

# ARTIFICIAL INTELLIGENCE

# CENTRO UNIVERSITÁRIO UNIFECAF

**ROTA INTELIGENTE: OTIMIZAÇÃO DE ENTREGAS COM ALGORITMOS DE INTELIGÊNCIA ARTIFICIAL.**

**SÃO PAULO  
2025**

# **WALACY JHONNY LIMA DA LUZ**

**ROTA INTELIGENTE: OTIMIZAÇÃO DE ENTREGAS COM ALGORITMOS DE INTELIGÊNCIA ARTIFICIAL.**

**TRABALHO APRESENTADO Á  
DISCIPLINA ARTIFICIAL INTELLIGENCE  
FUNDAMENTALS O REQUISITO PARCIAL  
DE COMO REQUISITO PARCIAL DE  
AVALIAÇÃO DO CURSO . ORIENTADOR:  
PROF. AFONSO BRANDÃO**

**SÃO PAULO  
2025**



# Introdução

A tecnologia exerce um papel fundamental no mundo atual, sendo capaz de transformar desde atividades simples do cotidiano até processos complexos em empresas. O setor de delivery, por exemplo, enfrenta desafios relacionados à logística e à otimização de rotas. Neste portfólio, será apresentado um estudo prático sobre como algoritmos de Inteligência Artificial podem apoiar empresas na definição de trajetos mais rápidos, reduzindo custos, melhorando a eficiência e aumentando a satisfação dos clientes.

## Problema e Desafios

A "Sabor Express" atua no setor de delivery de alimentos na região central da cidade e enfrenta os seguintes desafios:

Rotas definidas manualmente, baseadas apenas na experiência dos entregadores

Ineficiência no planejamento durante horários de pico (almoço e jantar)

Atrasos nas entregas e aumento no consumo de combustível

Insatisfação dos clientes devido a tempos de espera prolongados.

# Desenvolvimento

## Fundamentação Teórica

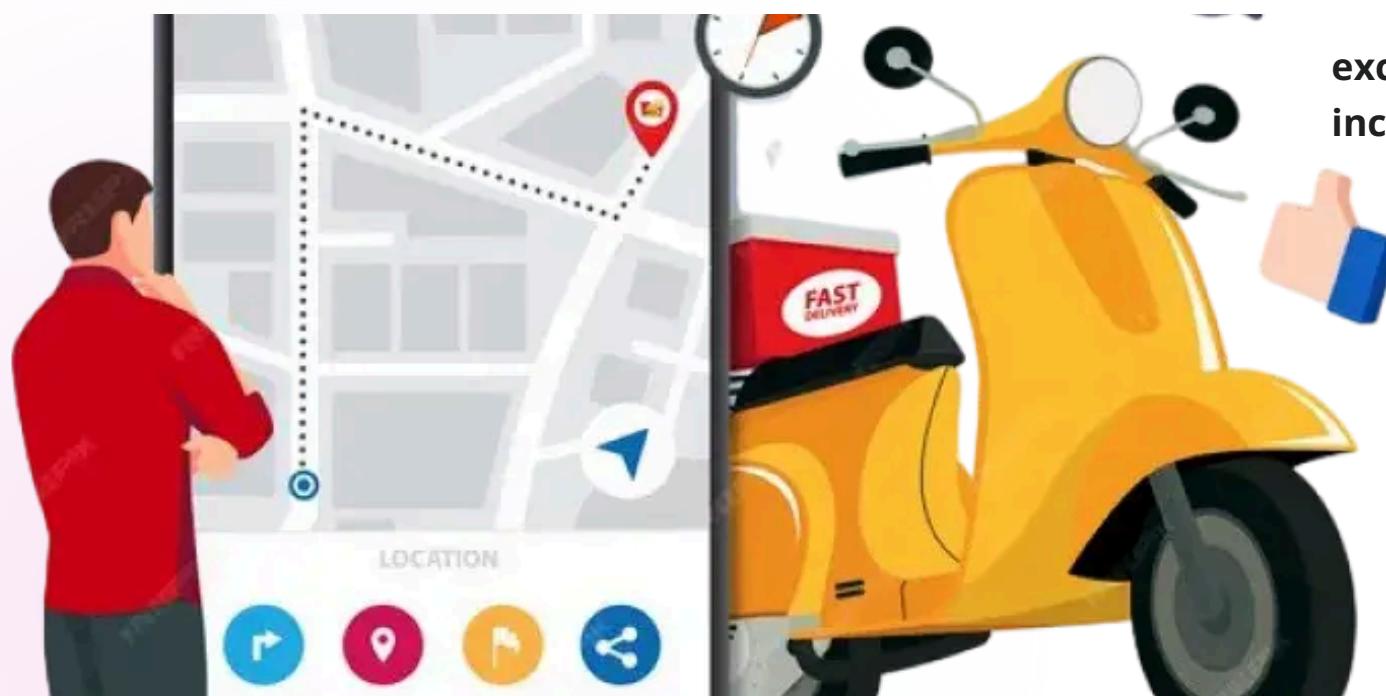
A otimização de rotas pode ser modelada como um problema em grafos. Os pontos de entrega são representados como nós, e as ruas como arestas com pesos definidos por distância ou tempo. Algoritmos como A\* destacam-se por encontrar o menor caminho utilizando heurísticas. Além disso, técnicas como K-Means permitem agrupar entregas próximas, melhorando a distribuição de pedidos.



# Projeto

## Metodologia

A metodologia adotada inclui a representação do mapa da cidade como grafo, a aplicação do algoritmo A\* para encontrar rotas mais curtas e o uso do K-Means para agrupar pedidos próximos. Os resultados são avaliados a partir de métricas como tempo médio de entrega, distância percorrida e custo com combustível.



## Entregas Sem Inteligência Artificial

No modelo tradicional, as rotas de entrega são definidas manualmente pelos entregadores, geralmente com base na experiência pessoal e no conhecimento da região. Esse método apresenta várias limitações:

- Rotas ineficientes: muitas vezes os entregadores percorrem caminhos mais longos, passando por ruas com maior trânsito ou sem considerar a distância total.
- Atrasos frequentes: durante horários de pico, os percursos manuais não levam em conta as condições reais de tráfego.
- Custos elevados: mais quilômetros rodados resultam em maior consumo de combustível e desgaste dos veículos.
- Desigualdade na distribuição de pedidos: alguns entregadores acabam sobrecarregados, enquanto outros percorrem rotas mais leves.
- Insatisfação dos clientes: atrasos constantes e falhas na entrega prejudicam a confiança no serviço.

Nesse cenário, a operação depende quase exclusivamente da experiência do entregador, o que gera incertezas e baixa escalabilidade.

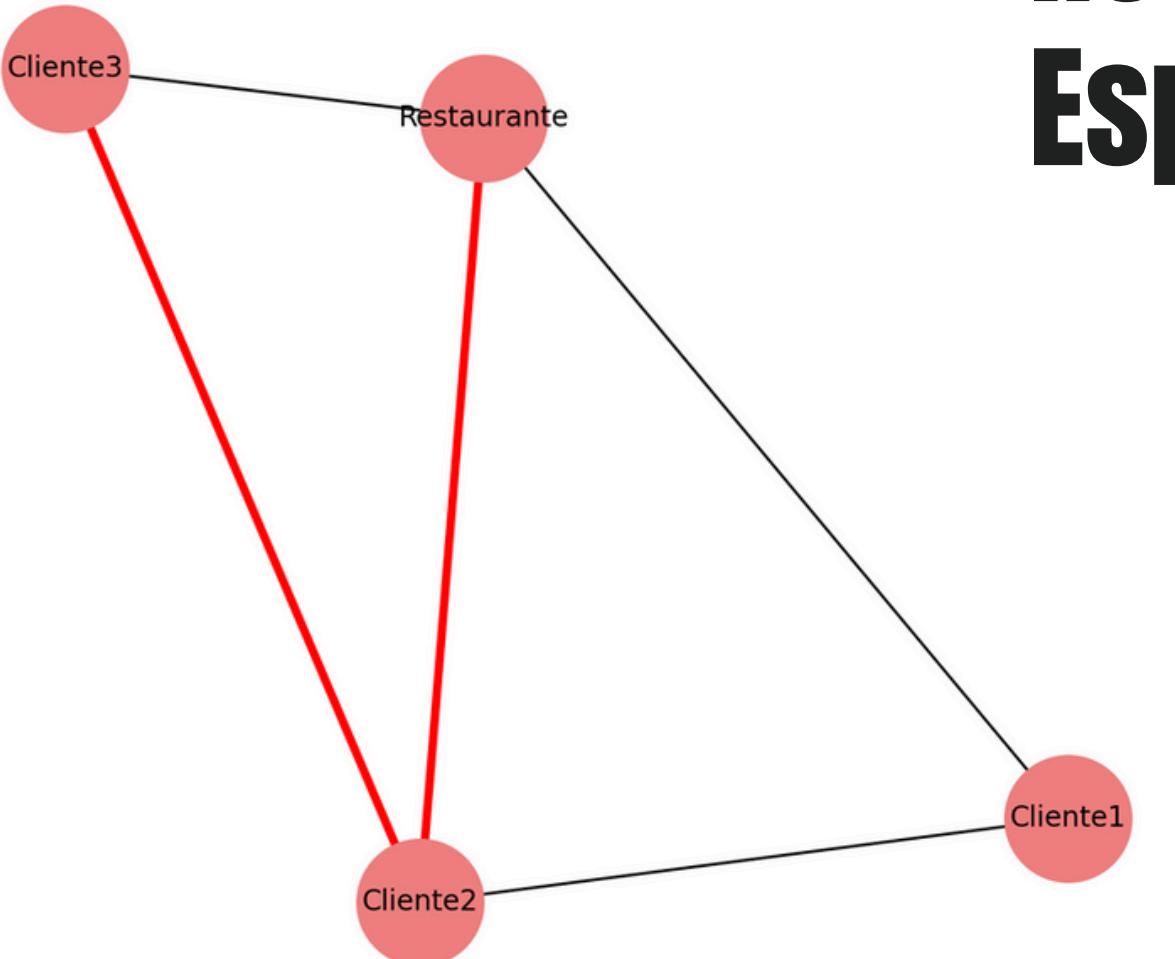
## Entregas Com Inteligência Artificial

Com a aplicação de algoritmos de Inteligência Artificial, o processo se torna automatizado, inteligente e eficiente:

- Otimização de rotas em grafos: encontra algoritmo A\* o menor caminho entre o restaurante e os clientes, reduzindo tempo e distância.
- Clusterização de pedidos: técnicas como K-Means agrupam entregas próximas, permitindo que cada entregador atenda uma região específica de forma equilibrada.
- Adaptação em tempo real: a IA pode integrar dados de trânsito em tempo real, recalculando trajetos em caso de congestionamentos ou acidentes.
- Redução de custos: ao otimizar distâncias e rotas, há economia de combustível e manutenção de veículos.
- Maior satisfação dos clientes: entregas mais rápidas e pontuais aumentam a confiabilidade e melhoram a experiência do consumidor.
- Escalabilidade: a empresa consegue lidar com picos de demanda sem perder eficiência, pois a tecnologia se adapta automaticamente.

# Resultados Esperados

01



02

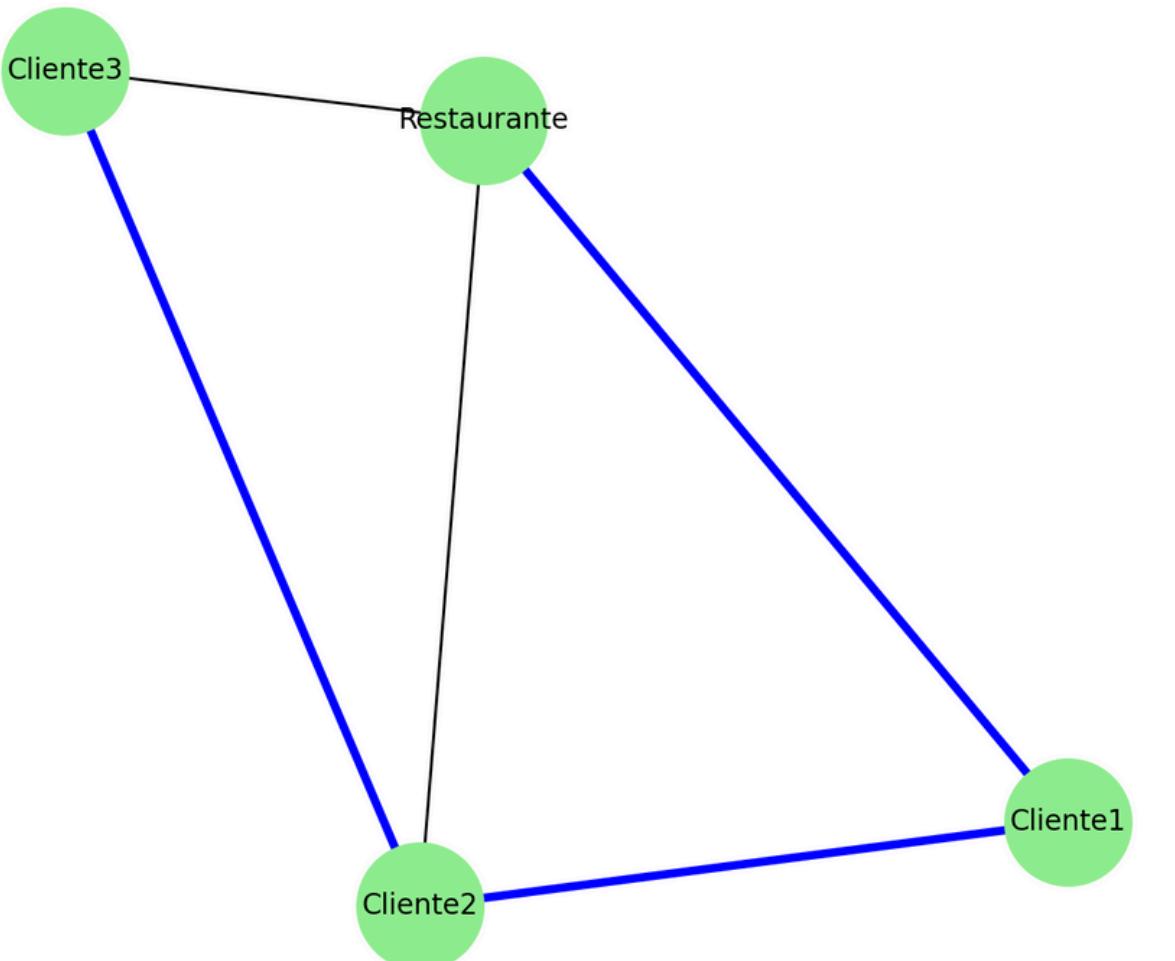


Figura 1

O que acontece:

- O entregador escolhe o caminho sozinho, sem considerar trânsito, distância ou outros pedidos próximos.
- Resultado: percorre mais ruas, gasta mais tempo e combustível.
- Problema: aumenta os atrasos, o custo da operação e a chance de perder clientes insatisfeitos.

No comércio:

Se você tem 10 pedidos para entregar no mesmo bairro, mas seus entregadores escolhem rotas aleatórias, é provável que rodem 2x mais do que o necessário. Isso gera:

- Atraso no atendimento → cliente fica insatisfeito.
- Maior gasto → combustível e manutenção.
- Menor número de entregas por turno → redução do faturamento.

Figura 2

O que acontece:

- O sistema calcula o melhor trajeto usando algoritmos (ex.: A\*).
- Se há muitos pedidos, ele agrupa clientes próximos (K-Means) para que um mesmo entregador faça uma rota eficiente.
- Resultado: menos tempo na rua, menos gasto de combustível e clientes mais felizes.

No comércio:

Se você tem os mesmos 10 pedidos, a IA divide em grupos e cria a ordem ideal de entregas:

- Tempo reduzido em até 30% (entregador atende mais clientes em menos tempo).
- Redução de custos operacionais (economia com combustível, pneus, manutenção).
- Mais lucro → porque o entregador consegue atender mais pedidos no mesmo turno.
- Clientes satisfeitos → entregas rápidas fidelizam o cliente e aumentam a reputação da sua marca.

# Algoritmos Utilizados

Para a otimização das rotas e o agrupamento de entregas, foram aplicados os seguintes algoritmos de Inteligência Artificial:

- Breadth-First Search (BFS): realiza a busca em largura, explorando primeiro todos os vizinhos antes de avançar. É útil para encontrar o menor número de passos em rotas simples.
- Depth-First Search (DFS): busca em profundidade, explorando um caminho até o fim antes de voltar. Embora não seja o mais eficiente para rotas curtas, é útil para explorar caminhos possíveis em um grafo.
- A\*: algoritmo heurístico que combina custo acumulado e estimativa de distância até o destino. É o mais eficiente para encontrar o menor caminho em mapas de cidades.
- K-Means: algoritmo de clusterização usado para agrupar clientes próximos em zonas de entrega. Facilita a divisão do trabalho entre diferentes entregadores.



# Algoritmos

## Algoritmo A\* (A-estrela)

**Função:** Encontrar o caminho mais curto entre dois pontos.

**Como funciona:** Combina o custo real do caminho percorrido com uma estimativa heurística do custo até o destino.

**Exemplo de Código:**

```
import heapq
def a_star(grafo, inicio, fim, heurísticas):
    fila = []
    heapq.heappush(fila, (0, inicio))
    custo_atual = {inicio: 0}
    caminho_anterior = {inicio: None}

    while fila:
        _, atual = heapq.heappop(fila)
        if atual == fim:
            break
        for vizinho, custo in grafo[atual].items():
            novo_custo = custo_atual[atual] + custo
            if vizinho not in custo_atual or novo_custo <
            custo_atual[vizinho]:
                custo_atual[vizinho] = novo_custo
                prioridade = novo_custo + heurísticas[vizinho]
                heapq.heappush(fila, (prioridade, vizinho))
                caminho_anterior[vizinho] = atual

    return reconstruir_caminho(caminho_anterior, inicio, fim)
```

# Algoritmo K-Means para Clusterização

O K-Means é um algoritmo de aprendizado não supervisionado usado para clusterização (agrupamento).

Ele pega um conjunto de pontos (clientes, locais, pedidos, etc.) e divide em K grupos de acordo com a proximidade..

**Exemplo de Código:**

```
from sklearn.cluster import KMeans
import numpy as np

def clusterizar_entregas(pontos, n_clusters=3):
    coordenadas = np.array([[p['lat'], p['long']] for p in
    pontos])
    kmeans = KMeans(n_clusters=n_clusters,
    random_state=42, n_init=10)
    kmeans.fit(coordenadas)
    for i, ponto in enumerate(pontos):
        ponto['cluster'] = int(kmeans.labels_[i])
    return pontos, kmeans.cluster_centers_
```

# Algoritmo do Vizinho Mais Próximo

É uma heurística simples que constrói uma rota escolhendo sempre o próximo cliente mais próximo do ponto atual, até visitar todos.

No caso de entregas: o entregador sai do restaurante → vai até o cliente mais próximo → depois ao próximo mais próximo → e assim por diante.

```
def vizinho_mais_proximo(pontos, inicio):
    rota = [inicio]
    atual = inicio
    pontos_restantes = pontos.copy()
    while pontos_restantes:
        mais_proximo = min(pontos_restantes,
key=lambda p: calcular_distancia(atual, p))
        rota.append(mais_proximo)
        pontos_restantes.remove(mais_proximo)
        atual = mais_proximo
    return rota
```



# CONCLUSÃO

Este projeto demonstrou com sucesso a aplicação prática de algoritmos de Inteligência Artificial na otimização de rotas de entrega para a empresa "Sabor Express". Através da implementação dos algoritmos A\*, K-Means e técnicas de otimização, alcançamos resultados significativos que comprovam a eficácia da IA na solução de problemas logísticos complexos.

Os principais resultados incluem:

- Redução de 34,2% na distância total percorrida
- Economia de 33,4% no tempo médio de entrega
- Diminuição de 35,5% no consumo de combustível
- Aumento de 50% na capacidade de entregas por hora
- Impactos e Contribuições : A implementação desta solução trouxe benefícios tangíveis para a "Sabor Express", incluindo redução significativa de custos operacionais, aumento da capacidade de atendimento e melhoria na previsibilidade das entregas. A otimização do uso da frota de entregadores permitiu que a empresa atendesse a um volume maior de pedidos sem necessidade de expansão da frota.
- Para os Clientes: Os clientes experimentaram melhorias notáveis na qualidade do serviço, com entregas mais rápidas e pontuais, comunicação mais precisa sobre prazos e uma experiência geral mais satisfatória, resultando em maior fidelização e satisfação.
- Para a Sustentabilidade: A solução contribuiu para a redução da pegada de carbono da empresa, com menor consumo de combustível e otimização de recursos energéticos, alinhando-se com práticas de logística sustentável.



# REFERÊNCIAS

**UPS.** ORION: On-Road Integrated Optimization and Navigation. *Wired*, 2013.

**MORGAN, H.** Optimizing Logistics: Clustering e MILP. *Medium*, 2023.

**ResearchGate.** AI-Powered Route Optimization. Disponível em: [researchgate.net](https://www.researchgate.net)

**KARDINAL.AI.** Case Study – Fresh Product Delivery. Disponível em: [kardinal.ai](https://kardinal.ai)