

# Sistema de Apoio à Decisão Nutricional com Algoritmo KNN

Roger da Palma Culau

Orientador: Prof. Herysson Rodrigues Figueiredo

Curso de Ciência da Computação  
Universidade Franciscana (UFN)  
Santa Maria – RS

2025

# Sumario

- 1 Introdução
- 2 Objetivos
- 3 Fundamentos de Nutrição
- 4 Metodologia
- 5 Projeto e Implementação
- 6 Resultados
- 7 Conclusão

# Introdução

- A alimentação adequada é fundamental para a manutenção da saúde, prevenção de doenças crônicas e qualidade de vida.
- Pacientes com restrições alimentares precisam de substituições que preservem o perfil nutricional, o que torna a escolha manual de alimentos uma tarefa complexa e demorada.
- A Tabela Brasileira de Composição de Alimentos (TACO) reúne dados nutricionais detalhados, mas em formato pouco prático para o uso direto na rotina do profissional.
- Nesse contexto, um Sistema de Apoio à Decisão que utiliza técnicas de aprendizado de máquina, como o algoritmo KNN, pode transformar esses dados em recomendações nutricionais mais ágeis, consistentes e personalizadas.

# Objetivo Geral

Desenvolver um Sistema de Apoio à Decisão para auxiliar nutricionistas na substituição de alimentos, com base em perfis nutricionais semelhantes obtidos a partir da Tabela TACO e do algoritmo KNN.

# Fundamentos de Nutrição

- Nutrição envolve processos de ingestão, digestão, absorção e utilização de nutrientes.
- Equilíbrio entre:
  - **Macronutrientes:** carboidratos, proteínas, lipídeos;
  - **Micronutrientes:** vitaminas e minerais.

# Tabela Brasileira de Composição de Alimentos (TACO)

- Uma das principais referência nacional em composição centesimal de alimentos.
- Disponibiliza, para cada alimento:
  - Energia (kcal);
  - Macronutrientes (carboidratos, proteínas, lipídeos);
  - Diversos micronutrientes.
- É utilizada como base de dados do sistema proposto.
- Permite a construção de vetores nutricionais para fins de comparação.

# Sistemas de Apoio à Decisão (SAD)

- Componentes principais:
  - ① Base de dados com informações;
  - ② Base de modelo;
  - ③ Interface para interação com o decisor.

# Algoritmo K-Nearest Neighbors (KNN)

- Técnica de aprendizado supervisionado baseada em instâncias.
- Busca de seus vizinhos mais próximos.
- Depende da definição de uma métrica de distância no espaço de características.



# Ferramentas e Tecnologias

- **Python** – implementação do backend e do algoritmo KNN.
- **FastAPI** – framework para criação da API REST.
- **SQLite** – banco de dados relacional leve para armazenamento da TACO.
- **React / Expo** – desenvolvimento da interface web.
- **HTML / CSS** – estruturação e estilização da camada visual.
- **GitHub** – controle de versão e repositório do código-fonte.

# Metodologia de Desenvolvimento

- Utilização de práticas do framework *Scrum*, adaptadas ao contexto acadêmico:
  - Sprints alinhadas ao calendário do curso.
  - Acompanhamento periódico com o orientador.
- Uso de quadro de tarefas (Trello) e versionamento com Git.

# Visão Geral do Sistema

- Sistema de Apoio à Decisão voltado à recomendação de alimentos substitutos.
- Fluxo principal:
  - ① Seleção de um alimento de referência.
  - ② Definição dos nutrientes que devem compor o perfil.
  - ③ Aplicação do KNN sobre a base TACO.
  - ④ Exibição de uma lista de alimentos similares com pontuação de similaridade.

# Requisitos e Histórias de Usuário

- Requisitos especificados em formato de histórias de usuário.
- Agrupamento em quatro épicos:
  - ① **Entrada de Dados**
  - ② **Processamento com KNN**
  - ③ **Exibição de Resultados**
  - ④ **Interface e Usabilidade**
- Exemplo: informar alimento de referência, ajustar nutrientes considerados e visualizar lista de recomendações.

# Construção da Base de Dados

- Utilização de arquivo `TACO.json` como fonte de dados.
- Conversão para **SQLite**:
  - Cada registro JSON torna-se uma linha na tabela de alimentos.
  - Atributos como `id`, `description`, `category`, `energy_kcal`, entre outros.
- Tratamento de dados:
  - Conversão de valores "NA" e "Tr" para nulos;
  - Normalização de formatos numéricos com vírgula decimal;
  - Arredondamento controlado conforme o nutriente.

```
1 [
2   {
3     "id": 1,
4     "description": "Arroz, integral, cozido",
5     "category": "Cereais e derivados",
6     "humidity_percents": 70.13866666666667,
7     "energy_kcal": 123.5348925,
8     "energy_kj": 516.86999022,
9     "protein_g": 2.58825,
10    "lipid_g": 1.00033333333333,
11    "cholesterol_mg": "NA",
12    "carbohydrate_g": 25.80975,
13    "fiber_g": 2.74933333333333,
14    "ashes_g": 0.463,
15    "calcium_mg": 5.204,
16    "magnesium_mg": 58.702,
17    "manganese_mg": 0.627333333333333,
18    "phosphorus_mg": 105.853,
19    "iron_mg": 0.262,
20    "sodium_mg": 1.24466666666667,
21    "potassium_mg": 75.1516666666667,
22    "copper_mg": 0.0203333333333333,
23    "zinc_mg": 0.682666666666667,
24    "retinol_mcg": "NA",
25    "re_mcg": "",
26    "rae_mcg": "",
27    "thiamine_mg": 0.08,
28    "riboflavin_mg": "Tr",
```

# KNN no Backend

- Backend em Python com API e acesso ao banco SQLite.
- Etapas principais:
  - ① Seleção das colunas nutricionais a partir dos parâmetros da requisição.
  - ② Carregamento do vetor do alimento base.
  - ③ Montagem da matriz de candidatos.
  - ④ Aplicação do KNN e ordenação por similaridade.
- Pré-processamento:
  - Imputação de valores ausentes;
  - Normalização *z-score* para harmonizar escalas entre nutrientes.

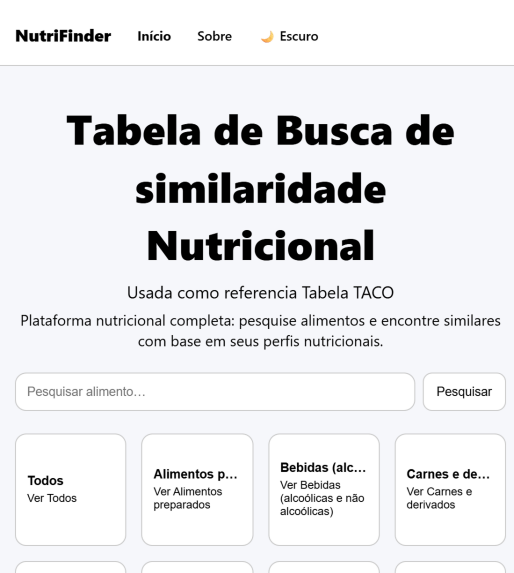
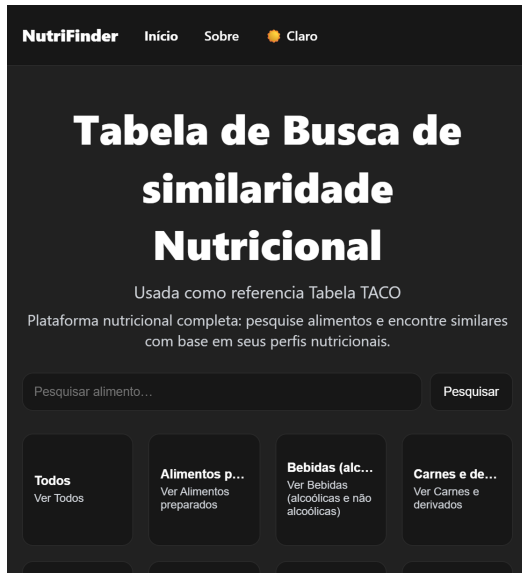
# Distância e Pontuação de Similaridade

- Distância calculada em espaço de atributos normalizados.
- Exemplo de distância euclidiana:

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

- Conversão da distância em score de similaridade (0 a 100):
  - Distâncias mínima e máxima utilizadas em normalização linear;
  - Menor distância recebe score mais elevado;
  - Limitação do score ao intervalo  $[0, 100]$ .

# Interface Web





# Demonstração de Uso

- Seleção de alimento base a partir da TACO.
- Definição de nutrientes.
- Execução da recomendação e exibição da lista de alimentos similares.
  - link video demonstração - <https://www.youtube.com/watch?v=EwvDNpPti2U>

# Conclusões

- O sistema integra:
  - Base de dados TACO em SQLite;
  - Algoritmo KNN calcula e exibe os mais similares;
  - Interface amigável e responsiva para interação com o usuário.
- O objetivo de apoiar a substituição de alimentos com base em perfis nutricionais semelhantes foi atingido.
- O sistema reduz o esforço manual e oferece critérios mais padronizados de comparação.

# Limitações e Trabalhos Futuros

- **Limitações:**

- Dependência da TACO como única fonte de dados;
- Ausência de personalização por paciente;
- Uso de uma métrica de distância.

- **Trabalhos futuros:**

- Testes com nutricionistas em ambiente real;
- Inclusão de outros bancos de alimentos;
- Incluir cálculo com peso para os alimentos;
- Pesquisas de novas métricas de similaridade e visualizações.

# Agradecimentos

Obrigado pela atenção!

Fico à disposição para perguntas.