

Projeto 1 – Logística Quantitativa Aplicada

Etapa 1: Formulação, Modelagem e Desenho Computacional

Disciplina: Logística

Instituição: Universidade de Brasília (UnB) — Faculdade de Tecnologia (FT)

Departamento: Engenharia de Produção (EPR)

Professor: João Gabriel de Moraes Souza

Discente: Antônio Augusto Maciel Guimarães (190084421)

Data de Entrega: 31 de janeiro de 2026

Resumo Executivo

Este projeto aborda o problema de **repositionamento de estoque em rede varejista** com alta variabilidade de demanda, utilizando o dataset **M5 Forecasting (Walmart)** do Kaggle. O objetivo é propor políticas de reposição que minimizem custos totais e garantam níveis de serviço adequados, integrando:

- **Previsão de demanda:** modelos de séries temporais (ETS, ARIMA, LightGBM, Ensemble) com avaliação por métricas robustas (WAPE, MAE, RMSE)
- **Modelagem determinística:** EOQ como baseline de custo total
- **Modelagem estocástica:** política (R,Q) com estoque de segurança derivado de variância dos resíduos
- **Risk pooling:** comparação entre estoque descentralizado vs. centralizado por estado/CD
- **Localização de CDs:** cenários com alocação de lojas e custos de transporte
- **Simulação:** planejada para Etapa 2 via SimPy com incerteza de demanda e lead time

O projeto integra os tópicos I a VII da disciplina, com foco central em **Gestão de Estoques Estocásticos (Tópico IV)**.

1. Formulação do Problema Logístico

1.1 Contexto Operacional

Uma rede varejista opera múltiplas lojas com milhares de produtos cujas vendas diárias são influenciadas por:

- Sazonalidade semanal e anual
- Eventos e feriados do calendário
- Promoções e alterações de preço
- Efeitos regionais (estado e localização da loja)

As decisões de reposição afetam diretamente:

- **Rupturas:** perda de vendas e degradação do nível de serviço
- **Excesso de estoque:** custos de capital, armazenagem e risco de obsolescência
- **Custos logísticos:** pedidos, transporte e operação de centros de distribuição

1.2 Objetivo Geral

Definir uma política de reposição que minimize o custo logístico esperado e garanta nível de serviço adequado, integrando previsão de demanda, gestão de estoques e cenários de centralização e localização.

1.3 Função Objetivo

No horizonte temporal T , minimizar:

$$\min$$

Onde:

- C_{pedido} : custo fixo por pedido (setup ou ordem de compra)
- $C_{holding}$: custo de manter estoque (h por unidade por período)
- $C_{ruptura}$: penalidade por falta (custo de oportunidade ou backorder)
- $C_{transporte}$: custo variável (proxy por distância \times volume)

1.4 Variáveis de Decisão

Por item i e loja s :

- $Q_{i,s}$: lote de reposição (quantidade por pedido)
- $R_{i,s}$: ponto de pedido (reorder point)

Em cenários de centralização e localização:

- $x_{s,c} \in \{0,1\}$: loja s atendida pelo centro de distribuição c
- $y_c \in \{0,1\}$: abertura do CD c em local candidato (opcional)

1.5 Restrições e Nível de Serviço

Nível de serviço (meta):

- **CSL (Cycle Service Level):** $P(\text{não faltar no ciclo}) \geq \alpha$ (exemplo: 95%)
- **Fill rate β :** fração atendida sem falta $\geq \beta$ (exemplo: 98%)

Restrições operacionais:

- Capacidade de estoque por loja: I_s^{max}
- Orçamento por ciclo: $\sum_i c_i Q_{i,s} \leq B_s$
- Lote mínimo ou múltiplo (embalagem, paletização)

1.6 Hipóteses

- Lead time L **determinístico** na Etapa 1; **estocástico** na Etapa 2
 - Custos K , h e p definidos por cenário e analisados via sensibilidade
 - Em recortes, assume-se independência parcial entre lojas; correlação será medida e discutida
-

2. Justificativa da Escolha do Tópico Central

Tópico Central: Gestão de Estoques Estocásticos (Tópico IV)

O eixo do projeto é o **Tópico IV** por refletir melhor o problema real: a demanda (e o lead time) são incertos, e a política ótima depende do trade-off entre custo, serviço e risco.

Interfaces com os demais tópicos:

- **I (Fundamentos):** trade-offs logísticos, métricas de desempenho (CSL, fill rate), custo total e estrutura do sistema logístico
- **II (Previsão de Demanda):** estima média μ e desvio padrão σ por série temporal ou grupo de produtos
- **III (Gestão Determinística):** EOQ e custo total como baseline e ponto de comparação
- **V (Simulação):** validação de políticas e níveis de serviço sob incerteza (Etapa 2)
- **VI (Localização):** desenho de CDs candidatos e alocação de lojas (custo de transporte + efeito no lead time)
- **VII (Centralização/Risk Pooling):** redução de variância efetiva e estoque de segurança total via agregação

Esta integração atende ao requisito de articulação conceitual e analítica com todos os tópicos da disciplina.

3. Base de Dados e Estratégia de Recorte

3.1 Dataset M5 Forecasting (Walmart)

Fonte: Kaggle – M5 Forecasting Accuracy Competition

Link: <https://www.kaggle.com/competitions/m5-forecasting-accuracy/data>

Arquivos principais:

- `sales_train_validation.csv`: vendas diárias históricas (formato wide, 30.490 séries temporais)
- `calendar.csv`: dados de data, eventos, feriados e features temporais
- `sell_prices.csv`: preços por item, loja e semana

3.2 Estrutura Hierárquica

O dataset permite análises em múltiplos níveis:

- **Geografia:** store → state → total
- **Catálogo:** item → dept → cat → total

3.3 Recorte Proposto

Para manter robustez analítica e viabilidade computacional:

Recorte A (principal): 2 lojas no mesmo estado + 2 categorias + aproximadamente 30 itens com maior volume de vendas

Recorte B (intermitência): 10 itens com alta porcentagem de zeros (demanda intermitente)

Recorte C (pooling): mesmas categorias, comparando cenário descentralizado vs. pooling por estado ou CD

A lógica do recorte preserva:

1. Hierarquia e variáveis exógenas (preço, calendário)
2. Comparabilidade de políticas de estoque
3. Reprodutibilidade do pipeline para a Etapa 2

4. Análise Exploratória de Dados (EDA)

4.1 Pré-processamento

Etapas realizadas:

1. Conversão de formato wide para long (melt das colunas de vendas diárias)
2. Merge com `calendar.csv` para adicionar features temporais e eventos
3. Merge com `sell_prices.csv` por (`store_id, item_id, w_m_yr_wk`)
4. Criação de features derivadas: `dow, month, week_of_year, is_weekend, is_event, snap_state, price_change`, etc.

4.2 Diagnósticos por Série Temporal

Entregáveis implementados:

- Série temporal de demanda diária e agregações semanais
- Decomposição sazonal (componentes de tendência, sazonalidade e resíduo)
- Estatísticas descritivas: média, desvio padrão, coeficiente de variação (CV)
- **Análise de intermitência:** porcentagem de zeros, intervalo médio entre vendas (ADI), classificação Syntetos-Boylan
- Identificação de outliers (picos) e associação com eventos ou mudanças de preço

4.3 Diagnósticos por Nível Hierárquico

Análises realizadas:

- Demanda agregada por categoria e departamento ao longo do tempo
- Comparação de padrões entre lojas (diferença sazonal e correlação cruzada)
- Relação preço-demanda (elasticidade aproximada por grupo de produtos)

4.4 Medidas Quantitativas Avançadas

- **ACF/PACF:** análise de autocorrelação em séries selecionadas para identificar padrões ARIMA
- **Correlação cruzada:** medida entre lojas para mesmos itens (relevante para risk pooling)
- **Heatmap de sazonalidade:** padrões dia da semana × mês por categoria

Resultados obtidos (50 itens analisados):

- Classificação de intermitência: 42% Errático, 36% Suave, 22% Grumoso
- Correlação média entre lojas (mesmo item): 0,35 a 0,60 (moderada a forte)
- Sazonalidade semanal evidente em 80% das séries

5. Previsão de Demanda

5.1 Objetivo da Previsão

Prever demanda futura $\hat{D}_{i,s}(t)$ e obter medida de incerteza (variância dos resíduos) para suportar o cálculo de estoque de segurança.

5.2 Modelos Implementados

Baselines (comparação justa):

- Naive: previsão = última observação
- Seasonal Naive: previsão = observação do mesmo dia da semana anterior
- Média Móvel: janela de 7 e 14 dias

Modelos estatísticos:

- **ETS (Exponential Smoothing)**: captura tendência e sazonalidade aditiva (período = 7)
- **ARIMA/SARIMA**: identificação automática de ordem via AIC
- **ARIMAX**: inclusão de variáveis exógenas (preço, eventos, dia da semana)

Modelos avançados:

- **LightGBM**: gradient boosting com features de lag (1, 7, 14, 28 dias), médias móveis, tendência e variáveis cíclicas
- **Ensemble**: combinação ponderada de ETS (25%), ARIMA (25%) e LightGBM (50%)
- **Decomposição Sazonal**: extração de padrão + regressão linear na tendência

5.3 Validação e Métricas

Estratégia de validação:

- Split temporal: treino (80% dos dados) → validação (20%)
- Janela rolante para simular previsão adaptativa

Métricas calculadas:

Métrica	Fórmula	Interpretação
MAE	$\frac{1}{n} \sum y_i - \hat{y}_i $	Erro absoluto médio
RMSE	$\sqrt{\frac{1}{n} \sum (y_i - \hat{y}_i)^2}$	Penaliza erros grandes
WAPE	$\frac{\sum y_i - \hat{y}_i }{\sum y_i }$	Robusto para escalas
SMAPE	$\frac{100}{n} \sum \frac{2 y_i - \hat{y}_i }{ y_i + \hat{y}_i }$	Simétrico, 0-200%

Table 1: Métricas de avaliação de previsão

Performance obtida (horizonte 28 dias, média dos itens):

- LightGBM: WAPE 12-25%, Acurácia 75-88%
- Ensemble: WAPE 15-28%, Acurácia 72-85%
- ETS: WAPE 18-40%, Acurácia 60-82%

5.4 Saída para Modelagem de Estoques

A política de estoque utiliza:

- μ : média prevista da demanda diária
 - σ : desvio padrão estimado via resíduos do modelo
-

6. Modelagem Quantitativa Determinística

6.1 EOQ (Economic Order Quantity)

O modelo EOQ estabelece o lote economicamente ótimo **sem incerteza**:

$$Q_{i,s}^* = \sqrt{\frac{2K_{i,s}D_{i,s}}{h_{i,s}}}$$

Onde:

- $D_{i,s}$: demanda anual estimada ($\bar{d}_{i,s} \times 365$)
- $K_{i,s}$: custo fixo por pedido (setup)
- $h_{i,s}$: custo de holding anual por unidade

6.2 Custo Total Determinístico

$$CT(Q) = \frac{KD}{Q} + \frac{hQ}{2}$$

Interpretação:

- $\frac{KD}{Q}$: custo de pedidos (quanto menor Q , mais pedidos por ano)
- $\frac{hQ}{2}$: custo médio de manutenção de estoque

6.3 Papel como Baseline

O EOQ serve como:

1. Ponto de comparação para políticas estocásticas

2. Estimativa inicial de Q antes de incorporar incerteza
 3. Referência de custo mínimo em cenário determinístico
-

7. Discussão Preliminar sobre Incerteza e Risco

7.1 Principais Fontes de Incerteza

- **Demanda:** sazonalidade, eventos, intermitência, promoções e mudanças de preço
- **Lead time:** variação logística, atrasos e rupturas do fornecedor
- **Correlação entre lojas:** afeta o ganho real de risk pooling
- **Erros de previsão:** caudas pesadas e picos podem dominar o risco

7.2 Política Estocástica (R,Q)

Assumindo demanda durante lead time aproximadamente Normal:

$$\mu_L = \mu \cdot L, \sigma_L = \sigma \cdot \sqrt{L}$$

$$R = \mu_L + z_\alpha \cdot \sigma_L$$

Onde:

- α : nível de serviço CSL desejado (exemplo: 95%)
- z_α : quantil da distribuição Normal (exemplo: $z_{0.95} = 1.645$)
- $SS = z_\alpha \sigma_L$: estoque de segurança

Exemplo numérico:

- $\mu = 10 \text{ un/dia}, \sigma = 3 \text{ un/dia}, L = 5 \text{ dias}, CSL = 95\%$
- $\mu_L = 50 \text{ un}, \sigma_L = 6.71 \text{ un}$
- $SS = 1.645 \times 6.71 = 11.03 \text{ un}$
- $R = 50 + 11.03 = 61.03 \text{ un}$

7.3 Intermittência

Para séries com alta porcentagem de zeros:

- Diagnóstico via ADI (Average Demand Interval) e CV^2
- Uso de distribuição empírica (amostragem) na simulação da Etapa 2
- Alternativas: métodos de Croston, TSB ou distribuição Tweedie

7.4 Centralização e Risk Pooling

Cenários comparados:

1. **Descentralizado:** estoque de segurança por loja
2. **Pooling por estado/CD:** estoque de segurança em agregado
3. **Híbrido:** itens classe A descentralizados; B/C centralizados

Métrica-chave:

$$\text{Redução} = 1 - \frac{SS_{pooled}}{\sum_s SS_s}$$

A redução depende da correlação entre lojas: menor correlação implica maior benefício do pooling.

7.5 Matriz de Risco

Dimensões qualitativas:

- **Eixo X (Probabilidade):** baixa → alta (estimada por frequência de eventos)
- **Eixo Y (Impacto):** custo de ruptura, dias sem estoque, perda de venda

Estratégias de mitigação:

- Aumentar CSL alvo
- Implementar risk pooling
- Revisar políticas de lead time (expediting)
- Adotar políticas (s,S) ou (P,S) para itens críticos

8. Desenho da Implementação Computacional (C3)

8.1 Arquitetura do Pipeline

Visão geral:

1. **Entrada:** dados M5 (vendas + calendário + preços)
2. **Pré-processamento:** conversão wide → long, merge, feature engineering
3. **EDA:** análise exploratória, estatísticas e visualizações
4. **Previsão:** modelos de séries temporais e machine learning
5. **Estoque:** EOQ, política (R,Q), risk pooling
6. **Localização:** cenários de CDs e alocação de lojas
7. **Saídas Etapa 1:** relatório técnico + especificação para Etapa 2
8. **Simulação (Etapa 2):** SimPy com incerteza estocástica

8.2 Estrutura de Pastas

```
projeto_m5/
data/
raw/ # Dados originais M5 (CSV)
processed/ # Dados processados (pickle)
src/
  config.py # Parâmetros do projeto
  io_load.py # Carregamento dos dados
  preprocess.py # Pré-processamento
  eda/
    eda_core.py # Estatísticas e CV
    plots.py # Visualizações
    intermittency.py # Análise de intermitência
  forecast/
    baselines.py # Naive, Seasonal Naive
    ets_arima.py # ETS e ARIMA/SARIMA
    exogenous.py # ARIMAX com regressoras
    metrics.py # MAE, RMSE, WAPE, SMAPE
    advanced.py # LightGBM, Ensemble
  inventory/
    eoq.py # EOQ e custo determinístico
    rq_policy.py # Política (R,Q) estocástica
    service_levels.py # CSL e Fill Rate
    pooling.py # Risk Pooling
  location/
    candidates.py # CDs candidatos
    allocation.py # Alocação loja→CD
    transport_cost_proxy.py
  simulation/ # Etapa 2
    simpy_env.py # Ambiente SimPy
  notebooks/
    o1_preprocess.ipynb
    o2_edu.ipynb
    o3_forecast.ipynb
    o4_inventory.ipynb
  results/
    main.py
    dashboard.py # Dashboard interativo
  requirements.txt
```

8.3 Pseudocódigo do Fluxo Principal

```
MAIN():
cfg = load_config()

// 1) Carregar e preprocessar
sales, calendar, prices = load_m5_raw(cfg)
df = to_long_format(sales)
df = merge_calendar_prices(df, calendar, prices)
df = add_features(df)
```

```

// 2) EDA + seleção de recortes
eda_report(df)
recortes = select_slices(df, strategy="A/B/C")

// 3) Forecast
for slice in recortes:
    train, val = time_split(slice)
    model = fit_forecast(train, models=["ETS","ARIMAX","LGBM"])
    yhat = predict(model, horizon=H)
    resid = compute_residuals(model, val)
    store_sigma(slice.id, resid)

// 4) Estoque determinístico (EOQ)
for series in recortes:
    D = annualize_mean(series)
    Q = eoq(D, K, h)

// 5) Estoque estocástico (R,Q)
for series in recortes:
    mu = forecast_mean(series)
    sigma = forecast_sigma(series)
    R = reorder_point(mu, sigma, L, CSL)

// 6) Pooling
compare_decentral_vs_pooled(recortes, grouping=["state","CD"])

// 7) Localização (proxy Etapa 1)
evaluate_location_scenarios(recortes, cd_candidates)

export_stage1_outputs()

```

8.4 Bibliotecas Utilizadas

Biblioteca	Finalidade
pandas, numpy	Manipulação de dados e estatística
statsmodels	ETS, ARIMA e diagnósticos
scikit-learn	Validação e modelos globais
lightgbm	Gradient boosting para previsão
streamlit, plotly	Dashboard interativo
simpy	Simulação de eventos discretos (Etapa 2)
matplotlib, seaborn	Gráficos e relatórios

Table 2: Bibliotecas previstas no projeto

8.5 Dashboard Interativo

Foi desenvolvido um dashboard Streamlit com 4 páginas:

1. **Visão Geral:** métricas globais, demanda por loja, intermitência
2. **Análise de Demanda:** séries temporais, estatísticas, sazonalidade
3. **Previsão:** comparação de 4 modelos (ETS, LightGBM, Ensemble, Decomposição)
4. **Gestão de Estoques:** EOQ, política (R,Q), análise de sensibilidade

Parâmetros configuráveis via sidebar:

- Lead time: 1–14 dias
 - CSL: 80%–99%
 - Horizonte de previsão: 7/14/21/28 dias
 - Custo de pedido (K): R\$ 10–500
 - Holding anual (h): 10%–50% do valor unitário
-

9. Incorporação da Incerteza na Etapa 2

9.1 Modelagem Estocástica na Simulação

Demanda:

- Distribuição empírica por série (amostragem histórica)
- Cenários de eventos (picos) e promoções

Lead time:

- Distribuição triangular ou normal truncada
- Parâmetros: mínimo, moda, máximo (ou média e desvio)

Políticas de estoque:

- (R,Q): revisão contínua, lote fixo
- (s,S): min-max, lote variável
- (P,S): revisão periódica

9.2 Experimentos e Métricas

Métricas de desempenho (saídas da simulação):

- Custo total: pedido + holding + ruptura + transporte
- CSL e fill rate realizados
- Dias com falta e backorders acumulados
- Estoque médio, máximo e variância
- Sensibilidade por CSL (90%, 95%, 98%), lead time e pooling

Monte Carlo:

- N repetições por cenário (exemplo: 30 replicações)
- Intervalos de confiança 95% para as métricas
- Comparação estatística entre políticas

9.3 Implementação com SimPy

Estrutura do ambiente de simulação:

```
class InventorySystem:
    def __init__(self, env, mu, sigma, L_dist, policy):
        self.env = env
        self.mu = mu
        self.sigma = sigma
        self.L_dist = L_dist
        self.policy = policy

    def generate_demand(self):
        # Amostragem empírica ou distribuição
        pass

    def check_reorder(self):
        # Verifica política (R,Q) ou (s,S)
        pass

    def receive_order(self):
        # Lead time estocástico
        pass
```

Execução:

```
env = simpy.Environment()
system = InventorySystem(env, mu, sigma, L_dist, policy)
env.run(until=365) # Horizonte de 1 ano
results = extract_metrics(system)
```

Referências

- Kaggle. (2020). *M5 Forecasting – Accuracy*. Disponível em: <https://www.kaggle.com/competitions/m5-forecasting-accuracy/data>
 - Código para execução e dados – *Github*. Disponível em: https://github.com/Tominho/Logistica_trabalho_1_2026_1
-

Apêndice A – Parâmetros Econômicos

Como o dataset M5 não fornece custos diretamente, foram utilizados **cenários parametrizados**:

Parâmetro	Valores Testados	Justificativa
K (custo pedido)	R\$ 50, 100, 200	Típico varejo pequeno/médio porte
h (holding anual)	20%, 30%, 40% do custo	Literatura Supply Chain
p (penalidade falta)	2h a 10h por dia	Margem perdida + goodwill
L (lead time)	3, 7, 14 dias	Distribuição regional típica

Table 3: Cenários de parâmetros econômicos

A validade do projeto vem de:

- Coerência metodológica
- Transparência dos cenários e sensibilidade
- Robustez dos resultados por diferentes parametrizações

Apêndice B – Classificação de Intermitência

Método Syntetos-Boylan (2005):

- $ADI = \frac{\text{Períodos totais}}{\text{Períodos com demanda}}$
- $CV^2 = \left(\frac{\sigma}{\mu}\right)^2$ (coeficiente de variação ao quadrado)

Classificação	Critério ADI	Critério CV^2
Suave	< 1.32	< 0.49
Errático	< 1.32	≥ 0.49
Grumoso	≥ 1.32	< 0.49
Intermitente	≥ 1.32	≥ 0.49

Table 4: Critérios de classificação de intermitência

Resultados (50 itens analisados):

- Errático: 21 itens (42%)
- Suave: 18 itens (36%)
- Grumoso: 11 itens (22%)
- Intermitente puro: 0 itens (0%)