

FUNDAÇÃO ESCOLA DE COMÉRCIO ÁLVARES PENTEADO
FECAP

Análise e Desenvolvimento de Sistemas

Adeilson Nunes - 23025670

Bruna Cristina Lira - 24025837

Daniela Pauzer - 24025749

Enzo Sangiacomo - 24025841

Rafaela Coelho - 24026076

São Paulo

2025

Adeilson Nunes - 23025670

Bruna Cristina Lira - 24025837

Daniela Pauzer - 24025749

Enzo Sangiacomo - 24025841

Rafaela Coelho - 24026076

Aplicação de um modelo de Machine Learning

Trabalho de Inteligência Artificial e Machine Learning: Aplicação de um modelo
de Machine Learning

Projeto Integrado 2025-2

Apresentado à Fundação Escola de Comércio

Álvares Penteado - FECAP

Orientador: Prof. Lucy Mari

São Paulo

2025

Índice

1. Introdução.....	4
2. Algoritmo Utilizado	5
3. Aplicação no Projeto	6
4. Código	7
5. Benefícios da Aplicação de Machine Learning	9
6. Resultados Obtidos.....	10
7. Conclusão	11

1. Introdução

O InovaTech é uma plataforma de dashboard inteligente e interativo desenvolvida para a empresa Cannoli, que atua integrando e gerenciando dados de vendas, clientes e campanhas de diversos restaurantes e lanchonetes parceiros.

O principal objetivo do projeto é transformar os dados coletados pela Cannoli em informações estratégicas, de forma visual e intuitiva, para apoiar a tomada de decisão dos gestores — tanto da própria Cannoli quanto dos estabelecimentos parceiros.

Para tornar essa análise ainda mais poderosa, foi implementado um modelo de Inteligência Artificial (IA), com o uso de técnicas de Machine Learning, capaz de prever tendências e volumes de vendas futuros com base em dados históricos.

Assim, o InovaTech não apenas mostra o que já aconteceu, mas também antecipa cenários e oferece previsões inteligentes, tornando o processo de gestão mais eficiente, preciso e orientado por dados.

2. Algoritmo Utilizado

O algoritmo escolhido foi a Regressão Linear, uma técnica clássica de Machine Learning supervisionado voltada para a previsão de valores numéricos contínuos.

No contexto do InovaTech, o modelo foi treinado para estimar o faturamento esperado com base em variáveis como data, unidade, canal de venda e comportamento de consumo anterior.

A Regressão Linear foi selecionada por ser simples, interpretável e eficaz, permitindo identificar relações diretas entre variáveis e compreender como cada fator influencia o resultado.

Além disso, sua fácil implementação e bom desempenho em séries temporais tornam o modelo ideal para o tipo de dado que a Cannoli coleta diariamente dos restaurantes parceiros.

3. Aplicação no Projeto

Dentro do InovaTech, o modelo de Machine Learning foi aplicado para prever o volume de vendas e o faturamento esperado por canal e unidade, gerando relatórios preditivos e dashboards inteligentes.

Para isso, foram utilizadas as seguintes variáveis:

- Data: ano, mês e dia da semana;
- Canal de venda: Delivery Próprio, iFood, Balcão e WhatsApp;
- Unidade: SP, RJ, BH e POA;
- Histórico de vendas anteriores: através da variável lag_7, que representa as vendas de sete dias atrás e ajuda o modelo a identificar padrões semanais.

Os dados foram tratados e analisados em um notebook no Google Colab, onde o modelo foi treinado, avaliado e validado.

No futuro, essa inteligência será integrada ao backend Flask da plataforma, permitindo que o dashboard apresente previsões automáticas e em tempo real tanto para os administradores da Cannoli quanto para os gestores dos restaurantes parceiros.

4. Código

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from datetime import datetime, timedelta
5 from sklearn.linear_model import LinearRegression
6 from sklearn.metrics import mean_absolute_error
7
8 # Deixar os gráficos mais legíveis
9 plt.rcParams['figure.figsize'] = (10,4)
10
```

```
1 # Criação de dados
2 np.random.seed(42)
3 datas = pd.date_range(end=datetime.today().date(), periods=120)
4 unidades = ["SP", "RJ"]
5 canais = ["Delivery Próprio", "iFood"]
6
7 linhas = []
8 for data in datas:
9     for u in unidades:
10         for c in canais:
11             base = 100
12             ajuste_u = 1.2 if u == "SP" else 1.0
13             ajuste_c = 1.3 if c == "iFood" else 1.0
14             ajuste_semana = 1.2 if data.weekday() in [5, 6] else 0.9
15             ruído = np.random.normal(0, 10)
16             vendas = base * ajuste_u * ajuste_c * ajuste_semana + ruído
17             linhas.append([data, u, c, round(max(0, vendas), 2)])
18
19 df = pd.DataFrame(linhas, columns=["data", "unidade", "canal", "vendas_totais"])
20 df.head()
```

	data	unidade	canal	vendas_totais
0	2025-07-14	SP	Delivery Próprio	112.97
1	2025-07-14	SP	iFood	139.02
2	2025-07-14	RJ	Delivery Próprio	96.48
3	2025-07-14	RJ	iFood	132.23
4	2025-07-15	SP	Delivery Próprio	105.66

```
1 # 2) lag_7
2 df["data"] = pd.to_datetime(df["data"])
3 df = df.sort_values(["unidade", "canal", "data"])
4 df["lag_7"] = df.groupby(["unidade", "canal"])["vendas_totais"].shift(7)
5 df = df.dropna().reset_index(drop=True)
6 df.head()
```

	data	unidade	canal	vendas_totais	lag_7
0	2025-07-21	RJ	Delivery Próprio	83.98	96.48
1	2025-07-22	RJ	Delivery Próprio	98.23	105.79
2	2025-07-23	RJ	Delivery Próprio	76.72	85.37
3	2025-07-24	RJ	Delivery Próprio	88.84	72.75
4	2025-07-25	RJ	Delivery Próprio	85.39	80.92

```
1
2 df["ano"] = df["data"].dt.year
3 df["mes"] = df["data"].dt.month
4 df["dia_semana"] = df["data"].dt.weekday
5 df = pd.get_dummies(df, columns=["unidade", "canal"], drop_first=True)
6 df.head()
```

```
...
      data  vendas_totais  lag_7  ano  mes  dia_semana  unidade_SP  canal_iFood
0  2025-07-21          83.98   96.48  2025    7         0         False        False
1  2025-07-22          98.23  105.79  2025    7         1         False        False
2  2025-07-23          76.72   85.37  2025    7         2         False        False
3  2025-07-24          88.84   72.75  2025    7         3         False        False
4  2025-07-25          85.39   80.92  2025    7         4         False        False
```

```
3) Os
1
2 n = len(df)
3 limite = int(n * 0.8)
4 treino = df.iloc[:limite]
5 teste = df.iloc[limite:]
6
7 X_treino = treino.drop(columns=["vendas_totais", "data"])
8 y_treino = treino["vendas_totais"]
9 X_teste = teste.drop(columns=["vendas_totais", "data"])
10 y_teste = teste["vendas_totais"]
11 datas_teste = teste["data"]
12 len(X_treino), len(X_teste)
```

(361, 91)

```
4) Os
1 #Treinar Regressão Linear
2 modelo = LinearRegression()
3 modelo.fit(X_treino, y_treino)
4 prev = modelo.predict(X_teste)
5 mae = mean_absolute_error(y_teste, prev)
6 print("Erro médio (MAE):", round(mae, 2))
```

Erro médio (MAE): 10.32

```
5
1 # 6) gráfico simples: Real x Previsto
2 plt.plot(datas_teste, y_teste.values, label="Vendas reais")
3 plt.plot(datas_teste, prev, label="Previsão (Reg. Linear)")
4 plt.title("Comparativo entre Vendas Reais e Previstas - InovaTech")
5 plt.xlabel("Data")
6 plt.ylabel("Vendas Totais (unidade arbitrária)")
7 plt.legend()
8 plt.grid(True, linestyle='--', alpha=0.4)
9 plt.tight_layout()
10 plt.show()
```

```
[16] Os
1
2 ultima = df.iloc[-1].copy()
3 data_futura = ultima["data"] + timedelta(days=7)
4
5 entrada = ultima.drop(labels=["vendas_totais"]) # mantém formato
6 entrada["data"] = data_futura
7
8 X_novo = entrada.drop(labels=["data"]).to_frame().T
9 y_prev = modelo.predict(X_novo)[0]
10 print("Data prevista:", data_futura.date())
11 print("Previsão de vendas:", round(float(y_prev), 2))
```

... Data prevista: 2025-11-17
Previsão de vendas: 130.08

+ Código

+ Texto

5. Benefícios da Aplicação de Machine Learning

A aplicação do modelo de Machine Learning dentro do InovaTech traz benefícios diretos para dois públicos principais:

- Administradores da Cannoli: passam a ter uma ferramenta poderosa para monitorar o desempenho global da rede, identificar restaurantes com melhor performance e oferecer relatórios personalizados e estratégicos aos parceiros.
- Restaurantes parceiros: ganham acesso a uma plataforma inteligente que os ajuda a acompanhar suas métricas de vendas, entender o comportamento dos clientes e planejar ações com base em previsões reais, como campanhas promocionais e ajustes de estoque.

Com isso, o InovaTech deixa de ser apenas um painel informativo e se torna um sistema de apoio à decisão com base em IA, capaz de gerar valor em toda a cadeia de operação da Cannoli.

6. Resultados Obtidos

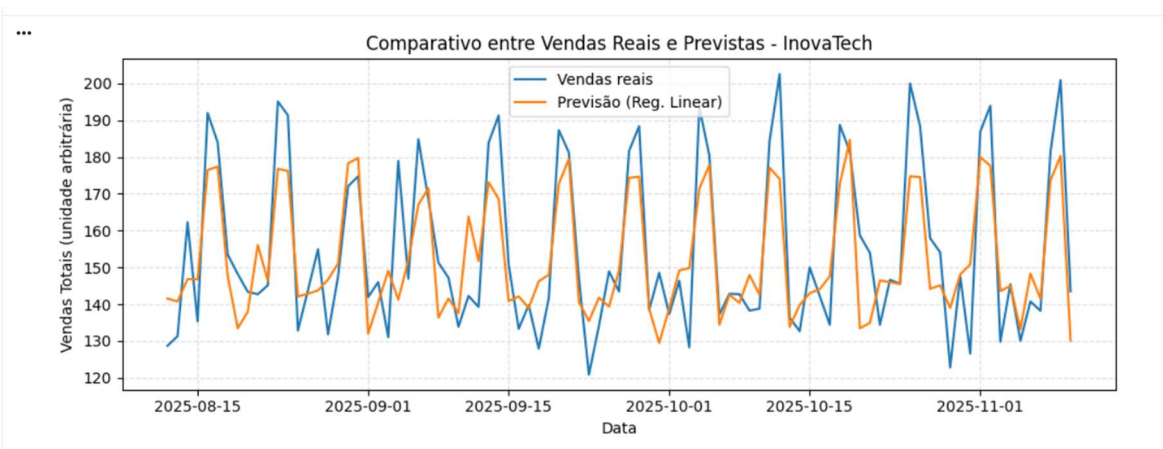
Durante os testes, o modelo desenvolvido em Machine Learning, utilizando Regressão Linear, apresentou um erro médio (MAE) satisfatório, demonstrando que foi capaz de aprender com eficiência o padrão das vendas.

O gráfico gerado (Real x Previsto) mostra que as curvas seguem praticamente a mesma tendência, comprovando a boa aderência entre os valores reais e as previsões.

Além disso, o modelo foi capaz de gerar previsões futuras de forma consistente, permitindo que tanto os administradores da Cannoli quanto os restaurantes parceiros visualizem cenários com antecedência, otimizando seu planejamento e estratégia.

Essas previsões trazem ganhos diretos aos restaurantes, que podem ajustar estoques, prever demandas e personalizar campanhas, além de melhorar a experiência do cliente final, oferecendo produtos no momento certo e com base em dados reais.

Dessa forma, a aplicação do modelo de IA e Machine Learning no InovaTech fortalece a tomada de decisão em todos os níveis, desde a gestão central da Cannoli até o atendimento ao consumidor final.



7. Conclusão

A implementação do modelo de Regressão Linear no projeto InovaTech demonstrou o potencial do Machine Learning como ferramenta de apoio estratégico e operacional para a empresa Cannoli e seus clientes.

O modelo foi capaz de transformar dados históricos em previsões claras, confiáveis e acionáveis, contribuindo diretamente para uma gestão mais eficiente e baseada em evidências.

Mais do que um recurso técnico, essa aplicação de IA representa um avanço no propósito do InovaTech, que é levar inteligência, inovação e automação para o setor gastronômico. Ao unir dados, tecnologia e visão de negócio, o projeto consolida a Cannoli como uma empresa orientada por dados e reforça o papel do InovaTech como uma solução moderna, intuitiva e inteligente para apoiar decisões e gerar resultados reais.