

**BONIFÁCIO LEITE DE OLIVEIRA FILHO**

**PROPOSTA DE SISTEMA INTELIGENTE PARA GERENCIAMENTO DE  
HIDROPONIA**

Trabalho de Conclusão de Curso apresentado à banca avaliadora do Curso de Engenharia de Computação, da Escola Superior de Tecnologia, da Universidade do Estado do Amazonas, como pré-requisito para obtenção do título de Engenheiro de Computação.

Orientador(a): Prof. Dr. Luis Cuevas Rodriguez

Manaus – Dezembro – 2024

## Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).  
**Sistema Integrado de Bibliotecas da Universidade do Estado do Amazonas.**

D278p

de Oliveira Filho, Bonifácio Leite

Proposta de Sistema Inteligente para Gerenciamento de Hidroponia  
/ Bonifácio Leite de Oliveira Filho . Manaus : [s.n], 2024.

93 f.: color.; 21,0 cm.

TCC - Graduação em Engenharia de Computação- Universidade do  
Estado do Amazonas, Manaus, 2024.

Inclui Bibliografia.

Inclui Anexo.

Orientador: Luis Cuevas Rodriguez.

1. Hidroponia. 2. Inteligência Artificial. 3. Automação. 4.  
Gerenciamento de Cultivo. I. Luis Cuevas Rodriguez (Orient.) II.  
Universidade do Estado do Amazonas. III. Título

CDU(1997)62:004



## FOLHA DE APROVAÇÃO

### PROPOSTA DE SISTEMA INTELIGENTE PARA GERENCIAMENTO DE HIDROPONIA

#### BONIFÁCIO LEITE DE OLIVEIRA FILHO

Trabalho de Conclusão de Curso defendido e aprovado pela banca avaliadora constituída pelos professores:

---

Prof. Dr. Luis Cuevas Rodríguez  
Presidente

---

Prof. DSc. Raimundo Corrêa de Oliveira  
Avaliador

---

Prof. DSc. Jucimar Maia da Silva Júnior  
Avaliador

# Resumo

Esta monografia apresenta o desenvolvimento de um sistema inteligente para o gerenciamento de hidroponia, integrando tecnologias de IoT, análise de dados e modelos de inteligência artificial (IA) para otimizar o cultivo hidropônico. O sistema foi projetado para monitorar continuamente os parâmetros do cultivo, automatizar tarefas manuais e oferecer recomendações em tempo real. A arquitetura proposta inclui dispositivos IoT responsáveis pela coleta de dados ambientais e nutricionais, controle de atuadores e comunicação com um servidor na nuvem, onde os dados são armazenados e processados.

Os dados coletados alimentam um servidor de IA, que utiliza modelos preditivos e de recomendação para inferir condições ideais e ajustar os parâmetros de cultivo, como nutrientes e irrigação. Além disso, o sistema fornece um painel de controle interativo, permitindo aos usuários monitorar o estado dos dispositivos, configurar automações personalizadas e acessar o histórico de dados. O uso de modelos de IA, como Random Forest, Redes Neurais e Gradient Boosting, aliado a um sistema de recomendação de fertilizantes, demonstrou eficácia na previsão e ajuste de condições agrícolas.

Com a capacidade de adaptar-se continuamente às necessidades específicas de diferentes tipos de cultivo, o sistema se apresenta como uma solução promissora para melhorar a produtividade, reduzir desperdícios e promover a sustentabilidade do cultivo hidropônico.

**Palavras-Chave:** Hidroponia; Inteligência Artificial; Automação; Gerenciamento de Cultivo;

# Abstract

This monograph presents the development of an intelligent system for hydroponic management, integrating IoT technologies, data analysis, and artificial intelligence (AI) models to optimize hydroponic farming. The system was designed to continuously monitor cultivation parameters, automate manual tasks, and provide real-time recommendations. The proposed architecture includes IoT devices responsible for collecting environmental and nutritional data, controlling actuators, and communicating with a cloud server where the data is stored and processed.

The collected data feeds an AI server that utilizes predictive and recommendation models to infer ideal conditions and adjust cultivation parameters such as nutrients and irrigation. Furthermore, the system provides an interactive dashboard, enabling users to monitor device states, configure personalized automations, and access historical data. The use of AI models such as Random Forest, Neural Networks, and Gradient Boosting, combined with a fertilizer recommendation system, demonstrated effectiveness in predicting and adjusting agricultural conditions.

With the ability to continuously adapt to the specific needs of different types of crops, the system emerges as a promising solution to enhance productivity, reduce waste, and promote the sustainability of hydroponic farming.

**Keywords:** Hydroponics; Artificial Intelligence; Automation; Cultivation Management.

# Agradecimentos

Agradeço ao meu pai, por todo esforço, por ser meu melhor amigo, por ser a minha inspiração para o homem que quero me tornar e por sempre ter um presunto irado.

Agradeço à minha mãe, por todo carinho, paciência, puxões de orelha e por fazer o melhor bolo do mundo.

Agradeço à minha namorada, Fátima, pela dedicação, por ser minha companheira e por me apoiar.

Agradeço ao meu amigo William, por ser meu parceiro de componentes, por ser meu sócio e por embarcar comigo em qualquer empreitada que pode se imaginar.

Agradeço à minha amiga Rafaela, pela ajuda nos momentos que precisei, pelo prú e por deixar meus dias no LUDUS mais divertidos.

Agradeço ao professor Cuevas, que foi muito mais do que o orientador deste trabalho ou professor, foi um orientador para a minha vida. Agradeço por me ensinar a como ser uma pessoa e um profissional melhor, e, principalmente, por me ensinar a ter temperança.

Agradeço ao professor Jucimar, pela oportunidade de participar do LUDUS, uma época que lembro com alegria e carinho, pelos ensinamentos e principalmente pela paciência no meu processo de amadurecimento como profissional.

Agradeço ao professor Raimundo, por tudo que aprendi com nossas conversas e por estar sempre disposto a orientar as minhas ideias mais ousadas.

Agradeço ao professor Flávio, pela amizade, conversas e orientações que duravam horas, mas pelas quais sempre tive muito apreço.

Agradeço aos meus professores por todo o ensinamento que me proporcionaram. Ensino esse que foi imprescindível para a minha formação e para chegar até aqui.

Por fim, agradeço à Universidade do Estado do Amazonas, que me proporcionou a oportunidade de me tornar um profissional de qualidade dentro da minha área e de fazer amigos que levarei para o resto da vida.

# Sumário

<b>Lista de Tabelas</b>	<b>ix</b>
<b>Lista de Figuras</b>	<b>xi</b>
<b>Lista de Códigos</b>	<b>xi</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Descrição do Problema . . . . .	1
1.2 Objetivos . . . . .	2
1.2.1 Objetivo Geral . . . . .	2
1.2.2 Objetivos Específicos . . . . .	2
1.3 Trabalhos Relacionados . . . . .	3
1.4 Metodologia . . . . .	8
<b>2 Referencial Teórico</b>	<b>11</b>
2.1 Hidroponia . . . . .	11
2.1.1 Vantagens . . . . .	11
2.1.2 Tipos de Sistemas Hidropônicos . . . . .	12
2.2 IoT e Automação no Cultivo Hidropônico . . . . .	15
2.2.1 Internet das Coisas - IoT . . . . .	15
2.2.2 Aplicações de IoT na Agricultura . . . . .	16
2.3 Inteligência Artificial e Otimização de Recursos . . . . .	16
2.4 Modelos de Inteligência Artificial . . . . .	19

2.4.1	Regressão Linear Múltipla:	19
2.4.2	Random Forest:	20
2.4.3	Gradient Boosting Regression:	20
2.4.4	Rede Neural Multicamadas (MLP):	21
2.5	Identificação de Outliers com Intervalo Interquartilico (IQR)	21
2.6	Validação de Modelos de Inteligência Artificial	23
2.6.1	Erro Quadrático Médio (MSE)	23
2.6.2	Coefficiente de Determinação ( $R^2$ )	23
2.6.3	Interpretação das Métricas	24
2.6.4	Limitações e Considerações	24
2.7	Mínimos Quadrados Lineares	24
2.7.1	Definição Matemática	25
2.7.2	Resíduos	26
2.7.3	Aplicação no Contexto de Fertilizantes	26
2.7.4	Modelo Prático	26
<b>3</b>	<b>Desenvolvimento</b>	<b>28</b>
3.1	O Estudo	28
3.2	Levantamento de Requisitos do Sistema	30
3.3	Levantamento de Casos de Uso	31
3.4	Componentes do Dispositivo Embarcado	36
3.5	Descrição Geral do Diagrama de Componentes	39
3.6	Diagrama de Sequência para Treinamento e Inferência do Modelo de Inteligência Artificial	40
3.7	Datasets	42
3.7.1	Dataset de soluções nutritivas <code>pro_crop_data.csv</code>	43
3.7.2	Dataset de fertilizantes <code>pro_fert_data.csv</code>	44
3.8	Dataset <code>pro_crop_data.csv</code>	46
3.8.1	Análise exploratória do dataset	46

<b>3.9 Dataset pro_fert_data.csv</b> . . . . .	54
<b>3.9.1 Análise exploratória do dataset</b> . . . . .	54
<b>3.10 Proposta de Pipeline para Recomendação de Fertilizantes</b> . . . . .	58
<b>3.11 Regressão Linear Múltipla</b> . . . . .	59
<b>3.11.1 Preparação e Tratamento dos Dados</b> . . . . .	59
<b>3.11.2 Criação da Variável-Alvo</b> . . . . .	60
<b>3.11.3 Divisão dos Dados</b> . . . . .	60
<b>3.11.4 Treinamento do Modelo</b> . . . . .	60
<b>3.11.5 Resultados</b> . . . . .	61
<b>3.12 Gradient Boosting Regressor</b> . . . . .	62
<b>3.12.1 Preparação e Tratamento dos Dados</b> . . . . .	62
<b>3.12.2 Treinamento do Modelo</b> . . . . .	63
<b>3.12.3 Resultados</b> . . . . .	63
<b>3.13 Redes Neurais Multicamadas</b> . . . . .	64
<b>3.13.1 Preparação e Tratamento dos Dados</b> . . . . .	64
<b>3.13.2 Treinamento do Modelo</b> . . . . .	65
<b>3.13.3 Resultados</b> . . . . .	65
<b>3.14 Random Forest Regressor</b> . . . . .	66
<b>3.14.1 Preparação e Tratamento dos Dados</b> . . . . .	66
<b>3.14.2 Treinamento do Modelo</b> . . . . .	67
<b>3.14.3 Resultados</b> . . . . .	67
<b>3.15 Sistema de Recomendação de Fertilizantes</b> . . . . .	68
<b>3.15.1 Estrutura do Sistema</b> . . . . .	68
<b>3.15.2 Processos e Decisões</b> . . . . .	69
<b>3.15.3 Resultados Esperados</b> . . . . .	71
<b>3.15.4 Exemplo de Aplicação</b> . . . . .	71
<b>4 Conclusão</b> . . . . .	<b>72</b>
<b>4.1 Trabalhos Futuros</b> . . . . .	73



# Lista de Tabelas

2.1	Aplicações de IA no setor de agricultura - Fonte: <a href="#">Raman et al. 2024</a>	18
3.1	Tabela de Requisitos Funcionais de Referência	29
3.2	Requisitos Não Funcionais de Referência	29
3.3	Tabela com os detalhes das culturas	43
3.4	Tabela com os detalhes dos fertilizantes	45
3.5	Descrição dos tipos de dados das colunas do dataset <code>pro_crop_data.csv</code>	47
3.6	Estatísticas descritivas das variáveis numéricas do dataset <code>pro_crop_data.csv</code>	48
3.7	Descrição dos tipos de dados das colunas do dataset <code>pro_fert_data.csv</code>	55
3.8	Estatísticas descritivas das variáveis numéricas do dataset <code>pro_fert_data.csv</code>	56
3.9	Resultados da Regressão Linear Múltipla	61
3.10	Resultados do Gradient Boosting Regressor	63
3.11	Resultados das Redes Neurais Multicamadas	66
3.12	Resultados do Random Forest Regressor	68
4.1	Descrição dos Requisitos do Sistema	75
4.2	Critérios de Aceitação dos Requisitos do Sistema	79

# Lista de Figuras

1.1 Sistema proposto por <a href="#">Chowdhury et al. 2020</a> .	4
1.2 Sistema proposto por <a href="#">Verma e Gawade 2021</a> .	5
1.3 Sistema proposto por <a href="#">Hostalrich et al. 2022</a> .	7
1.4 Sistema proposto por <a href="#">Hostalrich et al. 2022</a> .	8
1.5 Modelo de desenvolvimento incremental.	10
2.1 Diagrama do Sistema NFT.	12
2.2 Diagrama Sistema Deep Water Culture - DWC - Fonte: <a href="#">Ecocenter 2023</a> .	13
2.3 Diagrama do Sistema Aeroponia - Fonte: <a href="#">Ecocenter 2023</a> .	13
2.4 Diagrama do Sistema Wick System - Fonte: <a href="#">Ecocenter 2023</a> .	14
2.5 Diagrama do Sistema Ebb and Flow - Fonte: <a href="#">Ecocenter 2023</a> .	14
2.6 Diagrama do Sistema Drip System - Fonte: <a href="#">Ecocenter 2023</a> .	15
3.1 Diagrama de Casos de Uso.	32
3.2 Diagrama de Componentes.	39
3.3 Diagrama de Sequência da Funcionalidade de Treino e Inferência.	41
3.4 Distribuições das variáveis numéricas do dataset <code>pro_crop_data.csv</code> .	50
3.5 Boxplot das variáveis numéricas do dataset <code>pro_crop_data.csv</code> .	52
3.6 Distribuição da coluna <code>crop_growth_stage</code> .	53
3.7 Distribuição da coluna <code>crop_type</code> .	54
3.8 Distribuições das variáveis numéricas do dataset <code>pro_fert_data.csv</code> .	57

# Lista de Códigos

# Capítulo 1

## Introdução

### 1.1 Descrição do Problema

A hidroponia é uma técnica de cultivo que permite o crescimento de plantas sem o uso de solo, utilizando soluções nutritivas em meio aquoso. Esta técnica é útil principalmente em locais onde o solo é escasso ou de baixa qualidade, como áreas urbanas densamente povoadas ou regiões com solo infértil. Além disso, a hidroponia é uma prática sustentável, pois promove a reciclagem de água, sendo uma solução econômica em regiões onde a água é um recurso limitado.

No entanto, um dos principais problemas ao gerenciar um sistema hidropônico é a necessidade de monitoramento contínuo e ajustes manuais dos parâmetros de cultivo, como pH, temperatura e níveis de nutrientes. Sem um sistema automatizado e inteligente, esses ajustes dependem de intervenção humana, o que pode levar a erros e, conseqüentemente, a perdas na produção e aumento dos custos operacionais.

Outro desafio é a falta de integração entre sistemas hidropônicos e tecnologia, como sensores e inteligência artificial (IA), que poderiam otimizar o cultivo. Sem a capacidade de analisar dados em tempo real e ajustar os parâmetros automaticamente, os produtores têm dificuldades em manter as condições ideais para as plantas, afetando a qualidade e a quantidade da produção. Além disso, a gestão de dados é um problema crítico, já que os sistemas hidropônicos geram muitos dados que precisam ser armazenados, analisados e usados para decisões. Sem um sistema eficiente, esses dados podem se tornar desorganizados e subutilizados, dificultando a melhoria

contínua do cultivo.

Diante desses desafios, este trabalho propõe o desenvolvimento de um sistema inteligente para o gerenciamento de hidroponia. A ideia é integrar tecnologias de sistemas embarcados, armazenamento e análise de dados e modelos de IA para resolver esses problemas, proporcionando uma gestão mais eficiente, sustentável e automatizada. O objetivo é melhorar a produtividade e a qualidade dos cultivos hidropônicos, ao mesmo tempo em que se promovem práticas agrícolas mais eficientes e economicamente viáveis.

## **1.2 Objetivos**

### **1.2.1 Objetivo Geral**

O objetivo deste trabalho é propor um sistema inteligente para o gerenciamento de hidroponia, integrando tecnologias de análise de dados, inteligência artificial (IA), sistemas embarcados e cloud, visando otimizar o cultivo através do monitoramento contínuo e automação de processos, promovendo uma gestão mais eficiente, além de melhorar a produtividade e a adaptabilidade do sistema às necessidades específicas de cada tipo de cultivo.

### **1.2.2 Objetivos Específicos**

1. Estudar e implementar tecnologias de análise de dados para otimização do cultivo hidropônico, com foco em monitoramento em tempo real;
2. Desenvolver modelos de inteligência artificial para o ajuste automático de parâmetros de cultivo, como pH, temperatura e nutrientes;
3. Projetar um sistema automatizado para o monitoramento contínuo e ajustes em tempo real dos parâmetros de cultivo, visando melhorar a eficiência e a sustentabilidade do processo hidropônico;
4. Garantir que o sistema seja capaz de se adaptar às diferentes necessidades dos tipos de cultivo hidropônico, com ajustes automáticos baseados em dados coletados.;

### 1.3 Trabalhos Relacionados

No contexto do desenvolvimento de sistemas inteligentes para gerenciamento de hidroponia, diversos estudos têm se dedicado à aplicação de tecnologias para otimizar o cultivo hidropônico. Esses sistemas visam unir soluções de armazenamento e análise de dados, com o uso de inteligência artificial (IA), sistemas embarcados e cloud computing para permitir o monitoramento em tempo real. A combinação dessas tecnologias busca não apenas melhorar a eficiência do cultivo, mas também oferecer um modelo de fácil manutenção, adaptável às diferentes necessidades de cada tipo de plantio, sem depender exclusivamente da intervenção humana para ajustes nos parâmetros ambientais e de nutrientes. Além disso, o uso de IA permite que o sistema aprenda e se ajuste automaticamente, promovendo uma gestão inteligente e otimizada do ambiente hidropônico.

[Chowdhury et al. 2020](#) aborda o desenvolvimento de um sistema automatizado para cultivo hidropônico vertical baseado em IoT, projetado para enfrentar os desafios climáticos da região do Golfo, Qatar. A proposta central é oferecer uma solução prática para a produção de alimentos em espaços internos. A necessidade desse sistema é acentuada pela aridez do clima, pela limitada disponibilidade de terras aráveis e pela dependência de importações, que chegam a suprir 90% das necessidades alimentares do país.

O sistema proposto [1.1](#) combina a técnica de cultivo em filme de nutrientes (NFT) com sensores e dispositivos IoT para monitoramento em tempo real de variáveis críticas, como pH, condutividade elétrica, temperatura, umidade e nível de água. Essas informações são enviadas a uma plataforma IoT, permitindo ao usuário acessar os dados remotamente por meio de dispositivos móveis ou computadores. A automação reduz a intervenção humana e garante condições ideais para o crescimento das plantas, além de enviar alertas em caso de falhas, como problemas no bombeamento de água.

Um dos pontos relevantes do projeto é a eficiência energética e hídrica. O sistema visa manter o consumo de energia controlado, inclusive monitorando o uso de ar-condicionado, essencial para regular a temperatura interna entre 19 e 28°C. No aspecto hídrico, o sistema consome apenas 8 a 10 litros de água por mês,

O projeto demonstrou sucesso em cultivos como hortelã, alface, tomate, pepino e outros vegetais, demonstrando sua viabilidade. A integração com a plataforma Thingspeak permite visualização em tempo real dos sensores e do estado dos atuadores do sistema. Além disso, a abordagem foi projetada para ser acessível, simples e de baixo custo, facilitando sua adoção em residências e pequenos espaços, além de deixar em aberto para trabalhos futuros a integração de algoritmos de aprendizado de máquina, otimizando ainda mais os parâmetros de controle.

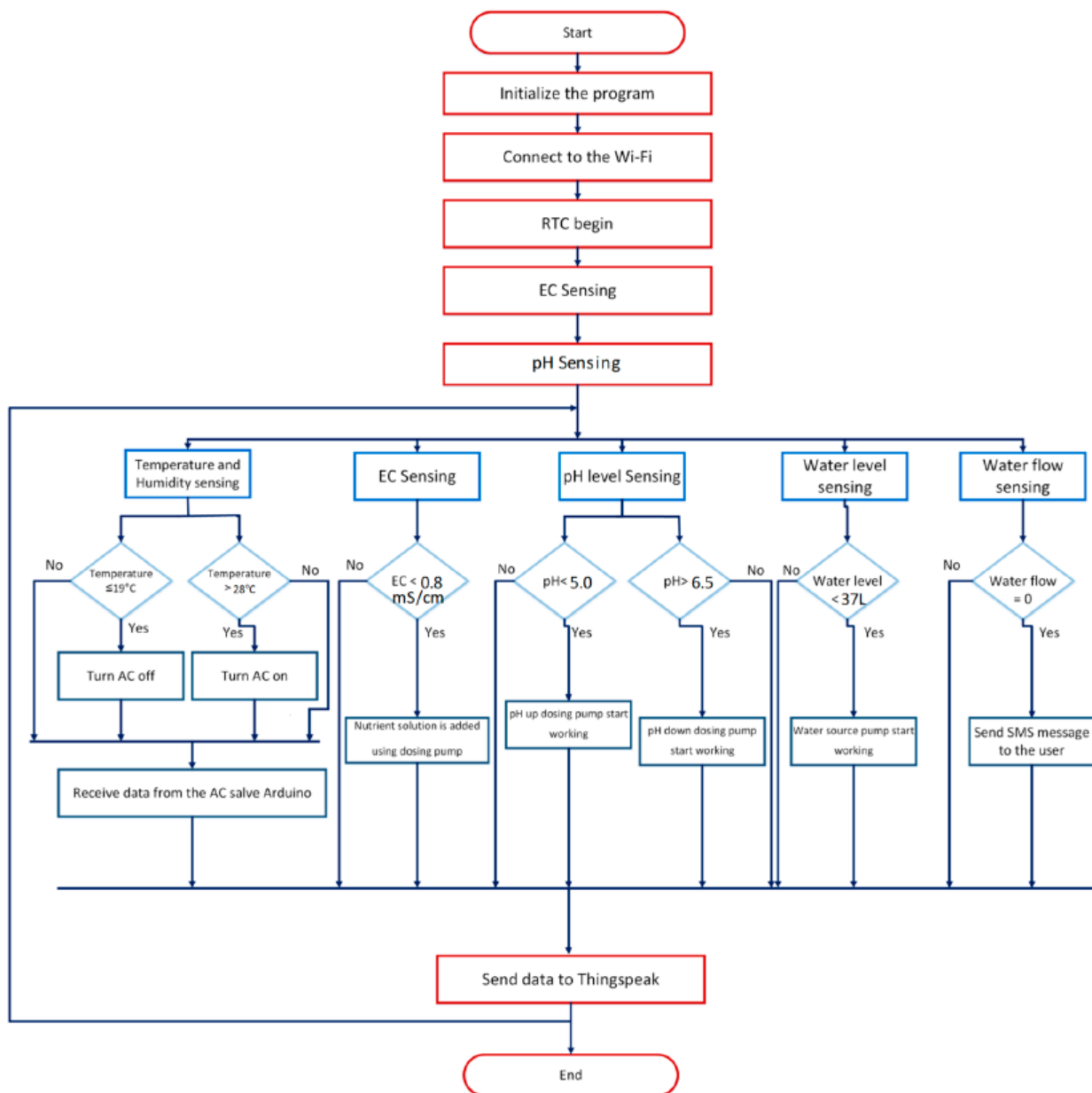


Figura 1.1: Sistema proposto por [Chowdhury et al. 2020](#).

[Verma e Gawade 2021](#) explora a aplicação de aprendizado de máquina para prever o cres-

cimento absoluto (CGR, ou "Absolute Crop Growth Rate") de tomates cultivados em sistemas hidropônicos. A pesquisa parte da premissa de que o manejo eficiente de nutrientes e a análise detalhada de fatores de crescimento podem ser utilizadas para melhorar a produtividade, especialmente em regiões com recursos naturais limitados. A solução proposta combina dados agrônômicos e técnicas de machine learning para melhorar os resultados do cultivo hidropônico, utilizando o sistema NFT (Técnica de Filme de Nutrientes).

O estudo envolveu a coleta de dados sobre parâmetros relevantes, como peso seco das plantas, limite de condutividade elétrica (EC), consumo de solução nutritiva (NS) e concentração de íons como sódio (Na), potássio (K), cálcio (Ca), magnésio (Mg) e nitrogênio (N). Os dados foram utilizados para treinar um modelo de aprendizado de máquina [1.1], que teve como objetivo prever o CGR das plantas em diferentes fases de desenvolvimento. O CGR é calculado como a taxa de variação do peso seco em função do tempo, podendo ser utilizado como indicador da eficiência de crescimento das plantas.

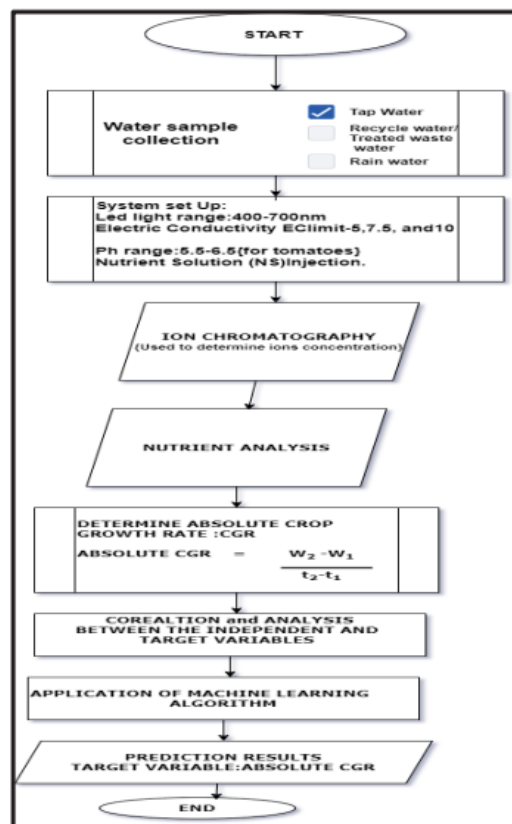


Figura 1.2: Sistema proposto por [Verma e Gawade 2021].

Os resultados obtidos demonstraram que o CGR variou significativamente durante o desenvolvimento das plantas. No estágio de amadurecimento dos frutos, o CGR atingiu o pico de 4,27 g/dia, enquanto no estágio inicial e no final da maturação o valor foi menor, indicando menor demanda de nutrientes. Além disso, a absorção de íons apresentou uma hierarquia: potássio foi o nutriente mais absorvido, seguido por nitrogênio, sódio, magnésio e cálcio. O baixo nível de absorção de cálcio apontou para a necessidade de ajustes na composição da solução nutritiva para melhorar o tamanho e a qualidade dos frutos obtidos.

O estudo realizou uma análise de correlação para identificar as variáveis mais influentes no crescimento das plantas. O peso seco das frutas e a EC apresentaram correlações positivas significativas, enquanto outros fatores, como a absorção de cálcio, tiveram correlação mais fraca. Isso indica que métodos tradicionais baseados apenas em EC podem ser insuficientes para um manejo eficiente de nutrientes, sendo necessário monitorar individualmente a concentração de íons.

O artigo demonstra como técnicas de aprendizado de máquina podem ser integradas ao manejo hidropônico para melhorar a eficiência do uso de recursos e a produtividade. O modelo desenvolvido oferece previsões do CGR, possibilitando ajustes na solução nutritiva e no manejo ambiental em tempo real.

[Hostalrich et al. 2022] discorre sobre o desenvolvimento de um sistema de gerenciamento inteligente para agricultura hidropônica baseado em tecnologia de Internet das Coisas (IoT). O objetivo é monitorar e controlar a produção, utilizando sensores e protocolos de comunicação sem fio para melhorar a eficiência, aumentar a produtividade e reduzir desperdício de recursos.

O sistema proposto é baseado em uma rede de sensores sem fio, que inclui dispositivos IoT conectados por protocolo Zigbee. Esse protocolo foi escolhido devido a sua baixa demanda energética, escalabilidade e capacidade de criar topologias em malha, facilitando a implementação e o crescimento do sistema. A arquitetura do sistema é dividida em quatro camadas: aplicação (interface do usuário), processamento (armazenamento e visualização de dados), transporte (comunicação entre dispositivos) e percepção (interação física com sensores e atuadores) [1.3].

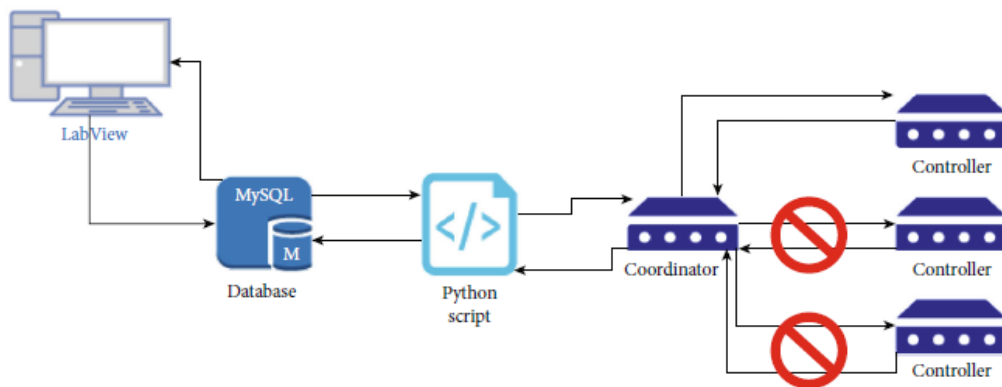


Figura 1.3: Sistema proposto por [Hostalrich et al. 2022](#).

Os dispositivos incluem sensores para medir parâmetros como temperatura ambiente, umidade, níveis de CO<sub>2</sub> e umidade do solo, além de sistemas externos como iluminação, refrigeração e ventilação. O sistema também possui uma interface gráfica desenvolvida em LabVIEW, que permite ao usuário configurar automações, definir períodos em que a iluminação do ambiente será ativada e monitorar os dados em tempo real. As automações são gerenciadas por um script em Python que processa os comandos enviados pela interface e os armazena em um banco de dados para execução.

Os resultados mostraram que as automações funcionaram conforme o esperado, ativando sistemas de refrigeração e ventilação assim que os limites de temperatura eram excedidos. Entre os benefícios citados estão a facilidade de uso e a adaptabilidade do sistema, que pode ser configurado para diferentes tipos de sensores e necessidades específicas de cultivo.

[Lukito, Baskoro e Ernawati 2024](#) trata do desenvolvimento de um sistema automatizado para controle e monitoramento da qualidade da água em sistemas hidropônicos, utilizando tecnologias baseadas em IoT e serviços em nuvem da Amazon Web Services (AWS). O objetivo principal é integrar hardware e software de forma eficiente para criar uma solução acessível e automatizada que auxilie no cultivo hidropônico com precisão e acessibilidade remota.

O sistema [1.4](#) é composto por sensores que monitoram parâmetros como pH, condutividade elétrica (EC), temperatura da água e temperatura ambiente. Um microcontrolador Arduino, utilizando ATmega2560, equipado com um módulo ESP8266, coleta esses dados e os envia para a nuvem através do protocolo MQTT, permitindo o controle automatizado e o monitoramento

em tempo real via dashboards. Essa arquitetura baseada na nuvem possibilita o ajuste remoto de nutrientes e pH, garantindo as condições necessárias para o crescimento das plantas.

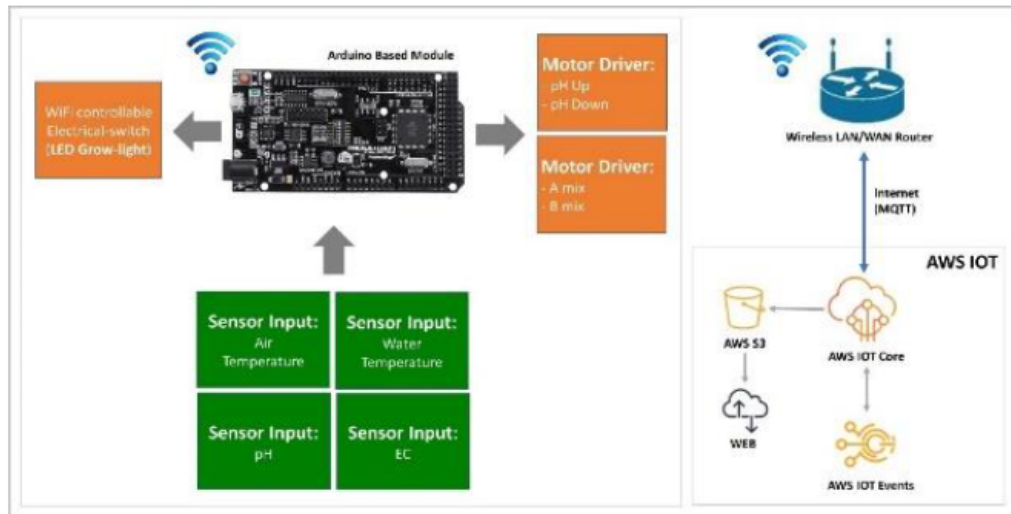


Figura 1.4: Sistema proposto por [Hostalrich et al. 2022](#).

O estudo apresenta várias etapas de desenvolvimento, incluindo a integração de dispositivos IoT com os módulos da AWS, como AWS IoT Core e AWS IoT Events, que permitem o acionamento de bombas dosadoras para ajuste automático dos parâmetros da água. Os resultados indicaram uma alta eficiência do sistema, com monitoramento contínuo e controle remoto estáveis, além de uma interface web estática que facilita o acesso ao sistema a partir de qualquer lugar. Entre os principais benefícios apontados estão a automação do controle de qualidade da água, a melhora na eficiência de manutenção dos níveis de nutrientes e pH, e a viabilização do monitoramento remoto, o que reduz significativamente a necessidade de intervenção humana.

## 1.4 Metodologia

O desenvolvimento deste trabalho foi realizado utilizando o método iterativo incremental. Segundo [Pressman 2016](#), este modelo organiza o desenvolvimento em sequências lineares escalonadas ao longo do tempo, cada uma produzindo "incrementos" entregáveis do projeto. Essa abordagem permitiu uma evolução contínua e flexível, possibilitando ajustes e melhorias em cada etapa do processo. O projeto foi estruturado em duas fases principais, correspondentes

às etapas de Trabalho de Conclusão de Curso 1 (TCC1) e Trabalho de Conclusão de Curso 2 (TCC2).

Na primeira fase, o foco esteve no estudo e concepção inicial da arquitetura do sistema de hidroponia inteligente. Esta etapa incluiu a análise de requisitos, a pesquisa das tecnologias adequadas para o desenvolvimento do hardware controlador e o levantamento das necessidades específicas do cultivo hidropônico. Sensores foram propostos para a coleta de dados ambientais e nutricionais, criando a base para o monitoramento contínuo das condições de cultivo. Além disso, foram elaborados diagramas de arquitetura para orientar o desenvolvimento posterior.

Na segunda fase, o trabalho concentrou-se na criação e validação dos modelos de inteligência artificial (IA) planejados para otimizar o cultivo hidropônico. Esses modelos foram desenvolvidos e treinados com dados representativos das condições de cultivo. Modelos como Regressão Linear Múltipla, Random Forest e Redes Neurais Multicamadas foram testados e avaliados quanto à sua capacidade de prever e ajustar parâmetros de cultivo, como nutrientes e irrigação.

Além dos modelos de IA, foi projetada a arquitetura completa do sistema, incluindo a integração com dispositivos IoT. Foi desenvolvido um plano de como esses dispositivos poderiam ser usados para coletar dados e controlar atuadores. Essa fase culminou na proposta de um sistema para monitoramento e recomendação de parâmetros críticos do cultivo.

Para guiar o desenvolvimento e documentar o processo, foram utilizados diagramas de componentes, sequência e casos de uso, descrevendo como as partes do sistema interagem entre si. Embora a implementação final do hardware e da integração em nuvem não tenha sido realizada, a elaboração desses elementos fornece uma base sólida para futuras implementações práticas.

A metodologia aplicada foi ilustrada por meio de um modelo de desenvolvimento incremental, que guiou o processo de criação e validação do sistema de hidroponia inteligente, como mostrado na Figura [1.5](#).

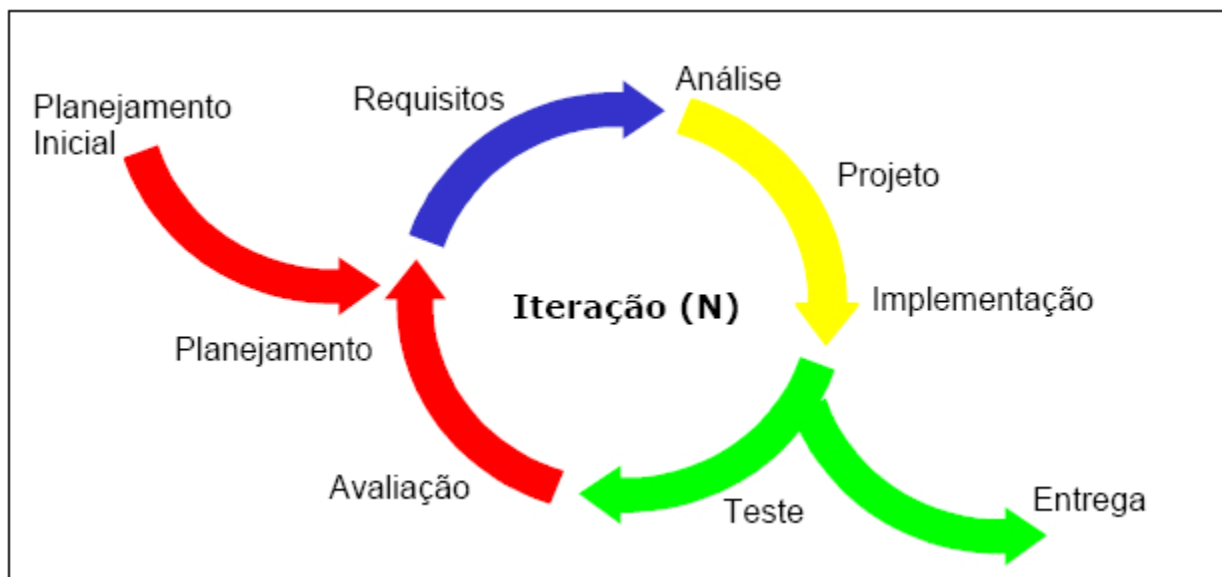


Figura 1.5: Modelo de desenvolvimento incremental

# Capítulo 2

## Referencial Teórico

Este trabalho apresenta uma proposta de sistema inteligente para o gerenciamento de hidroponia, integrando tecnologias de IoT e inteligência artificial com o objetivo de otimizar o cultivo hidropônico. Portanto, serão abordados conceitos atrelados a esse tema, como **Hidroponia, IoT e Automação no Cultivo Hidropônico, Inteligência Artificial e Otimização de Recursos, Modelos de Inteligência Artificial, Identificação de Outliers com Intervalo Interquartilico, Validação de Modelos de IA e Mínimos Quadrados Lineares.**

### 2.1 Hidroponia

Os sistemas hidropônicos são uma alternativa eficiente ao cultivo agrícola tradicional. Definidos como métodos de cultivo sem solo, esses sistemas utilizam soluções nutritivas diretamente na água, permitindo que as plantas obtenham os nutrientes necessários para o crescimento de maneira controlada [Son, Kim e Ahn 2020]. Essa técnica permite que as plantas absorvam diretamente os nutrientes essenciais, eliminando a necessidade de processos naturais de decomposição ou fertilização do solo para disponibilizar esses elementos.

#### 2.1.1 Vantagens

Essa abordagem oferece a possibilidade de cultivo em espaços limitados e em regiões com baixa qualidade do solo. Além disso, sistemas hidropônicos são conhecidos por sua eficiência no uso

de água, consumindo até 90% menos água do que os métodos tradicionais, e por proporcionar colheitas mais rápidas devido ao controle preciso das condições ambientais [Ghorbel et al. 2022].

Outra vantagem significativa dos sistemas hidropônicos é a redução do uso de pesticidas, pois o ambiente controlado minimiza o risco de pragas e doenças [Son, Kim e Ahn 2020]. Isso promove não apenas uma produção mais sustentável como também resulta em alimentos de maior qualidade. Também pode-se destacar que os sistemas hidropônicos permitem maior produtividade em comparação com o cultivo em solo, tornando-se uma solução viável para enfrentar os desafios da segurança alimentar global, especialmente em áreas urbanas e regiões com recursos agrícolas limitados [Ghorbel et al. 2022].

## 2.1.2 Tipos de Sistemas Hidropônicos

### 1. Técnica de Filme de Nutrientes (NFT)

A Técnica de Filme de Nutrientes (NFT) utiliza uma fina camada de solução nutritiva que flui continuamente pelas raízes das plantas. Essa camada é projetada para fornecer oxigênio suficiente às raízes, enquanto mantém um suprimento constante de nutrientes. O NFT é muito utilizado devido à sua simplicidade e eficiência em sistemas de cultivo hidropônico em larga escala, especialmente para culturas como alface e ervas. [Helmy et al. 2017]

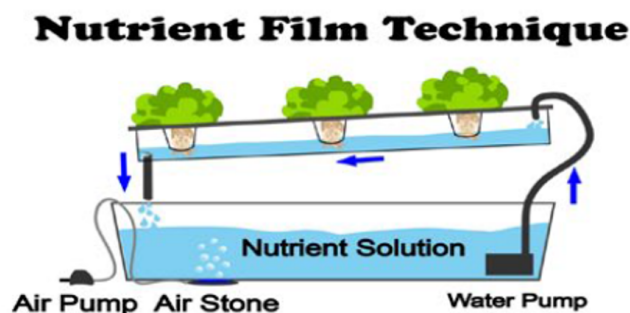


Figura 2.1: Diagrama do Sistema NFT.

### 2. Cultura em Água (Deep Water Culture - DWC)

No sistema de Cultura em Água, as plantas flutuam em uma solução nutritiva oxigenada. As raízes são submersas diretamente na solução, permitindo assim o contato contínuo

com os nutrientes e o oxigênio. O DWC é conhecido por sua eficiência no crescimento das plantas, principalmente em ambientes internos. Ele é ideal para o cultivo de plantas de rápido crescimento e de grande porte, como tomates e pepinos. O oxigênio é fornecido por bombas de ar, que evitam a asfixia das raízes. [Nursyahid et al. 2021]

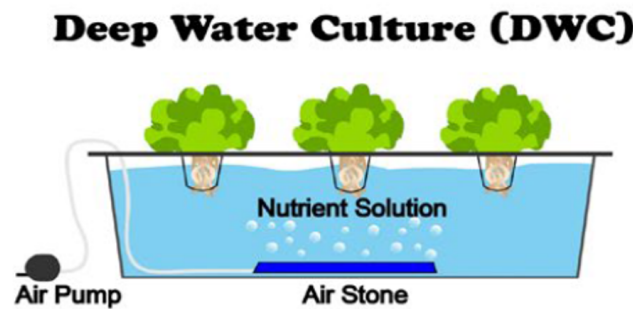


Figura 2.2: Diagrama Sistema Deep Water Culture - DWC - Fonte: [Ecocenter 2023].

### 3. Aeroponia

A aeroponia é um sistema em que as raízes das plantas ficam suspensas no ar e são pulverizadas com uma solução nutritiva em intervalos regulares. Esse método garante que as raízes recebam oxigênio em maior quantidade, o que promove um crescimento rápido e saudável. A aeroponia é eficaz para cultivos de alta qualidade e em espaços reduzidos, como em estufas urbanas. No entanto, requer um controle rigoroso da pulverização e manutenção frequente dos equipamentos para evitar obstruções. [Ritter et al. 2001]

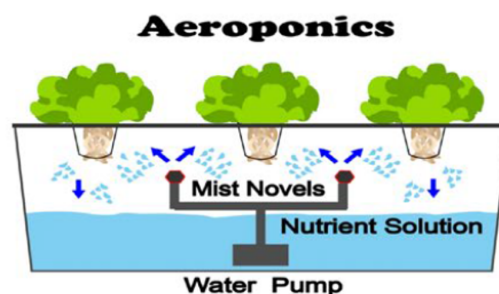


Figura 2.3: Diagrama do Sistema Aeroponia - Fonte: [Ecocenter 2023].

### 4. Wick System

O Wick System é um método de hidroponia em que se utiliza um pavio que conecta o reservatório de nutrientes ao substrato das plantas. A solução nutritiva é transportada de

forma passiva pelo pavio até as raízes. Esse sistema é ideal para pequenos cultivos e plantas de baixa demanda hídrica, como ervas e alface. Sua simplicidade o torna econômico e de fácil manutenção. [Harahap, Harahap e Gultom 2020]



Figura 2.4: Diagrama do Sistema Wick System - Fonte: [Ecocenter 2023].

## 5. Ebb and Flow

O sistema Ebb and Flow funciona inundando temporariamente o substrato das plantas com solução nutritiva, que depois é drenada de volta ao reservatório. Esse processo ocorre em ciclos controlados, garantindo que as raízes tenham acesso alternado a nutrientes e oxigênio. Esse sistema é adequado para uma ampla variedade de plantas, promovendo crescimento saudável em ambientes controlados. Contudo, requer maior monitoramento e manutenção para evitar entupimentos ou falhas no ciclo. [Daud, Handika e Bintoro 2018]

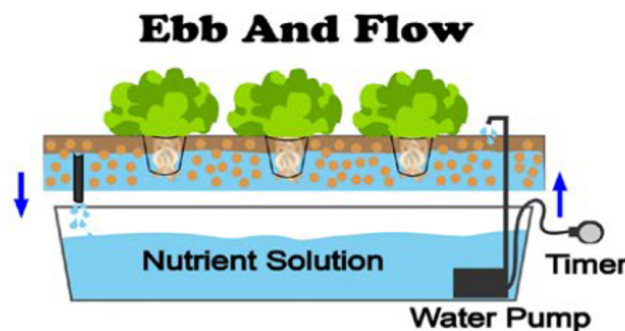


Figura 2.5: Diagrama do Sistema Ebb and Flow - Fonte: [Ecocenter 2023].

## 6. Drip System

O Drip System é um dos métodos mais utilizados na hidroponia comercial. Nesse sistema, uma solução nutritiva é gotejada diretamente na base de cada planta por meio de tubos e emissores. Isso permite um controle preciso da quantidade de nutrientes e água fornecida

a cada planta, tornando-o eficiente e adaptável para diferentes tipos de culturas. Embora exija um investimento inicial maior devido ao sistema de irrigação, ele é altamente eficiente para grandes plantações e ambientes controlados. [Shrestha e Dunn 2010]

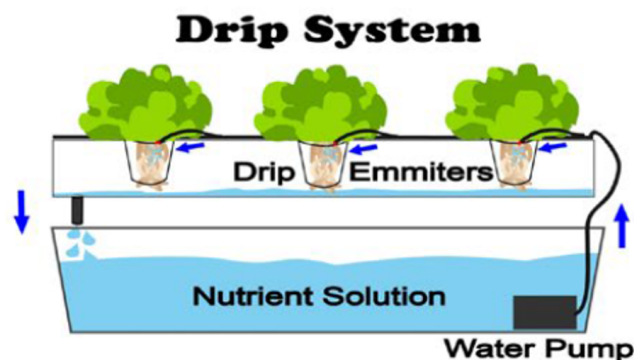


Figura 2.6: Diagrama do Sistema Drip System - Fonte: [Ecocenter 2023].

## 2.2 IoT e Automação no Cultivo Hidropônico

### 2.2.1 Internet das Coisas - IoT

A Internet das Coisas é um conceito que emergiu no final dos anos 1990, quando a comunicação entre dispositivos começou a ganhar forma devido à popularização da internet e sensores. O termo "IoT" foi caracterizado por descrever sistemas conectados capazes de interagir com o mundo físico por meio de dados em tempo real [Sharma, Shamkuwar e Singh 2019]. Desde então, a IoT evoluiu bastante, abrangendo agora diversas tecnologias, como sensores, atuadores, processamento de dados e comunicação em rede.

Atualmente, a IoT é utilizada em diversos setores. Suas aplicações incluem desde o monitoramento em tempo real em sistemas industriais até a criação de casas inteligentes e automação de processos agrícolas. No setor de saúde, por exemplo, dispositivos vestíveis coletam dados biométricos, permitindo diagnósticos mais rápidos e precisos. Na agricultura, IoT facilita o controle de irrigação, medição de nutrientes e monitoramento climático, promovendo eficiência e sustentabilidade [Suresh et al. 2014].

Além de suas amplas aplicações, a IoT se destaca por sua relevância nos dias de hoje, especialmente com a chegada de tecnologias como 5G, que aumentam a praticidade no de-

envolvimento e aplicação assim como a velocidade de transmissão de dados. Essa evolução permite que milhões de dispositivos funcionem simultaneamente, ampliando as possibilidades e aplicações de automação e tomada de decisão baseada em dados [Sharma, Shamkuwar e Singh 2019] [Suresh et al. 2014].

### 2.2.2 Aplicações de IoT na Agricultura

O uso da Internet das Coisas na agricultura vem desde o início dos anos 2000, quando dispositivos conectados começaram a ser integrados para aumentar a eficiência na produção agrícola. A primeira aplicação relevante foi em sistemas de monitoramento remoto de irrigação, permitindo o controle de bombas e o monitoramento do uso de água em tempo real. Esses sistemas eram baseados em sensores que forneciam dados sobre umidade do solo e temperatura, ajudando a reduzir desperdícios e otimizar recursos [Suresh et al. 2014].

Por exemplo, uma aplicação importante da IoT na agricultura é o monitoramento e manejo de rebanhos. Sensores conectados e dispositivos vestíveis, como coleiras inteligentes, permitem o rastreamento em tempo real da localização, saúde e comportamento dos animais. Esses dispositivos podem monitorar sinais vitais, como temperatura corporal e frequência cardíaca, identificando precocemente problemas de saúde e permitindo intervenções rápidas. Além disso, sistemas de IoT ajudam a otimizar a alimentação ao fornecer dados sobre os hábitos alimentares e o consumo de água dos animais, contribuindo para o aumento da eficiência na produção pecuária [Kamilaris, Kartakoullis e Prenafeta-Boldú 2017].

## 2.3 Inteligência Artificial e Otimização de Recursos

A Inteligência Artificial (IA) é um ramo da ciência da computação que busca criar sistemas capazes de realizar tarefas que normalmente requerem inteligência humana, como reconhecimento de padrões, aprendizado e tomada de decisão. Esses sistemas são desenvolvidos por meio de algoritmos e modelos matemáticos que simulam o funcionamento do cérebro humano. Concebida nos anos 1950, a IA vem evoluindo, permitindo sua aplicação em diversas áreas, incluindo

saúde, transporte, finanças e agricultura [Ayed e Hanana 2021] [Raman et al. 2024].

A IA tem sido aplicada a diversos setores ao oferecer soluções inteligentes para desafios complexos. Na saúde, por exemplo, algoritmos de aprendizado de máquina são usados para prever diagnósticos precoces, identificar padrões em exames médicos e desenvolver medicamentos. No setor de transporte, a IA é essencial para a operação de veículos autônomos, otimizando rotas e reduzindo acidentes. Já na indústria, sistemas baseados em IA gerenciam linhas de produção, detectando falhas em tempo real e ajustando parâmetros automaticamente para aumentar a eficiência operacional [Ayed e Hanana 2021]. Além disso, no varejo, a IA é aplicada na personalização de experiências do consumidor, análise de comportamento de compra e gerenciamento de estoques com base em previsões de demanda [Sharma, Shamkuwar e Singh 2019]. No setor agrícola, ela contribui para aumentar a produtividade, otimizar o uso de recursos e reduzir impactos ambientais. A Inteligência Artificial tem sido utilizada em práticas agrícolas por oferecer ferramentas avançadas para monitorar o solo, prever safras, detectar doenças em plantas e gerenciar sistemas de irrigação com precisão. As diversas aplicações de IA na agricultura estão detalhadas na Tabela 2.1

AI Technique	Uses
Fuzzy Logic: SRC-DSS	Can classify soils according to associated risks. [Lopez et al. 2008]
ANN	Prediction of enzyme activity and classification of soil structure, texture, moisture, and temperature. [Zhao X. e Elshorbagy 2009]
HONN	Soil moisture estimation. [Elshorbagy e Parasuraman 2008]
AI-based moisture analyzers	Precise temperature control. [Hanson e Others 2007]
Satellite imagery	Multi-depth soil temperature assessment. [Singh e Sanikhani 2018]
Non-tuned data intelligent model	Soil temperature assessment. [Sanikhani A. e Others 2018]
AI for mapping and modeling	Assessment of soil organic matter. [Ghorbani et al. 2019]
AI model	Assessment of soil fertility. [Kouadio et al. 2018]
Fuzzy Cognitive Map	Prediction of cotton yield and support for decision-making. [Papa-georgiou, Markinos e Gemtos 2011]
SVM-based model	Chlorophyll assessment. [Nieto et al. 2016]
Kernel of SVM model	Fertilizer management. [Mustafa et al. 2004]
AdaBoost	Crop management activities. [Hassanijalilian et al. 2020]
SLFNs	Crop monitoring. [O'Sullivan et al. 2019]
Regression tree technique	Crop yield prediction. [Park, Hwang e Vlek 2005]
Web-based intelligent disease diagnosis system	Accurate and rapid identification of diseases. [Kolhe et al. 2011]
Computer Vision System	High-speed multitasking diagnosis. [Balledda et al. 2014]
Genetic Algorithm	Quick resolution of plant pathological issues. [Kolhe et al. 2011]
CNN model	Detection of leaf diseases. [Selvaraj et al. 2019]
AI-based intelligent system	Tea leaf disease detection. [Ismail e Mustikasari 2013]
Deep neural networks	Identification of leaf diseases. [Sladojevic et al. 2016]
ANN-based control	Irrigation automation. [Kumar, Maurya e Verma 2022]
PLSR algorithms	Improved efficiency and economic feasibility. [El-Hendawy et al. 2019]
AI-IoT Pivot Irrigation	Smart irrigation. [Debauche et al. 2020]
Multilayer neural model	Reduced evaporation due to optimized scheduling. [Kolhe et al. 2011]
Machine learning algorithm	Prediction and mitigation of drought stress. [Arvind et al. 2017]
Fuzzy logic controller with wireless sensors	Drip irrigation automation. [Anand et al. 2015]
ANN (Feed Forward)	Optimized water usage in smart farms. [Cruz et al. 2017]
Zigbee protocol	Validated experimental results. [Al-Ali et al. 2015]

Tabela 2.1: Aplicações de IA no setor de agricultura - Fonte: [Raman et al. 2024]

## 2.4 Modelos de Inteligência Artificial

A Inteligência Artificial baseia-se em uma variedade de modelos matemáticos e computacionais, projetados para abordar problemas complexos em diversos domínios. Esses modelos utilizam métodos estatísticos e algoritmos para extrair padrões, realizar previsões e tomar decisões a partir de dados. No contexto deste trabalho, são explorados modelos relevante para a proposta atual, descritos a seguir com ênfase em suas características, fundamentos teóricos e aplicações práticas.

### 2.4.1 Regressão Linear Múltipla:

A Regressão Linear Múltipla é uma extensão da regressão linear simples, que modela a relação entre uma variável dependente  $y$  e várias variáveis independentes  $X_1, X_2, \dots, X_n$ . Sua formulação é expressa como:

$$y = \beta_0 + \beta_1 X_1 + \dots + \beta_n X_n + \epsilon$$

onde:

- $y$ : Variável dependente (alvo);
- $X_i$ : Variáveis independentes (preditores);
- $\beta_i$ : Coeficientes ajustados pelo modelo;
- $\epsilon$ : Termo de erro.

A simplicidade e interpretabilidade tornam a regressão linear múltipla ideal para identificar relações lineares claras, embora limitada em cenários com dados altamente não lineares [Draper 1998].

### 2.4.2 Random Forest:

Proposto por Leo Breiman em 2001, o Random Forest combina Árvores de Decisão em um modelo de ensemble para reduzir o overfitting e melhorar a precisão. A predição final é calculada por:

$$\hat{y} = \frac{1}{B} \sum_{b=1}^B h_b(x)$$

onde  $h_b(x)$  é a previsão da  $b$ -ésima árvore. É amplamente usado em bioinformática, sistemas de recomendação e classificação de imagens [Breiman 2001](#).

### 2.4.3 Gradient Boosting Regression:

O Gradient Boosting, introduzido por Jerome H. Friedman em 1999, constrói modelos sequenciais para corrigir erros dos modelos anteriores. Cada novo modelo é ajustado com base no gradiente da função de perda, como segue:

$$R_m(x) = R_{m-1}(x) - \gamma \nabla L(y, R_{m-1}(x))$$

onde:

- $R_m(x)$ : Modelo ajustado na  $m$ -ésima iteração;
- $\nabla L(y, R_{m-1}(x))$ : Gradiente da função de perda;
- $\gamma$ : Taxa de aprendizado.

Gradient Boosting é eficiente em capturar padrões complexos e é amplamente utilizado em previsões financeiras, sistemas de recomendação e detecção de fraudes [Friedman 2001](#).

### 2.4.4 Rede Neural Multicamadas (MLP):

Inspirada no modelo neuronal humano, as Redes Neurais Multicamadas (MLP) são compostas por camadas de neurônios interconectados. Cada neurônio realiza:

$$h_i = \sigma \left( \sum_{j=1}^n w_{ij} x_j + b_i \right)$$

onde:

- $\sigma$ : Função de ativação (e.g., *ReLU*, *sigmoid*);
- $w_{ij}$ : Pesos ajustados durante o treinamento;
- $b_i$ : Viés associado ao neurônio  $i$ .

Este modelo é extremamente flexível e adequado para dados complexos, mas requer grandes volumes de dados e recursos computacionais significativos. Aplicações incluem previsão de séries temporais, visão computacional e sistemas de recomendação [Rumelhart, Hinton e Williams 1986].

## 2.5 Identificação de Outliers com Intervalo Interquartilício (IQR)

O Intervalo Interquartilício (*Interquartile Range* - IQR) é uma medida estatística amplamente utilizada para identificar a dispersão central de um conjunto de dados e detectar valores discrepantes, ou *outliers*. Ele é definido como a diferença entre o terceiro quartil ( $Q3$ ) e o primeiro quartil ( $Q1$ ):

$$IQR = Q3 - Q1$$

onde:

- $Q1$  (primeiro quartil) representa o valor abaixo do qual estão 25% dos dados;
- $Q3$  (terceiro quartil) representa o valor abaixo do qual estão 75% dos dados.

O IQR é útil para resumir a variabilidade de um conjunto de dados, pois foca nos 50% centrais, ignorando influências de extremos. Ele é frequentemente utilizado em conjunto com os limites para identificação de *outliers*:

$$\text{Limite inferior} = Q1 - k \cdot IQR$$

$$\text{Limite superior} = Q3 + k \cdot IQR$$

onde  $k$  é geralmente fixado em 1.5, mas pode ser ajustado dependendo do contexto.

**Aplicações do IQR** A análise baseada no IQR tem amplo uso em diversos domínios:

- Identificação de *outliers* em dados financeiros e de saúde [Aggarwal 2017];
- Pré-processamento de dados para modelos de aprendizado de máquina, reduzindo ruído nos dados [Kuhn e Johnson 2019];
- Detecção de anomalias em séries temporais e sistemas sensíveis [Rousseeuw e Leroy 1993].

**Vantagens do IQR**

- É robusto contra valores extremos, uma vez que considera apenas os quartis  $Q1$  e  $Q3$ , ao contrário de medidas como média e desvio padrão [Aggarwal 2017].
- Facilita a detecção de padrões anômalos sem ser influenciado por grandes desvios em dados não normais [Kuhn e Johnson 2019].

**Limitações do IQR** Embora útil, o IQR pode ser insensível a variações em distribuições bimodais ou em datasets pequenos. Além disso, o  $k$  fixo (1.5) pode precisar de ajustes para diferentes domínios e contextos [Rousseeuw e Leroy 1993].

## 2.6 Validação de Modelos de Inteligência Artificial

A validação de modelos de aprendizado supervisionado é essencial para medir seu desempenho e adequação ao problema proposto. Duas métricas amplamente utilizadas na avaliação de modelos de regressão são o *Mean Squared Error* (MSE) e o coeficiente de determinação ( $R^2$ ).

### 2.6.1 Erro Quadrático Médio (MSE)

O MSE é uma métrica que mede o erro médio ao quadrado entre as previsões do modelo ( $\hat{y}_i$ ) e os valores reais ( $y_i$ ). Ele é dado por:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

onde:

- $n$ : Número de observações no conjunto de teste;
- $y_i$ : Valor real do alvo para a  $i$ -ésima observação;
- $\hat{y}_i$ : Valor previsto pelo modelo para a  $i$ -ésima observação.

Valores menores de MSE indicam que o modelo possui melhores previsões, uma vez que os erros estão mais próximos de zero. Entretanto, como o erro é medido em unidades quadráticas, interpretar diretamente seu valor pode ser menos intuitivo.

### 2.6.2 Coeficiente de Determinação ( $R^2$ )

O  $R^2$  mede a proporção da variabilidade dos dados que o modelo consegue explicar, sendo definido como:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

onde  $\bar{y}$  é a média dos valores reais ( $y_i$ ).

O  $R^2$  varia entre:

- $R^2 = 1$ : Indica que o modelo explica perfeitamente a variabilidade dos dados.

- $0 < R^2 < 1$ : O modelo explica parcialmente a variabilidade dos dados.
- $R^2 \leq 0$ : O modelo não consegue capturar a variabilidade dos dados e pode ser pior do que um modelo trivial (como prever sempre a média  $\bar{y}$ ).

### 2.6.3 Interpretação das Métricas

No contexto deste trabalho, o MSE foi usado para avaliar o erro absoluto das predições, enquanto o  $R^2$  forneceu uma visão da capacidade do modelo em capturar a variabilidade dos dados. Estas métricas foram aplicadas em todos os modelos testados, incluindo *Gradient Boosting Regression*, *Random Forest Regression*, *Regressão Linear Múltipla* e *Redes Neurais Multicamadas*. Os valores obtidos permitiram uma comparação direta entre os modelos e a escolha da abordagem mais adequada.

### 2.6.4 Limitações e Considerações

Embora úteis, essas métricas possuem limitações:

- O MSE é sensível a *outliers*, que podem inflacionar os erros quadráticos [Willmott e Matsuura 2005].
- O  $R^2$  pode ser inflado artificialmente em modelos muito complexos, como aqueles com alto número de parâmetros [Nagelkerke 1991].

**Aplicações e Contexto** Métricas como o MSE e o  $R^2$  são amplamente aplicadas em problemas de previsão e análise de dados em domínios como saúde, economia e agricultura, fornecendo uma base sólida para avaliar o desempenho de modelos [Kuhn e Johnson 2019].

## 2.7 Mínimos Quadrados Lineares

Os **mínimos quadrados lineares** são um método amplamente utilizado para resolver sistemas de equações lineares onde uma solução exata não é possível devido a inconsistências ou sistemas

sobredeterminados. O objetivo do método é minimizar a soma dos quadrados das discrepâncias entre os valores previstos pelo modelo e os valores observados [Strang 1986, Boyd e Vandenberghe 2004].

### 2.7.1 Definição Matemática

Dado um sistema de equações lineares da forma:

$$A \cdot \mathbf{x} \approx \mathbf{b}$$

onde:

- $A$ : Matriz de coeficientes ( $m \times n$ ).
- $\mathbf{x}$ : Vetor de incógnitas ( $n \times 1$ ).
- $\mathbf{b}$ : Vetor de resultados ( $m \times 1$ ).

A solução  $\mathbf{x}$  é determinada de forma a minimizar a norma euclidiana dos resíduos  $\mathbf{r}$ :

$$\mathbf{r} = \mathbf{b} - A \cdot \mathbf{x}$$

Assim, o problema de mínimos quadrados lineares pode ser escrito como:

$$\min_{\mathbf{x}} \|\mathbf{r}\|^2 = \min_{\mathbf{x}} \|\mathbf{b} - A \cdot \mathbf{x}\|^2$$

A solução exata para este problema é dada pela **equação normal**:

$$A^T \cdot A \cdot \mathbf{x} = A^T \cdot \mathbf{b}$$

ou, alternativamente, por métodos numéricos como a **decomposição em valores singulares** (*Singular Value Decomposition*, SVD), implementados em bibliotecas computacionais como NumPy.

## 2.7.2 Resíduos

Os **resíduos** ( $\mathbf{r}$ ) representam a diferença entre os valores previstos pelo modelo ( $A \cdot \mathbf{x}$ ) e os valores reais ( $\mathbf{b}$ ). Sua norma quadrática ( $\|\mathbf{r}\|^2$ ) é usada para avaliar a precisão do ajuste. Valores de resíduos altos indicam que o modelo não consegue representar adequadamente os dados observados, sendo uma métrica crucial para avaliar a qualidade da solução [Aster, Borchers e Thurber 2005](#).

## 2.7.3 Aplicação no Contexto de Fertilizantes

Neste trabalho, o método dos mínimos quadrados lineares foi aplicado para otimizar a combinação de fertilizantes que atendem às necessidades nutricionais específicas de uma cultura agrícola. O problema pode ser modelado da seguinte forma:

$$A \cdot \mathbf{x} \approx \mathbf{b}$$

onde:

- $A$ : Matriz representando os nutrientes disponíveis em cada fertilizante.
- $\mathbf{x}$ : Vetor indicando as quantidades de cada fertilizante necessárias.
- $\mathbf{b}$ : Vetor das demandas nutricionais da cultura.

## 2.7.4 Modelo Prático

Considere uma cultura que necessita de 200 g de nitrogênio, sem demanda para fósforo ou potássio. Suponha que a ureia seja o único fertilizante disponível, contendo 46% de nitrogênio e 0% de fósforo e potássio. A matriz  $A$  e o vetor  $\mathbf{b}$  são dados por:

$$A = \begin{bmatrix} 46 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 200 \end{bmatrix}$$

A solução pelo método dos mínimos quadrados lineares é obtida resolvendo:

$$\mathbf{x} = \arg \min_{\mathbf{x}} \|\mathbf{b} - A \cdot \mathbf{x}\|^2$$

Neste caso, a solução resulta em:

$$\mathbf{x} = \frac{\mathbf{b}}{A} = \frac{200}{46} \approx 4.35$$

indicando que aproximadamente 4,35 kg de ureia são necessários para atender à demanda de nitrogênio.

# Capítulo 3

## Desenvolvimento

Neste capítulo serão abordados os passos dados para obter a arquitetura do sistema proposto e a modelagem para a implementação do sistema inteligente de gerenciamento de hidroponia. Serão explicadas cada uma das fases descritas na metodologia empregada no desenvolvimento deste projeto, detalhando as atividades envolvidas em cada etapa. O desenvolvimento do sistema é baseado em sistemas apresentados em outros artigos, com a adição de tecnologias como inteligência artificial e cloud computing. O foco será na integração dessas tecnologias para otimizar o cultivo hidropônico, bem como na automação e controle dos processos.

### 3.1 O Estudo

A primeira fase de estudos feitos para o desenvolvimento deste projeto é o estudo das tecnologias já existentes. Isso diz respeito não somente ao que está relacionado à hidroponia, mas também a todas as funcionalidades a que elas se propõem. Abaixo, seguem as tabelas de Requisitos Funcionais e Requisitos Não Funcionais propostos segundo [Khaoula et al. 2021](#) e [Mehra et al. 2018](#).

<b>Requisito</b>	<b>Descrição</b>
RF01	O sistema deve monitorar continuamente os níveis de pH, temperatura, umidade, condutividade elétrica, CO <sub>2</sub> , TAN, entre outros parâmetros ambientais.
RF02	O sistema deve controlar automaticamente bombas de água e outros atuadores para manter as condições ideais para peixes e plantas.
RF03	O sistema deve permitir a visualização dos dados coletados em um dashboard intuitivo.
RF04	O sistema deve enviar alertas quando os parâmetros monitorados estiverem fora dos limites estabelecidos.
RF05	O sistema deve permitir a configuração de rotinas de ativação e desativação de dispositivos.
RF06	O sistema deve armazenar dados históricos para análise posterior.
RF07	O sistema deve integrar modelos de IA para otimização do cultivo hidropônico.
RF08	O sistema deve ser capaz de adaptar-se às necessidades específicas de diferentes tipos de cultivo hidropônico.
RF09	O sistema deve ser capaz de operar de forma autossuficiente utilizando energia solar.
RF10	O sistema deve permitir a integração com sistemas domésticos existentes, como Home Assistant e OpenHAB.

Tabela 3.1: Tabela de Requisitos Funcionais de Referência

<b>Requisito</b>	<b>Descrição</b>
RNF01	O sistema deve ser interoperável, permitindo a comunicação entre diferentes dispositivos e plataformas.
RNF02	O sistema deve ser seguro, utilizando certificados digitais x509 e TLS para comunicação segura.
RNF03	O sistema deve ser escalável, permitindo a adição de novos sensores e atuadores conforme necessário.
RNF04	O sistema deve ser de baixo custo, utilizando componentes acessíveis e tecnologias open-source.
RNF05	O sistema deve ser flexível, permitindo modificações e atualizações sem impactar o funcionamento geral.

Tabela 3.2: Requisitos Não Funcionais de Referência

Embora nem todos os requisitos identificados sejam utilizados diretamente no desenvolvimento deste projeto, principalmente ao considerar que há requisitos que não vão de encontro ao que é proposto aqui, eles servirão como base para orientar a implementação, facilitando analisar o que pode ser interessante e o que pode ser descartado, com o intuito de atingir os objetivos atuais, dessa forma, a adaptação desses requisitos permitirá a criação de um sistema ajustado à

atual proposta, alinhado com as necessidades específicas do cultivo hidropônico e do objetivos previamente descritos.

A utilização desses requisitos como referência serve para garantir que o sistema desenvolvido atenda aos padrões necessários. Por exemplo, a integração de sensores para monitoramento contínuo de parâmetros como pH, temperatura e umidade, embora não seja uma novidade, é essencial para garantir a precisão e a confiabilidade dos dados coletados. Além disso, a implementação de modelos de inteligência artificial para otimização do cultivo hidropônico pode ser aprimorada com base nas práticas recomendadas.

Outro aspecto importante é a escalabilidade do sistema. Ao considerar os requisitos de escalabilidade, podemos garantir que o sistema seja capaz de crescer e se adequar a diferentes tamanhos de operações hidropônicas, desde pequenas instalações domésticas até grandes fazendas comerciais. A interoperabilidade também é um fator crucial, permitindo que o sistema se comunique eficientemente com outros dispositivos e plataformas, facilitando a integração com tecnologias existentes e futuras.

Em resumo, ao utilizar esses requisitos como base, o projeto não apenas se beneficiará de práticas estabelecidas anteriormente, mas também será um norte para a elaboração de um sistema mais robusto, que atenderá às necessidades do usuário, aos objetivos propostos e irá facilitar upgrades ao adicionar novos módulos ou modificar os já existentes.

## 3.2 Levantamento de Requisitos do Sistema

A tabela [4.1](#) descreve os requisitos levantados para o funcionamento e operação do sistema proposto neste projeto, baseando-se em dispositivos conectados à nuvem, com funcionalidades específicas para controle e monitoramento remoto. Os requisitos abordam desde a interação física com os dispositivos até o armazenamento e gerenciamento dos dados coletados.

Inicialmente, o sistema deve permitir a interação do usuário com os dispositivos através de um botão físico para ligar e desligar, além de permitir a configuração de rede Wi-Fi por parte do device, criando uma rede local para a conexão e configurações iniciais do mesmo. Após a conexão à rede, o usuário poderá acessar um dashboard para monitoramento e controle, onde

poderá também configurar e editar os dados do dispositivo.

No que se refere ao servidor, ele deverá possuir uma interface que permita o controle remoto de dispositivos, a visualização dos dados em tempo real, o histórico de leituras e o gerenciamento dos modelos de aprendizado de máquina que serão utilizados para a automação dos estados dos atuadores do dispositivo embarcado. O sistema de IA pode gerenciar automaticamente a hidroponia, com base nas leituras dos sensores, ajustando os atuadores de acordo com o tipo de plantio e as necessidades do cultivo.

Adicionalmente, o sistema precisa garantir a segurança dos dados, obedecendo à LGPD, e ser capaz de lidar com grande volume de dispositivos e usuários simultaneamente, sem degradação do desempenho. O armazenamento e recuperação de modelos de IA devem ser eficientes e sem impacto negativo no desempenho do sistema, garantindo também a integridade dos dados armazenados e a disponibilidade do sistema. Esses requisitos têm o propósito de garantir a robustez, segurança e escalabilidade, além de possuir uma interface de fácil uso para os usuários.

Enquanto isso, a tabela [4.2](#) detalha os critérios de aceitação relacionados aos requisitos funcionais e não funcionais. Esses critérios fornecem especificações que garantem o funcionamento correto de cada requisito proposto.

### 3.3 Levantamento de Casos de Uso

Os casos de uso apresentados na Figura [?] representam as principais funcionalidades disponíveis para os usuários e componentes envolvidos no processo de monitoramento e controle automatizado. Eles estão distribuídos entre os atores principais: **Usuário**, **Dispositivo Embarcado**, **Servidor Backend**, **Servidor Frontend** e **Servidor de Inteligência Artificial (IA)**.

Cada caso de uso é projetado para atender a uma necessidade específica do sistema, seja relacionada à configuração inicial, monitoramento em tempo real, processamento de dados ou automação de tarefas baseadas em inteligência artificial. Durante o desenvolvimento e analisando as necessidades do projeto, obteve-se principalmente os seguintes casos de uso:

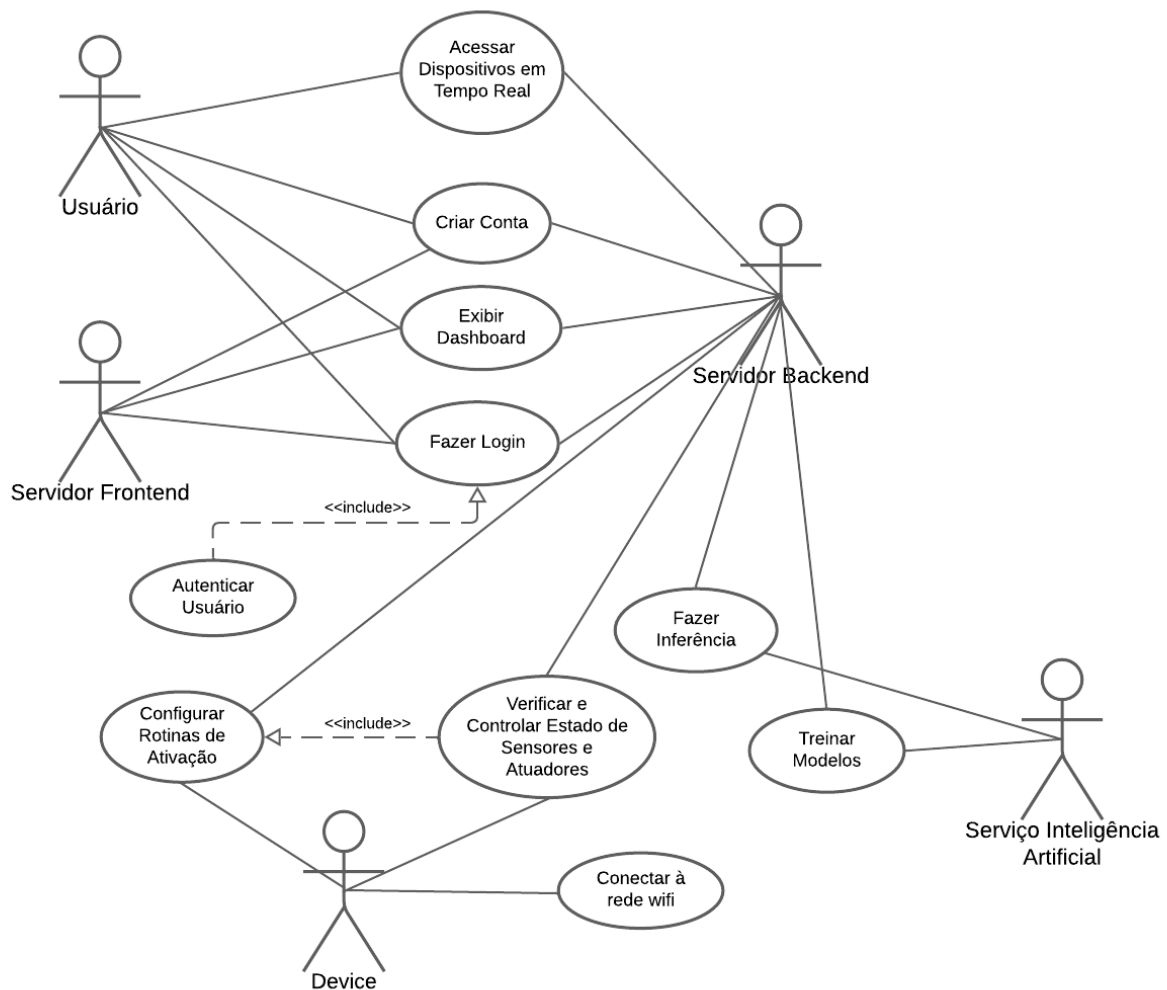


Figura 3.1: Diagrama de Casos de Uso.

### Conectar à Rede Wi-Fi

- **Ator Primário:** Dispositivo Embarcado.
- **Descrição:** O dispositivo embarcado cria uma rede local Wi-Fi, permitindo que o usuário configure sua conexão com a rede doméstica ou comercial. Esse caso de uso é essencial para a integração do dispositivo com os servidores na nuvem.
- **Fluxo Principal:**
  1. O dispositivo embarcado inicia o modo de configuração de rede.
  2. O usuário conecta ao dispositivo via interface local.

3. Os dados da rede Wi-Fi são enviados ao dispositivo, que salva as configurações e se conecta à rede.

### Login no Sistema

- **Atores Primários:** Usuário, Servidor Backend, Servidor Frontend.
- **Descrição:** Permite que o usuário autentique suas credenciais para acessar o sistema. A interação é feita pelo usuário através do frontend e a autenticação é realizada pelo backend, garantindo segurança e controle de acesso.
- **Fluxo Principal:**
  1. O usuário insere suas credenciais no Frontend.
  2. O Frontend repassa as informações ao Backend.
  3. O Backend verifica as credenciais e retorna uma resposta (sucesso ou falha).

### Criar Conta no Sistema

- **Atores Primários:** Usuário, Servidor Backend, Servidor Frontend.
- **Descrição:** Permite que o usuário registre uma nova conta no sistema, fornecendo dados como nome, e-mail e senha. O backend valida as informações e armazena os dados na base de dados.
- **Fluxo Principal:**
  1. O usuário preenche os dados no Frontend.
  2. O Frontend envia as informações ao Backend.
  3. O Backend faz a validação e cria a conta no sistema.

### Configurar Rotinas de Ativação

- **Atores Primários:** Usuário, Servidor Backend, Dispositivo Embarcado.

- **Descrição:** Permite que o usuário configure horários, condições para a ativação automática dos atuadores, como a bomba de água e o dosador de nutrientes e permite a ativação do modo automático.
- **Fluxo Principal:**
  1. O usuário define os parâmetros no Frontend (Horários de ativação e quantidade, definidos manualmente, ou modo de ativação automático, que utiliza as predições do modelo de IA).
  2. O Frontend envia os dados ao Backend.
  3. O Backend salva as configurações e as repassa ao Dispositivo Embarcado.

### Verificar e Controlar Estados dos Sensores e Atuadores

- **Atores Primários:** Usuário, Servidor Backend, Dispositivo Embarcado.
- **Descrição:** Permite que o usuário consulte e controle o estado do dispositivo embarcado em tempo real. Inclui informações como leituras de sensores e status dos atuadores.
- **Fluxo Principal:**
  1. O usuário solicita informações no Frontend.
  2. O Frontend se comunica com o Backend.
  3. O Backend obtém os dados do Dispositivo Embarcado e os retorna ao Frontend.

### Acessar Dispositivos em Tempo Real

- **Atores Primários:** Usuário, Servidor Frontend, Servidor Backend.
- **Descrição:** Permite ao usuário visualizar dados atualizados dos dispositivos conectados, como leituras de sensores e histórico de atividades.
- **Fluxo Principal:**
  1. O usuário acessa o dashboard no Frontend.

2. O Frontend solicita informações ao Backend.
3. O Backend retorna os dados solicitados.

### Treinar Modelos

- **Atores Primários:** Servidor Backend, Servidor de IA.
- **Descrição:** O backend envia dados no histórico de leituras dos sensores ao servidor de IA para treinamento de modelos de predição com o intuito de melhorar as decisões do sistema.
- **Fluxo Principal:**
  1. O Backend seleciona os dados relevantes e os envia ao Servidor de IA.
  2. O Servidor de IA treina o modelo utilizando os dados fornecidos.
  3. O modelo treinado é validado em um conjunto de teste, caso tenha um desempenho superior ao modelo anterior, o novo modelo é armazenado e indexado no Servidor de IA.

### Fazer Inferência

- **Atores Primários:** Servidor Backend, Servidor de IA, Dispositivo Embarcado.
- **Descrição:** O backend envia dados, oriundos da leitura do sensores do dispositivo embarcado, em tempo real ao servidor de IA com o intuito realizar inferências baseadas no modelo mais recente. A inferência pode resultar em ações automáticas, como ajustes na dosagem de nutrientes.
- **Fluxo Principal:**
  1. O Backend envia dados ao Servidor de IA.
  2. O Servidor de IA processa os dados e retorna uma inferência.
  3. Se necessário, o Backend aciona o Dispositivo Embarcado para realizar a ação sugerida.

### 3.4 Componentes do Dispositivo Embarcado

O dispositivo embarcado proposto neste projeto é um sistema integrado para monitoramento e controle de ambientes agrícolas, com foco em sistemas hidropônicos. Ele utiliza sensores para coletar dados do ambiente e da solução nutritiva, como temperatura, umidade, pH e condutividade elétrica (EC), além de monitorar micro e macro nutrientes. Além disso, o dispositivo é capaz de atuar diretamente no ambiente por meio de uma bomba de água e de dosadores de nutrientes, garantindo a automação do processo de cultivo. Durante o decorrer da pesquisa e analisando as necessidades do projeto, obteve-se os seguintes componentes relevantes para o desenvolvimento:

- **Microcontrolador ESP32:** Responsável por integrar e gerenciar os sensores e atuadores. O ESP32 foi escolhido devido a sua capacidade de processamento, que possibilita a execução de cálculos e controle em tempo real, e seu suporte a conectividade Wi-Fi e bluetooth nativas, essencial para comunicação com o servidor, com o intuito de desenvolver a função de monitoramento e controle. Além disso, ele possui pinos analógicos e digitais, permitindo a conexão direta com diferentes sensores e módulos. Sua eficiência energética é uma vantagem significativa para sistemas embarcados que operam continuamente.
- **Sensor de pH:** O monitoramento do pH da solução nutritiva é fundamental para garantir que as plantas possam absorver os nutrientes de maneira eficiente. Um pH fora da faixa ideal pode causar diversos problemas, como a indisponibilidade de certos nutrientes ou toxicidade, prejudicando o crescimento e a saúde do cultivo. Este sensor mede os níveis de acidez ou alcalinidade da solução, fornecendo dados que permitem ajustes automáticos na dosagem de nutrientes para manter o equilíbrio necessário. O sinal captado pelo sensor é processado por um circuito condicionador e enviado ao ESP32. De lá, os dados são transmitidos ao servidor backend, onde serão armazenados e utilizados para treino de modelos de IA ou para inferência.
- **Sensor de Condutividade Elétrica (EC):** O sensor de condutividade elétrica mede a capacidade de uma solução aquosa de conduzir corrente elétrica, proporcional à concentração

de íons dissolvidos. Essa medição é relevante para identificar a necessidade de suplementação ou ajuste na solução nutritiva. Sua integração ao sistema permite monitorar, em tempo real, deficiências ou excessos de nutrientes, considerando que diferentes plantas possuem níveis ótimos de condutividade elétrica para absorção eficiente de nutrientes.

- **Sensor de Umidade:** A umidade relativa do ar é um parâmetro essencial no cultivo hidropônico, pois está diretamente ligada à transpiração e ao equilíbrio hídrico das plantas. Este sensor monitora a umidade em tempo real, ajudando a evitar condições de estresse hídrico que podem ocorrer quando o ar está muito seco, forçando as plantas a perderem mais água do que conseguem absorver. Isso pode levar ao fechamento dos estômatos, reduzindo a fotossíntese e comprometendo o crescimento. Integrado ao ESP32, o sensor permite ajustes rápidos no ambiente, como ativação de sistemas de ventilação ou nebulização, para manter as condições ideais de cultivo, especialmente em ambientes controlados como estufas.
- **Sensor de Temperatura:** A temperatura é um fator essencial no cultivo hidropônico, tanto para o ambiente quanto para a solução nutritiva. Este sensor permite monitorar esses aspectos, garantindo que as condições estejam dentro dos níveis ideais para o crescimento das plantas. Temperaturas muito altas podem aumentar a transpiração e causar desidratação, limitar a fotossíntese pelo fechamento dos estômatos e até comprometer o funcionamento de enzimas importantes para o metabolismo. Além disso, o calor excessivo pode gerar substâncias prejudiciais às células, dificultar o transporte de nutrientes e reduzir o oxigênio disponível na solução nutritiva. Com essas informações, o sistema ajusta automaticamente as condições, garantindo um ambiente saudável para as plantas.
- **Sensor de Macro e Micronutrientes:** Este sensor é importante para monitorar os níveis de nutrientes na solução nutritiva. Nutrientes como nitrogênio, fósforo e potássio (macro) são essenciais para processos como o desenvolvimento estrutural, produção de energia e a divisão celular, enquanto micronutrientes como ferro e zinco desempenham papéis críticos em reações enzimáticas e na formação de clorofila.

A deficiência de nutrientes pode resultar em problemas como crescimento lento, amarelamento das folhas e baixa produtividade, enquanto o excesso pode causar toxicidade, interferindo na absorção de outros nutrientes e provocando desequilíbrios no desenvolvimento do cultivo. Com os dados fornecidos por este sensor, é possível ajustar a formulação da solução nutritiva, com o intuito de garantir que as plantas tenham o equilíbrio ideal de nutrientes.

- **Bomba de Água:** Este atuador movimenta a solução nutritiva pelo sistema hidropônico, garantindo que as plantas tenham acesso constante aos nutrientes. Acionada pelo ESP32 via módulo de relé, a bomba opera de maneira automatizada, ajustando o fluxo de acordo com as leituras dos sensores. Isso reduz o desperdício de água e melhora a eficiência geral do sistema.
- **Dosador de Nutrientes:** Projetado para adicionar nutrientes à solução de forma precisa, este atuador é essencial para manter os níveis corretos de nutrientes com base nas leituras fornecidas pelos sensores de pH, EC e micronutrientes. Controlado pelo ESP32, ele permite ajustes em tempo real, promovendo condições ideais de crescimento para as plantas.
- **Fonte de Alimentação:** Para garantir a operação estável e segura de todos os componentes, o dispositivo é alimentado por uma fonte bivolt (110/220V), que converte a energia para tensões de 5V e 3.3V. Esta configuração protege os circuitos sensíveis e assegura que os sensores e atuadores funcionem de maneira confiável.
- **Módulos de Relé:** Esses módulos atuam como interruptores eletrônicos que permitem ao ESP32 controlar dispositivos de alta potência, como a bomba de água e o dosador de nutrientes. Eles isolam os componentes sensíveis do sistema de possíveis picos de corrente, garantindo segurança e longevidade ao dispositivo.

### 3.5 Descrição Geral do Diagrama de Componentes

O diagrama de componentes apresenta a arquitetura modular do sistema, destacando os principais blocos funcionais e suas interações. Ele organiza o sistema em camadas e módulos, facilitando a compreensão das responsabilidades de cada parte e suas relações.

O sistema é composto por quatro módulos principais: Frontend, Backend, Dispositivo Embarcado e Servidor de Inteligência Artificial. Cada módulo desempenha um papel específico no funcionamento geral:

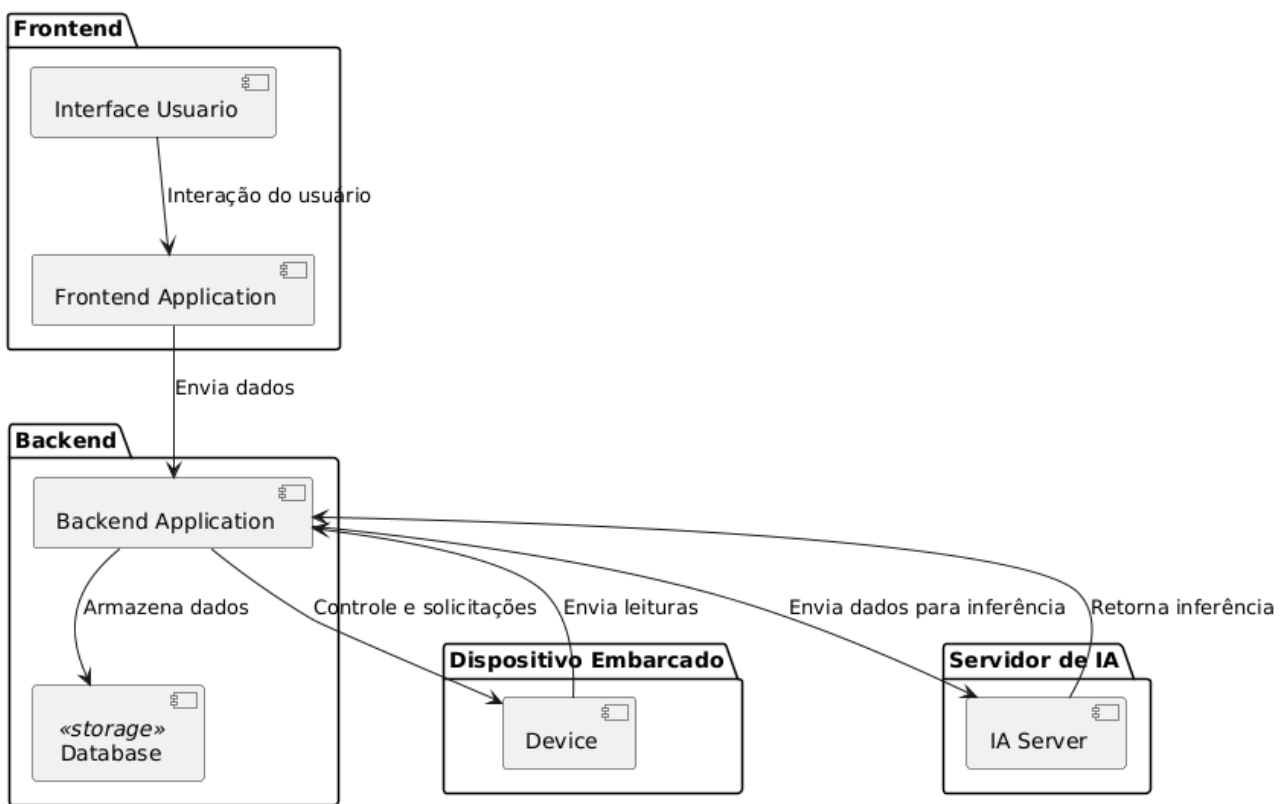


Figura 3.2: Diagrama de Componentes.

- Frontend: Interface de interação com o usuário, onde comandos e configurações são coletados e os dados recebidos do backend são exibidos em relatórios ou dashboards.
- Backend: O núcleo do sistema, responsável por processar informações do frontend e do dispositivo embarcado, gerenciar o banco de dados e comunicar-se com o servidor de IA.
- Dispositivo Embarcado: Inclui sensores e atuadores que coletam dados ambientais e exe-

cutam ações, como controle de nutrientes e circulação de água, baseados em comandos do backend.

- **Servidor de Inteligência Artificial (IA):** Realiza o treinamento de modelos e gera inferências com base nos dados enviados pelo backend, otimizando o controle do sistema.

O diagrama descreve o fluxo de dados e as interações entre os módulos, demonstrando como eles se comunicam para garantir o funcionamento ideal do sistema de controle hidropônico.

### 3.6 Diagrama de Sequência para Treinamento e Inferência do Modelo de Inteligência Artificial

O diagrama de sequência para a funcionalidade de treinamento e inferência do modelo de inteligência artificial representa uma das aplicações mais importantes do sistema, pois evidencia a capacidade do sistema de permitir ao dispositivo integrar dados do ambiente hidropônico com algoritmos de inteligência artificial. Ele descreve, de forma detalhada, como os diferentes componentes do sistema, **Frontend**, **Backend**, **Dispositivo Embarcado** e **Servidor de Inteligência Artificial (IA)** interagem.

O diagrama apresentado na Figura [?] detalha como o fluxo de informações é iniciado com a inserção de dados pelo usuário e termina com a aplicação automática de fertilizantes pelo dispositivo embarcado, apontando o papel de cada componente nesse ciclo. Essa integração entre análise de dados e atuação física é fundamental para o diferencial competitivo da proposta e a automação do sistema.

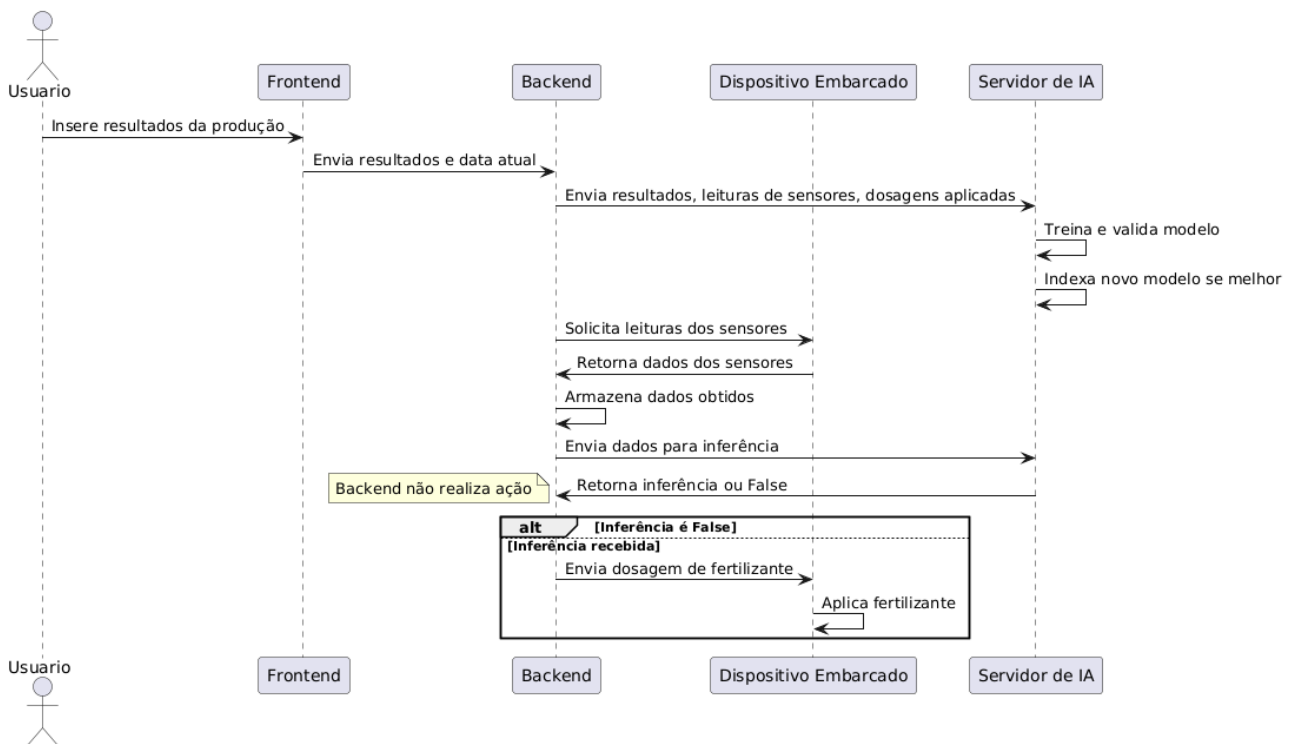


Figura 3.3: Diagrama de Sequência da Funcionalidade de Treino e Inferência.

### 1. Inserção de Resultados:

- O **Usuário** insere os resultados da produção no **Frontend**, fornecendo informações importantes para o sistema.
- O **Frontend** envia esses dados, juntamente com a data atual, para o **Backend**.

### 2. Envio de Dados ao Servidor de IA:

- O **Backend** envia os resultados da produção, as leituras de sensores e as dosagens de fertilizantes aplicadas ao longo do tempo para o **Servidor de IA**.

### 3. Treinamento e Validação de Modelos:

- O **Servidor de IA** utiliza os dados recebidos para treinar e validar um modelo de IA.
- Caso o novo modelo seja validado como superior ao anterior, ele é indexado como o modelo mais recente para futuras inferências.

#### 4. Leitura de Sensores:

- O **Backend** solicita ao **Dispositivo Embarcado** que realize leituras dos sensores ambientais e de solução.
- O **Dispositivo Embarcado** coleta os dados e os envia ao **Backend**.

#### 5. Armazenamento e Inferência:

- O **Backend** armazena os dados obtidos dos sensores e os envia ao **Servidor de IA** para realizar uma inferência.
- O **Servidor de IA** retorna uma inferência ou um valor **False**, indicando que nenhuma ação é necessária.

#### 6. Ações Automáticas:

- Se a inferência for **False**, o **Backend** não realiza nenhuma ação.
- Caso contrário, o **Backend** envia ao **Dispositivo Embarcado** a dosagem de fertilizante recomendada.
- O **Dispositivo Embarcado** aplica a dosagem indicada ao sistema hidropônico.

### 3.7 Datasets

Os dados utilizados neste projeto foram originados a partir do dataset utilizado em [Chak e Ahuja 2021]. Foi possível obter dois datasets principais, denominados `pro_crop_data.csv` e `pro_fert_data.csv`, contendo informações detalhadas sobre requisitos nutricionais das plantas e características dos fertilizantes.

Os datasets obtidos foram os seguintes:

### 3.7.1 Dataset de soluções nutritivas pro\_crop\_data.csv

O dataset de soluções nutritivas contém informações sobre as condições ideais de cultivo para diversas plantas. Ele possui as seguintes colunas:

Label	Valor
<b>crop_id</b>	Identificador único para cada planta.
<b>crop_name</b>	Nome da planta.
<b>crop_temp</b>	Temperatura ideal para o cultivo, medida em graus Celsius.
<b>crop_humidity</b>	Umidade ideal do ar, medida em porcentagem.
<b>crop_ph</b>	Faixa ideal de pH do solo.
<b>crop_nitrogen</b>	Quantidade ideal de nitrogênio, medida em partes por milhão (ppm).
<b>crop_phosphorus</b>	Quantidade ideal de fósforo, medida em partes por milhão (ppm).
<b>crop_potassium</b>	Quantidade ideal de potássio, medida em partes por milhão (ppm).
<b>crop_calcium</b>	Quantidade ideal de cálcio, medida em partes por milhão (ppm).
<b>crop_magnesium</b>	Quantidade ideal de magnésio, medida em partes por milhão (ppm).
<b>crop_sulfur</b>	Quantidade ideal de enxofre, medida em partes por milhão (ppm).
<b>crop_copper</b>	Quantidade ideal de cobre, medida em partes por milhão (ppm).
<b>crop_chlorine</b>	Quantidade ideal de cloro, medida em partes por milhão (ppm).
<b>crop_boron</b>	Quantidade ideal de boro, medida em partes por milhão (ppm).
<b>crop_iron</b>	Quantidade ideal de ferro, medida em partes por milhão (ppm).
<b>crop_zinc</b>	Quantidade ideal de zinco, medida em partes por milhão (ppm).
<b>crop_manganese</b>	Quantidade ideal de manganês, medida em partes por milhão (ppm).
<b>crop_molybdenum</b>	Quantidade ideal de molibdênio, medida em partes por milhão (ppm).
<b>crop_nickel</b>	Quantidade ideal de níquel, medida em partes por milhão (ppm).
<b>crop_cobalt</b>	Quantidade ideal de cobalto, medida em partes por milhão (ppm).
<b>crop_sodium</b>	Quantidade ideal de sódio, medida em partes por milhão (ppm).
<b>crop_ec</b>	Condutividade elétrica ideal do solo, medida em dS/m (decisiemens por metro).
<b>crop_light</b>	Intensidade de luz ideal para o cultivo, medida em lux.
<b>crop_growth_stage</b>	Estágio de crescimento da planta.
<b>crop_type</b>	Tipo da cultura, como hortaliças, grãos, etc.

Tabela 3.3: Tabela com os detalhes das culturas

A Tabela [3.3](#) apresentada detalha as principais variáveis associados ao desenvolvimento e manutenção do campo nutritivo necessário em sistemas hidropônicos. Cada entrada descreve parâmetros relevantes para o crescimento ideal do cultivo, incluindo condições ambientais, nutricionais e características gerais. Esses atributos servem como base para monitorar e ajustar

os insumos necessários.

A tabela especifica as quantidades ideais de macro e micronutrientes, como nitrogênio, fósforo, potássio, cálcio e magnésio, assim como condições ambientais, como temperatura, umidade, pH e condutividade elétrica. Além disso, outros campos abordam aspectos como intensidade de luz, estágios de crescimento e o tipo de cultura, permitindo uma adaptação precisa do manejo às características individuais de cada espécie. Combinando essas informações, a tabela serve como uma base para fundamentar decisões.

### **3.7.2 Dataset de fertilizantes `pro_fert_data.csv`**

O dataset de fertilizantes contém informações detalhadas sobre fertilizantes e suas características. Ele possui as seguintes colunas:

Label	Valor
<b>fert_id</b>	Identificador único para cada fertilizante.
<b>fert_name</b>	Nome do fertilizante.
<b>fert_nitrogen</b>	Quantidade de nitrogênio fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_phosphorus</b>	Quantidade de fósforo fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_potassium</b>	Quantidade de potássio fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_calcium</b>	Quantidade de cálcio fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_magnesium</b>	Quantidade de magnésio fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_sulfur</b>	Quantidade de enxofre fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_iron</b>	Quantidade de ferro fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_zinc</b>	Quantidade de zinco fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_boron</b>	Quantidade de boro fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_manganese</b>	Quantidade de manganês fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_copper</b>	Quantidade de cobre fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_molybdenum</b>	Quantidade de molibdênio fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_nickel</b>	Quantidade de níquel fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_cobalt</b>	Quantidade de cobalto fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_sodium</b>	Quantidade de sódio fornecida pelo fertilizante, medida em partes por milhão (ppm).
<b>fert_type</b>	Tipo do fertilizante, como orgânico ou sintético.

Tabela 3.4: Tabela com os detalhes dos fertilizantes

A Tabela 3.4 detalha as principais variáveis associadas à tabela nutricional e utilização de fertilizantes em sistemas hidropônicos. Cada entrada descreve atributos relevantes para a seleção e aplicação eficaz dos produtos, destacando propriedades físicas, químicas e nutricionais.

A tabela especifica os níveis de macro e micronutrientes presentes em cada fertilizante, como nitrogênio, fósforo, potássio, cálcio, magnésio e enxofre, além de elementos traço, como

ferro, zinco, manganês e cobre. Também são detalhados parâmetros físico-químicos, como pH, condutividade elétrica, solubilidade e potencial de toxicidade, que são importantes para avaliar a compatibilidade com a solução hidropônica e garantir uma aplicação segura.

Além disso, a tabela categoriza os fertilizantes por tipo, como reguladores de pH, fontes orgânicas ou inorgânicas, e descreve seu papel no manejo do cultivo, desde a correção de deficiências nutricionais até a melhoria da estrutura química da solução. Informações operacionais, como recomendações de uso e precauções, complementam os dados, fornecendo orientações claras e práticas para o manejo ideal.

Combinando essas informações, a tabela serve como uma base para decisões fundamentadas para a escolha e dosagem de fertilizantes em sistemas hidropônicos.

## 3.8 Dataset `pro_crop_data.csv`

### 3.8.1 Análise exploratória do dataset

Como primeiro passo da análise exploratória, foram verificadas as dimensões do dataset, resultando em uma estrutura com 4011 linhas e 25 colunas, indicando que o conjunto de dados contém 4011 registros distribuídos entre 25 variáveis.

Como segundo passo, realizou-se a verificação dos tipos de dados presentes em cada coluna do conjunto analisado. O resultado desse levantamento está apresentado na Tabela [3.5](#).

<b>Coluna</b>	<b>Não Nulos</b>	<b>Tipo de Dado</b>
<code>crop_id</code>	4011	<code>int64</code>
<code>crop_name</code>	4011	<code>object</code>
<code>crop_temp</code>	4011	<code>float64</code>
<code>crop_humidity</code>	4011	<code>float64</code>
<code>crop_ph</code>	4011	<code>float64</code>
<code>crop_nitrogen</code>	4011	<code>float64</code>
<code>crop_phosphorus</code>	4011	<code>float64</code>
<code>crop_potassium</code>	4011	<code>float64</code>
<code>crop_calcium</code>	4011	<code>float64</code>
<code>crop_magnesium</code>	4011	<code>float64</code>
<code>crop_sulfur</code>	4011	<code>float64</code>
<code>crop_copper</code>	4011	<code>float64</code>
<code>crop_chlorine</code>	4011	<code>float64</code>
<code>crop_boron</code>	4011	<code>float64</code>
<code>crop_iron</code>	4011	<code>float64</code>
<code>crop_zinc</code>	4011	<code>float64</code>
<code>crop_manganese</code>	4011	<code>float64</code>
<code>crop_molybdenum</code>	4011	<code>float64</code>
<code>crop_nickel</code>	4011	<code>float64</code>
<code>crop_cobalt</code>	4011	<code>float64</code>
<code>crop_sodium</code>	4011	<code>float64</code>
<code>crop_ec</code>	4011	<code>float64</code>
<code>crop_light</code>	4011	<code>float64</code>
<code>crop_growth_stage</code>	4011	<code>object</code>
<code>crop_type</code>	4011	<code>object</code>

Tabela 3.5: Descrição dos tipos de dados das colunas do dataset `pro_crop_data.csv`

Posteriormente, foi realizada uma análise descritiva das colunas numéricas do dataset, com o objetivo de sumarizar suas características estatísticas. A [3.6](#) apresenta as estatísticas descritivas, incluindo média, desvio padrão, valores mínimos e máximos, e quartis para cada variável numérica.

Variável	Contagem	Média	Desvio Padrão	Mínimo	25%	50% (Mediana)	Máximo
<code>crop_temp</code>	4011	25.13	1.14	20.51	24.35	25.15	29.13
<code>crop_humidity</code>	4011	350.10	10.53	312.68	343.12	349.97	390.16
<code>crop_ph</code>	4011	7.16	0.75	4.43	6.65	7.17	10.04
<code>crop_nitrogen</code>	4011	71.66	3.55	58.77	69.28	71.64	84.72
<code>crop_phosphorus</code>	4011	20.01	1.48	14.62	19.02	20.02	25.57
<code>crop_potassium</code>	4011	10.00	0.16	9.40	9.89	10.01	10.58
<code>crop_calcium</code>	4011	3.00	0.02	2.93	2.99	3.00	3.08
<code>crop_magnesium</code>	4011	0.20	0.01	0.16	0.19	0.20	0.24
<code>crop_sulfur</code>	4011	5.00	0.02	4.93	4.98	5.00	5.08
<code>crop_copper</code>	4011	0.50	0.00	0.49	0.50	0.50	0.51
<code>crop_chlorine</code>	4011	0.50	0.00	0.49	0.50	0.50	0.51
<code>crop_boron</code>	4011	0.50	0.00	0.49	0.50	0.50	0.51
<code>crop_iron</code>	4011	0.50	0.00	0.49	0.50	0.50	0.51
<code>crop_zinc</code>	4011	0.50	0.00	0.49	0.50	0.50	0.51
<code>crop_manganese</code>	4011	0.50	0.00	0.49	0.50	0.50	0.51
<code>crop_molybdenum</code>	4011	0.50	0.00	0.49	0.50	0.50	0.51
<code>crop_nickel</code>	4011	0.50	0.02	0.42	0.48	0.50	0.56
<code>crop_cobalt</code>	4011	0.50	0.11	0.09	0.42	0.50	0.95
<code>crop_sodium</code>	4011	0.71	0.29	0.00	0.51	0.73	1.50
<code>crop_ec</code>	4011	1.99	0.96	0.00	1.26	2.01	4.00
<code>crop_light</code>	4011	547.37	114.03	200.00	469.06	548.42	900.00

Tabela 3.6: Estatísticas descritivas das variáveis numéricas do dataset `pro_crop_data.csv`

Com base na Tabela 3.6, nota-se que as variáveis exibem variações significativas em suas estatísticas descritivas, o que mostra a diversidade dos dados coletados. Por exemplo, a variável `crop_temp` apresenta uma média de aproximadamente 25.13 com um desvio padrão de 1.14, indicando uma variação moderada em torno da temperatura média das amostras. Já a variável `crop_light` possui valores máximos muito superiores à média, evidenciando a presença de valores extremos.

A análise evidencia a diferença nas escalas das variáveis. Enquanto a variável `crop_ec` varia entre 0 e 4, `crop_light` apresenta valores entre 200 e 900. Essas discrepâncias requerem escalonamento ou padronização para evitar que variáveis em escalas diferentes influenciem de forma desigual os modelos. Esse ajuste assegura que todas as variáveis contribuam igualmente para o treinamento do modelo.

A etapa seguinte consiste em examinar a distribuição das variáveis numéricas, com o objetivo de compreender padrões e possíveis anomalias nos dados. A análise da distribuição das variáveis numéricas é importante para compreender a forma como os dados estão distribuídos,

identificando padrões gerais, como simetria, assimetria, picos ou possíveis anomalias. Essa etapa ajuda a determinar se transformações nos dados podem ser necessárias para melhorar a adequação aos modelos de aprendizado de máquina. A Figura [3.4](#) apresenta a visualização dessas distribuições para as variáveis numéricas.

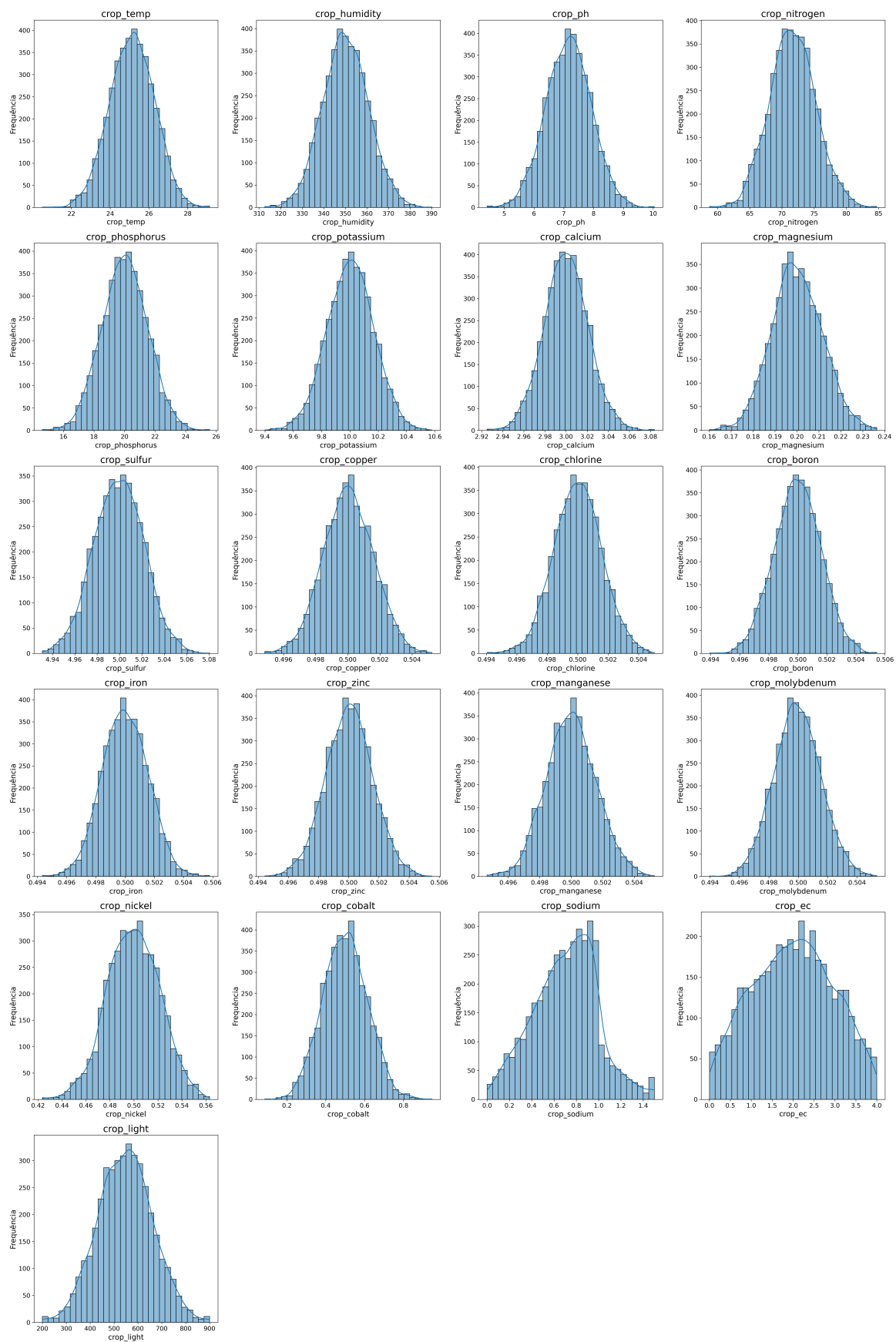


Figura 3.4: Distribuições das variáveis numéricas do dataset pro\_crop\_data.csv

Observa-se que variáveis como `crop_ec` e `crop_light` apresentam distribuições altamente assimétricas, caracterizadas por caudas longas e concentrações de valores em uma faixa específica. Essa assimetria indica a necessidade de tratamento de dados, já que valores extremos podem distorcer métricas como a média e influenciar negativamente os modelos preditivos, especialmente aqueles sensíveis a escala, como regressão linear e redes neurais.

Para abordar esses desafios, o método escolhido para o tratamento de valores extremos foi o intervalo interquartil (*Interquartile Range*, IQR). Essa técnica é eficaz para identificar e tratar outliers, baseando-se em uma análise estatística da distribuição. O IQR calcula a diferença entre o terceiro e o primeiro quartis, definindo um limite inferior e superior para detecção de outliers. Essa abordagem é preferível em relação a técnicas como transformação logarítmica, que pode não ser adequada para variáveis com valores iguais a zero, ou truncamento direto, que pode eliminar informações importantes de variáveis críticas. Ao tratar assimetrias e valores extremos, o IQR contribui para a melhoria da qualidade dos dados e para a redução de viés nos modelos.

Para aprofundar a análise exploratória, foram gerados boxplots das variáveis numéricas, permitindo uma visualização clara da dispersão dos dados, identificação de possíveis outliers e avaliação da distribuição de cada variável.

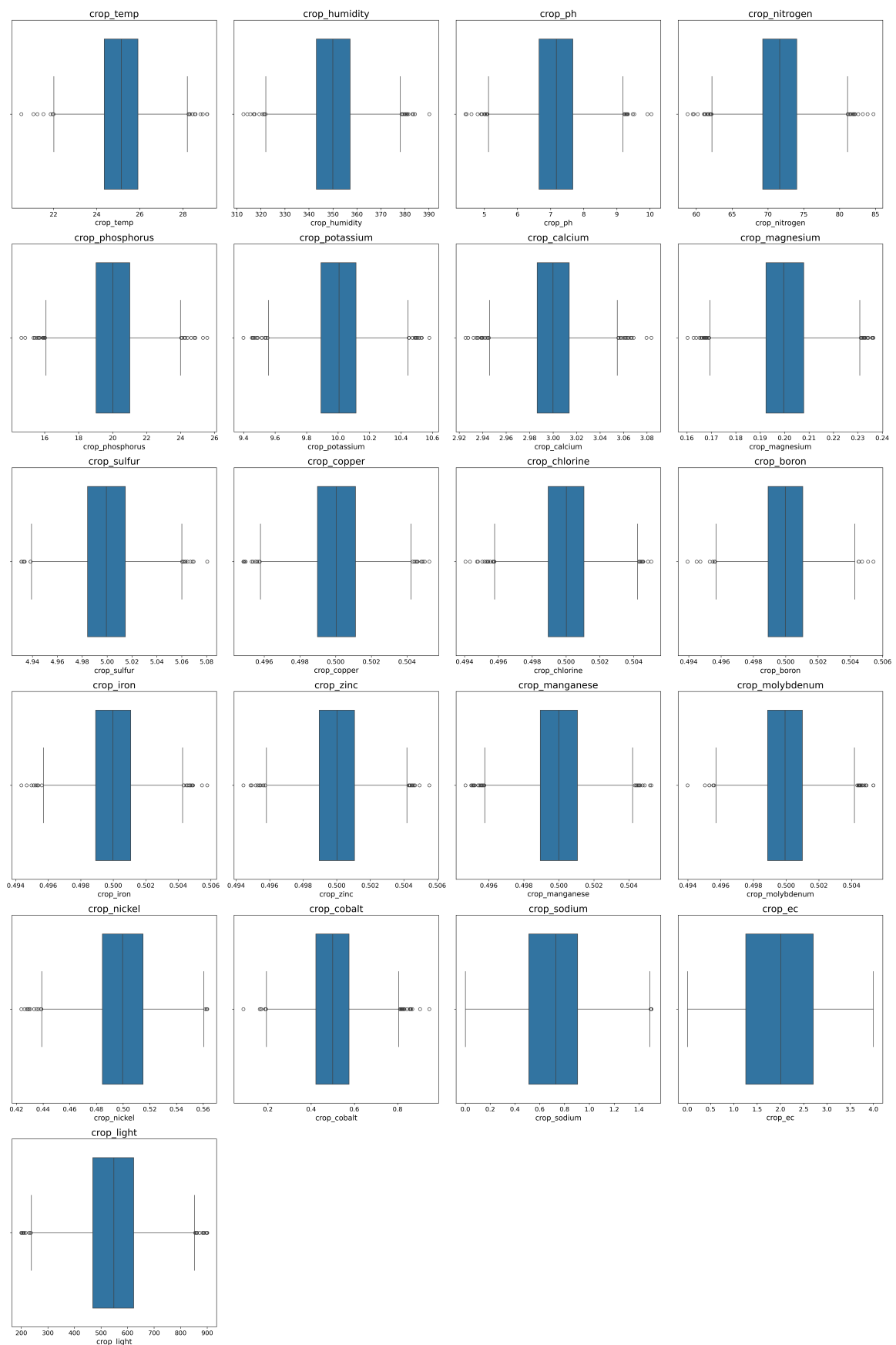


Figura 3.5: Boxplot das variáveis numéricas do dataset pro\_crop\_data.csv

A análise dos boxplots apresentados na Figura 3.5 revela padrões distintos de dispersão entre as variáveis, com algumas apresentando faixas de valores concentradas e outras indicando a presença de outliers significativos. Esses outliers são particularmente evidentes em variáveis como `crop_ec` e `crop_light`, reforçando a necessidade de estratégias de tratamento como o uso do método do intervalo interquartil (IQR) para minimizar o impacto de valores extremos nos modelos preditivos subsequentes.

Em seguida, foi realizada a análise das variáveis categóricas, com foco em compreender a distribuição das categorias presentes nas colunas `crop_growth_stage` e `crop_type`. Essa etapa é relevante para identificar possíveis desequilíbrios ou padrões que possam impactar as etapas de modelagem.

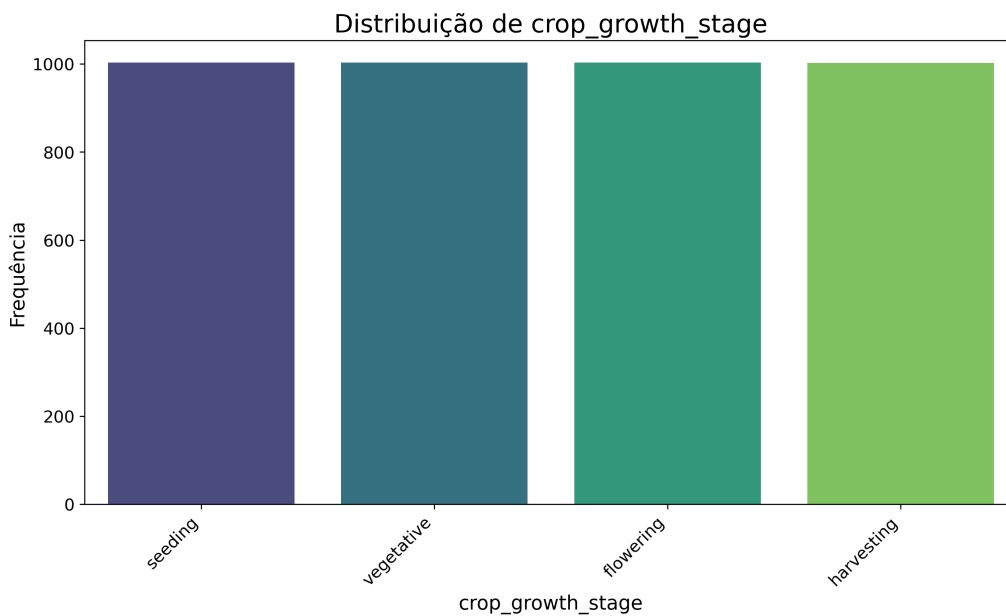


Figura 3.6: Distribuição da coluna `crop_growth_stage`

De acordo com a Figura 3.6, observa-se que a coluna `crop_growth_stage` apresenta uma distribuição predominantemente equilibrada entre as categorias, com algumas variações. Esse equilíbrio é importante para evitar enviesamento em modelos de aprendizado que considerem essa variável, minimizando a necessidade de técnicas de balanceamento.

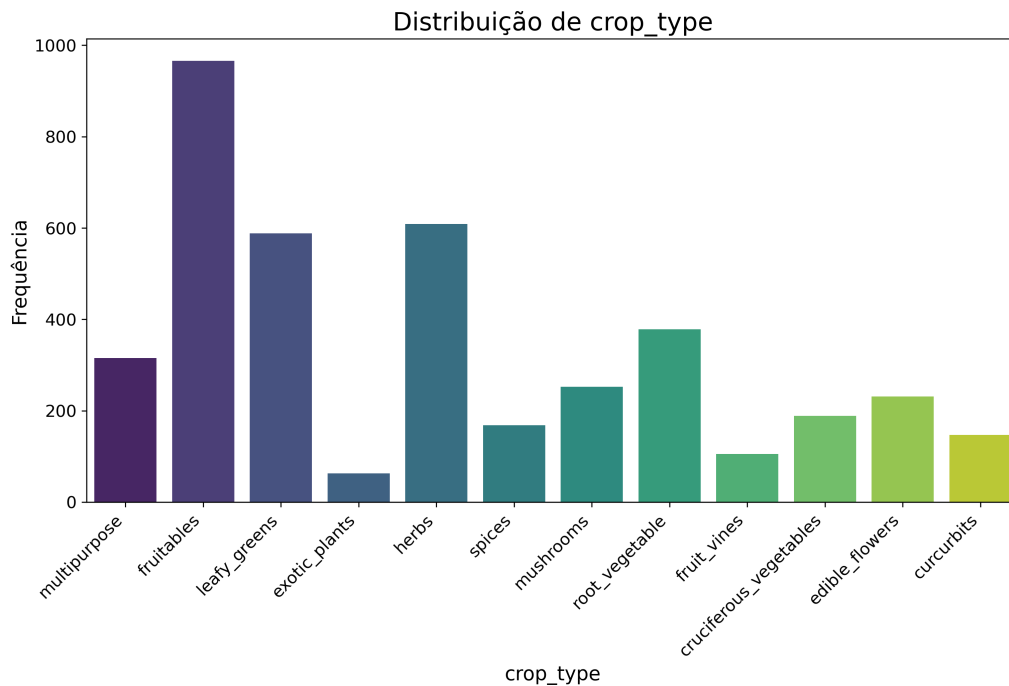


Figura 3.7: Distribuição da coluna crop\_type

A Figura 3.7 revela que a coluna `crop_type` apresenta uma concentração notável em determinadas categorias, enquanto outras possuem uma frequência significativamente menor. Esse desequilíbrio pode impactar a generalização dos modelos treinados e, se necessário, pode requerer técnicas como oversampling ou undersampling para garantir um desempenho adequado em todas as categorias.

## 3.9 Dataset pro\_fert\_data.csv

### 3.9.1 Análise exploratória do dataset

O dataset `pro_fert_data.csv` contém informações detalhadas sobre diferentes fertilizantes, incluindo suas propriedades químicas, físicas e características específicas. Este conjunto de dados é fundamental para as análises realizadas neste trabalho, permitindo modelar a relação entre os nutrientes disponíveis em fertilizantes e as necessidades nutricionais das culturas agrícolas.

Para compreender a estrutura do dataset, foram avaliadas as colunas disponíveis, verificando a completude dos dados e os tipos de cada atributo. A Tabela 3.7 apresenta a descrição das

colunas presentes no dataset, indicando o número de valores não nulos e seus respectivos tipos de dados.

<b>Coluna</b>	<b>Não Nulos</b>	<b>Tipo de Dado</b>
<code>fert_id</code>	208	<code>int64</code>
<code>fert_name</code>	208	<code>object</code>
<code>fert_type</code>	208	<code>object</code>
<code>fert_ph</code>	208	<code>float64</code>
<code>fert_ec</code>	208	<code>float64</code>
<code>fert_nitrogen</code>	208	<code>int64</code>
<code>fert_phosphorus</code>	208	<code>int64</code>
<code>fert_potassium</code>	208	<code>int64</code>
<code>fert_calcium</code>	208	<code>int64</code>
<code>fert_magnesium</code>	208	<code>int64</code>
<code>fert_sulfur</code>	208	<code>int64</code>
<code>fert_copper</code>	208	<code>int64</code>
<code>fert_chlorine</code>	208	<code>int64</code>
<code>fert_boron</code>	208	<code>int64</code>
<code>fert_iron</code>	208	<code>int64</code>
<code>fert_zinc</code>	208	<code>int64</code>
<code>fert_manganese</code>	208	<code>int64</code>
<code>fert_molybdenum</code>	208	<code>int64</code>
<code>fert_nickel</code>	208	<code>int64</code>
<code>fert_cobalt</code>	208	<code>int64</code>
<code>fert_sodium</code>	208	<code>int64</code>
<code>total</code>	208	<code>float64</code>
<code>fert_toxicity</code>	208	<code>object</code>
<code>fert_temp</code>	208	<code>int64</code>
<code>fert_solubility</code>	208	<code>object</code>
<code>fert_role</code>	208	<code>object</code>
<code>fert_remark</code>	208	<code>object</code>

Tabela 3.7: Descrição dos tipos de dados das colunas do dataset `pro_fert_data.csv`

Posteriormente, foi realizada uma análise descritiva das colunas numéricas do dataset, com o objetivo de sumarizar suas características estatísticas. A Tabela [3.8](#) apresenta as estatísticas descritivas, incluindo média, desvio padrão, valores mínimos e máximos, e quartis para cada variável numérica. Essas informações fornecem uma visão geral do comportamento dos dados, destacando variações e potenciais anomalias que podem influenciar as etapas subsequentes de modelagem e análise.

Variável	Contagem	Média	Desvio Padrão	Mínimo	25%	50% (Mediana)	Máximo
<code>fert_ph</code>	208	6.67	1.46	3.53	5.65	6.71	10.01
<code>fert_ec</code>	208	0.55	0.13	0.21	0.46	0.53	0.96
<code>fert_nitrogen</code>	208	80.38	170.47	0.00	0.00	0.00	812.00
<code>fert_phosphorus</code>	208	62.52	155.64	0.00	0.00	0.00	773.00
<code>fert_potassium</code>	208	47.26	144.52	0.00	0.00	0.00	797.00
<code>fert_calcium</code>	208	24.47	79.55	0.00	0.00	0.00	544.00
<code>fert_magnesium</code>	208	13.97	63.25	0.00	0.00	0.00	602.00
<code>fert_sulfur</code>	208	22.89	105.78	0.00	0.00	0.00	988.00
<code>fert_copper</code>	208	17.28	75.92	0.00	0.00	0.00	753.00
<code>fert_chlorine</code>	208	19.28	99.86	0.00	0.00	0.00	740.00
<code>fert_boron</code>	208	11.21	52.85	0.00	0.00	0.00	315.00
<code>fert_iron</code>	208	35.85	96.63	0.00	0.00	0.00	494.00
<code>fert_zinc</code>	208	24.08	108.36	0.00	0.00	0.00	769.00
<code>fert_manganese</code>	208	13.94	65.16	0.00	0.00	0.00	550.00
<code>fert_molybdenum</code>	208	24.38	107.03	0.00	0.00	0.00	697.00
<code>fert_nickel</code>	208	3.04	13.22	0.00	0.00	0.00	133.00
<code>fert_cobalt</code>	208	6.62	35.64	0.00	0.00	0.00	290.00
<code>fert_sodium</code>	208	7.13	42.77	0.00	0.00	0.00	442.00
<code>total</code>	208	421.53	319.76	4.49	110.45	393.42	999.85
<code>fert_temp</code>	208	42.96	7.11	25.00	38.75	45.00	50.00

Tabela 3.8: Estatísticas descritivas das variáveis numéricas do dataset `pro_fert_data.csv`

Com base na Tabela 3.8, observa-se que as variáveis apresentam uma ampla diversidade de valores, refletindo as diferenças de composição entre os fertilizantes. Por exemplo, a variável `fert_nitrogen` apresenta uma média de 80.38 com um desvio padrão elevado de 170.47, indicando uma alta variação na concentração de nitrogênio entre os fertilizantes analisados. Da mesma forma, a variável `total`, que representa a soma total de nutrientes, varia de 4.49 a 999.85, destacando a diversidade nos perfis de fertilizantes.

As disparidades nas escalas das variáveis também são evidentes. Enquanto `fert_ec` varia entre 0.21 e 0.96, `fert_nitrogen` alcança valores máximos de 812. Essas diferenças requerem atenção especial durante o pré-processamento, como escalonamento ou normalização, para garantir que os modelos matemáticos ou de aprendizado de máquina não sejam indevidamente influenciados por variáveis em escalas maiores.

A análise descritiva demonstra a importância de compreender a composição química dos fertilizantes para otimizar seu uso em aplicações agrícolas específicas. A próxima etapa envolve a análise das distribuições das variáveis numéricas, conforme ilustrado na Figura 3.8, com o objetivo de identificar padrões de comportamento, simetrias, assimetrias e valores extremos.

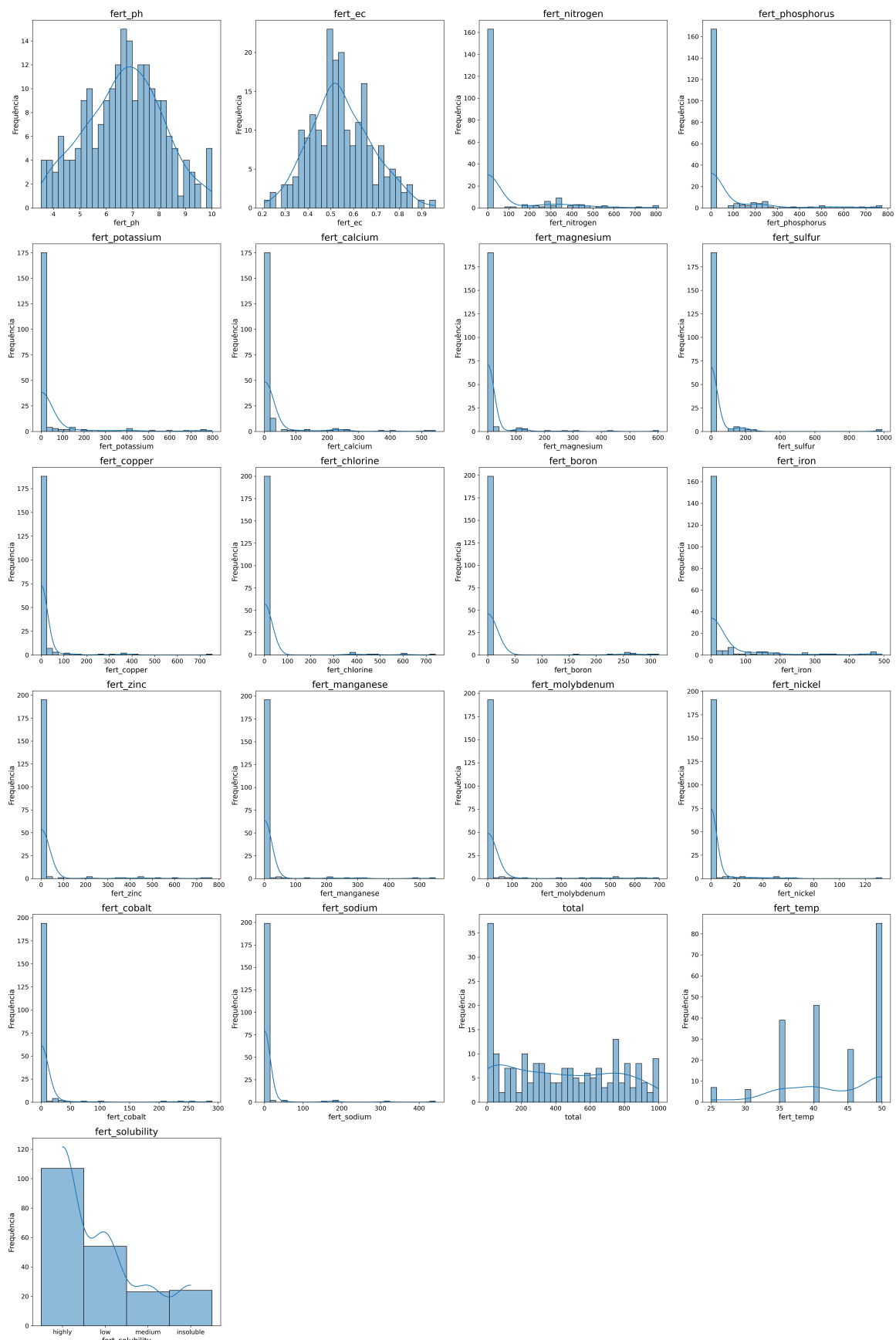


Figura 3.8: Distribuições das variáveis numéricas do dataset pro\_fert\_data.csv

No caso do dataset `pro_fert_data`, observou-se um elevado desbalanceamento nas distribuições de variáveis numéricas, com a presença de assimetrias e valores extremos em nutrientes como `fert_nitrogen`, `fert_phosphorus` e `fert_potassium`. Apesar dessas características, optou-se por não realizar um tratamento extensivo dos dados, dada a natureza específica da análise a ser conduzida.

A abordagem empregada neste trabalho baseia-se na função de cálculo de mínimos quadrados lineares (`np.linalg.lstsq`), que é robusta em relação às escalas e distribuições das variáveis. Essa escolha elimina a necessidade de pré-processamentos complexos, como normalização ou remoção de outliers, pois a função considera diretamente a relação linear entre os nutrientes fornecidos pelos fertilizantes e as demandas da cultura.

Além disso, a pipeline, que será apresentada posteriormente, não requer etapas de modelagem preditiva baseadas em aprendizado de máquina, mas sim a resolução de sistemas de equações lineares. Assim, o foco recai sobre a eficiência e a precisão da combinação de fertilizantes para atender às demandas nutricionais, reduzindo a importância de ajustes prévios aos dados.

### 3.10 Proposta de Pipeline para Recomendação de Fertilizantes

Esta proposta apresenta um pipeline baseado em inteligência artificial para determinar as condições ideais de solução nutritiva e recomendar fertilizantes específicos e suas quantidades ideais, com base nos parâmetros do cultivo e nas exigências nutricionais das plantas. O pipeline é dividido em duas etapas principais: determinação da composição ideal da solução nutritiva e recomendação de fertilizantes. A pipeline será da seguinte forma:

#### 1. Entrada de Dados:

Os dados de entrada representam informações sobre o cultivo hidropônico, oriundas da leitura de sensores, feitas pelo device, incluindo parâmetros como pH da solução, condutividade elétrica e composição atual de nutrientes.

#### 2. Modelo Previsor para Condições Ideais:

Um modelo de IA será treinado para inferir a composição ideal da solução nutritiva (concentrações de nitrogênio, fósforo, potássio, cálcio, magnésio, enxofre e micronutrientes) considerando as condições de entrada e dados históricos. O objetivo é fornecer uma recomendação para manter a solução nutritiva balanceada.

3. **Função de Recomendação de Fertilizantes:** A partir das condições ideais previstas, uma segunda função será utilizada para determinar os fertilizantes adequados. Nesta função, o fertilizante a ser aplicado (definido pelo usuário para o cultivo) e os ajustes necessários serão usados como parâmetros de entrada. O sistema emprega a função `np.linalg.lstsq` para calcular as quantidades ideais de fertilizantes a serem aplicadas, resolvendo sistemas de equações lineares que relacionam os nutrientes disponíveis com as demandas previstas. A função utiliza o dataset `pro_fert_data` como base, garantindo que as recomendações estejam alinhadas com a composição e as características dos fertilizantes disponíveis.
4. **Saída:** A saída do pipeline será uma recomendação contendo, a quantidade necessária de nutrientes a serem aplicados na solução nutritiva.

## 3.11 Regressão Linear Múltipla

A Regressão Linear Múltipla foi selecionada neste estudo devido à sua simplicidade e eficácia em capturar relações lineares entre variáveis independentes e dependentes. Essa escolha se mostrou relevante, pois a interpretabilidade dos coeficientes ajustados pelo modelo é um ponto-chave, permitindo análises mais claras e comunicáveis.

### 3.11.1 Preparação e Tratamento dos Dados

O processo começou com o carregamento do dataset por meio da biblioteca `pandas`, permitindo sua manipulação como um *DataFrame*. As variáveis foram classificadas em dois grupos: numéricas e categóricas, de modo a aplicar tratamentos específicos. As variáveis numéricas foram normalizadas utilizando o `StandardScaler`, assegurando que tivessem média zero e desvio padrão unitário. Esse passo é fundamental para evitar que variáveis de maior magnitude dominem

o modelo de Regressão Linear.

Já as variáveis categóricas foram transformadas por meio de *One-Hot Encoding*, gerando vetores binários que representam cada categoria. Uma categoria base foi excluída para evitar multicolinearidade, garantindo que o modelo permanecesse estável e os coeficientes interpretáveis.

### 3.11.2 Criação da Variável-Alvo

Uma etapa crucial foi a definição da variável-alvo, `ideal_conditions`, já que o dataset original não a possuía. Utilizou-se o método do Intervalo Interquartil (IQR), que define limites robustos para identificar condições agrícolas ideais. Valores dentro do intervalo  $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$  foram rotulados como ideais (1), enquanto os demais receberam o rótulo de não ideais (0). Essa abordagem ajudou a criar um critério objetivo e resistente a outliers para identificar padrões nos dados.

### 3.11.3 Divisão dos Dados

O dataset foi dividido em dois conjuntos: 80% dos registros foram destinados ao treinamento e 20% para o teste. Essa separação seguiu práticas padronizadas em ciência de dados, garantindo que o modelo fosse ajustado em uma amostra representativa e avaliado em dados não vistos anteriormente. Essa divisão é fundamental para evitar problemas como *overfitting*, em que o modelo se ajusta excessivamente aos dados de treinamento e perde capacidade de generalização.

### 3.11.4 Treinamento do Modelo

O treinamento do modelo consistiu em ajustar os coeficientes das variáveis independentes para minimizar o erro entre as previsões e os valores reais do conjunto de treinamento. Esse ajuste foi realizado utilizando o método dos mínimos quadrados, que otimiza os coeficientes do modelo para reduzir a soma dos erros ao quadrado.

Matematicamente, o método minimiza a função de custo, definida como a soma dos quadrados dos resíduos:

$$\min_{\beta} \|\mathbf{y} - X\beta\|^2,$$

onde  $X$  é a matriz das variáveis independentes,  $\mathbf{y}$  o vetor dos valores observados e  $\beta$  o vetor de coeficientes ajustados. A solução analítica do problema é obtida por meio das equações normais:

$$\beta = (X^T X)^{-1} X^T \mathbf{y}.$$

Esse método foi escolhido por sua eficiência e simplicidade, produzindo estimativas confiáveis dos coeficientes quando os pressupostos do modelo linear são atendidos. Durante o treinamento, a normalização das variáveis numéricas garantiu que todas contribuíssem de maneira proporcional para o ajuste do modelo. Além disso, a separação prévia entre conjunto de treinamento e teste foi crucial para evitar *overfitting*, assegurando que o modelo aprenda padrões gerais e tenha boa capacidade de generalização.

Os coeficientes ajustados fornecem uma interpretação direta da relação entre as variáveis independentes e a variável dependente, sendo um ponto-chave na análise de impacto dos fatores agrícolas sobre os resultados modelados.

### 3.11.5 Resultados

Os resultados do modelo foram avaliados com base em duas métricas principais: o Erro Quadrático Médio (MSE) e o Coeficiente de Determinação ( $R^2$ ). O MSE obteve um valor de 0.1146, indicando discrepâncias consideráveis entre os valores previstos e os reais. O  $R^2$ , por sua vez, apresentou um valor de -0.0885, sugerindo que o modelo não conseguiu capturar a variabilidade dos dados. Esse valor negativo implica que, para este conjunto de dados, a previsão da média da variável-alvo para todos os exemplos seria mais eficiente do que o modelo treinado.

Métrica	Valor
Erro Quadrático Médio (MSE)	0.1146
Coeficiente de Determinação ( $R^2$ )	-0.0885

Tabela 3.9: Resultados da Regressão Linear Múltipla

Os resultados indicam que, embora a Regressão Linear Múltipla seja uma ferramenta poderosa para identificar relações lineares, ela pode ser limitada em cenários onde as variáveis apresentam relações não lineares ou alta complexidade. O desempenho insatisfatório pode ser atribuído à variabilidade dos dados ou à presença de padrões não lineares, que não foram capturados pelo modelo.

Apesar disso, a Regressão Linear Múltipla serviu como um ponto de partida essencial para compreender o comportamento inicial dos dados e as possíveis lacunas no processo de modelagem. Esse aprendizado direcionará os ajustes necessários e a exploração de modelos mais avançados em etapas subsequentes.

## 3.12 Gradient Boosting Regressor

O Gradient Boosting Regressor foi escolhido para este trabalho devido à sua capacidade de modelar relações não lineares complexas, corrigindo iterativamente os erros de previsões anteriores. Sua abordagem baseada em árvores de decisão em sequência torna-o uma solução robusta para cenários onde os dados apresentam alta variabilidade e padrões não lineares.

### 3.12.1 Preparação e Tratamento dos Dados

Seguindo o mesmo procedimento descrito para o modelo de Regressão Linear Múltipla, o dataset foi carregado utilizando a biblioteca `pandas` e manipulado como um *DataFrame*. As variáveis foram novamente separadas em dois grupos: numéricas e categóricas. As variáveis numéricas foram processadas diretamente no modelo, enquanto as categóricas foram transformadas por meio de *One-Hot Encoding*, garantindo uma representação adequada e equitativa sem introduzir ordens artificiais entre categorias.

A variável-alvo `ideal_conditions` também foi criada utilizando o método do Intervalo Interquartil (IQR). Valores considerados dentro dos limites  $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$  foram rotulados como ideais (1), enquanto os demais registros receberam o rótulo de não ideais (0).

Para assegurar uma avaliação justa e evitar *overfitting*, os dados foram divididos em dois conjuntos: 80% dos registros para treinamento e 20% para teste. Essa divisão, alinhada ao pipeline estabelecido no modelo de Regressão Linear, possibilita uma comparação uniforme de desempenho entre os diferentes modelos.

### 3.12.2 Treinamento do Modelo

O treinamento do Gradient Boosting Regressor baseou-se na construção de árvores de decisão sequenciais, em que cada árvore adicionada ao modelo busca corrigir os erros residuais acumulados pelas árvores anteriores. Esse processo iterativo permite que o modelo se ajuste gradualmente aos padrões presentes nos dados, otimizando sua capacidade de previsão.

A função de perda, geralmente o erro quadrático médio (MSE) ou outra métrica adequada ao problema, é minimizada em cada estágio do treinamento. Cada árvore ajusta-se para reduzir os resíduos calculados a partir da previsão do modelo atual e dos valores reais, contribuindo para um modelo final mais preciso. Esse método é particularmente eficaz para capturar relações não lineares e interações complexas entre as variáveis.

### 3.12.3 Resultados

O desempenho do modelo foi avaliado utilizando as métricas de Erro Quadrático Médio (MSE) e Coeficiente de Determinação ( $R^2$ ). O MSE alcançou o valor de 0.0298, indicando um erro médio baixo nas previsões, enquanto o  $R^2$  obteve 0.717, sugerindo que o modelo explicou cerca de 71,7% da variabilidade nos dados. Portanto o Gradient Boosting Regressor foi capaz de capturar padrões relevantes nos dados, apresentando um desempenho consideravelmente superior ao modelo de Regressão Linear.

Métrica	Valor
Erro Médio Quadrático (MSE)	0.0298
Coeficiente de Determinação ( $R^2$ )	0.717

Tabela 3.10: Resultados do Gradient Boosting Regressor

Os resultados obtidos com o Gradient Boosting Regressor destacam sua eficiência na mode-

lagem de relações não lineares, demonstrando um erro médio baixo e um alto grau de explicação da variabilidade dos dados. Esse modelo é particularmente adequado para problemas em que padrões complexos e interações entre variáveis desempenham papéis críticos, fornecendo uma base sólida para análise e tomada de decisão.

### 3.13 Redes Neurais Multicamadas

As Redes Neurais Multicamadas (*Multilayer Perceptrons*, MLP) foram selecionadas devido à sua capacidade de capturar relações não lineares complexas e processar grandes volumes de dados. Essa escolha se alinha ao objetivo de modelar padrões intrincados presentes nos dados e fornecer previsões precisas. Além disso, a flexibilidade de ajuste de hiperparâmetros permite adaptar o modelo às características específicas do problema, otimizando seu desempenho.

#### 3.13.1 Preparação e Tratamento dos Dados

Seguindo o pipeline já estabelecido para modelos anteriores, os dados foram carregados utilizando a biblioteca `pandas`, convertidos em um *DataFrame* e organizados em variáveis numéricas e categóricas. As variáveis numéricas foram normalizadas utilizando o `StandardScaler`, ajustando os dados para média zero e desvio padrão unitário, enquanto as variáveis categóricas foram codificadas por meio de *One-Hot Encoding*. Essa abordagem garantiu uma representação equitativa e apropriada das categorias sem introduzir ordens artificiais.

A variável-alvo, `ideal_conditions`, foi definida usando o método do Intervalo Interquartil (IQR). Valores dentro do intervalo  $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$  foram rotulados como ideais (1), enquanto os demais foram considerados não ideais (0). Essa definição robusta contra outliers assegurou uma separação confiável entre as classes.

Para lidar com o desbalanceamento da variável-alvo, foi utilizado o método `SMOTE` (*Synthetic Minority Oversampling Technique*), que aumenta a representatividade da classe minoritária por meio de exemplos sintéticos, melhorando o desempenho do modelo na predição de ambas as classes.

### 3.13.2 Treinamento do Modelo

O treinamento do modelo envolveu a construção de Redes Neurais Multicamadas com múltiplas camadas ocultas. Essas redes utilizam funções de ativação como *ReLU* para capturar relações não lineares entre as variáveis. O ajuste dos pesos foi realizado por meio do algoritmo *backpropagation*, utilizando o otimizador Adam, que combina eficiência computacional e estabilidade no ajuste dos parâmetros.

A seleção de hiperparâmetros foi feita utilizando `GridSearchCV` com validação cruzada, explorando combinações de:

- Estruturas de camadas ocultas: [(128, 64), (128, 64, 32), (64, 32), (256, 128)].
- Funções de ativação: ReLU e Tanh.
- Otimizadores: Adam e SGD.
- Taxas de aprendizado: [0.001, 0.01, 0.0001].

Os melhores hiperparâmetros encontrados foram:

- **Estrutura de Camadas Ocultas:** (128, 64, 32).
- **Função de Ativação:** ReLU.
- **Otimizador:** Adam.
- **Taxa de Aprendizado:** 0.01.

O processo iterativo de treinamento ajustou os pesos para minimizar a função de custo, garantindo que o modelo capturasse padrões não lineares complexos nos dados.

### 3.13.3 Resultados

Os resultados foram avaliados com base em duas métricas principais: o Erro Quadrático Médio (MSE) e o Coeficiente de Determinação ( $R^2$ ). O MSE obteve o valor de 0.0287, indicando um

erro médio baixo nas predições. O  $R^2$  apresentou um valor de 0.885, mostrando que o modelo explicou aproximadamente 88,5% da variância dos dados.

Métrica	Valor
Erro Médio Quadrático (MSE)	0.0287
Coefficiente de Determinação ( $R^2$ )	0.885

Tabela 3.11: Resultados das Redes Neurais Multicamadas

Portanto, as Redes Neurais Multicamadas foram capazes de capturar padrões complexos nos dados, apresentando um desempenho superior aos modelos anteriores. O baixo MSE e o alto  $R^2$  destacam a eficácia do modelo em prever os valores ideais, tornando-o uma ferramenta interessante para análise e tomada de decisão em cenários de alta variabilidade e relações não lineares entre as variáveis.

## 3.14 Random Forest Regressor

O *Random Forest Regressor* foi escolhido para este trabalho devido à sua capacidade de capturar relações não lineares, lidar com dados heterogêneos e fornecer uma interpretação direta da importância das variáveis. A robustez desse modelo, associada à sua habilidade de reduzir *overfitting* através de ensemble learning, torna-o uma solução ideal para prever condições complexas de cultivo.

### 3.14.1 Preparação e Tratamento dos Dados

Os dados foram carregados utilizando a biblioteca `pandas` e manipulados como um *DataFrame*, permitindo uma análise inicial e separação entre variáveis numéricas e categóricas. As variáveis numéricas foram processadas diretamente no modelo após o tratamento, enquanto as categóricas foram transformadas utilizando *One-Hot Encoding*, garantindo compatibilidade com o *Random Forest Regressor* sem introduzir relações espúrias entre categorias. Para evitar multicolinearidade, foi aplicado o parâmetro `drop='first'`, removendo uma categoria base de cada variável.

A variável-alvo `ideal_conditions` foi definida a partir do método do Intervalo Interquartil (IQR), classificando registros dentro do intervalo  $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$  como ideais (1) e os demais como não ideais (0). Essa abordagem robusta a outliers assegura uma definição clara para a análise.

O conjunto de dados foi dividido em 80% para treinamento e 20% para teste, com uma semente aleatória (`random_state=42`) garantindo a reprodutibilidade do processo. Essa divisão seguiu o pipeline aplicado em modelos anteriores, assegurando uma avaliação consistente entre diferentes abordagens.

### 3.14.2 Treinamento do Modelo

O treinamento do *Random Forest Regressor* foi realizado com 100 árvores (`n_estimators=100`), utilizando configurações padrão para obter uma avaliação inicial do modelo. O algoritmo constrói múltiplas Árvores de Decisão e combina suas previsões, reduzindo a variância e melhorando a precisão geral.

Durante o treinamento, cada árvore foi ajustada com subconjuntos aleatórios de dados e variáveis, garantindo maior diversidade e robustez ao modelo. Essa abordagem ajuda a mitigar problemas comuns como *overfitting*, além de fornecer um método inerente para avaliar a importância das variáveis.

### 3.14.3 Resultados

Os resultados do modelo foram avaliados utilizando o Erro Quadrático Médio (MSE) e o Coeficiente de Determinação ( $R^2$ ). O MSE alcançou 0.0097, indicando um erro médio muito baixo nas previsões, enquanto o  $R^2$  obteve 0.908, demonstrando que o modelo explicou 90,8% da variância nos dados. Esses resultados destacam o excelente desempenho do *Random Forest Regressor* para prever condições ideais de cultivo.

Métrica	Valor
Erro Médio Quadrático (MSE)	0.0097
Coefficiente de Determinação ( $R^2$ )	0.908

Tabela 3.12: Resultados do Random Forest Regressor

O *Random Forest Regressor* demonstrou ser altamente eficaz para este problema, destacando-se por seu desempenho superior em termos de MSE e  $R^2$ . Sua capacidade de capturar interações complexas entre variáveis e fornecer uma análise clara de importância faz dele uma escolha robusta para prever condições ideais de cultivo.

## 3.15 Sistema de Recomendação de Fertilizantes

O sistema de recomendação de fertilizantes foi projetado para calcular as combinações ideais de produtos que atendam às demandas nutricionais específicas de uma cultura agrícola. Ele utiliza o método dos mínimos quadrados lineares para determinar a proporção de fertilizantes que melhor satisfazem os requisitos nutricionais, considerando os fertilizantes disponíveis. Nesta seção, detalhamos a estrutura do sistema, os principais processos e as decisões tomadas para garantir sua eficiência.

### 3.15.1 Estrutura do Sistema

O sistema baseia-se em duas funções principais que trabalham em conjunto para processar as demandas nutricionais da cultura e fornecer recomendações de fertilizantes:

- `convert_crop_to_fert`: Realiza a conversão das necessidades nutricionais da cultura para um formato compatível com os nutrientes disponíveis nos fertilizantes.
- `calculate_fertilizer_mix`: Determina a combinação ideal de fertilizantes utilizando mínimos quadrados lineares, reduzindo as discrepâncias entre as demandas da cultura e os nutrientes fornecidos.

### 3.15.2 Processos e Decisões

#### Conversão de Nutrientes

A função `convert_crop_to_fert` foi implementada para traduzir as demandas nutricionais da cultura em um formato padronizado e compatível com os dados dos fertilizantes. Este processo é essencial devido às possíveis diferenças de nomenclatura entre os dados da cultura e dos fertilizantes. Para garantir robustez, a função atribui valores padrão (zero) para nutrientes ausentes, evitando problemas de processamento e assegurando a consistência do sistema.

#### Cálculo da Mistura de Fertilizantes

A função `calculate_fertilizer_mix` representa o núcleo do sistema de recomendação. Ela utiliza os mínimos quadrados lineares para encontrar a melhor combinação de fertilizantes que atenda às demandas nutricionais. Os principais componentes deste processo são descritos a seguir:

**Construção do Vetor Alvo** O vetor alvo é criado com base nas demandas nutricionais convertidas pela função anterior. Cada posição no vetor representa a demanda por um nutriente específico, garantindo que os cálculos matemáticos sejam realizados de forma estruturada e consistente.

**Construção da Matriz de Nutrientes** A matriz de nutrientes é derivada das informações dos fertilizantes disponíveis. Cada linha representa um fertilizante, enquanto as colunas correspondem aos nutrientes disponíveis. A precisão numérica é assegurada ao converter todos os valores para o tipo `float64`, essencial para cálculos confiáveis.

**Resolução do Sistema** A resolução do sistema é realizada utilizando a função `np.linalg.lstsq`, que aplica o método dos mínimos quadrados para minimizar a diferença entre o vetor alvo e os nutrientes fornecidos pela combinação dos fertilizantes. Matematicamente, o problema pode ser representado como:

$$\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|^2,$$

onde:

- $A$ : Matriz de nutrientes (*nutrient matrix*), derivada dos fertilizantes disponíveis.
- $\mathbf{b}$ : Vetor de demandas nutricionais (*target vector*).
- $\mathbf{x}$ : Vetor solução, representando as quantidades de cada fertilizante.

A solução é obtida de forma analítica pelas **equações normais**:

$$\mathbf{x} = (A^T A)^{-1} A^T \mathbf{b},$$

onde  $A^T$  representa a transposta da matriz  $A$ .

A função `np.linalg.lstsq` implementa essa resolução de forma eficiente, utilizando decomposições numéricas robustas, como a Decomposição em Valores Singulares (*Singular Value Decomposition*, SVD), para lidar com problemas mal condicionados ou sistemas com soluções aproximadas. Essa abordagem é particularmente eficaz quando o número de nutrientes (colunas de  $A$ ) excede o número de fertilizantes (linhas de  $A$ ).

Essa metodologia assegura uma solução ótima no sentido dos mínimos quadrados, minimizando os resíduos  $\mathbf{r}$ , que são definidos como:

$$\mathbf{r} = \mathbf{b} - \mathbf{Ax}.$$

O valor dos resíduos fornece uma métrica quantitativa para avaliar a precisão da solução e sua adequação às demandas nutricionais específicas.

**Garantia de Valores Não Negativos** Os resultados obtidos, que representam as quantidades de fertilizantes recomendadas, são ajustados para valores não negativos utilizando a função `np.maximum`. Essa restrição é fundamental, uma vez que valores negativos não têm significado prático no contexto agrícola.

### 3.15.3 Resultados Esperados

Uma combinação ideal de fertilizantes e suas respectivas quantidades, que minimiza as discrepâncias entre as demandas nutricionais da cultura e os nutrientes fornecidos.

A aplicação deste sistema apresenta benefícios significativos, como redução de custos, minimização de desperdícios e atendimento otimizado às necessidades nutricionais das culturas. A precisão e a eficácia das recomendações dependem diretamente da qualidade e da completude dos dados sobre fertilizantes e culturas agrícolas disponíveis. Isso ressalta a importância de um gerenciamento adequado dos dados para alcançar resultados confiáveis e de impacto positivo.

### 3.15.4 Exemplo de Aplicação

Um exemplo prático do funcionamento do sistema foi baseado em uma cultura com alta demanda de nitrogênio, necessitando de 200 unidades, mas sem requerimentos para fósforo ou potássio. Ao utilizar a função `calculate_fertilizer_mix`, considerando a ureia como o único fertilizante disponível, o sistema determinou a quantidade necessária de 24.55 miligramas de ureia para atender à demanda de nitrogênio. O valor de 24.55 miligramas de ureia é considerado interessante pois, com uma concentração de 46% de nitrogênio (característica da ureia), essa quantidade atende de forma eficiente à demanda de 200 unidades de nitrogênio da cultura. Essa recomendação minimiza o desperdício de fertilizantes e reduz os custos, ao mesmo tempo em que assegura que as necessidades nutricionais da planta sejam supridas de maneira precisa. A pequena quantidade recomendada destaca a precisão do sistema em calcular apenas o necessário para a cultura específica, evitando excessos que poderiam levar a prejuízos econômicos.

# Capítulo 4

## Conclusão

Este trabalho apresentou o a proposta de um sistema inteligente para o gerenciamento de hidroponia, integrando tecnologias de IoT, aprendizado de máquina e automação para otimizar processos agrícolas. A proposta demonstrou ser eficaz ao abordar os principais desafios do cultivo hidropônico, como a necessidade de monitoramento contínuo, ajustes manuais e gerenciamento de dados.

Durante o desenvolvimento, foram estabelecidos requisitos para o sistema, casos de uso foram definidos para mapear as interações entre os usuários e o sistema, facilitando a identificação de funcionalidades essenciais. O diagrama de sequência para a funcionalidade de inferência e o diagrama de componentes componentes foram elaborados para modelar os processos internos e externos, permitindo uma visão do funcionamento do sistema.

Neste projeto foi dada maior ênfase à implementação de rotinas de análise de dados e modelos de inteligência artificial; dessa forma, foi possível criar uma pipeline capaz de recomendar parâmetros críticos. Os resultados obtidos com os modelos de inteligência artificial, como a Regressão Linear Múltipla, Random Forest e Redes Neurais Multicamadas, demonstraram diferentes graus de eficácia em prever e ajustar os parâmetros de cultivo. Foi possível inferir que o modelo Random Forest destacou-se pela alta capacidade de explicação da variância dos dados, enquanto o sistema de recomendação de fertilizantes mostrou-se robusto na alocação de recursos para atender às demandas nutricionais específicas das culturas.

## 4.1 Trabalhos Futuros

- **Desenvolvimento do Frontend:** Implementar uma interface gráfica para interação com o sistema e visualização dos resultados.
- **Expansão e Enriquecimento da Base de Dados:** Realizar a coleta e integração de dados adicionais, com maior variação em dados ambientais e nutricionais, para aumentar a representatividade do dataset. Visando melhorar a capacidade de generalização dos modelos.
- **Validar o Consumo de Recursos:** Realizar a validação dos recursos computacionais consumidos pelos modelos de IA, considerando o equilíbrio entre Coeficiente de Determinação e recursos utilizados.

**Parte I**

**Anexos**

Tabela 4.1: Descrição dos Requisitos do Sistema

<b>Requisito</b>	<b>Descrição</b>
RF001	O sistema deve permitir que o usuário ligue o dispositivo através de um botão físico.
RF002	O sistema deve permitir que o usuário desligue o dispositivo através de um botão físico.
RF003	O sistema deve permitir que o dispositivo crie uma rede local Wi-Fi para permitir conexões locais para configurações.
RF004	O sistema deve permitir que, após conectar-se à rede local Wi-Fi criada pelo dispositivo, o usuário possa acessar um dashboard através da conexão Wi-Fi do dispositivo.
RF005	O sistema deve permitir que o usuário conecte o dispositivo a uma rede Wi-Fi através de uma tela específica no dashboard acessível via conexão Wi-Fi local.
RF006	O sistema deve permitir que o dispositivo se comunique com um servidor na nuvem para enviar e receber dados.
RF007	O sistema deve permitir que o servidor na nuvem possua uma interface front-end que exiba um dashboard para monitoramento e configuração dos dispositivos, consulta de histórico, entrada de resultados e tomada de decisões.
RF008	O sistema deve permitir que o dashboard no servidor na nuvem liste todos os dispositivos conectados para monitoramento e gerenciamento.
RF009	O sistema deve permitir que o usuário associe cada dispositivo à sua conta no servidor na nuvem para gerenciamento e monitoramento personalizados.

Continua na próxima página

Tabela 4.1: Descrição dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Descrição</b>
RF010	O sistema deve permitir que o servidor na nuvem forneça acesso em tempo real a cada dispositivo cadastrado para monitoramento e controle.
RF011	O sistema deve permitir que o servidor na nuvem verifique e controle o estado de atividade do dispositivo, podendo ser Ativo, Offline ou Em Pausa.
RF012	O sistema deve permitir que o dashboard exiba o histórico de leituras e estados dos dispositivos para monitoramento detalhado.
RF013	O sistema deve permitir que o dispositivo faça a leitura dos dados dos sensores integrados para monitoramento e controle.
RF014	O sistema deve permitir que o dispositivo armazene dados localmente antes de enviá-los ao servidor na nuvem. Após o envio bem-sucedido, o dispositivo pode apagar os dados localmente.
RF015	O sistema deve permitir que o usuário configure rotinas de ativação do dispositivo através do dashboard, definindo horários e condições específicas.
RF016	O sistema deve permitir que o usuário apague um dispositivo registrado através do dashboard, removendo-o da lista de dispositivos associados à sua conta.
RF017	O sistema deve permitir que o usuário edite os dados do dispositivo através do dashboard, incluindo informações como nome, localização e outras configurações relevantes.

Continua na próxima página

Tabela 4.1: Descrição dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Descrição</b>
RF018	O sistema deve permitir que o servidor na nuvem armazene os dados dos atuadores dos dispositivos para monitoramento e análise.
RF019	O sistema deve permitir que o servidor na nuvem armazene os dados dos sensores dos dispositivos para monitoramento e análise.
RF020	O sistema deve permitir que o servidor na nuvem treine modelos de aprendizado de máquina utilizando os dados de entrada fornecidos pelos usuários, incluindo resultados das colheitas, leituras de sensores e estados dos atuadores.
RF021	O sistema deve possuir o modo automático para gerenciamento de hidroponia, permitindo que o usuário ative o modo automático para cada dispositivo, considerando o tipo de plantio e as leituras dos sensores como entrada para um modelo de IA que definirá os estados dos atuadores, com o objetivo de gerenciar automaticamente o sistema de hidroponia.
RF022	O sistema deve permitir que o servidor de inteligência artificial armazene e gereencie os modelos de IA treinados, garantindo que eles estejam disponíveis para uso e atualização conforme necessário.
RF023	O sistema deve permitir que novos usuários criem uma conta para acessar e utilizar as funcionalidades disponíveis.
RF024	O sistema deve permitir que os usuários deletem suas contas, removendo todos os dados associados.
RF025	O sistema deve permitir que os usuários façam login utilizando suas credenciais para acessar suas contas e funcionalidades disponíveis.

Continua na próxima página

Tabela 4.1: Descrição dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Descrição</b>
RF026	O sistema deve permitir que os usuários façam logout de suas contas, encerrando a sessão atual de forma segura.
RNF001	O sistema deve obedecer à LGPD, garantindo a proteção dos dados pessoais dos usuários, inclusive na coleta de dados para treinamento dos modelos de IA. Todos os dados de sensores de todos os usuários devem ser tratados de acordo com a legislação vigente.
RNF002	O sistema deve implementar medidas de segurança para proteger os dados dos usuários contra acesso não autorizado, perda ou corrupção.
RNF003	O sistema deve ser capaz de lidar com um grande número de dispositivos e usuários simultaneamente, sem degradação significativa de desempenho.
RNF004	O sistema deve ser intuitivo e fácil de usar para todos os tipos de usuários, com uma interface amigável e bem documentada.
RNF005	O sistema deve ser altamente disponível e confiável, minimizando o tempo de inatividade e garantindo a integridade dos dados.
RNF006	O sistema deve garantir que o armazenamento e a recuperação dos modelos de IA sejam realizados de forma eficiente, sem impactar negativamente o desempenho do sistema.
RNF007	O sistema deve ser capaz de gerenciar um grande número de modelos de IA sem degradação de desempenho.

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF001	<ol style="list-style-type: none"> <li>1. O dispositivo deve ligar ao pressionar o botão.</li> <li>2. O dispositivo deve indicar visualmente que está ligado.</li> </ol>
RF002	<ol style="list-style-type: none"> <li>1. O dispositivo deve desligar ao pressionar o botão.</li> <li>2. O dispositivo deve indicar visualmente que está desligado.</li> </ol>
RF003	<ol style="list-style-type: none"> <li>1. O dispositivo deve criar uma rede Wi-Fi local ao ser ativado.</li> <li>2. A rede Wi-Fi deve ser detectável por outros dispositivos dentro do alcance.</li> <li>3. Deve ser possível conectar-se à rede Wi-Fi do device e acessar a interface de configuração do dispositivo.</li> </ol>
RF004	<ol style="list-style-type: none"> <li>1. O dispositivo deve exibir um dashboard acessível via navegador web ao conectar-se à rede Wi-Fi local.</li> <li>2. O dashboard deve permitir a configuração e monitoramento do dispositivo.</li> <li>3. O acesso ao dashboard deve ser seguro e protegido por autenticação.</li> </ol>
RF005	<ol style="list-style-type: none"> <li>1. O dashboard deve incluir uma tela para configuração de rede Wi-Fi.</li> <li>2. O usuário deve poder selecionar e conectar-se a redes Wi-Fi disponíveis.</li> <li>3. O dispositivo deve confirmar a conexão bem-sucedida à rede Wi-Fi selecionada.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF006	<ol style="list-style-type: none"> <li>1. O dispositivo deve ser capaz de estabelecer uma conexão segura com o servidor na nuvem.</li> <li>2. O dispositivo deve enviar dados ao servidor na nuvem conforme especificado.</li> <li>3. O dispositivo deve receber e processar dados do servidor na nuvem corretamente.</li> </ol>
RF007	<ol style="list-style-type: none"> <li>1. O servidor na nuvem deve hospedar um dashboard acessível via navegador web.</li> <li>2. O dashboard deve exibir dados recebidos dos dispositivos em tempo real.</li> <li>3. O dashboard deve permitir a configuração remota dos dispositivos.</li> <li>4. O dashboard deve permitir a consulta do histórico de leituras, resultados e previsões do modelo de inteligência artificial.</li> <li>5. O dashboard deve permitir a entrada de dados manuais relacionados aos resultados obtidos referente à colheita.</li> </ol>
RF008	<ol style="list-style-type: none"> <li>1. O dashboard deve exibir uma lista de todos os dispositivos atualmente conectados.</li> <li>2. A lista deve incluir informações relevantes de cada dispositivo, como status de conexão e identificadores únicos.</li> <li>3. O usuário deve poder interagir com cada dispositivo listado para obter mais detalhes ou realizar configurações.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF009	<ol style="list-style-type: none"> <li>1. O usuário deve poder autenticar-se no servidor na nuvem.</li> <li>2. O usuário deve poder adicionar dispositivos à sua conta através do dashboard utilizando um token único associado a cada dispositivo.</li> <li>3. O sistema deve confirmar a associação bem-sucedida do dispositivo à conta do usuário.</li> </ol>
RF010	<ol style="list-style-type: none"> <li>1. O servidor deve manter uma conexão ativa com todos os dispositivos cadastrados.</li> <li>2. O usuário deve poder acessar dados em tempo real de cada dispositivo através do dashboard.</li> <li>3. O usuário deve poder enviar comandos em tempo real para os dispositivos através do dashboard.</li> </ol>
RF011	<ol style="list-style-type: none"> <li>1. O servidor deve exibir o estado atual de cada dispositivo (Ativo, Offline, Em Pausa) no dashboard.</li> <li>2. O usuário deve poder alterar o estado do dispositivo através do dashboard.</li> <li>3. O dispositivo deve responder às mudanças de estado enviadas pelo servidor.</li> </ol>
RF012	<ol style="list-style-type: none"> <li>1. O dashboard deve exibir um histórico detalhado de todas as leituras de dados dos dispositivos.</li> <li>2. O dashboard deve exibir um histórico dos estados de atividade dos dispositivos (Ativo, Offline, Em Pausa).</li> <li>3. O usuário deve poder filtrar e visualizar o histórico por período de tempo e tipo de dado.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF013	1. O sistema deve permitir que o dispositivo faça a leitura dos dados dos sensores integrados para monitoramento e controle.
RF014	1. O dispositivo deve armazenar dados localmente até que uma conexão com o servidor na nuvem esteja disponível. 2. O dispositivo deve enviar os dados armazenados ao servidor na nuvem assim que possível. 3. Após a confirmação do envio bem-sucedido, o dispositivo deve apagar os dados localmente para liberar espaço.
RF015	1. O dashboard deve incluir uma interface para configuração de rotinas de ativação. 2. O usuário deve poder definir horários específicos para ativação do dispositivo. 3. O usuário deve poder definir condições adicionais para a ativação do dispositivo (por exemplo, dias da semana, eventos específicos). 4. O dispositivo deve ativar automaticamente conforme as rotinas configuradas.
RF016	1. O dashboard deve incluir uma opção para apagar dispositivos registrados. 2. O usuário deve poder selecionar e confirmar a remoção de um dispositivo. 3. O dispositivo deve ser removido da lista de dispositivos associados à conta do usuário após a confirmação.

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF017	<ol style="list-style-type: none"> <li>1. O dashboard deve incluir uma interface para edição dos dados do dispositivo.</li> <li>2. O usuário deve poder modificar e salvar as alterações nos dados do dispositivo.</li> <li>3. As alterações devem ser refletidas imediatamente no sistema e no dispositivo.</li> </ol>
RF018	<ol style="list-style-type: none"> <li>1. O servidor deve receber e armazenar dados dos atuadores enviados pelos dispositivos.</li> <li>2. Os dados dos atuadores devem ser armazenados de forma segura e organizada.</li> <li>3. O usuário deve poder acessar e visualizar os dados dos atuadores através do dashboard.</li> </ol>
RF019	<ol style="list-style-type: none"> <li>1. O servidor deve receber e armazenar dados dos sensores enviados pelos dispositivos.</li> <li>2. Os dados dos sensores devem ser armazenados de forma segura e organizada.</li> <li>3. O usuário deve poder acessar e visualizar os dados dos sensores através do dashboard.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF020	<ol style="list-style-type: none"> <li>1. O servidor deve coletar e armazenar dados de entrada dos usuários, resultados das colheitas, leituras de sensores e estados dos atuadores.</li> <li>2. O servidor deve utilizar esses dados para treinar modelos de aprendizado de máquina.</li> <li>3. Os modelos treinados devem ser avaliados e validados para garantir precisão e relevância.</li> <li>4. O usuário deve poder visualizar os resultados e insights gerados pelos modelos treinados através do dashboard.</li> </ol>
RF021	<ol style="list-style-type: none"> <li>1. O dashboard deve incluir uma opção para ativar o modo automático para cada dispositivo.</li> <li>2. O usuário deve poder selecionar o tipo de plantio sendo feito.</li> <li>3. O sistema deve utilizar as leituras dos sensores como entrada para um modelo de IA.</li> <li>4. O modelo de IA deve definir os estados dos atuadores com base nas leituras dos sensores e no tipo de plantio.</li> <li>5. O sistema de hidroponia deve ser capaz de se gerenciar automaticamente utilizando IA.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF022	<ol style="list-style-type: none"> <li>1. O sistema deve ser capaz de armazenar modelos de IA treinados no servidor de inteligência artificial.</li> <li>2. O sistema deve registrar metadados associados a cada modelo de IA, incluindo data de treinamento, versão e parâmetros utilizados.</li> <li>3. Os modelos de IA armazenados devem estar disponíveis para uso em inferências e outras operações.</li> <li>4. O sistema deve permitir a atualização dos modelos de IA com novos dados de treinamento, mantendo um histórico de versões.</li> <li>5. O sistema deve gerenciar versões dos modelos de IA, permitindo acesso a versões anteriores quando necessário.</li> <li>6. O sistema deve notificar o administrador em caso de falha ao armazenar ou atualizar um modelo de IA.</li> <li>7. O acesso aos modelos de IA deve ser restrito a usuários autorizados.</li> </ol>
RF023	<ol style="list-style-type: none"> <li>1. O sistema deve fornecer uma interface para criação de conta.</li> <li>2. O usuário deve poder inserir informações necessárias, como nome, e-mail e senha.</li> <li>3. O sistema deve validar as informações fornecidas e criar a conta do usuário.</li> <li>4. O usuário deve receber uma confirmação de criação de conta bem-sucedida.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RF024	<ol style="list-style-type: none"> <li>1. O sistema deve fornecer uma interface para deleção de conta.</li> <li>2. O usuário deve poder confirmar a deleção da conta.</li> <li>3. O sistema deve remover todos os dados associados à conta do usuário após a confirmação.</li> <li>4. O usuário deve receber uma confirmação de deleção de conta bem-sucedida.</li> </ol>
RF025	<ol style="list-style-type: none"> <li>1. O sistema deve fornecer uma interface para login.</li> <li>2. O usuário deve poder inserir seu e-mail e senha.</li> <li>3. O sistema deve validar as credenciais fornecidas.</li> <li>4. O usuário deve ser autenticado e redirecionado para o dashboard após um login bem-sucedido.</li> </ol>
RF026	<ol style="list-style-type: none"> <li>1. O sistema deve fornecer uma opção para logout no dashboard.</li> <li>2. O usuário deve poder confirmar a ação de logout.</li> <li>3. O sistema deve encerrar a sessão do usuário e redirecioná-lo para a página de login.</li> </ol>
RNF001	<ol style="list-style-type: none"> <li>1. O sistema deve obter consentimento explícito dos usuários para a coleta e uso de seus dados.</li> <li>2. Os dados pessoais devem ser armazenados e processados de forma segura.</li> <li>3. O sistema deve permitir que os usuários acessem, corrijam e excluam seus dados pessoais.</li> <li>4. O uso de dados para treinamento de modelos de IA deve ser transparente e conforme a LGPD.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RNF002	<ol style="list-style-type: none"> <li>1. Os dados devem ser criptografados durante a transmissão e armazenamento.</li> <li>2. O sistema deve incluir autenticação de dois fatores para acesso ao dashboard.</li> <li>3. Logs de acesso devem ser mantidos para auditoria.</li> </ol>
RNF003	<ol style="list-style-type: none"> <li>1. O sistema deve suportar múltiplos dispositivos conectados simultaneamente.</li> <li>2. O dashboard deve ser responsivo às requisições de forma eficiente.</li> <li>3. O sistema deve ser capaz de escalar horizontalmente para acomodar mais usuários e dispositivos.</li> </ol>
RNF004	<ol style="list-style-type: none"> <li>1. O dashboard deve passar por testes de usabilidade com uma pontuação aceitável de satisfação dos usuários.</li> <li>2. A documentação do usuário deve estar disponível e ser compreensível.</li> <li>3. O sistema deve incluir tutoriais e ajuda contextual.</li> </ol>
RNF005	<ol style="list-style-type: none"> <li>1. O sistema deve ter uma disponibilidade alta.</li> <li>2. O sistema deve incluir mecanismos de backup e recuperação de dados.</li> <li>3. O sistema deve ser capaz de detectar e recuperar-se automaticamente de falhas.</li> </ol>

Continua na próxima página

Tabela 4.2: Critérios de Aceitação dos Requisitos do Sistema (Continuação)

<b>Requisito</b>	<b>Critério de Aceitação</b>
RNF006	<ol style="list-style-type: none"> <li>1. O armazenamento de um modelo de IA deve ser realizado de forma eficiente.</li> <li>2. A recuperação de um modelo de IA para uso em inferências deve ser rápida e eficiente.</li> <li>3. O desempenho geral do sistema não deve ser impactado negativamente durante operações de armazenamento e recuperação de modelos de IA.</li> <li>4. O sistema deve manter uma latência de resposta aceitável durante operações normais.</li> <li>5. O sistema deve incluir mecanismos de monitoramento para detectar e alertar sobre possíveis degradações de desempenho relacionadas ao armazenamento e recuperação de modelos de IA.</li> </ol>
RNF007	<ol style="list-style-type: none"> <li>1. O sistema deve ser capaz de gerenciar modelos de IA sem degradação perceptível de desempenho.</li> <li>2. O sistema deve suportar a adição de novos modelos de IA sem necessidade de reconfiguração manual significativa.</li> <li>3. O acesso direto aos modelos de IA deve ser controlado e restrito a usuários autorizados, com logs de acesso mantidos para auditoria.</li> </ol>

# Referências Bibliográficas

AGGARWAL, C. C. *An Introduction to Outlier Analysis*. [S.l.]: Springer, 2017.

AL-ALI, A. R. et al. Zigbee-based irrigation system for home gardens. In: *Proceedings of International Conference on Communications, Signal Processing, and Their Applications (ICCSPA)*. [S.l.: s.n.], 2015.

ANAND, K. et al. Automatic drip irrigation system using fuzzy logic and mobile technology. In: *IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR)*. [S.l.: s.n.], 2015.

ARVIND, G. et al. Automated irrigation with advanced seed germination and pest control. In: *IEEE Technological Innovations in ICT for Agriculture and Rural Development*. [S.l.: s.n.], 2017. p. 8273687.

ASTER, R.; BORCHERS, B.; THURBER, C. *Parameter Estimation and Inverse Problems*. [S.l.]: Academic Press, 2005.

AYED, R. B.; HANANA, M. Artificial intelligence to improve the food and agriculture sector. *Journal of Food Quality*, Wiley Online Library, v. 2021, n. 1, p. 5584754, 2021.

BALLEDA, K. et al. Agpest: An efficient rule-based expert system to prevent pest diseases of rice wheat crops. *Proceedings of 8th International Conference on Intelligent Systems and Control*, p. 10–11, 2014.

BOYD, S.; VANDENBERGHE, L. *Convex Optimization*. [S.l.]: Cambridge University Press, 2004.

BREIMAN, L. Random forests. *Machine Learning*, 2001.

CHAK, J.; AHUJA, K. Smart hydroponic system. Mahakal Institute of Technology, Ujjain, 2021.

CHOWDHURY, M. E. H. et al. Design, construction and testing of iot based automated indoor vertical hydroponics farming test-bed in qatar. *Sensors*, v. 20, n. 19, 2020. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/20/19/5637>.

CRUZ, J. R. D. et al. Water usage optimization of smart farm automated irrigation system using artificial neural network. In: *Proceedings of 5th International Conference on Information and Communication Technology (ICoICT)*. [S.l.: s.n.], 2017.

- DAUD, M.; HANDIKA, V.; BINTORO, A. Design and realization of fuzzy logic control for ebb and flow hydroponic system. *International Journal of Scientific and Technology Research*, v. 7, n. 9, p. 138–144, 2018.
- DEBAUCHE, O. et al. Edge ai-iot pivot irrigation, plant diseases, and pests identification. *Procedia Computer Science*, v. 177, p. 40–48, 2020.
- DRAPER, N. *Applied regression analysis*. [S.l.]: McGraw-Hill. Inc, 1998.
- ECOCENTER. *Sistemas Hidropônicos: Principais Tipos e Categorias*. 2023. Acessado em 30 de novembro de 2024. Disponível em: <https://loja.ecocenter.pt/post/sistemas-hidroponicos-principais-tipos-e-categorias>.
- EL-HENDAWY, S. E. et al. Comparative performance of spectral reflectance indices and multivariate modeling for assessing agronomic parameters in advanced spring wheat lines under two contrasting irrigation regimes. *Frontiers in Plant Science*, v. 10, p. 1537, 2019.
- ELSHORBAGY, A.; PARASURAMAN, K. Soil moisture estimation using honn models. *Journal of Environmental Modelling*, v. 22, p. 270–278, 2008.
- FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, JSTOR, p. 1189–1232, 2001.
- GHORBANI, M. A. et al. Artificial intelligence-based fast and efficient hybrid approach for spatial modelling of soil electrical conductivity. *Soil Tillage Res.*, v. 186, p. 152–164, 2019.
- GHORBEL, R. et al. Hydroponic technology for green fodder production: Concept, advantages, and limits. *6th International Students Science Congress*, 09 2022.
- HANSON, B.; OTHERS. Ai-based soil moisture analyzer for precise temperature control. *Agricultural Systems*, v. 94, p. 103–112, 2007.
- HARAHAP, M. A.; HARAHAP, F.; GULTOM, T. The effect of ab mix nutrient on growth and yield of pak choi (*brassica chinensis* l.) plants under hydroponic wick system condition. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2020. v. 1485, n. 1, p. 012028.
- HASSANIJALILIAN, O. et al. Rating iron deficiency in soybean using image processing and decision-tree based models. *Remote Sens.*, v. 12, n. 24, p. 1224–4143, 2020.
- HELMY et al. Nutrient film technique (nft) hydroponic monitoring system based on wireless sensor network. In: *2017 IEEE International Conference on Communication, Networks and Satellite (Comnetsat)*. [S.l.: s.n.], 2017. p. 81–84.
- HOSTALRICH, D. et al. Intelligent management of hydroponic systems based on iot for agrifood processes. *Journal of Sensors*, Wiley Online Library, v. 2022, n. 1, p. 9247965, 2022.
- ISMAIL, M.; MUSTIKASARI. Intelligent system for tea leaf disease detection. *IPSJ Technical Report*, p. 1–4, 2013.

KAMILARIS, A.; KARTAKOULLIS, A.; PRENAFETA-BOLDÚ, F. X. A review on the practice of big data analysis in agriculture. *Computers and Electronics in Agriculture*, Elsevier, v. 143, p. 23–37, 2017.

KHAOULA, T. et al. Architecture design of monitoring and controlling of iot-based aquaponics system powered by solar energy. *Procedia Computer Science*, v. 191, p. 493–498, 2021. ISSN 1877-0509. The 18th International Conference on Mobile Systems and Pervasive Computing (MobiSPC), The 16th International Conference on Future Networks and Communications (FNC), The 11th International Conference on Sustainable Energy Information Technology. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877050921014630>.

KOLHE, S. et al. A web-based intelligent disease-diagnosis system. *Expert Syst Appl.*, v. 38, n. 12, p. 14592–14601, 2011.

KOUADIO, L. et al. Artificial intelligence approach for the prediction of robusta coffee yield using soil fertility properties. *Comput. Electron Agric.*, v. 155, p. 324–338, 2018.

KUHN, M.; JOHNSON, K. *Applied Predictive Modeling*. [S.l.]: Springer, 2019.

KUMAR, M.; MAURYA, P.; VERMA, R. Future of indian agriculture using ai and machine learning tools and techniques. *The New Advanced Society: Artificial Intelligence and Industrial Internet of Things Paradigm*, v. 1, p. 447–472, 2022.

LOPEZ, E. M. et al. A fuzzy expert system for soil characterization. *Environmental International*, v. 34, p. 950–958, 2008.

LUKITO, D.; BASKORO, R.; ERNAWATI, E. Development of water quality control and monitoring system for hydroponic plants with a cloud-based system. *Journal of Computer Science*, Science Publications, v. 20, n. 4, p. 658–669, 2024. Disponível em: <https://thescipub.com/pdf/jcssp.2024.658.669.pdf>.

MEHRA, M. et al. Iot based hydroponics system using deep neural networks. *Computers and electronics in agriculture*, Elsevier, v. 155, p. 473–486, 2018.

MUSTAFA, M. R. U. et al. Svm-based geospatial prediction of soil erosion under static and dynamic conditioning factors. *MATEC Web of Conferences*, v. 203, p. 1–11, 2004.

NAGELKERKE, N. J. D. A note on a general definition of the coefficient of determination. *Biometrika*, v. 78, p. 691–692, 1991.

NIETO, P. G. et al. A hybrid pso optimized svm-based model for predicting a successful growth cycle of the spirulina platensis from raceway experiments data. *J Comput Appl Math.*, v. 291, p. 293–303, 2016.

NURSYAHID, A. et al. Analysis of deep water culture (dwc) hydroponic nutrient solution level control systems. *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, v. 1108, n. 1, p. 012032, mar 2021. Disponível em: <https://dx.doi.org/10.1088/1757-899X/1108/1/012032>.

O’SULLIVAN, C. A. et al. Sifns for crop monitoring. *Agronomy*, v. 11, n. 6, p. 1210, 2019.

- PAPAGEORGIOU, E. I.; MARKINOS, A. T.; GEMTOS, T. A. Fuzzy cognitive map-based approach for predicting crop production as a basis for decision support system in precision agriculture application. *Appl. Soft Comput.*, v. 11, n. 4, p. 3643–3657, 2011.
- PARK, S. J.; HWANG, C. S.; VLEK, P. L. G. Comparison of adaptive techniques to predict crop yield response under varying soil and land management conditions. *Agric Syst.*, v. 85, n. 1, p. 59–81, 2005.
- PRESSMAN, B. R. M. R. S. *Engenharia de Software Uma Abordagem Profissional*. 8. ed. [S.l.]: Bookman, 2016. 43–44 p.
- RAMAN, R. K. et al. Reconnoitering precision agriculture and resource management: A comprehensive review from an extension standpoint on artificial intelligence and machine learning. *Indian Research Journal of Extension Education*, v. 24, n. 1, p. 108–123, 2024.
- RITTER, E. et al. Comparison of hydroponic and aeroponic cultivation systems for the production of potato minitubers. *Potato Research*, Springer, v. 44, n. 2, p. 127–135, 2001.
- ROUSSEEUW, P. J.; LEROY, A. M. *Robust Regression and Outlier Detection*. [S.l.]: Wiley, 1993.
- RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *nature*, Nature Publishing Group UK London, v. 323, n. 6088, p. 533–536, 1986.
- SANIKHANI A., S. N.; OTHERS. Non-tuned data intelligent model for soil temperature assessment. *Journal of Precision Agriculture*, v. 72, p. 48–57, 2018.
- SELVARAJ, M. G. et al. Ai-powered banana diseases and pest detection. *Plant Methods*, v. 15, n. 1, p. 1–11, 2019.
- SHARMA, N.; SHAMKUWAR, M.; SINGH, I. The history, present and future with iot. *Internet of things and big data analytics for smart generation*, Springer, p. 27–51, 2019.
- SHRESTHA, A.; DUNN, B. Hydroponics. Oklahoma Cooperative Extension Service, 2010.
- SINGH, N.; SANIKHANI, A. Satellite imagery for soil assessment in agriculture. *Journal of Soil Science*, v. 64, n. 5, p. 530–540, 2018.
- SLADOJEVIC, S. et al. Deep neural networks-based recognition of plant diseases by leaf image classification. *Comput. Intelligence and Neuroscience*, v. 2016, p. 1–11, 2016.
- SON, J. E.; KIM, H. J.; AHN, T. I. Chapter 20 - hydroponic systems. In: KOZAI, T.; NIU, G.; TAKAGAKI, M. (Ed.). *Plant Factory (Second Edition)*. Second edition. Academic Press, 2020. p. 273–283. ISBN 978-0-12-816691-8. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780128166918000200>.
- STRANG, G. *Introduction to Linear Algebra*. [S.l.]: Wellesley-Cambridge Press, 1986.
- SURESH, P. et al. A state of the art review on the internet of things (iot) history, technology and fields of deployment. In: IEEE. *2014 International conference on science engineering and management research (ICSEMR)*. [S.l.], 2014. p. 1–8.

VERMA, M. S.; GAWADE, S. D. A machine learning approach for prediction system and analysis of nutrients uptake for better crop growth in the hydroponics system. In: IEEE. *2021 international conference on artificial intelligence and smart systems (ICAIS)*. [S.l.], 2021. p. 150–156.

WILLMOTT, C. J.; MATSUURA, K. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, v. 30, p. 79–82, 2005.

ZHAO X., P. K.; ELSHORBAGY, A. Artificial neural network-based prediction of soil properties. *Journal of Hydrology*, v. 362, p. 1–18, 2009.