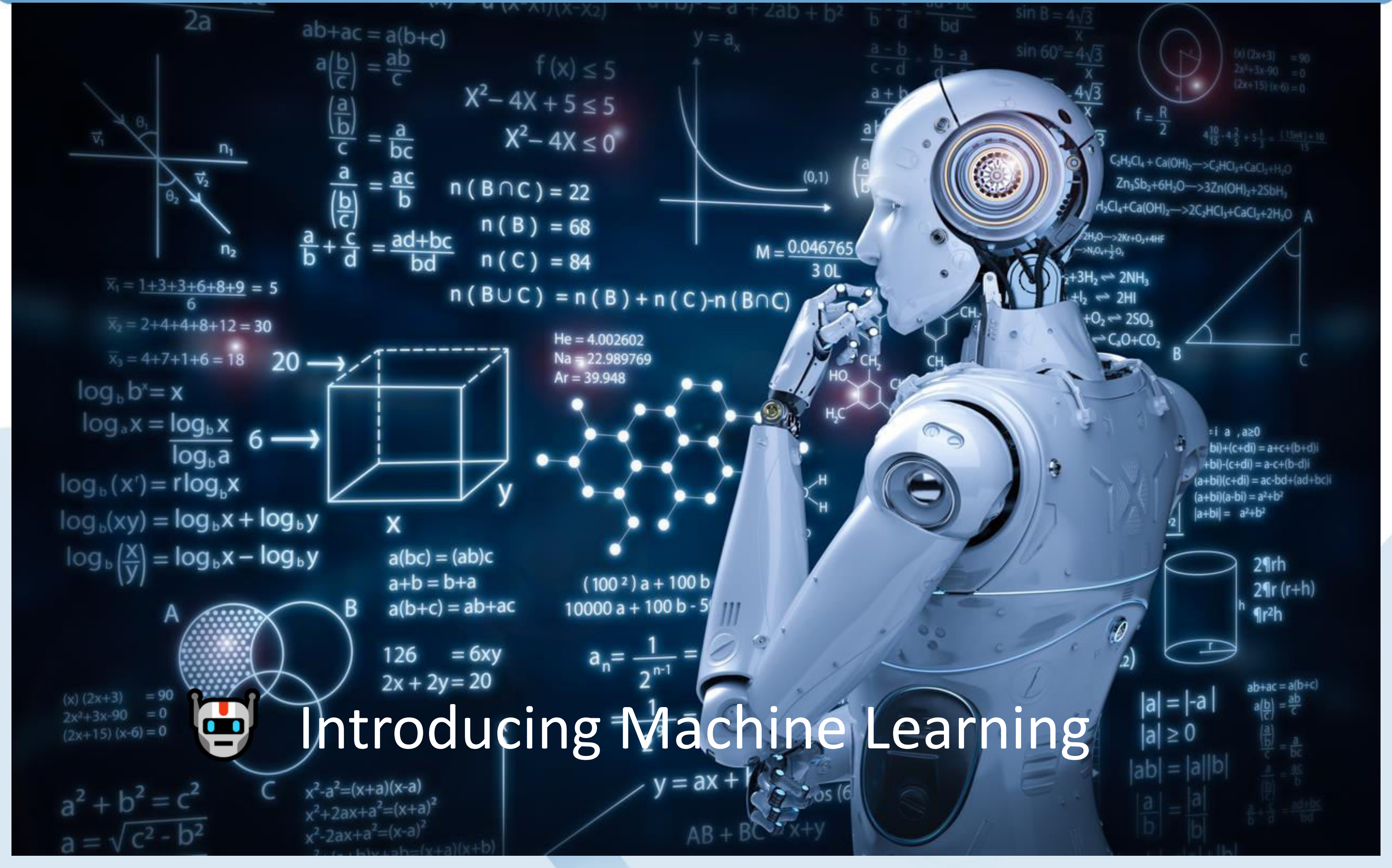
Machine Learning


ab+ac = a(b+c)
f (x) ≤ 5
$X^2 - 4X + 5 \le 5$
$X^2 - 4X \le 0$
n (B∩C) = 22
n (B) = 68
n (C) = 84
n (B∪C) = n (B) + n (C)-n (B∩C)
$\frac{a}{b} + \frac{c}{d} = \frac{ad+bc}{bd}$
$\bar{x}_1 = \frac{1+3+3+6+8+9}{6} = 5$
$\bar{x}_2 = 2+4+4+8+12 = 30$
$\bar{x}_3 = 4+7+1+6 = 18$
$\log_b b^x = x$
$\log_a x = \frac{\log_b x}{\log_b a}$
$\log_b (x^r) = r\log_b x$
$\log_b(xy) = \log_b x + \log_b y$
$\log_b \left(\frac{x}{y}\right) = \log_b x - \log_b y$
20
6
x
y
a(bc) = (ab)c
a+b = b+a
a(b+c) = ab+ac
126 = 6xy
2x + 2y= 20
He = 4.002602
Na = 22.989769
Ar = 39.948
$M = \frac{0.046765}{3\ 0L}$
(0,1)
y = a
A
B
C
$a^2 + b^2 = c^2$
$a = \sqrt{c^2 - b^2}$
$x^2 - a^2 = (x+a)(x-a)$
$x^2 + 2ax + a^2 = (x+a)^2$
$x^2 - 2ax + a^2 = (x-a)^2$
(x) (2x+3) = 90
$2x^2+3x-90 = 0$
(2x+15) (x-6) = 0
$a_n = \frac{1}{2^{n-1}}$
y = ax +
AB + BC
$|a| = |-a|$
$|a| \ge 0$
$|ab| = |a||b|$
$f = \frac{R}{2}$
$\sin 60° = \frac{4\sqrt{3}}{x}$

Introducing Machine Learning

© TemplatesWise.com

About the Author

- Created By: Mohammad Salman
- Experience: 19 Years +
- Designation: Corporate   Trainer .NET

MICROSOFT CERTIFIED TRAINER
Microsoft
MCT
Trainer
2022 - 2023

Microsoft
CERTIFIED
Professional

Microsoft
Specialist

Microsoft
CERTIFIED
Solutions Developer

Icons Used

Questions

Tools

Hands-on Exercise

Coding Standards

Questions?

Reference

Try it Out

Informative Slide

Mandatory Slide

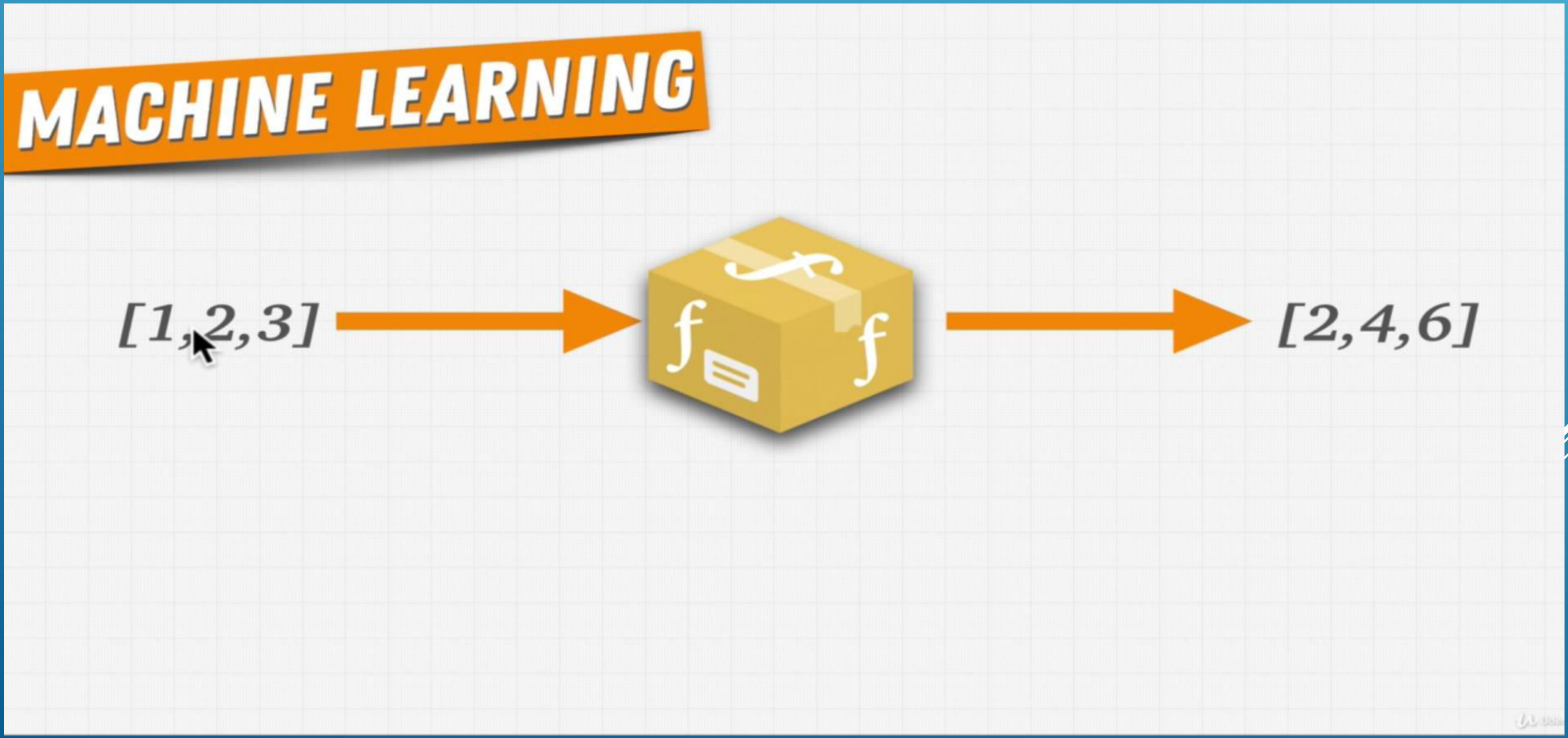
Welcome Break

- Machine Learning (ML) is a subset of Artificial Intelligence that enables systems to learn from data and make predictions without explicit programming.

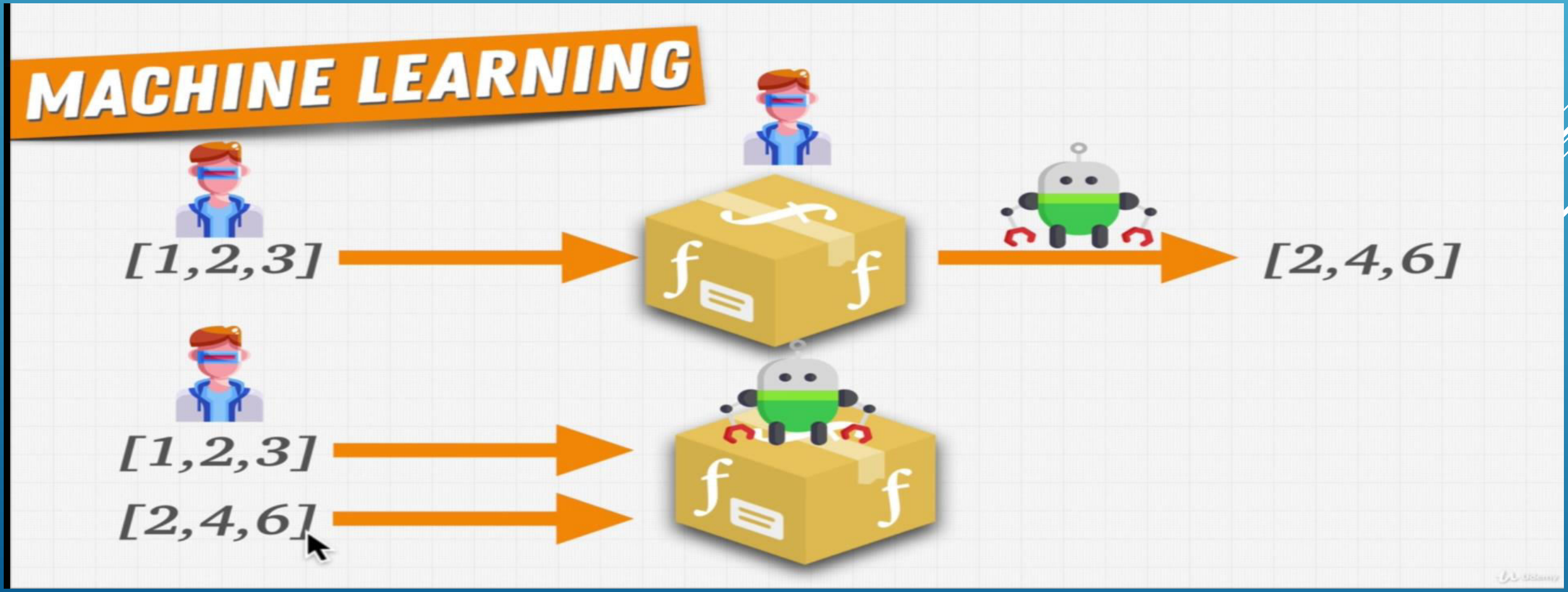
- Flow diagram → Data → Model → Prediction → FeedbackKey Point:
- ML = Learning patterns from data.

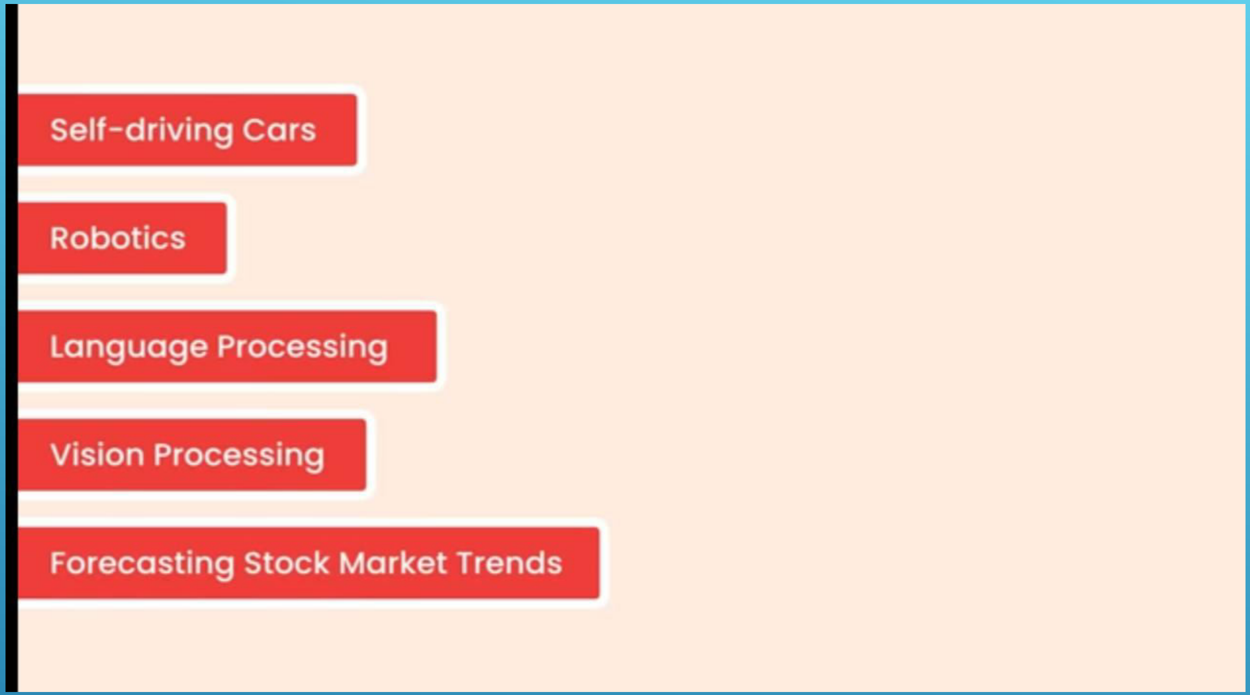
WHAT IS MACHINE LEARNING

- In traditional programming, a human writes the rule or formula (f) that transforms input to output
- [1,2,3] → Function (f) → [2,4,6]
- The input data [1,2,3] goes into a function or algorithm (f).The output is [2,4,6].



- One represents the human-written function.
- The next represents the machine-learned function.
- The robot now replaces the human’s role in creating the function





Self-driving Cars

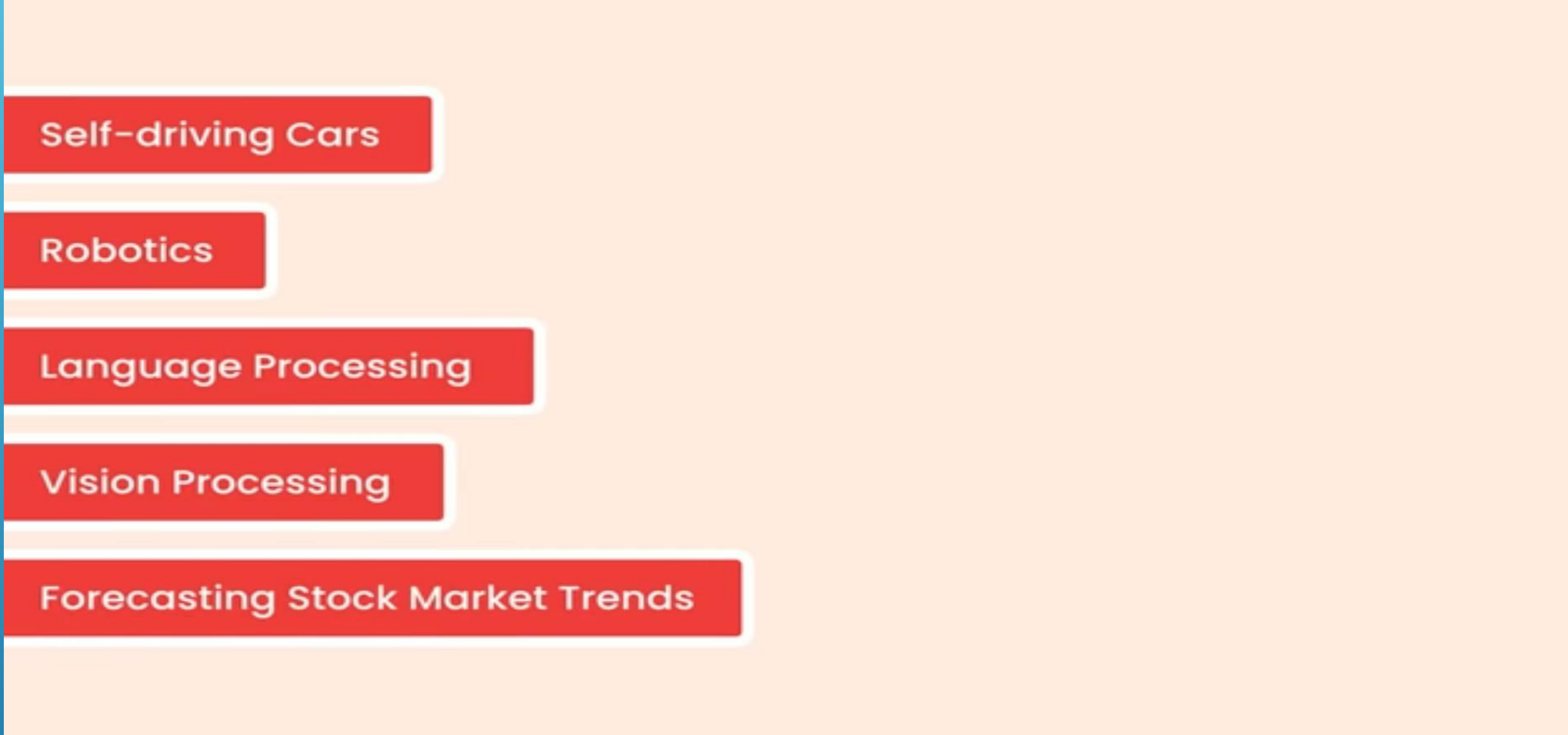
Robotics

Language Processing

Vision Processing

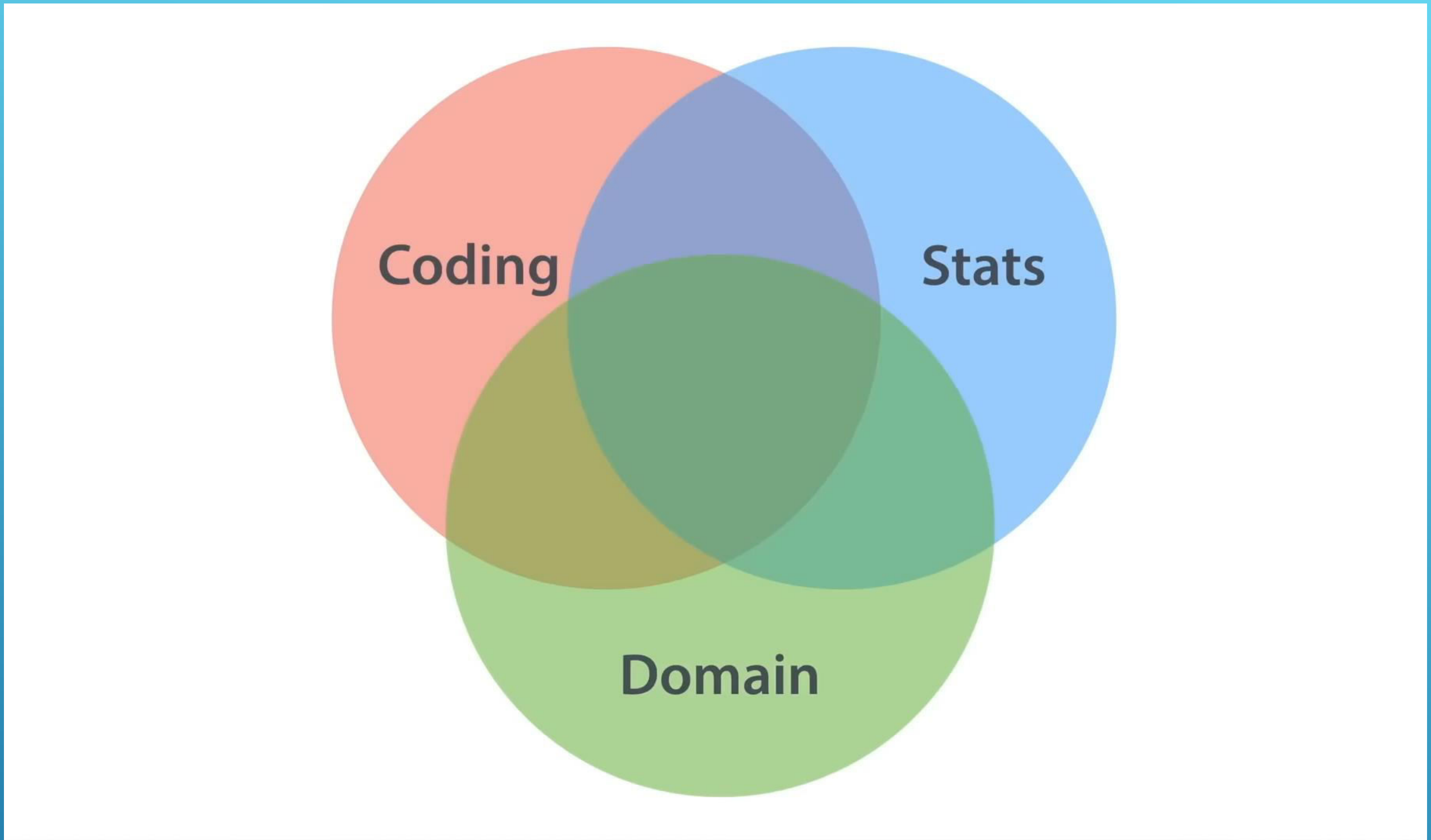
Forecasting Stock Market Trends

WHAT IS MACHINE LEARNING

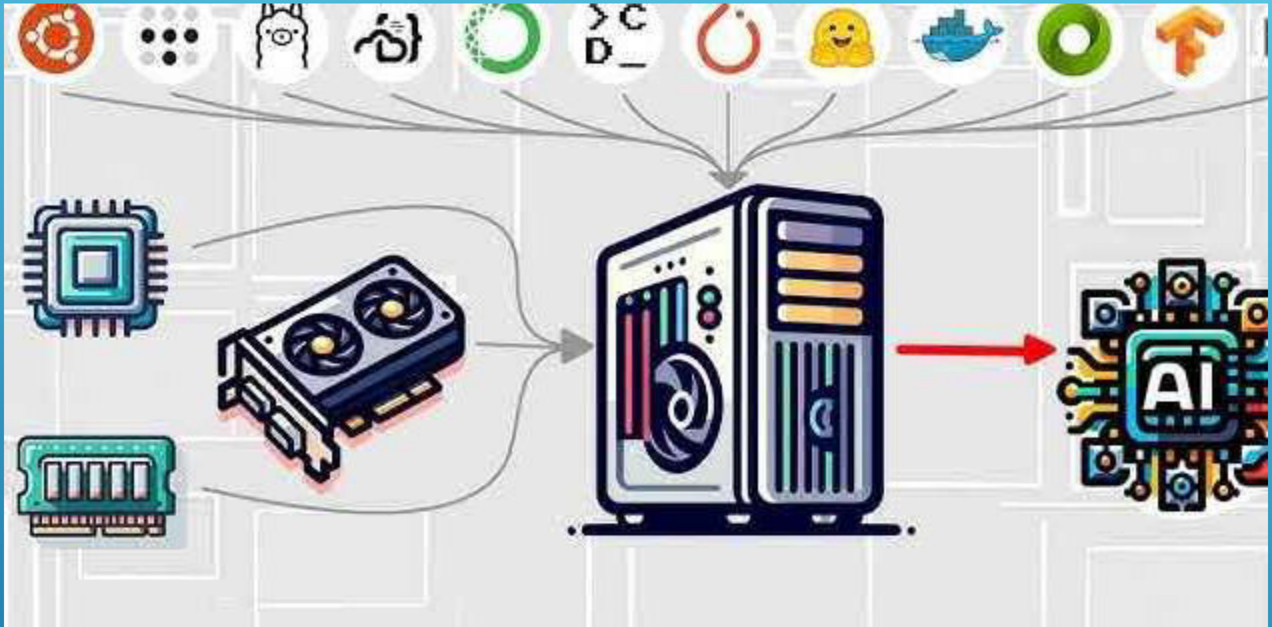


- Self-driving Cars
- Robotics
- Language Processing
- Vision Processing
- Forecasting Stock Market Trends

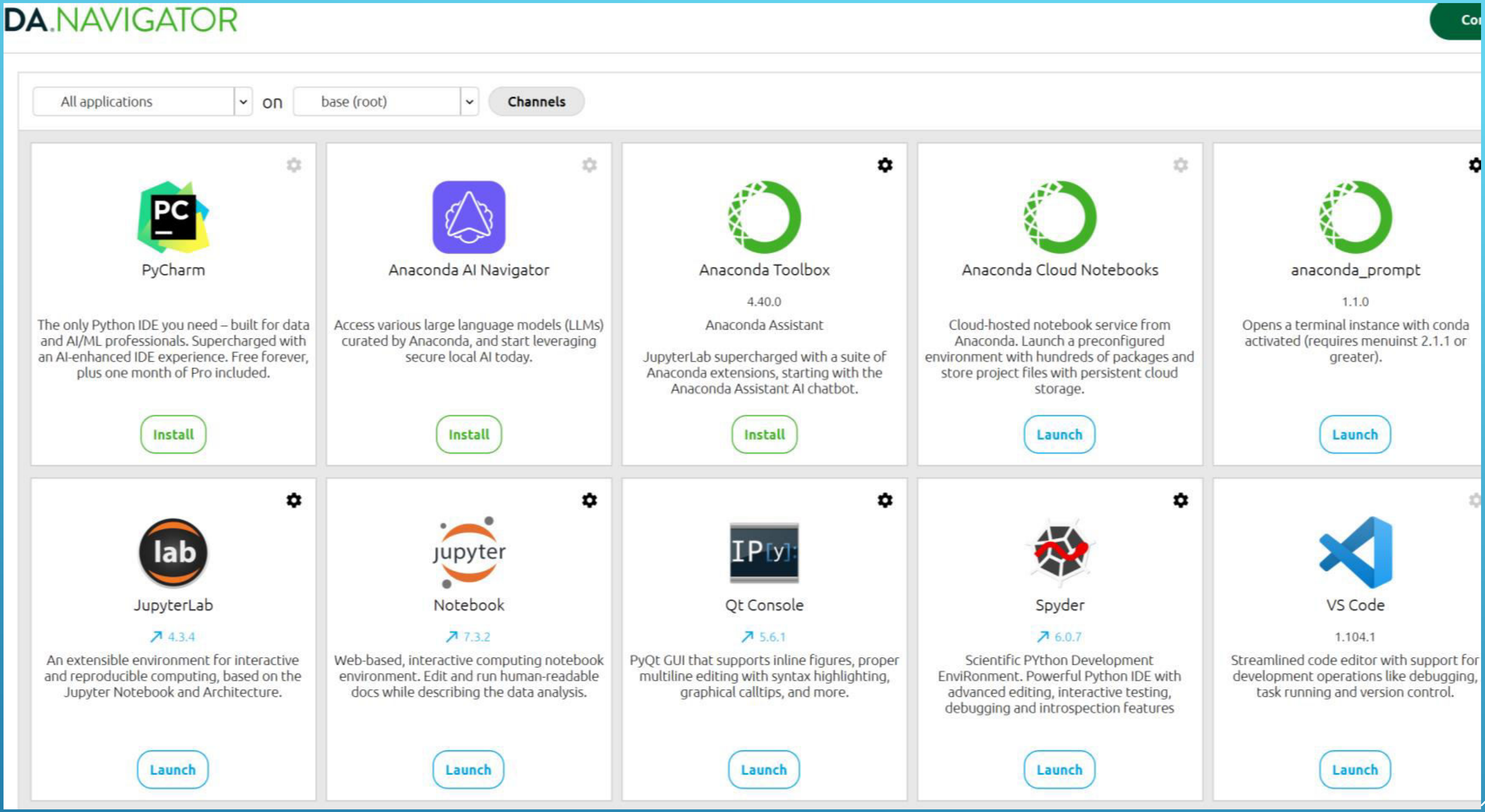
MACHINE LEARNING USAGE



MACHINE LEARNING(TO LEARN)



MACHINE LEARNING SOFTWARE



All applications on base (root)

PyCharm

The only Python IDE you need – built for data and AI/ML professionals. Supercharged with an AI-enhanced IDE experience. Free forever, plus one month of Pro included.

Anaconda AI Navigator

Access various large language models (LLMs) curated by Anaconda, and start leveraging secure local AI today.

Anaconda Toolbox

4.40.0

Anaconda Assistant

JupyterLab supercharged with a suite of Anaconda extensions, starting with the Anaconda Assistant AI chatbot.

Anaconda Cloud Notebooks

Cloud-hosted notebook service from Anaconda. Launch a preconfigured environment with hundreds of packages and store project files with persistent cloud storage.

anaconda_prompt

1.1.0

Opens a terminal instance with conda activated (requires menuinst 2.1.1 or greater).

JupyterLab

↗ 4.3.4

An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook

↗ 7.3.2

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Qt Console

↗ 5.6.1

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder

↗ 6.0.7

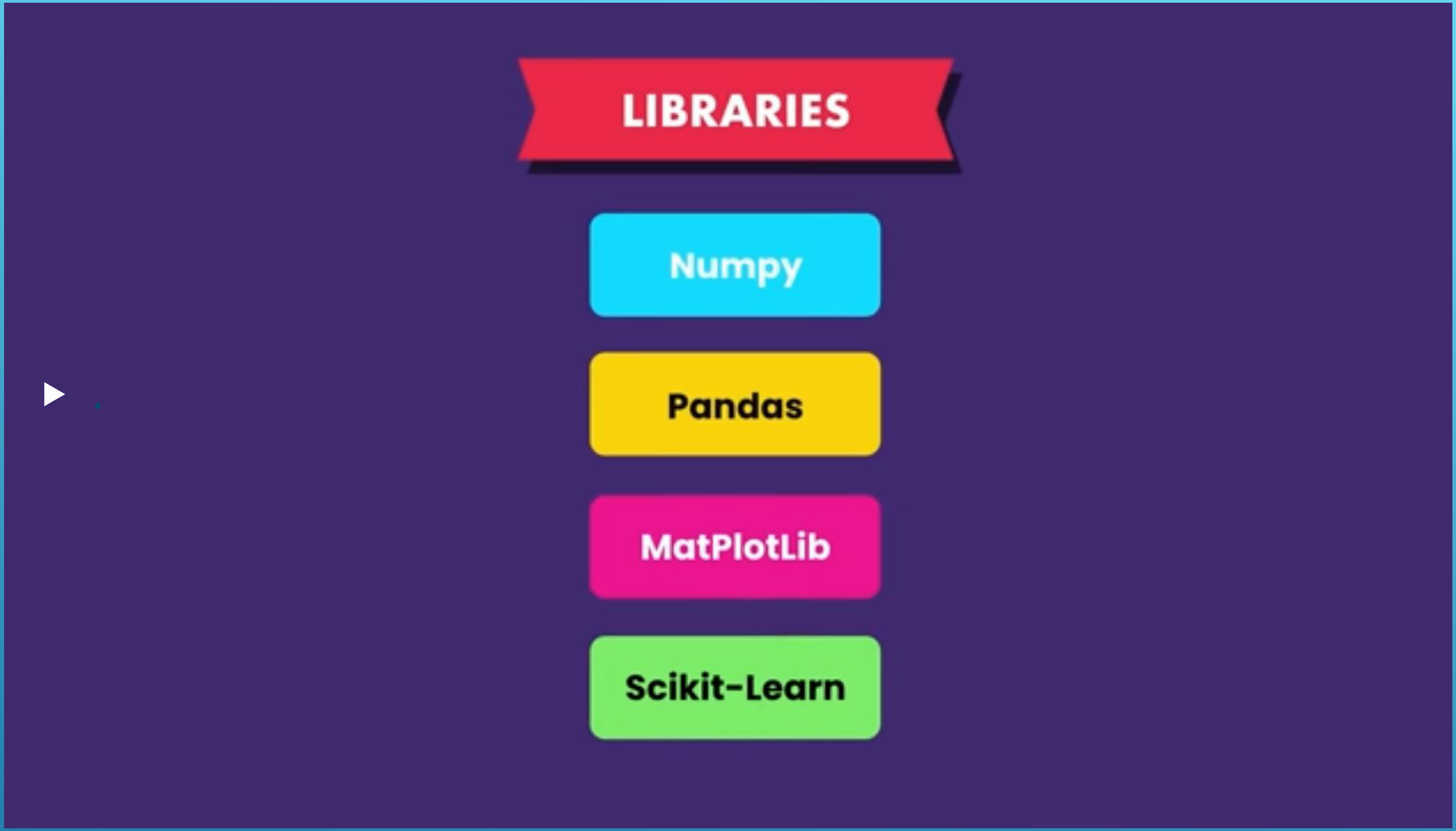
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

VS Code

1.104.1

Streamlined code editor with support for development operations like debugging, task running and version control.

MACHINE LEARNING SOFTWARE



LIBRARIES

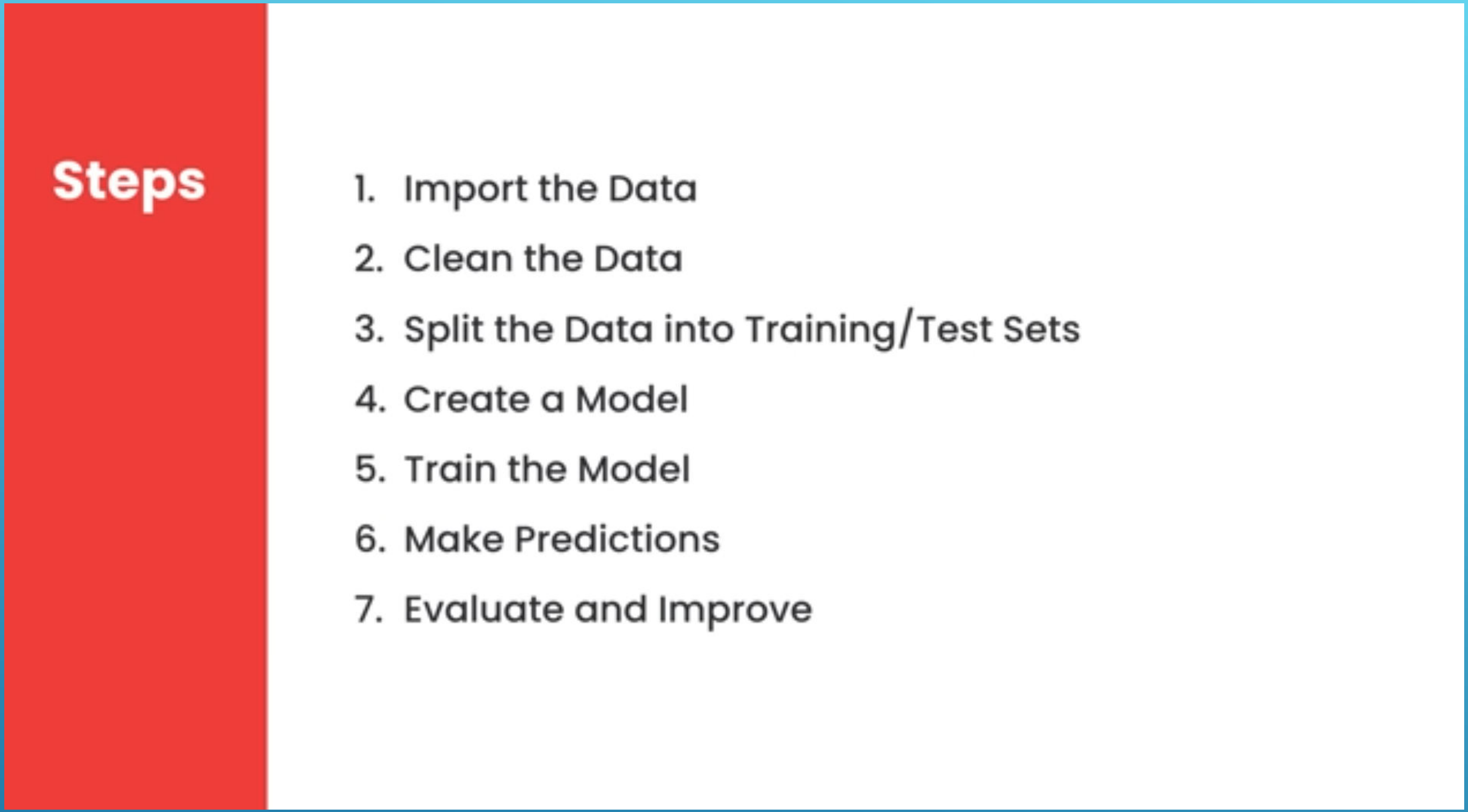
Numpy

Pandas

MatPlotLib

Scikit-Learn

WHAT IS MACHINE LEARNING



Steps

1. Import the Data
2. Clean the Data
3. Split the Data into Training/Test Sets
4. Create a Model
5. Train the Model
6. Make Predictions
7. Evaluate and Improve

WHAT IS MACHINE LEARNING

1. Getting the data ready

Collect or load your dataset.

Clean, preprocess, and split it into features (X) and labels (y).

2.Choosing the right estimator/algorithm

Pick the appropriate ML model (e.g., Linear Regression, Decision Tree, KNN).

3.Fitting the model

Train it on your data so it learns the patterns.

4. Evaluating the model

Measure accuracy, precision, recall, or other metrics.

5. Improving the model

Tune hyperparameters or use better features.

6. Saving and loading the model

Store the trained model for future predictions.

7. Putting it all together

Deploy or integrate it into an application.

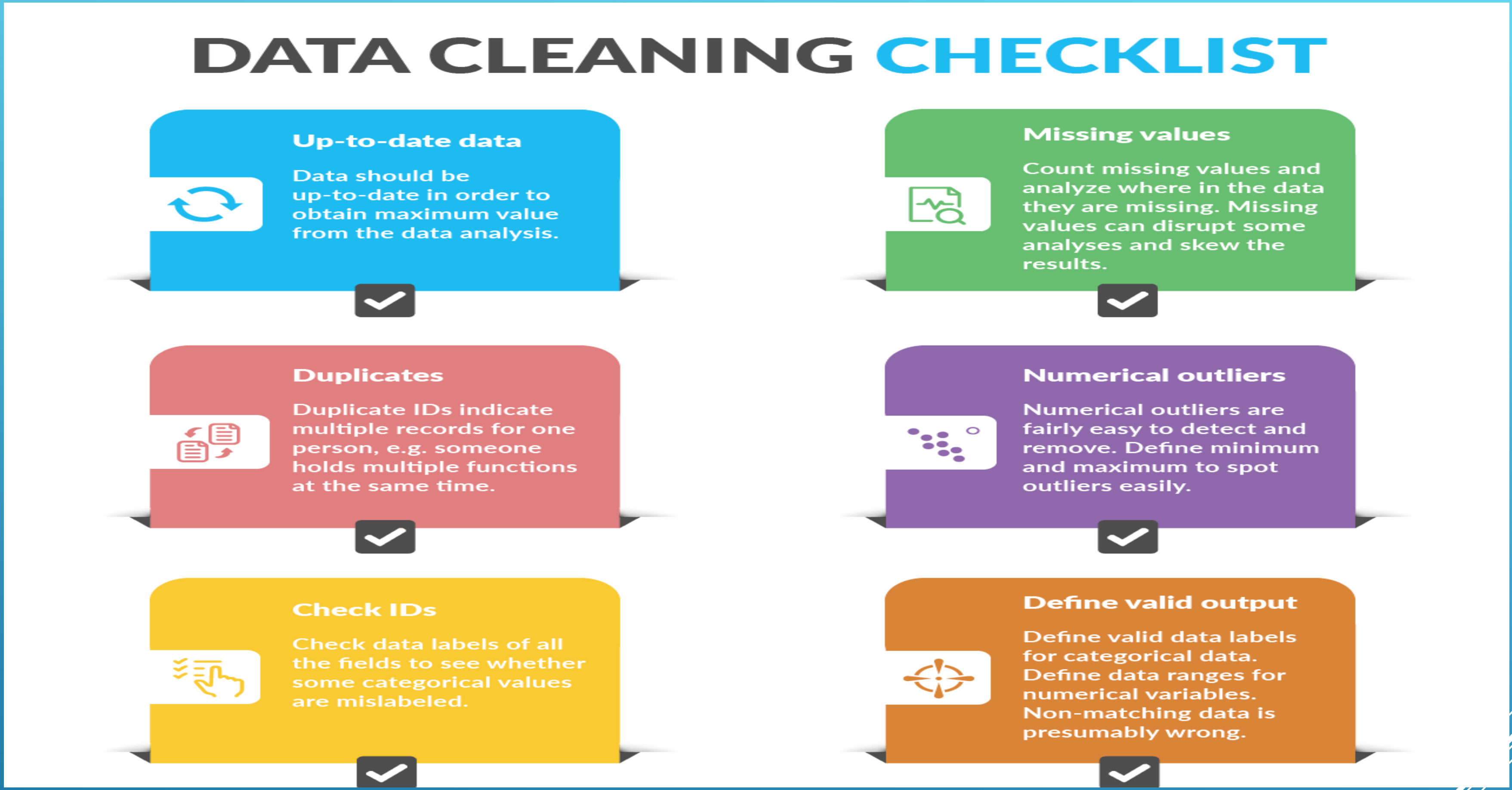
THE SCIKIT-LEARN WORKFLOW

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y)
```

THE SCIKIT-LEARN WORKFLOW

```
import pandas as pd
df = pd.read_csv("students_scores.csv")
df.isnull().sum()
df.fillna(df.mean(), inplace=True)
```

WHAT IS DATA CLEANING



DATA CLEANING CHECKLIST

Up-to-date data

Data should be up-to-date in order to obtain maximum value from the data analysis.

Missing values

Count missing values and analyze where in the data they are missing. Missing values can disrupt some analyses and skew the results.

Duplicates

Duplicate IDs indicate multiple records for one person, e.g. someone holds multiple functions at the same time.

Numerical outliers

Numerical outliers are fairly easy to detect and remove. Define minimum and maximum to spot outliers easily.

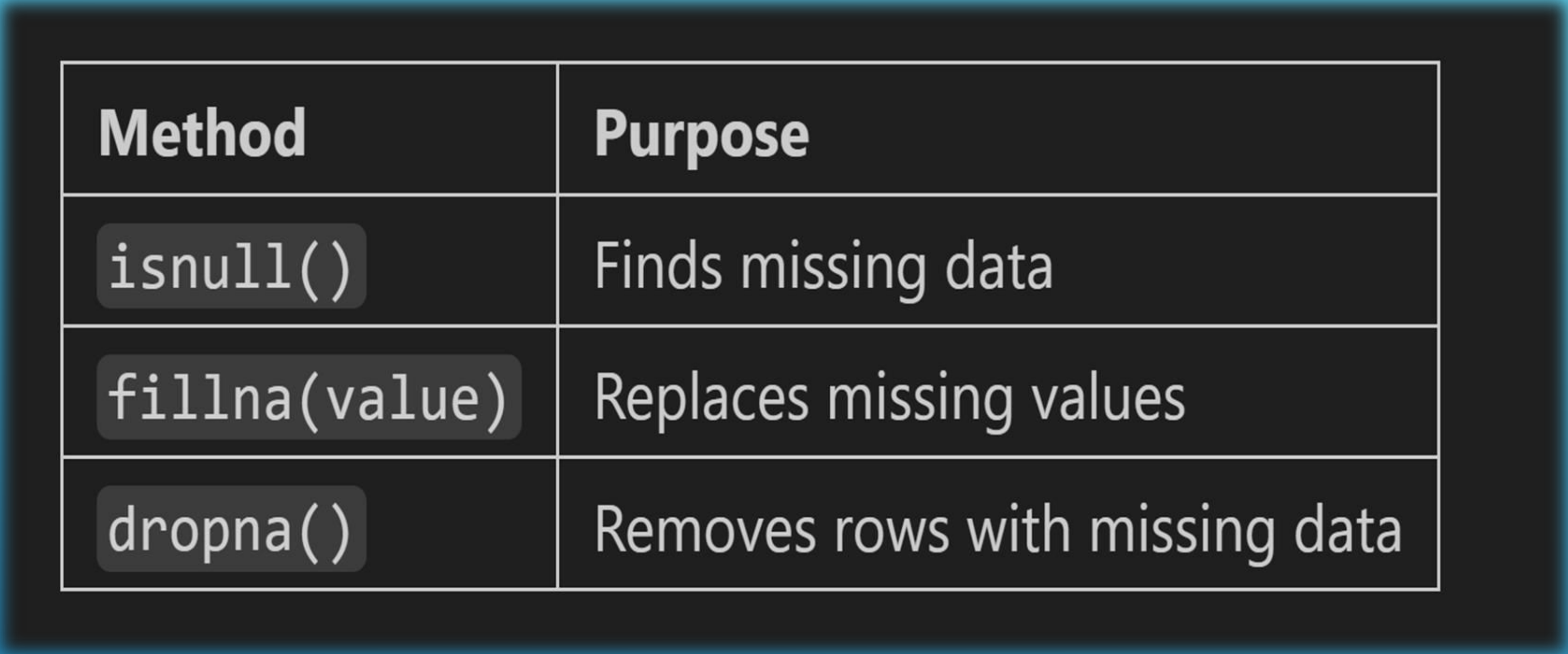
Check IDs

Check data labels of all the fields to see whether some categorical values are mislabeled.

Define valid output

Define valid data labels for categorical data. Define data ranges for numerical variables. Non-matching data is presumably wrong.

WHAT IS DATA CLEANING



Method	Purpose
`isnull()`	Finds missing data
`fillna(value)`	Replaces missing values
`dropna()`	Removes rows with missing data

WHAT IS DATA CLEANING

converts categorical values into numeric codes (integers).
Each unique category is assigned a number

LABEL ENCODING

```
from sklearn.preprocessing import LabelEncoder
# Sample data
colors = ['Red', 'Blue', 'Green', 'Blue', 'Red']
# Create encoder
le = LabelEncoder()
# Fit and transform
encoded = le.fit_transform(colors)
print(encoded)
```

LABEL ENCODING

One-Hot Encoding converts categorical variables into binary columns (0 or 1) — one for each category.

It prevents the model from assuming any ordinal relationship between categories.

ONE-HOT ENCODING

```
from sklearn.preprocessing import OneHotEncoder

import pandas as pd

# Sample data

data = pd.DataFrame({'Color': ['Red', 'Blue', 'Green', 'Blue', 'Red']})

# Create encoder

ohe = OneHotEncoder(sparse_output=False)

# Fit and transform

encoded = ohe.fit_transform(data[['Color']])

encoded_df = pd.DataFrame(encoded,
columns=ohe.get_feature_names_out(['Color']))

print(encoded_df)
```

ONE-HOT ENCODING

Feature	Label Encoding	One-Hot Encoding
Output	Single numeric column	Multiple binary columns
Example	Red → 2	Red → [0,0,1]
Suitable for	Ordinal data	Nominal data
Risk	Implies order relationship	No order assumption
Libraries	`LabelEncoder`	`OneHotEncoder, pd.get_dummies()`

LABEL ENCODING VS ONE-HOT ENCODING

LABEL ENCODING

```
from sklearn.preprocessing import LabelEncoder
import pandas as pd

data = pd.DataFrame({'Satisfaction': ['Low', 'Medium', 'High', 'Low']})
le = LabelEncoder()
data['Satisfaction_Encoded'] = le.fit_transform(data['Satisfaction'])
print(data)
```

LABEL ENCODING

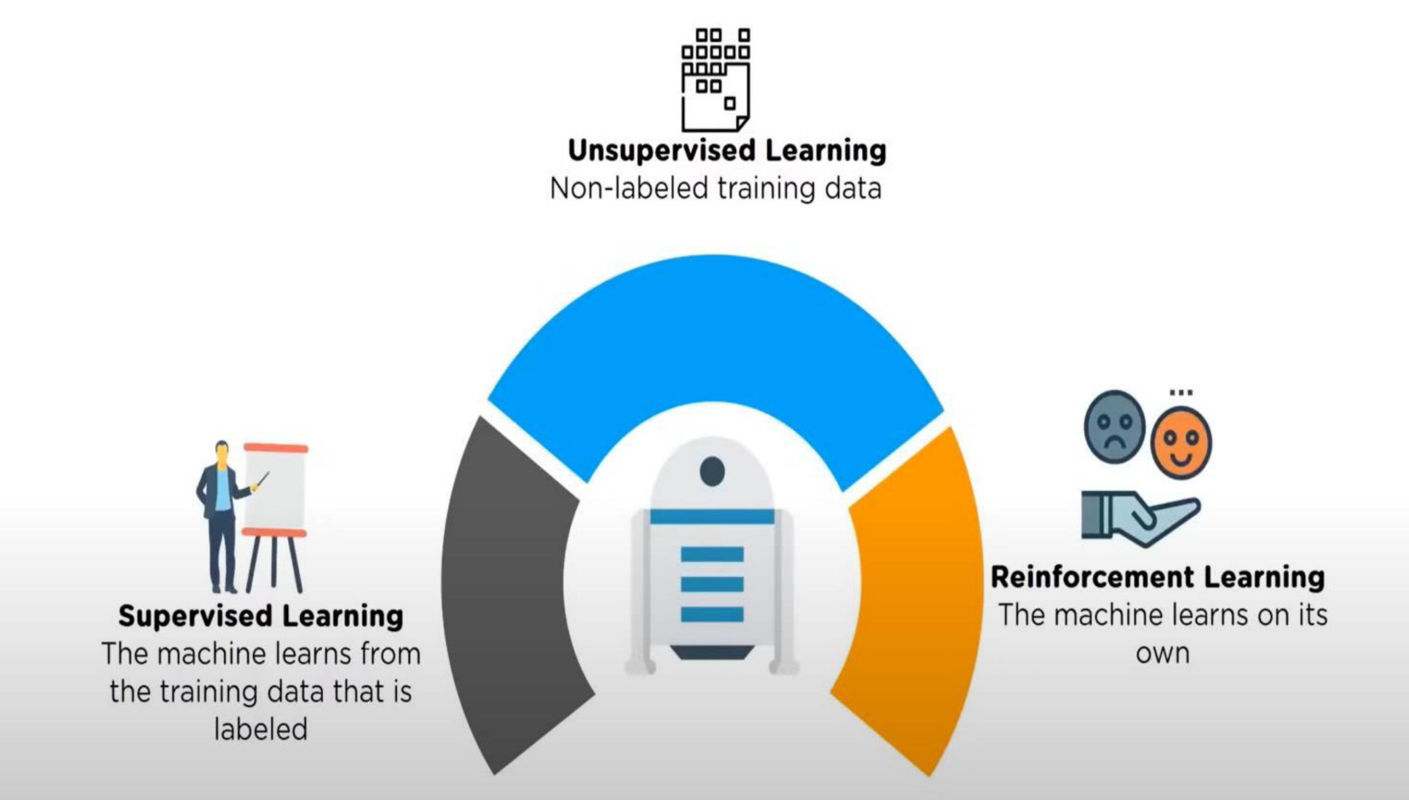
```
from sklearn.preprocessing import LabelEncoder
import pandas as pd

data = pd.DataFrame({'Satisfaction': ['Low', 'Medium', 'High', 'Low']})
le = LabelEncoder()
data['Satisfaction_Encoded'] = le.fit_transform(data['Satisfaction'])
print(data)
```

ONE-HOT ENCODING

```
import pandas as pd
from sklearn.preprocessing import OneHotEncoder

data = pd.DataFrame({'Region': ['North', 'South', 'East', 'West', 'South']})
ohe = OneHotEncoder(sparse_output=False)
encoded = ohe.fit_transform(data[['Region']])
encoded_df = pd.DataFrame(encoded, columns=ohe.get_feature_names_out(['Region']))
print(encoded_df)
```

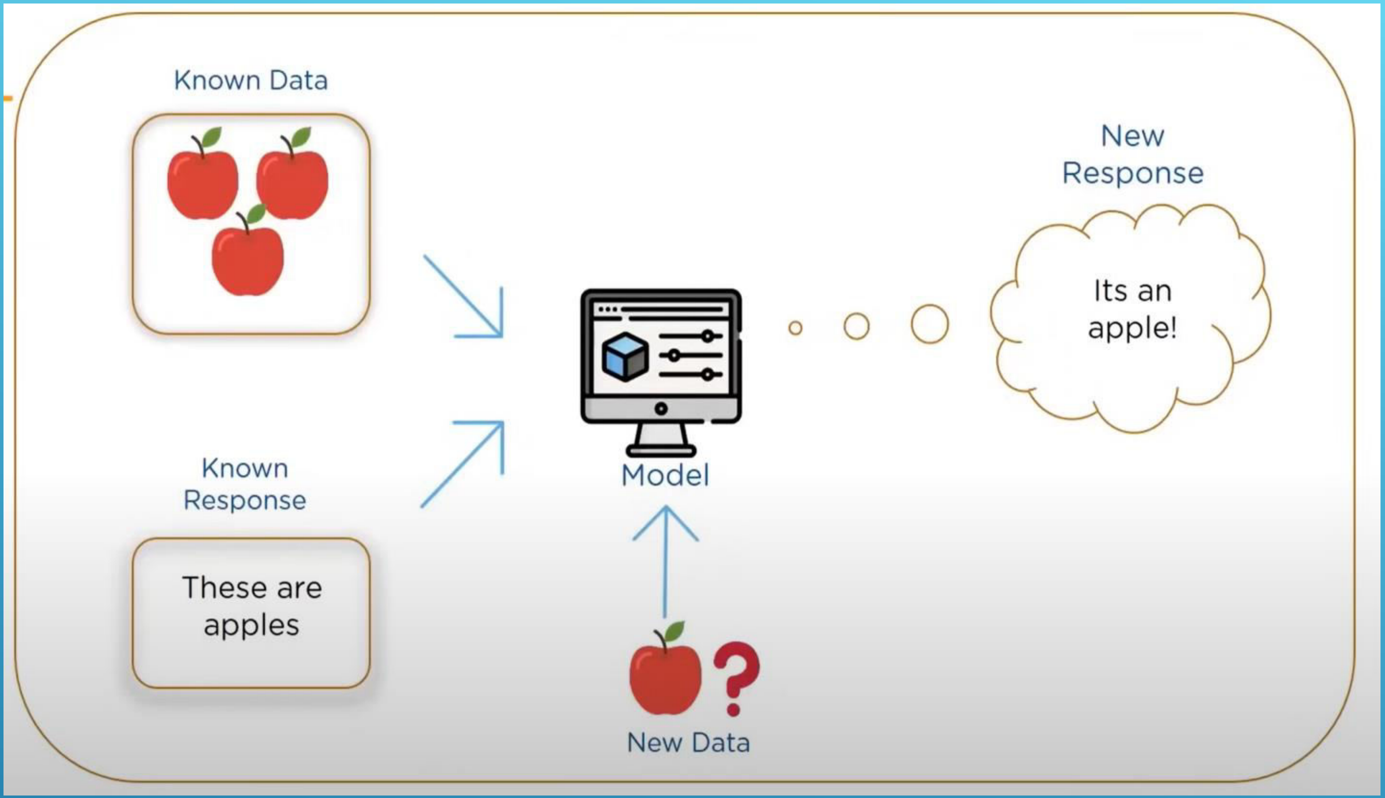


Unsupervised Learning
Non-labeled training data

Supervised Learning
The machine learns from the training data that is labeled

Reinforcement Learning
The machine learns on its own

TYPES OF MACHINE LEARNING TECHNIQUES



1. SUPERVISED LEARNING

1. SUPERVISED LEARNING

Definition:

The model is trained using labelled data meaning both input (X) and output (Y) are known.

Goal:

Learn a mapping from inputs to outputs so it can predict new results accurately

Examples:

Predicting house prices (Regression)

Email spam detection (Classification)

Predicting student marks based on study hours

1. SUPERVISED LEARNING

Definition:

The model is trained using labelled data meaning both input (X) and output (Y) are known.

Goal:

Learn a mapping from inputs to outputs so it can predict new results accurately

Examples:

Predicting house prices (Regression)

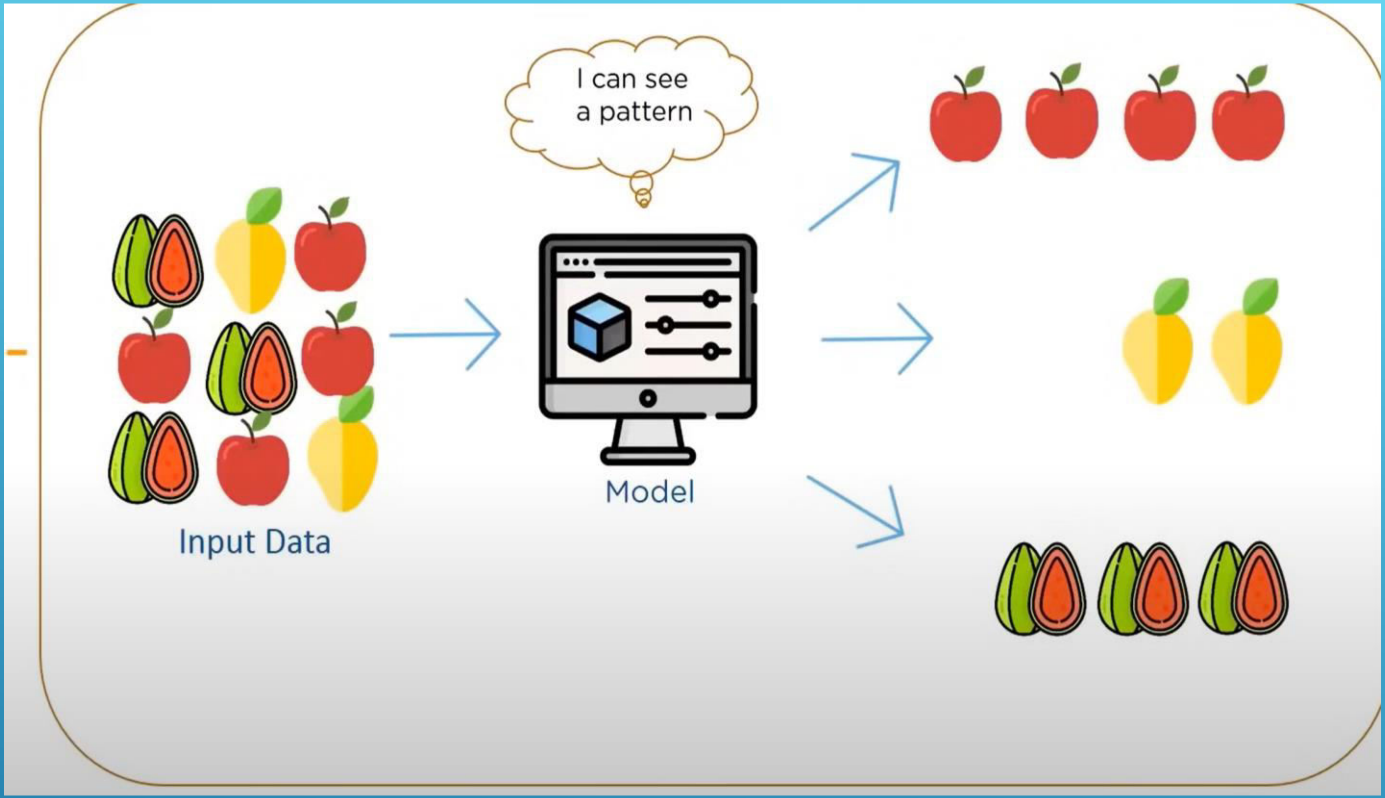
Email spam detection (Classification)

Predicting student marks based on study hours

Algorithms:

- Linear Regression
- Logistic Regression
- Decision Trees
- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)

1. SUPERVISED LEARNING



2. UNSUPERVISED LEARNING

2. UNSUPERVISED LEARNING

Definition:

The model is trained using **unlabeled data** — only inputs (X) are provided, and the system finds hidden patterns.

Goal: Discover structure, grouping, or relationships in the data.

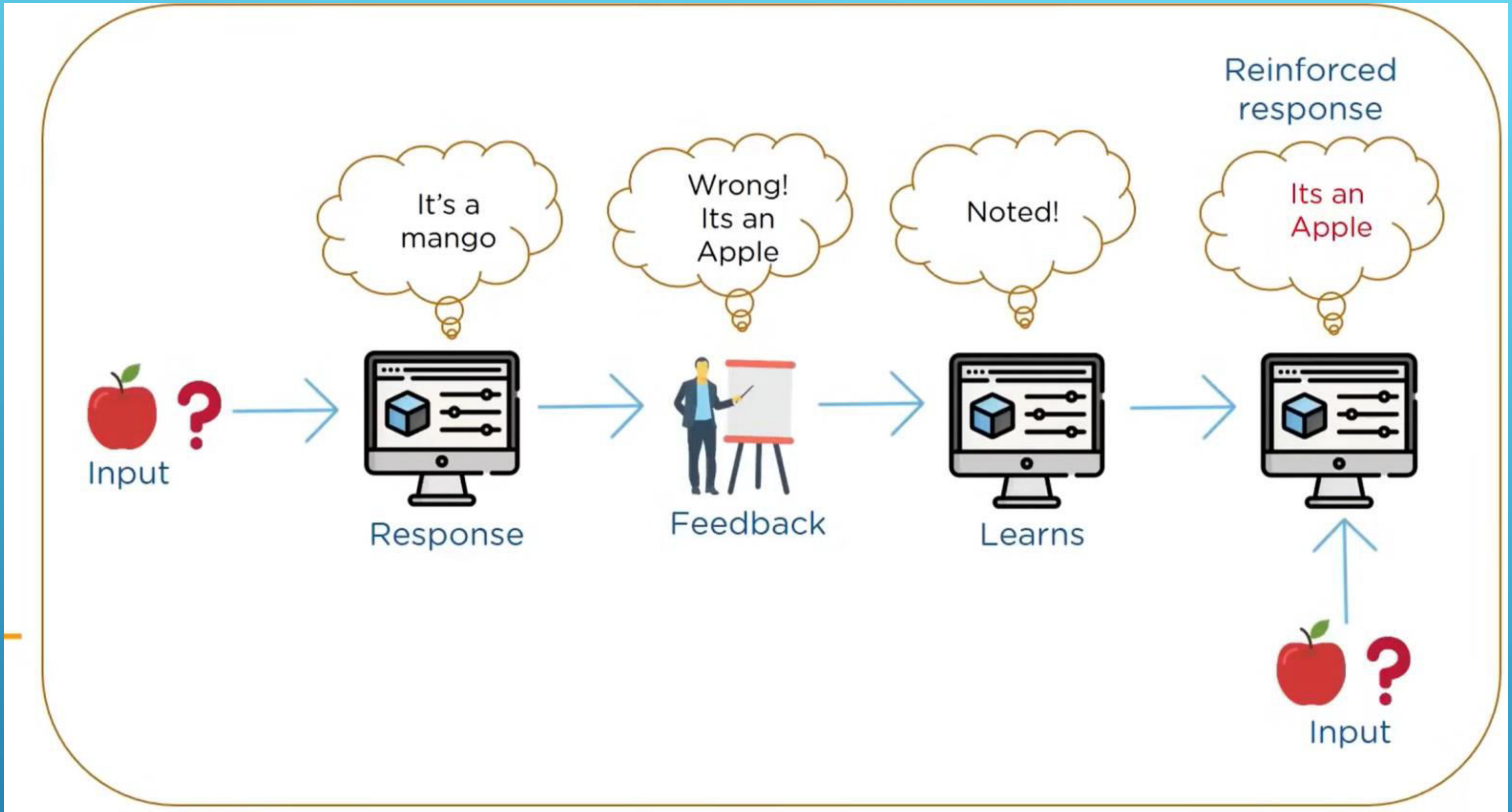
Examples:

- Customer segmentation
- Market basket analysis
- Grouping similar news articles

2. UNSUPERVISED LEARNING

Algorithms:

- K-Means Clustering
- Hierarchical Clustering
- PCA (Principal Component Analysis)
- Association Rules (Apriori Algorithm)



3. REINFORCEMENT LEARNING

Definition:

The model learns by **interacting with an environment**. It receives **rewards or penalties** for actions and improves over time.

Goal: Maximize cumulative reward by learning the best strategy or policy.

Examples:

- Game-playing AI (Chess, Go, etc.)
- Self-driving cars
- Robot navigation

3. REINFORCEMENT LEARNING

- Reinforcement Learning (RL) is a type of Machine Learning where an agent learns to make decisions by interacting with an environment
- The agent performs **actions**
- Receives **rewards or penalties** based on the action
- Learns A **policy** — the best strategy to maximize the total reward over time

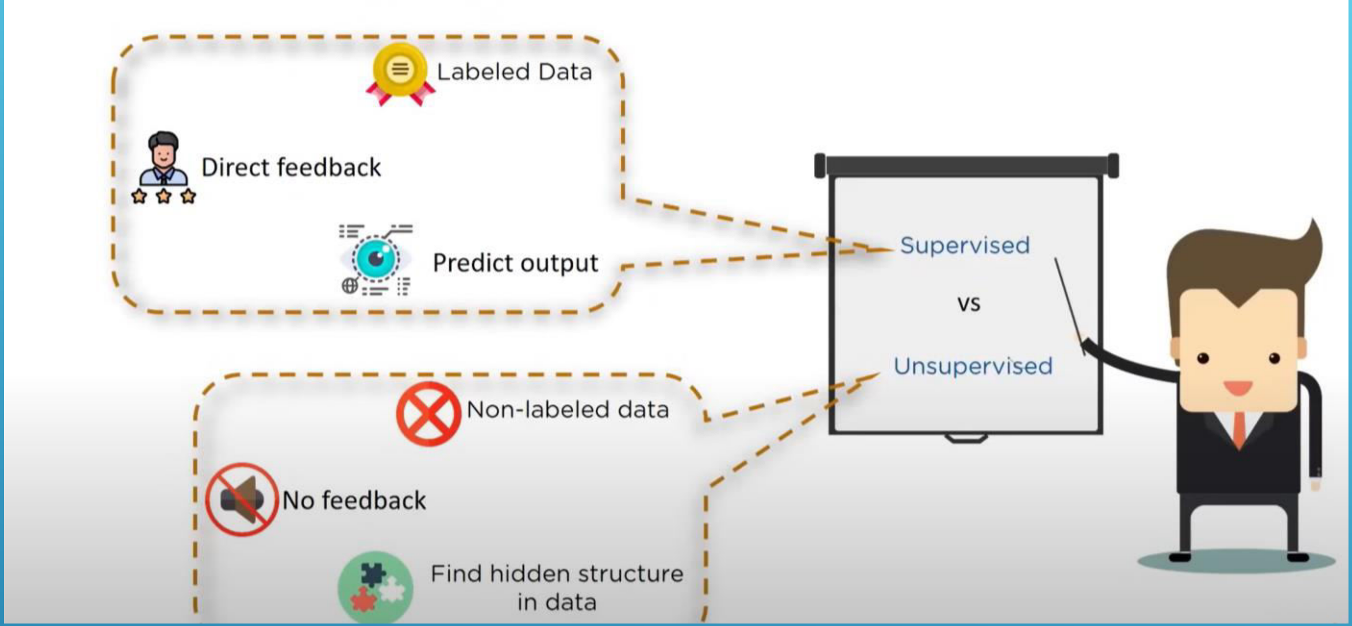
3. REINFORCEMENT LEARNING (RL)

Algorithms:

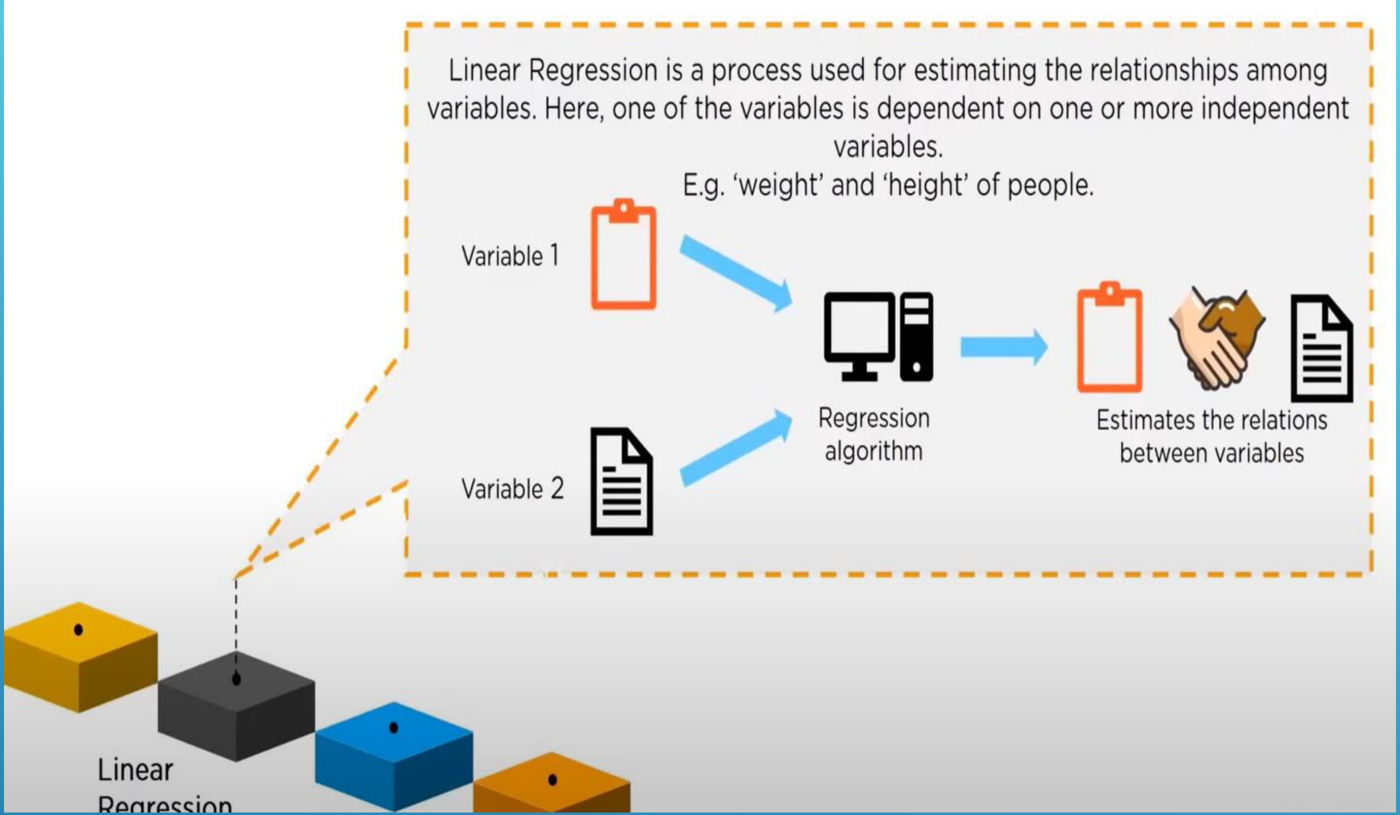
Q-Learning, SARSA, DQN

REINFORCE, PPO, TRPO

3. REINFORCEMENT LEARNING (RL)



SUPERVISED VS UNSUPERVISED LEARNING



Linear Regression is a process used for estimating the relationships among variables. Here, one of the variables is dependent on one or more independent variables.

E.g. 'weight' and 'height' of people.

SUPERVISED LEARNING

- .

THANK YOU!