



DM,
Autumn, 2026



Pandas Introduction & Data Cleaning

Faculty of DS & AI
Autumn semester, 2026

Trong-Nghia Nguyen



Business AI Lab

Content

- **Pandas Introduction**
- Basic Pandas for BA
- Data Cleaning
- Data Visualization
- CnsPlot (adv.)

GROUP'S PROJECT REQUIREMENT

- First Presentation on **23 Sept. 2026**.
- Each group should have at **least 4 members**, class leader please **sends** Group list **before 18 Sept. 2026 (1 excel file)**
- In this presentation:
 - Introduction to your topic. Urgency (rationale/motivation).
 - Input/ output? Application
 - Literature review
 - Method (if able) - Plan.



Pandas Introduction

Pandas

- Pandas is an indispensable tool for data analysts using Python.
- The name "pandas" is derived from "panel data"—multidimensional **tabular data** in econometrics.
- It is built on **NumPy** but designed to make working with **tabular data easier**.



Pandas Introduction

Pandas

Conventions for referring to the library:

```
python  
  
import numpy as np  
import pandas as pd
```

Pandas Introduction

Two core data structures

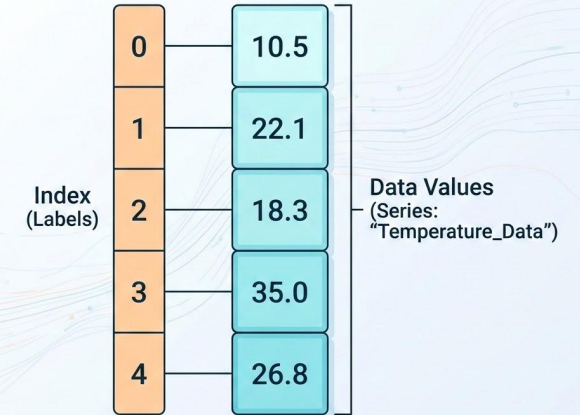
- **Series** — a one-dimensional array with labels (similar to a column):

```
python
s = pd.Series([45000, 35000, 50000])
s
# 0    45000
# 1    35000
# 2    50000
```

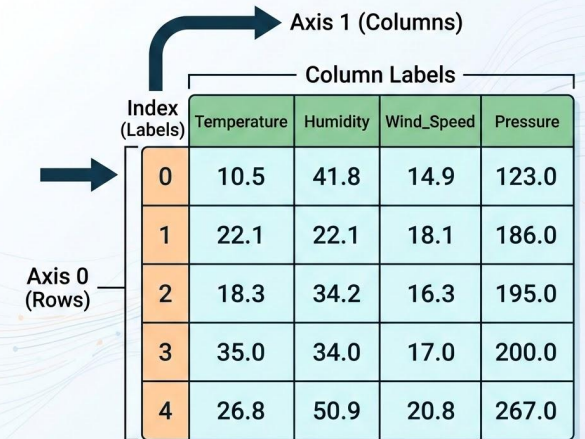
- **DataFrame** — two-dimensional table (many Series combined), is the "data table"

```
python
df = pd.DataFrame({
    'Ten': ['An', 'Binh', 'Chi'],
    'Tuoi': [28, 35, 41],
    'ChiTieu': [45000, 35000, 50000]
})
```

Pandas Series: 1D Labeled Array



Pandas DataFrame



Content

- Pandas Introduction
- **Basic Pandas for BA**
- Data Cleaning
- Data Visualization
- CnsPlot (adv.)

Basic Pandas for BA

Load & quickly view data

python

```
df = pd.read_csv('customers.csv')    # đọc file CSV thành DataFrame

df.head()        # 5 dòng đầu
df.shape         # (số dòng, số cột)
df.columns       # danh sách tên cột
df.info()        # kiểu dữ liệu & số ô không rỗng mỗi cột
df.describe()    # thống kê mô tả cho cột số (mean, min, max, quartiles)
```

Basic Pandas for BA

Columns selection

python

```
df['ChiTieu']          # chọn một cột → trả về Series  
df[['Ten', 'ChiTieu']] # chọn nhiều cột → trả về DataFrame
```

Basic Pandas for BA

Selecting rows using `loc` and `iloc`

python

```
df.loc[0]           # dòng theo NHÃN chỉ số  
df.iloc[0]          # dòng theo VỊ TRÍ (0-based)  
df.iloc[0:5]        # 5 dòng đầu (slicing giống list)  
df.loc[0:2, ['Ten', 'Tuoi']] # dòng 0-2, chỉ 2 cột
```

- `loc` — select by label (index name / column name).
- `iloc` — select by position number (like slicing list).
- Remember: `iloc` = integer location (location); `loc` = label (label). Slicing on Session 2 now becomes `df.iloc[row, column]`.

Basic Pandas for BA

Filter rows based on a condition (boolean filter)

python

```
df[df['Tuoi'] > 30]           # khách trên 30 tuổi  
df[df['ChiTieu'] >= 40000]    # chi tiêu ≥ 40k  
df[(df['Tuoi'] > 30) & (df['ChiTieu'] > 40000)]  # kết hợp 2 điều kiện
```

Content

- Pandas Introduction
- Basic Pandas for BA
- **Data Cleaning**
- Data Visualization
- CnsPlot (adv.)

Data Cleaning

Detecting and handling missing values

python

```
df.isnull().sum()          # đếm ô thiếu mỗi cột
df['Tuoi'].fillna(df['Tuoi'].mean())  # điền trung bình cho cột số
df['Gioitinh'].fillna(df['Gioitinh'].mode()[0])  # điền mode cho phân loại
df.dropna()                # bỏ các dòng có ô thiếu
```

Data Cleaning

Replace inconsistent values

python

```
df['IsActive'].replace({'99': 'N', '?': 'N'})      # sửa giá trị lỗi  
df['Gender'].replace({'Male': 'M', 'female': 'F', 'f': 'F'})  # chuẩn hóa nhãn
```

Data Cleaning

Remove duplicates and remove columns

python

```
df.drop_duplicates()           # bỏ dòng trùng  
df.drop(columns=['CustomerID']) # bỏ cột thừa
```

Data Cleaning

Summarize & Group (Pattern Discovery)

python

```
df['City'].value_counts()          # đếm theo nhóm  
df.groupby('Gender')['ChiTieu'].mean() # chi tiêu trung bình theo giới tính  
df.sort_values('ChiTieu', ascending=False) # sắp xếp giảm dần
```

Content

- Pandas Introduction
- Basic Pandas for BA
- Data Cleaning
- **Data Visualization**
- CnsPlot (adv.)

Data Visualization

Matplotlib

python

```
import matplotlib.pyplot as plt

df['Tuoi'].plot(kind='hist') # biểu đồ tần suất (phân bố tuổi)
df['City'].value_counts().plot(kind='bar') # cột: số khách mỗi thành phố
df.plot(kind='scatter', x='Tuoi', y='ChiTieu') # phân tán: tuổi vs chi tiêu
plt.show()
```

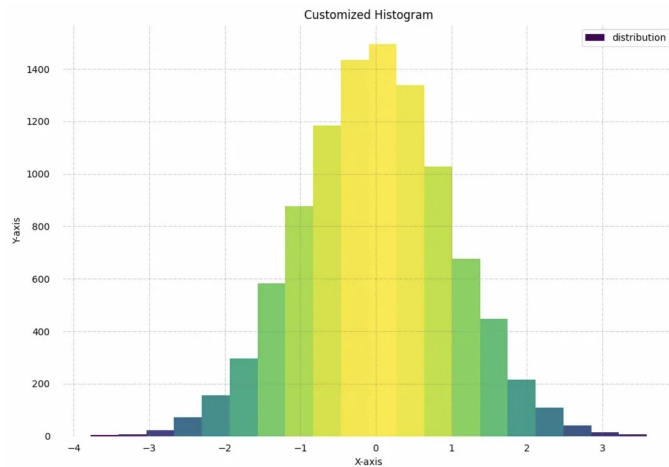
- Commonly used `kind` types: `hist` (distribution), `bar` (group comparison), `box` (outlier detection), `scatter` (relationship), `line` (trend).



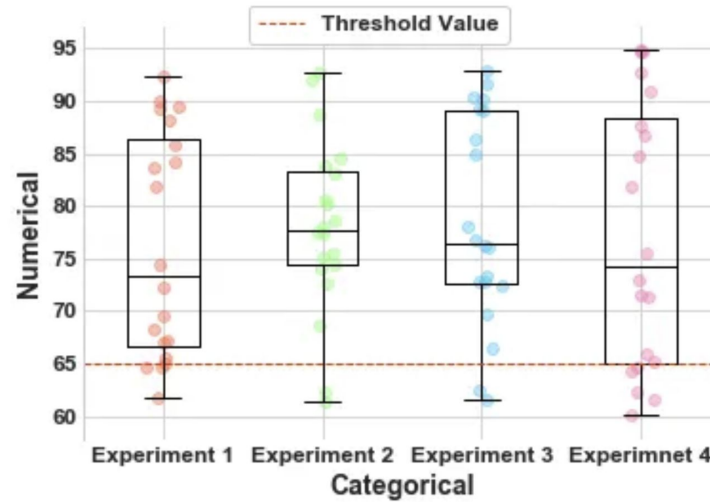
Data Visualization

Matplotlib

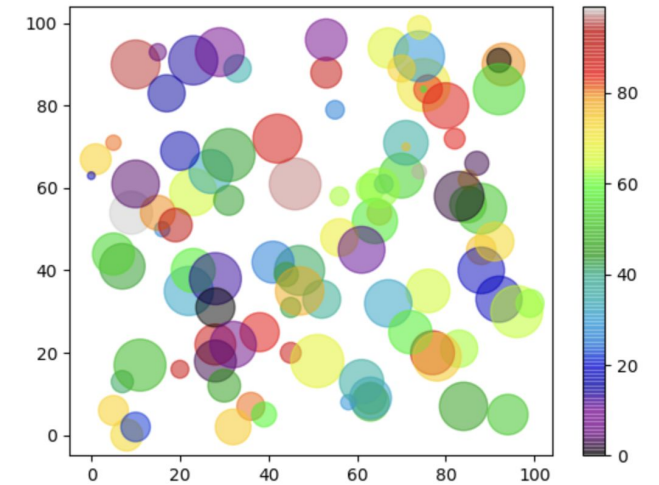
- Commonly used `kind` types: `hist` (distribution), `bar` (group comparison), `box` (outlier detection), `scatter` (relationship), `line` (trend).



Hist



Box



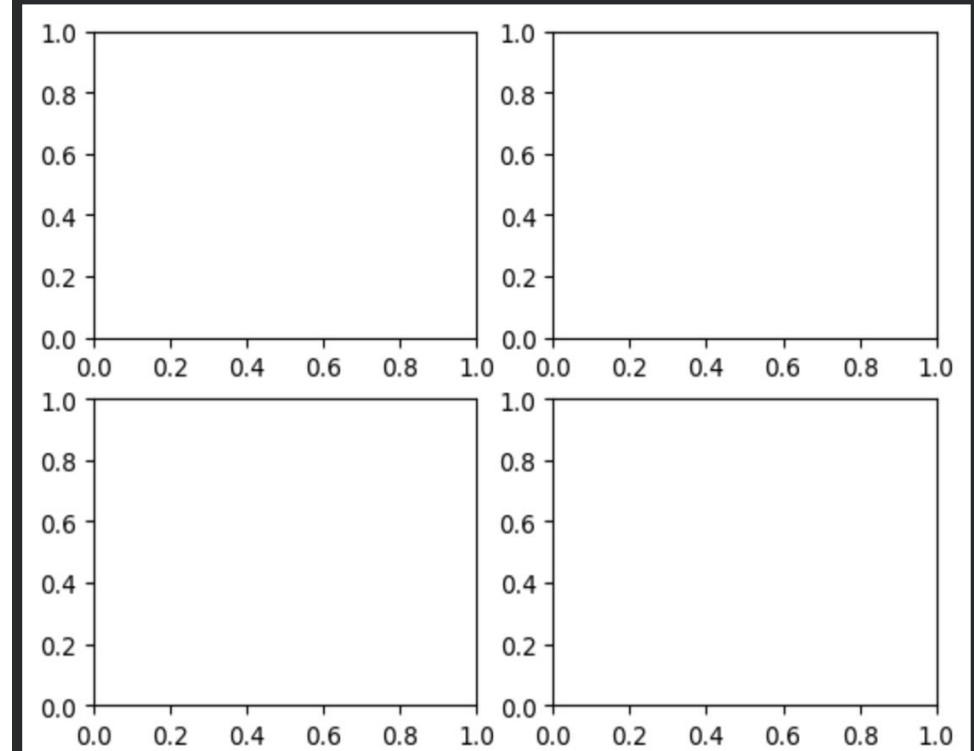
Scatter

Data Visualization

Figures and subplots

- Plots in Matplotlib are contained within an object called a **Figure**. You can create a new figure using `plt.figure`.
- Creating multiple subplots:
- Example: 2×2 grid
 - Create a 2×2 layout (up to four plots)
 - Select the first of the four subplots (numbered starting from 1)
 -  Result:
 - If you create two additional subplots
 - You will have a visualization containing three subplots

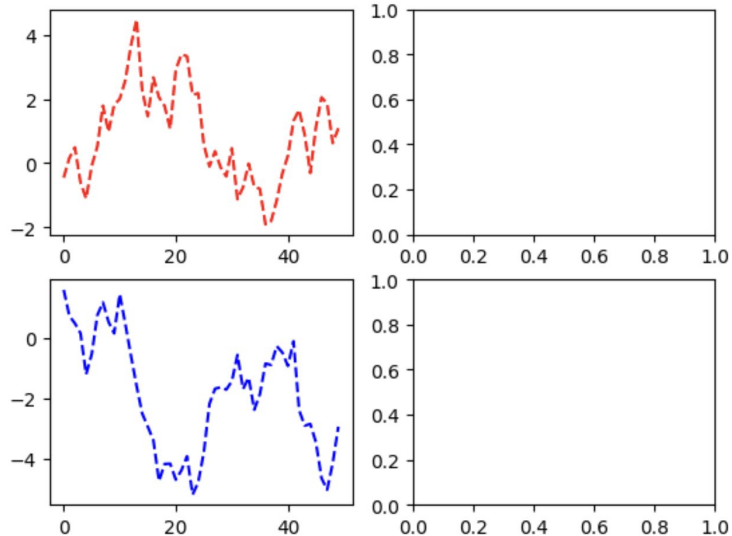
```
1 fig = plt.figure() # Phải tạo lại fig vì cell trước có thể đã được t
2 ax1 = fig.add_subplot(2, 2, 1)
3 ax2 = fig.add_subplot(2, 2, 2)
4 ax3 = fig.add_subplot(2, 2, 3)
5 ax4 = fig.add_subplot(2, 2, 4)
6 # plt.savefig("chart1.png") # Để lưu hình
```



Data Visualization

Figures and subplots

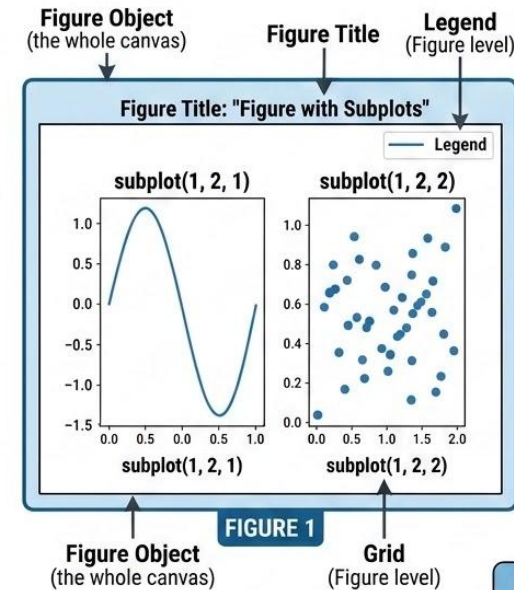
```
1 fig = plt.figure()
2 ax1 = fig.add_subplot(2, 2, 1)
3 ax2 = fig.add_subplot(2, 2, 2)
4 ax3 = fig.add_subplot(2, 2, 3)
5 ax4 = fig.add_subplot(2, 2, 4)
6
7 ax1.plot(np.random.standard_normal(50).cumsum(), color="red", linestyle="dashed")
8 #ax2.plot(np.random.standard_normal(50).cumsum(), color="green", linestyle="dashed")
9 ax3.plot(np.random.standard_normal(50).cumsum(), color="blue", linestyle="dashed")
10 #ax4.plot(np.random.standard_normal(100).cumsum(), color="black", linestyle="dashed")
11
12 plt.savefig("chart2.png")
```



1 Matplotlib Figure vs Subplots

1 WHAT IS A FIGURE?

The Container for all plot elements.
A canvas for graphics (can contain multiple axes/subplots, colorbars, legends).



2 WHAT ARE SUBPLOTS (AXES)?

Independent charts contained within a single Figure. Each possesses its own coordinate system (X & Y axes) for plotting data.

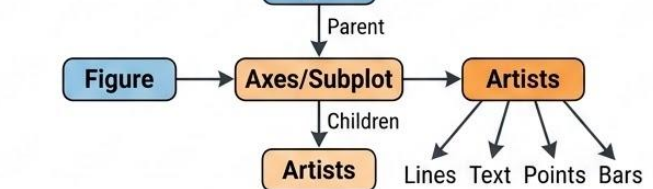
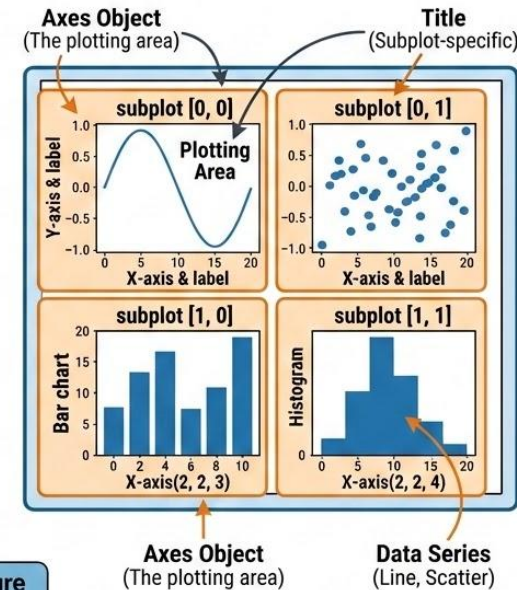


Figure = The overall canvas, Subplots = individual charts

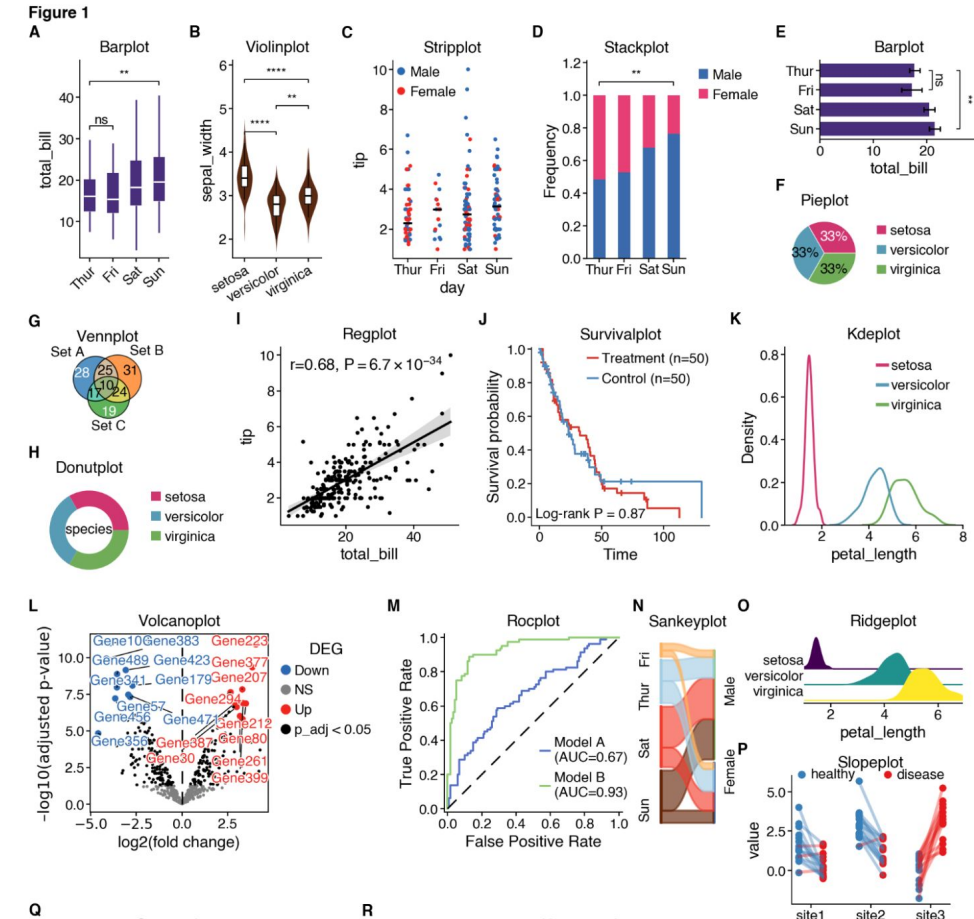
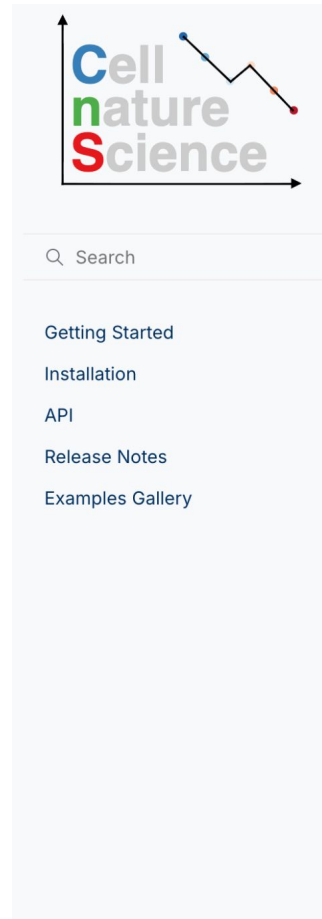
Content

- Pandas Introduction
- Basic Pandas for BA
- Data Cleaning
- Data Visualization
- **CnsPlot (adv.)**

CnsPlot (adv.)

CnsPlot

- **Cnsplots** is a Python visualization library for **publication-quality figures**—meeting the standards of top-tier journals such as **Cell**, **Nature**, and **Science** (abbreviated as CNS).
- It is built on **matplotlib + seaborn** but produces attractive visuals out of the box with a very simple API, and exports vector files (SVG/PDF) that can be edited in Adobe Illustrator.



CnsPlot (adv.)

Install

```
bash
```

```
pip install cnsplots
```

(Requirement: Python 3.10+. On Colab, run: `!pip install cnsplots` .)

Test:

```
python
```

```
import cnsplots  
print(cnsplots.__version__)
```

CnsPlot (adv.)

Quick Start

python

```
import cnsplots as cns

df = cns.datasets.load_dataset("tips")    # dữ liệu mẫu có sẵn
cns.figure(100, 150)                     # kích thước (rộng x cao, tính bằng pixel)
cns.boxplot(data=df, x="day", y="total_bill")
cns.savefig("figure.svg")                 # lưu hình vector
```

In these functions: `data` is a DataFrame; `x`, `y`, and `hue` are column names in that DataFrame.

CnsPlot (adv.)

Quick Start

python

```
# Box plot – so sánh phân bố + phát hiện outlier
cns.figure(80, 100)
cns.boxplot(data=df, x="day", y="total_bill", hue="sex")

# Bar plot – so sánh giá trị giữa các nhóm
cns.barplot(data=df, x="day", y="total_bill", hue="sex")

# Scatter plot – mối liên hệ giữa hai biến số
cns.scatterplot(data=df, x="total_bill", y="tip", hue="day")

# Histogram – phân bố một biến
cns.histplot(data=df, x="total_bill")
```

cnsplots has many other types (violin, line, heatmap, ROC, confusion matrix...) — see [Examples Gallery](#).

CnsPlot (adv.)

Quick Start

Each function returns a matplotlib **Axes** object, the labels should be adjusted as usual:

```
python
```

```
cns.figure(80, 100)
ax = cns.boxplot(data=df, x="day", y="total_bill")
ax.set_xlabel("Ngày trong tuần")
ax.set_ylabel("Hóa đơn ($)")
ax.set_title("Chi tiêu theo ngày")
cns.savefig("hinh.pdf")      # lưu vector cho báo cáo (svg / pdf / png)
```

Thank you!