

---

## Utilização de Modelo Preditivo Para Definir a Saúde de Bovinos

### Use of a Predictive Model to Define Cattle Health

---

**Fabício Xavier Cerci**ORCID: <https://orcid.org/0000-0003-4434-5396>

FATEC SENAI Technology College, Brazil

fabriciocerci@gmail.com

**Abraão Nazário**ORCID: <https://orcid.org/0000-0001-7073-1667>

FATEC SENAI Technology College, Brazil

abraao.nazario@ufms.br

**Guilherme Falcão da Silva Campos**ORCID: <https://orcid.org/0000-0003-4434-5396>

University of Vale do Rio dos Sinos, Brazil

guilherme.campos@ufmt.br

**Layla Leão Lima Teixeira**ORCID: <https://orcid.org/0009-0000-6833-4515>

CIMATEC SENAI Technology College, Brazil

laylaleao@gmail.com

---

### RESUMO

Segundo a Embrapa, o Brasil é um dos maiores produtores de carne bovina, com o maior rebanho em 2015. Este artigo propõe um modelo preditivo utilizando tecnologias disponíveis no campo para identificar a saúde dos animais, permitindo um tratamento mais eficaz e melhorando a qualidade do produto. Utilizamos dados do Kaggle e programação em Python com bibliotecas relevantes para criar o algoritmo preditivo. Após testes, o algoritmo DecisionTreeClassifier foi escolhido e treinado com o conjunto de dados. O modelo alcançou aproximadamente 100% de precisão na identificação da saúde dos animais utilizando informações não utilizadas no treinamento. Porém, o tamanho do dataset pode ter impactado negativamente nos testes e treinamentos. Apesar disso, conclui-se que é possível utilizar um modelo preditivo baseado em dados colhidos com tecnologias existentes para a tomada ágil de medidas e aumentar a possibilidade de recuperação dos animais.

**Palavras-chave:** machine learning; modelo preditivo; bovinos.

---

## ABSTRACT

According to Embrapa, Brazil is a leading beef producer globally, with the largest cattle herd in 2015. This article aims to use existing technologies in the field to create a predictive model that determines the health of animals using collected parameters, improving treatment efficiency and product quality. Kaggle data on cattle activities was used to develop the predictive algorithm using Python and relevant libraries. Tests identified the DecisionTreeClassifier algorithm as the best predictor for the dataset. It achieved an approximate 100% accuracy in identifying the health status of animals. However, the dataset's size may have affected test outcomes negatively. Nonetheless, data collected from existing technologies can be used to create a predictive model, allowing faster actions to increase the chances of animal recovery.

Keywords: machine learning; predictive model; cattle.

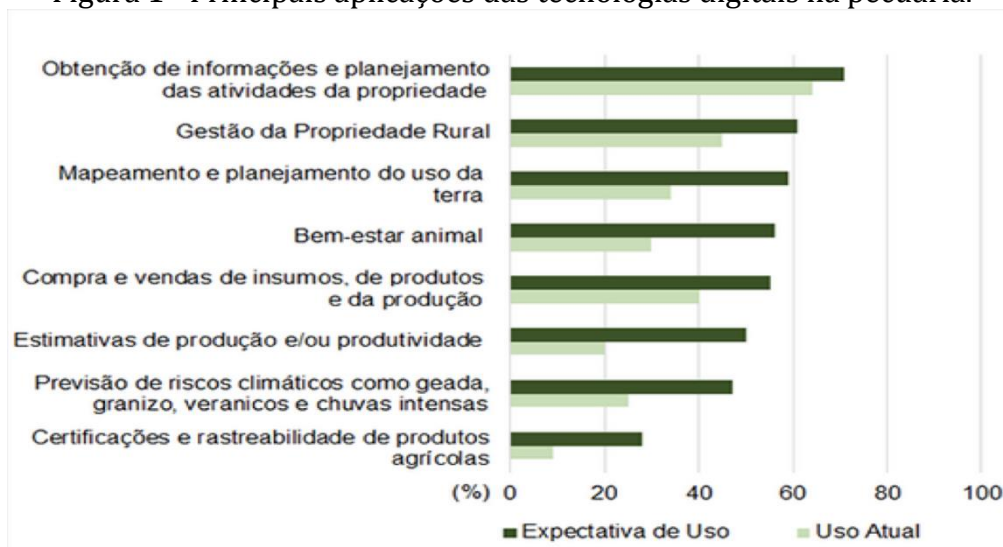
---

## INTRODUÇÃO

De acordo com a Embrapa, o Brasil é um dos mais importantes produtores de carne bovina do mundo, sendo o segundo maior consumidor e o segundo maior exportador, tendo em 2015 o maior rebanho com 209 milhões de cabeças (EMBRAPA, 2022).

Através de uma pesquisa realizada por (BOLFE et al., 2021), verificou-se que o Brasil além de ser um dos maiores produtores, também no campo da agropecuária apenas 18% dos pecuaristas entrevistados informaram que não utilizam nenhuma ferramenta digital, e conforme demonstrado na figura 1 abaixo, 64% utilizam aplicativos móveis para obtenção de dados e gestão, 45% utilizam plataforma para gestão, 34% utilizam sensores remotos como satélites e drones, 40% utilizam aplicativos móveis para compra de insumos ou comercialização de produtos, 30% utilizam alguma ferramenta para averiguar o bem-estar animal, 25% para previsão climática, 20% sensores remotos ou sensoriais e 9% utilizam para certificação e rastreabilidade da sua produção.

Figura 1 - Principais aplicações das tecnologias digitais na pecuária.



Fonte: BOLFE, et al. (2021).

No entanto, apesar do uso generalizado de tecnologia na pecuária, observamos que sua aplicação para o bem-estar animal está abaixo do esperado, o que é crucial para o ganho de peso e a longevidade dos animais. Com base nessa constatação, o objetivo deste trabalho é aproveitar o avanço da digitalização na agropecuária para desenvolver um modelo preditivo. Esse modelo utilizará inteligência artificial e machine learning para determinar a saúde dos bovinos, por meio da coleta de dados realizada por outras tecnologias, como chips subcutâneos, dispositivos em forma de bolachas, cabrestos e balanças inteligentes. Dessa forma, busca-se utilizar as vantagens proporcionadas pela tecnologia para melhorar a qualidade de vida dos animais, maximizando seu ganho de peso e prolongando sua vida.

### Tecnologias para a Coleta de Dados em Bovinos

Este tópico estará abordando as principais tecnologias de coletas de dados que são utilizadas para ajudar o produtor nessa coleta e na tomada de decisões.

### Balança de Pesagem Inteligente

O principal dado coletado pela balança é o peso do animal, onde através dele é possível saber se o animal ganhou ou perdeu peso. Uma balança inteligente comercializada que realiza esse trabalho é a BalPass, sendo possível obter os seguintes

benefícios (COIMMA, 2022): monitoramento do peso do rebanho diariamente; visualização de animais acima ou abaixo do rendimento; informação de momento ideal de venda; acompanhamento de resultados econômicos da atividade do produtor; monitoramento do ponto de equilíbrio financeiro de cada animal; contagem diária dos animais e comparação de rendimentos entre fazendas, piquetes, lotes e animais.

De acordo com a Embrapa (2022), a balança BalPass possibilita ao produtor ter: "... confiabilidade dos dados; identificação de ganho de peso do rebanho em tempo real; acesso remoto (Mobile ou Web) aos dados do rebanho; pesagem dos animais livres em campo; diminuição do estresse dos animais durante a pesagem; baixo risco de lesão de carcaça."

*Figura 2 - Balança inteligente BalPass.*



Fonte: COIMMA (2022).

### **Cabrestos Inteligentes**

O cabresto inteligente normalmente é um dispositivo no qual um chip no cabresto do animal realiza a coleta dos dados, o grande benefício desse método é que ele é um dispositivo não-invasivo ao animal, já que fica em um equipamento no qual o animal já está acostumado. Um exemplo comercial importante de um cabresto inteligente é o BEP (BOVINE ELECTRONIC PLATFORM), produto desenvolvido pela INDEXT, o qual permite coletar dados como: frequência cardíaca, frequência respiratória, temperatura corporal do animal, temperatura e umidade do ambiente, radiação ultravioleta, entre outros (INDEXT, 2022).

Figura 3 - Cabresto Inteligente.



Fonte: (RODRIGUES, 2022).

### Tag inteligente

As tags inteligentes ficam na orelha do animal, onde coletam dados diversos, como temperatura, localização, quantidade de passos, entre outros. Um exemplo de tag inteligente é a desenvolvida pela empresa Zoetis, chamado de SMARTBOW sendo possível através dos dados coletados detectar calor, movimentação individual de cada animal, localizar animais em tempo real, identificar problemas de saúde, entre outros (SMARTBOW, 2022).

Figura 4 - Figura 4 - SMARTBOW tag inteligente.



Fonte: PORTAL DO AGRONEGÓCIO (2022).

### Big Data

Com a grande evolução e inovação da tecnologia, ocorreu um crescimento exponencial de dados, pois a todo momento estamos gerando uma enorme quantidade de dados em todos os setores, como indústrias, negócios, ciência e na vida pessoal. Entretanto, esse crescimento vem sendo um desafio para as tecnologias de processamento de dados convencionais, como banco de dados e data warehouses, que estão se tornando inadequadas para lidar com o montante de dados gerados (CHEN et al., 2013).

Segundo Chen et al. (2013), é esse problema que define o desafio chamado Big Data. Por ser um problema comum e de grande importância, ele veio ganhando relevância, mas ainda não há uma definição sobre o que é exatamente o Big Data, acreditava-se que deveria possuir datasets de tamanhos acima da capacidade dos métodos tradicionais, sendo capaz de capturar, gerenciar e processar os dados em um tempo aceitável. Seguindo esse princípio, pesquisadores definiram aspectos importantes do Big Data chamado de 3V's que seriam: Velocidade, Variedade e Volume.

## **Inteligência Artificial**

De acordo com Lobo (2018), a inteligência artificial é uma área da ciência da computação que propõe através do desenvolvimento de softwares simular a capacidade humana de percepção de problemas, podendo assim, identificar seus pontos e com isso propor e ou tomar decisões sem a intervenção humana. Assim, a inteligência artificial já vem mudando a vida de várias pessoas, através de veículos independentes, cirurgias realizadas por robôs, automação de atendimento e serviços financeiros, entre outras áreas (LOBO, 2018).

## **Machine Learning**

Machine learning é um algoritmo computacional que utiliza dados de entrada para realizar uma tarefa em específico, sem que seja literalmente programada para produzir um resultado. Através da repetição o algoritmo se adapta e melhora cada vez mais para realizar a tarefa para a qual foi designado.

Inicialmente para que o algoritmo consiga ter um ponto de partida é realizado um processo de aprendizagem através de treino, onde são passados os dados de entrada e os resultados esperados de saída. Após o treino, o algoritmo realiza um processo de

otimização onde ele possa além de demonstrar resultados nos dados de entradas dispostos nos testes, também nas entradas de dados desconhecidos. Com o tempo esse algoritmo vai aprendendo com seus acertos e erros, tornando-se cada vez mais preciso (EL NAQA; MURPHY, 2015).

## **Algoritmos machine learning**

Este tópico estará abordando algoritmos de machine learning utilizados neste trabalho explicando suas características:

### **Algoritmo de Classificação e Regressão**

A regressão em machine learning foca na predição de valores numéricos baseando-se nos valores dos atributos. Entretanto, a classificação já utiliza os valores dos atributos para criar uma afiliação a uma certa classe específica (KIRCHNER; HEBERLE; LOWE, 2015).

### **Árvore de Decisão**

De acordo com Napoleão (2018), as árvores de decisão: "...foram apresentadas pela primeira vez em 1975, por J. Ross Quinlan, em seu livro Machine Learning. Anos depois, em 1983, Quinlan criou o primeiro algoritmo para gerar árvores decisórias, chamado Iterative Dichotomiser 3 (ID3). Dessa forma, o autor consagrou-se como o “pai das Árvores de Decisórias”."

A árvore de decisão é um algoritmo que cria caminhos possíveis como um fluxograma com ramificações e nós, assim organizando caminhos a serem seguidos para a tomada de decisões, podendo cada nó ser correspondente a uma ação, decisão, probabilidade ou condição (NAPOLEÃO, 2018).

A árvore de decisões é muito utilizada quando a classificação dos dados não é binária, sendo assim muito mais simples que outros algoritmos para lidar com esses cenários, além de que não é necessário se preocupar muito com o tratamento desses dados (SACRAMENTO, 2021).

### **Máquina de Vetores de Suporte**

O algoritmo de máquina de vetores de suporte ou support vectors machine é utilizado como a árvore de decisões tanto para classificação ou regressão, a grande diferença é que ele separa de maneira linear as classes/rótulos de um conjunto de dados. Essas fronteiras lineares que o algoritmo cria são chamados de hiperplanos, e sua análise tenta encontrar quais desses hiperplanos é o melhor para o conjunto de dados utilizado (OLIVEIRA JÚNIOR, 2010; REMÍGIO, 2020).

### **Multilayer Perceptron**

O algoritmo multilayer perceptron é o modelo mais utilizado nas aplicações de redes neurais utilizando o treino de algoritmo de retropropagação do erro (back-propagation). O algoritmo MLP é uma rede neural com múltiplos elementos denominados neurônios, esses elementos ficam entre os dados de entrada e saída, podendo assim ser ajustados de acordo com a complexidade necessária na saída (RAMCHOUN et al., 2016). A figura a seguir demonstra como o algoritmo MLP funciona.

### **K-Nearest Neighbors**

O K-Nearest Neighbors é um algoritmo de classificação comumente utilizado para realizar reconhecimento estatístico de padrões (NISSILA, 2023). De acordo com Ranjan, Kumar E Radhika (2019), o algoritmo funciona através da comparação da distância entre o parâmetro pesquisado com todos os dados de treinamento e assim atribui ele a uma classe representada pela maioria de atributos “vizinhos”.

Como descrito por Ranjan, Kumar E Radhika (2019), o algoritmo toma essa decisão através dos “vizinhos” mais próximos, e com isso os seus parâmetros são de extrema importância para determinar o valor calculado pelo algoritmo, alguns desses parâmetros são: `n_neighbors`: O número de “vizinhos” próximos para usar como padrão para as pesquisas; `weights`: Esse parâmetro determina como cada ponto é tratado e qual seu peso na pesquisa, como exemplo pontos mais próximos podem ter um peso e influência maior que pontos distantes; `algorithm`: Há vários algoritmos diferentes para fazer os cálculos de “vizinhos” mais próximos e `metric`: Esse parâmetro determina qual métrica é utilizada para especificar a distância no algoritmo.

### **Gaussian Naive Bayes**



O algoritmo de Gaussian Naive Bayes é um classificador probabilístico simples e fácil que utiliza como base a aplicação do teorema de Bayes. O algoritmo trata cada variável de maneira independente, com isso esse classificador pode ser treinado com o aprendizado supervisionado de maneira bem eficiente e utilizado em situações complexas do mundo real. Sua vantagem é que ele depende de pouca mensuração dos dados de treinamento, os quais são necessários para a classificação e caracterização dos dados (KAMEL; ABDULAH; AL-TUWAIJARI, 2019).

### **AdaBoost**

De acordo com Schapire (2013), Boosting é uma abordagem de machine learning que consiste na ideia de criar uma regra de alta precisão através da combinação de várias regras relativamente fracas e imprecisas. O algoritmo do AdaBoost de Freund e Schapire foi considerado o pioneiro a ser implementado de maneira prática, mesmo assim permanece sendo um dos mais estudados e utilizados, sendo aplicado para vários propósitos. Durante anos houveram muitas tentativas para explicar como o AdaBoost funciona, porque funciona e quando ele funciona ou falha (CHAVES, 2012; WU; NAGAHASHI, 2014).

### **Métricas de avaliação**

Este tópico descreve as métricas utilizadas para realizar a avaliação de performance dos algoritmos utilizados no trabalho.

### **Matriz de confusão**

Como descrito por Maria Navim e Pankaja (2016), a matriz de confusão é uma tabela gerada para a classificação de datasets binários, sendo utilizada para definir a performance do classificador. Ela é definida pelos seguintes parâmetros: True positives (TP) - Onde a predição é verdadeira e valor real também é verdadeiro; True Negatives (TN) - Onde a predição é falsa e valor real também é falso; False Positives (FP) - Onde a predição é verdadeira e valor real é falso e False Negatives (FN) - Onde a predição é falsa e valor real é verdadeiro.

Figura 5 - Matriz de Confusão.

	Predicted: No	Predicted: Yes
Actual: No	TN	FP
Actual: Yes	FN	TP

Fonte: Maria Navim e Pankaja (2016).

## Precision

De acordo com Sujatha e Mahalakshmi (2020), precision é uma métrica que leva em consideração a proporção de true positive (TP) em relação a todos positivos preditos pelo algoritmo (TP+FP). Como demonstrado no cálculo abaixo:

*Equação 1 - Cálculo Métrica Precision. Fonte: Sujatha e Mahalakshmi (2020).*

$$\text{Precision} = \frac{TP}{TP+FP}$$

## Recall

A métrica recall, como descrita por Akhtar et al. (2021), é definida pela proporção de true positive (TP) em relação ao número total de dados que realmente eram positivos. A fórmula da métrica pode ser vista na imagem abaixo:

*Equação 2 - Cálculo Métrica Recal. Fonte: Akhtar et al. (2021, p. 71).*

$$\text{Recall} = \frac{TP}{(TP + FN)}$$

## F1 score

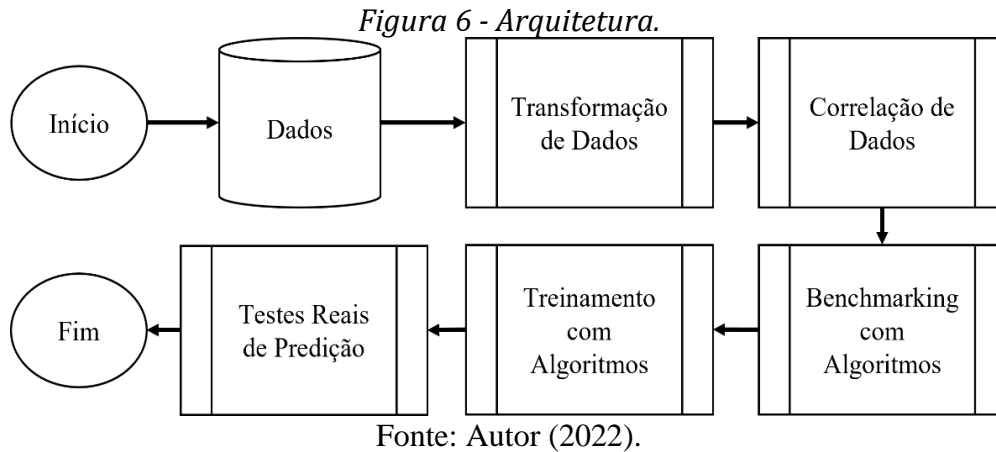
A métrica F1 score, de acordo com Sujatha e Mahalakshmi (2020), é a combinação das métricas precision e recall. Sendo calculada através do ajuste fino entre as métricas precision e recall, podendo ser demonstrada pelo cálculo abaixo:

*Equação 3 - Cálculo Métrica F 1 Score. Fonte: Sujatha e Mahalakshmi (2020).*

$$F1 = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

## METODOLOGIA

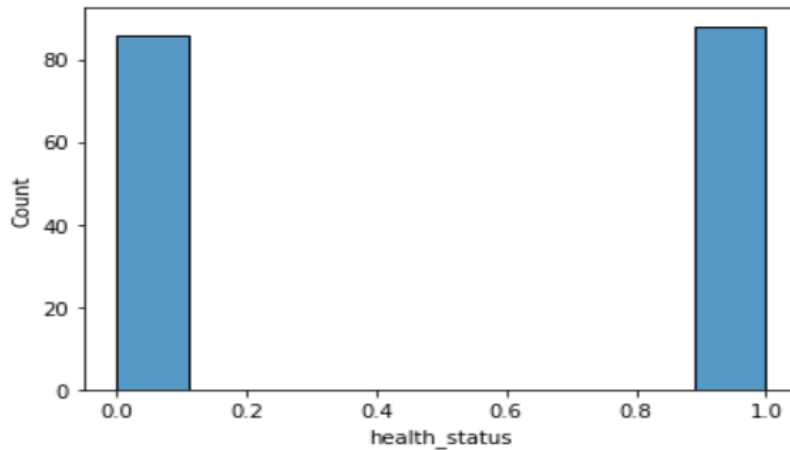
Para a realização deste projeto foi utilizada uma arquitetura focada inicialmente na obtenção de dados e treinamento do algoritmo escolhido, onde foram realizados testes com dados reais.



## Dataset

No desenvolvimento foi utilizado um dataset do site Kaggle encontrado no link (<https://www.kaggle.com/datasets/khushupadhyay/cow-health-prediction?resource=download>) chamado de "*cow-health-prediction*". O dataset possui 178 linhas onde cada linha define a saúde do bovino com as seguintes variáveis: temperatura corporal, raça, produção de leite, taxa de respiração, capacidade de movimentação, duração do sono, taxa de batimentos cardíacos, duração de alimentação, tempo deitado, tempo ruminando, tempo com o rúmen cheio, consistência fecal e status da saúde. O dataset está com os dados bem distribuídos tendo praticamente a mesma quantidade de linhas com bovinos doentes ou saudáveis, como demonstrado na imagem abaixo:

Figura 7 -Distribuição dos Dados.



Fonte: Autor (2022).

### Normalização/Transformação dos Dados

Para utilizar os dados para realizar a correlação foi realizada a transformação de alguns dados que eram textos para números como demonstrado na tabela 1.

Após essa transformação, todos os dados do datasets estavam representados por números, mas além dessa transformação foi realizada a normalização desses números, o que consiste em transformar todos os valores numéricos entre períodos entre 0 e 1.

**Tabela 1** - Transformação dos Dados. Fonte: Autor (2022)

Nome da coluna	Valor antes da transformação	Valor após transformação
health_status	health	1
health_status	unhealthy	0
breed_type	Cross Breed	1
breed_type	Normal Breed	0
faecal_consistency	ideal	0
faecal_consistency	extremely firm	1
faecal_consistency	Fresh blood in faeces	2
faecal_consistency	Black faece	3
faecal_consistency	very liquid faeces	4

A normalização colabora para que os algoritmos tenham resultados mais assertivos e performáticos. Para realizar a normalização o seguinte cálculo foi utilizado:

Figura 8 - Distribuição dos Dados.

$$\frac{(df - df.min())}{(df.max() - df.min())}$$

Fonte: Autor (2022).

Posteriormente a normalização podemos ver como ficou alguns dados como exemplo na imagem abaixo:

Figura 9 - Exemplo dos Dados Normalizados.

body_temperature	breed_type	milk_production	respiratory_rate
0.823529	0.0	0.813084	0.971429
0.882353	1.0	0.271028	0.285714
0.549020	0.0	0.957944	0.514286
0.490196	0.0	0.813084	0.371429
0.509804	0.0	0.957944	0.542857

Fonte: Autor (2022).

## Correlação dos Dados

Em seguida da transformação e normalização dos dados foi realizada a correlação dos dados para realizar a descoberta de quais variáveis possuem mais influência na saúde do bovino. Onde o resultado foi o demonstrado abaixo:

Figura 10 - Correlação dos Dados Normalizados.

health_status	1.000000
walking_capacity	0.954491
sleeping_duration	0.932883
body_temperature	0.875290
ruminating	0.836912
faecal_consistency	0.812956
respiratory_rate	0.797662
heart_rate	0.794643
eating_duration	0.702646
lying_down_duration	0.616001
milk_production	0.581765
rumen_fill	0.183694
breed_type	0.034619
body_condition_score	0.008280

Fonte: Autor (2022).

Através do resultado demonstrado na imagem foram escolhidos os 4 parâmetros que mais possuem correlação com a saúde do bovino, as quais foram: capacidade de movimentação, duração de sono, temperatura corporal e tempo ruminando.

### Benchmarking com Algoritmos

Para realização dos benchmarkings dos algoritmos duas coisas precisavam ser definidas antes de serem levantados os algoritmos a serem utilizados: O problema a ser resolvido é uma regressão ou classificação? Qual métrica utilizar para realizar a avaliação da performance dos algoritmos? A primeira pergunta é fácil de ser respondida, pois como queremos *classificar* o animal entre saudável ou doente, o problema deve ser resolvido através da classificação, pois precisamos que o algoritmo atribua ao animal em análise a uma das duas classes. A segunda pergunta depende da primeira, e como foi realizada a classificação para resolver o problema, foram levantadas três métricas de classificação: precision, recall e f1 score. Entre essas, foi escolhida a f1 score por ser uma mistura e balanceamento das métricas precision e recall.

### Treinamento e Escolha do Algoritmo

Para treinamento e escolha do algoritmo, foi realizada primeiramente a divisão em X (features) e y (labels), onde em X estava o dataframe com os 4 parâmetros escolhidos através da correlação e no y os valores binários em relação a saúde do bovino.

*Figura 11 - Correlação dos Dados Normalizados.*

```
X = df_reduzido[col[:4]]  
y = df_reduzido.health_status
```

Fonte: Autor (2022).

Em sequência foi utilizada a função *train\_test\_split* do scikitlearn com o parâmetro para separar 33% do dataset para realização do teste dos algoritmos.

*Figura 12 - Separação para Treinamento dos Algoritmos.*

```
[33] X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.33)
```

Fonte: Autor (2022).

Portanto foi realizado um loop utilizando a métrica f1 score para validar vários algoritmos de classificação os validando através de: tempo de execução, score de treinamento e score de teste, como demonstrado na imagem abaixo:

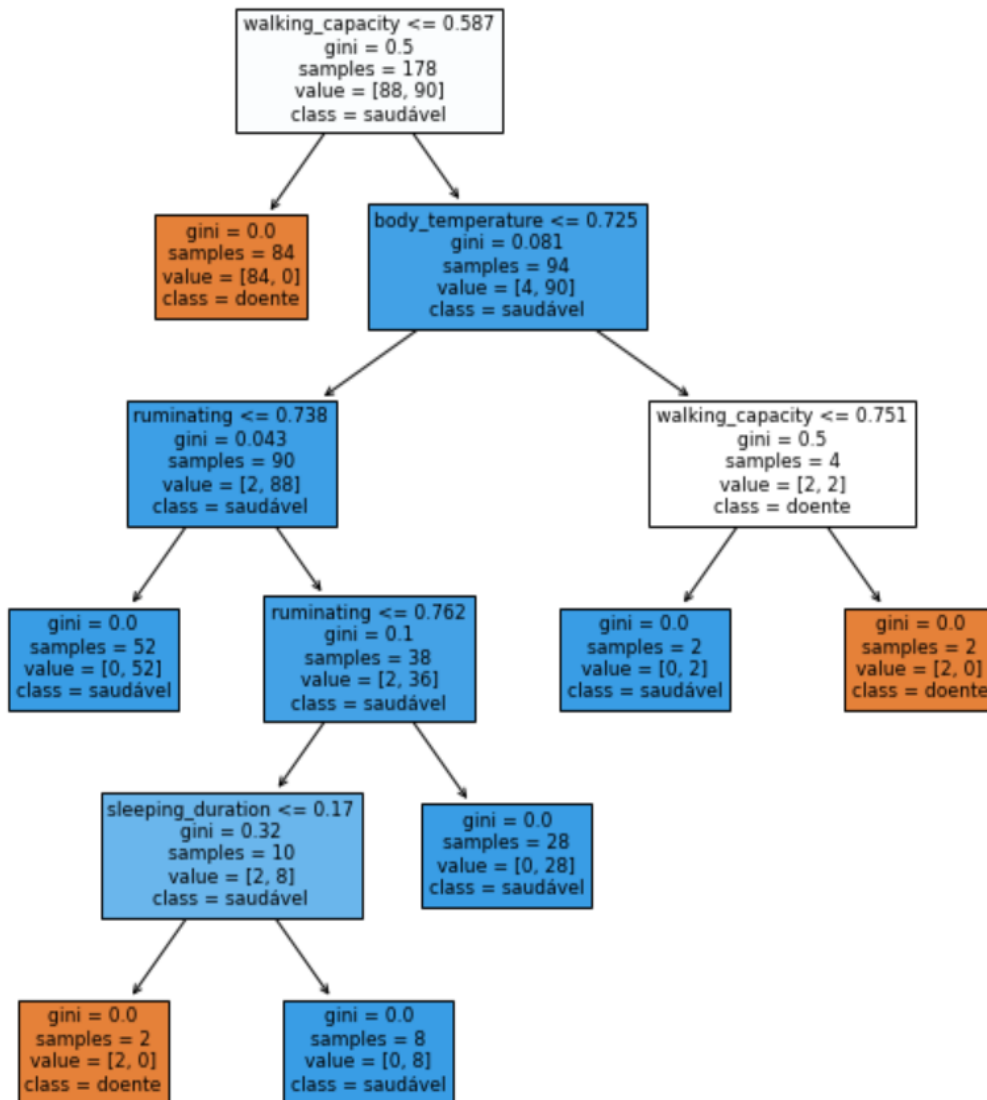
*Figura 13 - Score de Algoritmos de Classificação.*

0	KNeighborsClassifier	0.003326	0.975610	0.952381
1	SVC	0.003022	0.975610	0.952381
2	GaussianNB	0.001726	0.983607	0.967742
3	DecisionTreeClassifier	0.002107	1.000000	1.000000
4	AdaBoostClassifier	0.087262	1.000000	1.000000
5	MLPClassifier	0.196289	0.975610	0.952381

Fonte: Autor (2022).

Os dois algoritmos que melhor performaram foram o DecisionTreeClassifier e AdaBoostClassifier, o qual foi escolhido o DecisionTreeClassifier, tanto pelo seu melhor tempo de execução, como também a sua possibilidade visual de demonstração de decisões como demonstrada na imagem abaixo:

Figura 14 - Decision Tree Classifier.



Fonte: Autor (2022).

Por último foi realizado um GridSearchCV para encontrar quais seriam os melhores parâmetros para o DecisionTreeClassifier.



Figura 15 - GridSearchCV.

```
from sklearn.metrics import make_scorer
from sklearn.model_selection import GridSearchCV, ShuffleSplit

classifier = DecisionTreeClassifier()
params = {'criterion': ['gini', 'entropy', 'log_loss'], 'splitter': ['best', 'random']}

funcao_scorer = make_scorer(f1_score)

grid = GridSearchCV(classifier, params, scoring=funcao_scorer)
inicio = time()
melhor_modelo = grid.fit(X,y)

[42] melhor_modelo.best_params_

{'criterion': 'gini', 'splitter': 'best'}
```

Fonte: Autor (2022).

## RESULTADO

Para o teste do algoritmo foi realizado no início após a normalização dos dados a separação de 4 linhas para realizar os testes e validar se o algoritmo conseguiria acertar o estado de saúde do animal. Onde todos os dados foram preditos de maneira correta pelo algoritmo, como demonstrado na imagem abaixo:

Figura 16 - Teste de Dados.

```
[96] df_rt[col].head()

   body_temperature  sleeping_duration  walking_capacity  ruminating  health_status
0         0.529412         0.159091         0.900036         0.750         0.0
1         0.666667         0.318182         0.846117         0.700         1.0
2         0.607843         0.090909         0.983521         0.725         1.0
3         0.784314         0.227273         0.854356         0.750         0.0

[97] col_rt = ['body_temperature', 'sleeping_duration', 'walking_capacity', 'ruminating']
df_rt_reduzido = df_rt[col_rt]
dados_bovino = df_rt_reduzido.values
status = {0: "doente", 1: "saudável"}

for i, dp in enumerate(melhor_modelo.predict(dados_bovino)):
    print(f'Existe grande probabilidade de o bovino estar {status[dp]}')

Existe grande probabilidade de o bovino estar doente
Existe grande probabilidade de o bovino estar saudável
Existe grande probabilidade de o bovino estar saudável
Existe grande probabilidade de o bovino estar doente
```

Fonte: Autor (2022).

Porém, há um problema em relação ao treinamento dos algoritmos, onde a maioria dos algoritmos teve um score de treinamento e teste muito altos, o que normalmente

demonstra que possivelmente há um "vício" no algoritmo, não se podendo acreditar muito em suas predições. Isso pode ter ocorrido devido à baixa quantidade de linhas existentes no dataset utilizado. Portanto, é importante destacar a possibilidade da utilização de dados coletados para a predição da saúde desses bovinos. É necessário, no entanto, que sejam realizados mais testes e treinamentos com um dataset maior para que se possa ter uma maior confiabilidade nas predições do algoritmo.

## CONCLUSÃO

Este artigo abordou coleta de dados a partir de tecnologia existentes no campo é um modelo preditivo que define a saúde do animal, para a tomada de medidas de maneira mais ágil, aumentando assim a possibilidade de recuperar esse animal.

Para isso utilizou-se de dispositivos inteligentes, como balanças, tags e cabrestos, a pecuária está gerando uma grande quantidade de dados que podem fornecer informações precisas aos produtores, garantindo a qualidade do produto, prevenindo a perda de animais e a disseminação de doenças.

Apesar do problema de poucas linhas existentes no dataset e dos dados serem de bovinos de outro país, o trabalho conseguiu demonstrar que é possível prever a saúde dos bovinos com base em dados coletados. Essa informação é crucial para a intervenção humana e tratamento individual de cada animal.

Como trabalhos futuros, têm-se o objetivo da criação de um protótipo para a coleta desses dados, podendo assim ter uma quantidade significativa de dados para um treinamento e predição com mais assertividade da saúde do bovino com base em culturas existentes no Brasil. Para melhorar a confiabilidade das predições do algoritmo, alguns testes que poderiam ser realizados são:

- Cross-validation: é uma técnica que consiste em dividir o dataset em  $k$  partes iguais, treinar o modelo em  $k-1$  partes e testá-lo na parte restante. Esse processo é repetido  $k$  vezes, alternando a parte de teste em cada iteração. Isso ajuda a avaliar a capacidade do modelo de generalizar para novos dados.

- Aumento do dataset: como mencionado anteriormente, a baixa quantidade de linhas no dataset pode ter afetado o desempenho do algoritmo. Portanto, uma solução seria aumentar o tamanho do dataset, coletando mais dados de animais saudáveis e doentes.
- Seleção de features: pode ser que algumas das features utilizadas no modelo não sejam relevantes para a predição da saúde dos animais. Portanto, é importante realizar uma análise de quais features são mais importantes e descartar as que não contribuem para a predição.
- Comparação com outros modelos: é importante comparar o desempenho do modelo utilizado com outros modelos de machine learning para avaliar se o modelo escolhido é o mais adequado para o problema em questão.

Esses são apenas alguns exemplos de testes que podem ser realizados para melhorar a confiabilidade das predições do algoritmo. Além disso, com esses dados e o modelo preditivo treinado, seria possível realizar a criação de um software, onde além de demonstrar a saúde do animal poderia realizar a rastreabilidade e trazer várias outras informações importantes para que o produtor tenha maior controle sobre o seu rebanho.

## REFERENCES

AKHTAR, A. et al. COVID-19 Detection from CBC using Machine Learning Techniques. **International Journal of Technology, Innovation and Management (IJTIM)**, v. 1, n. 2, p. 65–78, 7 dez. 2021.

BOLFE, É. et al. **Tecnologias digitais na pecuária: aplicações, desafios e expectativas**. [s.l.] Embrapa, 2021.

CHAVES, B. B. **Estudo do algoritmo adaboost de aprendizagem de máquina aplicado a sensores e sistemas embarcados**. [s.l.] Universidade de São Paulo, 2012.

CHEN, J. et al. Big data challenge: a data management perspective. **Frontiers of Computer Science**, v. 7, n. 2, p. 157–164, abr. 2013.

COIMMA. **Coimma Troncos e Balanças**. Disponível em: <<https://coimma.com.br/balpass>>. Acesso em: 29 nov. 2022.

EL NAQA, I.; MURPHY, M. J. What Is Machine Learning? Em: EL NAQA, I.; LI, R.; MURPHY, M. J. (Eds.). **Machine Learning in Radiation Oncology**. Cham: Springer International Publishing, 2015. p. 3–11.

EMBRAPA. **Qualidade da carne bovina**. Disponível em:  
<<https://www.embrapa.br/qualidade-da-carne/carne-bovina>>. Acesso em: 29 nov. 2022.

INDEXT. **BOVINE ELECTRONIC PLATAFORM**. , 2022. Disponível em:  
<<https://www.indext.com.br/bep>>. Acesso em: 29 nov. 2022

KAMEL, H.; ABDULAH, D.; AL-TUWAIJARI, J. M. **Cancer Classification Using Gaussian Naive Bayes Algorithm**. 2019 International Engineering Conference (IEC). **Anais...** Em: 2019 INTERNATIONAL ENGINEERING CONFERENCE (IEC). Erbil, Iraq: IEEE, jun. 2019. Disponível em:  
<<https://ieeexplore.ieee.org/document/8950650/>>. Acesso em: 20 jul. 2023

KIRCHNER, J.; HEBERLE, A.; LOWE, W. **Classification vs. Regression - Machine Learning Approaches for Service Recommendation Based on Measured Consumer Experiences**. 2015 IEEE World Congress on Services. **Anais...** Em: 2015 IEEE WORLD CONGRESS ON SERVICES (SERVICES). New York City, NY, USA: IEEE, jun. 2015. Disponível em: <<http://ieeexplore.ieee.org/document/7196537/>>. Acesso em: 20 jul. 2023

LOBO, L. C. Inteligência artificial, o Futuro da Medicina e a Educação Médica. **Revista Brasileira de Educação Médica**, v. 42, n. 3, p. 3–8, set. 2018.

MARIA NAVIN, J. R.; PANKAJA, R. Performance Analysis of Text Classification Algorithms using Confusion Matrix. p. 2454–4698, 2016.

NAPOLEÃO, B. M. **Árvores Decisórias**. Disponível em:  
<<https://ferramentasdaqualidade.org/arvores-decisorias>>. Acesso em: 29 nov. 2022.

NISSILA, L. A. **Análise Comparativa de Métodos Computacionais para a Classificação de Estoque: Redes Neurais, kNN e SVM**. Ouro Preto/MG: Universidade Federal de Ouro Preto, 2023.

OLIVEIRA JÚNIOR, G. M. **Máquina de Vetores Suporte: estudo e análise de parâmetros para otimização de resultado**. Trabalho de Graduação—Pernambuco/PE: Universidade Federal de Pernambuco, 2010.

PORTAL DO AGRONEGÓCIO. **Zoetis lança o mais avançado sistema para monitoramento de vacas leiteiras**. Disponível em:  
<<https://www.portaldoagronegocio.com.br/pecuaria/bovinos-leite/noticias/zoetis-lanca-o-mais-avancado-sistema-para-monitoramento-de-vacas-leiteiras>>. Acesso em: 29 nov. 2022.

RAMCHOUN, H. et al. Multilayer Perceptron: Architecture Optimization and Training. **International Journal of Interactive Multimedia and Artificial Intelligence**, v. 4, n. 1, p. 26, 2016.

RANJAN, G. S. K.; KUMAR VERMA, A.; RADHIKA, S. **K-Nearest Neighbors and Grid Search CV Based Real Time Fault Monitoring System for Industries**. 2019 IEEE 5th International Conference for Convergence in Technology (I2CT). **Anais...**

Em: 2019 IEEE 5TH INTERNATIONAL CONFERENCE FOR CONVERGENCE IN TECHNOLOGY (I2CT). Bombay, India: IEEE, mar. 2019. Disponível em: <<https://ieeexplore.ieee.org/document/9033691/>>. Acesso em: 20 jul. 2023

REMÍGIO, M. **Maquinas de Vetores de Suporte**. Disponível em: <<https://medium.com/@msremigio/>>. Acesso em: 29 nov. 2022.

RODRIGUES. **Internet das Coisas monitora produtividade e bem-estar animal em sistemas de ILPF**. Disponível em: <<https://www.embrapa.br/busca-de-noticias/-/noticia/63903853/internet-das-coisas-monitora-productividade-e-bem-estar-animal-em-sistemas-de-ilpf>>. Acesso em: 29 nov. 2022.

SACRAMENTO, G. **ÁRVORE DE DECISÃO: ENTENDA ESSE ALGORITMO DE MACHINE LEARNING**. Disponível em: <<https://blog.somostera.com/data-science/arvores-de-decisao>>. Acesso em: 30 nov. 2022.

SMARTBOW. **Herd Monitoring Software | SMARTBOW.** , 2022. Disponível em: <<https://www.smartbow.com/en-gb/home.aspx>>. Acesso em: 29 nov. 2022

SUJATHA, P.; MAHALAKSHMI, K. **Performance Evaluation of Supervised Machine Learning Algorithms in Prediction of Heart Disease**. 2020 IEEE International Conference for Innovation in Technology (INOCON). **Anais...** Em: 2020 IEEE INTERNATIONAL CONFERENCE FOR INNOVATION IN TECHNOLOGY (INOCON). Bangluru, India: IEEE, 6 nov. 2020. Disponível em: <<https://ieeexplore.ieee.org/document/9298354/>>. Acesso em: 20 jul. 2023

WU, S.; NAGAHASHI, H. Parameterized AdaBoost: Introducing a Parameter to Speed Up the Training of Real AdaBoost. **IEEE Signal Processing Letters**, v. 21, n. 6, p. 687–691, jun. 2014.