

# Análise Multidimensional de Dados e Previsão por Séries Temporais

**Disciplina:** Análise Preditiva

**Curso:** Bacharelado em Engenharia de Software

**Formato:** Projeto em Grupo

**Prazo de Entrega:** 24/11/2025

## 1. Objetivo da Atividade

Esta atividade tem como objetivo avaliar a capacidade dos estudantes de aplicar técnicas de análise multidimensional de dados e previsão por séries temporais em um problema real.

## 2. Competências Avaliadas

Esta atividade avaliará as seguintes competências:

- **Análise e Tratamento de Dados:** Capacidade de identificar, coletar, limpar e preparar dados de diferentes formatos para análise
- **Modelagem Preditiva:** Habilidade de aplicar técnicas de previsão por séries temporais utilizando modelos estatísticos e de machine learning
- **Uso de Ferramentas:** Domínio de bibliotecas Python (pandas, numpy, scikit-learn, statsmodels) para análise de dados
- **Análise Crítica:** Capacidade de interpretar resultados, identificar padrões e extrair insights relevantes dos dados
- **Comunicação Técnica:** Habilidade de apresentar resultados de forma clara, estruturada e fundamentada para audiências técnicas

## 3. Descrição do Projeto

### 3.1. Tarefa Geral

Cada grupo deverá selecionar um **dataset de séries temporais disponível publicamente na internet**, realizar uma análise exploratória completa dos dados, aplicar técnicas de previsão, avaliar o desempenho dos modelos desenvolvidos e construir um dashboard. O projeto deverá culminar em uma **apresentação oral** e na entrega de um **relatório técnico e código-fonte**.

### 3.2. Datasets Sugeridos

Os grupos poderão escolher um dos seguintes datasets ou propor outro similar, **desde que aprovado pela professora com antecedência mínima de 1 semana**:

#### Opção 1: Consumo de Energia Elétrica

- **Fonte:** UCI Machine Learning Repository - Individual Household Electric Power Consumption
- **URL:** <https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption>
- **Descrição:** Medições de consumo elétrico de uma residência francesa ao longo de 4 anos com resolução de 1 minuto
- **Desafio:** Prever consumo futuro considerando padrões diários, semanais e sazonais

#### Opção 2: Tráfego Web - Wikipedia

- **Fonte:** Kaggle - Web Traffic Time Series Forecasting
- **URL:** <https://www.kaggle.com/c/web-traffic-time-series-forecasting>
- **Descrição:** ~145.000 séries temporais representando visualizações diárias de páginas da Wikipedia

- **Desafio:** Prever o número de visualizações futuras considerando tendências e eventos especiais

### Opção 3: Qualidade do Ar

- **Fonte:** UCI Machine Learning Repository - Air Quality Dataset ou Beijing PM2.5 Data
- **URL:** <https://archive.ics.uci.edu/ml/datasets/Air+Quality>
- **Descrição:** Medições horárias de poluentes atmosféricos e variáveis meteorológicas
- **Desafio:** Prever níveis de poluição (PM2.5, NO2, etc.) com base em dados históricos

### Opção 4: Vendas no Varejo

- **Fonte:** UCI Machine Learning Repository - Online Retail Dataset
- **URL:** <https://archive.ics.uci.edu/ml/datasets/online+retail>
- **Descrição:** Transações de uma loja online britânica entre 2010-2011
- **Desafio:** Prever demanda futura de produtos e identificar padrões de sazonalidade

### Opção 5: Preços de Ações

- **Fonte:** Yahoo Finance ou similar
- **Descrição:** Dados históricos de preços de ações (abertura, fechamento, volume) de empresas listadas em bolsa
- **Desafio:** Prever tendências de preços futuros utilizando dados históricos e indicadores técnicos

### Opção 6: Dados Climáticos

- **Fonte:** NOAA (National Oceanic and Atmospheric Administration)
- **URL:** <https://www.noaa.gov/education/resource-collections/data/historical>
- **Descrição:** Registros de temperatura, precipitação, umidade e outras variáveis meteorológicas
- **Desafio:** Prever temperaturas futuras ou probabilidade de precipitação

## 4. Etapas do Projeto

O projeto deverá seguir uma metodologia estruturada e incluir todas as etapas descritas a seguir

### Etapas 1: Coleta e Exploração de Dados (20%)

#### Atividades:

#### 1. Aquisição dos Dados:

- Realizar o download e carregamento do dataset escolhido
- Documentar a fonte, características e período temporal dos dados
- Verificar a integridade e completude do dataset

#### 2. Análise Exploratória de Dados (EDA):

- Gerar estatísticas descritivas (média, mediana, desvio padrão, etc.)
- Identificar valores ausentes, outliers e inconsistências
- Criar visualizações para entender a distribuição e comportamento temporal dos dados
- Analisar tendências, sazonalidade e ciclos presentes na série temporal

#### Critérios de Avaliação:

- Qualidade e profundidade da análise exploratória
- Identificação correta de características da série temporal
- Clareza e adequação das visualizações apresentadas
- Documentação completa do processo de coleta

## **Etapa 2: Preparação e Tratamento dos Dados (20%)**

### **Atividades:**

#### **1. Limpeza de Dados:**

- Tratamento de valores ausentes (imputação, remoção ou interpolação)
- Identificação e tratamento de outliers
- Correção de inconsistências e duplicatas

#### **2. Transformação dos Dados:**

- Normalização ou padronização quando necessário
- Criação de features temporais (dia da semana, mês, trimestre, feriados, etc.)
- Agregação temporal quando apropriado (dados horários → diários, etc.)
- Decomposição da série em componentes (tendência, sazonalidade, resíduos)

#### **3. Divisão dos Dados:**

- Separação em conjuntos de treino, validação e teste respeitando a ordem temporal
- Garantir que o conjunto de teste represente pelo menos 20% dos dados mais recentes

### **Critérios de Avaliação:**

- Adequação das técnicas de limpeza aplicadas
- Qualidade das transformações e feature engineering
- Justificativa das escolhas metodológicas
- Preservação da integridade temporal dos dados

## **Etapa 3: Modelagem Preditiva (30%)**

### **Atividades:**

#### **1. Implementação de Modelos Baseline:**

- Modelo ingênuo (naive forecast)
- Média móvel simples
- Suavização exponencial simples

#### **2. Implementação de Modelos Avançados (mínimo 2):**

##### **Modelos Estatísticos:**

- **ARIMA** (AutoRegressive Integrated Moving Average)
- **SARIMA** (Seasonal ARIMA)
- **Holt-Winters** (Triple Exponential Smoothing)

##### **Modelos de Machine Learning:**

- **Prophet** (Facebook Prophet)

- **LSTM** (Long Short-Term Memory Networks)
- **Regressão com features temporais**
- 3. **Otimização de Hiperparâmetros:**
  - Realizar busca de parâmetros ótimos (grid search, auto\_arima, etc.)
  - Documentar o processo de seleção dos parâmetros
- 4. **Validação Cruzada:**
  - Aplicar validação cruzada temporal (time series cross-validation)
  - Avaliar a estabilidade dos modelos em diferentes períodos

**Critérios de Avaliação:**

- Implementação correta dos algoritmos
- Variedade e adequação dos modelos
- Processo de otimização de hiperparâmetros
- Aplicação adequada de técnicas de validação

**Etapa 4: Avaliação e Comparação de Modelos (20%)**

**Atividades:**

1. **Cálculo de Métricas de Performance: MAE** (Mean Absolute Error)
  - **RMSE** (Root Mean Squared Error)
  - **MAPE** (Mean Absolute Percentage Error)
  - **R<sup>2</sup>** (Coeficiente de Determinação)
2. **Análise Comparativa:**
  - Criar tabela comparativa com métricas de todos os modelos
  - Gerar gráficos de previsões vs. valores reais para cada modelo
  - Analisar resíduos dos modelos (autocorrelação, normalidade, heterocedasticidade)
3. **Seleção do Modelo Final:**
  - Justificar a escolha do melhor modelo com base nas métricas e contexto do problema
  - Discutir limitações e pontos fortes do modelo escolhido

**Critérios de Avaliação:**

- Cálculo correto e interpretação adequada das métricas
- Profundidade da análise comparativa
- Qualidade das visualizações de resultados
- Justificativa robusta da escolha do modelo final

**Etapa 5: Apresentação e Documentação (10%)**

**Atividades:**

1. **Relatório Técnico:**
  - Documento em formato PDF ou Jupyter Notebook
  - Estrutura: Introdução, Metodologia, Resultados, Discussão, Conclusões, Referências

- Incluir todas as visualizações e tabelas relevantes

## 2. Código-Fonte:

- Scripts Python ou Jupyter Notebook bem documentado
- Código organizado, comentado e reproduzível
- Incluir arquivo README com instruções de execução.

## Apresentação Oral:

- Duração: 15-20 minutos
- Slides claros e objetivos
- Demonstração ao vivo do código

## Critérios de Avaliação:

- Clareza e organização do relatório técnico
- Qualidade da documentação do código
- Qualidade visual e estrutura da apresentação
- Capacidade de comunicação e resposta a perguntas

## 5. Critérios de Avaliação

A avaliação seguirá os seguintes critérios:

### 5.1. Coleta e Exploração de Dados (20 pontos)

Critério	Excelente (18-20)	Bom (14-17)	Satisfatório (10-13)	Insuficiente (0-9)
<b>Aquisição e Documentação</b>	Dataset bem escolhido, fonte documentada completamente, justificativa clara da escolha	Dataset adequado, fonte documentada, justificativa presente	Dataset adequado, documentação básica	Dataset inadequado ou documentação ausente
<b>Análise Exploratória</b>	EDA profunda e abrangente, estatísticas relevantes, identificação correta de padrões temporais	EDA adequada, principais estatísticas presentes, identificação de padrões básicos	EDA superficial, estatísticas básicas, poucos padrões identificados	EDA ausente ou inadequada
Critério	Excelente (18-20)	Bom (14-17)	Satisfatório (10-13)	Insuficiente (0-9)
<b>Visualizações</b>	Gráficos claros, informativos e bem formatados, análise completa de tendências e sazonalidade	Gráficos adequados, análise básica de componentes temporais	Gráficos básicos, análise superficial	Visualizações inadequadas ou ausentes

## 5.2. Preparação e Tratamento de Dados (20 pontos)

<b>Critério</b>	<b>Excelente (18-20)</b>	<b>Bom (14-17)</b>	<b>Satisfatório (10-13)</b>	<b>Insuficiente (0-9)</b>
<b>Limpeza de Dados</b>	Tratamento completo e justificado de valores ausentes e outliers, documentação detalhada	Tratamento adequado de valores ausentes, identificação de outliers	Tratamento básico de problemas evidentes	Limpeza inadequada ou ausente
<b>Transformações</b>	Transformações apropriadas e bem justificadas, feature engineering criativo e relevante	Transformações adequadas com justificativa básica	Transformações básicas aplicadas	Transformações inadequadas ou ausentes
<b>Divisão Temporal</b>	Divisão correta respeitando temporalidade, proporções adequadas, validação cruzada implementada	Divisão adequada respeitando temporalidade	Divisão presente mas com problemas menores	Divisão inadequada ou ausente

## 5.3. Modelagem Preditiva (30 pontos)

<b>Critério</b>	<b>Excelente (27-30)</b>	<b>Bom (21-26)</b>	<b>Satisfatório (15-20)</b>	<b>Insuficiente (0-14)</b>
<b>Modelos Baseline</b>	Múltiplos baselines implementados e bem documentados	Pelo menos um baseline implementado corretamente	Baseline básico presente	Baseline ausente ou incorreto
<b>Modelos Avançados</b>	3+ modelos avançados implementados corretamente, variedade de abordagens	2 modelos avançados implementados corretamente	1 modelo avançado implementado	Modelos avançados ausentes ou incorretos
<b>Otimização</b>	Otimização sistemática de hiperparâmetros, processo documentado, múltiplas tentativas	Otimização básica implementada e documentada	Tentativa de otimização presente	Otimização ausente
<b>Validação</b>	Validação cruzada temporal implementada corretamente, análise de estabilidade	Divisão treino/teste adequada	Validação básica presente	Validação inadequada

## 5.4. Avaliação e Comparação (20 pontos)

<b>Critério</b>	<b>Excelente (18-20)</b>	<b>Bom (14-17)</b>	<b>Satisfatório (10-13)</b>	<b>Insuficiente (0-9)</b>
<b>Métricas</b>	Múltiplas métricas calculadas e interpretadas corretamente, análise profunda	Métricas principais presentes e interpretadas	Métricas básicas presentes	Métricas ausentes ou incorretas

<b>Comparação</b>	Análise comparativa profunda, tabelas e gráficos informativos, análise de resíduos	Comparação adequada entre modelos, visualizações presentes	Comparação básica presente	Comparação inadequada ou ausente
<b>Seleção e Justificativa</b>	Escolha bem fundamentada com múltiplos critérios, discussão de limitações	Escolha justificada, limitações mencionadas	Escolha presente com justificativa básica	Escolha não justificada

### 5.5. Apresentação e Documentação (10 pontos)

<b>Critério</b>	<b>Excelente (9-10)</b>	<b>Bom (7-8)</b>	<b>Satisfatório (5-6)</b>	<b>Insuficiente (0-4)</b>
<b>Relatório Técnico</b>	Estrutura profissional, escrita clara, análise completa, referências adequadas	Estrutura adequada, escrita clara, análise presente	Estrutura básica, escrita compreensível	Relatório inadequado ou ausente
<b>Código e Dashboard</b>	Código limpo, bem documentado, reproduzível, boas práticas de programação, dashboard intuitivo e bem construído.	Código funcional, documentação presente, dashboard intuitivo e bem construído.	Código funcional, documentação básica, dashboard intuitivo.	Código não funcional ou mal documentado, ausência de dashboard.
<b>Apresentação Oral</b>	Apresentação clara, profissional, boa comunicação, domínio do conteúdo	Apresentação adequada, boa comunicação	Apresentação básica, comunicação compreensível	Apresentação inadequada

## 6. Entregas Obrigatórias

Os grupos deverão entregar os seguintes materiais

### 6.1. Formato Digital (via plataforma da disciplina)

#### 1. Relatório Técnico em PDF

- Nome do arquivo: N3\_Grupo[X]\_Relatorio.pdf
- Estrutura completa conforme descrito na Etapa 5

#### 2. Código-Fonte Completo

- Formato: Jupyter Notebook (.ipynb) ou scripts Python (.py)
- Nome do arquivo: N3\_Grupo[X]\_Codigo.ipynb, pasta zipada ou repositório do Github.
- Incluir arquivo requirements.txt com dependências
- Incluir arquivo README.md com instruções de execução

#### 3. Apresentação em PDF ou PowerPoint

- Nome do arquivo: N3\_Grupo[X]\_Apresentacao.pdf ou .pptx

#### 4. Dataset Processado (se aplicável)

Nome do arquivo: N3\_Grupo[X]\_Dataset.csv

### 6.2. Apresentação Oral

- Data e horário conforme cronograma divulgado
- Todos os membros do grupo devem participar ativamente da apresentação
- Duração: 15-20 minutos

## 8. Orientações Importantes

### 8.1. Trabalho em Grupo

- Todos os membros devem contribuir ativamente em todas as etapas
- A nota será individual

### 8.2. Integridade Acadêmica

- Todo código deve ser original ou devidamente citado
- Plágio resultará em nota zero para todos os membros do grupo
- É permitido consultar tutoriais e documentação, mas o código deve ser adaptado e compreendido

### 8.3. Uso de Ferramentas

**Linguagem obrigatória:** Python 3.8+

**Bibliotecas recomendadas:**

- Manipulação: pandas, numpy
- Visualização: matplotlib, seaborn, plotly
- Modelagem: statsmodels, scikit-learn, prophet, tensorflow/keras
- Avaliação: sklearn.metrics

### 8.4. Apoio e Dúvidas

Dúvidas podem ser esclarecidas durante as aulas práticas ou por e-mail.

## 9. Critérios de Sucesso

Um projeto excelente deve demonstrar:

- **Rigor Metodológico:** Aplicação correta e sistemática das técnicas aprendidas
- **Criatividade:** Feature engineering criativo e análise aprofundada
- **Clareza:** Comunicação clara dos resultados e processos
- **Reprodutibilidade:** Código organizado e documentado que pode ser executado por terceiros
- **Pensamento Crítico:** Análise crítica dos resultados, discussão de limitações e sugestões de melhorias
- **Aplicabilidade:** Conexão dos resultados com problemas reais

## 11. Recursos Adicionais

### 11.1. Datasets Adicionais

- Monash Time Series Forecasting Repository
- Kaggle Time Series Competitions
- UCI Machine Learning Repository
- CESNET Time Series Dataset