

# *Introdução a Ciência de Dados*



Professor: Alex Pereira

# *Escopo desta Disciplina*

- Story Telling com dados
  - Teoria, e
  - Prática
    - ✓ Com Plotnine e D3
- Construção de Dashboards (Paineis)
  - Teoria, e
  - Prática
    - ✓ Com Google Looker Studio
- Coleta de Dados
  - Webscraping
- Manipulação e Validação de Dados
  - Com pandas
- Armazenamento dos dados no BigQuery (Google)

# *A escolha das Ferramentas, Aplicações e trade-offs*

- Custo de Transação, é o custo para
  - **realizar qualquer negociação econômica ao participar de um mercado**
    - ✓ custo/tempo de planejamento, decisão, mudança de planos, resolução de disputas, transporte/entrega/forma de consumo, pós-venda, taxas de pagamento, comissões
- Tendências do mercado de TI
  - **Zerar o custo de licença e diminuir os custos de transação da adesão**
    - ✓ Gmail, Google Drive, Google Analytics, Google Looker Studio, Redes Sociais
      - Extrair lucro (propaganda) a partir do acesso aos dados
        - Quando a licença é gratuita o produto é você.
  - **Free-tier (Amostra grátis)**
    - ✓ Google Cloud Platform (Bigquery e outros), AWS (Redshift e outros)
- Consequências para a Administração Pública
  - **Celeridade e economicidade**
    - ✓ Ao custo de compartilhar os dados da administração pública

# *As ferramentas deste curso*

- Google Looker Studio
  - Sem custo de licença
  - Integração facilitada do com o BigQuery, Google Sheets e Analytics
  - Infraestrutura gerenciada pelo Google
    - ✓ Única solução de Dashboard as a Service (DaaS)
      - sem custo de licença em que se pode publicar abertamente um painel
  - Baixo custo de transação de adesão
- Google Colab
- Ferramentas de IA Generativa
  - ChatGPT, Perplexity, Gemini, etc



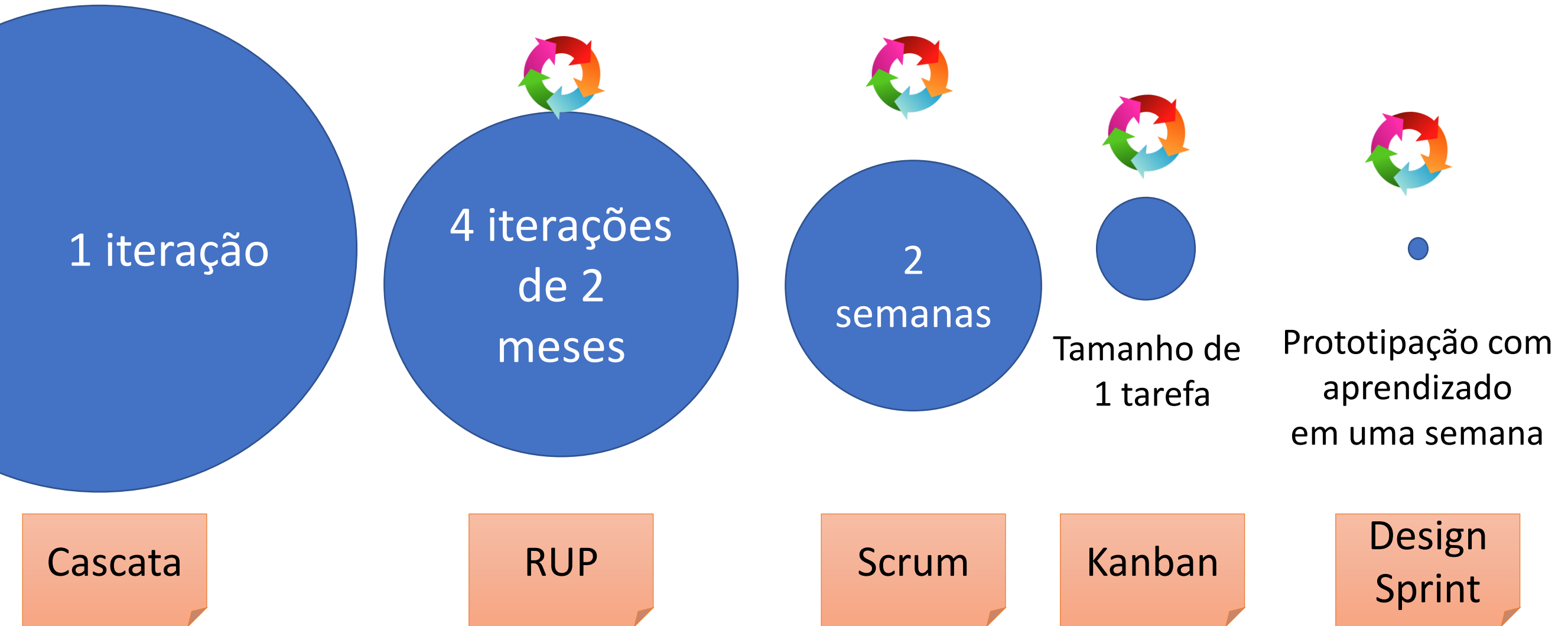
# *Contar uma história com dados*

- Habilidade fundamental, de importância crescente
  - + dados digitais
  - Potencializa sua capacidade de
    - ✓ argumentação e convencimento
- Formação de agenda de políticas públicas
  - Teoria da Lata do Lixo (Garbage Can)
    - ✓ Uma coleção de escolhas encontra um problema
      - Entidades envolvidas: soluções, problemas e tomadores de decisão
    - ✓ Nesse contexto, **publicar** suas soluções na forma de **histórias com dados**
      - Pode aumentar a probabilidade de soluções encontrarem os problemas
        - Ferramentas com baixo custo de transação proporcionam esse benefício
  - O papel do cientista de dados
    - ✓ Descobrir e informar



# ***Carga Cognitiva e 2 maneiras de usar a IA***

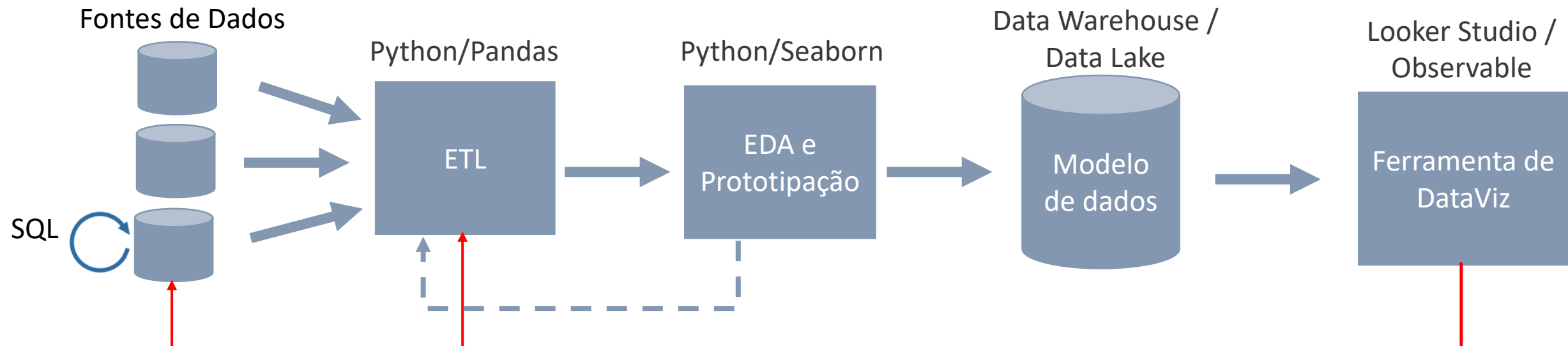
# *Tamanho decrescente das iterações nas metodologias de gestão*



- Feedback antecipado e frequente

# Sugestão de Metodologia de Trabalho

- Para elaborar painéis ou histórias com dados
  - Defina um problema e seu público alvo,
    - ✓ estude as variáveis disponíveis aplique uma técnica de ideação
- Use uma metodologia iterativa de construção do modelo de dados
  - Minimize o tamanho das iterações
    - ✓ Iterações são inevitáveis e esperadas, **minimize o custo para iterar!**
      - Erro comum: validar as transformações somente na última etapa.

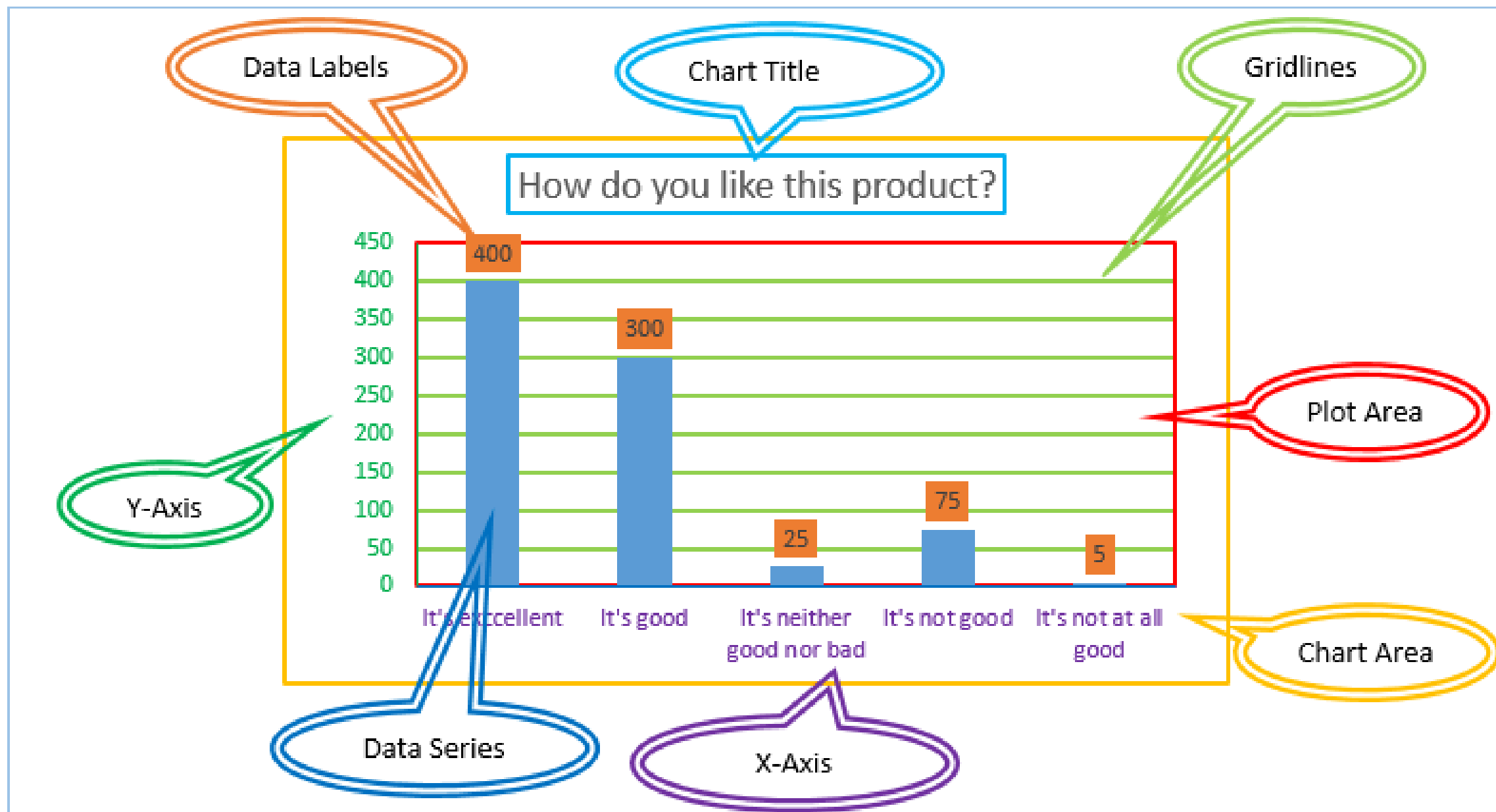




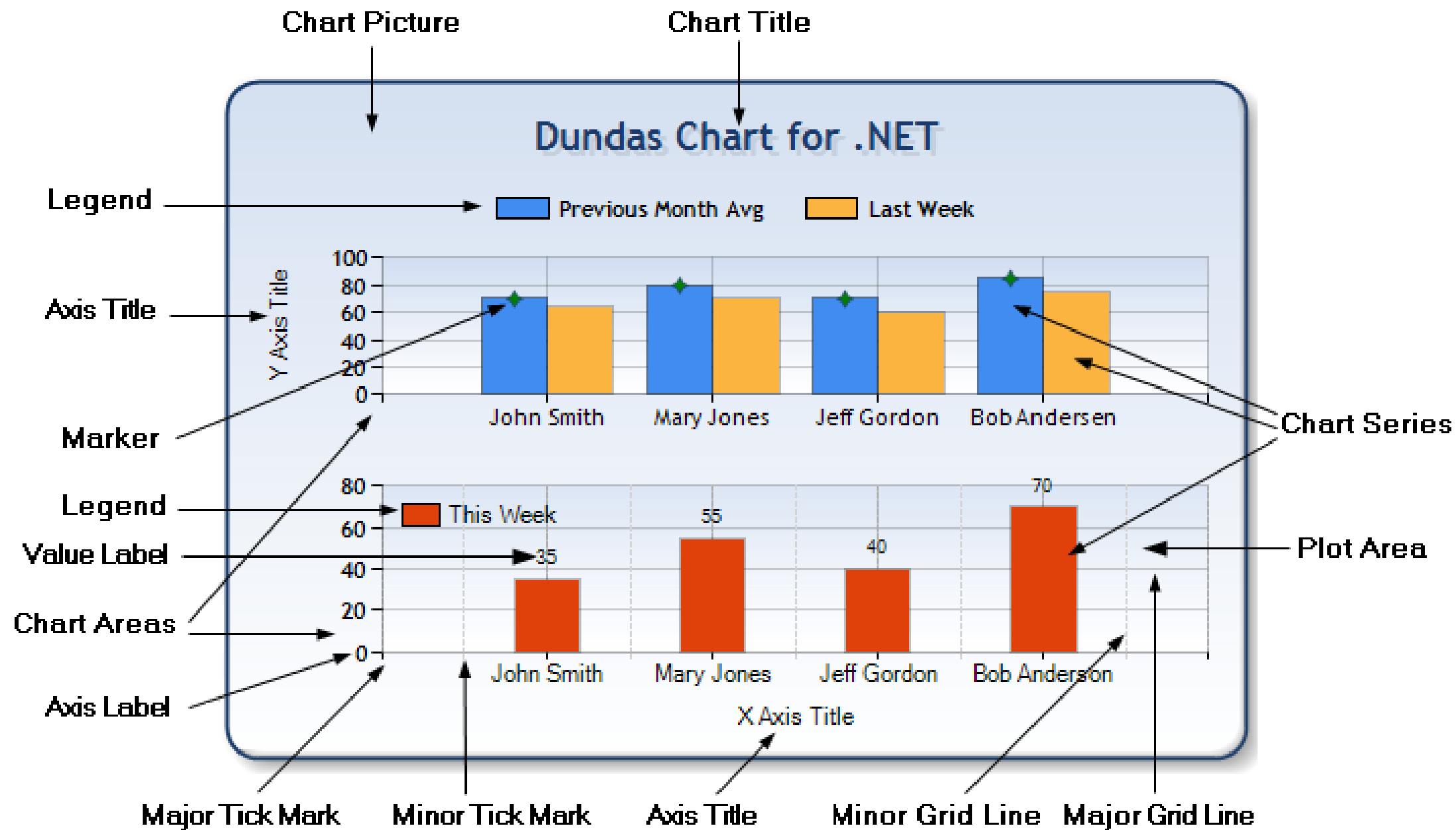
# *Analise de dados na Perplexity Labs e Manus.ai*

- Baixo esforço para obter uma análise bem preliminar
- Exemplos
  - Perplexity Labs
  - [Manus.ai](#)

# *Nomenclatura dos elementos de um gráfico*



# *Nomeclatura dos elementos de um gráfico*



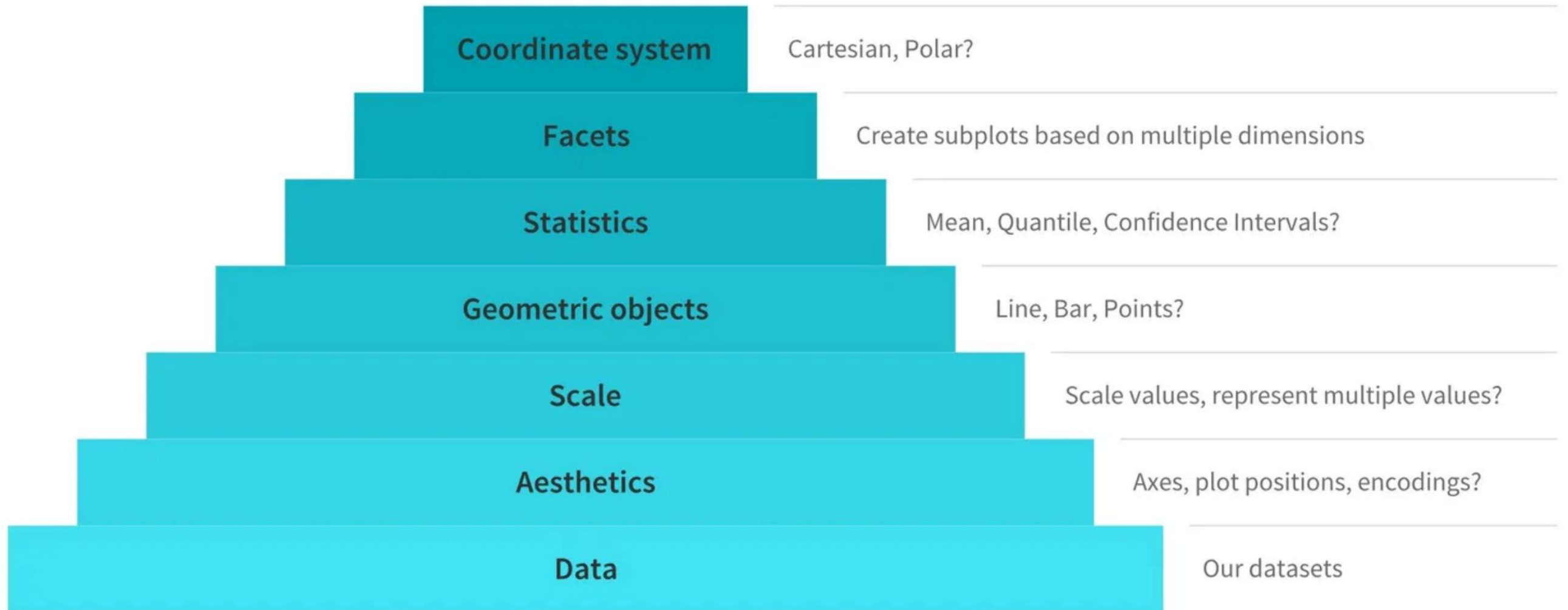
# ***Gráficos no Colab com IA***

# *Grammar of Graphics*

- Gramática
  - um conjunto de regras que regem o uso de uma língua
- Grammar of Graphics (Leland Wilkinson)
  - é uma ferramenta que nos permite descrever concisamente os componentes de um gráfico;
  - Separa os dados da sua representação visual.
- Mudança de mindset
  - Em vez de uma função para criar um gráfico de barras
    - ✓ Usa-se uma função para mapear (encoding) de variáveis nos eixos, exemplo:
      - x: variável categórica (ex.: empresas A, B e C)
      - y: variável contínua (ex.: faturamento)
    - ✓ Geometria: barra
  - Construir gráficos a partir de suas partes elementares (building blocks)

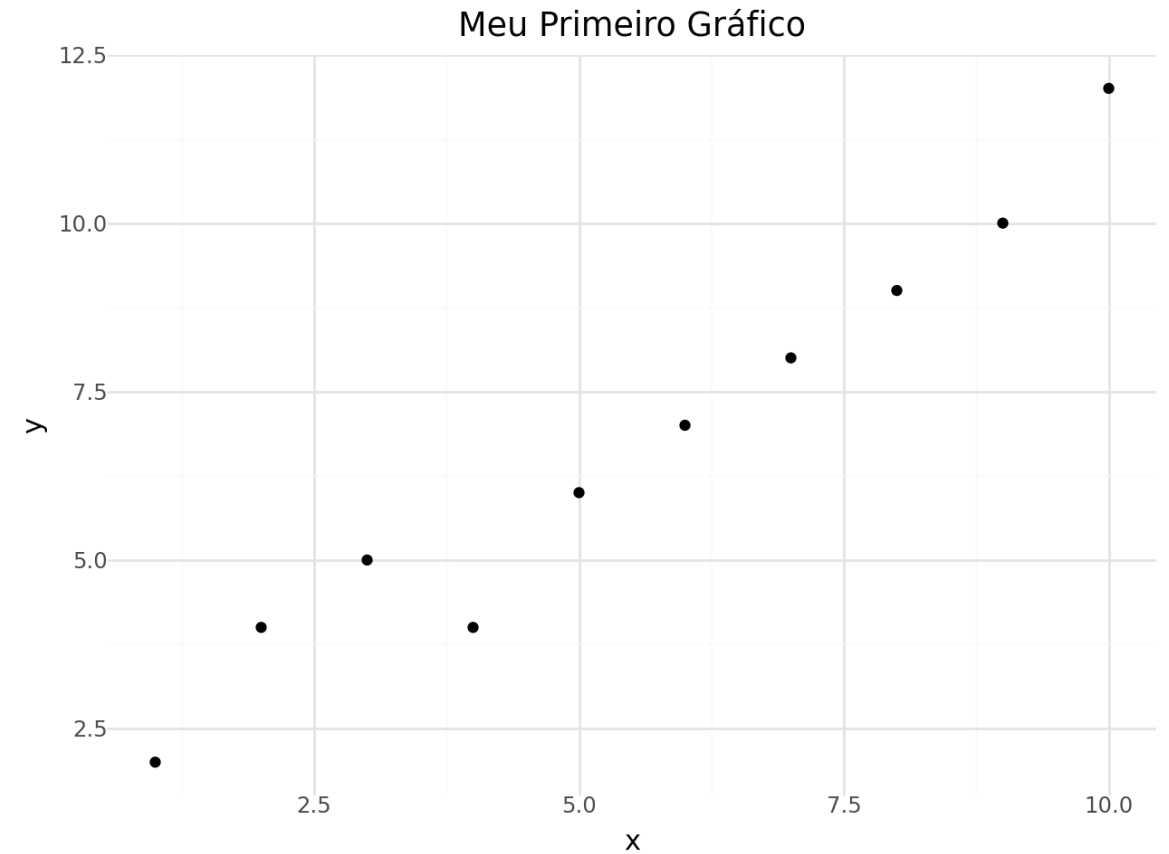


# *Grammar of Graphics: componentes*



# *Plotnine: GoG no Python*

```
from plotnine import ggplot, aes, geom_point,  
labs, theme_minimal  
import pandas as pd  
  
df = pd.DataFrame({  
    'x': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],  
    'y': [2, 4, 5, 4, 6, 7, 8, 9, 10, 12]  
})  
  
(  
    ggplot(df, aes(x='x', y='y'))  
    + geom_point()  
    + labs(title='Meu Primeiro Gráfico')  
    + theme_minimal()  
)
```



# *O que são esses components (no Plotnine)*

## 1. Dados (Data)

A fonte de dados, geralmente um DataFrame do Pandas.

## 2. Estéticas (Aesthetics)

Mapeamento de variáveis para propriedades visuais (x, y, cor, tamanho).

## 3. Geometrias (Geoms)

Formas geométricas que representam os dados (pontos, linhas, barras).

## 4. Escalas (Scales)

Controlam como os dados são mapeados para as propriedades visuais.

## 5. Facetas (Facets)

Divisão do gráfico em múltiplos painéis baseados em variáveis.

## 6. Estatísticas (Stats)

Transformações estatísticas aplicadas aos dados.

## 7. Coordenadas (Coordinates)

Sistema de coordenadas para posicionar os elementos.



# Tipos de Gráficos Comuns

## 1. Gráfico de Dispersão (Scatter Plot)

Mostra a relação entre duas variáveis numéricas.

```
geom_point()
```

## 2. Gráfico de Barras (Bar Chart)

Compara valores entre diferentes categorias.

```
geom_bar(stat='identity')
```

## 3. Gráfico de Linhas (Line Plot)

Mostra tendências ao longo do tempo ou sequência.

```
geom_line()
```

## 4. Histograma

Mostra a distribuição de uma variável numérica.

```
geom_histogram()
```

## 5. Box Plot

Mostra a distribuição estatística dos dados.

```
geom_boxplot()
```

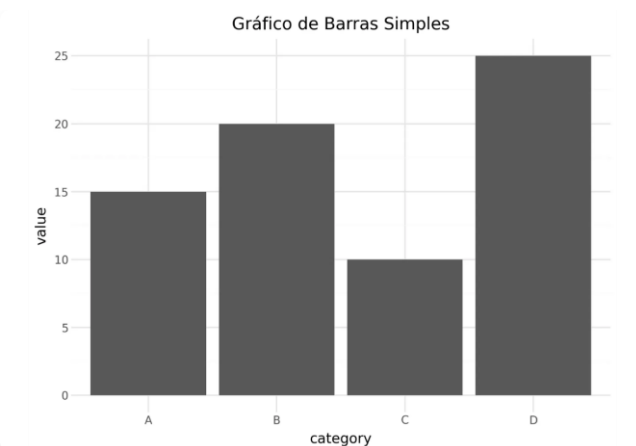


Gráfico de Barras

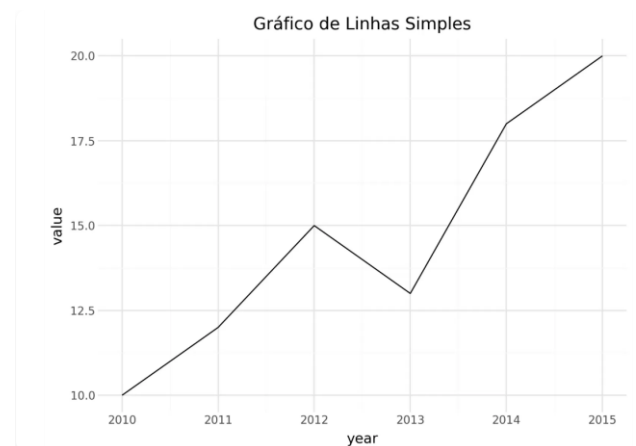
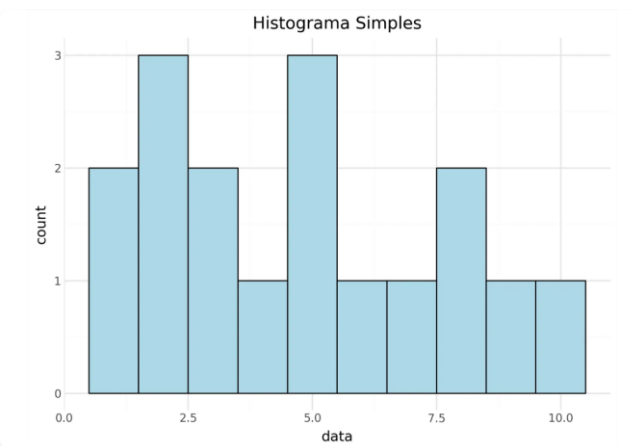
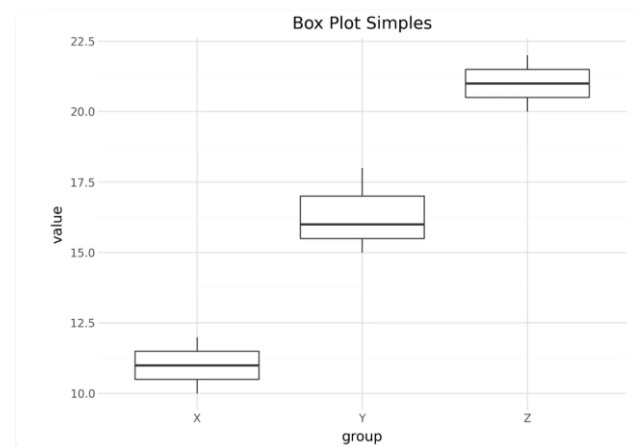


Gráfico de Linhas



Histograma



Box Plot

# Personalização e Refinamento

## H Títulos e Rótulos

Adicione títulos, subtítulos e rótulos aos eixos.

```
+ labs(title='Título', x='Eixo X', y='Eixo Y')
```

## 🎨 Cores e Temas

Personalize as cores e aplique temas predefinidos.

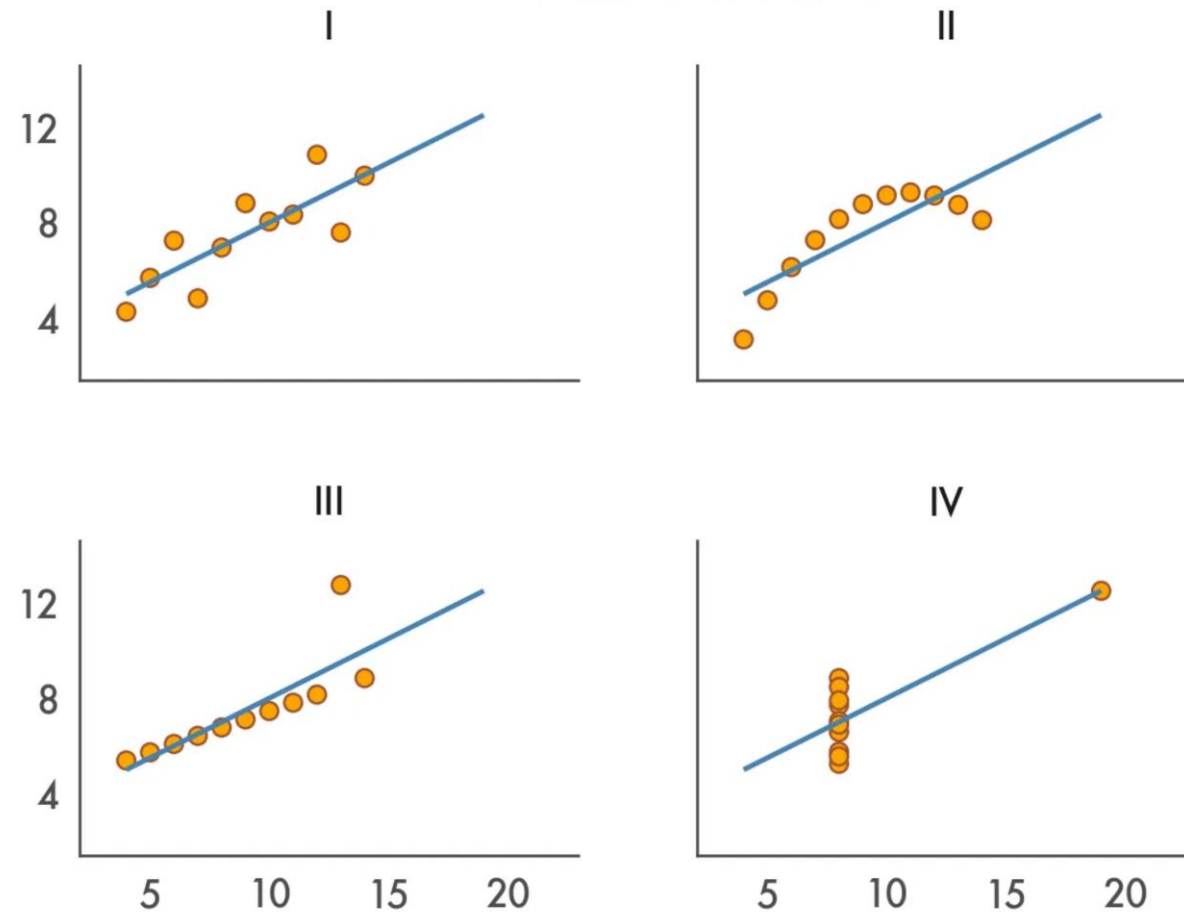
```
+ scale_color_brewer(palette='Set1')  
+ theme_minimal()
```

## 📊 Múltiplos Gráficos (Faceting)

Divida seu gráfico em múltiplos painéis.

```
+ facet_wrap('~categoria')
```

## Anscombe's Quartet





# Recursos Adicionais

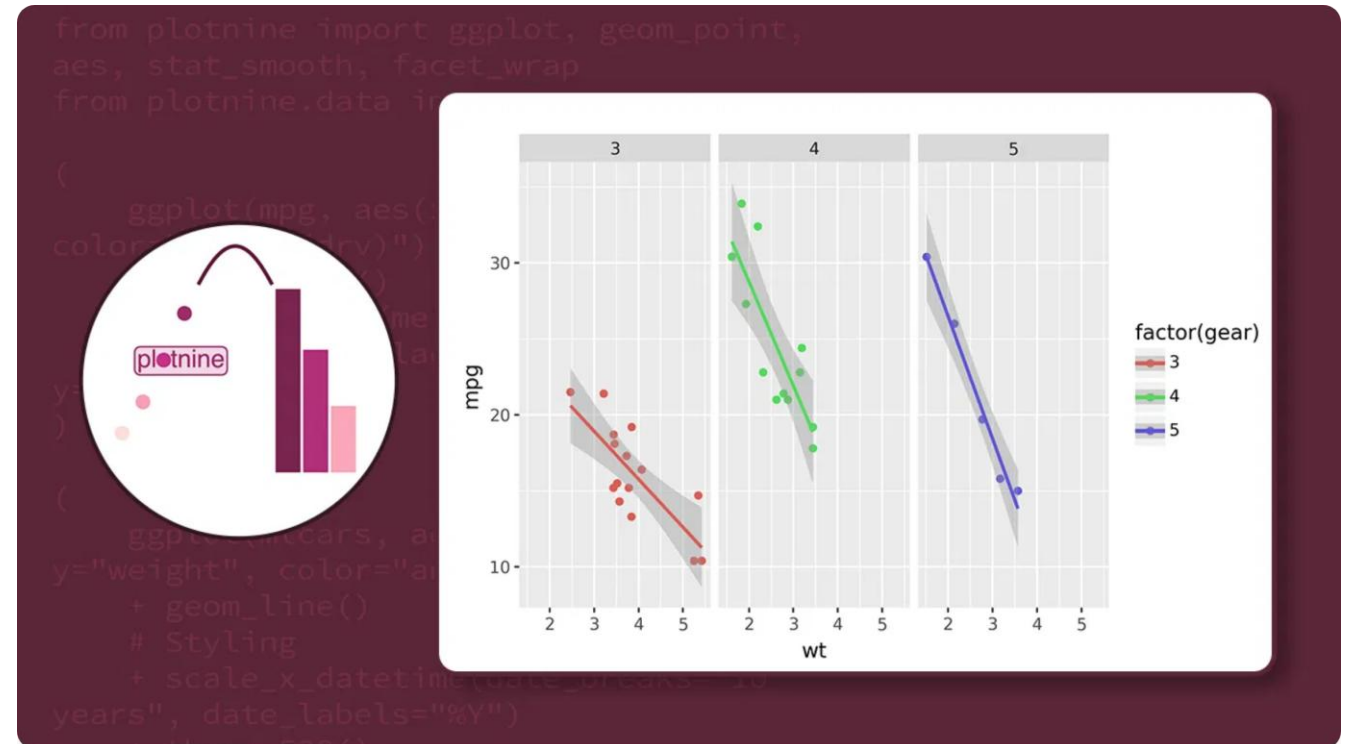
## Benefícios do Plotnine

- ✓ Sintaxe consistente e declarativa
- ✓ Integração perfeita com o ecossistema Python
- ✓ Facilidade para criar visualizações complexas
- ✓ Ideal para análise exploratória de dados

## Recursos Adicionais

 [Documentação Oficial do Plotnine](#)

 [Galeria de Exemplos](#)



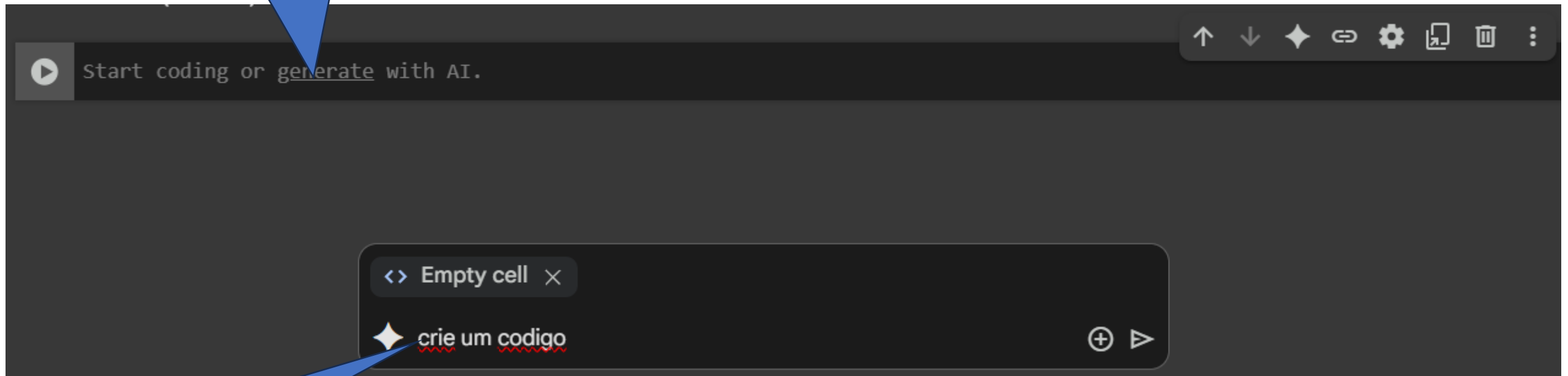
# *Aula Interativa*

- É uma aula demonstrativa
  - A aula prática com tempo individual para resolver os exercícios
    - será depois
- Interaja respondendo as perguntas do Professor
  - Não tente resolver os exercícios no seu PC
    - Se preciso "amarre-se ao mastro do navio"
      - como Ulisses
- Atenção dividida
  - Resolva uma multiplicação e um Soma 1



# *Refatoração de Código no Gemini no Google Colab*

1. Selecionar o código



2. Digitar seu prompt e clicar em Accept

# Refatoração de Código no Gemini no Google Colab

0s

```
from plotnine import (  
    ggplot,  
    aes,  
    geom_bar,  
    labs, position_fill  
)  
  
plot = (  
    ggplot(df, aes(x='sigla_uf', y='naotomou2aDose', fill='vacina'))  
    + geom_bar(stat='identity')  
    + labs(title='Pessoas sem 2a dose por UF e Vacina', x='UF', y='Número de pessoas', fill='Vacina')  
)  
  
plot
```

1. Selecionar o código

2. Clicar na Estrela -> Transform code

Generate code  
Explain code  
Transform code

Pessoas sem 2a dose por UF e Vacina

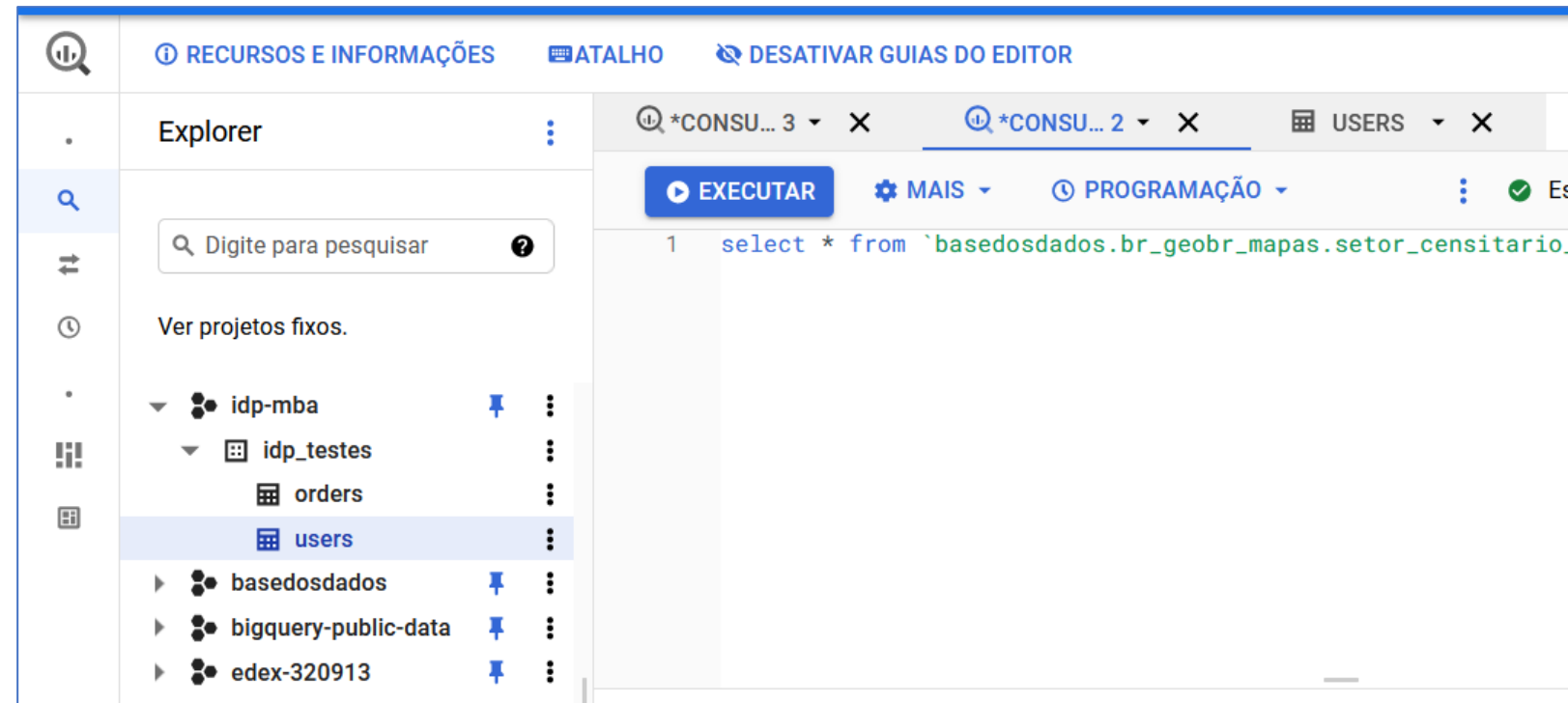
from plotnine import (

crie um código

3. Digitar seu prompt e clicar em Accept

# BigQuery

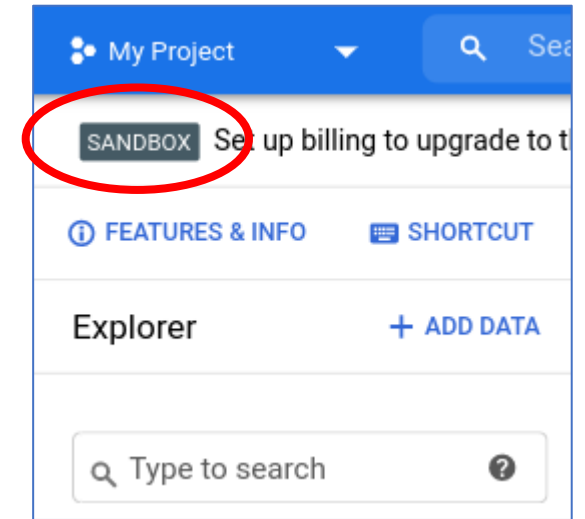
- Data Warehouse
  - é um tipo de sistema de gerenciamento de dados projetado para ativar e fornecer suporte às atividades de Business Intelligence (BI).
- Solução Google de Armazenamento de Dados (Data Warehouse)
  - Gerenciado
  - Em escala de petabyte
  - Baixo Custo





# BigQuery Sandbox

- Sandbox para teste sem cartão de crédito
  - <https://www.youtube.com/watch?v=JLXLCv5nUCE>
  - ✓ <https://console.cloud.google.com/bigquery>
  - Se não houver um projeto, crie um.
  - Query: 1TB/mês, Storage: 10TB/mês, tabelas expiram em 60 dias
- Sem billing account ativada
  - O ícone do sandbox aparece
    - ✓ Crie uma conta google ou encerre a billing account
      - Tutorial para encerrar billing account
        - <https://cloud.google.com/billing/docs/how-to/manage-billing-account>
- **Use o Sandbox!!**
  - Nem que pra isso seja necessário criar outra conta Google.



# Base dos Dados – Projeto Open Source

- Acessar dados no BigQuery
  - Explicações em <https://basedosdados.org>
    - ✓ Acesso direto
      - <https://console.cloud.google.com/bigquery?p=basedosdados&page=project>
  - Projeto cuja missão é:
    - Organizar e mapear bases de dados brasileiras e internacionais.



# *SQL é o sabre de luz de um Cientista de Dados*



# *Algumas Queries no BigQuery*

# *Algumas Queries Simples*

- Selecionar empresas da base de CNPJ

- `SELECT * FROM `basedosdados.br_me_cnpj.empresas` where data="2024-12-18" LIMIT 1000`

- Selecionar licitações

- `SELECT * FROM `basedosdados.br_cgu_licitacao_contrato.licitacao` LIMIT 10`



## *Query de pessoas que não tomaram a 2ª dose da vacina do COVID-19 (1)*

- Selecionar **pessoas** que ainda não tomaram a 2ª dose
  - Esta é uma tabela de transação
    - ✓ e não de entidades

```
SELECT distinct(v.id_paciente)
FROM `basedosdados.br_ms_vacinacao_covid19.microdados_vacinacao` v
WHERE v.vacina != '88'
GROUP BY v.id_paciente
HAVING min(v.data_aplicacao) = max(v.data_aplicacao)
LIMIT 10
```

\* O LIMIT é opcional (recomendável) para testes

# *Query de pessoas que não tomaram a 2ª dose da vacina do COVID-19 e estão atrasadas*

```
select sum( CASE
    WHEN vagg.vacina='86' THEN
        case when (DATE_DIFF(current_date, vagg.data_aplicacao, day) > 30) then 1 else 0 end
    ELSE
        case when (DATE_DIFF(current_date, vagg.data_aplicacao, day) > 90) then 1 else 0 end
    END ) AS naotomou2aDose
    , vagg.vacina
from `basedosdados.br_ms_vacinacao_covid19.microdados_vacinacao` vagg
where vagg.id_paciente in
( SELECT distinct(v.id_paciente)
  FROM `basedosdados.br_ms_vacinacao_covid19.microdados_vacinacao` v
  WHERE v.vacina != '88'
  GROUP BY v.id_paciente
  HAVING min(v.data_aplicacao) = max(v.data_aplicacao) )
group by vacina;
```

## *Links Importantes*

- Google Colab (Notebooks)
  - <https://colab.research.google.com> (Conta no Google/Gmail)
- Repositório da Disciplina no Github
  - [https://github.com/alexlopespereira/mba\\_enap/tree/main/CD](https://github.com/alexlopespereira/mba_enap/tree/main/CD)

# ***Datas e Horários***

- Horários

- 2as e 6as de Manhã

- ✓ 9h00 as 12h00

- 4as a Noite

- ✓ 19h as 22h

# *Material de Estudo*

- [Slides](#) de Teoria
  - Com demonstrações em formato vídeo
- Cadernos Colab
  - [Teoria](#)
  - [Exercícios e Atividades](#)
- Vídeos das Aulas
  - Disponíveis no ambiente de aprendizagem virtual da Enap

# Avaliação

- Até 3 Exercícios por semana
  - Exercícios atrasados valem 60% da nota.
  - Entregues até domingo 8h00 AM
- Convenção deste curso: Exercícios valem nota, Atividades não.
- O exercício da penúltima semana será
  - De tema aberto (à conveniência dos alunos)
- Fórmula de Cálculo
  - $$\text{Nota} = \sum_{i=1}^K \frac{\alpha \times N_i}{K}, \alpha = 0.6 \text{ (se em atraso)}, K = \text{numero de exerc\u00edcios}$$

# *Referências Bibliográficas*

- KNAFLIC, C. N. (2018). Storytelling with data: a data visualization guide for business professionals.
- McKinney, W. (2018). Python for data analysis: Data wrangling with pandas, NumPy, and IPython.
- HURST, L. (2020). Hands on with Google Looker Studio: a data citizen's survival guide.  
<https://onlinelibrary.wiley.com/doi/book/10.1002/9781119616238>.
- WEXLER, S., SHAFFER, J., & COTGREAVE, A. (2017). The big book of dashboards: visualizing your data using real-world business scenarios.
- <https://www.storytellingwithdata.com/podcast>
- <https://seaborn.pydata.org/>
- <https://pandas.pydata.org/docs/>
- <https://d3js.org>
- <https://numpy.org/doc/stable/reference/index.html>