

# Plano de Aula

## Introdução à Ciência de Dados com Python para Administração

---

### ⌚ Contexto [Assistido por IA]

Este plano de aula foi estruturado para alunos da graduação em Administração que precisam compreender como a **Ciência de Dados** apoia decisões estratégicas, operacionais e táticas em organizações. O foco é **80% conceitual** — enfatizando pensamento analítico, entendimento de processos e interpretação de dados — e **20% técnico**, utilizando a linguagem Python de forma introdutória. O curso foi organizado em **16 dias**, cada qual com **4 blocos de 50 min**, incluindo **2 dias (8 blocos)** dedicados exclusivamente a exercícios práticos.

---

### ⌚ Objetivos de [Assistido por IA]

Ao final do curso, o aluno será capaz de:

1. **Explicar** o ciclo de vida da Ciência de Dados e seu papel na tomada de decisão organizacional.
  2. **Identificar** problemas administrativos que podem ser resolvidos com análise de dados.
  3. **Interpretar e avaliar** métricas básicas de análise estatística e exploratória de dados.
  4. **Utilizar** Python para carregar, limpar e visualizar conjuntos de dados simples.
  5. **Aplicar** conceitos de modelagem básica, como regressão linear, em contextos administrativos.
  6. **Comunicar** resultados de análises de maneira clara e orientada ao negócio.
- 

### ⌚ Habilidades/Competências Desenvolvidas [Assistido por IA]

- Pensamento analítico e resolução de problemas.
  - Raciocínio lógico aplicado a negócios.
  - Interpretação de dados para tomada de decisão.
  - Iniciação em Ciência de Dados (Data Literacy).
  - Fundamentos de programação em Python.
  - Uso de IA generativa como apoio à aprendizagem e análise exploratória.
-

## Conteúdo Programático (16 dias) [Assistido por IA]

- **Dias 1–4: Fundamentos e contextualização (conceitual)**
  1. O que é Ciência de Dados; impacto no mundo dos negócios.
  2. Pensamento estatístico e tipos de dados.
  3. Ciclo de vida do projeto de dados (definição → coleta → limpeza → análise → apresentação).
  4. Introdução à linguagem Python (ambiente, sintaxe, variáveis, funções).
- **Dias 5–8: Manipulação e exploração de dados (conceitual + técnico)**
  5. Introdução a NumPy e Pandas.
  6. Importação e limpeza básica de dados.
  7. Exploração inicial de datasets administrativos.
  8. Visualização de dados com bibliotecas básicas (matplotlib / pandas).
- **Dias 9–12: Modelagem e aplicações (conceitual)**
  9. Tipos de modelos preditivos (supervisionado x não supervisionado).
  10. Regressão linear e problemas de previsão em Administração.
  11. Métricas de avaliação de modelos.
  12. Riscos, limitações e ética em Ciência de Dados.
- **Dias 13–14: Aulas práticas (exercícios de fixação)**
  - Manipulação, análise e visualização de dados administrativos.
  - Mini-projetos orientados (grupos).
- **Dias 15–16: Aplicações integradas e apresentação de mini-projeto**
  - Desenvolvimento de uma análise aplicada ao contexto da Administração.
  - Comunicação dos resultados em linguagem executiva.

---

## Metodologia Passo a Passo [Assistido por IA]

Cada dia possui **4 blocos de 50 minutos (200 min)**. Abaixo, a estrutura geral recomendada.

- **◊ Bloco 1 (50 min) — Exposição dialogada**
  - 10 min: Abertura e retomada do dia anterior
  - 25 min: Apresentação conceitual do tópico
  - 15 min: Discussão de casos reais

- ◇ **Bloco 2 (50 min) — Demonstração guiada**
  - 10 min: Introdução prática do tema
  - 30 min: Demonstração em Python (com IA como tutor auxiliar)
  - 10 min: Perguntas orientadas
- ◇ **Bloco 3 (50 min) — Atividade orientada**
  - 10 min: Distribuição dos exercícios
  - 30 min: Realização individual ou em duplas
  - 10 min: Correção/explicação
- ◇ **Bloco 4 (50 min) — Fechamento e síntese**
  - 20 min: Debate sobre aplicações administrativas
  - 20 min: Registro dos aprendizados (learning journal)
  - 10 min: Preparação para o próximo encontro
-  **Aulas práticas (Dias 13–14)**
  - 20 min: Revisão dos conteúdos técnicos
  - 140 min: Exercícios/mini-projeto
  - 40 min: Apresentação parcial + feedback
-  **Como a IA entra na metodologia**
  - Apoio à explicação de conceitos com exemplos gerados sob demanda.
  - Sugestão automática de códigos em Python em ambiente controlado.
  - Geração de visualizações rápidas.
  - Ferramenta de revisão de raciocínio (“explique como cheguei a este resultado”).
  - Comparação de soluções múltiplas para o mesmo problema.

---

## **Recursos [Assistido por IA]**

- Sala com projetor ou TV interativa.
  - Ambiente Python (Jupyter Notebook, Google Colab).
  - Conjunto de dados administrativos (vendas, RH, logística, marketing).
  - Livros-base indicados (Grus, Downey, McKinney).
  - IA generativa como tutor auxiliar nas dúvidas e exemplos.
-



## Avaliação [Assistido por IA]

- **Formas de avaliação**
  - Exercícios práticos.
  - Mini-projeto final.
  - Participação e engajamento em discussões.
  - Relatório executivo explicando descobertas e decisões geradas pelos dados.
- **Rubrica de avaliação**
- **Rubrica (Critérios × Desempenho)**

Critério	Excelente	Bom	A melhorar
Compreensão conceitual	<i>Explica com clareza e profundidade os conceitos</i>	<i>Explica os conceitos com pequenas limitações</i>	<i>Explicações superficiais ou incorretas</i>
Aplicação prática	<i>Utiliza Python de forma precisa e coerente</i>	<i>Aplica Python com pequenos erros</i>	<i>Dificuldade significativa na aplicação</i>
Análise e interpretação	<i>Interpreta dados e resultados com rigor analítico</i>	<i>Interpreta com alguma imprecisão</i>	<i>Interpretações inadequadas ou incompletas</i>
Clareza na comunicação	<i>Relatório e apresentação objetivos e estruturados</i>	<i>Boa comunicação com pequenas falhas de organização</i>	<i>Comunicação confusa ou pouco estruturada</i>
Qualidade do mini-projeto	<i>Projeto completo, bem executado e contextualizado</i>	<i>Projeto adequado, porém com lacunas menores</i>	<i>Projeto incompleto ou pouco alinhado ao problema</i>

Recomendação para inclusão:

- **Aprendizagem multisserviço:** disponibilização de materiais em texto, vídeo e exemplos práticos.
- **Apoio individual com IA:** alunos com menor familiaridade em programação podem solicitar explicações passo a passo em linguagem simplificada.
- **Desafios opcionais:** alunos mais avançados podem aprofundar análises com datasets maiores ou técnicas adicionais.
- **Flexibilização de prazos curtos:** para estudantes com dificuldades momentâneas ou necessidades educacionais específicas.
- **Uso de linguagem clara:** evitar jargões técnicos sem explicação prévia.

---

## Referências [Assistido por IA]

- **Grus, Joel.** *Data Science do Zero – Noções Fundamentais com o Python*. Editora Aça Books.
  - **Downey, Allen B.** *Pense em Python: Pense como um cientista da computação*. Editora Novatec.
  - **McKinney, Wes.** *Python para Análise de Dados: Tratamento de Dados com Pandas, NumPy e IPython*. Editora Novatec.
-

## **Lesson Plan**

### Introduction to Data Science with Python for Business Administration

#### Context [Supported by AI]

This lesson plan is for undergraduate Business Administration students who need to understand how Data Science helps with strategic, operational, and tactical decisions in organizations. The focus is 80% on concepts—thinking analytically, understanding processes, and interpreting data—and 20% technical, using Python in a simple way. The course is organized in 16 days, each with 4 blocks of 50 minutes. Two days (8 blocks) are only for practical exercises.

#### Objectives [Supported by AI]

At the end of the course, students will be able to:

- Explain the Data Science life cycle and its role in decision-making in organizations.
- Identify business problems that can be solved with data analysis.
- Interpret and evaluate basic statistics and data exploration.
- Use Python to load, clean, and show simple data sets.
- Apply basic modeling concepts, like linear regression, in business situations.
- Share analysis results clearly and with a business focus.

#### Skills/Competencies Developed [Supported by AI]

- Analytical thinking and problem solving.
- Logical reasoning for business.
- Data interpretation for decision-making.
- Introduction to Data Science (Data Literacy).
- Basic Python programming.
- Using generative AI to help learning and data analysis.

#### Course Content (16 days) [Supported by AI]

##### Days 1–4: Fundamentals and context (conceptual)

- What is Data Science? Impact in business.
- Statistical thinking and types of data.
- Data project life cycle (define → collect → clean → analyze → present).
- Introduction to Python (environment, syntax, variables, functions).

##### Days 5–8: Data handling and exploration (conceptual + technical)

- Introduction to NumPy and Pandas.
- Importing and basic cleaning of data.
- First exploration of business datasets.
- Data visualization with basic libraries (matplotlib / pandas).

##### Days 9–12: Modeling and applications (conceptual)

- Types of predictive models (supervised vs. unsupervised).
- Linear regression and forecasting problems in Business Administration.

- Model evaluation metrics.
- Risks, limits, and ethics in Data Science.

Days 13–14: Practice classes (reinforcement exercises)

- Handling, analyzing, and visualizing business data.
- Guided mini-projects (groups).

Days 15–16: Integrated applications and mini-project presentation

- Developing an analysis for a business context.
- Presenting results in executive language.

### Step-by-Step Methodology [Supported by AI]

Each day has 4 blocks of 50 minutes (200 min). The general structure is below.

- Block 1 (50 min) — Dialogued presentation
  - 10 min: Opening and review of the last class
  - 25 min: Presenting the concept
  - 15 min: Discussion of real cases
- Block 2 (50 min) — Guided demonstration
  - 10 min: Practical introduction
  - 30 min: Demonstration in Python (AI as assistant)
  - 10 min: Guided questions
- Block 3 (50 min) — Guided activity
  - 10 min: Giving exercises
  - 30 min: Doing exercises alone or in pairs
  - 10 min: Correction/explanation
- Block 4 (50 min) — Closing and summary
  - 20 min: Debate about business applications
  - 20 min: Record learnings (learning journal)
  - 10 min: Prepare for next class
- Practice classes (Days 13–14)
  - 20 min: Review technical content
  - 140 min: Exercises/mini-project
  - 40 min: Partial presentation + feedback

How AI is used in the methodology:

- Helping explain concepts with real-time examples.
- Giving Python code suggestions in a controlled environment.
- Creating quick visualizations.
- Reasoning review tool (“explain how I got this result”).
- Comparing different solutions for the same problem.

### Resources [Supported by AI]

- Classroom with projector or interactive TV.
- Python environment (Jupyter Notebook, Google Colab).
- Business datasets (sales, HR, logistics, marketing).
- Recommended books (Grus, Downey, McKinney).
- Generative AI as an assistant for questions and examples.

### Assessment [Supported by AI]

Ways to assess:

- Practical exercises.
- Final mini-project.
- Participation and engagement in discussions.
- Executive report explaining findings and decisions from data.
- 

### Assessment Rubric

Criteria	Excellent	Good	Needs Improvement
Concept understanding	Explains concepts clearly and deeply	Explains concepts with small mistakes	Superficial or incorrect explanations
Practical application	Uses Python precisely and correctly	Uses Python with small errors	Has trouble using Python
Analysis and interpretation	Interprets data and results with good analysis	Interprets with some mistakes	Poor or incomplete interpretation
Clear communication	Report and presentation are clear and well structured	Good communication with some organization problems	Confusing or poorly structured communication
Mini-project quality	Project is complete, well done and fits the problem	Project is good but has some minor gaps	Incomplete or not related to the problem

Recommendation for inclusion:

- Multi-service learning: offer materials in text, video, and practical examples.
- AI individual support: students less familiar with programming can get step-by-step explanations in simple language.
- Optional challenges: advanced students can explore larger datasets or more techniques.
- Flexible short deadlines: for students with temporary difficulties or special needs.
- Use clear language: avoid technical words without first explaining them.



## References

- Grus, Joel. Data Science from Scratch – Fundamental Concepts with Python.  
Açta Books.
- Downey, Allen B. Think Python: How to Think Like a Computer Scientist.  
Novatec.
- McKinney, Wes. Python for Data Analysis: Data Treatment with Pandas, NumPy  
and IPython. Novatec.