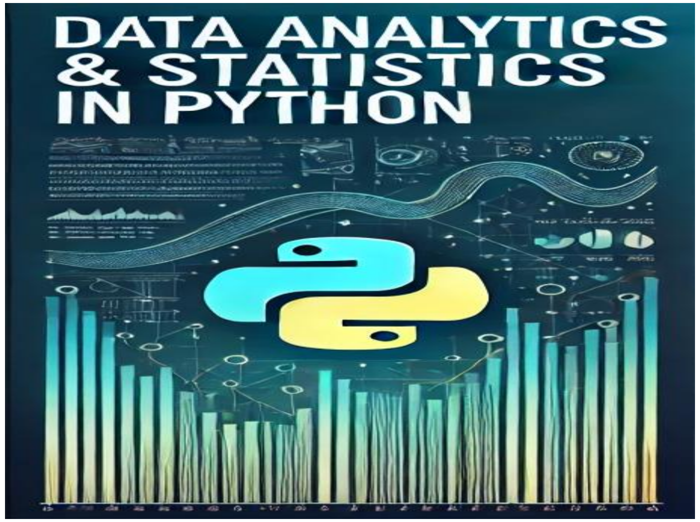
Data Analytics & Statistics in Python:

Recap of Sessions 1–6 & Cryptocurrency Analysis Mini-Project



Learning data-driven decision-making with Python

Instructor: Hamed Ahmadinia, Ph.D.

Email: hamed.ahmadinia@metropolia.fi

Concepts of Today

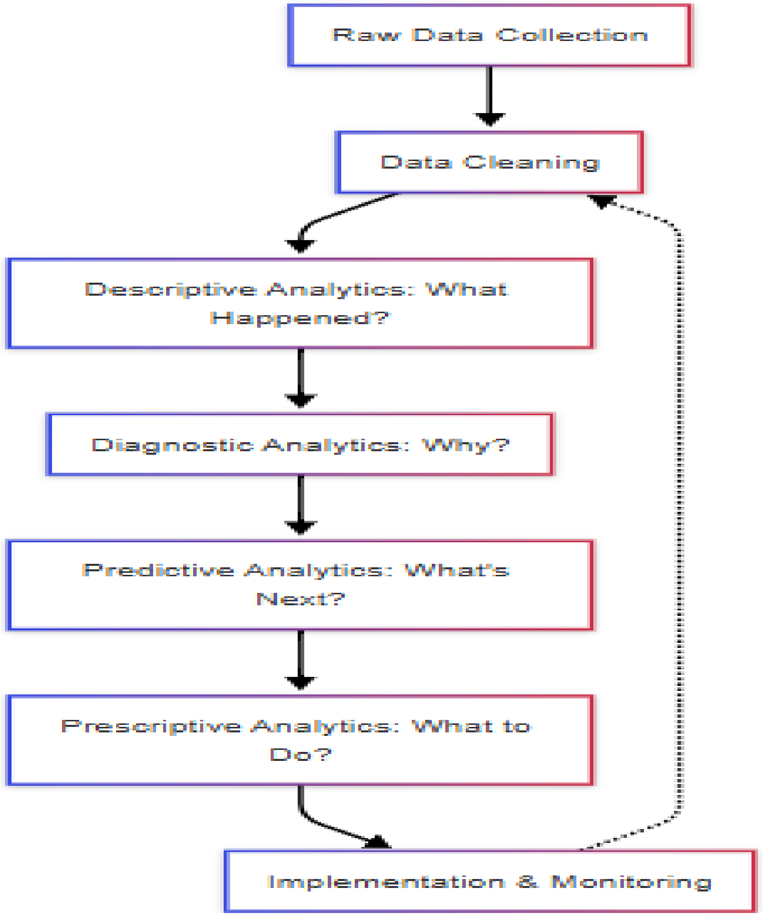
Session Agenda:

- Course Overview
- Cryptocurrency Mini-project Overview
- Jupyter Notebook Walkthrough
- Descriptive Stats, Visualisation & Hypothesis Testing
- Predictive Insights & Token Recommendation
- Kahoot Quiz

Session 1: Introduction & Fundamentals

- Overview of data analytics and its types (Descriptive, Diagnostic, Predictive, Prescriptive)

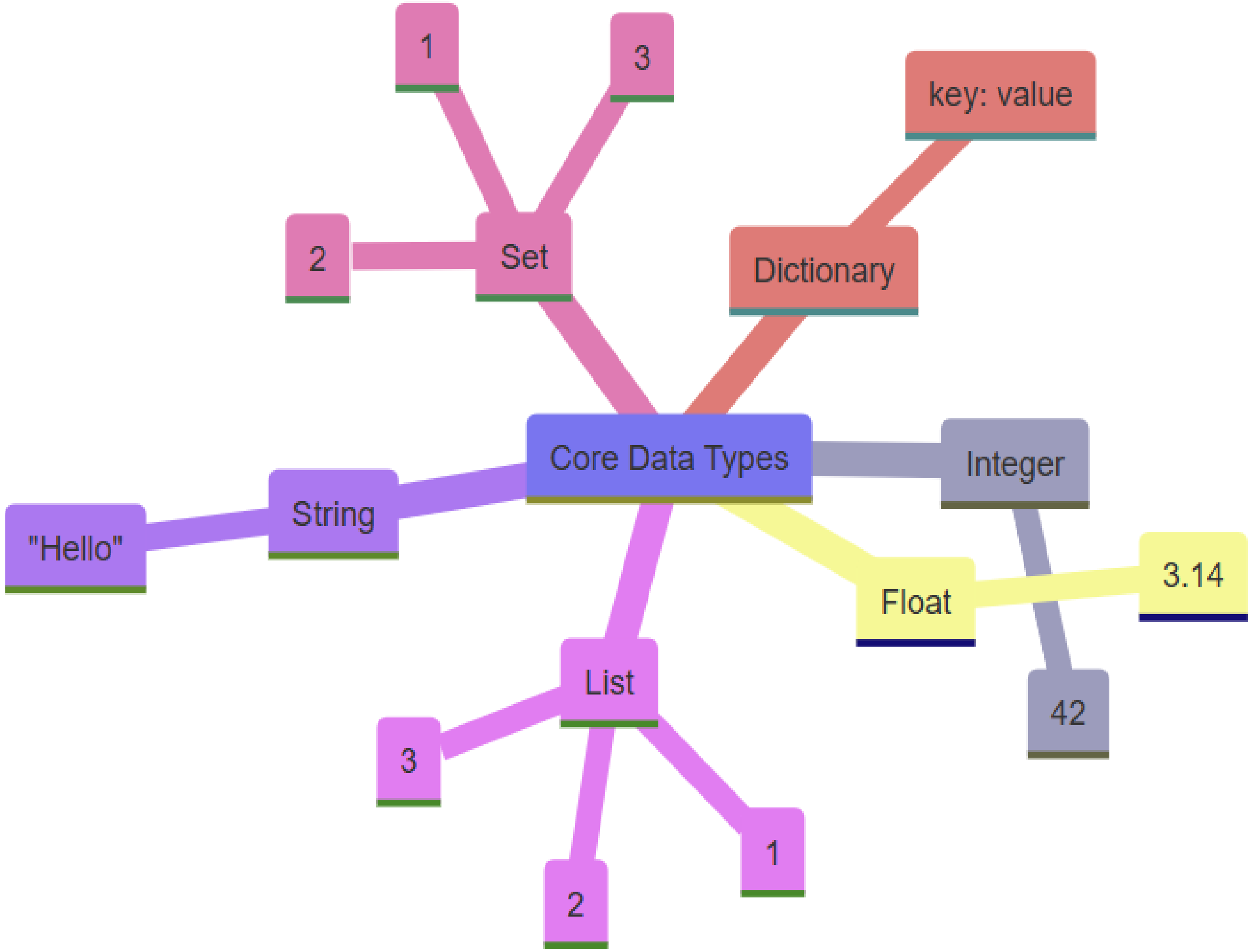
- Course structure and recommended tools (Anaconda, VS Code, etc.)


Raw Data Collection
Data Cleaning
Descriptive Analytics: What Happened?
Diagnostic Analytics: Why?
Predictive Analytics: What's Next?
Prescriptive Analytics: What to Do?
Implementation & Monitoring

Python Basics Recap

Core data types:
integers, floats, strings,
lists, sets, dictionaries

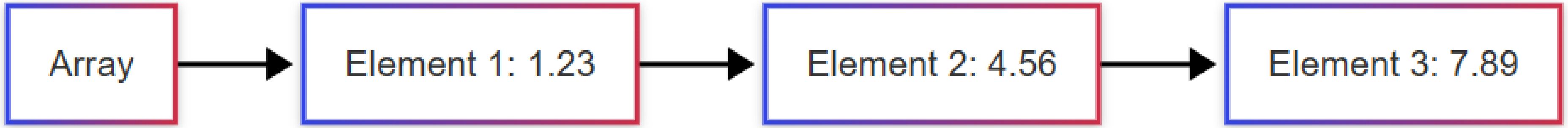
Basic control flows:
if/else, loops,
functions, file I/O



Working with Data Frames & Arrays (Session 2)

- **Introduction to NumPy arrays** (np.array, np.arange, np.linspace)
- **Creating and manipulating Pandas Data Frames**

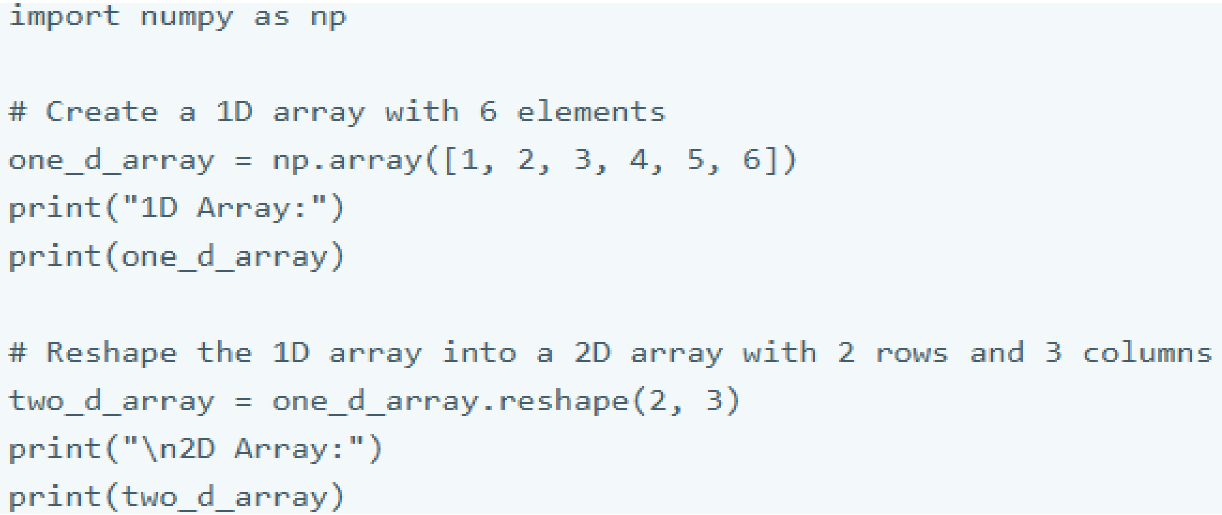
A	B	C
1	2	3
4	5	6
7	8	9



NumPy: Array Creation & Reshaping

Array creation: np.array, np.zeros, np.ones, np.eye

Reshaping arrays: reshape(), ravel(), transpose, newaxis



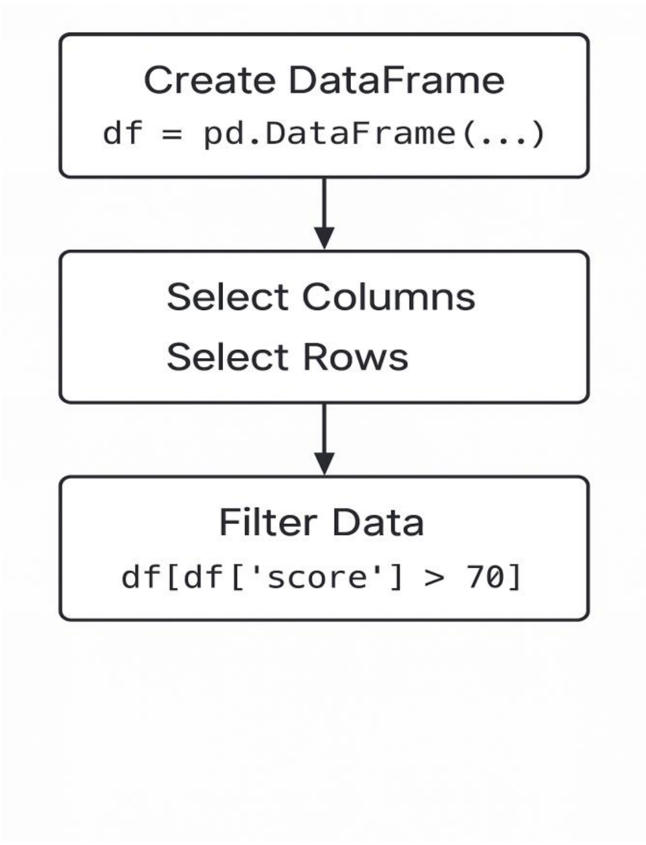
```
import numpy as np

# Create a 1D array with 6 elements
one_d_array = np.array([1, 2, 3, 4, 5, 6])
print("1D Array:")
print(one_d_array)

# Reshape the 1D array into a 2D array with 2 rows and 3 columns
two_d_array = one_d_array.reshape(2, 3)
print("\n2D Array:")
print(two_d_array)
```

- **np.array([1, 2, 3, 4, 5, 6])**: Creates a 1D Numpy array with the elements 1, 2, 3, 4, 5, and 6.
- **reshape(2, 3)**: Reshapes the 1D array into a 2D array with 2 rows and 3 columns.
- **print**: Outputs the arrays to the console.

Pandas Data Handling Essentials

CREATING DATA FRAMES, INDEXING, AND SELECTING DATA

EDITING DATA: ADDING/DROPPING COLUMNS, FILTERING, GROUPING



Create DataFrame

`df = pd.DataFrame(...)`

Select Columns

Select Rows

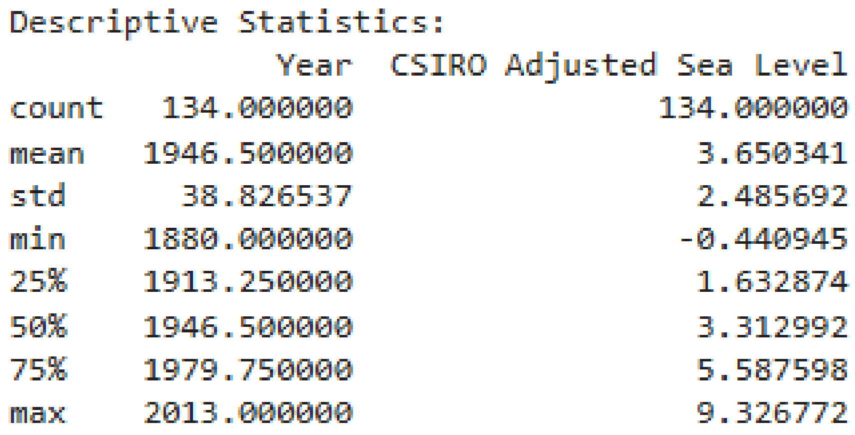
Filter Data

`df[df['score'] > 70]`

Session 3: Descriptive Statistics Overview

Measures of central tendency: mean, median, mode

Measures of spread: range, quantiles, IQR, variance, standard deviation

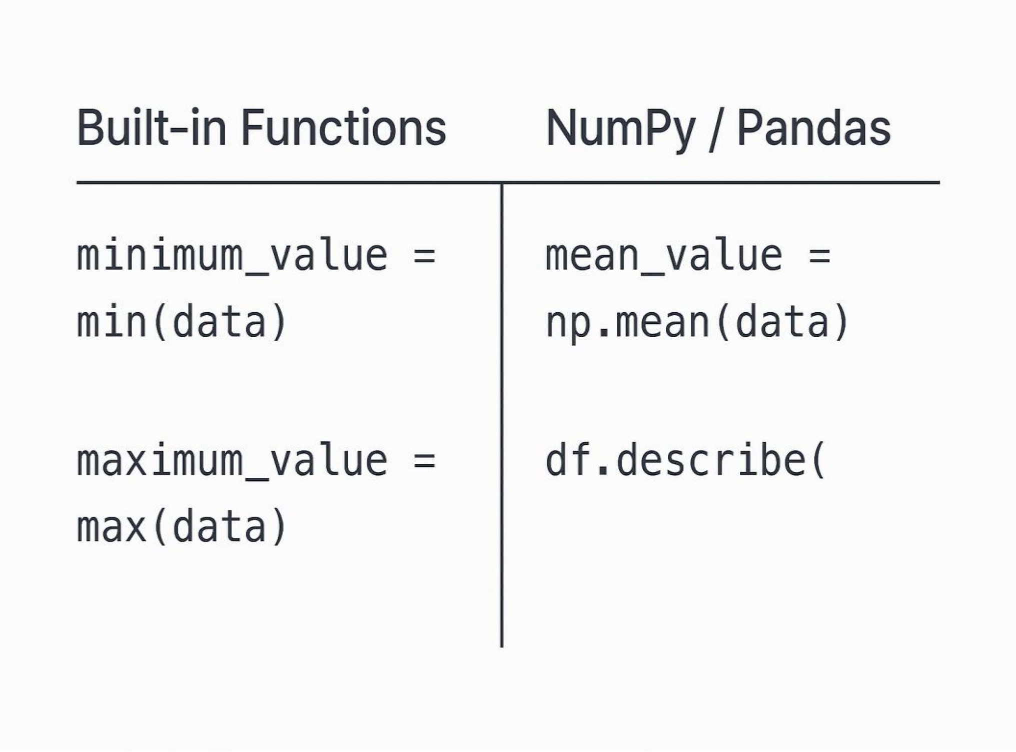


```
Descriptive Statistics:
             Year  CSIRO Adjusted Sea Level
count   134.000000                134.000000
mean   1946.500000                  3.650341
std      38.826537                  2.485692
min    1880.000000                 -0.440945
25%    1913.250000                  1.632874
50%    1946.500000                  3.312992
75%    1979.750000                  5.587598
max    2013.000000                  9.326772
```

Python Functions for Descriptive Statistics

BUILT-IN FUNCTIONS (MIN(), MAX()) AND NUMPY METHODS (NP.MEAN(), NP.MEDIAN())

PANDAS METHODS: DF.DESCRIBE(), DF.MIN(), DF.MAX()

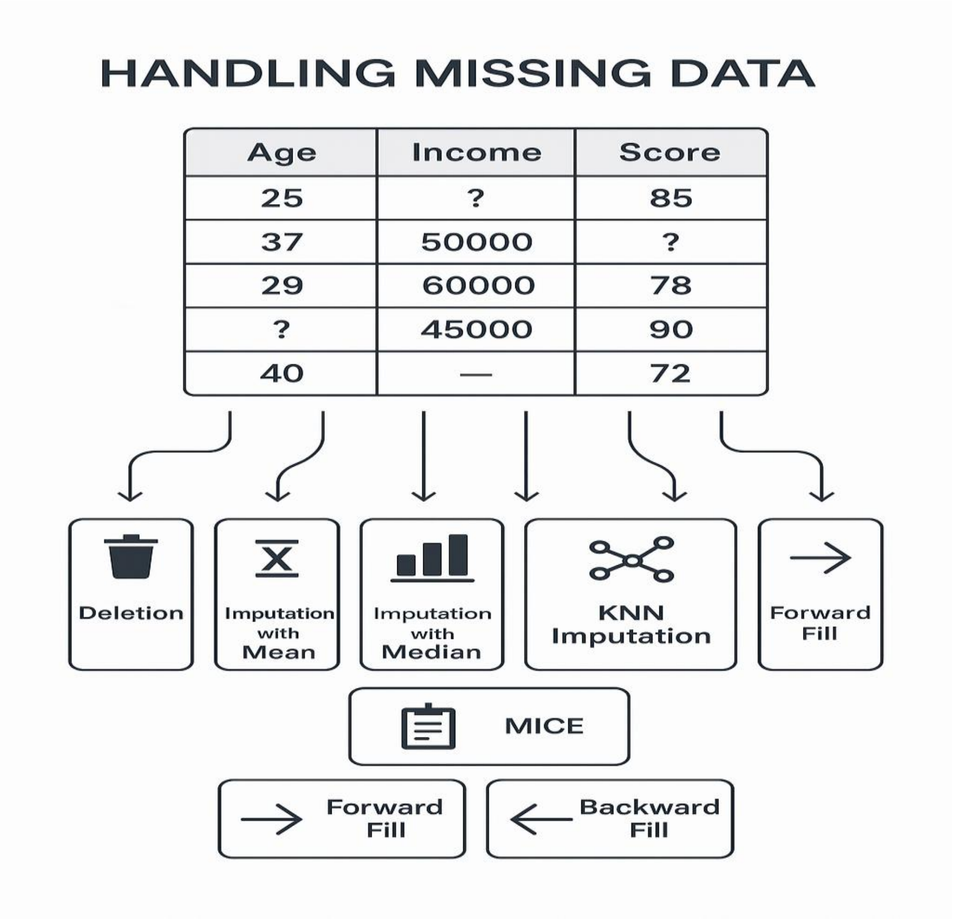


Built-in Functions	NumPy / Pandas
`minimum_value = min(data)`	`mean_value = np.mean(data)`
`maximum_value = max(data)`	`df.describe(`

Handling Missing Data

TYPES OF MISSING DATA: MCAR, MAR, MNAR

METHODS: DELETION, BASIC IMPUTATION (MEAN, MEDIAN, MODE), ADVANCED TECHNIQUES (KNN, MICE)



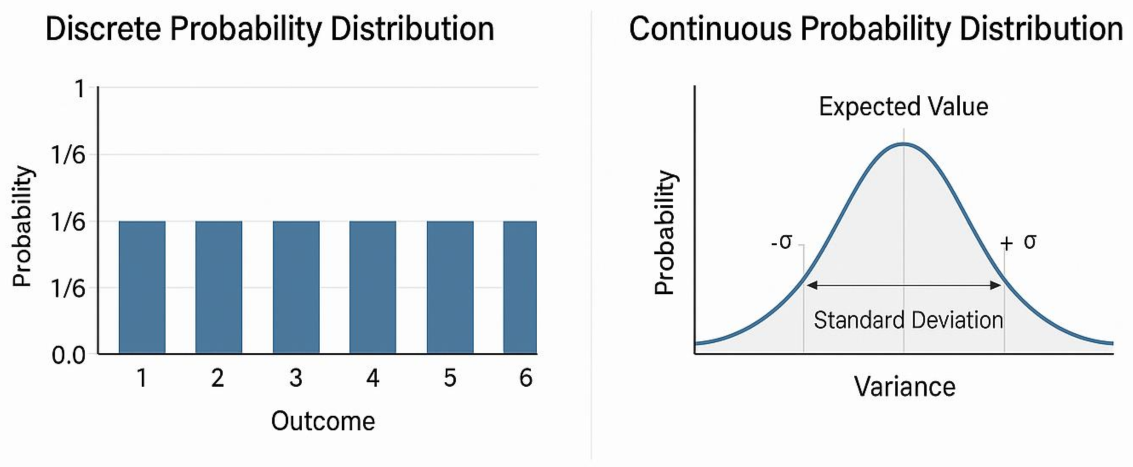
HANDLING MISSING DATA

Age	Income	Score
25	?	85
37	50000	?
29	60000	78
?	45000	90
40	—	72

Session 4: Probability & Variability

Probability distributions: discrete vs. continuous

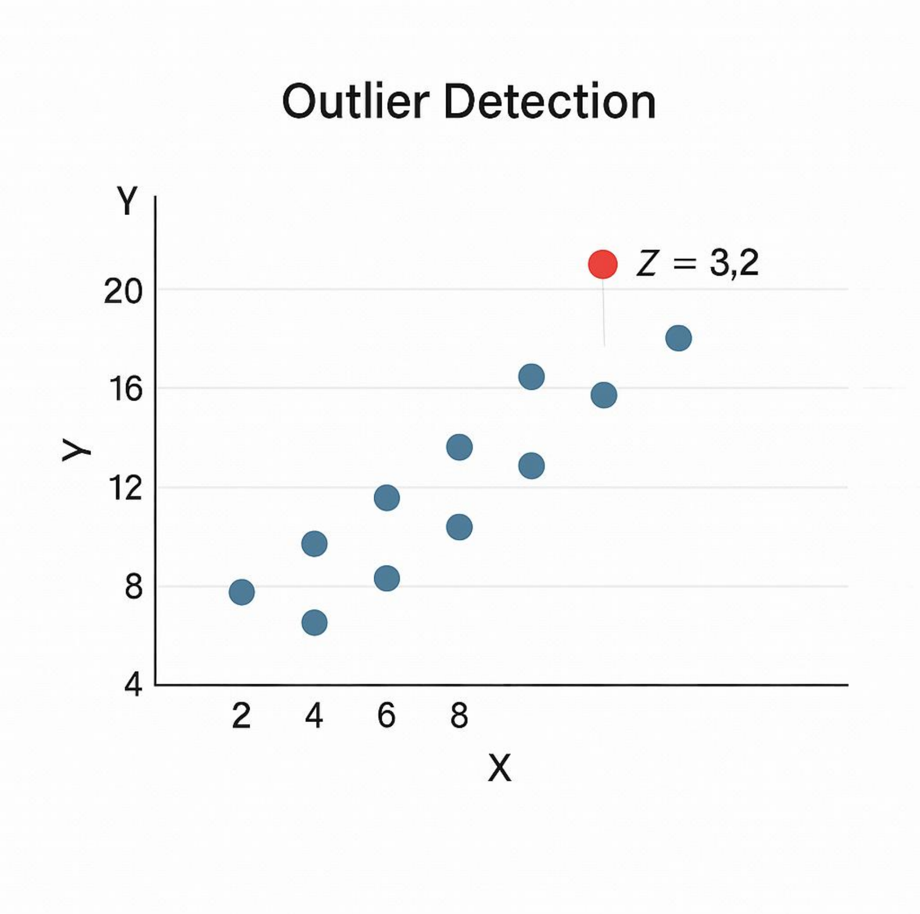
Key concepts: expected value, variance, standard deviation, and the normal distribution



Z-Score & Outlier Detection

Z-score formula: z = (x – μ) / σ

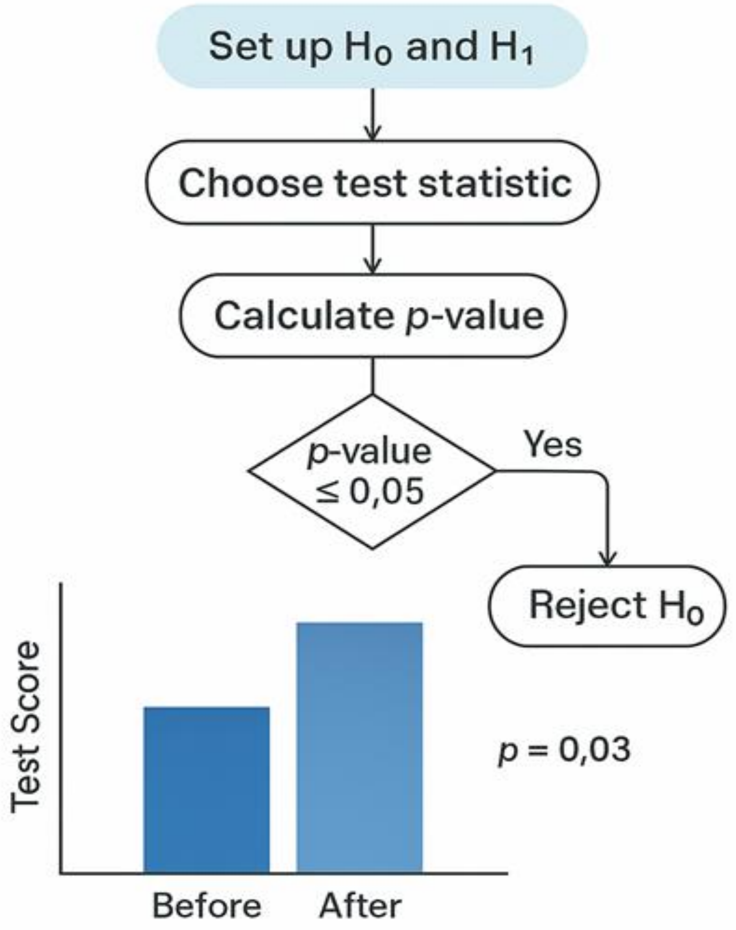
Using z-scores to identify outliers (typically |z| > 3)



Hypothesis Testing Overview

Null vs. alternative hypotheses, p-values, significance levels

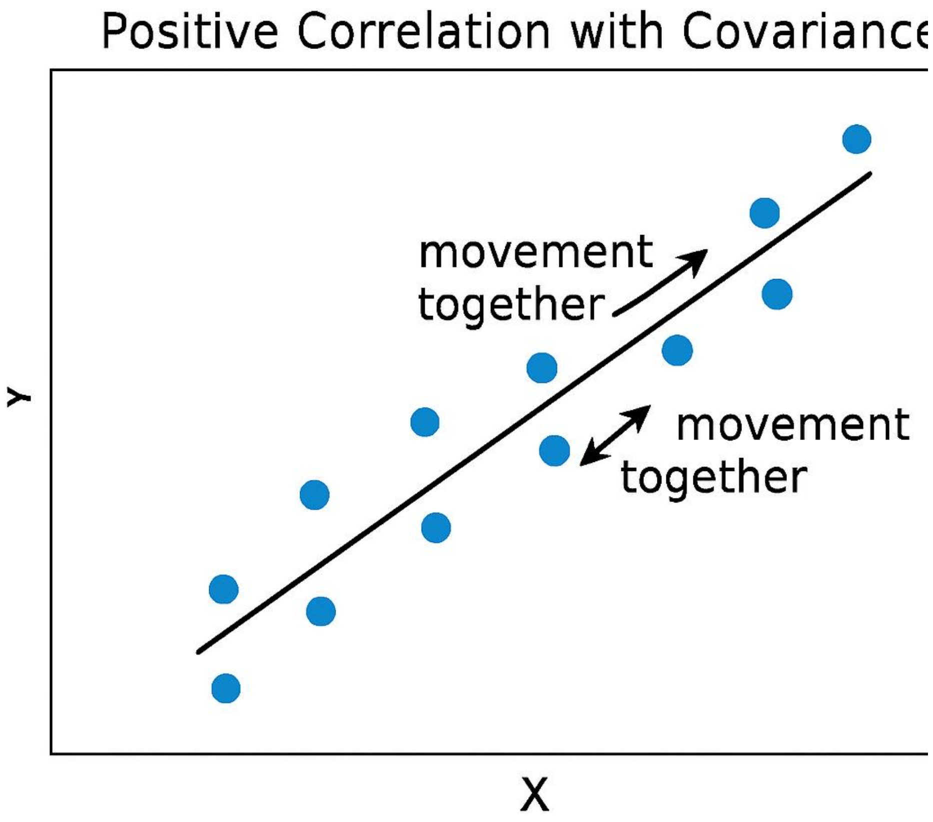
Overview of one-sample, two-sample, and paired sample tests



Session 5: Relationships Between Variables

UNDERSTANDING COVARIANCE AND CORRELATION

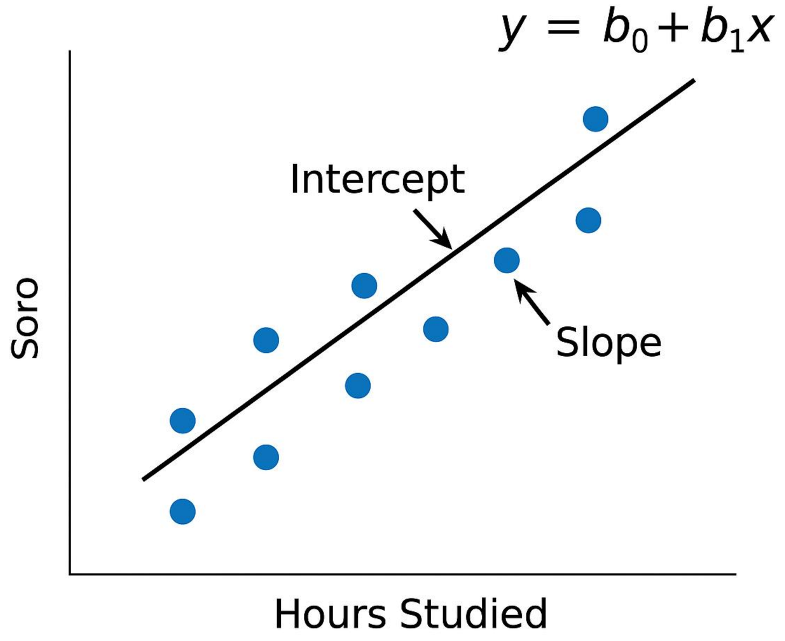
DIFFERENT CORRELATION METRICS: PEARSON, SPEARMAN, KENDALL



Introduction to Linear Regression

BASIC REGRESSION EQUATION: $Y = B_0 + B_1X$

EXTENSION TO MULTIVARIATE REGRESSION

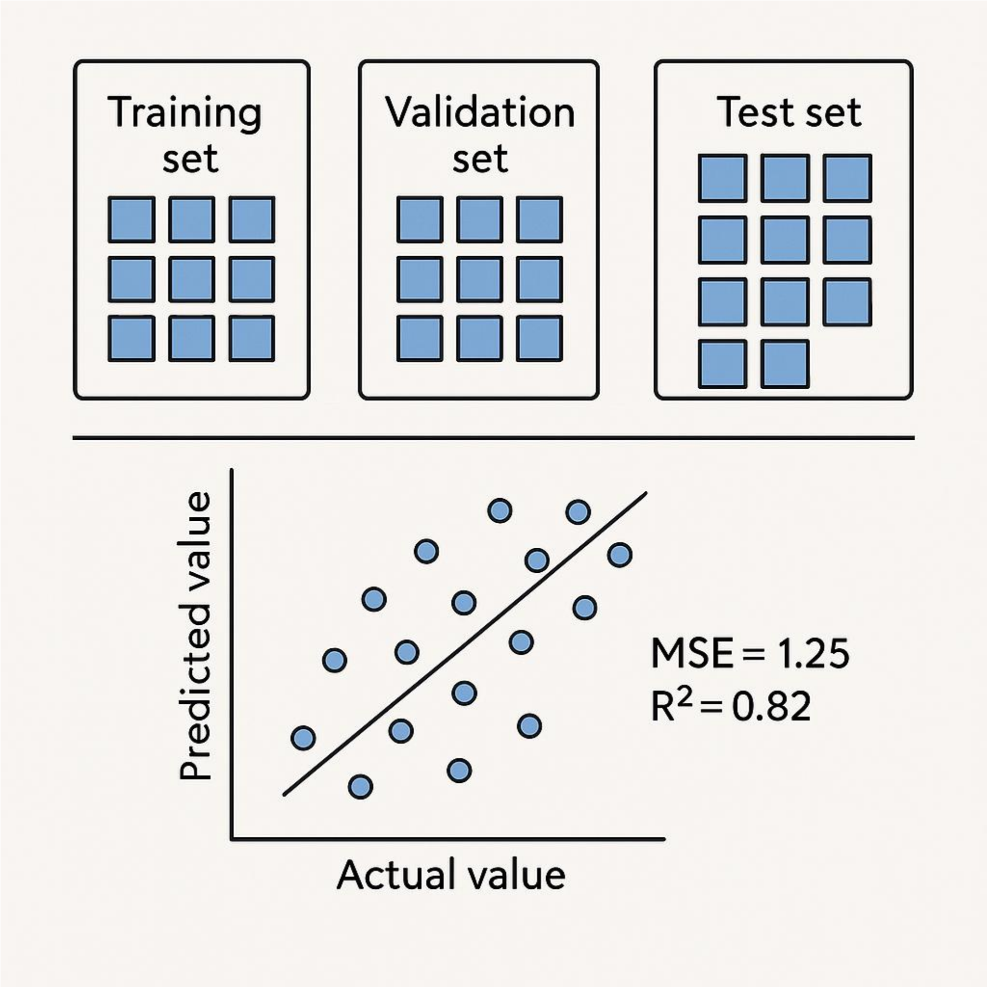


Evaluating Regression Models

Data splitting: training, validation, test sets

Evaluation metrics: Mean Squared Error (MSE) and R^2 score

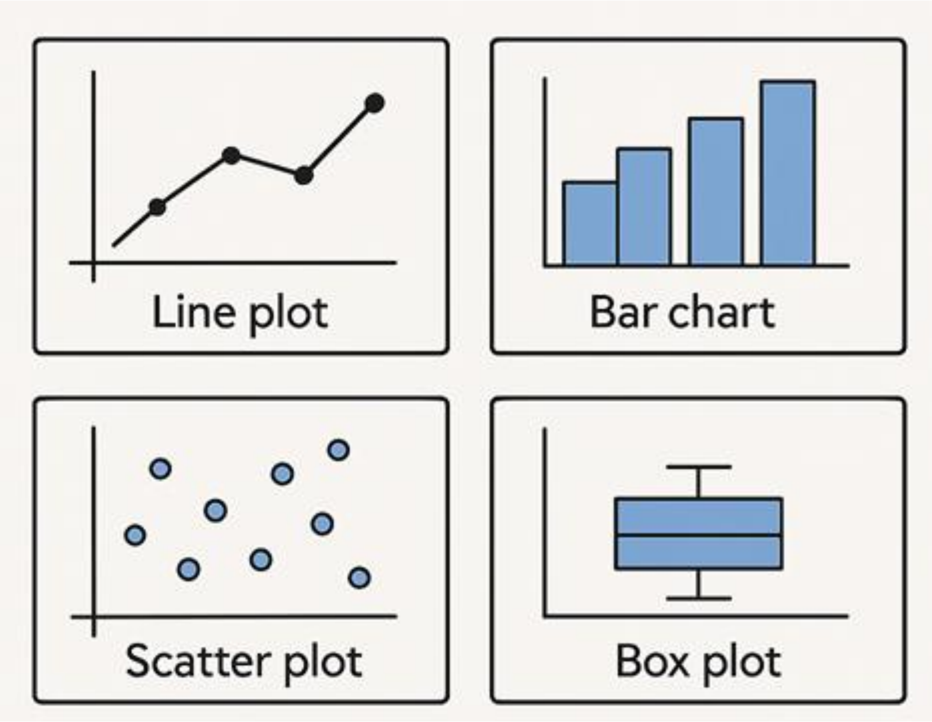
Concepts of overfitting and underfitting



Session 6: Data Visualization Fundamentals

Importance of visualization for communication

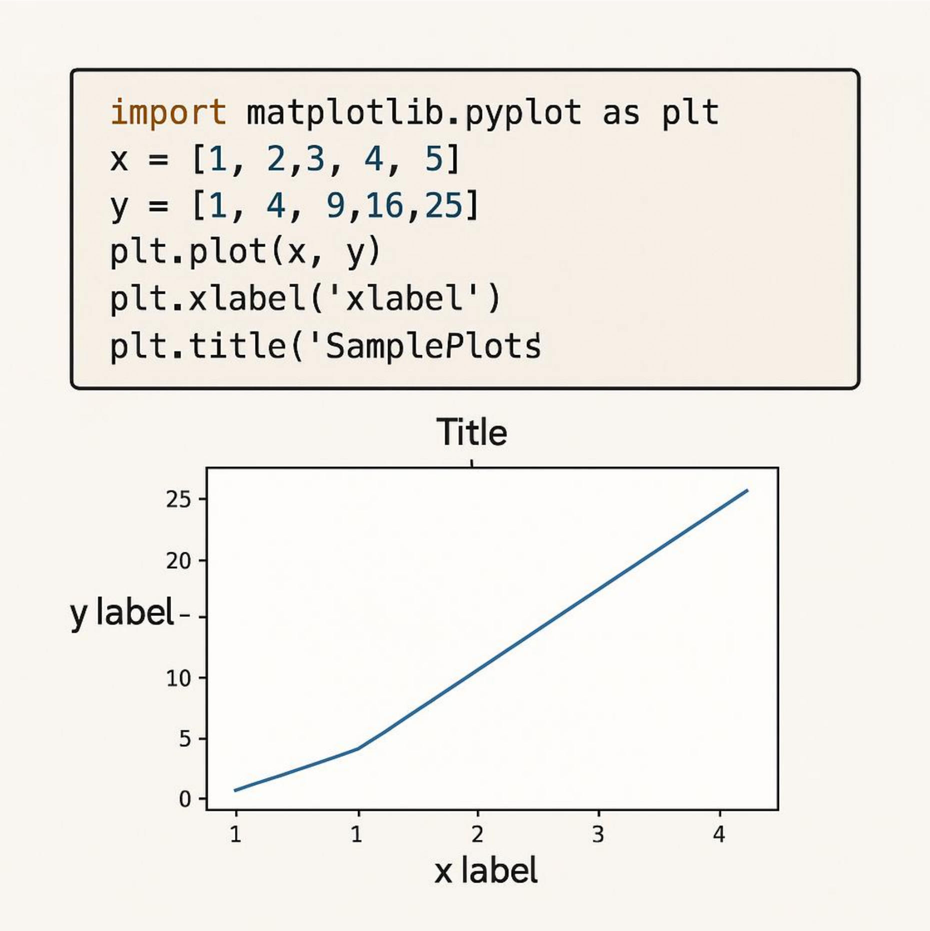
Common chart types: line plots, bar charts, histograms, scatter plots, box plots



Matplotlib: The Basics

- CORE PYPLOT FUNCTIONS: PLT.PLOT(), PLT.XLABEL(), PLT.YLABEL(), PLT.TITLE()

- CREATING SUBPLOTS AND ADDING ANNOTATIONS

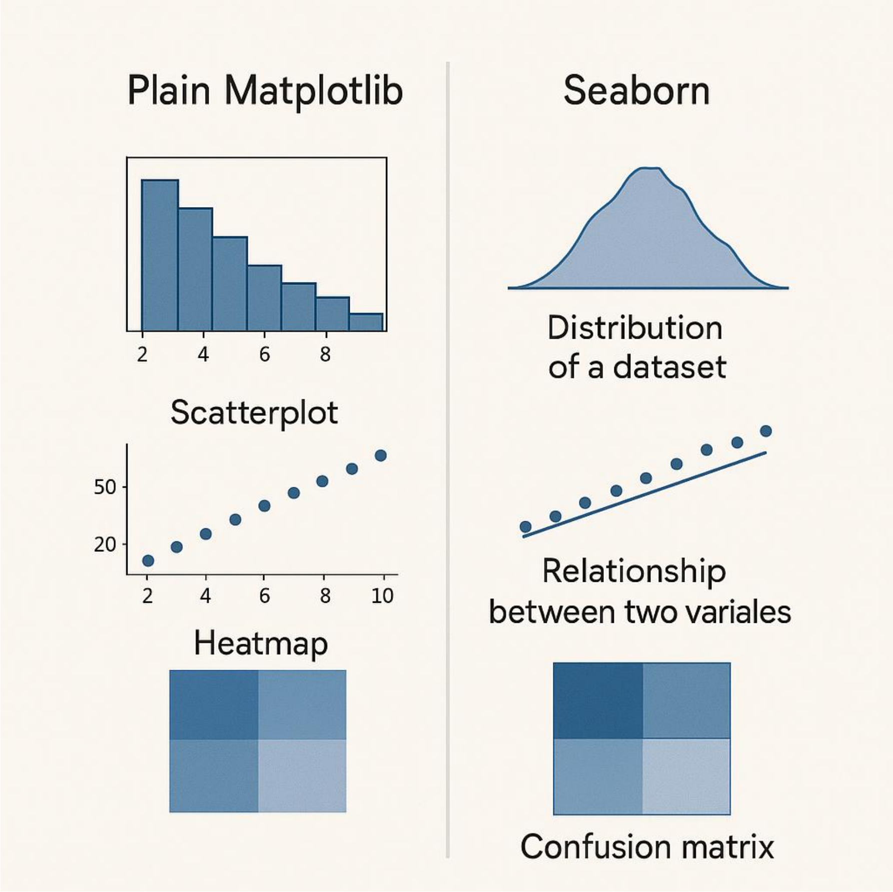


```
import matplotlib.pyplot as plt
x = [1, 2,3, 4, 5]
y = [1, 4, 9,16,25]
plt.plot(x, y)
plt.xlabel('xlabel')
plt.title('SamplePlots
```

Seaborn: Enhancing Visualisations

- **Generating plots with sns.histplot(), sns.scatterplot(), sns.boxplot(), sns.heatmap(), sns.pairplot()**

Advantages: Cleaner visualizations with minimal code



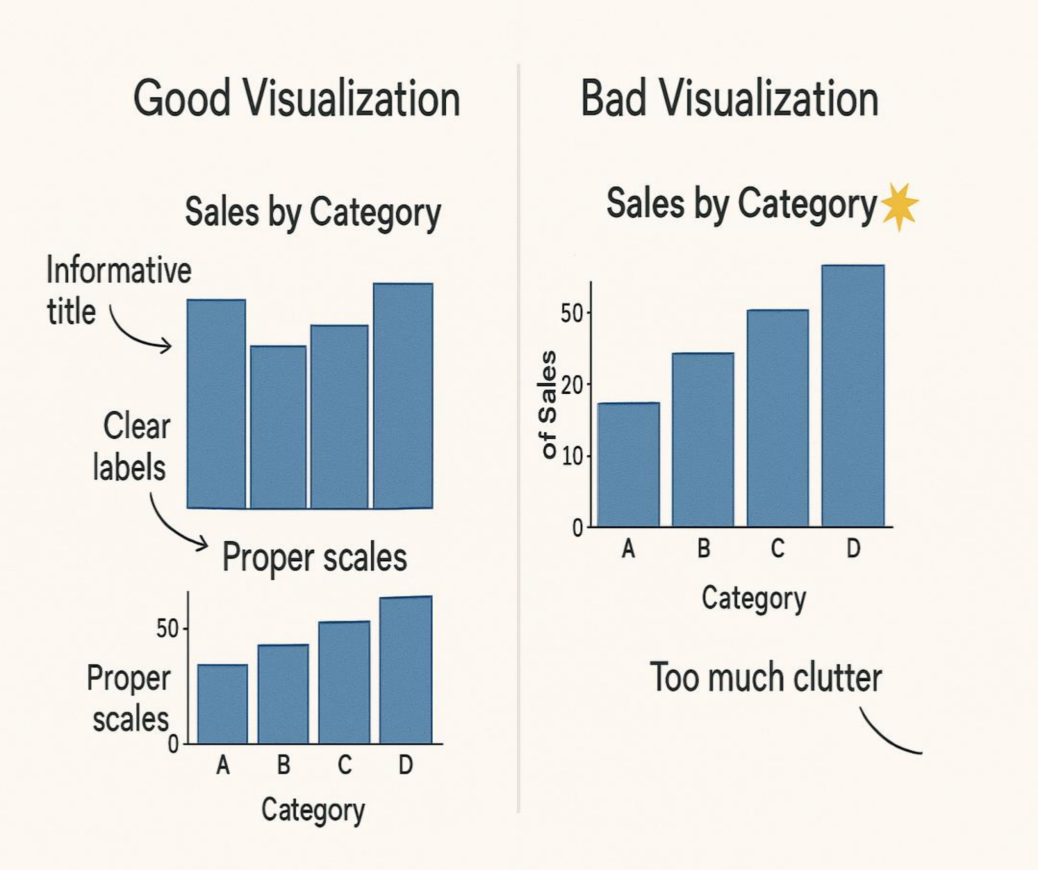
Good vs. Poor Data Visualization

1

Criteria for effective visualizations: clarity, accuracy, proper labeling, minimal clutter

2

Common pitfalls: Misleading scales, poor color choices, unnecessary effects

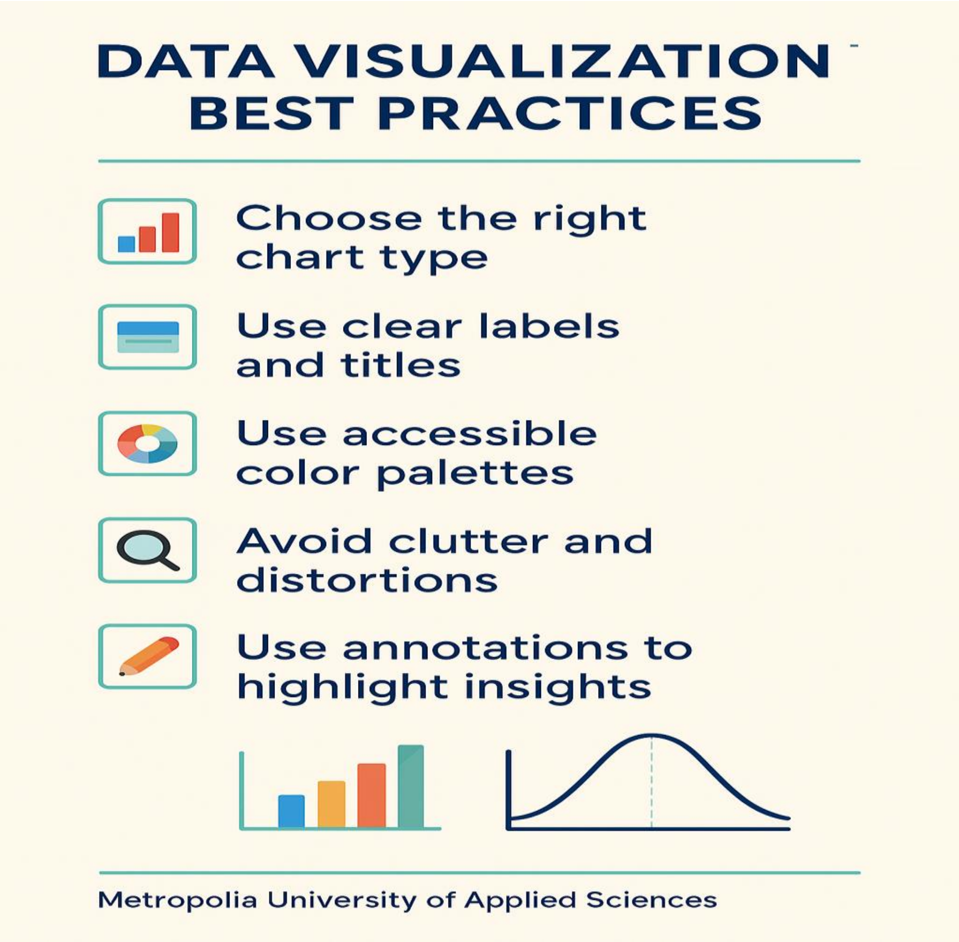


Best Practices for Data Visualization

Selecting the appropriate chart for your data

Using accessible color palettes and clear labels

The importance of annotation



DATA VISUALIZATION BEST PRACTICES

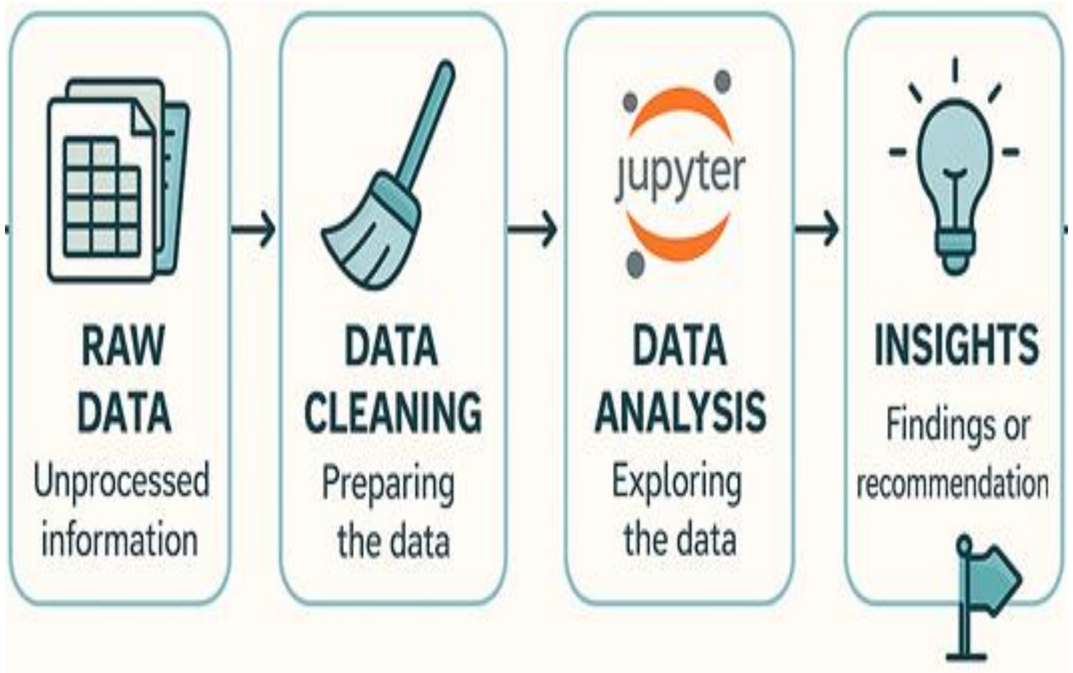
- Choose the right chart type
- Use clear labels and titles
- Use accessible color palettes
- Avoid clutter and distortions
- Use annotations to highlight insights

Metropolia University of Applied Sciences

Integrating Analysis & Visualization

END-TO-END WORKFLOW: DATA CLEANING → STATISTICAL ANALYSIS → VISUALIZATION

REAL-WORLD EXAMPLES OF ACTIONABLE INSIGHTS



RAW DATA
Unprocessed information

DATA CLEANING
Preparing the data

DATA ANALYSIS
Exploring the data

INSIGHTS
Findings or recommendation

Recap of Key Python Functions & Methods

01

NumPy essentials: np.array(), np.mean(), np.reshape()

02

Pandas operations: DataFrame manipulation, df.describe(), df.fillna()

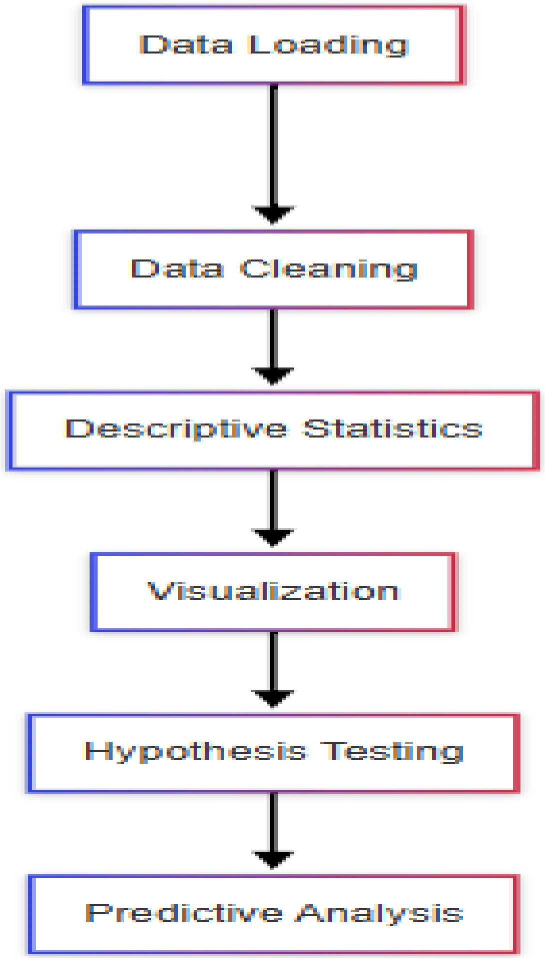
03

Visualization functions: plt.plot(), sns.heatmap(), sns.boxplot()

Cryptocurrency Mini-Project Overview

INTRODUCTION TO THE MINI-PROJECT

OBJECTIVES: ANALYZE HISTORICAL CRYPTOCURRENCY DATA (2015–2025) USING THE METHODS LEARNED



Data Loading

Data Cleaning

Descriptive Statistics

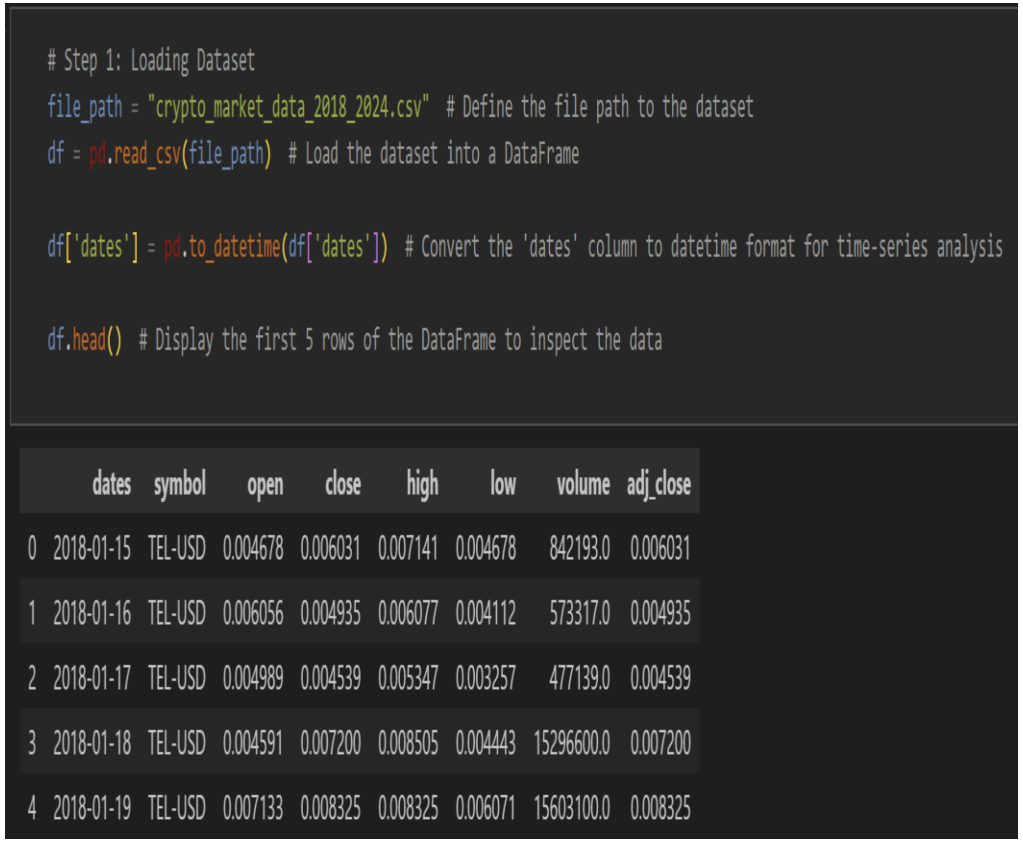
Visualization

Hypothesis Testing

Predictive Analysis

Load the Dataset and Explore Basic Information

Load	Load the cryptocurrency dataset (2015–2025)
Inspect	Inspect structure using .head() and .info()
Check	Check for missing values and data types
Initial	Initial shape and data cleaning steps



```
# Step 1: Loading Dataset
file_path = "crypto_market_data_2018_2024.csv"  # Define the file path to the dataset
df = pd.read_csv(file_path)  # Load the dataset into a DataFrame

df['dates'] = pd.to_datetime(df['dates'])  # Convert the 'dates' column to datetime format for time-series analysis

df.head()  # Display the first 5 rows of the DataFrame to inspect the data
```

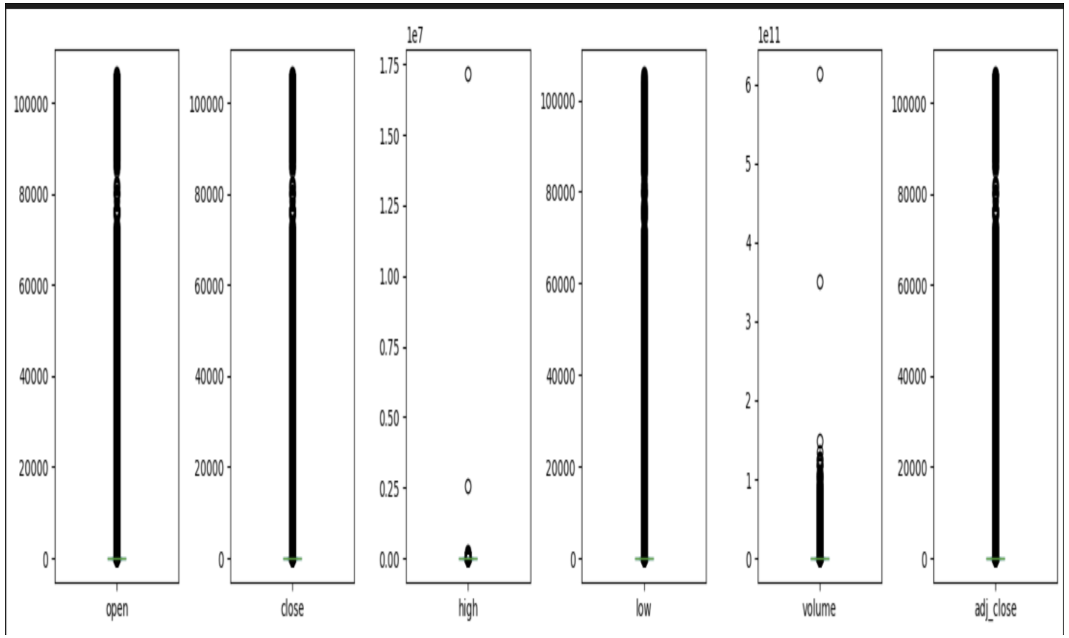
```
        dates   symbol      open     close      high       low      volume  adj_close
0  2018-01-15  TEL-USD  0.004678  0.006031  0.007141  0.004678    842193.0   0.006031
1  2018-01-16  TEL-USD  0.006056  0.004935  0.006077  0.004112    573317.0   0.004935
2  2018-01-17  TEL-USD  0.004989  0.004539  0.005347  0.003257    477139.0   0.004539
3  2018-01-18  TEL-USD  0.004591  0.007200  0.008505  0.004443  15296600.0   0.007200
4  2018-01-19  TEL-USD  0.007133  0.008325  0.008325  0.006071  15603100.0   0.008325
```

Perform a descriptive summary of the dataset

Basic metrics: mean, median, standard deviation of price, volume, and market cap

Identify trends by year and by cryptocurrency token

Detect unusual values or outliers using .describe() and visual tools (boxplots, z-scores)

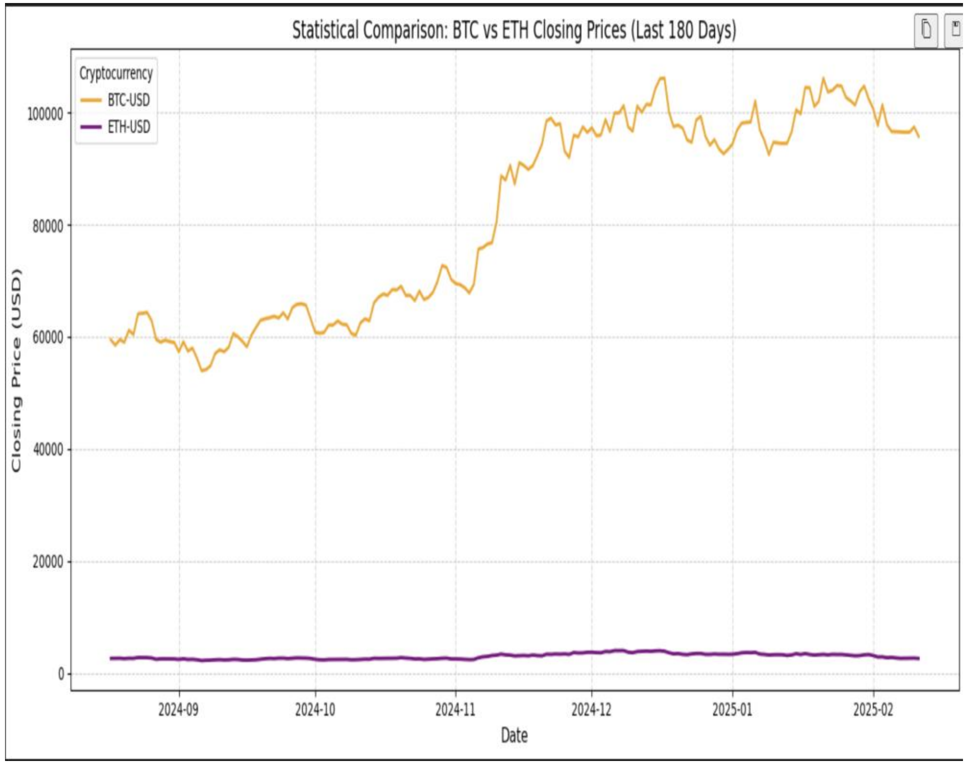


Perform a time-based analysis

GROUP PRICE, VOLUME, AND MARKET CAP DATA BY MONTH AND YEAR

VISUALIZE LONG-TERM TRENDS ACROSS 2015–2025

IDENTIFY **MAJOR SHIFTS** IN TOKEN PERFORMANCE OVER TIME (E.G., BULL/BEAR PHASES)



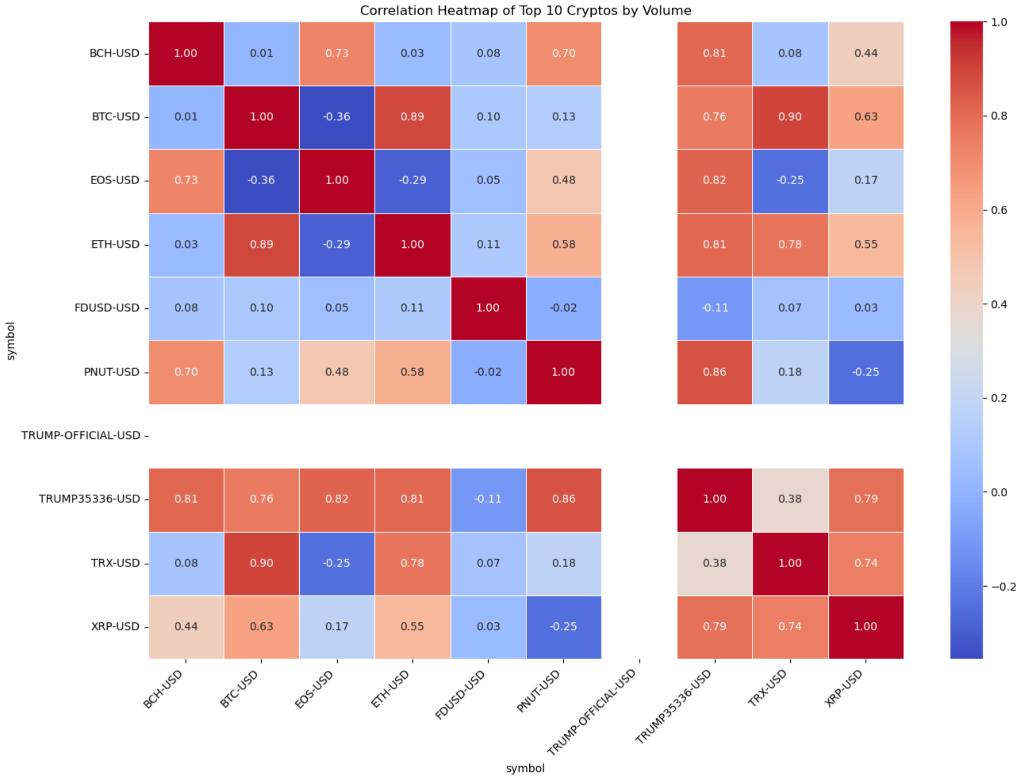
Visualize Data

Histograms: Token popularity and distribution

Boxplots: Detect outliers in price and volume

Line charts: Explore market trends from 2015 to 2025

Heatmaps: Visualize correlations among key features

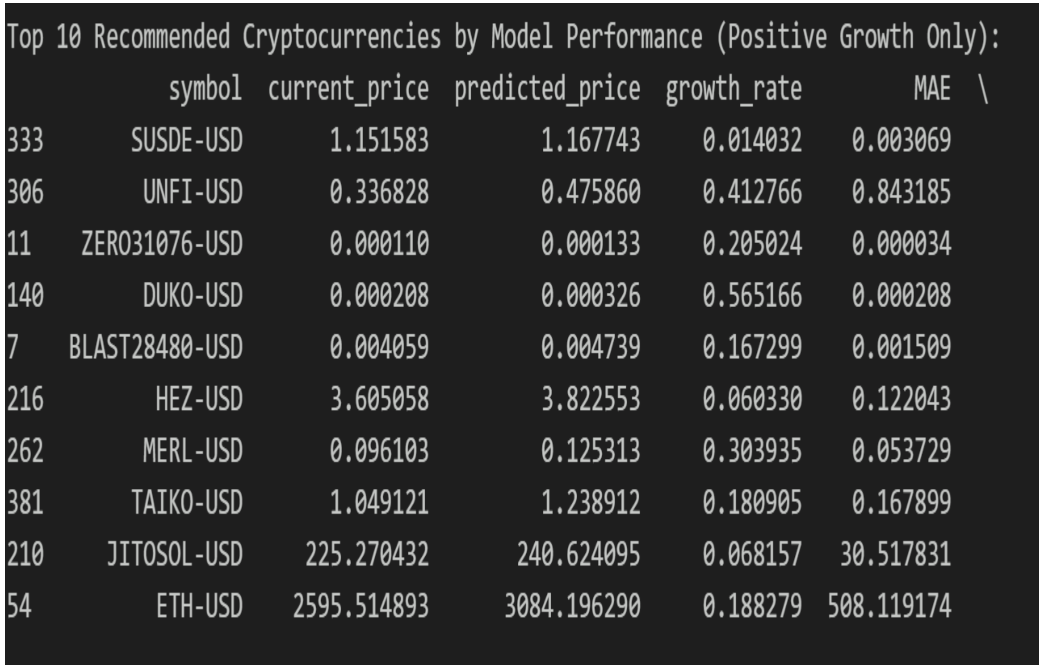


Predictive Analysis (Optional)

Identify variables affecting token price trends

Use **regression or time-series analysis** to model price movement

Make a final recommendation: Which token(s) might be profitable to invest in?

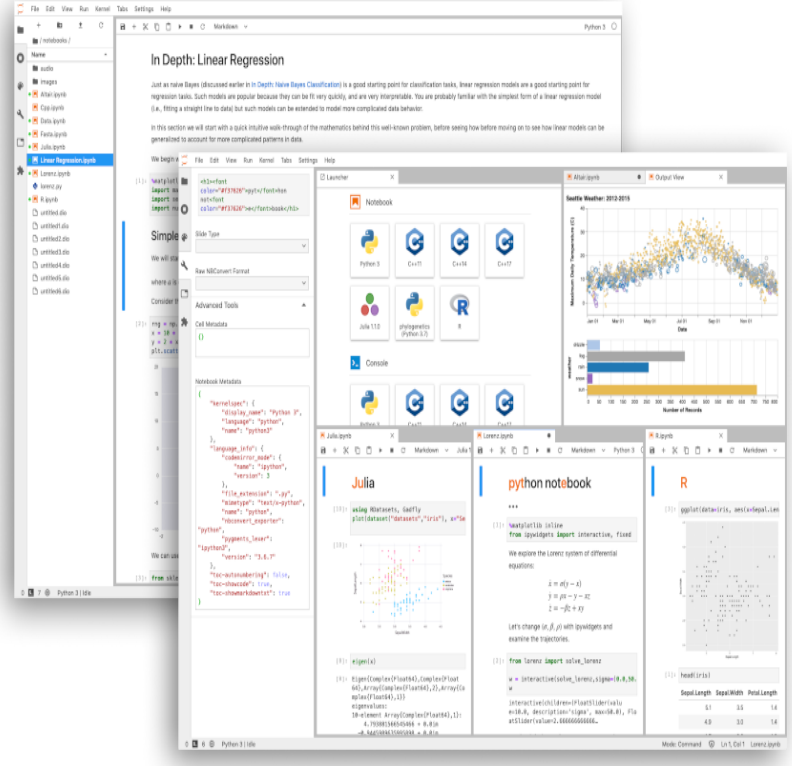


```
Top 10 Recommended Cryptocurrencies by Model Performance (Positive Growth Only):
              symbol  current_price  predicted_price  growth_rate         MAE  \
333        SUSDE-USD       1.151583         1.167743     0.014032    0.003069
306         UNFI-USD       0.336828         0.475860     0.412766    0.843185
11     ZERO31076-USD       0.000110         0.000133     0.205024    0.000034
140         DUKO-USD       0.000208         0.000326     0.565166    0.000208
7     BLAST28480-USD       0.004059         0.004739     0.167299    0.001509
216          HEZ-USD       3.605058         3.822553     0.060330    0.122043
262         MERL-USD       0.096103         0.125313     0.303935    0.053729
381        TAIKO-USD       1.049121         1.238912     0.180905    0.167899
210      JITOSOL-USD     225.270432       240.624095     0.068157   30.517831
54           ETH-USD    2595.514893      3084.196290     0.188279  508.119174
```

Notebook Review

Notebook Walk-through

- **Project Title**: Cryptocurrency Historical Data Analysis
- **Dataset**: Crypto historical data (2015–2025)
- **Goals**:
  - Clean and preprocess data
  - Compute descriptive statistics and visualize trends
  - Conduct hypothesis testing on market behavior
  - Develop predictive models for token price movement
  - Deliver actionable recommendations for potential profitable investments



Kahoot Quiz Time!

Let’s Test Our Knowledge!

Reference

- Vohra, M., & Patil, B. (2021). A Walk Through the World of Data Analytics. , 19-27. https://doi.org/10.4018/978-1-7998-3053-5.ch002.
- VanderPlas, J. (2016). Python data science handbook: Essential tools for working with data. O'Reilly Media. Available at https://jakevdp.github.io/PythonDataScienceHandbook/
- Severance, C. (2016). Python for everybody: Exploring data using Python 3. Charles Severance. Available at https://www.py4e.com/html3/
- McKinney, W. (2017). *Python for data analysis: Data wrangling with pandas, NumPy, and Jupyter*. O'Reilly Media.