



Relatório de Criação do Sistema de Padaria

1. Introdução

Este relatório apresenta o desenvolvimento de um sistema de gestão para uma padaria, com foco em controle de produção (telas de pão), registros de vendas, pedidos e previsões automatizadas com uso de Machine Learning. O sistema foi planejado para auxiliar no controle diário de produção e fornecer suporte à tomada de decisão com base em dados históricos.

2. Etapas do Desenvolvimento

2.1 Levantamento de Requisitos

A primeira etapa consistiu na identificação das necessidades da padaria, como:

- Registro das produções diárias de pão (tipo grossa e fina)
- Registro das vendas realizadas por turno
- Controle de pedidos
- Geração de previsões para melhor planejamento da produção

2.2 Coleta e Estruturação de Dados

Foram definidos os principais dados necessários, como:

- Quantidade de telas produzidas por tipo e turno
- Quantidade vendida por tipo e horário
- Pedidos realizados
- Previsão de produção com base no histórico

2.3 Modelagem do Banco de Dados

Foi criado o banco **padaria** com as seguintes tabelas:

Tabela: telas

Registra a quantidade de telas produzidas por data, tipo e turno.

Campos:

- `id_telas, data, semana`
- `telas_grossa_manha, telas_grossa_tarde`
- `telas_fina_manha, telas_fina_tarde`

Tabela: `telas_vendidas3`

Registra a quantidade real de telas vendidas por data.

Relaciona-se com `telas` via `id_telas`.

Tabela: `horarios`

Registra os detalhes das vendas por horário, tipo e turno.

Relaciona-se com `telas` via `id_telas`.

Tabela: `pedidos`

Registra os pedidos realizados, quantidade de pacotes por tipo e valor total.

Tabela: `predicao`

Criada apenas para armazenar dados previstos de produção (não faz parte do funcionamento rotineiro).

3. Estrutura e Funcionamento do Código Principal

O sistema foi desenvolvido em Python, com conexão ao banco MySQL e menu interativo por terminal. A função principal `main()` organiza a execução das funcionalidades:

Menu de Opções:

Opção	Ação
1	Inserir dados de produção (telas)
2	Inserir dados de horários
3	Gerar predição semanal
4	Ver ou alterar dados da tabela <code>telas</code>
5	Ver histórico de produção por data

6	Realizar pedidos automaticamente
7	Realizar pedidos manualmente
8	Encerrar o sistema

As funções são organizadas para modularidade e reutilização.

Principais Funções:

- **processa_turno()**: processa dados de vendas para um tipo/turno.
- **inserir_horarios_personalizado()**: permite registrar vendas por turno sem interação com o usuário.
- **inserir_pedidos_manual()** e **inserir_pedidos_automatizado()**: controlam os pedidos e calculam valores.
- **criar_predicao_semana()**: gera a previsão de produção com base em dados históricos.

4. Predição de Produção: Como Funciona?

A previsão é feita para os próximos 7 dias com base na quantidade de telas vendidas (dados da **telas_vendidas3**). O objetivo é antecipar a demanda para:

- Grossa manhã / tarde
- Fina manhã / tarde

Fluxo da Predição:

1. Coleta os dados históricos da tabela **telas_vendidas3**.
 2. Extrai o dia da semana e converte para variável categórica.
 3. Treina **um modelo de regressão linear para cada tipo/turno** com base no dia da semana.
 4. Usa os modelos treinados para prever as quantidades futuras.
 5. Pergunta ao usuário se deseja inserir os dados no banco.
 6. Permite alterar manualmente os valores previstos.
-

5. Justificativa do Uso da Regressão Linear

O algoritmo de **regressão linear** foi escolhido por ser:

- Simples e eficaz para padrões cíclicos (como dias da semana)
- Fácil de interpretar (ajuste de linha para prever valores contínuos)
- Adequado para dados com relação clara entre variáveis (ex: “terça-feira” tende a vender X pães)

Como é usado:

- O dia da semana é convertido com OneHotEncoder.
 - O modelo aprende quantas telas são vendidas em cada dia da semana.
 - Isso permite prever a quantidade de telas futuras com base nos padrões anteriores.
-

6. Pesquisa para Escolha do Algoritmo

A escolha do algoritmo foi fundamentada em estudos acadêmicos, como:



Estudo sobre Machine Learning em Saúde (Resumo):

- Aplicação de ML para prever óbitos em idosos
- Usaram 5 algoritmos (logística, redes neurais, etc.)
- Regressão logística apresentou bom desempenho com AUC > 0,70








Monografia sobre Previsão de Hipertensão e Diabetes:

- Usou regressão logística e redes neurais com base em dados sociodemográficos
- Mostrou que ML pode prever padrões complexos em dados simples

Com base nesses estudos, optou-se por um modelo simples e interpretável: **regressão linear**, ideal para iniciar a previsão da demanda da padaria, mas com possibilidade de testar outros modelos no futuro.

7. Pontos Fortes do Sistema

-  Interface de terminal simples e interativa
 -  Registro detalhado de produção e vendas
 -  Geração de previsões com aprendizado de máquina
 -  Flexível: permite ajustes manuais e correção de dados
 -  Baseado em boas práticas de modelagem e análise de dados
-

8. Melhorias Futuras

- Adicionar interface gráfica (GUI ou Web)
 - Validar entradas do usuário (para evitar erros)
 - Salvar logs de previsões e alterações
 - Testar algoritmos mais robustos (ex: Random Forest, XGBoost)
 - Implementar login de usuários e controle de permissões
 - Evitar duplicidade na inserção de dados
-

9. Conclusão

O sistema desenvolvido atinge o objetivo de auxiliar a padaria no controle da produção e vendas, usando dados históricos para prever demandas futuras. O uso de Machine Learning, mesmo em sua forma mais simples (regressão linear), já oferece ganhos importantes em organização e planejamento, com base em evidências de estudos acadêmicos.