no Brasil, ela enfrenta alguns desafios para o aumento da produção: a falta de clareza na formação de preços; entraves para conseguir financiamentos privados; empecilhos na comercialização, sobretudo no processo de escoamento da produção; e baixa produtividade observada em algumas regiões (CONTINI et al., 2019).

Dentre as várias soluções disponíveis para o aumento da produtividade, a estimativa/previsão da produtividade tem sido utilizada para gerenciar o plantio e, consequentemente, diminuir os custos e gerenciar melhor os insumos disponíveis (VALE, 2019). Assim, o processo na totalidade pode ser melhorado, pois, por meio dele, é possível empregar ferramentas, cultivares e formas de manejo mais adequados onde será realizado o cultivo.

A estimativa/previsão da produtividade de uma cultura agronômica é um problema desafiador, pois o resultado final da colheita é fruto da interação do genótipo da planta e do ambiente. Considerando o ambiente, o clima é o fator que mais influência as culturas durante o seu ciclo de desenvolvimento (DAHIKAR; RODE, 2014). O manejo também tem um papel importante como, por exemplo, a profundi-

dade de semeadura e a densidade de plantio. Além destas, a disponibilidade hídrica no solo pode ser crítica para algumas culturas como, por exemplo, o milho (CRUZ et al., 2006).

O desenvolvimento de técnicas capazes de capturar essas relações e predizer eficazmente a produtividade da safra são desejáveis, dado que o conhecimento prévio conduz a um melhor aproveitamento dos recursos disponíveis. Muitas técnicas já foram aplicadas para esse propósito como, por exemplo, as técnicas de aprendizado de máquina (BANNERJEE et al., 2018). Isso ocorre devido ao aumento dos dados gerados na agricultura que, no que lhe concernem, permitem o emprego de técnicas de aprendizagem de máquina onde os dados são um insumo importante para o desenvolvimento e aplicação desses algoritmos (VENDRUSCULO; OLIVEIRA, 2010).

Uma técnica de aprendizagem de máquina que tem sido utilizada para a predição da produtividade de culturas agronômicas e tem tido sucesso em diversas áreas é a Rede Neural Artificial (RNA). A grande vantagem no uso dessa técnica é a sua capacidade de generalização a partir de um conjunto de dados que representa um problema além da sua capacidade de mapeamento de problemas não lineares (ZOU; HAN; SO, 2008).

Uma das RNAs que tem sido utilizada para estimativas de produtividade é a *Perceptron* de Múltiplas Camadas (MLP) caracterizada por tratar a produtividade como uma função implícita da entrada da rede (fatores que afetam a produtividade) (GUIMARÃES, 2019). As MLPs têm tido sucesso em criar modelos preditivos através dos parâmetros de desenvolvimento da planta, características do solo, manejo e condições climáticas com um pequeno erro médio para a produtividade (KAUL; HILL; WALTHALL, 2005; JI et al., 2007; KHAKI; WANG, 2019).

Assim, este trabalho tem como objetivo construir modelos preditivos de RNAs capazes de estimar a produtividade baseada nos fatores que afetam o desenvolvimento da cultura do milho. Nas subseções seguintes, os objetivos, as contribuições do trabalho bem como a justificativa são apresentados.

1.1 OBJETIVOS

O objetivo geral e os específicos são descritos, respectivamente, nas subseções 1.1.1 e 1.1.1.1.

1.1.1 Geral

O objetivo geral é construir uma RNA do tipo MLP para estimar a produtividade de híbridos de milho no Vale do Paranapanema, São Paulo, Brasil, considerando os parâmetros de desenvolvimento de plantas e dados climáticos.

1.1.1.1 Específicos

Os objetivos específicos são:

1. Pré-processar o conjunto de dados;
2. Imputar dados faltantes na base de dados;
3. Otimizar um conjunto de hiperparâmetros para os modelos de MLPs;
4. Construir um modelo de MLP para estimar a produtividade de milho, em função somente dos dados de desenvolvimento de planta;
5. Construir um modelo de MLP para estimar a produtividade de milho em função da variabilidade climática; e
6. Identificar as variáveis de maior peso em relação ao modelo de MLP criado.

1.2 CONTRIBUIÇÕES DO TRABALHO

Este trabalho contribuirá, por meio dos resultados obtidos pela RNA, com análises de produtividade sobre variedades de cultivares de milho convencionais de ciclo precoce e super precoce na região de estudo. Em relação ao conjunto de dados,

contribuirá com os procedimentos necessários para o pré-processamento da base de dados. Além disso, contribuirá com o entendimento sobre quais variáveis apresentam maior peso na estimativa da produtividade do milho para os modelos de MLPs treinados.

1.3 JUSTIFICATIVA

Um tópico que vem sendo discutido nos últimos anos é o crescimento populacional. Conforme esse aumento, há uma necessidade de produzir maiores quantidades de alimentos para suprir a demanda global. Contudo, os limites de terras para as áreas agricultáveis estão se esgotando (GUIMARÃES, 2019; MORETO, 2019; SRIVASTAVA et al., 2019). Assim sendo, chegará em um ponto onde a expansão de áreas não será mais possível. Dessa forma, técnicas mais precisas para melhorar o processo do cultivo, especificamente, para prever/estimar a produtividade, são indispensáveis para o uso adequado das terras (MICHELON, 2016) e, consequentemente, para o aumento da produtividade.

Em relação à produtividade das culturas agrícolas, ela depende de muitos fatores que não são controláveis como, por exemplo, a variabilidade climática no decorrer do desenvolvimento dos estádios[1] da planta. Estimativas servem para ter um controle sobre os custos de produção e a tentativa de otimizar ganhos futuros (PICOLI, 2007; RIZZI; RUDORFF, 2007; FILIPPI et al., 2019). Ou seja, antes da colheita, há possibilidade de criar planos para o plantio em relação às estimativas preditas possibilitando, assim, uma melhor tomada de decisão e tornando o processo de produção mais controlável apesar das incertezas enfrentadas durante o desenvolvimento da planta.

1.4 DELIMITAÇÕES DO TRABALHO

O trabalho limita-se em desenvolver modelos de RNAs do tipo MLP para análise de desempenho da cultura do milho. A validação será realizada por meio de duas

[1]Estádio se refere a um período de desenvolvimento da planta.

métricas de avaliação do modelo: raiz quadrada do erro médio (RMSE - *Root Mean Square Error*) e erro quadrático médio (MSE - *Mean Squared Error*). Além disso, os resultados obtidos estão limitados ao ambiente do Vale do Paranapanema e aos tipos de cultivares abordados (DUARTE; SAWAZAKI, 2018, 2019), não sendo generalizáveis para outros locais e outros cultivares. Outra limitação, é o conjunto de dados não possuir dados genéticos e de solos.

2 REVISÃO DA LITERATURA

Neste capítulo são apresentados os conceitos necessários para o entendimento dos capítulos seguintes. Na seção 2.1 são definidos os modelos de regressão e as métricas de avaliação. Na seção 2.2 é apresentado o desenvolvimento geral das redes neurais, especificamente os modelos alimentados adiante, características principais dessa rede e de seu treinamento. A seção 2.3 apresenta as considerações implementacionais dos modelos de aprendizado de máquina. Na seção 2.4 são apresentados os aspectos fenológicos e os fatores que influenciam na produtividade da cultura do milho. Por fim, na seção 2.5, são apresentados os trabalhos correlatos.

2.1 MODELOS DE REGRESSÃO

Modelos de regressão são modelos estatísticos que descrevem o comportamento de uma variável dependente y a partir de outras variáveis independentes x. Eles também podem ser definidos como uma distribuição de probabilidade condicional $P(y|x)$ (BISHOP, 2006). De forma geral, modelos de regressão descrevem esse relacionamento a partir de uma função $f(x)$, onde $x, y \in \mathbb{R}$. Esta formalização, $f : \mathbb{R} \to \mathbb{R}$, é o que diferencia os modelos de regressão dos modelos para tarefas de classificação, nos quais a saída é uma classe de um conjunto finito.

Conforme expõem Zeviani, Ribeiro junior e Bonat (2013), modelos de regressão permitem: (1) explicar a relação entre variáveis, ou em parte; (2) quantificar a influência de variáveis dependentes; (3) selecionar variáveis dependentes mais relevantes; (4) realizar predições de y a partir de dados observados ou não observados; e, por fim, (5) avaliar a incerteza do processo. Dependendo do modelo adotado como solução, os pontos (2) e (3) podem se tornar uma tarefa complexa, os quais são conhecidos como "caixa preta" onde as redes neurais pertencem a estas classes de modelos.

Descobrir uma função f que descreva a relação de y e x se resume em encontrar uma aproximação f de uma função alvo h, onde h define exatamente o relaciona-

mento de y e x, se houver. Esta aproximação é definida sobre um conjunto de N exemplos, conforme o par ordenado (x_1, y_1), (x_2, y_2), (x_3, y_3), ..., (x_n, y_n). Existem diversas formas para encontrar f ou ajustar f aos dados de exemplo. Estas formas e outras restrições especificam os tipos de modelos existentes. Esta seção apresenta a classe de modelos lineares (Seção 2.1.1) e descreve algumas métricas de avaliação (Seção 2.1.2).

2.1.1 Modelos de Regressão Linear

Modelos de regressão linear aplicam restrições na forma da função f, onde o interesse é facilitar a inferência ou a interpretação dos resultados. Em outras palavras, são modelos caracterizados pela linearidade dos parâmetros (HOCKING, 2003). Formalmente são definidos como:

$$y = w_0 + w_1 x^1 + w_2 x^2 + ... + w_m x^m \tag{2.1}$$

onde w_i é o i-ésimo coeficiente ou parâmetro a ser ajustado. Segundo a Equação 2.1, funções não lineares, por exemplo $y = x^2$, podem ser definidas como uma hipótese de um modelo linear, enquanto os parâmetros forem lineares. O comportamento das variáveis independentes são desconsideradas na definição.

A Equação 2.1 define, de forma geral, os modelos lineares. No entanto, é comum a utilização de modelos com mais restrições, na forma:

$$y = w_0 + w_1 x \tag{2.2}$$

Estes são conhecidos comumente como modelos lineares simples, onde existem apenas uma variável de entrada. Note que esta restrição impede a não linearidade da variável de entrada, implicando em uma função que traça uma reta onde a inclinação e o posicionamento são ajustados pelos pesos w.

Em problemas reais ou com maior complexidade, modelos lineares simples não são uteis devido à alta dimensionalidades desses problemas. Por outro lado, a partir da definição da Equação 2.2, podem ser estendidos para o caso multivariável:

$$y = w_0 + w_1 x + w_2 x_2 + ... + w_m x_m \tag{2.3}$$

onde m é quantidade de variáveis explicativas.

Uma das desvantagens do uso de modelos lineares é a incapacidade de mapear problemas inerentemente não-lineares. Problemas agronômicos, em grande parte, possuem essa característica. Uma forma de contornar é aplicar uma função logística em y, possibilitando o mapeamento das variáveis (SANTOS et al., 2005). Esta variação é conhecida como regressão logística que tem como desvantagem a dificuldade de interpretação. Apesar das RNAs (veja a Seção 2.2) possuírem os mesmos desafios da regressão logística, elas são ótimas alternativas para o mapeamento não-linear e têm ganhado notoriedade no meio científico pelo poder de reconhecimento de padrões (SANTOS et al., 2005).

2.1.2 Avaliação e Análises de Modelos Regressão

Assim que o modelo de regressão é definido, quer seja um modelo linear ou não-linear, para modelar um problema, é necessário encontrar métricas que avaliem o desempenho dos modelos ou a qualidade das predições sobre N exemplos de dados. O erro médio quadrático (MSE, *Mean Squared Error*) é comumente utilizado como função de custo, o qual auxilia no processo de ajuste da função aos dados (BISHOP, 2006). O MSE é calculado como:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(f(x_i) - y_i')^2 \tag{2.4}$$

onde $f(x_i)$ é a i-éssima saída do modelo e y_i' a saída desejada.

MSE também pode ser utilizado como métrica, entretanto, devido ao seu valor quadrático, não é útil para a interpretação dos resultados. Assim, a raiz do MSE (RMSE, *Root Mean Squared Error*) é conveniente nesses casos e é calculada como:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i')^2}. \tag{2.5}$$

O coeficiente de Correlação de Pearson ou apenas coeficiente de Person mede a relação entre duas variáveis, onde os valores variam entre -1 e 1. Se o coeficiente for

igual 1, a relação é linear crescente. Se for igual a -1, a relação é linear decrescente. O coeficiente é calculado como:

$$\rho = \frac{\sum_{i=1}^{n}(y) - \bar{y})(y_i' - \bar{y}')}{\sqrt{\sum_{i=1}^{n}(y - \bar{y})^2 \sum_{i=1}^{n}(y_i' - \bar{y}')^2}} \tag{2.6}$$

onde $\bar{y}$ e $\bar{y}'$ são os valores médios de y e y'. Note que ρ está sendo calculado sobre a saída do modelo e a variável desejada.

2.2 REDES NEURAIS ARTIFICIAIS

As redes neurais artificiais, as quais são conhecidas apenas por redes neurais ou RNAs, são modelos de aprendizagem de máquina amplamente utilizadas em diversas aplicações. Embora o seu uso tenha sido difundido apenas nas últimas décadas, o seu desenvolvimento se iniciou na década de 40 com o trabalho pioneiro de McCulloch e Pitts (1943). Neste trabalho, foi proposto o primeiro modelo formal de neurônio a partir da análise das atividades do sistema nervoso com o uso de lógica proposicional. O seu caráter baseado na lógica proposicional desenvolveu um modelo binário.

Um próximo marco para as redes neurais, não desconsiderando os vários avanços produzidos até então, foi a introdução das redes neurais *Perceptron*, ou apenas Perceptron, proposto por Rosenblatt (1958). O *Perceptron* foi a primeira rede capaz de reconhecer padrões linearmente separáveis utilizando um único neurônio ligado às entradas de dados. Ele é o precursor do desenvolvimento das arquiteturas modernas de redes neurais. Pode-se atribuir a ele o título de primeira rede que usa aprendizagem, o que não é oferecido no trabalho de McCulloch e Pitts (1943).

Por meio das *Perceptrons*, uma das arquiteturas mais utilizadas no reconhecimento de padrões em diversos problemas (SANTOS et al., 2005), foi concebido as redes neurais Perceptron de múltiplas camadas. Estas redes superam os limites das Perceptrons, o quais apenas poderiam mapear problemas lineares (MINSKY; PAPERT, 1969). O sucesso dessas redes é devido ao uso eficiente do algoritmo de *backpropagation* o que permitiu o seu treinamento, uma tarefa até então desconhecida (WERBOS, 1974; RUMELHART; HINTON; WILLIAMS, 1986).

Os primeiros estudos, em sua maioria, tentavam entender o funcionamento neural. Como consequência, desenvolveram-se métodos poderosos em diversas áreas como, por exemplo, visão computacional (KRIZHEVSKY; SUTSKEVER; HINTON, 2012), reconhecimento de voz (MIAO ZHENJIANG; YUAN BAOZONG, 1994) e problemas combinatoriais (ESPINOSA-MENESES et al., 2019).

Uma breve introdução das redes neurais de múltiplas camadas (MLPs, *Multilayer Perceptron* - Perceptron de Múltiplas Camadas) é dada nas subseções seguintes. Assim, na Subseção 2.2.1, a arquitetura geral e os principais componentes das MLPs são apresentados. Na subseção 2.2.2, o paradigma de aprendizagem dessas redes é exposto. Por fim, na subseção 2.2.3, o algoritmo de treinamento das MLPs é descrito.

2.2.1 Perceptron de Múltiplas Camadas

As RNAs apresentadas neste trabalho são as redes alimentadas adiante, conhecidas na literatura como *Perceptron* de múltiplas camadas. O termo alimentada adiante é devido ao fluxo linear de computação realizado nos dados de entrada. Especificamente, a computação começa a partir da camada de entrada repassando os dados continuamente às camadas intermediárias que, por fim, são computadas pela camada de saída. Exemplos de redes que usam este fluxo linear são as Redes Neurais Convolucionais (*Convolucional Neural Network*) geralmente usadas para reconhecimento de imagens.

O elemento principal das MLPs são os neurônios. Definidos primeiramente no trabalho de McCulloch e Pitts (1943), eles são unidades simples de processamento que recebem como entrada a saída da computação dos neurônios conectados anteriormente a eles. Para isso, em cada neurônio existe um conjunto de conexões valoradas (pesos ou parâmetros) de modo que o neurônio computa a entrada somando todos os pesos e, após isso, aplica uma transformação chamada função de ativação. Formalmente um neurônio é definido como:

$$u = \sum_{i=1}^{n} (w_i x_i) + b \tag{2.7}$$

e

$$v = \alpha(u). \tag{2.8}$$

Na equação 2.7, a entrada representada por x_i refere-se a i-ésima entrada do neurônio u. O peso correspondente à entrada x_i é representado por w_i e, diferente do peso de um neurônio biológico, w_i pode assumir valores negativos. Além disso, cada neurônio tem uma constante atrelada b. O seu propósito é ajustar a reta formada por u (HAYKIN, 2009).

Eles ainda são organizados de forma hierárquica em camadas as quais têm um número arbitrário de neurônios. A primeira camada recebe como entrada os dados que serão processados pela rede. O processamento de todos os neurônios dessa camada serão repassados para a próxima repetindo esse processo até a última camada, que produzirá a saída da rede. O número de neurônios na camada de saída é a quantidade de classes C, no caso de um classificador, ou somente um neurônio para uma regressão. A quantidade de camadas entre a primeira e última está relacionado com a profundidade da MLP (GOODFELLOW; BENGIO; COURVILLE, 2016). As camadas podem ser representadas como uma composição de funções, como mostrado na equação 2.9.

$$f(X) = f^{L+1}(f^{L}(f^{L-1}(...f^{1}(X)))) \tag{2.9}$$

Retornando à equação 2.8, $\alpha(u)$ é conhecida como função de ativação do neurônio sendo baseada no conceito das atividades nervosas em que neurônios são disparados segundo o seu potencial elétrico. A função de ativação $\alpha(u)$ tem o objetivo de criar funções não lineares para o neurônio u (BEZERRA, 2016) para que seja possível modelar problemas não-lineares (SHARMA; SHARMA; ATHAIYA, 2020). Além disso, esta função limita o espaço de saída (HAYKIN, 2003). Comumente, as funções de ativação utilizadas são as sigmoides (logística e hiperbólica) e a função retificadora linear (*Rectified Linear Unit*, ReLU). A escolha adequada da função de ativação impacta o treinamento e, consequentemente, o desempenho da rede (HAYOU; DOUCET; ROUSSEAU, 2018). A Figura 1 traça o gráfico das funções

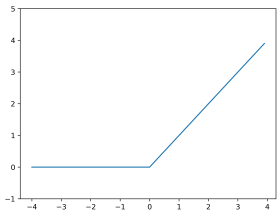
de ativação e as equações 2.10, 2.11, 2.12 apresentam as funções de ativação ReLU, logística e hiperbólica respectivamente.

$$\alpha(u) = max(0, u) \tag{2.10}$$

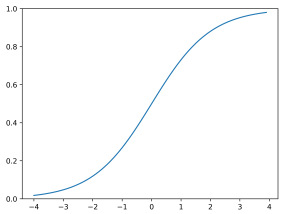
$$\alpha(u) = \frac{1}{1 + e^{-u}} \tag{2.11}$$

$$\alpha(u) = \frac{e^{u} - e^{-u}}{e^{u} + e^{-u}} \tag{2.12}$$

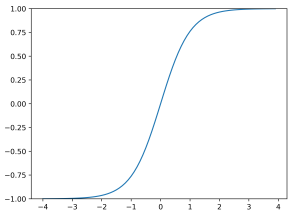
Figura 1: Funções de ativação comumente usadas para a construção de modelos de redes neurais MLP.



(a) ReLU



(b) Logística



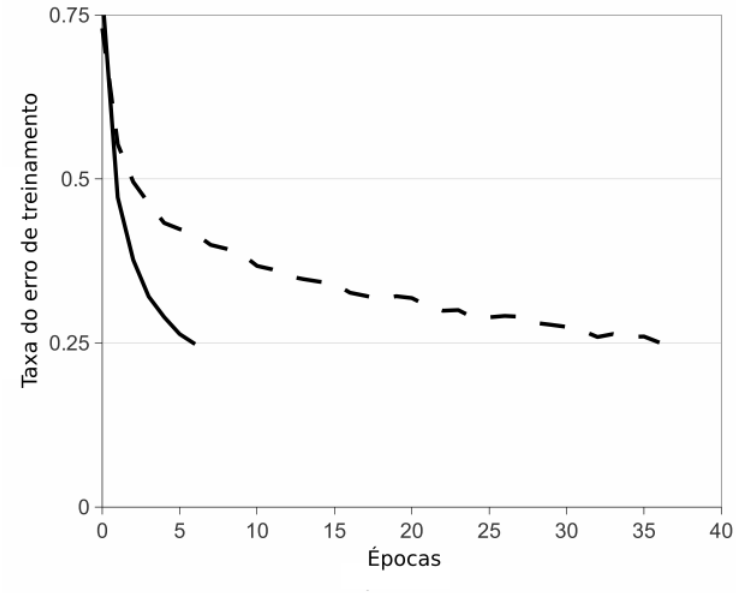
(b) Hiperbólica

Fonte: Autoria própria

Vale ressaltar que, durante o desenvolvimento das MLPs ou redes profundas, observou-se que o uso de funções de ativação do tipo sigmoide nas camadas ocultas durante o treinamento da rede tendem a saturar o gradiente da função de ativação em um dado momento (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Como consequência, o tempo de convergência é maior. Na Figura 2, as curvas de treinamento da ReLU (linha sólida) e da logística (linha pontilhada) são traçadas. Elas representam a relação de taxa de erro de treinamento (é igual a $1 - taxa_{acerto}$) pelo número de épocas (definida na seção 2.2.3). Assim, a Figura 2 ilustra a eficiência da ReLU sobre a função logística em redes de múltiplas camadas.

De forma geral, as MLPs podem ser representadas como um grafo orientado onde os vértices do grafo são os neurônios e as arestas (ponderadas) são suas conexões. Comumente, essa representação, ilustrada na Figura 3, é chamada de arquitetura, pois define como os neurônios estão organizados. Para simplificar a ilustração, os pesos associados a cada neurônio foram omitidos e os *bias* deixados implícitos. Esse

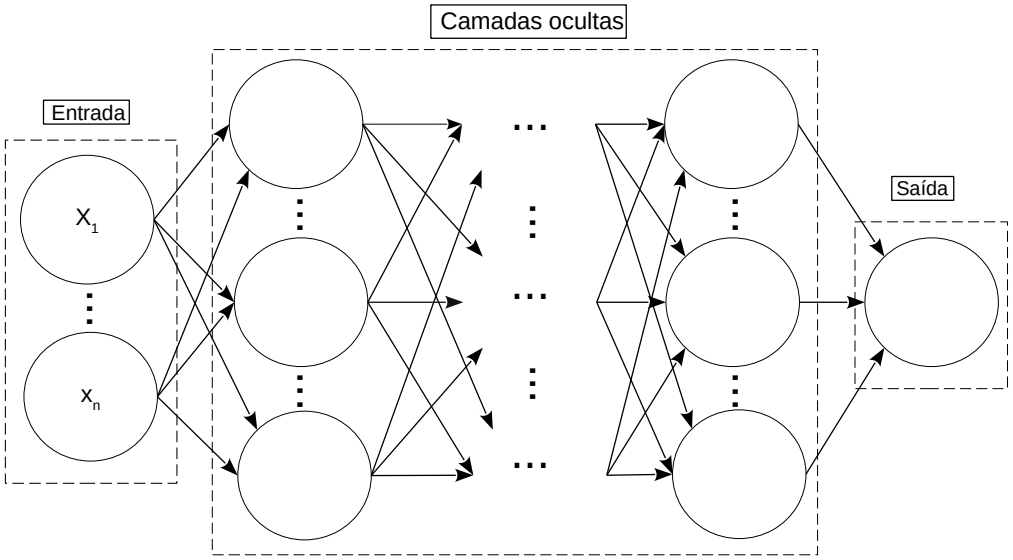
Figura 2: Curvas de comportamento do treinamento de uma rede profunda.



Fonte: Traduzido de (KRIZHEVSKY; SUTSKEVER; HINTON, 2012).

modelo representa as MLPs para problemas de regressão.

Figura 3: Representação da MLP como um grafo dirigido acíclico.



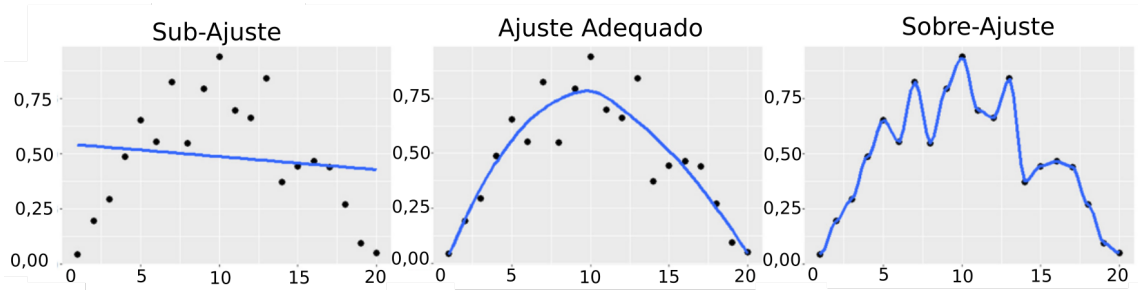
Fonte: autoria própria.

2.2.2 Aprendizagem em Redes Neurais

Conforme é explanado em NATARAJAN (1991), a aprendizagem é o processo em que um modelo de aprendizagem de máquina constrói uma aproximação adequada de um conceito desconhecido através de uma amostra de exemplos do domínio do problema. Especificamente, o modelo aprende uma função que se aproxima a uma função desconhecida. Diz-se que uma função generaliza se a mesma produz resultados adequados para exemplos ainda não vistos.

A Figura 4 ilustra três possibilidades de ajuste da função de aprendizagem: sub-ajuste (curva da esquerda, *underfitting*), ajuste adequado (curva do meio) e sobre-ajuste (curva da direita). Conforme Mutasa, Sun e Ha (2020), o sub-ajuste ocorre quando a função não aprendeu padrões suficientes nos dados. O segundo caso é quando a função obtém um bom desempenho para dados não vistos (generaliza). Por fim, o sobre-ajuste ocorre quando a função não obtém um bom desempenho para dados não vistos.

Figura 4: Formas de ajuste da curva da função de aprendizagem.



Fonte: Traduzido de (MUTASA; SUN; HA, 2020)..

RNAs, no que lhe concernem, aprendem padrões em dados por meio do ajuste dos seus parâmetros livres pela interação com o ambiente (MENDEL; MCLAREN, 1970). Este processo descrito é definido na literatura como um processo de aprendizagem (HAYKIN, 2003). Exitem diversas formas de ajustar os pesos em uma rede neural. Um processo padrão de ajuste de pesos em redes de múltiplas camadas baseado em Gradiente Descendente e *backpropagation* é apresentado na Subseção 2.2.3.

Existem dois paradigmas de aprendizado de máquina que são frequentemente

usados em redes neurais: aprendizado supervisionado e não-supervisionado. Neste trabalho é definido apenas a aprendizagem supervisionada, pois, segundo o estado da arte, é o frequentemente utilizado para a solução do problema de estimativa/previsão de produtividade.

No paradigma de aprendizado supervisionado, a RNA, aprenderá através de um conjunto de dados rotulados $D = \{(X_1, y_1), ..., (X_n, y_n)\}$, chamado conjuntos de treinamento (MELLO; PONTI, 2018; MURPHY, 2012). Nestes conjuntos, X_i é um vetor de característica $[x_1, ..., x_n]$ com $x_i \in \mathbb{R}$ representando um objeto de interesse e $y \in \{1, ..., C\}$, $C \in \mathbb{N}$, ou $y \in \mathbb{R}$. No primeiro caso, a RNA é um classificador com C sendo a quantidade de classes existente para a classificação. No segundo caso, a RNA é um regressor. Neste trabalho, considera-se a RNA como um regressor devido à natureza do problema que requer um valor real (produtividade $\in \mathbb{R}$) e não uma classe.

Para todos os casos (classificador ou regressor), deseja-se uma função $f : \mathbb{R} \to \mathbb{R}$ que estime $y = f(X)$ (ZOU; HAN; SO, 2008). Note-se que, para bases de dados que contenham variáveis categóricas (não numéricas), as mesmas devem ser mapeadas para alguma representação numérica, uma vez que as RNAs aprendem sobre dados numéricos.

2.2.3 Treinamento

Dado a definição dos principais componentes das MLPs e o tipo de aprendizagem (supervisionada), a rede ainda não é capaz de produzir generalizações sobre os dados. Isso ocorre porque a rede ainda não foi treinada adequadamente sobre o conjunto de dados, visto que os pesos da rede são inicialmente fixados aleatoriamente ou por algoritmos que auxiliam na convergência da rede como, por exemplo, o algoritmo proposto por Glorot e Bengio (2010). Consequentemente, um erro grande é obtido entre a saída da rede e as amostras de dados.

Antes de se ajustar os pesos, uma função de custo J das predições realizadas pela MLP é definido. A função de custo mede a qualidade da predição y' em relação

à predição real y (PONTI; COSTA, 2018). A escolha de J dependerá do domínio do problema. Comumente, as funções de custo utilizadas para regressão são o MSE e o MAE (Mean Absolute Error - Error Absoluto Médio). Outras funções de custo e análises empíricas são apresentadas no trabalho de Lathuiliere et al. (2020).

Com a função de custo apropriada definida, a descida do gradiente é aplicada com o objetivo de minimizar o custo da função J. RNAs que utilizam esse tipo de aprendizagem são conhecidas como aprendizagem baseada em gradiente descendente (GOODFELLOW; BENGIO; COURVILLE, 2016). A descida do gradiente é um método padrão de atualização gradual dos pesos. Entretanto, devido ao seu alto custo computacional para o cálculo das derivadas parciais para todos os exemplos da base de dados, variantes que escolhem m exemplos estocasticamente foram propostas.

O hiperparâmetro m define o tamanho do mini lote (*batch size*) de treinamento. Outros hiperparâmetros importantes na fase de treinamento é o número de ciclos (ou épocas) e a taxa de aprendizagem. As épocas definem o número de vezes que os otimizadores irão executar sobre a base de dados e a taxa de aprendizagem o tamanho do passo que os pesos serão ajustados.

Alguns otimizadores variante são: Gradiente Descendente Estocástico (Stochastic Gradient Descent, SGD), Propagação da Raiz Quadrada Média (*Root Mean Square Propagation*, RMSProp) e Momentos Adaptativos (adaptive Moments, Adam). O trabalho de Lathuiliere et al. (2020) também explora outros otimizadores em problemas de regressão.

O otimizador é utilizado em conjunto com o algoritmo *backpropagation*. O *backpropagation* é um algoritmo iterativo que atualiza sistematicamente os pesos a partir da retropropagação do erro (RUMELHART; HINTON; WILLIAMS, 1986). Conforme Haykin (2003), o algoritmo de *backpropagation* é, em geral, dividido em dois passos. Assim, dado um exemplo i:

1. propagação: a propagação inicia-se com a entrada i enviando o sinal até a camada de saída da rede. O erro entre a saída da rede y' e a saída desejada y

é gerado e repassado para as camadas anteriores para atualização dos pesos;

2. retropropagação: o erro gerado no passo anterior é utilizado para a camada anterior o qual atualizará os pesos desta camada. Após a atualização dos pesos, o erro é repassado para as camadas anteriores até que todos os pesos sejam atualizados.

Os passos descritos são repetidos pelo tamanho da amostra e serão executados pelo número de épocas definido.

2.3 QUESTÕES IMPLEMENTACIONAIS DOS MODELOS

Existem questões importantes a serem consideradas na implementação de qualquer modelo de aprendizagem de máquina e na base de dados. O pré-processamento, discutido na seção 2.3.1, efetua algumas transformações necessárias nos dados antes de serem repassados ao algoritmo de aprendizagem. Os pré-processamentos que serão discutidos e aplicados neste trabalho são: imputação de dados faltantes, codificação de variáveis categóricas e normalização do conjunto de dados de entrada para o modelo.

As RNAs possuem muitos hiperparâmetros a serem definidos durante a fase de implementação do modelo. Na seção 2.3.2, uma técnica para a otimização automática dos hiperparâmetros é apresentada. Além disso, na seção 2.3.3, o procedimento *k-fold cross-validation* é apresentado como método utilizado na validação do desempenho obtido no modelo.

2.3.1 Pré-processamento

Muitas bases de dados requerem atenção antes mesmo de serem repassadas para modelos de aprendizado de máquina. Essas bases possuem frequentemente dados duplicados, informações faltantes, ruídos nos dados e dados em diferentes escalas (BATISTA, 2003). Isso dificulta ou impede o processo de treinamento dos algoritmos de aprendizado supervisionado. Assim, os processos de pré-processamento ou

engenharia de dados são, então, uma etapa necessária para manter a qualidade do conjunto de dados.

O pré-processamento engloba um conjunto de técnicas que lidam com as nuances contidas nessas bases. Algumas ações exigem pouco esforço por parte do engenheiro de dados, como, por exemplo, a exclusão de dados duplicados, e outras uma análise mais cuidadosa nos dados, a imputação de dados faltantes.

Dados faltantes estão frequentemente presentes em base de dados reais. Considerando um conjunto de dados representado como uma matriz de duas dimensões, onde as linhas são exemplos/observações e as colunas as variáveis/atributos de um dado problema, um dado faltante é alguma informação desconhecida de um atributo. As razões que levam a ausência desses dados podem ser diversas: falha humana, falha no instrumento de captura dos dados, a não resposta a pesquisas realizadas em campo, entre outras. Considerando as MLPs, elas não permitem como entrada esse tipo de dados, sendo necessários exemplos com dados completos. Uma forma de lidar com essa situação é a exclusão desses exemplos do conjunto de dados. Em bases com uma quantidade relativamente pequena de exemplos, isso pode acarretar perda de informação podendo, consequentemente, diminuir o desempenho dos modelos.

Outra forma de lidar com esse problema é a imputação de dados. A imputação de dados substitui o valor faltante por outro. As técnicas de imputação em sua maioria podem ser divididas em imputação única e múltipla. A imputação única faz-se somente uma vez a imputação de dados estimando o valor com base nas variáveis que estão completas. Por outro lado, a imputação múltipla cria m bases de dados imputados a fim de lidar com a incerteza inerente de se imputar dados faltantes (RUBIN, 1996). A imputação única é a técnica utilizada nesse trabalho.

A técnica adotada para imputação dos dados é baseado no algoritmo *Expactation Maximization* sendo conhecida como *iteration inputation*. A ideia geral é estimar a máxima verossimilhança por meio das predições realizadas dos dados faltantes. Esse método tenta diminuir, a cada iteração, o erro das predições adicionando sempre exemplos estimados no conjunto de dados. O algoritmo 1, apresentado por Richman, Trafalis e Adrianto (2009), descreve o método de imputação dos dados. Como é

observado no algoritmo 1, as estimativas dos dados faltantes é realizada criando um regressor com base nas variáveis completas da base. O método não define um regressor fixo, ele pode ser definido segundo as características do problema, não se limitando à regressores lineares. Outra característica do algoritmo é a escolha do δ_{min} que faz parte da condição de parada.

Algoritmo 1 Procedimento para a imputação de dados.

1: Crie um novo conjunto de dados d somente com os exemplos sem valores faltantes;
2: Crie um novo conjunto de dados d' somente com os exemplos com valores faltantes;
3: Para cada coluna n em d', crie uma função de regressão g com base em d;
4: Estime os valores faltantes por meio da função de regressão g definida no passo anterior;
5: Impute os valores estimados em d';
6: Combine os exemplos estimados em d;
7: Novamente, para cada coluna n em d', construa uma função de regressão g com base no novo d;
8: Estime os valores faltantes com a nova função de regressão definida no passo anterior;
9: Combine os exemplos estimados em d;
10: Calcule o erro para cada coluna com função de erro δ;
11: Pare se $\delta \leq \delta_{min}$, caso contrário volte a passo 7;

Outra tarefa a ser realizada no pré-processamento é transformar as variáveis qualitativas ou categóricas para dados numéricos, pois as RNAs não permitem dados categóricos como entrada da rede. A transformação adotada é apresentada na seção 3.1.1.

Por fim, as variáveis da base de dados podem estar em escalas diferentes. Muitos algoritmos de aprendizagem podem não convergir devido a esse problema. O trabalho de Glorot e Bengio (2010) analisou as dificuldades para o treinamento de redes profundas alimentadas adiante. Ele observou que as variáveis nesse estado di-

ficultam o processo de treinamento da rede no sentido do tempo de convergência ser maior quando os dados não estão normalizados. A normalização permite que todas as variáveis estejam na mesma escala sendo a normalização MinMax a comumente utilizada. Ela transforma os dados em um intervalo de [0,1] por meio da equação 2.13 .

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{2.13}$$

onde, x' é novo valor de x ,x é um valor de um atributo t, x_{max} e x_{min} são os valores máximo e mínimo de t respectivamente.

2.3.2 Otimização de hiperparâmetros

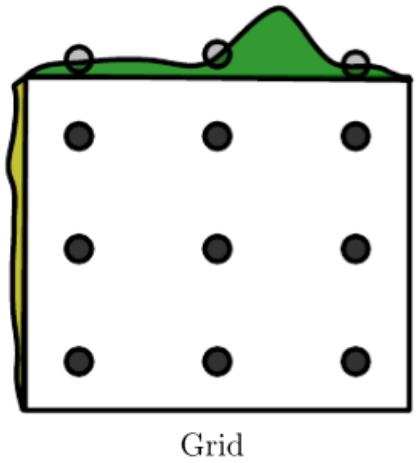
Algoritmos de aprendizado de máquina exigem muitos parâmetros a serem definidos a priori, isto é, parâmetros, chamados hiperparâmetros, que não são aprendidos durante o treinamento do modelo. Nas MLPs, função de ativação, números de épocas, otimizador, taxa de aprendizagem, função de custo, método de inicialização, camadas ocultas, quantidades de neurônios em cada camada e *batch size* são exemplos de hiperparâmetros. Eles exercem um impacto significativo no desempenho dos modelos (GOODFELLOW; BENGIO; COURVILLE, 2016).

Como citado anteriormente, existem muitos hiperparâmetros a serem configurados a priori. A fim de obter o conjunto de hiperparâmetros ótimos, a configuração exige um esforço e um domínio sobre o problema que muitas vezes pode ser um processo demorado (FEURER; HUTTER, 2019). A otimização de hiperparâmetros são conjuntos de técnicas que otimizam sistematicamente um conjunto de hiperparâmetros definidos pelo usuário a fim de encontrar os hiperparâmetros com o melhor desempenho.

Das técnicas mais simples de serem implementadas, o *Grid Search* é a mais comum. *Grid Search* faz uma busca exaustiva em todos os hiperparâmetros definidos (Figura 5). Nela, os hiperparâmetros ótimos serão aqueles que tem o melhor desempenho. Especificamente, dado os conjuntos que definem os hiperparâmetros,

é realizado o produto cartesiano desses conjuntos e repassado para algoritmo de aprendizado.

Figura 5: Otimização de hiperparâmetros por meio do *Grid Search*.



Fonte: Adaptado (BERGSTRA; BENGIO, 2012).

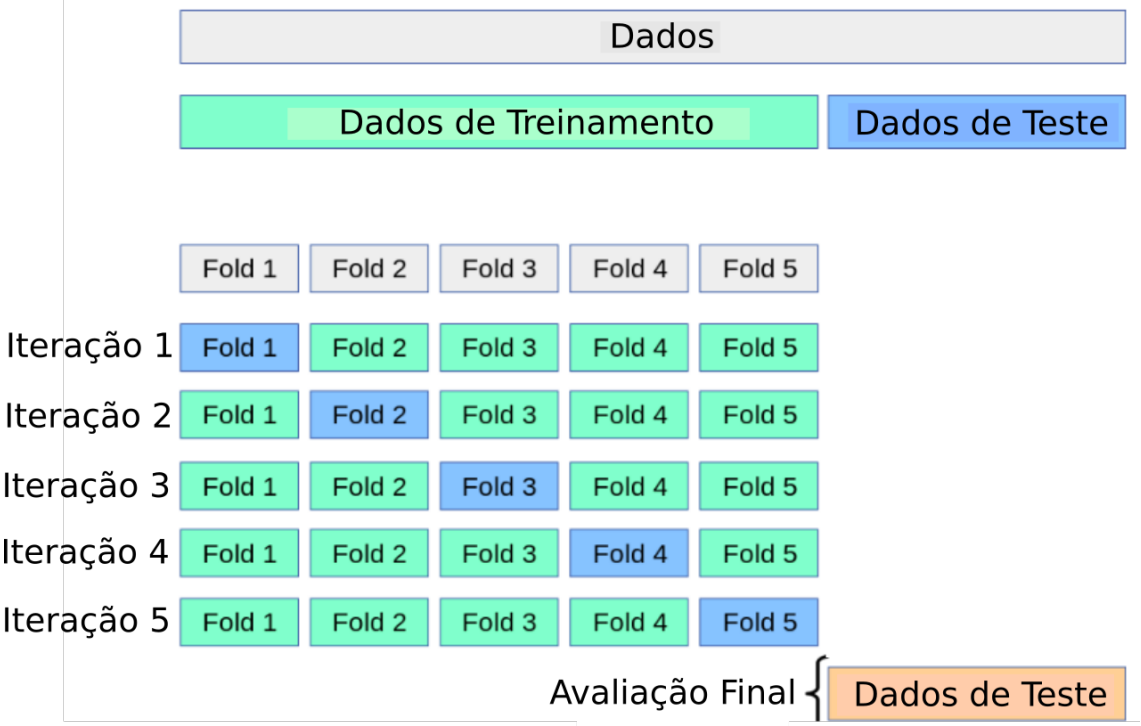
2.3.3 Validação Cruzada

A validação dos modelos de aprendizagem de máquina, referente ao conjunto de dados, dá-se pelo particionamento do conjunto de dados em três partes: conjunto de treinamento, de teste e de validação. A aplicação desse particionamento em um conjunto de dados relativamente pequeno pode levar a resultados não confiáveis (BISHOP, 2006).

Um procedimento frequentemente utilizado para a solução desse problema é o *k-fold cross-validation* (*k-fold*). O *k-fold* divide o conjunto de dados em k conjuntos. Assim, treina k modelos com os $k - 1$ conjuntos e, em cada iteração do *k-fold*, o conjunto restante é utilizado como teste. A avaliação geral é dada pela média dos k modelos (SHALEV-SHWARTZ; BEN-DAVID, 2013). A Figura 6 ilustra o procedimento de particionamento do *k-fold*.

Na Figura 6, o conjunto de dados é disposto em uma lista com todos os exemplos. Assim, o *k-fold* vai particioná-la em k partes formando os conjuntos de exemplos. Os $k - 1$ conjuntos serão usados para treinamento do modelo e o k-ésimo restante para teste. Os componentes de $k - 1$ serão mudados um de cada vez até formar os

Figura 6: Processo de particionamento do conjunto de dados *k-fold*.



Fonte: Adaptado e traduzido de (PEDREGOSA et al., 2011).

k modelos e cada modelo terá sua própria avaliação das métricas. O valor de k é frequentemente escolhido como 5, 10 ou 20 (ANGUITA et al., 2012).

2.4 CULTURA DO MILHO

O milho (Zea mays L.) é considerado a maior cultura agrícola do mundo, ultrapassando cerca 1 milhão de toneladas por safra. Os principais países que produzem milho são os Estados Unidos, a China e o Brasil (CONTINI et al., 2019). No Brasil, a safra de 2020/2021 obteve 104,9 milhões de toneladas em uma área de 18.442,2 mil hectares contra os 80.709,5 milhões da safra de 2017/2018 em uma área de 11,6 mil hectares (CONAB, 2020). Contudo, o milho enfrenta algumas barreiras no seu desenvolvimento. Uma delas está relacionada à baixa produtividade em algumas regiões do Brasil (MIRANDA; LÍCIO, 2014).

Os contínuos investimentos nessa cultura é devido ao consumo humano do cereal que está presente na culinária mundial. Além deste consumo, o milho ainda é

utilizado como ração para animais, sendo estes os maiores consumidores (CONTINI et al., 2019).

A produtividade do milho é definida em função da interação da genética e do ambiente. Para alcançar o potencial produtivo definido pela genética é necessário que as condições do ambiente estejam de acordo com as necessidades do desenvolvimento da planta. Nas subseções seguintes é definido o ciclo de desenvolvimento bem como os fatores que influenciam as capacidades da produção do milho.

2.4.1 Estádios de Crescimento/Desenvolvimento

A identificação dos estádios do desenvolvimento da planta de milho é dividido em vegetativo (V) e reprodutivo (R). Para cada fase, há uma subdivisão desse período representado pelo número n (MAGALHÃES; DURÃES, 2006). O estádio vegetativo refere-se, em geral, ao desenvolvimento das folhas e o estádio reprodutivo ao desenvolvimento dos grãos da espiga. A duração de cada estádio depende do tipo de cultivar do milho. A Tabela 1 apresenta os estádios de desenvolvimento da planta de milho.

Tabela 1: Estádios vegetativos e reprodutivos da planta de milho.

Vegetativo	Reprodutivo
VE, emergência	R1, Embonecamento
V1, 1ª folha desenvolvida	R2, Bolha d'água
V2, 2ª folha desenvolvida	R3, Leitoso
V3, 3ª folha desenvolvida	R4, Pastoso
V4, 4ª folha desenvolvida	R5, Formação de dente
V(n), nª folha desenvolvida	R6, Maturidade Fisiológica
VT, pendoamento	-

Fonte: Retirado de (MAGALHÃES; DURÃES, 2006).

Conforme a Tabela 1, o primeiro estádio vegetativo é o vegetativo emergencial (VE). Ele compreende o período entre a semeadura e a emergência, o que decorre em duas semanas sob uma temperatura de 15ºC (BERGAMACHI; MATZENAUER,

2014). Neste estádio, a planta cresce em torno de 2,5 a 4,0 cm no solo (MAGALHÃES; DURÃES, 2006). Os estádios de V1 a V(n) referem-se ao crescimento de cada folha que é afetado drasticamente pela temperatura do solo (BERGAMACHI; MATZENAUER, 2014). O último estádio vegetativo é o VT (pendoamento). A duração dessa fase depende do ciclo da cultivar e das condições climáticas (RITCHIE; HANWAY; BERSON, 1986).

A primeira fase do estádio reprodutivo é o embonecamento (R1). Ela se inicia quando todos os estilos-estigmas ("cabelo" da espiga) estão formados permitindo, assim, o início da polinização, onde o estilo-estigma captura um grão de pólen e começa a fertilizar o óvulo das espigas. Nos estágios seguintes (R2, R3, R4 e R5), o grão de milho começa a se desenvolver mudando seu amadurecimento. Na última fase (R6), os grãos já alcançaram o seu limite máximo de peso seco (BERGAMACHI; MATZENAUER, 2014).

É importante ressaltar que, em outras literaturas, a identificação dos estádios podem mudar. Em algumas literaturas são especificados mais estádios vegetativo e reprodutivo, como a floração e a maturação.

2.4.2 Fatores que Afetam a Produtividade do Milho

A produtividade de qualquer cultura depende de muitos fatores para que o seu potencial genético seja alcançado. Isso inclui o genótipo, o clima e o manejo. Especificamente, o crescimento do milho está limitado por disponibilidade hídrica, temperatura, radiação solar e época de plantio (CRUZ et al., 2006). A seguir são descritos alguns destes fatores.

A temperatura afeta todo o ciclo de desenvolvimento da planta. A temperatura do ambiente deve estar entre 10° e 30°C. Temperaturas abaixo disso limitam o crescimento da planta ou, acima, diminuem o rendimento dos grãos. O ideal é em torno de 24º e 30ºC (MONTEIRO, 2009; CRUZ et al., 2006).

O milho é uma das culturas que mais necessitam de água durante os estádios de desenvolvimento. O consumo diário é de 5 a 7,5 mm de água (PEREIRA FILHO,

2002). Um déficit hídrico no estádio R1, por exemplo, tem como consequência a redução do rendimento dos grãos.

O rendimento e a produtividade também são afetados pela radiação solar devido ao milho ser uma espécie com o metabolismo fotossintético. A redução da radiação solar em dias nublados, por exemplo, retardam a maturação dos grãos comparados aos dias onde a radiação solar pode ser considerada normal (MONTEIRO, 2009).

Outro aspecto diz respeito à época de semeadura. Procura-se por datas de semeadura onde as condições climáticas são ótimas para os estádios V e R. Isto é, com dias mais longos, período de temperaturas mais elevadas e alta disponibilidade de radiação solar (CRUZ et al., 2006).

2.5 ESTADO DA ARTE

As RNAs têm sido aplicadas de várias formas na previsão ou na estimativa da produtividade de culturas agronômicas. Alguns trabalhos focam em características específicas do ambiente, como clima e solo, ou genótipo como a genética da planta. No Brasil, as estimativas vêm sendo aplicadas em diferentes localidades e com diferentes características para o cultivo.

Soares et al. (2015) avaliaram o desempenho da MLP para a previsão da produtividade da cultura do milho, no município de Jaguari, região central do Estado do Rio Grande do Sul, Brasil. Com a hipótese de que as variáveis morfológicas (índice de área foliar, matéria verde total, altura de planta e número de plantas) pudessem predizer a produtividade. Os resultados obtidos demonstram que a hipótese era valida obtendo, para o conjunto de validação, um erro pequeno.

Leal et al. (2015) propuseram uma rede neural com base somente em atributos de solos para a predição da produtividade do milho e, por meio disso, entender a influência dessas variáveis na produtividade para definição de um sítio de manejo diferenciado. Os dados para treinamento da MLP foram de experimentos conduzidos em 2010 e 2011 no município Chapadão do Céu, Goiás. Os atributos considerados foram: teor de matéria orgânica, capacidade de troca catiônica, saturação de bases

e teor de argila. O modelo foi comparado com uma regressão linear múltipla. Os resultados da MLP foram superiores aos da regressão linear. Leal et al. (2015) sugerem que, para diminuir os erros da rede, é necessário incluir outras variáveis e testar outras arquiteturas.

Alves (2016) aplicou uma MLP para estimativa da soja como base nos hábitos de crescimento, densidade de semeadura e características agronômicas. Os dados foram extraídos de um experimento conduzido na safra 2013/2014 em Anápolis, Goiás. Com um conjunto de dados de 65 exemplos, a rede obteve índice de acerto de 98% para os dados de treinamento e 72% para os dados de validação.

Michelon (2016) avaliou duas técnicas de aprendizado de máquina, redes neurais artificiais (do tipo MLP) e máquina de vetores de suporte, para a estimativa da produtividade da soja em função dos macronutrientes da folha da soja. Além disso, aplicou uma técnica para a redução de dimensionalidade. Os dados foram coletados em 2012 no município de Serranópolis do Iguaçu, Paraná, em duas áreas experimentais. Foi obtido um erro de 0,19676 $kg \cdot ha^{-1}$ para a MLP. O autor sugere incluir novas características para a predição como: características da planta e do solo, como os componentes químicos, físicos e de relevo do solo; o tamanho das folhas e a quantidade de grãos por planta em diferentes estágios.

Guimarães (2019) criou modelos de predição da produtividade da soja com base nas características de solo e clima de três biomas (Amazônia, Pantanal e Cerrado) do estado do Mato Grosso. As variáveis viáveis consideradas em cada estádio de desenvolvimento da planta foram: evapotranspiração de referência, coeficiente da cultura, evapotranspiração da cultura, precipitação pluvial, armazenamento de água no solo, alteração do armazenamento, evapotranspiração real, perda e produtividade estimada, temperatura, capacidade de água disponível, razão entre evapotranspiração real e evapotranspiração da cultura, precipitação, deficiência hídrica e temperatura máxima e mínima. Utilizou-se três modelos de aprendizado de máquina: redes neurais artificiais do tipo MLP, *random florest* e *extreme gradient boosting*. No total, foram 27 municípios selecionados com base no menor risco climático para o cultivo da soja. O conjunto de dados são do período de 2010 a 2018 contendo 29160

exemplos. Foi obtido um erro de 47,34 $kg \cdot ha^{-1}$ para a MLP.

O trabalho de Santos (2020) estimou e predisse a produtividade anual da soja na região do MATOPIBA (Maranhão, Tocantins, Piauí e Bahia) em função das variáveis climáticas mensais (temperatura do ar, precipitação, radiação global) e dos componentes de balanço hídricos (evapotranspiração de cultivo, armazenamento, evapotranspiração real de cultivo, deficiência e excedentes hídricos) durante o desenvolvimento da planta por meio de uma rede neural artificial profunda. O tamanho do conjunto de dados foi de 920 exemplos. Este trabalho não obteve um erro pequeno. Em média, ele obteve um erro de 167,85 $kg \cdot ha^{-1}$.

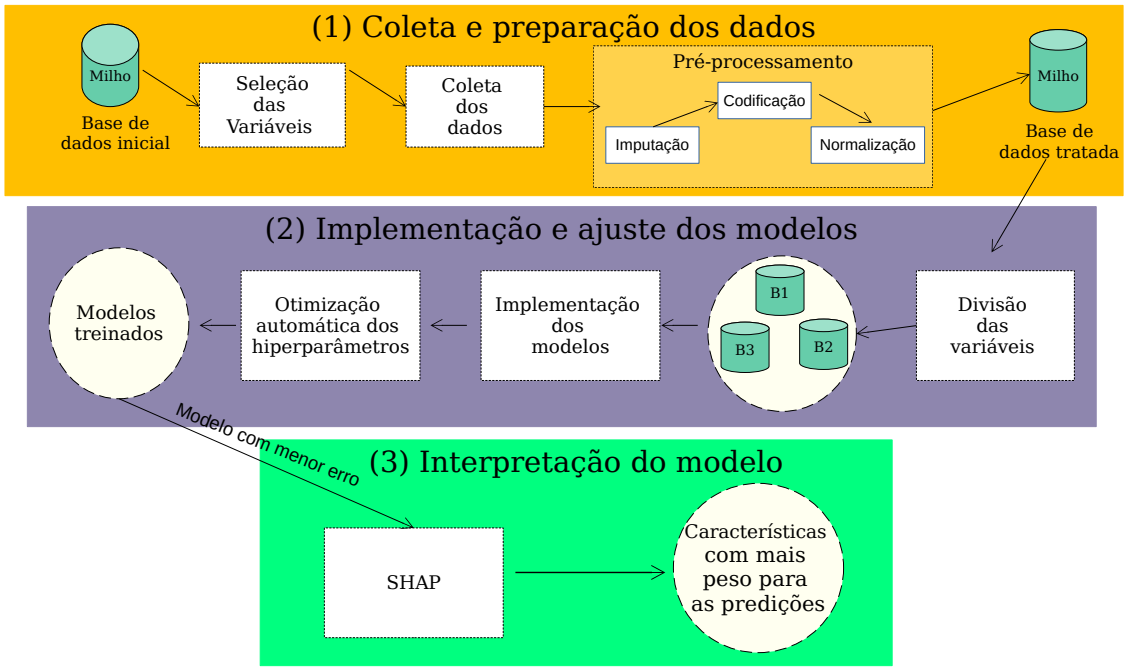
Trautmann (2020) desenvolveu vários modelos de aprendizagem de máquina para a estimativa da produtividade do trigo. Em relação às RNAs, esse trabalho desenvolveu um modelo em função do nitrogênio e dos fatores ambientais para a predição da produtividade do trigo em dois tipos de sistemas de sucessão, soja/trigo e milho/trigo. Os dados para o treinamento e a validação do modelo são provenientes do município Augusto Pestana-RS no período de 2012 a 2018. A MLP construída obteve um erro 100 $kg \cdot ha^{-1}$

Dentre os trabalhos citados nesta seção, observou-se que o uso das RNAs foi viável tanto para um conjunto de variáveis pequeno quanto para um grande, obtendo um erro médio aceitável. Para um conjunto de variáveis menor que 100 exemplos, a rede não conseguiu generalizar para os dados de validação, ocorrendo um sobre ajuste aos dados. Isso ocorreu nos trabalhos de Alves (2016) e Leal et al. (2015). Espera-se que um conjunto de dados acima de 500 exemplos, considerando diferentes fatores como entrada para rede, seja suficiente para estimar a produtividade, assim como visto no trabalho de Soares et al. (2015) e Guimarães (2019).

3 METODOLOGIA

Este capítulo apresenta os materiais e métodos adotados para o desenvolvimento deste trabalho. A figura 7 resume as etapas divididas em três módulos:(1) coleta e preparação dos dados, (2) implementação e ajuste dos modelos, e (3) interpretação do modelo.

Figura 7: Etapas do desenvolvimento do trabalho.



Fonte: Autoria própria.

O primeiro módulo é apresentado na seção 3.1. Este apresenta as variáveis selecionadas, os locais onde foram retirados os dados e os procedimentos para o tratamento do conjunto de dados. Nos procedimentos, o algoritmo *iterative imputation* é aplicado para imputar dados faltantes a base de dados. Além disso, é aplicada à transformação de dados categóricos e normalização.

No segundo módulo, para fins dos objetivos deste trabalho, as variáveis da base de dados foram divididas em três bases com a mesma quantidade de dados, mas com variáveis distintas. A primeira base contém os dados agronômicos, a segunda os dados climáticos e de balanço hídrico, e, a terceira, tanto as variáveis agronômicas quanto as climáticas. Dessa forma, foram criados três modelos de MLP. A

implementação, ajuste dos modelos e ambiente de execução são descritos na seção 3.2.

Por fim, no terceiro módulo, apresentado na seção 3.3, é aplicado um método para a interpretação dos modelos. O SHAP, descrito na seção 3.3, tem como objetivo determinar as características mais importantes consideradas pelo modelo com menor erro.

3.1 COLETA E PREPARAÇÃO DOS DADOS

Os dados (características agronômicas) foram coletados de avaliações de híbridos de milho provenientes de experimentos conduzidos por e Duarte e Sawazaki (2018, 2019) no estado de São Paulo no vale do Paranapanema. Os cultivares, também, podem ser vistos no trabalho de Duarte e Sawazaki (2018, 2019). Esses dados contêm a produtividade de cada cultivar semeada em diferentes municípios da região do Paranapanema nos anos de 2018 e 2019 (Tabela 2).

Tabela 2: Municípios dos experimentos.

Município	Latitude	Longitude
Bernardino de Campos	-23,0148	-49,4731
Capão Bonito	-24,0027	-48,3503
Cândido Mota	-22,7467	-50,3880
Cruzália	-22,7439	-50,7898
Ibirarema	-22,8190	-50,0744
Manduri	-22,0061	-49,3206
Maracaí	-22,6155	-50,6683
Palmital	-22,7879	-50,2191
Pedrinhas Paulista	-22,8144	-50,7909

Fonte: Autoria própria.

As variáveis, selecionadas conforme o estado da arte (seção 2.5), podem ser divididas em: características após o desenvolvimento da planta (características agronômicas), variáveis climáticas e balanço hídrico sequencial. Além destas, também estão

incluídas estatísticas descritivas. A tabela 3 resume as informações gerais da base de dados com $kg \cdot ha^{-1}$ sendo a unidade de medida da produtividade.

Tabela 3: Informações gerais da base de dados.

Descrição	
Número de Locais	9
Anos	2018-2019
Média da Produtividade	6112,65
Desvio padrão	1799,26
Produtividade Mínima	570,01
Produtividade Máxima	9631,17
Número de Variáveis de Características Agronômicas	8
Número de Variáveis Climáticas	10
Número de Variáveis de Balanço Hídrico	5
Números de Observações	598

Fonte: Autoria própria.

As variáveis que compreendem as características agronômicas, as variáveis climáticas, o balanço hídrico e as estatísticas descritivas são listadas abaixo:

- Características agronômicas: altura da planta, altura da espiga, relação entre altura de espiga e altura de planta, número de plantas acamadas, número de plantas quebradas, número de dias para o florescimento, população de plantas, produtividade, umidade dos grãos na colheita e índice de espiga por planta;
- Variáveis climáticas: precipitação total diária, temperatura máxima, média e mínima, pressão atmosférica média diária, temperatura do ponto do orvalho média diária, umidade relativa do ar média e mínima, velocidade média do vento, rajada máxima do vento diário;
- Balanço hídrico sequencial: armazenamento (ARM), evapotranspiração real (ETR), deficit, excedente, evapotranspiração de referência (ETo);
- Estatísticas descritivas: média, desvio padrão e amplitude em função da produtividade por ano, local e cultivar.

Os dados climáticos e de balanço hídrico sequencial foram retirados das estações automáticas do Instituto Nacional de Meteorologia (INMET) o qual disponibiliza uma variedade de dados meteorológicos diários através de sua plataforma (INMET, 2020). Como o INMET não possui estações meteorológicas em alguns municípios, foi necessário considerar os dados das estações mais próximas. A tabela 4 mostra as estações consideradas mais próximas dos municípios.

Tabela 4: Estações meteorológicas consideradas.

Município	Estação	Distância (Km)
Bernardino de Campos	Ourinhos A716	40
Capão Bonito	Itapeva A714	54
Cândido Mota	Ourinhos A716	58,8
Cruzália	Ourinhos A716	94,1
Ibirarema	Ourinhos A716	27,2
Manduri	Avaré A725	43
Maracaí	Ourinhos A716	89,9
Palmital	Ourinhos A716	41,2
Pedrinhas Paulista	Ourinhos A716	94,5

Fonte: Autoria própria.

Para todas as variáveis climáticas e de balanço hídrico, foram considerados os ciclos de desenvolvimento do milho. Assim, foram considerados variáveis climáticas VE, V4, V8, F (florescimento), R1_2 (estádios de embonecamento e bolha d'água), R3_4 (estádios leitoso e Pastoso), R5, e R6. Considerou-se um ciclo de 120 dias para que as cultivares completassem todos os estádios de desenvolvimento. A tabela 5 apresenta a quantidade de dias consideradas para cada estádio. Nas variáveis precipitação total diária, ARM e excedente foram consideradas a soma acumulativa de cada fase descrita na tabela 5. Nas demais variáveis foram consideradas a média de cada período.

A Figura 8 apresenta uma amostra do conjunto de dados, apresentando variáveis de característica agronômicas, climática, e balanço hídrico.

Tabela 5: Período considerado para as variáveis climáticas e de balanço hídrico.

	Estádios							
	VE	V4	V8	F	R1_2	R3_4	R5	R6
Dias	0-10	10-40	40-50	50-60	60-75	75-90	90-105	105-120

Fonte: Autoria própria.

Figura 8: Amostra do conjunto de dados

.

Local	Ano	Cultivar	...	Precipitação_ VE	Temperatura Máxima_VE	Umidade Relativa do Ar Média_VE	...	ETo_R6	Produtividade
Pedrinhas Paulista	2018	DKB255	...	33	28	80,9	...	2,326391035	3551,3
Bernadinho de Campos	2018	XB8018	...	13	33,4875	79,61	...	1,18631299	6144,83
Cruzália	2018	MG711	...	27	30	78,55	...	3,405397639	7977,49
Pedrinhas Paulista	2019	DKB255	...	33	30,18	77,93	...	2,410736606	7974,40
Bernadinho de Campos	2019	XB8018	...	33	29,65	82,97	...	2,302020688	4339,50
Cruzália	2019	MG711	...	33	33,4875	80,9	...	4,003326189	4998,44

Fonte: Autoria própria.

3.1.1 Pré-Processamento

A base de dados foi processada antes da implementação efetiva dos modelos. Três procedimentos foram necessários para a preparação dos dados, são eles: imputação de dados faltantes, codificação de variáveis categóricas e normalização.

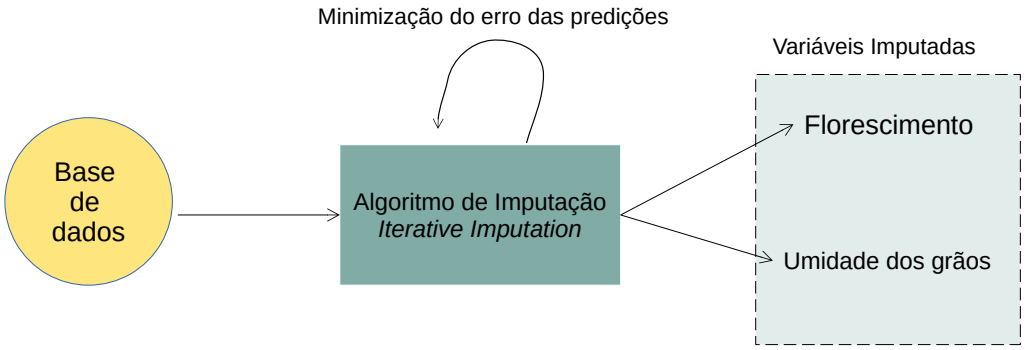
A base de dados continha alguns exemplos com dados faltantes, referente às variáveis de características agronômicas: florescimento e umidade dos grãos. Esses dados faltantes representavam 6% do total da base. O algoritmo *iterative imputation*, descrito no referencial teórico (seção 2.3.1), foi adotado para a imputação desses dados. A Figura 9 apresenta um esquema de imputação das variáveis florescimento e umidade dos grãos. A parametrização necessária para a execução do *iterative imputation* é apresentada na Tabela 6.

Tabela 6: Parâmetros *iterative imputation*

Parâmetro	Valor
Regressor	MLP
δ_{min}	0,1

Fonte: Autoria própria.

Figura 9: Esquema de imputação das variáveis faltantes.



Fonte: Autoria própria.

O regressor definido para as predições foi uma Rede Neural MLP (Tabela 6). A escolha desse regressor é defendida pelos resultados obtidos no trabalho de Richman, Trafalis e Adrianto (2009) e pela capacidade das RNAs capturarem as relações não-lineares entre os dados como constatou o mesmo.

Assim como foi especificado no algoritmo 1, o conjunto de dados é particionado em S e S'. O S possui todas as variáveis com dados completos. E o S' possui somente variáveis com valores faltantes. A base utilizada nessa etapa foi apresentada na seção anterior (seção 3.1).

Dessa forma, foram construídas dois modelos de MLP para estimar as variáveis do conjunto S' (Figura 9) com os mesmos hiperparâmetros (Tabela 7). Para o treinamento dos modelos, o conjunto S foi particionado em um conjunto de treinamento e um conjunto de teste. A base de treinamento possui 80% de exemplos de S e a de teste possui 20%.

O parâmetro δ_{min} defini quantas iterações serão executadas pelo *iterative imputation*. O δ_{min} também defini o erro mínimo aceitável em relação as iterações anteriores. Por meio de experimentos prévios, o δ_{min} foi de 0,1, isto é o erro alcançado das predições feitas pelo regressor em relação a iteração anterior foram de 0,1.

Após a imputação dos dados, aplicaram-se as transformações nas variáveis ca-

Tabela 7: Hiperparâmetros para a imputação de dados.

Hiperparâmetro	Valor
Camadas ocultas	3
Neurônios por camada	64
Função de ativação camada oculta	ReLu
Número de épocas	300
Otimizador	ADAM
Taxa de aprendizagem	0,001
Função de custo	MSE
Método de Inicialização de pesos	Glorot_normal

Fonte: Autoria própria.

tegóricas, cultivar e ano. A codificação adotada foi transformá-las em função da produtividade anual dado a variável de interesse. As equações 3.1, 3.2, 3.3 e 3.4 apresentam a transformação das variáveis categóricas.

$$local_r = \frac{1}{l}\sum_{i=1}^{l} p_i - p_{ano}. \tag{3.1}$$

onde, l é o número de observações realizadas no $r-ésimo$ local, p_i é a $i-ésima$ produtividade do $local_r$ e p_{ano} é a média das observações realizados em um determinado ano.

$$ano_s = \frac{1}{a}\sum_{j=1}^{a} p_j - p_{ano}. \tag{3.2}$$

onde, a é o número de observações realizados no ano_s, p_j é a $j-ésima$ produtividade realizada no ano_s.

$$c_k = p_k - p_{local} \tag{3.3}$$

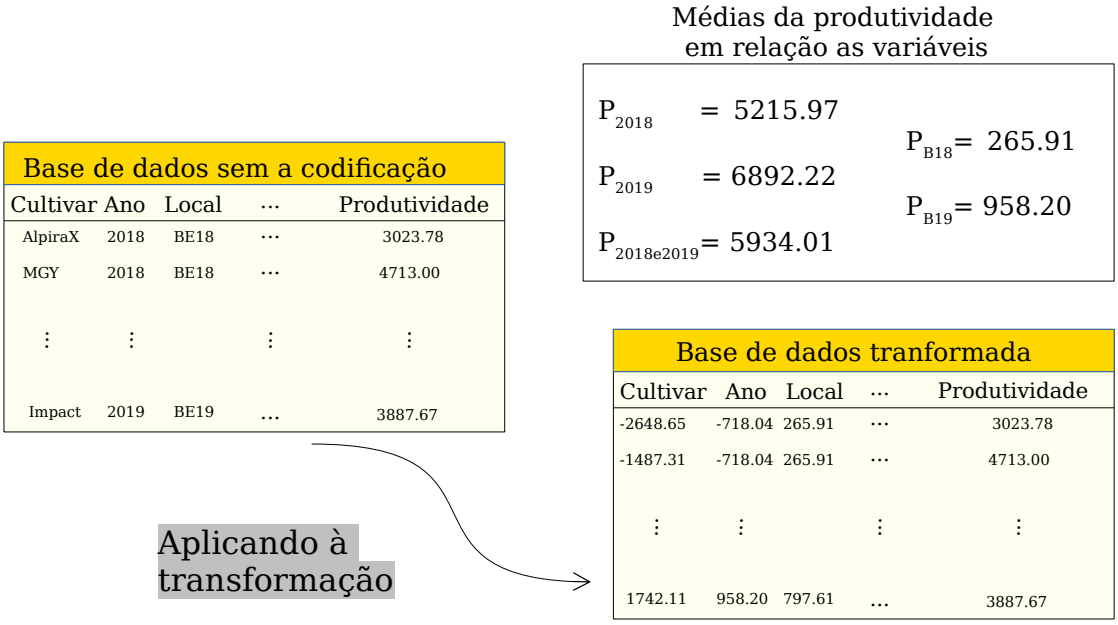
$$cultivar_e = \frac{1}{e}\sum_{k=1}^{e} c_k. \tag{3.4}$$

onde, p_k é a $k-ésima$ observação da produtividade da cultivar c_k, p_{local} é a média da produtividade de um determinado local, e é o número de observações de uma

cultivar e c_k é a $k - ésima$ diferença entre p_k e p_{local}.

Em todas as equações (3.1, 3.2, 3.3, 3.4) as variáveis categóricas estão em função da produtividade média observada. Essa codificação permite atribuir uma avaliação para essas variáveis. A Figura 10 ilustra a transformação adotada.

Figura 10: Transformação das variáveis categóricas. (Os valores presentes na figura não correspondem aos valores reais da base de dados).



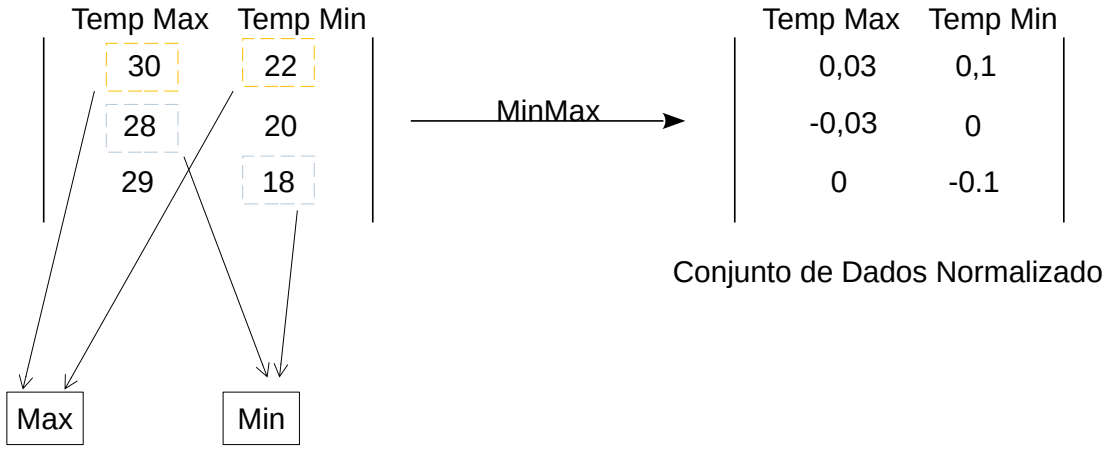
Fonte: Autoria própria.

No procedimento de normalização, o conjunto de dados de entrada da rede foi normalizado pelo MinMax (seção 2.3) considerando o intervalo [-1, 1], pois algumas variáveis assumem valores negativos. Ao considerar esse intervalo, a equação do MinMax é modificada para a equação 3.5,

$$(x \cdot \frac{max + min}{2}) \cdot (\frac{max + min}{2})^{-1} \tag{3.5}$$

que permite a base de dados ter valores negativos. A Figura 11 ilustra o procedimento apresentando duas variáveis do conjunto de dados, Temp Max (temperatura máxima) e Temp Min (temperatura mínima), o conjunto de entrada à esquerda sem nenhuma transformação e, à direita, o conjunto transformado.

Figura 11: Normalização do conjunto de dado de entrada.



Fonte: Autoria própria.

3.2 IMPLEMENTAÇÃO E AJUSTE DOS MODELOS

O conjunto de dados foi separado em três bases distintas. A primeira considerando as características agronômicas; a segunda com as variáveis climáticas e de balanço hídrico; e a terceira com todas as variáveis citadas anteriormente. Assim sendo, foram implementados três modelos de rede distintas com a mesma finalidade, a tarefa de predição da produtividade como saída dos modelos. Além desses, mais três modelos foram construídos considerando os conjuntos de dados com as imputações de dados faltantes. A Tabela 8 sumariza todos os modelos construídos mostrando a base de dados e o modelo.

Tabela 8: Modelos construídos.

Base de dados	Modelo
Características agronômica	MLP_1
Clima e balanço hídrico	MLP_2
Base completa	MLP_3
Características agronômica imputada	MLP_4
Clima e Balanço hídrico imputado	MLP_5
Base completa imputada	MLP_6

Fonte: Autoria própria.

A hiperparametrização necessária nesses modelos foi definida por meio do *Grid Search* e *k-fold cross-validation* a fim de encontrar um conjunto de hiperparâmetros com o menor erro na função de custo. Para isso, foi considerado previamente um conjunto de hiperparâmetros a serem testados pelo *Grid Search* e avaliados pelo *k-fold cross-validation*. A Tabela 9 apresenta esse conjunto de hiperparâmetros. Importante ressaltar que algumas configurações foram testadas e consideradas por meio do estado da arte resultando nos hiperparâmetros da Tabela 9. Considerou-se $k = 5$ para o *k-fold cross-validation* que é um dos valores utilizados na literatura (PEDREGOSA et al., 2011).

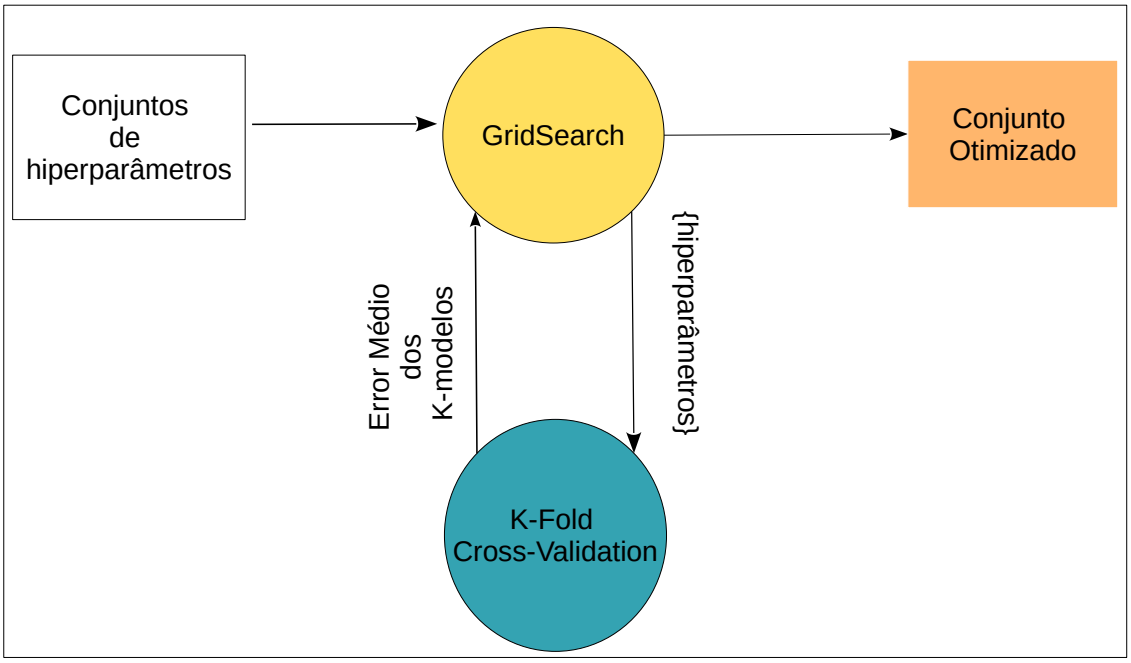
Tabela 9: Entrada de Hiperparâmetros para o *GridSearch*

Hiperparâmetro	**Conjunto**
Camadas ocultas	{1, 2, 3}
Neurônios	{32, 64, 128}
Função de ativação	{ReLu}
Método de inicialização de pesos	{*glorot_ normal*}
Épocas	{300, 600, 1000}
Batch size	{32, 64, 128}
Otimizador	{ADAM}
Taxa de apredizado	{0,005, 0,003, 0,001}
Função de custo	{MSE}

Fonte: Autoria própria.

A Figura 12 apresenta como *GridSearch* e o *k-fold cross-validation* podem ser utilizados em conjunto para otimizar um conjunto de hiperparâmetros. Primeiro são definidos os conjuntos de hiperparâmetros iniciais. Após, é calculado o produto cartesiano desses conjuntos e, para cada elemento do produto cartesiano, o erro é estimado pelo *k-fold cross-validation*. Após todos os hiperparâmetros serem avaliados, é retornado um conjunto de hiperparâmetros com o menor custo.

Figura 12: GridSeach e *k-fold cross-validation*.



Fonte: Autoria própria.

3.2.1 Ambiente de Execução e Tecnologias

Os experimentos foram executados no serviço disponibilizado pelo *Google Research*[1], chamado *Google Colaboratory*[2] e frequentemente referido como *Colab*. Especificamente, *Colab* é um *host* de serviços baseados no *Jupiter notebook* onde é possível executar códigos interativamente na linguagem python além de possuir um conjunto de ferramentas para o desenvolvimento de modelos de deep learning (BISONG, 2019; GOOGLE, 2018). A GPU (*Graphics Processing Unit*) utilizada foi a disponibilizada pelo Colab para acelerar o treinamento da rede. As especificações gerais do ambiente computacional é dado a seguir:

- Sistema operacional: Linux Ubuntu versão 18.04.5 LTS.
- Linguagem: Python versão 3.6.9.
- GPU: Tesla P100-PCIE, 3584 núcleos CUDA, 16 GB de memória, clock de 1328 MHz.

[1]https://research.google/
[2]https://colab.research.google.com

Os modelos foram construídos com o *TensorFlow*[3] e *Keras*[4]. O *Keras* é uma API (*Application Programming Interface - Interface de Programação de Aplicações*) de alto nível que facilita a construção de qualquer modelo de RNA (CHOLLET, 2015). *TensorFlow* é especializado em modelos baseados em aprendizagem profunda e possui diversos modelos pré-construídos. *Keras* é apenas o *front-end*, a implementação real é feita pela plataforma de aprendizagem de máquina *TensorFlow* criado pela *Google* (ABADI et al., 2016).

O *k-fold cross-validation* utilizado na validação do modelo corresponde ao da biblioteca de proposito geral de aprendizado de máquina *scikit-learn*[5] que possui um conjunto de ferramentas para o desenvolvimento de modelos de aprendizagem de máquina (PEDREGOSA et al., 2011). Além disso, o *scikit-learn* foi utilizado para a execução do otimizador de parâmetros *Grid Search* e para o pré-processamento dos dados.

3.3 INTERPRETAÇÃO DO MODELO

As RNAs são costumeiramente referidas como caixas pretas pelo fato de não ser uma tarefa trivial a interpretação de como o modelo resultou na saída. Isso ocorre pela divisão de responsabilidade entre os neurônios ao realizarem uma tarefa. Neste trabalho, aplicou-se o método SHAP (SHapley Additive exPlanations) (Figura 13) para que o modelo com o melhor desempenho possa ser interpretado.

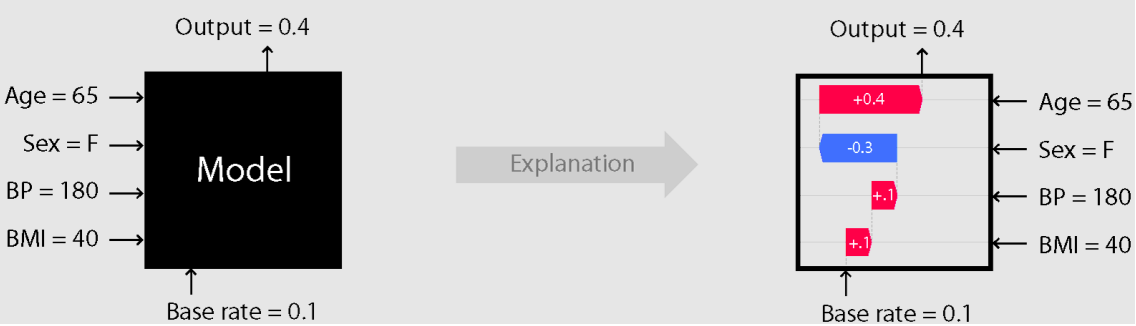
O SHAP é basedo principalmente em duas técnicas: *Shapley Values* e *Local Surrogate* (LIME) (LUNDBERG; LEE, 2017). *Shapley Values* é baseado na teoria dos jogos e tenta explicar como um exemplo X resultou em y. Para isso, ele considera que cada valor das características do exemplo (atributo) é um jogador que contribui na predição y. *Shapley Values* é a média marginal de todas as possíveis contribuições de uma característica. Em conjunto com o *Shapley Values*, o LIME tenta entender a causa de um modelo de caixa preta gerar determinadas predições. Para isso, ele

[3]https://www.tensorflow.org/

[4]https://keras.io/

[5]https://scikit-learn.org/

Figura 13: Interpretação dos modelos por SHAP.



Fonte: Adaptado de (LUNDBERG, 2017) .

constrói um modelo que seja interpretável (modelos de caixa branca) através de um novo conjunto de dados com ruídos (RIBEIRO; SINGH; GUESTRIN, 2016). O trabalho de Lundberg e Lee (2017) aprofunda os conceitos aqui abordados. A biblioteca que implementa o SHAP e usada neste trabalho está disponível no repositório do github[6].

[6]https://github.com/slundberg/shap

4 ANÁLISE DE RESULTADOS

Neste capítulo são apresentados e discutidos os resultados dos modelos treinados segundo a metodologia adotada. Os resultados são organizados em modelos treinados com a base de dados sem a imputação dos dados (seção 4.1) e com a imputação de dados (seção 4.2). Por fim, são analisados os resultados da aplicação do método SHAP no modelo com o melhor desempenho apresentado obtendo os pesos das variáveis nas predições da produtividade. Nas duas primeiras seções, são apresentados os gráficos de treinamento e validação, e os resultados do *k-fold cross-validation*.

Antes de apresentar os resultados, vale ressaltar que a unidade de medida da variável produtividade é $kg \cdot ha^{-1}$. Outro ponto importante, somente o modelo de cada base de dados (conforme a Tabela 8) com melhor desempenho é analisado nas seções posteriores.

4.1 MODELOS SEM IMPUTAÇÃO DE DADOS

A Tabela 10 apresenta os hiperparâmetros de três modelos - MLP_1, MLP_2 e MLP_3 - com menor RMSE dos 142 modelos treinados pelo *Grid Search* e *k-fold cross-validation*. Uma desvantagem conhecida do uso de *Grid Search* é o grande custo computacional para a sua execução, o que inviabiliza expor o método a uma grande carga de hiperparâmetros. Assim, há a possibilidade dos hiperparâmetros não serem o ótimo global e sim ótimos locais devido ao pequeno conjunto de hiperparâmetros. Todavia é justificado pelo custo computacional do *Grid Search*.

Os modelos com três camadas ocultas não obtiveram hiperparâmetros ótimos definidos pelo *Grid Search* e *k-fold cross-validation* nas bases de dados sem a imputação de dados, apenas os modelos com uma ou duas camadas ocultas, como pode ser observado na tabela 10. Uma possível resposta pode ser: o modelo tende a se sobre-ajustar aos dados de treinamento quanto mais se empilha camadas nas redes MLP obtendo, assim, erros superiores no conjunto de validação (HE et al., 2015). Consequentemente, não são candidatos a serem escolhidos pelo *Gridsearch*.

Tabela 10: Hiperparâmetros ótimos *Gridsearch* e *k-fold cross-validation* nas bases de dados sem a imputação de dados.

	Modelos		
Hiperparâmetro	**MLP_1**	**MLP_2**	**MLP_3**
Camadas ocultas	2	1	1
Neurônios	64	64	64
Função de ativação	ReLu	ReLu	ReLu
Otimizador	ADAM	ADAM	ADAM
Função de custo	MSE	MSE	MSE
Método de inicialização de pesos	*Glorot_normal*	*Glorot_normal*	*Glorot_normal*
Épocas	1000	1000	1000
Batch size	128	32	64
Taxa de aprendizagem	0,001	0,001	0,001

Fonte: Autoria própria.

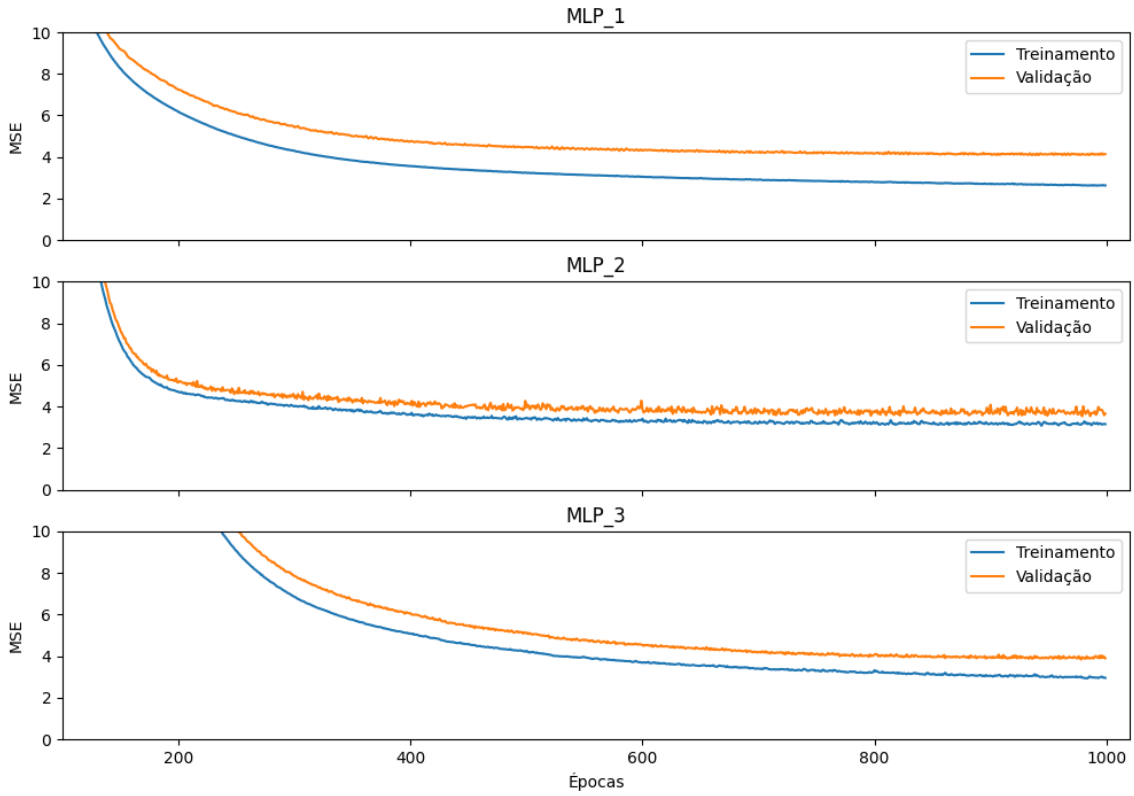
Observa-se, também, que em todos os modelos a quantidade de neurônios foi idêntica, 64 neurônios.

Outra observação é o tamanho do *batch size*. O modelo MLP_1 precisou de um lote de exemplos maior para que o treinamento fosse mais consistente. Isso pode ter ocorrido porque as variáveis de características agronômicas não explicam de forma concisa o que difere no *batch size* dos modelos MLP_2 e MLP_3. Em todos os modelos, a quantidade de épocas e a taxa de aprendizagem foi a mesma. Outros hiperparâmetros (função de ativação, otimizador, função de custo e método de inicialização de pesos) são considerados padrão na construção de modelos MLP (no sentido de serem frequentemente encontrados na literatura de redes neurais).

A Figura 14 apresenta as curvas de aprendizagem dos modelos MLP_1, MLP_2, MLP_3 com o erro (MSE) em função das épocas. Por meio das curvas de aprendizagem, é possível analisar o comportamento do treinamento e como ocorre o sobre-ajuste e sub-ajuste.

Em todos os modelos houve um treinamento conciso sem muita variação entre o treinamento e a validação, e sem sobre-ajuste aos dados de treinamento. No

Figura 14: Função de custo treinamento e validação do *k-fold cross-validation* sem imputação de dados.



Fonte: Autoria própria.

modelo MLP_2, que considera o clima e o balanço hídrico como entrada, obteve valores mais próximos entre o conjunto de treinamento e de validação, como pode ser observado na distância entre as curvas de treinamento e validação na Figura 14. No entanto, houve variação durante as épocas o que é evidenciado pelos picos no gráfico. A causa possível pode ser a quantidade de dados disponível e o *batch size* (assim como o modelo MLP_1 com *batch size* de 128). Ao considerar todas variáveis (MLP_1), observa-se que o comportamento das curvas foi semelhante ao modelo MLP_2, contudo mais consistente. O Modelo MLP_1 é o que teve maior distância entre a curva de treinamento e a de validação devido às variáveis não explicarem, como comentado anteriormente, suficientemente a produtividade.

A Tabela 11 apresenta o desempenho dos modelos MLP_1, MLP_2 e MLP_3 considerando o desempenho do treinamento e da validação segundo as métricas de avaliação, RMSE e correlação, além do erro da função de custo. A Tabela traz uma

comparação entre os modelos.

Tabela 11: Desempenho dos modelos sem imputação de dados na predição da produtividade de grãos em $kg \cdot ha^{-1}$ para diferentes locais na segunda safra, no Vale do Paranapanema, SP, em dois anos agrícolas.

Função de Custo/Métrica	MLP_1	MLP_2	MLP_3
MSE - Treinamento	263,587	315,133	295,013
RMSE - Treinamento	162,309	177,458	171,664
Correlação - Treinamento	99,564	99,449	99,537
MSE - Validação	413,801	365,302	389,571
RMSE - Validação	201,268	190,851	197,283
Correlação -Validação	99,375	99,094	99,207

Fonte: Autoria própria.

Dos modelos comparados, a MLP_1 obteve o maior RMSE (201,268 $kg \cdot ha^{-1}$) e o modelo MLP_2 o menor RMSE (177,458 $kg \cdot ha^{-1}$) no conjunto de validação. Considera-se que ambos os modelos obtiveram um erro aceitável. Considerando a correlação, todos os modelos obtiveram valores semelhantes. Esses resultados são satisfatórios devido à quantidade de dados disponíveis e adequados para a realidade obtida na avaliação de cultivares de milho em programas de melhoramento.

Com base nas estimativas de erros, os modelos construídos obtiveram um desempenho aceitável considerando o estado da arte. Os modelos se adequaram bem ao problema, tiveram um treinamento conciso, mesmo tendo uma limitação na quantidade dos dados. O modelo MLP_3, que considerou todas as variáveis, é o mais homogêneo em seu desempenho sendo o mais adequado para a predição da produtividade. Assim, considera-se que os modelos conseguiram generalizar para dados não vistos.

4.2 MODELOS COM IMPUTAÇÃO DE DADOS

A Tabela 12 apresenta os resultados dos hiperparâmetros dos modelos MLP_4, MLP_5 e MLP_6 treinados pelo *GridSearch* e *k-fold cross-validation* considerando

a imputação da base de dados. Nela, pode-se observar que, ao imputar os dados faltantes na base de dados, outro conjunto de hiperparâmetros foi necessário para atingir o ótimo local dos hiperparâmetros submetidos ao *GridSearch*. Isso indica que os exemplos que foram removidos continham padrões importantes para o problema.

Tabela 12: Hiperparâmetros ótimos *GridSearch* e *k-fold cross-validation* nas bases de dados com a imputação de dados.

	Modelos		
Hiperparâmetros	**MLP_4**	**MLP_5**	**MLP_6**
Camadas ocultas	2	1	1
Neurônios	128	128	64
Função de ativação	ReLu	ReLu	ReLu
Otimizador	ADAM	ADAM	ADAM
Função de custo	MSE	MSE	MSE
Método de inicialização de pesos	*Glorot_normal*	*Glorot_normal*	*Glorot_normal*
Épocas	1000	600	1000
Batch size	128	32	64
Taxa de aprendizagem	0,005	0,001	0,001

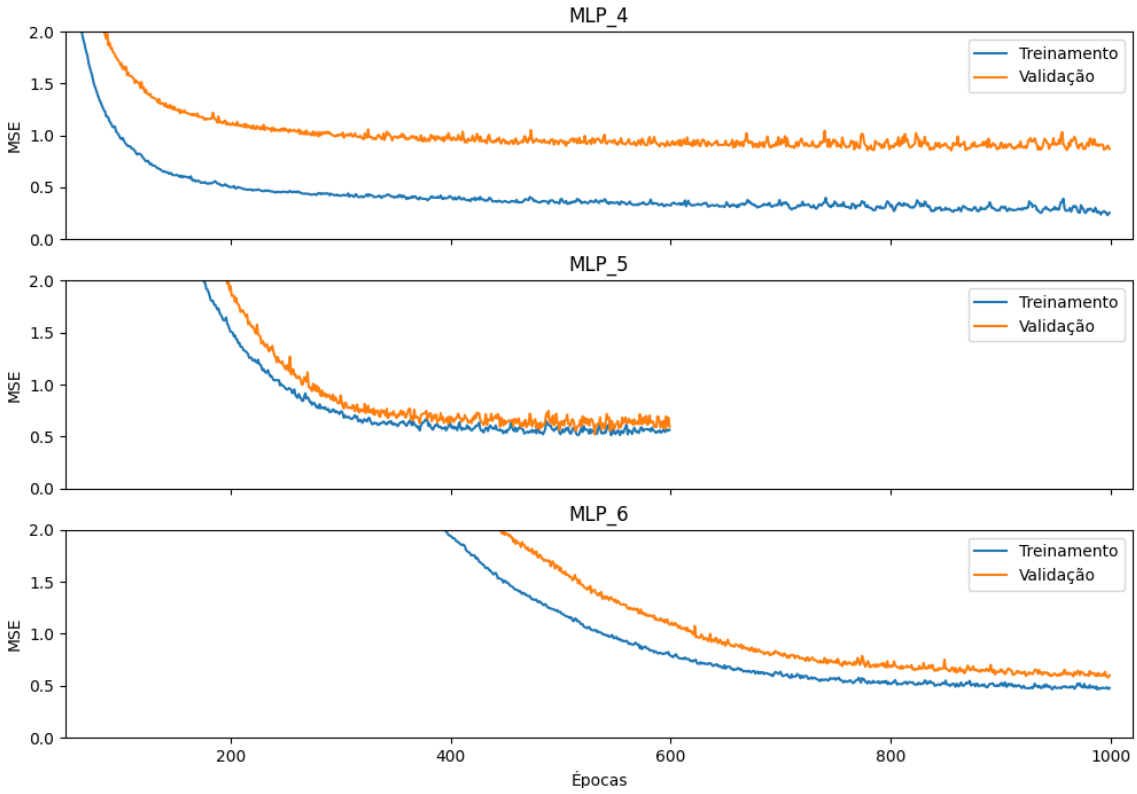
Fonte: Autoria própria.

As alterações foram na quantidade de neurônios, épocas, *batch size* e taxa de aprendizagem. Considerando que cada neurônio corresponde a um hiperplano, uma quantidade superior de neurônios (128 para os modelos MLP_4 e MLP_6) foi necessária para separar adequadamente o espaço de instâncias. Somente o modelo MLP_6 que permaneceu com a mesma quantidade de neurônios dos modelos com imputação de dados. Na Taxa de aprendizagem, o modelo MLP_4 obteve 0,005. Isso representa que o modelo estava "preso" em pobres locais com uma taxa de aprendizagem inferior. Uma taxa de aprendizagem maior permitiu "escapar" dos mesmos. Para os outros modelos continuou em 0,001. Apenas o modelo MLP_5 obteve uma quantidade menor de épocas para convergir, 600 épocas.

Na Figura 15 é possível notar os impactos da imputação de dados no treinamento dos modelos. Diferente dos modelos sem imputação de dados, os erros foram inferiores. Contudo, o treinamento foi menos consistente, como pode ser observado

nos picos presentes nos gráficos do modelo MLP_4, MLP_5, e MLP_6. Uma das possíveis razões é que a imputação dos dados aumentou a variabilidade dos mesmos.

Figura 15: Função de custo treinamento e validação do *k-fold cross-validation* com imputação de dados



Fonte: Autoria própria .

O modelo MLP_4 comparado ao modelo MLP_1 se sobre-ajustou mais aos dados de treinamento (Figuras 14 e 15). Como o MLP_1, o modelo MLP_4 teve o mesmo problema das variáveis agronômicas não serem o suficiente para explicar a produtividade apesar de uma correção indireta. Isso sugere que mais variáveis são necessárias e que as variáveis climáticas são fundamentais para o problema em questão. Isto pode ser confirmado pelos modelos MLP_6 e MLP_3 que consideram todas as variáveis da base de dados. Nestes dois, as curvas de treinamento e de validação são mais próximas entre si.

Considerando o modelo MLP_5, ao comparar as curvas dele às curvas dos modelos MLP_4 e MLP_6, percebe-se que as curvas do MLP_5 são mais próximas entre si. Além disso, a quantidade de épocas necessárias para convergir foi menor.

O que pode-se notar desse modelo, com ou sem a imputação de dados, é que as variáveis explicam bem o comportamento da variável alvo. Tal afirmação é validada segundo ao que foi explanado na seção 2.4 e por meio do trabalho de Khaki e Wang (2019) que obtiveram resultados semelhantes ao construir uma Rede Convolucional para predição da produtividade no *corn belt*, Estados Unidos da América.

A Tabela 13 apresenta o desempenho dos modelos MLP_4, MLP_5 e MLP_6. O modelo que obteve o menor RMSE foi MLP_5 com 70,651 $kg \cdot ha_{-1}$, demonstrando seu excelente ajuste com um erro de apenas um saco de milho por hectare. Também é possível perceber que o erro de treinamento e de validação são semelhantes, assim, como é visto no gráfico da Figura 15. Isso sugere que esse modelo conseguiu generalizar. Em contrapartida, o modelo MLP_4 obteve o maior RMSE de validação (89,651 $kg \cdot ha^{-1}$), o que permite observar um ajuste do modelo aos exemplos de treinamento.

Entre esses três modelos, MLP_6 obteve RMSE de validação de 74,352 $kg \cdot ha_{-1}$. Considerando o RMSE de treinamento, o modelo não teve grandes diferenças ao compará-lo ao modelo MLP_5. Entretanto, pode se notar que ao considerar as características agronômicas, o modelo MLP_6 tende a aumentar o erro. Em todos os modelos há uma alta correlação, não tendo diferenças significativas ao compará-los aos modelos sem imputação de dados.

Tabela 13: Desempenho dos modelos com imputação de dados a predição da produtividade de grãos em $kg \cdot ha^{-1}$ para diferentes locais na segunda safra, no Vale do Paranapanema, SP, em dois anos agrícolas.

Função de Custo/Métrica	MLP_4	MLP_5	MLP_6
MSE - Treinamento	25,456	56,368	47,693
RMSE - Treinamento	49,316	74,682	68,674
Correlação - Treinamento	99,97	99,930	99,939
MSE - Validação	86,89	59,514	59,935
RMSE - Validação	89,731	70,651	74,352
Correlação - Validação	99,889	99,875	99,916

Fonte: Autoria própria.

Segundos os resultados obtidos e a análise feita não somente dos erros dos modelos, a imputação de dados obteve um desempenho superior ao compará-la com a base sem os dados faltantes. Isso sugere que o problema em questão necessita de uma quantidade razoável de dados para que o desempenho da rede seja semelhante ao que foi visto no estado da arte. Neste trabalho foi considerado dois anos de experimentos, entretanto, mais anos são necessários para que a rede capture as variações que ocorrem com passar dos anos nos locais dos experimentos.

Embora o desempenho não seja o mínimo possível, os modelos conseguiram detectar os efeitos não lineares entre o ambiente e genótipo da planta. Tais interações são importantes a serem detectadas pelos modelos. Pode-se concluir que as redes implementadas conseguiram extrair um padrão nos dados mesmo que os locais dos experimentos e cultivares fossem distintos, o que mostra a capacidade das RNAs na generalização. Por fim, considera-se dentre todas as redes implementadas, a MLP_5 e a MLP_6 como as mais consistentes e com um erro aceitável para as predições.

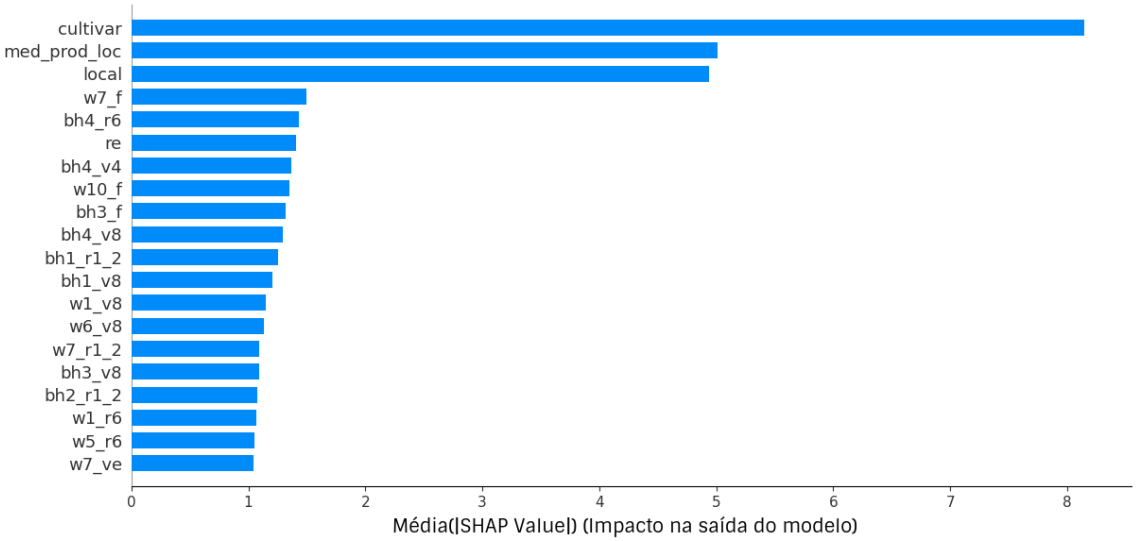
4.3 INTERPRETAÇÃO DO MODELO

Nas seções anteriores, as análises dos resultados foram feitas por meio do desempenho do modelo. Neste tipo de análise, muitas vezes não é possível entender o significado biológico realizado pelo próprio modelo ao receber uma entrada. Nesta seção é estimado, por meio do método SHAP, as variáveis com maior impacto.

O impacto médio das variáveis com maior peso é apresentado na Figura 16. No eixo horizontal, o *SHAP value* representa a magnitude das variáveis na saída do modelo MLP_6. No eixo vertical, as variáveis com maior peso. Os nomes das variáveis estão abreviadas.

A produtividade de grãos é explicada pelo potencial genético da cultivar, do ambiente físico e climático além da interação da cultivar com o ambiente (BORÉM; MIRANDA; FRITSCHE-NETO, 2021). Como pode ser observado na Figura 16, a variável com maior impacto foi a cultivar fato justificado pela importância da cultivar na produtividade. Ela define o potencial genético da planta de milho bem

Figura 16: Impacto médio da entrada na saída do modelo.



Fonte: Autoria própria.

como as suas limitações em determinados ambientes. Especificamente, os cultivares definem: adaptação ao clima de cultivo, adaptação ao solo de cultivo, ciclo da cultura, retenção de elementos nutritivos e adaptação a insetos, pragas e doenças. Esses fatores são importantes no resultado final da colheita (MIRANDA; BRAUN et al., 2021).

As variáveis média da produtividade local (med_prod_local) e local também tiveram um grande impacto na saída. A med_prod_local define o valor da produtividade média que pode ser obtido em determinado local mostrando o efeito ambiental. A variável local define o ambiente. Isso significa que o modelo conseguiu capturar os efeitos do ambiente sobre a produtividade bem como a interação entre o genótipo e o ambiente. Além disso, o ambiente é um importante fator que pode variar espacialmente no decorrer do tempo (BORÉM; MIRANDA; FRITSCHE-NETO, 2021).

As variáveis climáticas e de balanço hídrico com um impacto razoável, conforme pode ser observado na Figura 16, são descritas a seguir:

- **Estádio emergencial e vegetativo**: umidade relativa do ar média diária (w7_ve), precipitação total diário (w1_v8), temperatura mínima diária

(w6_v8), excedente (bh4_v4), ARM (bh1_v8), deficit (bh3_v8);

- **Estádio de florescimento**: umidade relativa do ar média diária (w7_f) e vento rajada máxima diária, deficit (bh3_f); e
- **Estádio reprodutivo**: umidade relativa o ar média diária (w7_r1_2), precipitação total diária (w1_r6), temperatura mínima diária (w5_r6), ARM (bh1_r1_2), ETR (bh2_r1_2), excedente (bh4_r6).

As variáveis w7_f e w10_f tiveram impacto no estádio de florescimento. A variável w7_f fora do recomendado nesse estádio pode afetar a produção de grãos reduzindo a fecundação (BERGAMACHI; MATZENAUER, 2014). A w10_f, apesar de não ter muito efeito sobre o rendimento final dos grãos, pode danificar as folhas da planta de milho (BERGAMACHI; MATZENAUER, 2014).

O estádio vegetativo é afetado pelas variáveis w7_ve, w6_v8 e w1_v8. Assim como no estádio de florescimento, w7_ve desempenha um papel importante na germinação das sementes como apontou o modelo. A w6_v8 é fator importante devido às necessidades hídricas para o desenvolvimento saudável do milho. A w1_v8 nesse estádio pode retardar o desenvolvimento caso não esteja em temperatura adequada pela planta.

No estádio reprodutivo, onde os grãos se formam, o modelo identificou que as variáveis w7_r1, w1_r6 e w5_r6 foram importantes para a saída da rede. Isso é justificado pelas intensas necessidades hídricas, térmicas e de umidade relativa do ar no início do enchimento dos grãos da planta de milho.

As variáveis de balanço hídrico, segundo a Figura 16, estão presentes em todos os estádios de desenvolvimento. Diferente das condições climáticas, o balanço hídrico de fato aponta a disponibilidade de água que está armazenada no solo e será utilizada pela planta (PEREIRA FILHO, 2002). Isso justifica o impacto dessas variáveis no modelo.

A variável re (relação entre altura da planta e espiga), também citada na Figura 16, é uma característica igualmente importante e está relacionada diretamente com o

acamamento das plantas. No que lhe concerne, o acamamento interfere no transporte de água e outros nutrientes da planta do milho (REPKE et al., 2012).

Vale ressaltar que as análises realizadas são importantes, pois, por meio delas, pode-se entender quais variáveis merecem mais atenção durante o desenvolvimento da planta nas fases de desenvolvimento em determinado local de semeadura. Assim, ações preventivas podem ser realizadas caso esses fatores não estejam alinhados.

5 CONCLUSÃO

A predição ou estimativa da produtividade de qualquer cultura agronômica, incluindo a cultura do milho, é um problema que traz benefícios para produção de alimentos. A tarefa de predição da produtividade requer que modelos de aprendizagem de máquina capturem as relações não-lineares entre o ambiente e o genótipo da planta. As MLPs são modelos que capturam as relações não-lineares entre os dados. Então, este trabalho objetivou a construir modelos de RNAs do tipo MLP para predizer a produtividade da cultura do milho no Vale do Paranapanema, São Paulo, considerando fatores que afetam a mesma.

As RNAs são modelos computacionais inspirados no funcionamento cerebral de seres humanos. A sua flexibilidade permite aplicá-las em diversos problemas ganhando notoriedade devido seu alto desempenho e sua generalização. Não obstante, vários trabalhos surgiram aplicando as RNAs para a predição da produtividade.

Este trabalho tirou proveito do estado da arte, tanto das RNAs quanto da cultura do milho para definir a metodologia adotada. Considerou-se variáveis agronômicas, condições climáticas e balanço hídrico como características para a predição da produtividade. Além de desenvolver os modelos, o método SHAP foi aplicado para interpretar os resultados dos modelos a fim de entender quais variáveis foram consideradas mais importantes nas predições.

Os resultados dos modelos foram satisfatórios levando em consideração a quantidade de dados expostos às RNAs. Os modelos conseguiram identificar variáveis importantes durante o estádio de desenvolvimento da planta e observou-se que as variáveis climáticas e de balanço hídricos merecem devida atenção. Além disso, o ambiente e o genótipo definem o potencial produtivo alcançável.

Conclui-se, por meio disso, que as RNAs são modelos adequados para a tarefa de predição de produtividade do milho. Sendo possível obter desempenhos aceitáveis para localidade de mesma região ainda que as condições climáticas sejam adversas. Por meio da imputação de dados foi possível compreender as necessidades de uma quantidade maior de dados, sendo recomendável usar conjuntos de dados com mais

anos agrícolas.

Para trabalhos futuros sugere-se uma base de dados com uma quantidade de dados maior que compreenda vários anos, a implementação de outros modelos de RNAs e uma comparação com métodos comuns na predição produtividade.

Referências

ABADI, M. et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. **CoRR**, abs/1603.04467, p. 1–21, 2016.

ALVES, G. R. **Estimativa da produtividade da soja com redes neurais artificiais**. 2016. Dissertação (Mestrado em Engenharia Agrícola) – Faculdade Engenharia de Sistemas Agroindustriais, Universidade Estadual de Goiás, Anápolis, 2016.

ANGUITA, D. et al. The 'K'in K-fold cross validation. In: ESANN, 2012, Bruges. **20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)**. Ottignies-Louvain-la-Neuve: i6doc.com publ, 2012. p. 441–446. Disponível em: <http://www.i6doc.com/en/livre/?GCOI=28001100967420>. Acesso em: 12 ago. 2020.

BANNERJEE, G. et al. Artificial intelligence in agriculture: A Literature Survey. **International Journal of Scientific Research in Computer Science Applications and Management Studies Artificial**, v. 7, p. 1–6, 2018.

BATISTA, G. E. d. A. P. A. **Pré-processamento de Dados em Aprendizado de Máquina Supervisionado**. 2003. Tese (Doutorado em Ciências) – Universidade de São Paulo, São Carlos, 2003.

BERGAMACHI, H.; MATZENAUER, R. **O Milho e o Clima**. Porto Alegre: Emater/RS-Ascar, 2014.

BERGSTRA, J.; BENGIO, Y. Random Search for Hyper-Parameter Optimization. **Journal of Machine Learning Research**, v. 13, p. 281–305, 2012.

BEZERRA, E. Introdução à Aprendizagem Profunda. In: 31° SIMPÓSIO BRASILEIRO DE BANCO DE DADOS, 2016, Salvador. **Tópicos em Gerenciamento de Dados e Informações**. Bahia: Sociedade Brasileira de Computação, 2016. p. 57–86. Disponível em: <http://www.sbbd2016.org/>. Acesso em: 5 fev. 2021.

BISHOP, C. M. **Patter Recognition and Machine Learning**. New York: Springer, 2006.

BISONG, E. **Building Machine Learning and Deep Learning Models on Google Cloud Platform**. Berkeley: Apress, 2019.

BORÉM, A.; MIRANDA, G.; FRITSCHE-NETO, R. **Melhoramento de Plantas**. São Paulo: Oficina do Texto, 2021.

CHOLLET, F. **Keras**. 2015. Disponível em: <https://keras.io/>. Acesso em: 11 nov. 2020.

CONAB. **Acompanhamento da Safra Brasileira de Grãos: Segundo Levantamento Safra 2020/2021**. 2020. Disponível em: <http://www.conab.gov.br>. Acesso em: 20 dez. 2020.

CONTINI, E. et al. Milho - Caracterização e Desafios Tecnológicos. **Desafios do Agronegócio Brasileiro**, v. 5, p. 1–45, 2019.

CRUZ, J. C. et al. **Manejo da Cultura do Milho**. Sete Lagoas: Embrapa, 2006. p. 1–12.

DAHIKAR, S. S.; RODE, S. V. Agricultural Crop Yield Prediction Using Artificial Neural Network Approach. **INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING**, v. 2, p. 683–686, 2014.

DUARTE, A. P.; SAWAZAKI, E. **AVALIAÇÃO REGIONAL DE CULTIVARES DE MILHO SAFRINHA IAC / APTA / CATI / EMPRESAS Resultados 2018**. Assis - SP, 2018.

DUARTE, A. P.; SAWAZAKI, E. **AVALIAÇÃO REGIONAL DE CULTIVARES DE MILHO SAFRINHA Resultados 2019**. Assis - SP, 2019.

ESPINOSA-MENESES, O. et al. Spiking Neural Net to Solve the Shortest Path NP Problem. In: ICMEAE, 2019. **2019 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)**. Mexico: IEEE, 2019. p. 68–73. Disponível em: <https://ieeexplore.ieee.org/document/9140162/>. Acesso em: 15 jun. 2020.

FAO. **OCDE-FAO Perspectivas Agrícolas 2015-2024**. 2015. Disponível em: <http://www.fao.org/publications/card/en/c/5413df90-c43d-42c3-89bd-3b956dfaa396/>. Acesso em: 19 nov. 2020.

FEURER, M.; HUTTER, F. Hyperparameter Optimization. In: HUTTER, F.; KOTTHOFF, L.; VANSCHOREN, J. (Org.). **Automated Machine Learning**. New York: Springer, 2019. cap. 1, p. 3–33.

FILIPPI, P. et al. An approach to forecast grain crop yield using multi-layered, multi-farm data sets and machine learning. **Precision Agriculture**, v. 20, p. 1015–1029, 2019.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: THE 13TH INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND STATISTICS, 2010, Laguna Resort. **JMLR: W&CP**. Italy, 2010. p. 249–256. Disponível em: <https://jmlr.org/proceedings/template.html>. Acesso em: 1 nov. 2020.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. Massachusetts: MIT Press, 2016.

GOOGLE. **Colaboratory: Frequently Asked Questions**. 2018. Disponível em: <https://research.google.com/colaboratory/faq.html>. Acesso em: 14 nov. 2020.

GUIMARÃES, E. d. S. **Aprendizado de Máquina aplicado à predição da produtividade da cultura da soja utilizando dados de clima e solo**. 2019. Dissertação (Mestrado em Matemática Estatística e Computação Aplicadas à Indústria) – Instituto de Ciências, Matemáticas e de Computação, Universidade de São Paulo, 2019.

HAYKIN, S. **Neural networks and Learning Machines**. Canada: Pearson, 2009.

HAYKIN, S. **Redes Neurais: princípios e prática**. Porto Alegre: Bookman, 2003.

HAYOU, S.; DOUCET, A.; ROUSSEAU, J. On the Selection of Initialization and Activation Function for Deep Neural Networks. In: ICLR, 2018, Ernest N. Morial Convention Center. **International Conference on Learning Representations**. New Orleans: ICLR, 2018. p. 1–19. Disponível em: <https://iclr.cc/Conferences/2019>. Acesso em: 22 jan. 2021.

HE, K. et al. Deep Residual Learning for Image Recognition. **Indian Journal of Chemistry - Section B Organic and Medicinal Chemistry**, v. 1, p. 770–778, 2015.

HOCKING, R. R. **Methods and Applications of Linear Models**. New Jersey: John Wiley & Sons, Inc., 2003.

INMET. **Banco de Dados Meterelógicos**. 2020. Disponível em: <https://bdmep.inmet.gov.br/>. Acesso em: 1 ago. 2020.

JI, B. et al. Artificial neural networks for rice yield prediction in mountainous regions. **The Journal of Agricultural Science**, v. 145, p. 249–261, 2007.

KAUL, M.; HILL, R. L.; WALTHALL, C. Artificial neural networks for corn and soybean yield prediction. **Agricultural Systems**, v. 85, p. 1–18, 2005.

KHAKI, S.; WANG, L. Crop Yield Prediction Using Deep Neural Networks. **Frontiers in Plant Science**, v. 10, p. 621, 2019.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. ImageNet classification with deep convolutional neural networks. In: ESANN, 2012, Lake Tahoe. **Proceedings of the 25th International Conference on Neural Information Processing Systems**. Nevada: Curran Associates Inc., 2012. p. 84–90. Disponível em: <https://dl.acm.org/doi/proceedings/10.5555/2999134>. Acesso em: 11 fev. 2021.

LATHUILIERE, S. et al. A Comprehensive Analysis of Deep Regression. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 42, p. 2065–2081, 2020.

LEAL, A. J. F. et al. Redes neurais artificiais na predição da produtividade de milho e definição de sítios de manejo diferenciado por meio de atributos do solo. **Bragantia**, v. 74, p. 436–444, 2015.

LUNDBERG, S. **SHAP**. 2017. Disponível em: <https://github.com/slundberg/shap>. Acesso em: 1 ago. 2021.

LUNDBERG, S.; LEE, S.-I. A unified approach to interpreting model predictions. **CoRR**, abs/1705.07874, p. 1–10, 2017.

MAGALHÃES, P. C.; DURÃES, F. O. M. **Fisiologia da Produção de Milho**. 2006.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The Bulletin of Mathematical Biophysics**, v. 5, p. 115–133, 1943.

MELLO, R. F. de; PONTI, M. A. **Machine Learning - A Practical Approach on the Statistical Learning Theory**. New York: Springer International Publishing, 2018.

MENDEL, J.; MCLAREN, R. Adaptive, Learning and Pattern Recognition Systems. In: J.M. MENDEL, K. F. (Ed.). **Mathematics in Science and Engineering**. Elsevier, 1970. cap. 8, p. 287–318.

MIAO ZHENJIANG; YUAN BAOZONG. Speech recognition by extended loop neural network. In: INTERNATIONAL SYMPOSIUM ON SPEECH, IMAGE PROCESSING AND NEURAL NETWORKS, 1994, Hong Kong. **Proceedings of ICSIPNN '94**. New York: IEEE, 1994. p. 335–338. Disponível em: <https://ieeexplore.ieee.org/document/344898>. Acesso em: 12 mai. 2021.

MICHELON, G. K. **Aplicação de técnicas de inteligência artificial na agricultura de precisão para estimar a produtividade da soja**. 2016. TCC (Bacharel em Ciência da computação) – Universidade Tecnológica Federal do Paraná, Medianeira, 2016.

MINSKY, M.; PAPERT, S. **Perceptrons**. Cambridge: MIT Press, 1969.

MIRANDA, G. V.; BRAUN, E. M. W. et al. Desempenho de híbridos de milho em diferentes épocas de semeadura na segunda safra em baixa altitude no extremo Oeste do Estado do Paraná. **Brazilian Journal of Development**, v. 7, p. 34794–34810, 2021.

MIRANDA, R. A. de; LÍCIO, A. M. A. **Diagnóstico dos Problemas e Pontencialidade da Cadeia Produtiva do Milho**. 2014.

MONTEIRO, J. E. B. A. **Agrometeorologia dos Cultivos: O fator meteorológico na produção agrícola**. Brasília: Embrapa, 2009.

MORETO, V. B. **Modelagem para Auxiliar na Otimização do Sistema "Climate-Smart-Agriculture" para Cultivo de Cana-de-Açúcar**. 2019. Tese (Doutorado em Agronomia) – Universidade Estadual Paulista, Anápolis, 2019.

MURPHY, K. P. **Machine Learning: A Probabilistic Perspective**. Cambridge: The MIT Press, 2012.

MUTASA, S.; SUN, S.; HA, R. Understanding artificial intelligence based radiology studies: What is overfitting? **Clinical Imaging**, v. 65, p. 96–99, 2020.

NATARAJAN, B. K. Learning Concepts on Countable Domains. In: KAUFMANN, M. (Org.). **Machine Learning**. Amsterdam: Elsevier, 1991. cap. 2, p. 7–40.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PEREIRA FILHO, I. A. **O Cultivo do Milho Verde**. Brasília: Embrapa, 2002.

PICOLI, M. C. A. **Estimativa Da Produtividade Agrícola Da Cana-De-Açúcar Utilizando Agregados De Redes Neurais Artificiais: Estudo De Caso Usina Catanduva**. 2007. Instituto Nacional de Pesquisas Espaciais, 2007.

PONTI, M. A.; COSTA, G. B. P. da. Como funciona o Deep Learning. **CoRR**, 2018.

REPKE, R. A. et al. Altura de planta, altura de inserção de espiga e número de plantas acamadas de cinco híbridos de milho. **XXIX Congresso Nacional De Milho E Sorgo**, n. 1, 2012.

RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. **CoRR**, abs/1602.04938, 2016. arXiv: 1602.04938. Disponível em: <http://arxiv.org/abs/1602.04938>.

RICHMAN, M. B.; TRAFALIS, T. B.; ADRIANTO, I. Missing Data Imputation Through Machine Learning Algorithms. In: ARTIFICIAL Intelligence Methods in the Environmental Sciences. Springer Netherlands, 2009.

RITCHIE, W. S.; HANWAY, J. J.; BERSON, G. o. **How a corn plant develops**. 1986.

RIZZI, R.; RUDORFF, B. F. T. Imagens do sensor MODIS associadas a um modelo agronÃ´mico para estimar a produtividade de soja. **Pesquisa AgropecuÃ¡ria Brasileira**, 2007.

ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. **Psychological Review**, 1958.

RUBIN, D. B. Multiple Imputation After 18+ Years. **Journal of the American Statistical Association**, v. 91, p. 473, 1996. ISSN 01621459. DOI: 10.2307/2291635. Disponível em: <https://www.jstor.org/stable/2291635?origin=crossref>.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning Internal Representations by Error Propagation. In: PARALLEL Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations. MIT Press, 1986.

SAATH, K. C. d. O.; FACHINELLO, A. L. Crescimento da demanda mundial de alimentos e restrições do fator terra no Brasil. **Revista de Economia e Sociologia Rural**, 2018.

SANTOS, A. M. D. et al. Using Artificial Neural Networks and Logistic Regression in the Prediction of Hepatitis A. **Rev Bras Epidemiol**, v. 8, n. 2, p. 117–126, 2005.

SANTOS, V. B. dos. **Estimação e previsão de produtividade de soja por redes neurais no MATOPIBA**. 2020. Diss. (Mestrado) – UNIVERSIDADE ESTADUAL PAULISTA.

SHALEV-SHWARTZ, S.; BEN-DAVID, S. **Understanding machine learning: From theory to algorithms**. 2013.

SHARMA, S.; SHARMA, S.; ATHAIYA, A. Activation Functions in Neural Networks. **International Journal of Engineering Applied Sciences and Technology**, 2020.

SOARES, F. C. et al. Predição da produtividade da cultura do milho utilizando rede neural artificial. **Ciencia Rural**, 2015.

SRIVASTAVA, P. et al. Understanding Soil Aggregate Dynamics and Its Relation With Land Use and Climate Change. In: CLIMATE Change and Agricultural Ecosystems. Elsevier, 2019.

TRAUTMANN, A. P. B. **Modelagem Matemática e Computacional da Produtividade do Trigo e Otimização do Uso do Nitrogênio nas Condições Fenológicas e Ambientais**. 2020. Tese (Doutorado) – UNIVERSIDADE REGIONAL DO NOROESTE DO ESTADO DO RIO GRANDE DO SUL.

VALE, T. M. **Modelagem Agrometeorológica para Estimar a Produtividade da Cultura da Soja no Estado do Tocantins**. 2019. Diss. (Mestrado) – UNIVERSIDADE FEDERAL DO TOCANTINS.

VENDRUSCULO, L. G.; OLIVEIRA, S. R. d. M. Modelo de previsão da produtividade agrícola para auxílio à Agricultura de Precisão a partir da identificação de padrões frequentes em base de dados espaço-temporal. In:

WERBOS, P. J. **Beyond regression : new tools for prediction and analysis in the behavioral sciences abstract**. 1974. Tese (Doutorado) – Harvard University.

ZEVIANI, W. m.; RIBEIRO JUNIOR, P. J.; BONAT, W. H. 58ªReunião Anual da Região Brasileira da Sociedade Internacional de Biometria (RBras) e o 15ºSimpósio de Estatística Aplicada à Experimentação Agronômica (SEAGRO). In: MODELOS de regressão não linear. 2013.

ZOU, J.; HAN, Y.; SO, S.-S. Overview of Artificial Neural Networks. In: 2008. p. 14–22.