

Introduction to Data Science

MODULE II – PART II

Data Cleaning & Exploration

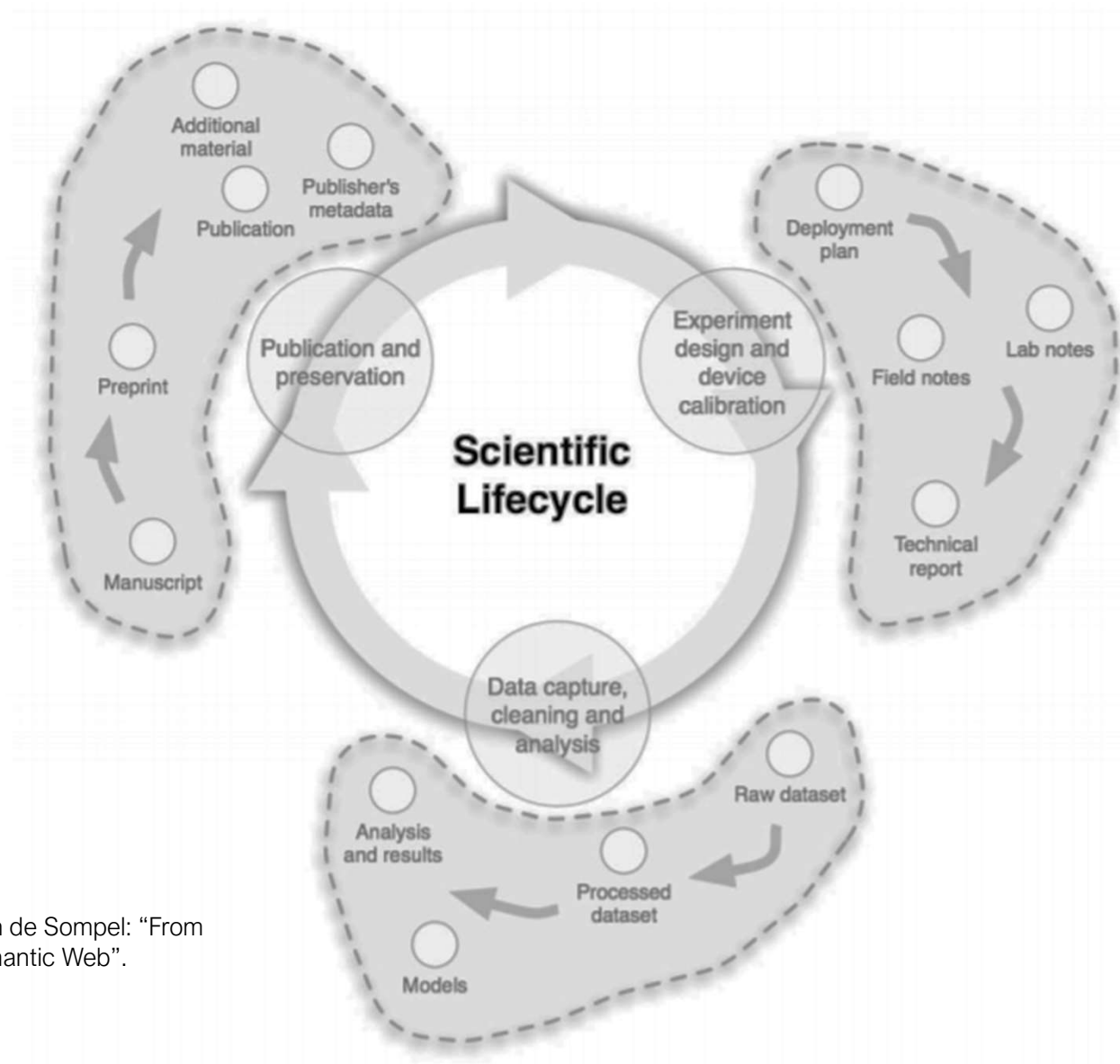
Prof Sergio Serra e Jorge Zavaleta

Scientific lifecycle

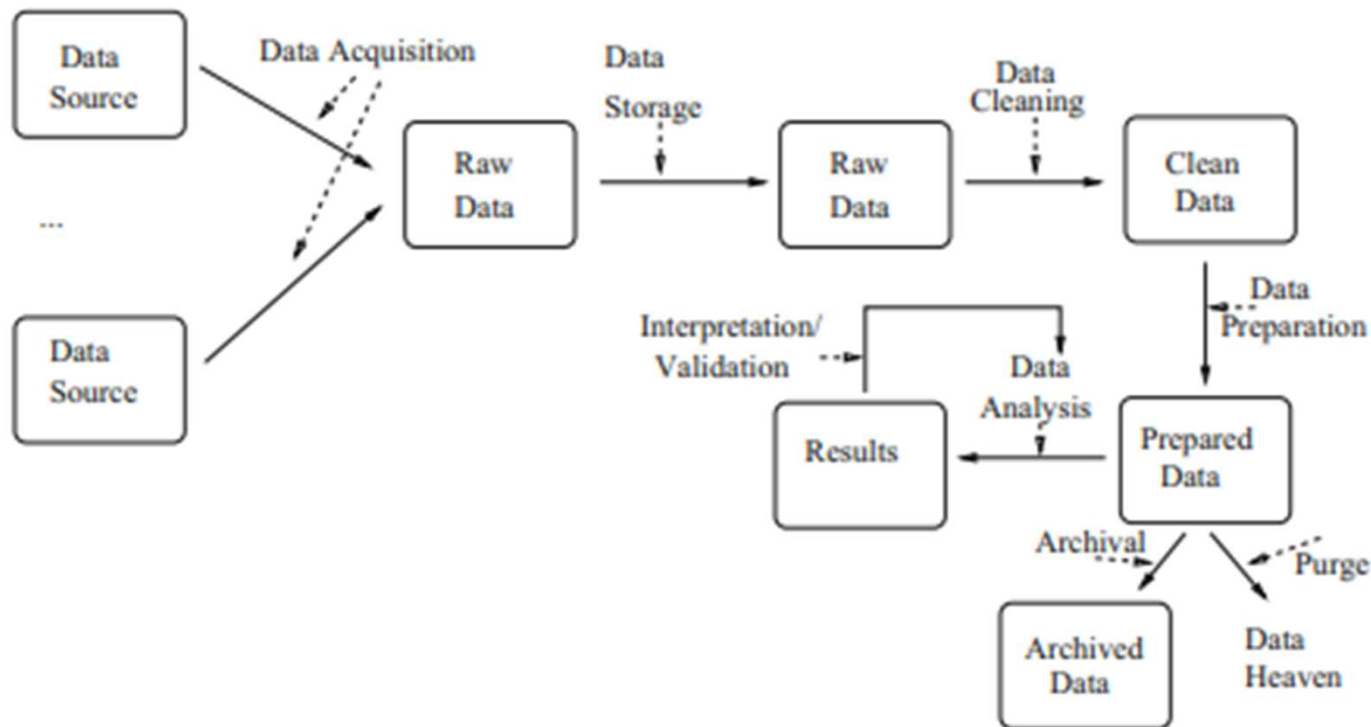
1

Very complex !

Alberto Pepe, Matthew Mayernik, Christine L. Borgman, Herbert Van de Sompel: "From Artifacts to Aggregations: Modeling Scientific Life Cycles on the Semantic Web".
<https://arxiv.org/ftp/arxiv/papers/0906/0906.2549.pdf>



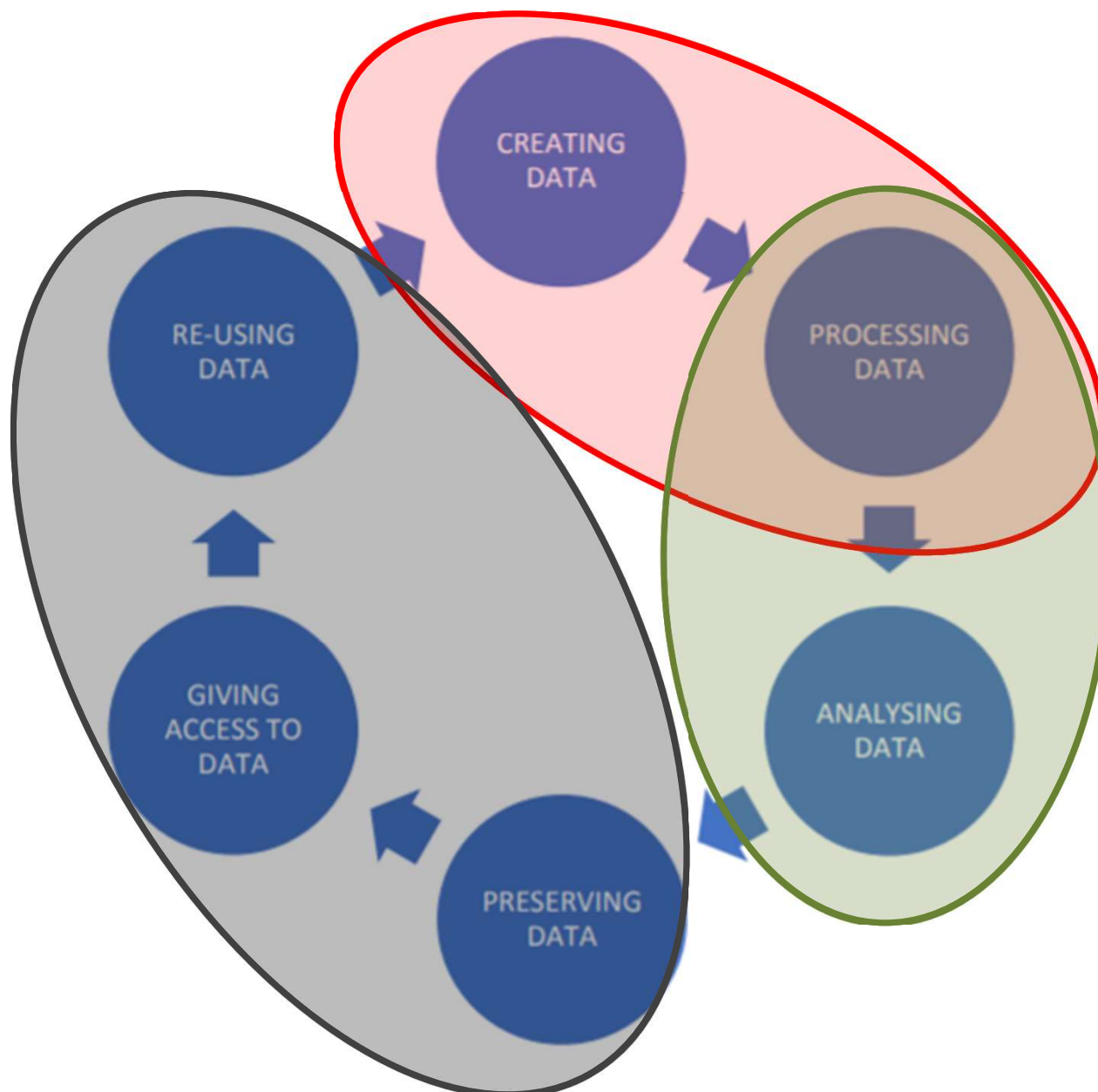
Data lifecycle (Data Science)



2

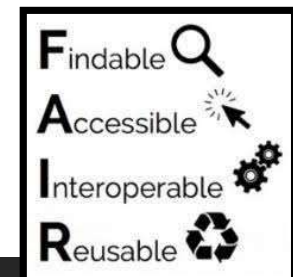
Very Poor !

Fig. 1.1 The data life cycle



1 + 2

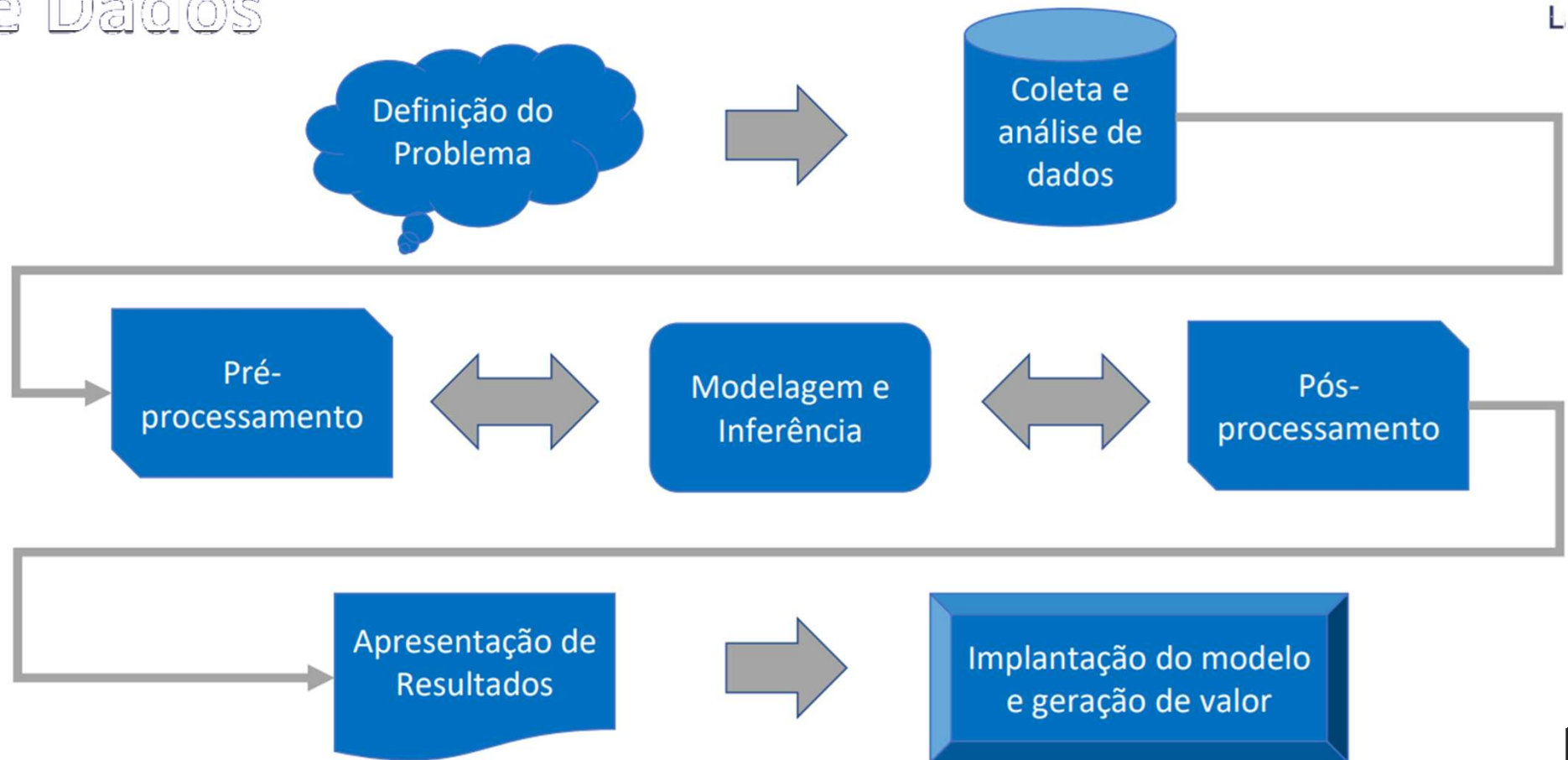
Very FAIR !



Ciclo de Vida de Projetos de Ciência de Dados

1. Entender o problema e definir objetivos - Que problema estou resolvendo?
2. Coletar e analisar os dados - De que informações preciso?
3. Preparar os dados - Como preciso tratar os dados?
4. Construir o modelo - Quais são os padrões nos dados que levam a soluções?
5. Avaliar e criticar o modelo - O modelo resolve meu problema?
6. Apresentar resultados - Como posso resolver o problema?
7. Distribuir o modelo - Como resolvo o problema no mundo real?

Ciclo de Vida de Projetos de Ciência de Dados



Etapa 1-2: Problema

Etapa 1 – Definição

1. Elencar as perguntas dos gestores, Requisitos funcionais e não-funcionais
2. Identificar as variáveis que desejam ser preditas ou descritas, assim como as que possivelmente são relacionadas
3. Classificar cada pergunta em um dos tipos de problemas de CD

Etapa 2 – Coleta de Dados

1. Verificar a disponibilidade das variáveis elencadas na etapa anterior
2. Modelar (se não existir) o DW/DM/DL, definir o processo de ETL/ELT e integrá-lo a uma ferramenta
3. Analisar os dados

Etapa 3-5: Recursos

Etapa 3 – Pre Processamento

- 1. Remover ou inputar dados faltantes e tratar dados inconsistentes
- 2. Corrigir ou amenizar outliers e desbalanceamento entre classes
- 3. Selecionar as variáveis e instâncias para compor o(s) modelo(s)

Etapa 4 – Modelagem e Inferência

- 1. Elencar os modelos possíveis e passíveis para cada tipo de problema
- 2. Estimar os parâmetros que compõem os modelos, baseando-se nas instâncias e variáveis pré-processadas
- 3. Avaliar os resultados de cada modelo, usando métricas e um processo justo de comparação

Etapa 5 – Pos Processamento

- 1. Combinar heurísticas de negócio com os modelos ajustados
- 2. Pós-avaliar tendo em vista os pontos fortes e dificuldades na implementação de cada um dos modelos

Etapa 6-7: Resultados

Etapa 6 - Apresentação de Resultado

1. Relatar a metodologia adotada para endereçar a solução às demandas dos gestores
2. Comparar os resultados do melhor modelo com o benchmark atual (caso haja)
3. Planejar os passos para a implantação da solução proposta

Etapa 7 – Implantação do modelo e geração de valor

1. Implantar o modelo em produção
2. Calcular os ganhos qualitativos (ganhos operacionais e de recursos humanos) e quantitativos (ROI e outras métricas)
3. Monitorar o modelo implantado

What? The Data Science Process

Ask an interesting question
& learn reproducibility

Get the Data

Explore the Data

Model the Data

Communicate/Visualize the Results

Plot the data.

Are there anomalies or egregious issues?

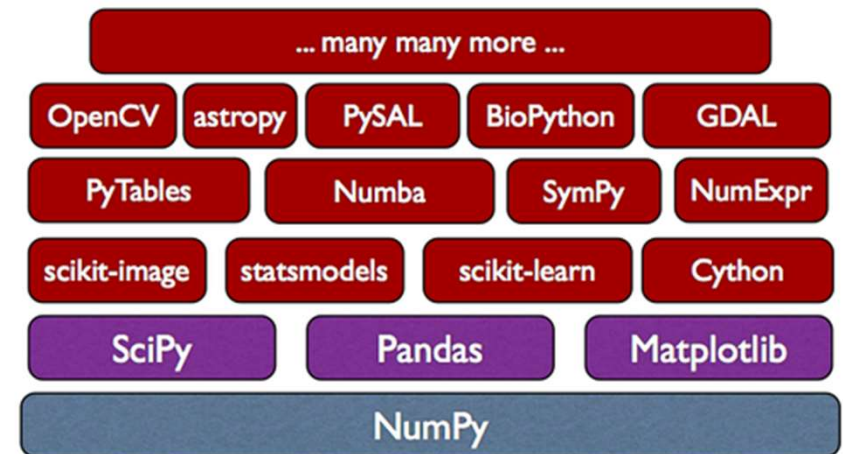
Are there patterns?

Modules II, III and IV

Data Cleaning Techniques

Data cleaning or cleansing is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database.

- Refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data.
- Missing Data
- Irregular Data (Outliers)
- Unnecessary Data — Repetitive Data, Duplicates and more
- Inconsistent Data — Capitalization, Addresses and more



Store and Explore Data

Why Pandas?

- Used by a lot of people
- Pandas is a fast, powerful, flexible and easy to use open-source **data analysis** and **manipulation tool**, built on top of the Python programming language.
- Allows for high-performance, easy-to-use data structures and data analysis
- Unlike NumPy library which provides multi-dimensional arrays,
- Pandas provides **1D table object** called **Series**
- Pandas provides **2D table object** called **DataFrame** (akin to a spreadsheet with column names and row labels).

Pandas

Series: a named, ordered dictionary

- The keys of the dictionary are the indexes
- Built on NumPy's `ndarray`
- Values can be any Numpy data type object

DataFrame: a table with named columns

- Represented as a Dict (col_name -> series)
- Each Series object represents a column

The diagram illustrates the addition of two Series to form a DataFrame. On the left, two Series are shown: 'apples' (values: 3, 2, 0, 1) and 'oranges' (values: 0, 3, 7, 2). These are added together (indicated by a '+' sign) to produce a DataFrame on the right. The DataFrame has two columns, 'apples' and 'oranges', with the same values as the original Series. The DataFrame is represented as a table with 4 rows and 2 columns.

	apples	oranges
0	3	0
1	2	3
2	0	7
3	1	2

The diagram shows a grid of cells representing a DataFrame. The grid is 5 rows high and 5 columns wide. The top row is shaded dark gray. The first column is shaded dark gray. The intersection of the first column and the first row (the top-left cell) is shaded a darker gray. A white rectangle highlights the cell at row 1, column 3. The word "row" is placed to the right of the grid, and the word "column" is placed below the grid.



Applications of Pandas

<https://data-flair.training/blogs/applications-of-pandas/>



Python Pandas Features

Multiple file
formats supported

Python
support

Input and
output tools

Optimized
performance

Great handling
of data

Handling
missing data

Grouping

Lot of time
series

Unique data

Cleaning
up data

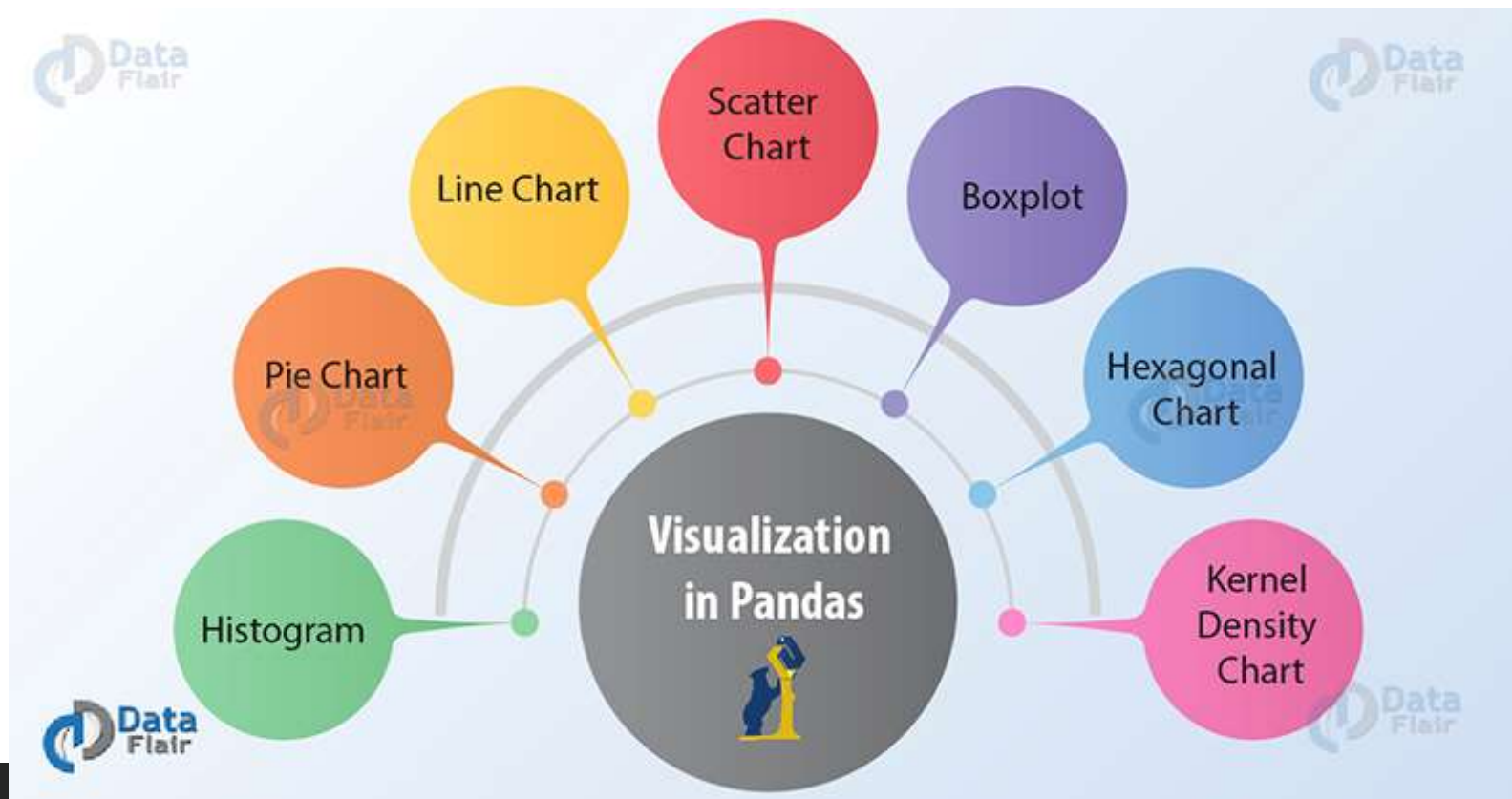
Alignment
and indexing

Merging and
joining of datasets

Visualize

Mask data

Visualization







Hands on...

NOTEBOOK:

PANDAS

A solid dark blue horizontal bar spanning the width of the slide at the bottom.