Lecture 2: Machine Learning

Er.Roshan Shrestha

roshanshrestha@islingtoncollege.edu.np

Agenda

- Machine Learning – Basic Concepts
- Applications of Machine Learning
- Types of Machine Learning
- Classification and clustering
- Scikit Learn
- Train Test Split
- KNN (K-nearest neighbors) classification
- Model Evaluation Metrics(Confusion matrix, Recall, Precision)

Data is everywhere!

- Data Explosion
- Smartphones, browsing,
- social networking and all
- We are being 'DATAFIED'

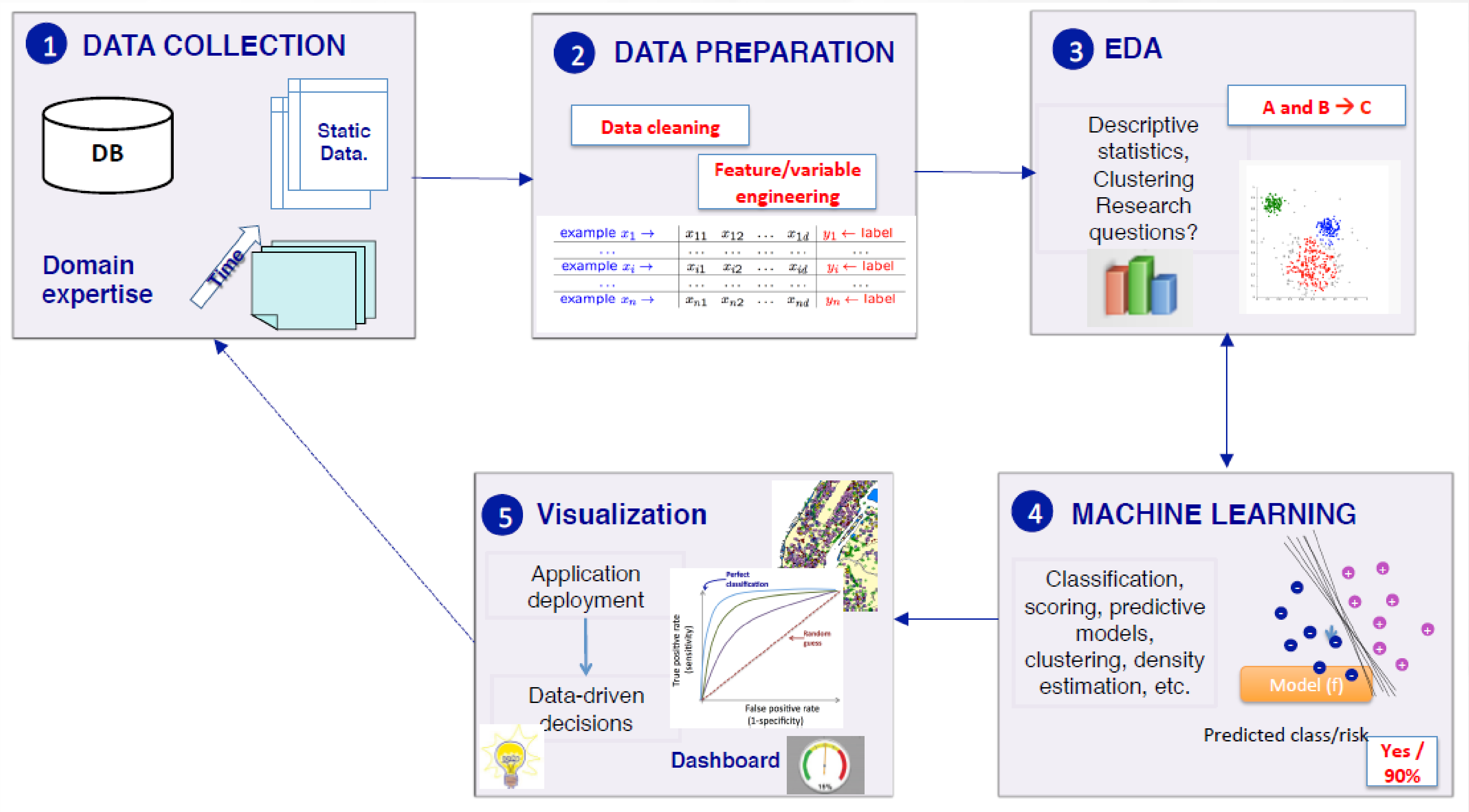
Different kinds of data

Data comes in:

different sizes and also flavors (types):

- *Texts*
- *Numbers*
- *Clickstreams*
- *Graphs*
- *Tables*
- *Images*
- *Transactions*
- *Videos*
- *And many more*

The Data Science Process



Machine Learning - Definition

- Way of making computers automatically learn and improve from the experience without being *explicitly programmed*.

"Machine learning is a branch of AI focused on building computer systems that learn from data. "



Machine Learning: Past, Present and Future

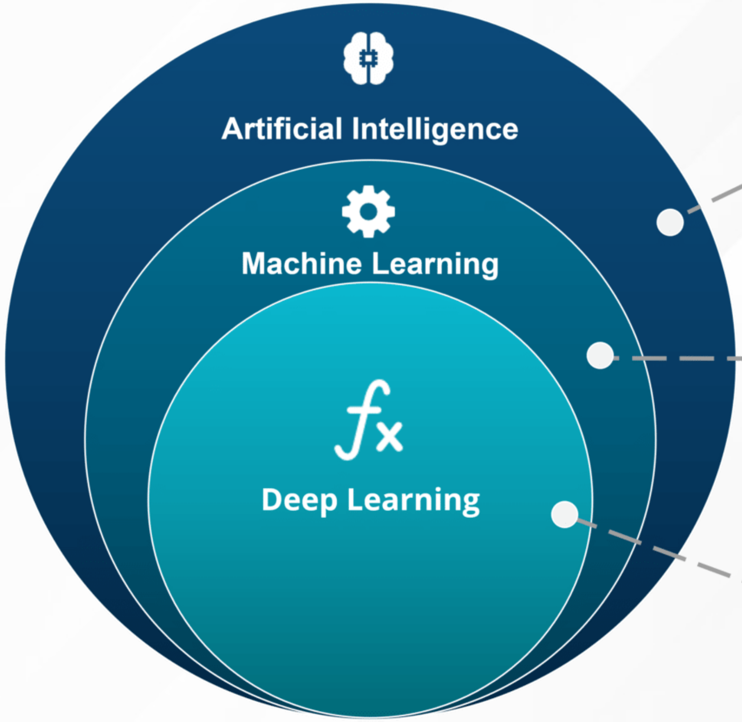
- **Symbolic AI:**
    - The dominant approa
      AI or GOFAI) knowled
- **Shift from rule-based**
- **Support Vector Machi**
- **Reinforcement Learni**
- **More and more data**
- **Deep Learning (2010s**
- **Natural Language Pro**
- **BERT, GPT**
- **AI in Real-World Appl**
- **Ethics and Bias**

Applications of ML

- *Spam filtering*
- *Credit card fraud detection*
- *Digit recognition on cheques, zip codes*
- *Detecting faces in images*
- *MRI image analysis*
- *Recommendation system*
- *Search engines*
- *Handwriting recognition*
- *Scene classification and many more*

informatics college • pokhara
ITAHARI INTERNATIONAL COLLEGE

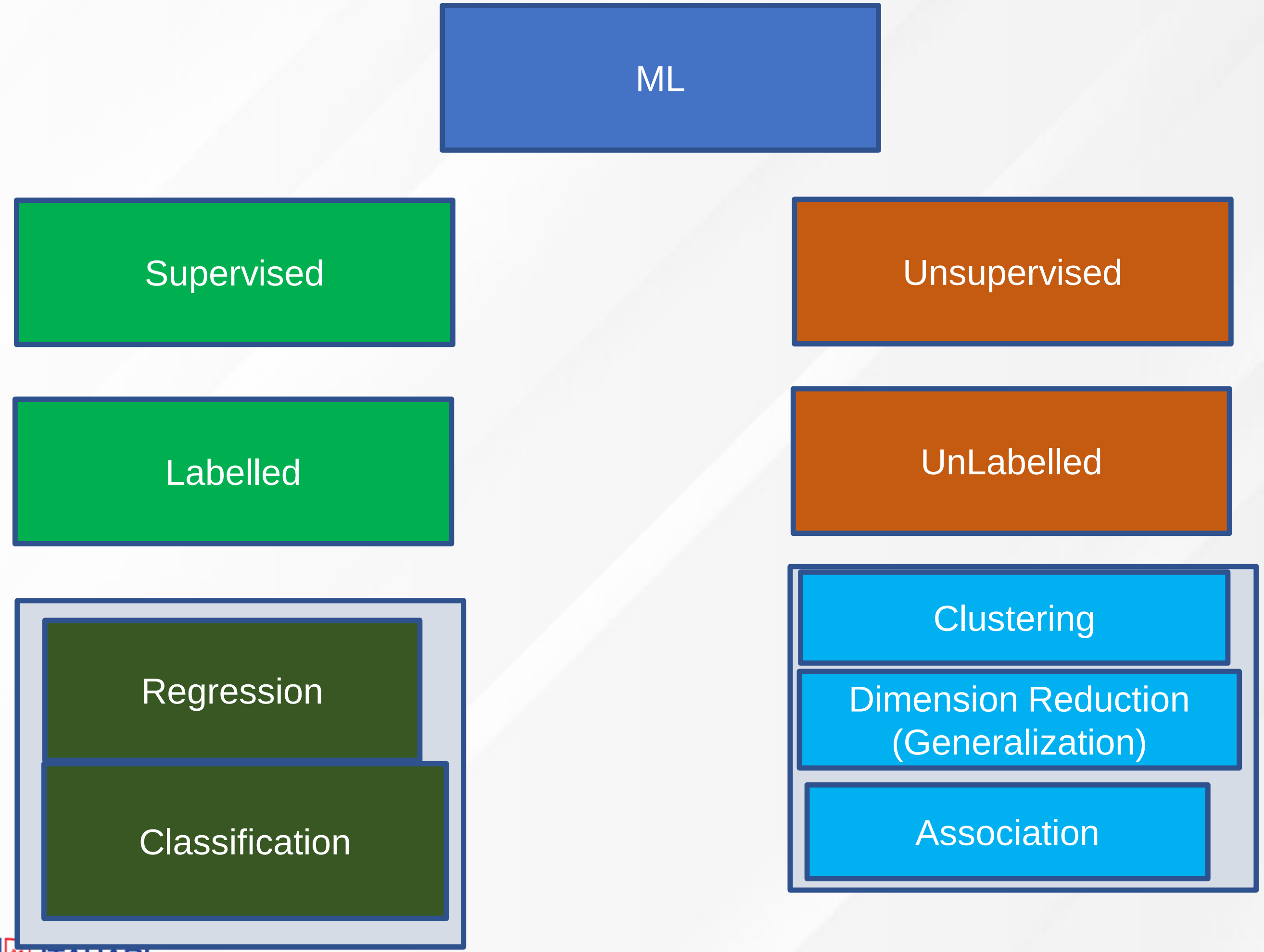
AI, ML and DL Relation



ARTIFICIAL INTELLIGENCE
A technique which enables machines to mimic human behaviour

MACHINE LEARNING
Subset of AI technique which use statistical methods to enable machines to improve with experience

DEEP LEARNING
Subset of ML which make the computation of multi-layer neural network feasible


ML
Supervised
Unsupervised
Labelled
UnLabelled
Regression
Classification
Clustering
Dimension Reduction
(Generalization)
Association

ML

- Supervised
  - Labelled
  - Predict a number
  - Predict a category
- Unsupervised
  - UnLabelled
  - Dividing by similarity
  - Finding hidden dependencies
  - Identifying the sequences

Stock Price Prediction

Weather Forecasting

Regression

House Price Prediction

Medical Expense Prediction

Demand Prediction

Classification

Email Spam Detection

Spam or not?

content, sender information and others

Customer Churn Prediction

whether a customer will leave a service or continue?

Disease Diagnosis

Fraud Detection

Image Recognition

Clustering

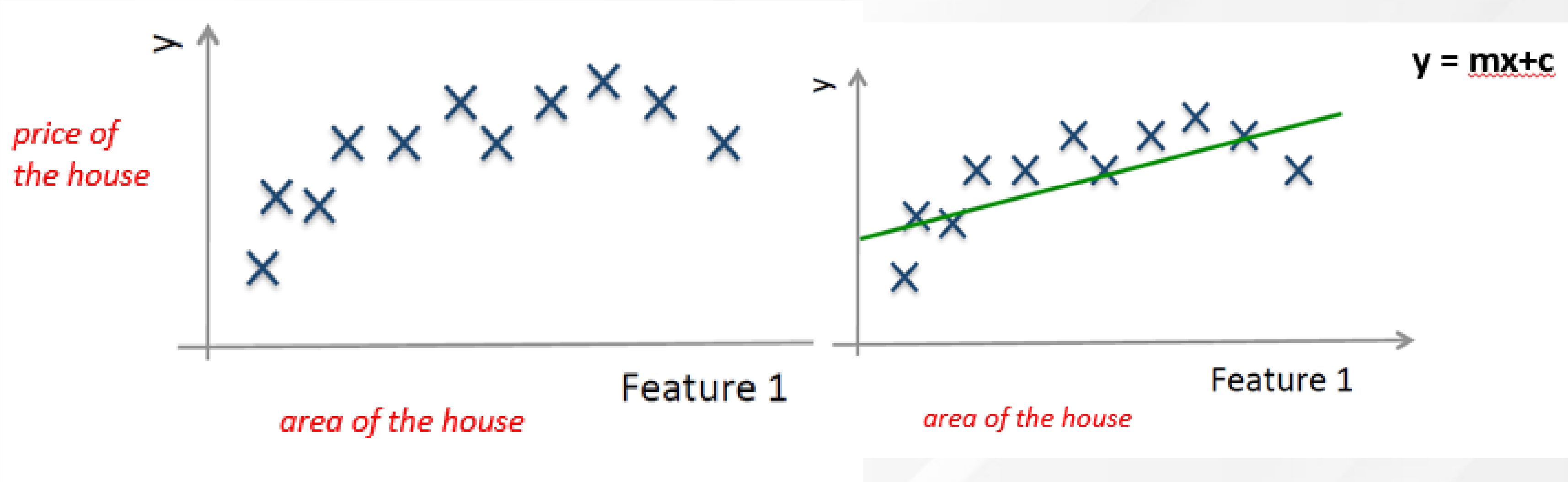
Customer Segmentation	Targeted marketing	purchasing behavior, demographics
Document and News Article Grouping	In which group does that article lies	topics or clusters based on content similarity
Anomaly Detection in Network Security	grouping normal patterns and identifying outliers	potential security breaches)
Social Network Analysis	Grouping people based on their social connections or activity	identify communities or influential groups.

Dimensionality Reduction		
Data Visualization	Reducing high-dimensional data to two or three dimensions	make visualization possible and patterns more observable.
Noise Reduction in Data	Removing irrelevant or redundant features from the dataset	improving model performance
Image Compression	Reducing the number of pixels/features in images while retaining essential information	efficient storage or analysis
Social Network Analysis	Grouping people based on their social connections or activity	identify communities or influential groups.

Association

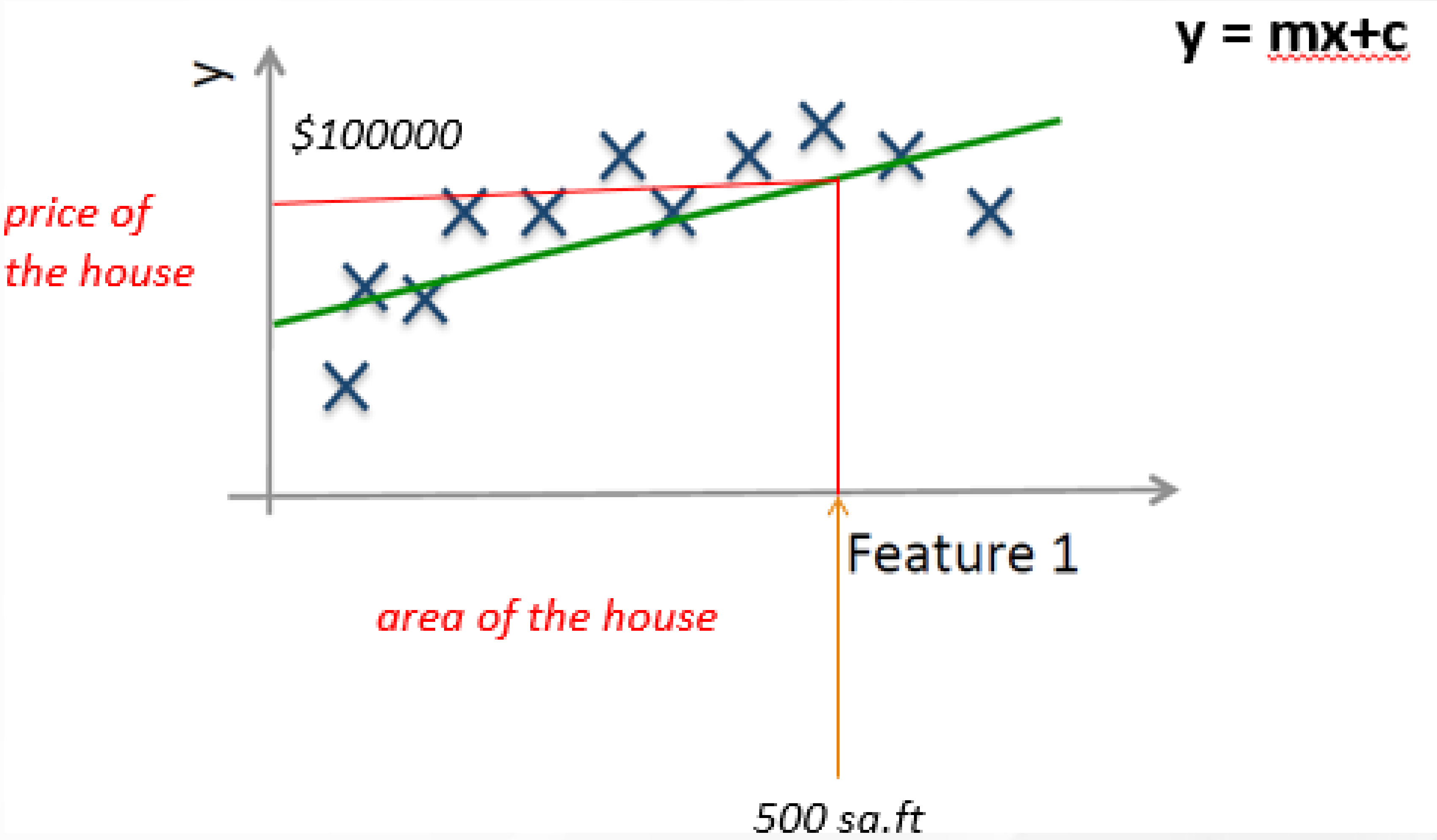
Market Basket Analysis	Analyzing purchase patterns to identify which items are frequently bought together	aiding in product placement and recommendations.
Cross-Selling and Up-Selling	Identifying which products or services can be recommended together	based on customer purchase history.
Product Recommendations	Suggesting products to customers based on items frequently bought together or by customers with similar preferences.	
Fraud Detection	Recognizing patterns of activities or transactions that commonly appear together in fraudulent schemes.	
Ad Click Prediction	Analyzing user behavior to understand which ads are often clicked together, helping with targeted ad placement	

Regression, Classification, Clustering



it requires labeled data to learn.

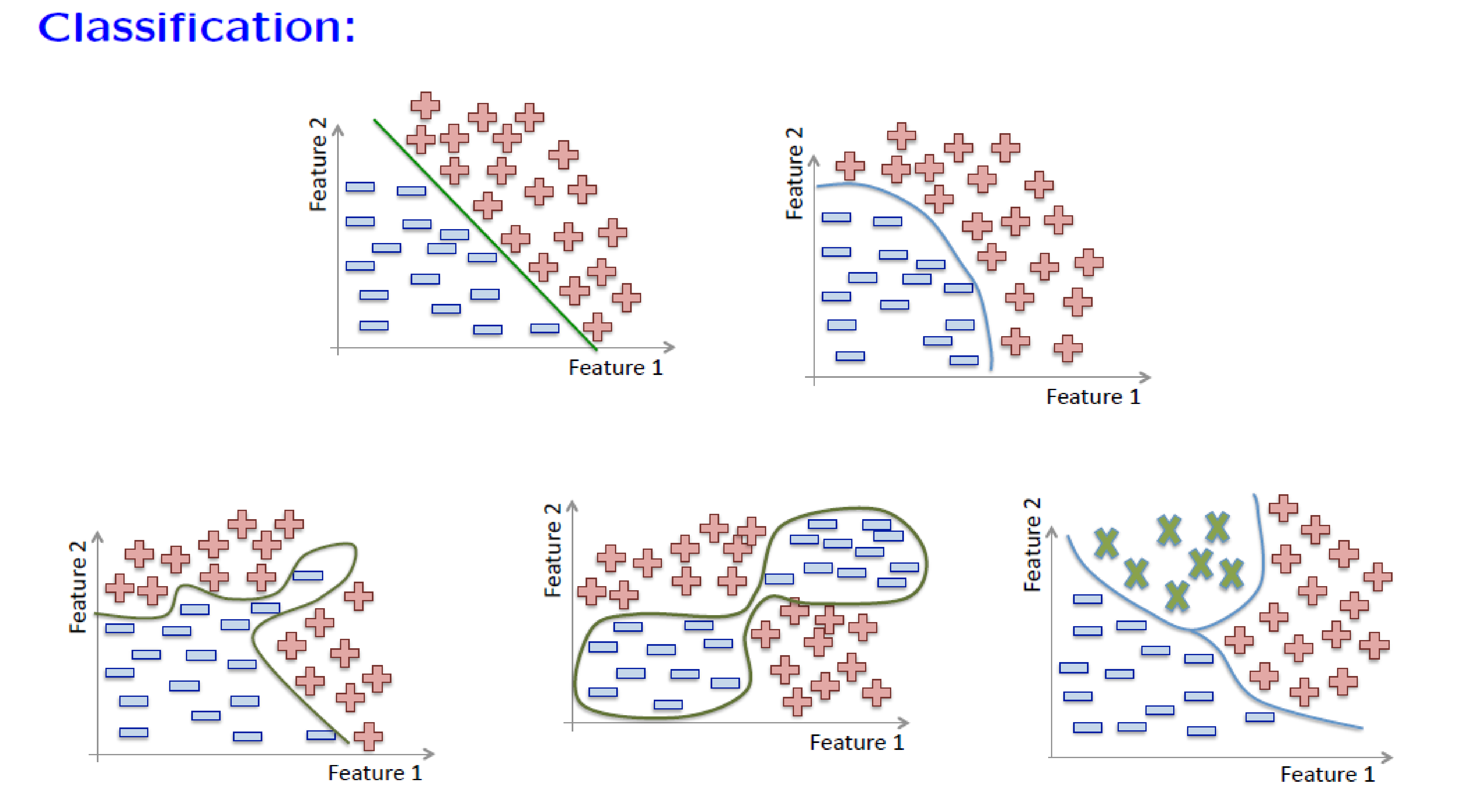
Regression


y = mx+c
y
$100000
price of
the house
Feature 1
area of the house
500 sq.ft

Classification

- **supervised learning** where the goal is to predict a categorical label (class) for a given input based on labeled training data.
- used when there is a predefined set of categories or classes, and the task is to assign each input to one of these classes.
- e.g., spam or not spam
- Predicting if a patient has a disease (Yes/No) based on health records.

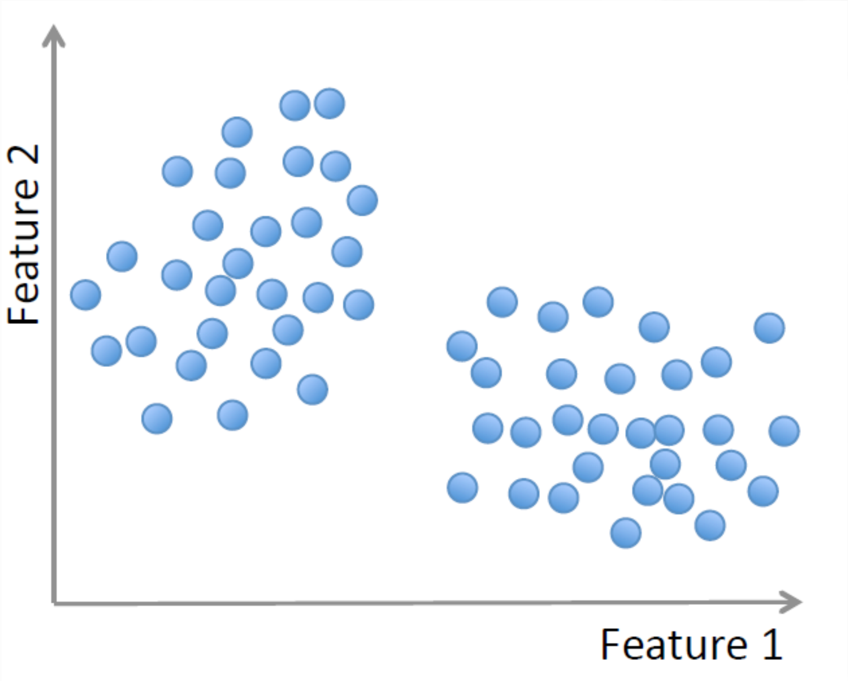
Classification

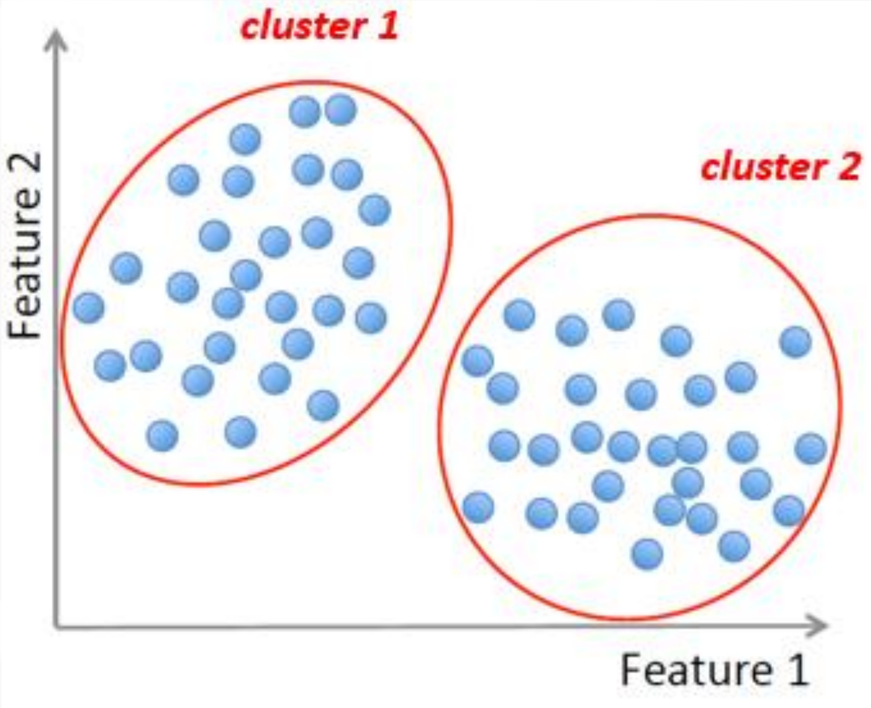


Clustering

- **unsupervised learning** where the goal is to group similar data points into clusters based on the patterns or structure in the data.
- In clustering, there are no predefined labels, and the algorithm tries to organize the data into meaningful groups.
- when we don't have labeled data and want to group similar data points together.
- Clustering is useful for exploratory data analysis when we want to find hidden patterns or structures in the data.
- Grouping users of a website based on browsing behavior to identify different customer segments.

Clustering





Classification plus Clustering

- **Scenario:**
- There is an e-commerce business. Both classification and clustering can be used in different contexts:
- **Classification**:
- to predict whether a customer will **purchase** a product or not based on previous data (e.g., their browsing history, past purchases, etc.).
  - Input: Customer features (age, gender, browsing behavior).
  - Output: Purchase (Yes/No).
- **Clustering**:
- to **group customers** into segments based on their behavior (e.g., high spenders, medium spenders, occasional buyers).
  - Input: Purchase history, number of items bought, spending habits.
  - Output: Grouping of customers into segments (clusters).

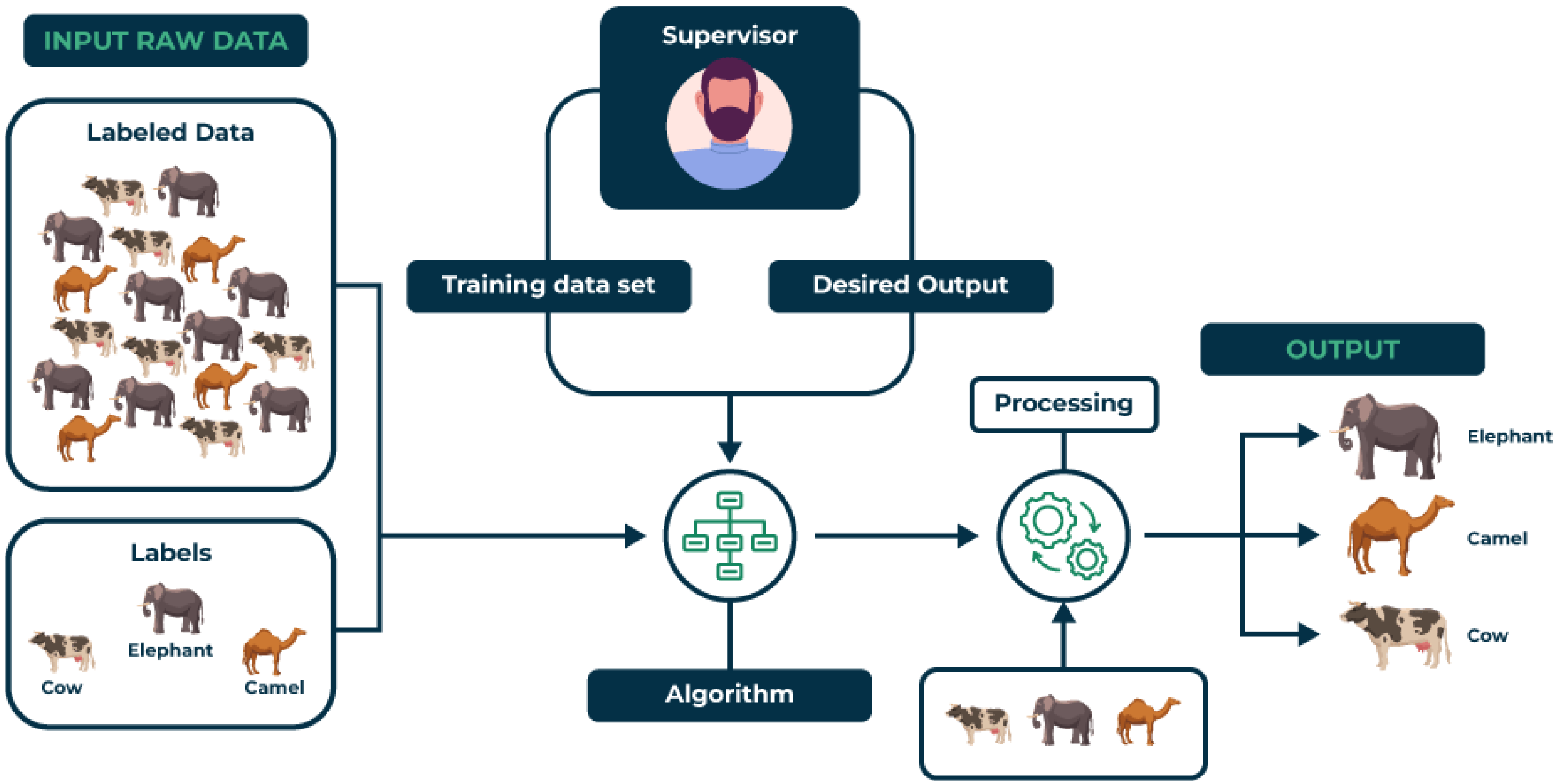
ML Algorithms

- Supervised Learning
- Unsupervised Learning
- Semi-Supervised Learning
- Reinforcement Learning
- Deep Learning
- Ensemble Learning

Supervised

- algorithm learns from labeled training data.
- model is provided with input-output pairs, and the goal is to learn a function that maps inputs to the correct outputs.
- Training data consists of input-output pairs (e.g., features and labels).
- The task is to predict labels or outcomes for new, unseen data.
- There are two main types of tasks: **regression** (predict continuous values) and **classification** (predict categorical labels).

Supervised Learning



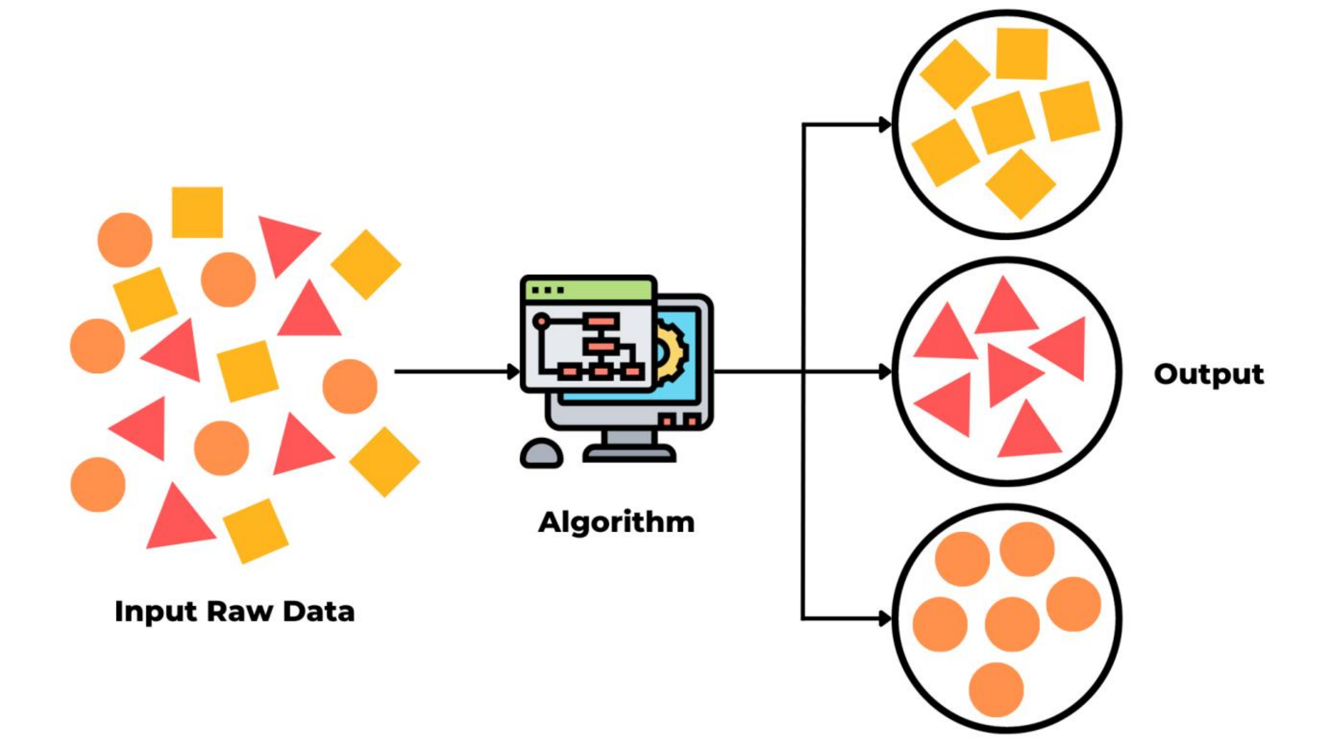
Supervised Algorithms

- Linear Regression (Regression)
  - *Predict House Price*
- Logistic Regression (Classification)
  - *Spam or no Spam*
- Decision Tree (Regression and Classification)
  - *Predicting customer churn*
- Random Forest (Regression and Classification)
  - *Predicting loan default risk*
- SVM (Regression and Classification)
  - *classify image of cats and dogs*
- K Nearest Neighbors (KNN) (Regression and Classification)
  - *Recommendation*

Unsupervised

- the algorithm learns **patterns** from **unlabelled data**.
- The goal is to find hidden structures in the data rather than predicting specific outcomes.
- The training data is not labeled; there are **no explicit input-output pairs**.
- It is used to discover the underlying structure or distribution in data.

islington college
(इस्लिङ्टन कलेज)
informatics college • pokhara
ITAHARI INTERNATIONAL COLLEGE


Input Raw Data
Algorithm
Output

Unsupervised Algorithms

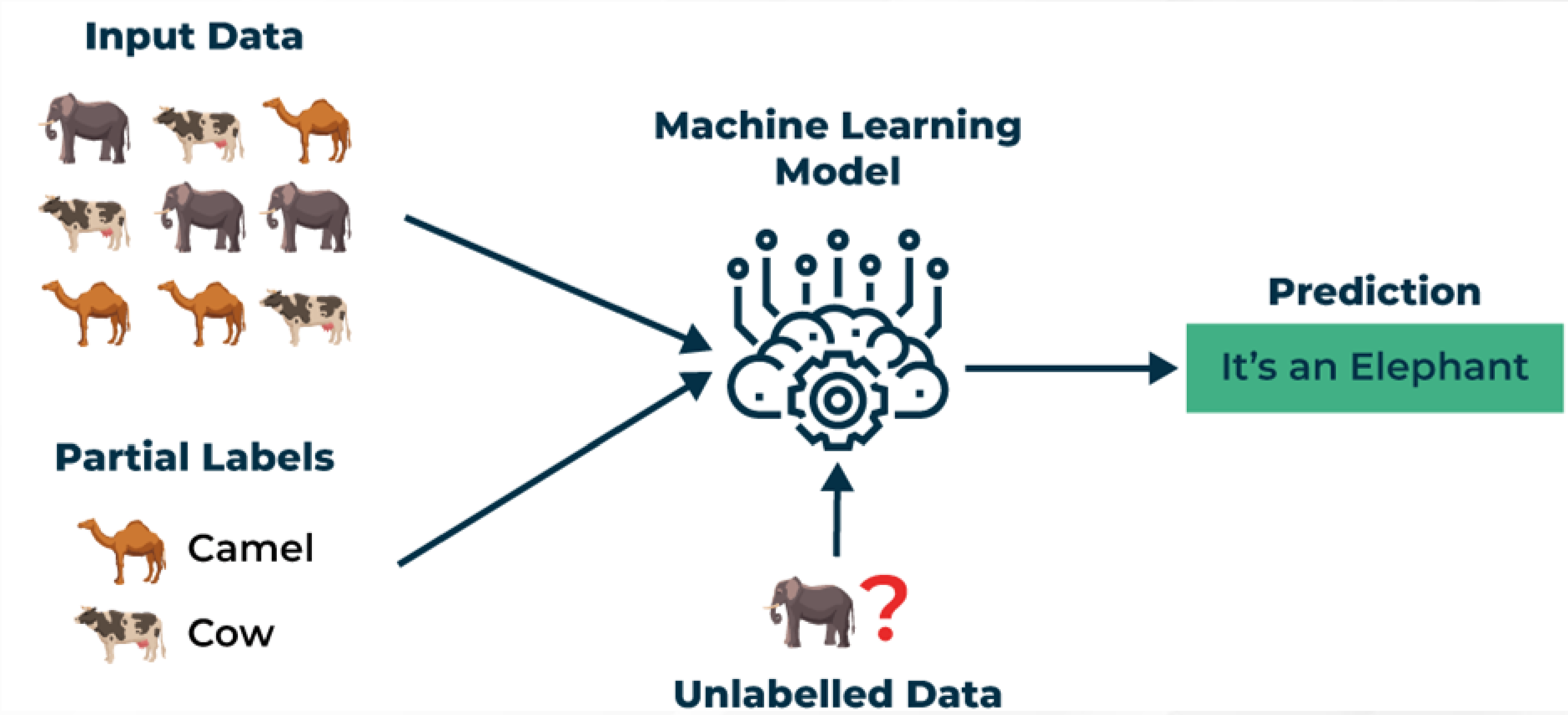
- K means Clustering (Clustering)
  - *Segmenting customers based on purchasing behavior.*
- Hierarchical Clustering (Clustering)
  - *Grouping similar news together*
- Principal Component Analysis (PCA) (Dimensionality Reduction)
  - *Reducing the number of features in a dataset while preserving its variance.*
- Autoencoders (Dimensionality Reduction)
  - *Anomaly detection by learning data representation.*
- Gaussian Mixture Models (GMM) (Clustering)
  - *Identifying Subgroups within data set*

Semi-Supervised Learning

- uses a small amount of labeled data and a large amount of unlabeled data.
- Combines aspects of both supervised and unsupervised learning.
- Useful when labeling data is expensive or time-consuming, but abundant unlabeled data is available.
- ***The algorithm tries to learn from the labeled data and generalizes this learning to the unlabeled data.***
- **Common Algorithms:**
- **Self-training**: Uses labeled data to train a model, then applies the model to the unlabeled data, iteratively adding the most confident predictions to the labeled set.
- **Co-training**: Uses two models that are trained on different features of the data. Each model labels new data for the other.

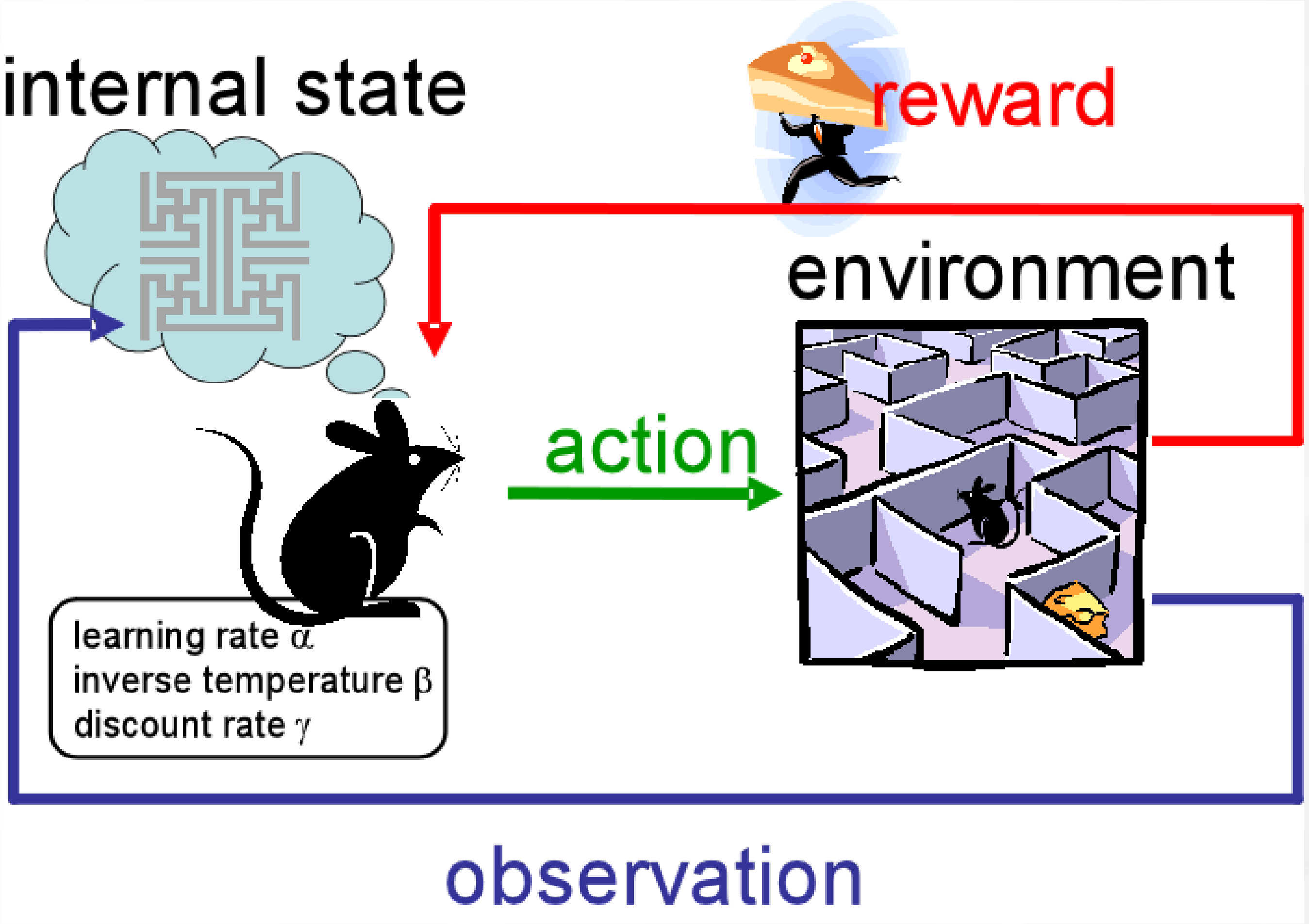
Application

- Semi-supervised learning is often used in image classification tasks, where labeling thousands of images can be labor-intensive.
- A small number of images can be labeled, and the model can extrapolate to learn from the large unlabeled dataset.


Input Data
Machine Learning Model
Prediction
It's an Elephant
Partial Labels
Camel
Cow
Unlabelled Data

Reinforcement Learning

- An agent learns to make decisions by interacting with an environment. The agent receives feedback in the form of ***rewards or penalties*** and **adjusts its actions** to **maximize cumulative rewards**.
- **No labeled data, but feedback** is provided after each action.
- The agent explores and exploits actions to find the optimal strategy or policy.
- RL is focused on **sequential decision-making** and **delayed rewards**.


internal state
reward
environment
action
learning rate α
inverse temperature β
discount rate γ
observation

Reinforcement Algorithms

- Q-Learning (Model-Free RL)
  - *Training a robot to navigate through a maze.*
- Deep Q Networks (DQN) (Deep RL)
  - *Teaching an AI agent to play video games*
- Policy Gradient Methods (Policy-Based RL)
  - *Controlling robotic arms in industrial settings.*
- Proximal Policy Optimization (PPO) (Deep RL)
  - *Learning strategies in multi-agent environments (e.g., game AI).*

Deep Learning

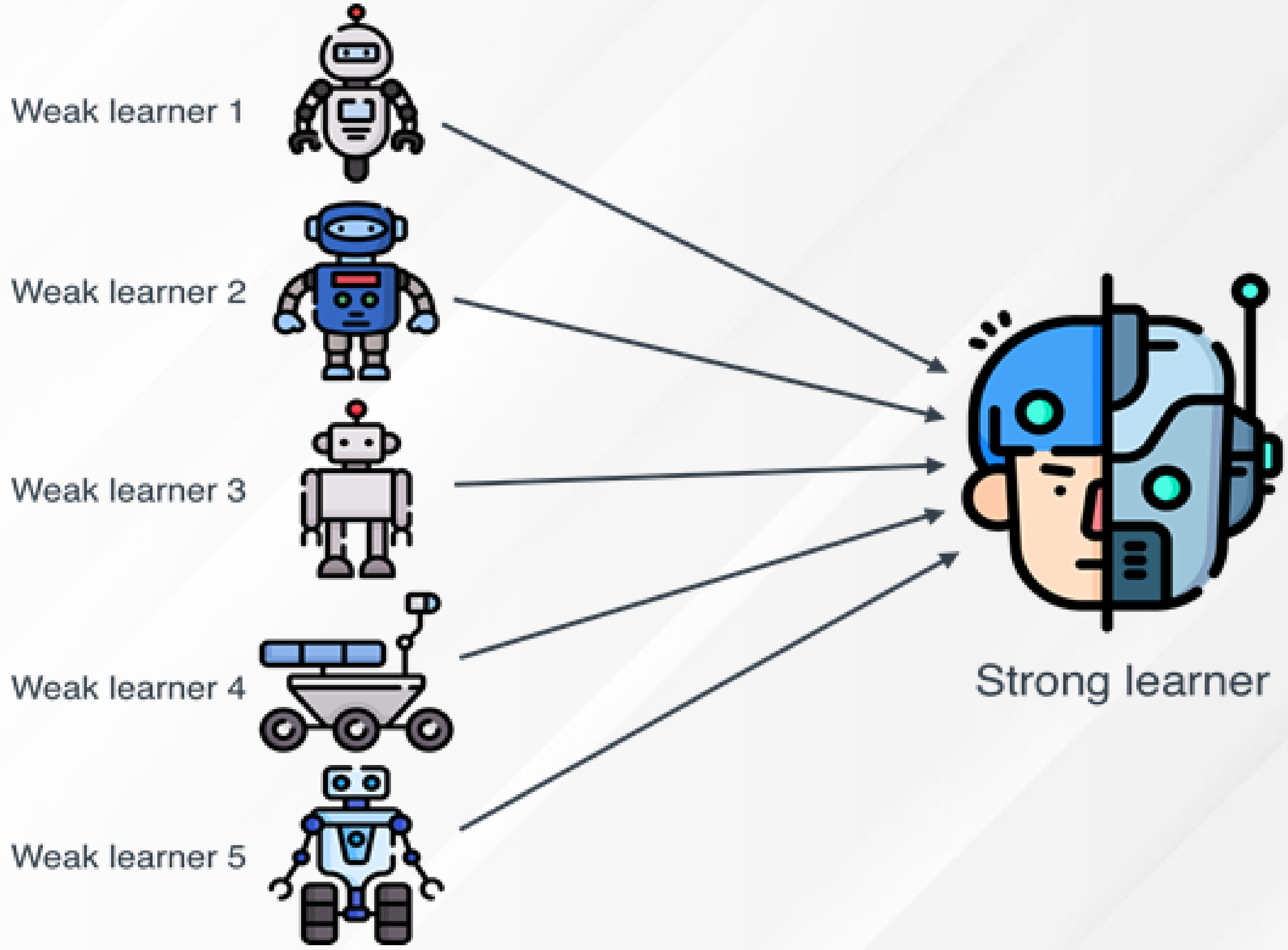
- Learns from large datasets using **multi-layered neural networks**.
- primarily associated with supervised learning but can also be applied to unsupervised and reinforcement learning tasks.
- Focuses on learning data representations through multiple layers of abstraction.
- Requires large amounts of data and computational power.
- Can automatically extract features without requiring manual feature engineering.

Deep Learning Algorithms

- Convolutional Neural Networks (CNNs) (Supervised)
  - Image recognition and classification (e.g., facial recognition).
- Recurrent Neural Networks (RNNs) (Supervised)
  - NLP
- Generative Adversarial Networks (GANs) (Unsupervised)
  - Generating realistic images or videos.
- Deep Q Networks (DQN) (Reinforcement Learning)
  - to play games or control robotic actions.

Ensemble Learning

- combines multiple machine learning models to improve overall performance.
- By aggregating the predictions of multiple models, ensemble methods tend to produce better results than individual models.
- *Combines the outputs of several weaker models* to create a stronger overall model.
- Helps to reduce bias, variance, or improve prediction accuracy.


Weak learner 1
Weak learner 2
Weak learner 3
Weak learner 4
Weak learner 5
Strong learner

Ensemble Learning Algorithms

- Random Forest (Supervised)
  - *predicting loan defaults by combining multiple decision trees.*
- Boosting (e.g., AdaBoost, XGBoost) (Supervised)
  - *Enhancing model accuracy in tasks like credit scoring.*
- Bagging (Supervised)
  - *Reducing overfitting by training multiple models on different subsets of data.*

Scikit-learn

- **open source library** and also commercially usable under BSD license.
- popular Python library for machine learning.
- provides simple and efficient tools for **data mining, data analysis,** and **machine learning**, with support for both **supervised** and **unsupervised learning**.
- built on top of *NumPy, SciPy, and matplotlib*.
- **Timeline:**
- 2007
- David Cournapeau, a French software engineer, started Scikit-learn as part of the Google Summer of Code (GSoC) project in 2007.
- first official release, ***version 0.1*** in 2010.
- ***By 2015, became an industry standard for machine learning***

informatics college • pokhara
ITAHARI INTERNATIONAL COLLEGE
Islington college (इस्लिङ्टन कलेज)

Scikit-learn

- Design Principles of Scikit-learn
- Simplicity and Consistency:
  - .fit() for training the model
  - .predict() for making predictions
  - .score() for evaluating the model.

Scikit-learn

Features:

- **Classification**
  - Identifying which category an object belongs to (e.g., spam detection).
- **Regression**
  - Predicting a continuous-valued attribute (e.g., predicting house prices).
- **Clustering**
  - Grouping similar objects into sets (e.g., customer segmentation).
- **Dimensionality Reduction**
  - Reducing the number of random variables under consideration (e.g. PCA).
- **Model Selection**
  - Comparing, validating, and selecting models (e.g. cross-validation).
- **Preprocessing**
  - Feature extraction, normalization, and other data transformations.

Scikit-learn

Installation:

- pip install -U scikit-learn
- conda install scikit-learn
- use Python distributions like ***Anaconda***

Advantages:

- Comprehensive and easy to use
- Integration with Other Libraries
- Strong community

Disadvantages

Not Designed for Deep Learning:

- excels in traditional ML algorithms, not suitable for deep learning, where frameworks like TensorFlow, PyTorch, or Keras are preferred.

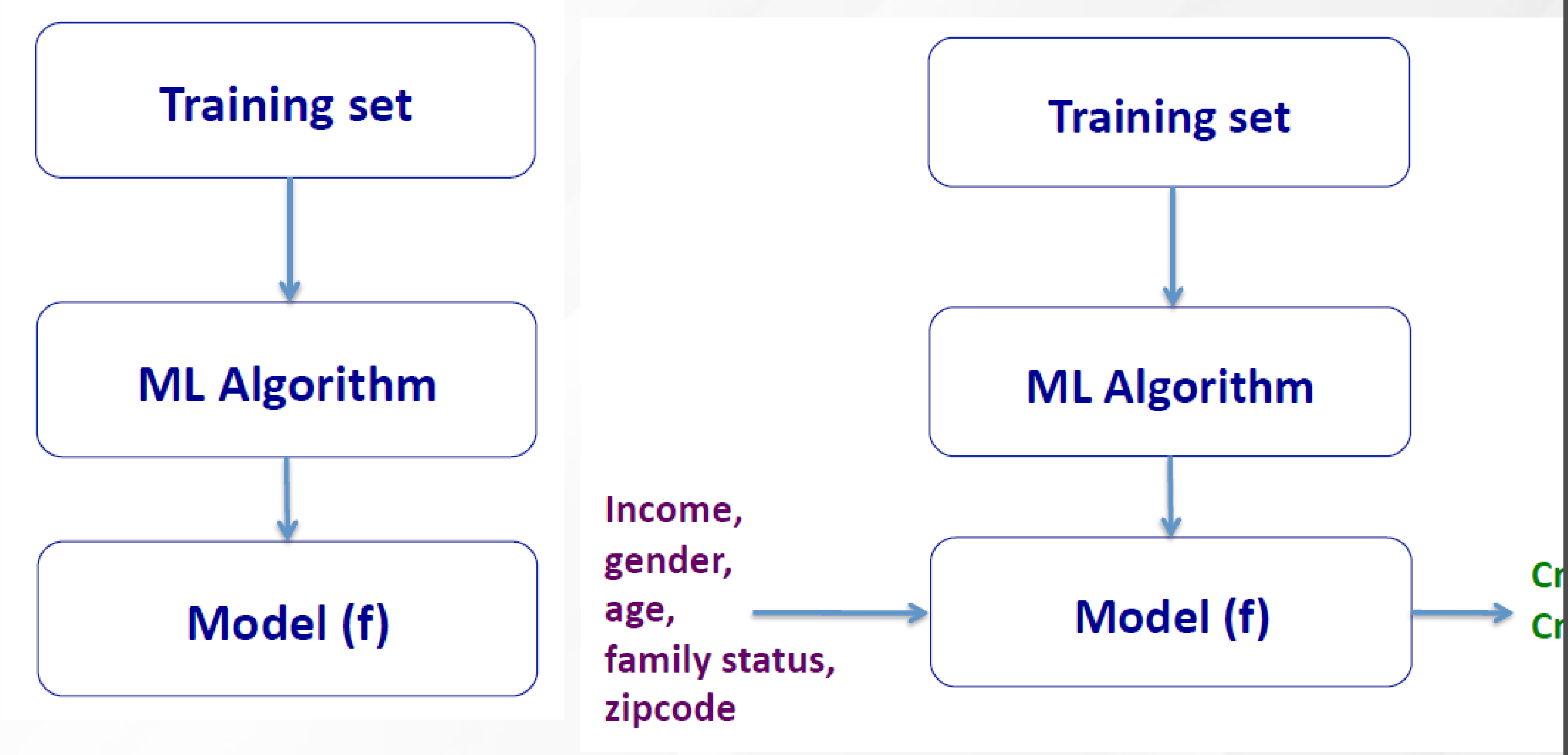
Memory Limitations:

- Scikit-learn works well with datasets that can fit into memory, but for large-scale datasets, distributed frameworks like Dask, Spark, or TensorFlow are better suited.

Train Test split

- helps evaluate how well a machine learning model generalizes to unseen data.
- **Training Set**: This is usually the larger portion of the data (e.g., 70-80%) used for training the model.
- **Test Set:** The remaining data (e.g., 20-30%) is held back and used to evaluate the model's performance.
- **Model Training**: The training set is used to teach the model by allowing it to learn patterns from the data.
- **Model Evaluation**: The test set is used to evaluate how well the model performs on new, unseen data. This helps to estimate the model's performance in the real world.
- **Prevent Overfitting**: If the model is tested on the same data it was trained on, it might simply memorize the data rather than generalizing from it, which leads to overfitting.

ML workflow



model predicts whether to given credit or not/how much amount of credit to give

K-nearest neighbors (KNN)

- K-Nearest Neighbors (KNN) is a simple, instance-based machine learning algorithm used for **classification** and **regression** tasks.
- *Not every ML method builds a model!*
- Main idea: Uses the similarity between examples
- Assumption: Two similar examples should have same labels
- Assumes all examples are points in a $\boldsymbol{d}$ dimensional space $\boldsymbol{R^d}$

K-nearest neighbors (KNN)

- **Euclidean Distance**
- **Manhattan Distance**
- **Minkowski Distance**
- Euclidean distance works well for continuous, real-valued data where features are on the same scale.
- Manhattan distance is more robust for high-dimensional spaces or data with many outliers.
- Minkowski Distance is a metric used to measure the similarity or dissimilarity between two time series by computing the sum of the absolute differences raised to a certain power.
- Depending on the chosen value of 'p', it can represent different distance measures such as the Manhattan distance (p=1), the Euclidean distance (p=2), or the infinite norm (p=∞).

K-nearest neighbors (KNN)

- KNN uses the standard *Euclidean distance* to define nearest neighbors (similar examples)
- Given two examples $\boldsymbol{p}$ and $\boldsymbol{q}$; $\boldsymbol{p}, \boldsymbol{q} \boldsymbol{\epsilon}\, \boldsymbol{R}^2$:

$$\boldsymbol{p} = (p_1, p_2); \boldsymbol{q} = (q_1, q_2)$$

$$\boldsymbol{d}(\boldsymbol{p}, \boldsymbol{q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$$

islington college (इस्लिङ्टन कलेज)
informatics college • pokhara
ITAHARI INTERNATIONAL COLLEGE

K-nearest neighbors (KNN)

- KNN uses the standard *Euclidean distance* to define nearest neighbors (similar examples)
- Given two examples $\boldsymbol{p}$ and $\boldsymbol{q}$; $\quad \boldsymbol{p}, \boldsymbol{q} \epsilon\, \boldsymbol{R}^{\boldsymbol{d}}$:

$$\boldsymbol{p} = (p_1, \ldots, p_d); \boldsymbol{q} = (q_1, \ldots, q_d)$$

$$\boldsymbol{d}(\boldsymbol{p}, \boldsymbol{q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ \ldots + (p_d - q_d)^2}$$

islington college (इस्लिङ्टन कलेज)
informatics college • pokhara
ITAHARI INTERNATIONAL COLLEGE

K-nearest neighbors (KNN)

Training algorithm:

Add each training example $(\boldsymbol{x}, \boldsymbol{y})$ to the dataset $\boldsymbol{D}$,
$\boldsymbol{x} \in \boldsymbol{R}^{d}, \boldsymbol{y} \in \{+\mathbf{1}, -\mathbf{1}\}$

Classification algorithm:

Given a new example $\boldsymbol{x}_q$ (***not present in the dataset***) to be classified,

1. compute the distances between $\boldsymbol{x}_q$ and all other points in dataset $\boldsymbol{D}$
2. find out the ***k points (neighbors)*** that are the closest to $\boldsymbol{x}_q$
3. assign $\boldsymbol{x}_q$ the label that the ***majority of its neighbors have***

k is a parameter that we will need to set

informatics college · pokhara

ITAHARI INTERNATIONAL COLLEGE

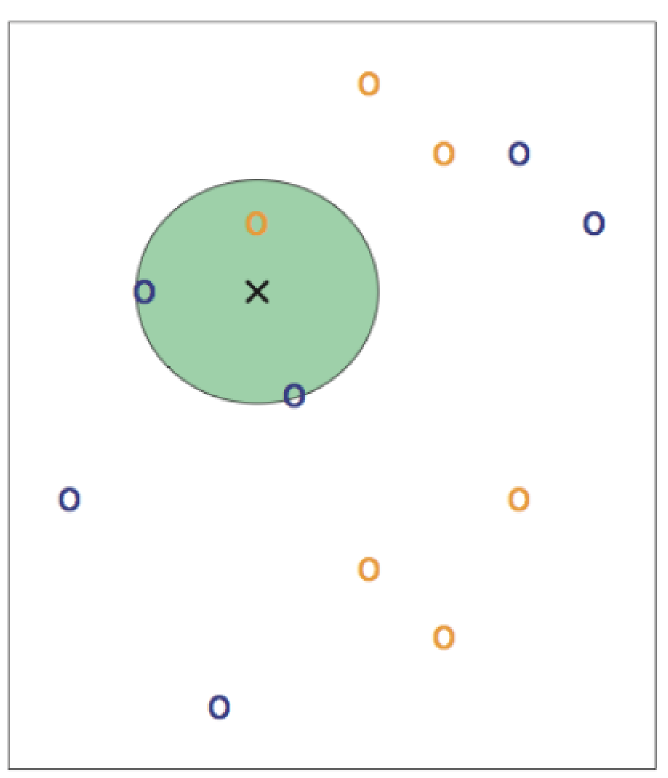
Best value of K

- A small k may lead to overfitting, while a large k can cause underfitting.
- **Cross Validation**: try a range of k values, and for each, compute the model's performance on different folds of the data. The goal is to find the value of k that gives the lowest error or highest accuracy on the validation set.
- **Elbow Method**: You can plot the error rate (or accuracy) against different values of k. As you increase k, the error rate typically decreases initially but then flattens out or starts to increase slightly. The "elbow" point in this curve is often considered the optimal value of k, where the error rate is low, but increasing k further doesn't provide significant gains.

K-nearest neighbors (KNN)

example 3-NN:

- find out the 3 nearest neighbors of a point x
- assign it the label that the majority of its neighbors have
- the point x will be classified as a blue point here



KNN: example

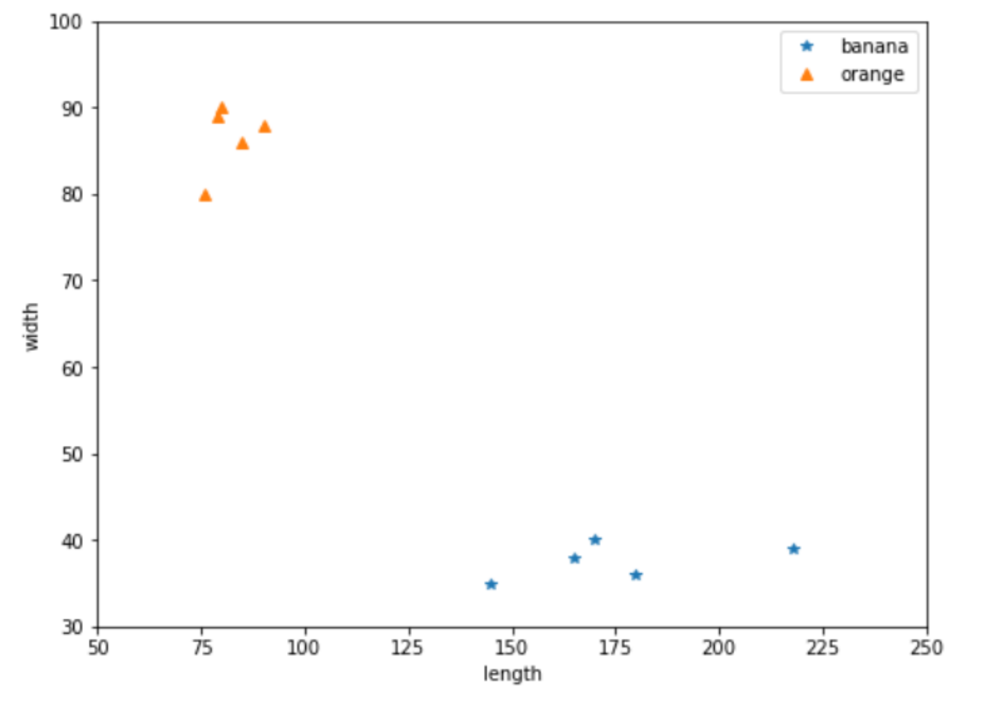
Fruit classification example

Dataset:

Fruit	Length	Width	Label
Fruit 1	165	38	Banana
Fruit 2	218	39	Banana
Fruit 3	76	80	Orange
Fruit 4	145	35	Banana
Fruit 5	90	88	Orange
Fruit 6	170	40	Banana
Fruit 7	180	36	Banana
Fruit 8	80	90	Orange
Fruit 9	85	86	Orange
Fruit 10	79	89	Orange

KNN: example

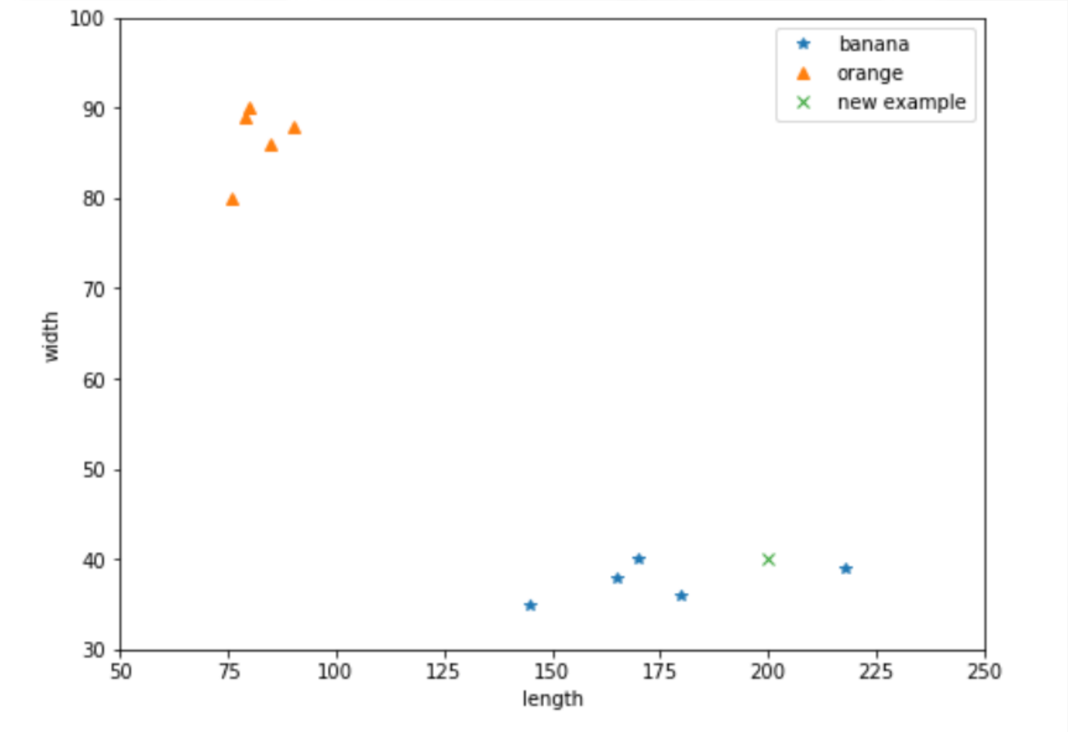
Fruit	Length	Width	Label
Fruit 1	165	38	Banana
Fruit 2	218	39	Banana
Fruit 3	76	80	Orange
Fruit 4	145	35	Banana
Fruit 5	90	88	Orange
Fruit 6	170	40	Banana
Fruit 7	180	36	Banana
Fruit 8	80	90	Orange
Fruit 9	85	86	Orange
Fruit 10	79	89	Orange



KNN: example

- we have a new example to classify as banana or orange

Fruit	Length	Width	Label
New Fruit	200	40	?



KNN example

- Find the distances between the *new example* and *all other examples* in the dataset

Fruit	Length	Width	Label	Distances
Fruit 1	165	38	Banana	35.05709629
Fruit 2	218	39	Banana	18.02775638
Fruit 3	76	80	Orange	130.2919798
Fruit 4	145	35	Banana	55.22680509
Fruit 5	90	88	Orange	120.0166655
Fruit 6	170	40	Banana	30
Fruit 7	180	36	Banana	20.39607805
Fruit 8	80	90	Orange	130
Fruit 9	85	86	Orange	123.8587906
Fruit 10	79	89	Orange	130.5450114
New Fruit	200	40	?	

KNN example

- In the case of ***K* = 3**, then find out *3 neighbors (points)* which are closest to the *new example (point)*

Fruit	Length	Width	Label	Distances
Fruit 1	165	38	Banana	35.05709629
Fruit 2	218	39	Banana	18.02775638
Fruit 3	76	80	Orange	130.2919798
Fruit 4	145	35	Banana	55.22680509
Fruit 5	90	88	Orange	120.0166655
Fruit 6	170	40	Banana	30
Fruit 7	180	36	Banana	20.39607805
Fruit 8	80	90	Orange	130
Fruit 9	85	86	Orange	123.8587906
Fruit 10	79	89	Orange	130.5450114
New Fruit	200	40	?	

KNN example

- assign the *new point* the *label* that the ***majority of its neighbors have***

Fruit	Length	Width	Label	Distances
Fruit 1	165	38	Banana	35.05709629
Fruit 2	218	39	Banana	18.02775638
Fruit 3	76	80	Orange	130.2919798
Fruit 4	145	35	Banana	55.22680509
Fruit 5	90	88	Orange	120.0166655
Fruit 6	170	40	Banana	30
Fruit 7	180	36	Banana	20.39607805
Fruit 8	80	90	Orange	130
Fruit 9	85	86	Orange	123.8587906
Fruit 10	79	89	Orange	130.5450114
New Fruit	200	40	Banana	

Evaluation Metrics

- evaluating the model using the **confusion matrix** and other related metrics like **precision, recall, and F1-score**
- A **confusion matrix** summarizes the number of true positives, true negatives, false positives, and false negatives, which helps understand where the model is making correct or incorrect predictions.
- **Precision**: The proportion of true positive predictions out of all positive predictions made by the model.
- **Recall**: The proportion of true positive predictions out of all actual positive instances.
- **F1-Score**: The harmonic mean of precision and recall, which balances both in a single metric.

K-nearest neighbors (KNN)

Pros:

- simple to implement
- works well in practice
- does not require to build a model

Cons:

- requires large space to store the entire training dataset
- SLOW !!!

End of Lecture 2

More in labs…