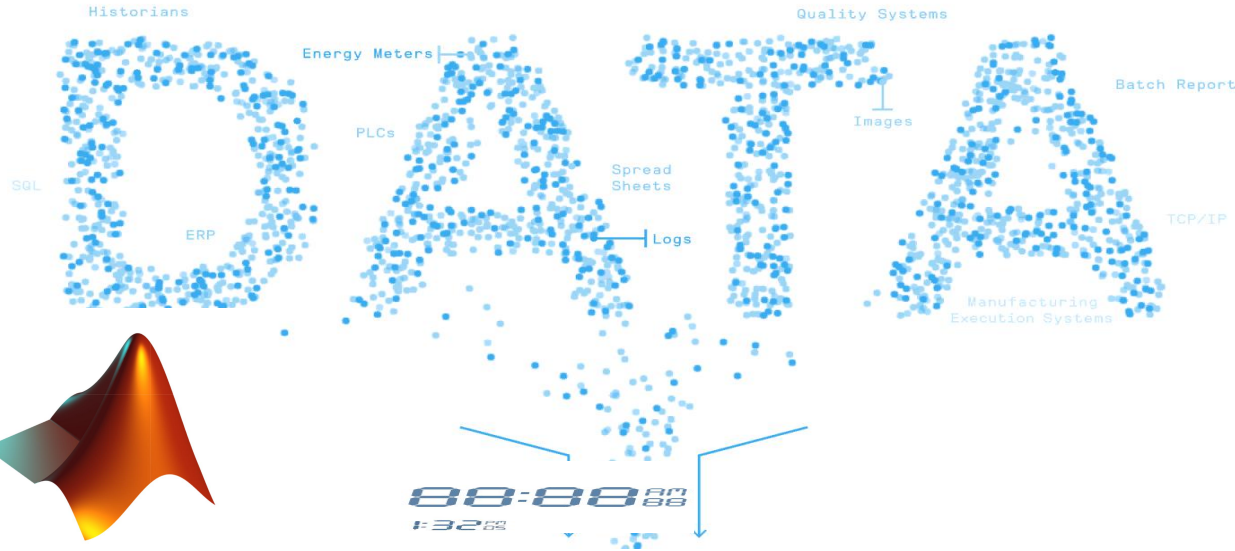
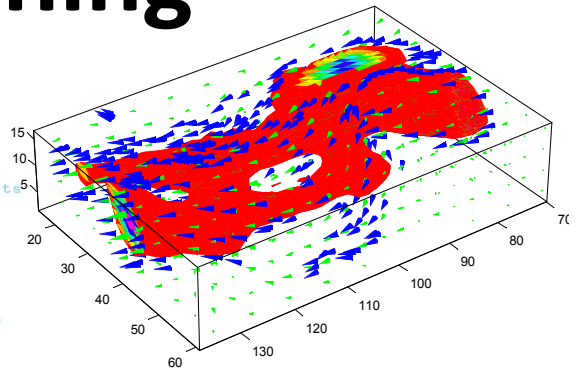
Introduction to Artificial Intelligence
- 05-01 Machine Learning

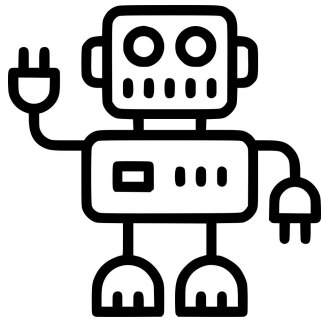




Dr Leo Chen
leo.chen@ieee.org
06/Sep/2021

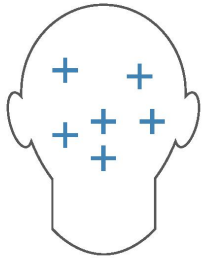
Module Contents

1. Introduction
2. Evolutionary Computation
3. Artificial Neural Network
4. Fuzzy Logic and Fuzzy Systems
5. More AI Subsets
6. AI and Industry 4.0
7. AI Applications
8. Labs
9. Courseworks



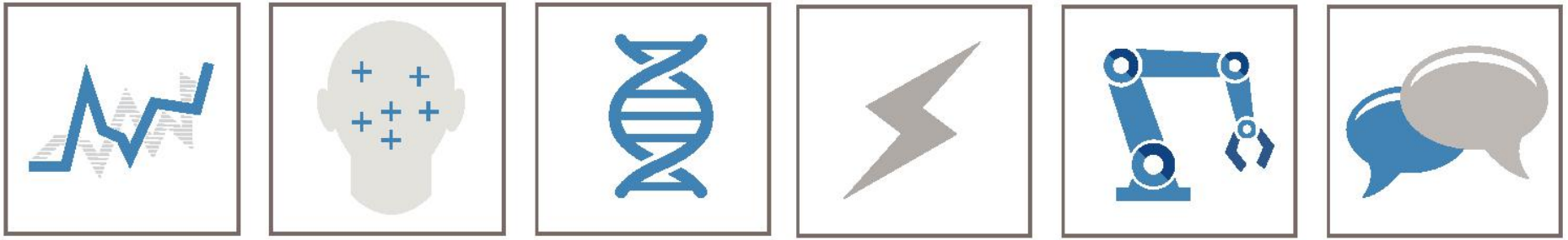
Chapter Contents

1. Deep Learning
2. Machine Learning
3. Swarm Intelligence
4. Heredity Algorithm
5. Quantum Computing
6. DNA Computing
7. Neuromorphic Computing



Section Contents

1. What is Machine Learning
2. Why Use Machine Learning
3. Fundamentals of Machine Learning
4. Applications
5. Interpretability



What is Machine Learning [1]

- Machine learning (**ML**) teaches computers to do what comes naturally to humans and animals: **learn from experience**.
- ML algorithms use computational methods to **"learn" information directly from data** without relying on a ***predetermined*** equation as a model.
- The ML algorithms **adaptively improve their performance** as the number of **samples available** for learning **increases**.
- ML algorithms find **natural patterns** in data that generate insight and help you make better **decisions** and **predictions**.

Timeline

with development of **Algorithms**, **Data** and **HPC**

1950s–1970s

Neural Networks

Early work with neural networks stirs excitement for "thinking machines."

1980s–2010s

Machine Learning

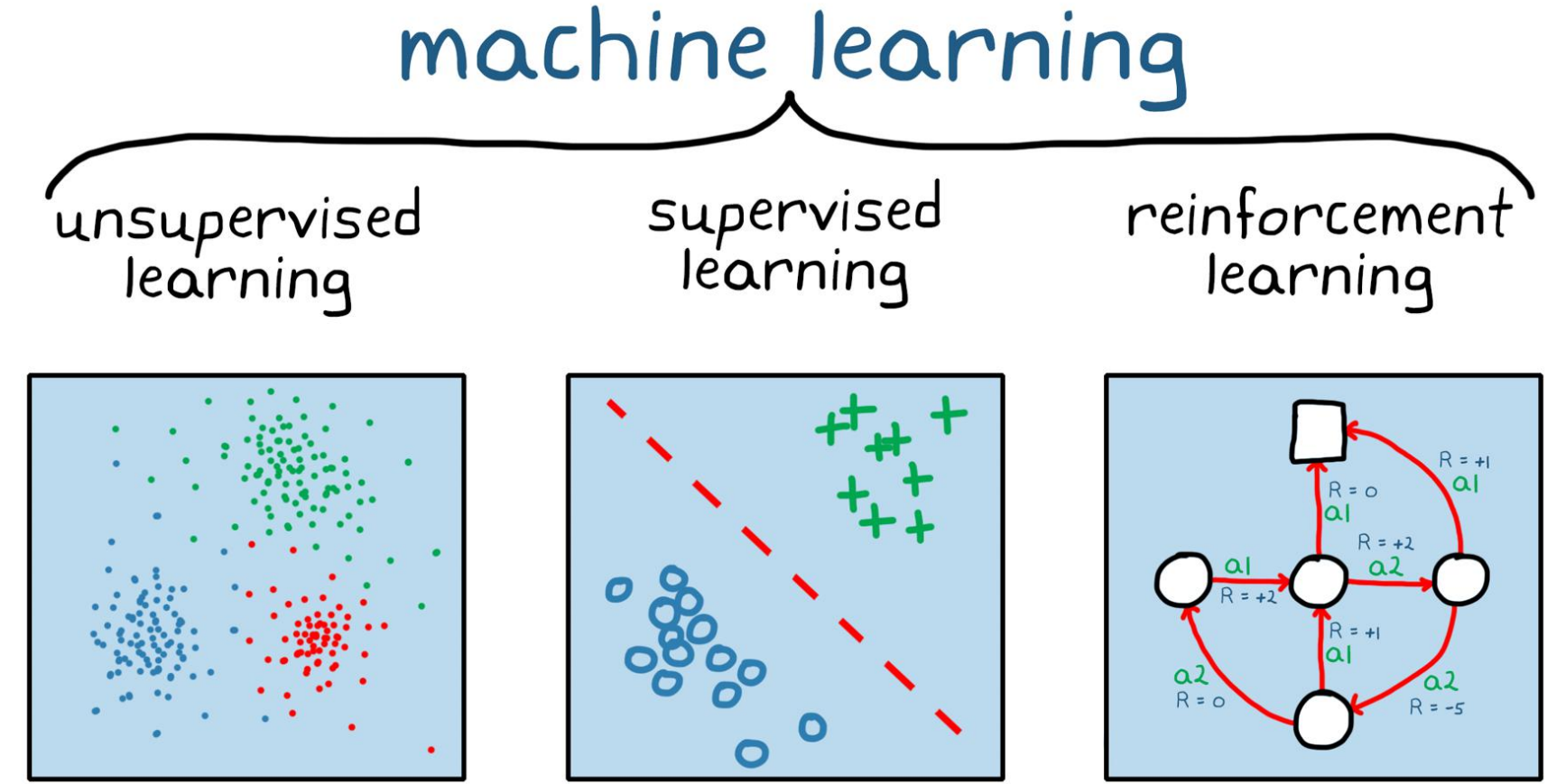
Machine learning becomes popular.

Present Day

Deep Learning

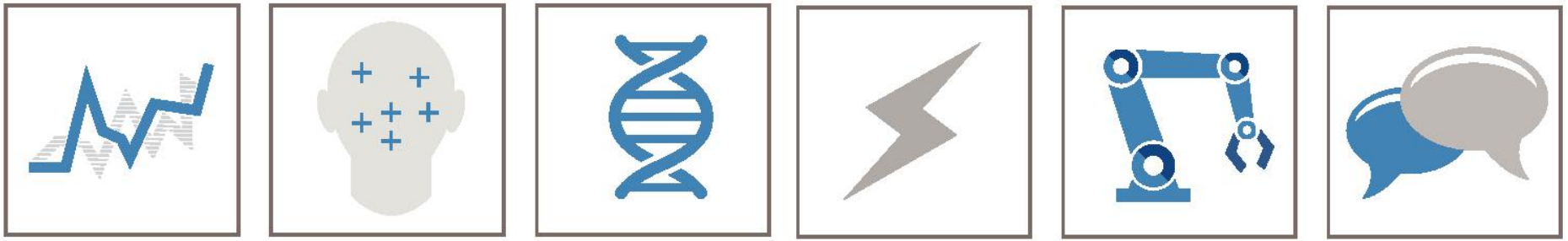
Deep learning breakthroughs drive AI boom.

Learning Paradigms (types of ML)



Section Contents

1. What is Machine Learning
2. Why Use Machine Learning
3. Fundamentals of Machine Learning
4. Applications
5. Interpretability



Why Use Machine Learning

- we've been able to **achieve** meaningful, useful **accuracy** on tasks that matter.
- ML has been used for classification on **images** and **text** for decades, but it struggled to cross the threshold – there's a baseline **accuracy** that algorithms need to have to work in business settings.
- ML is finally enabling us to cross that line in places we weren't **able** to before.

When Should You Use Machine Learning?

- Consider using ML when you have a **complex task** or problem involving **a large amount of data** and **lots** of **variables**, but **no existing formula** or equation. For example, machine learning is a good option if you need to handle situations like these:
- **Hand-written** rules and equations are too complex-as in face recognition and speech recognition.
- The rules of a task are constantly changing-as in **fraud detection** from transaction records.
- The nature of the data keeps changing, and the program needs to adapt-as in **automated trading**, energy demand forecasting, and predicting shopping trends.

Section Contents

1. What is Machine Learning
2. Why Use Machine Learning
3. Fundamentals of Machine Learning
4. Applications
5. Interpretability

Fundamentals of Machine Learning

Machine learning uses two types of techniques:

- **Unsupervised learning**, which **finds** hidden **patterns** or intrinsic structures in input data.

'05-01-01 Unsupervised Learning.pptx'

- **Supervised learning**, which trains a model on known input and output data so that it can **predict** future outputs

'05-01-02 Supervised Learning.pptx'

- **Reinforcement Learning** will be discussed in *'05-01-03 Reinforcement Learning.pptx'*

Questions to Consider Before You Start

Every ML workflow begins with **three** questions:

- *What kind of data are you working with?*
- *What insights do you want to get from it?*
- *How and where will those insights be applied?*

Your answers to these questions help you decide whether to use **supervised** or **unsupervised** learning.

Choose supervised learning if you need to train a model to make a prediction--for example, the future value of a continuous variable, such as temperature or a stock price, or a classification—for example, identify makes of cars from webcam video footage.

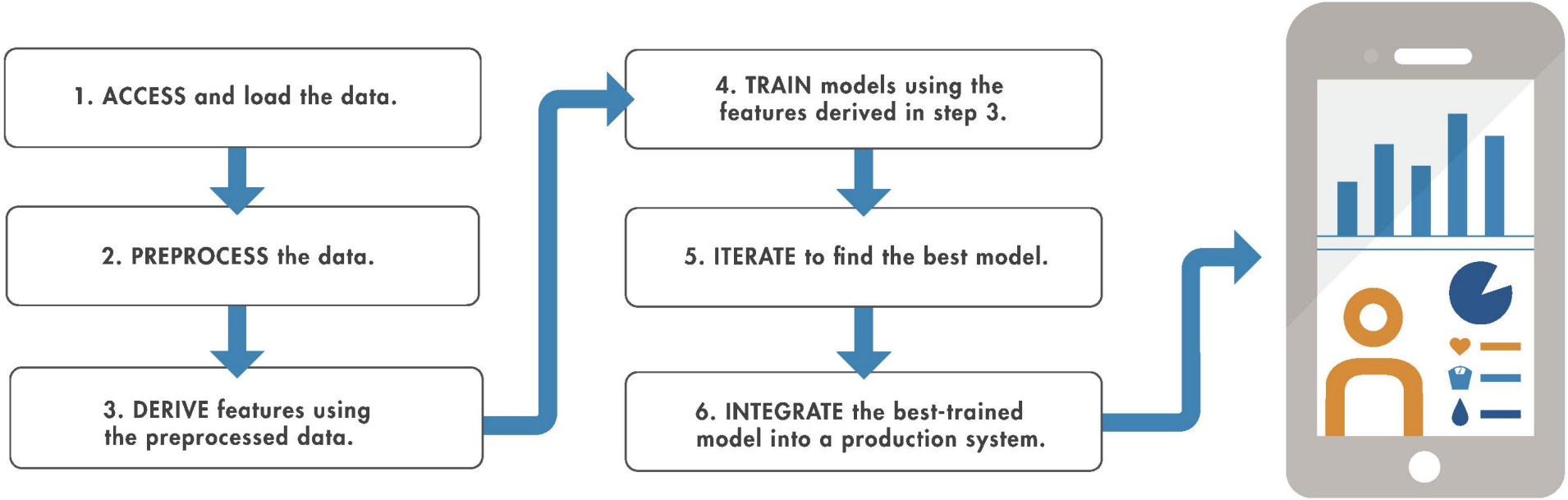
Supervised Learning

Choose unsupervised learning if you need to explore your data and want to train a model to find a good internal representation, such as splitting data up into clusters.

Unsupervised Learning

Workflow at a Glance

Training a Model to Classify Physical Activities



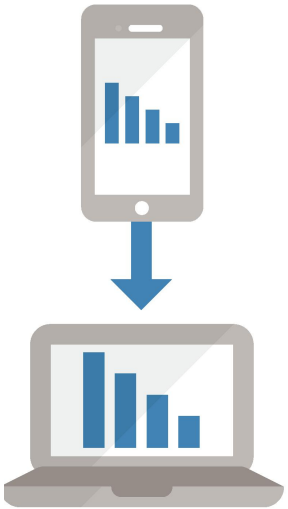
Step 1: Load the Data

To load data from the **accelerometer** and **gyroscope** we do the following:

1.Sit down holding the phone, log data from the phone, and store it in a text file labeled **'Sitting'**;

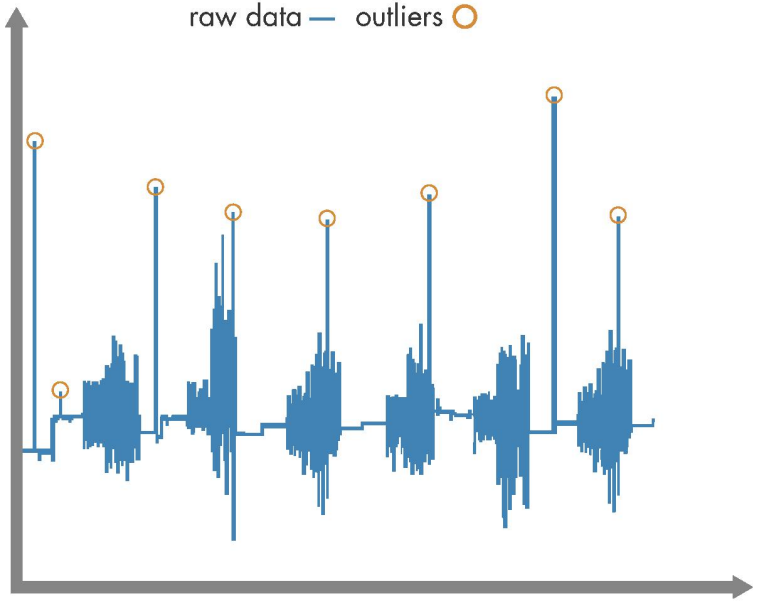
2.Stand up holding the phone, log data from the phone, and store it in a second text file labeled **'Standing'**;

3.**Repeat** the steps until we have data for each activity we want to classify.



Step 1: Load the Data

- Machine learning algorithms **aren't smart** enough to tell the difference between **noise** and **valuable** information.
- Before using the data for training, we need to make sure it's **clean and complete.**



Step 2: Preprocess the Data

- 1. Look for **outliers**–data points that **lie outside** the rest of the data. (find **strange** data points)
- 2. Check for **missing** values (perhaps we **lost** data because the connection dropped during recording). We could simply ignore the missing values, but this will reduce the size of the data set. Alternatively, we could **substitute** approximations for the missing values by **interpolating** or using **comparable** data from **another** sample.

Step 2: Preprocess the Data

- 3.**Remove gravitational** effects from the accelerometer data so that our algorithm will **focus** on the **movement** of the **subject**, not the movement of the phone. A simple high-pass filter such as a biquad filter is commonly used for this.
- 4. Divide the data into **two sets**. We save part of the data for **testing** (the **test** set) and **use the rest** (the **training** set) to build models. This is referred to as holdout, and is a useful cross-validation technique.

Step 3: Derive Features

- **Deriving features** (also known as feature engineering or feature extraction) is one of the most important parts of machine learning.
- It turns **raw data into information** that a machine learning algorithm can use.

Step 3: Derive Features

- For the activity tracker, we want to **extract features** that capture the frequency content of the accelerometer data.
- These features will help the **algorithm** distinguish between **walking** (low frequency) and **running** (high frequency).
- We create a new table that includes the selected features.

Step 3: Derive Features

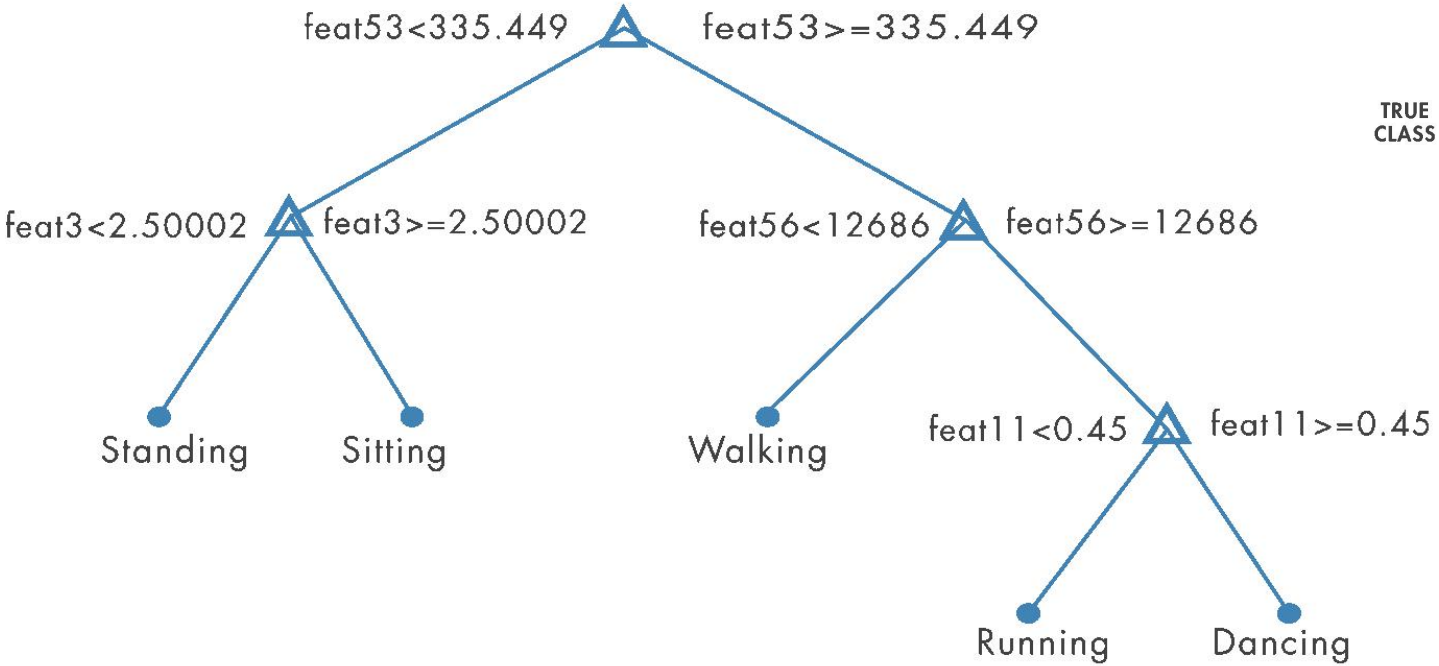
- **Sensor data**
  - ✓ Peak analysis
  - ✓ Pulse and transition metrics
  - ✓ Spectral measurements
- **Image and video data**
  - ✓ Bag of visual words
  - ✓ Histogram of oriented gradients (HOG)
  - ✓ Minimum eigenvalue algorithm
  - ✓ Edge detection

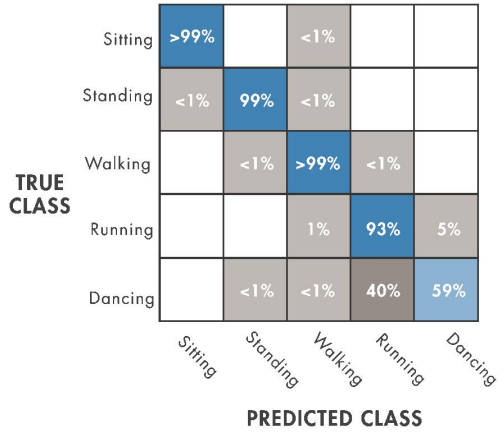
Step 3: Derive Features

- **Transactional data**
  - ✓ Timestamp decomposition
  - ✓ Aggregate value calculation

Step 4: Build and Train the Model

When building a model, it's a good idea to start with something simple; it will be faster to run and easier to interpret.





Step 5: Improve the Model

Improving a model can take two different directions: make the model **simpler** or add **complexity**.

Simplify

- Correlation matrix
- Principal component analysis (PCA)
- Sequential feature reduction

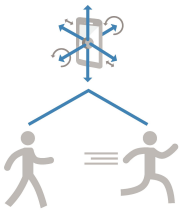
Next, we look at ways to reduce the model itself. We can do this by:

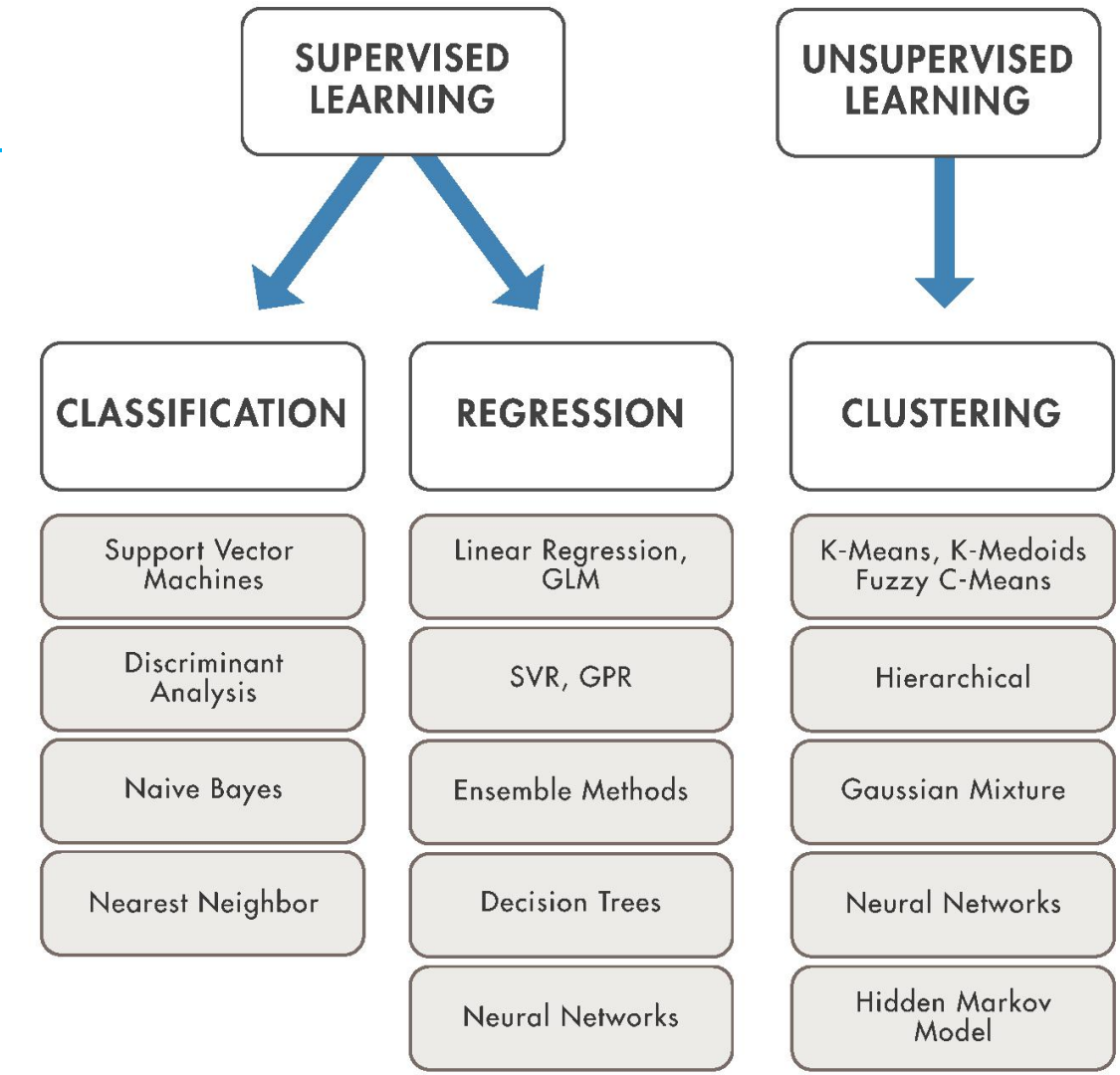
- Pruning branches from a decision tree
- Removing learners from an ensemble

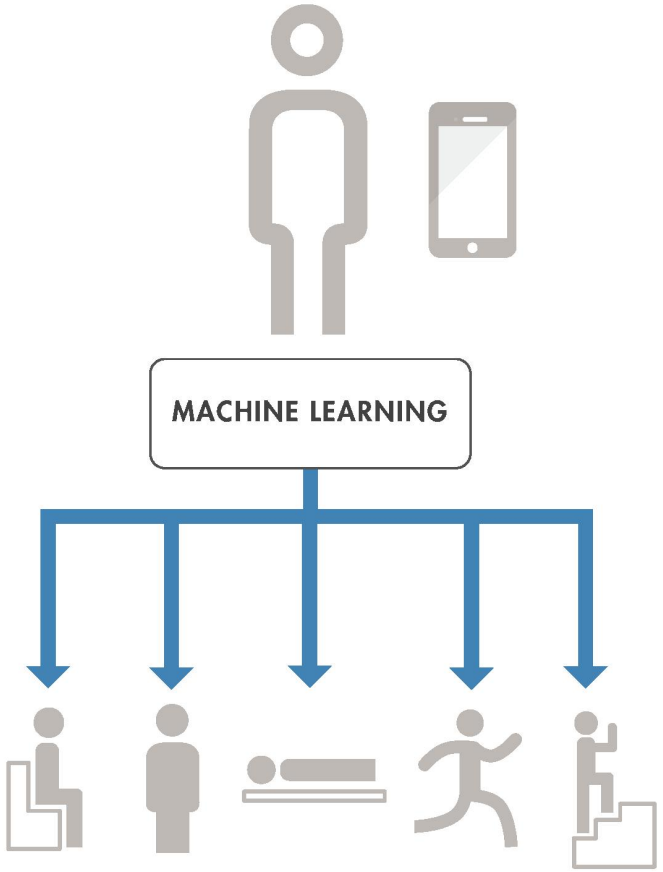
Step 5: Improve the Model

Add Complexity:

- Use model combination - **merge** multiple simpler models into a larger model that is better able to represent the trends in the data than any of the simpler models could on their own.
- Add more data sources - look at the **gyroscope** data as well as the **acceleromter** data. The gyroscope records the orientation of the cell phone during activity. This data might provide unique signatures for the different activities; for example, there might be a combination of acceleration and rotation that's unique to running.

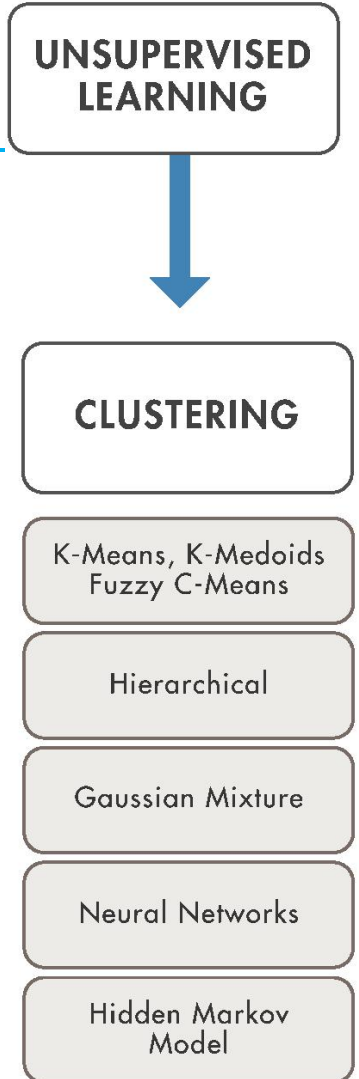


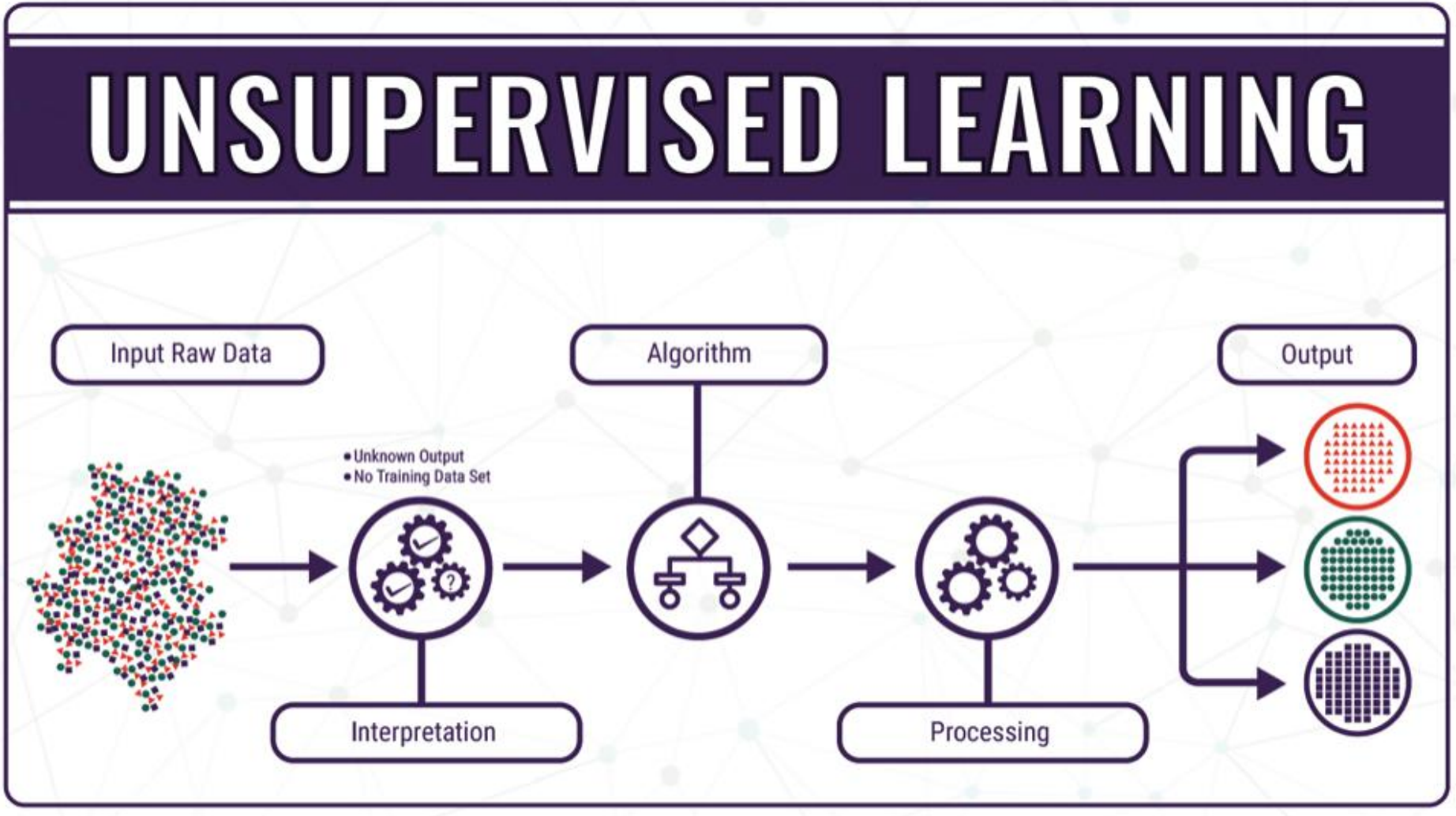

SUPERVISED LEARNING
UNSUPERVISED LEARNING
CLASSIFICATION
REGRESSION
CLUSTERING
Support Vector Machines
Discriminant Analysis
Naive Bayes
Nearest Neighbor
Linear Regression, GLM
SVR, GPR
Ensemble Methods
Decision Trees
Neural Networks
K-Means, K-Medoids Fuzzy C-Means
Hierarchical
Gaussian Mixture
Neural Networks
Hidden Markov Model


MACHINE LEARNING

When to Consider Unsupervised Learning

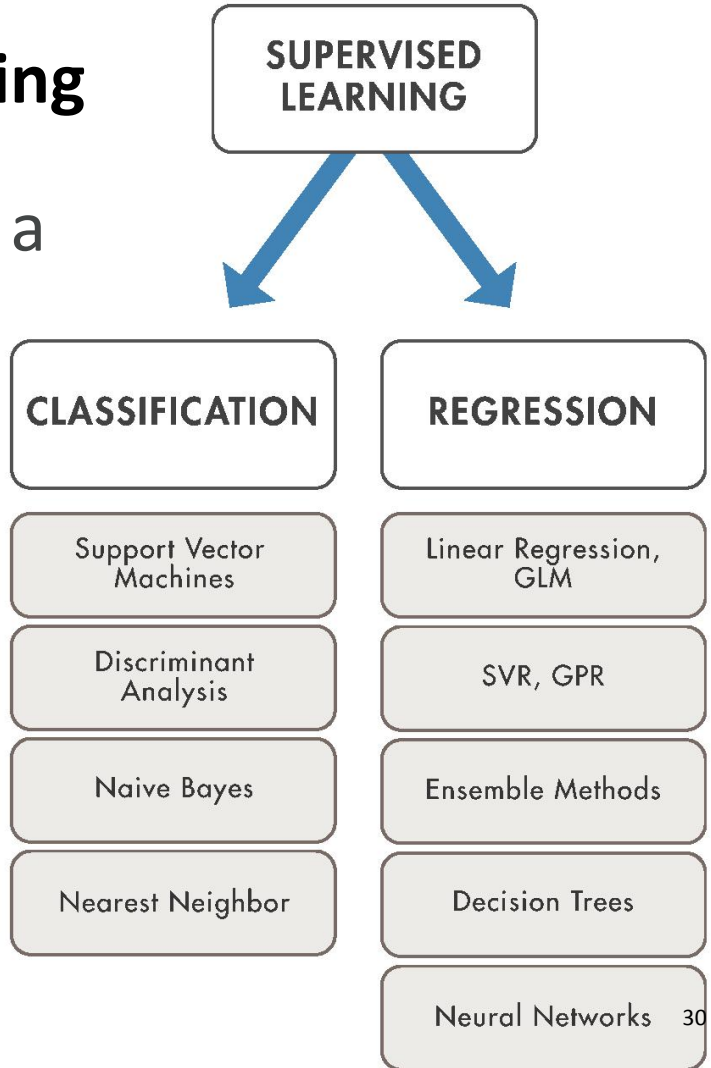
- **Unsupervised learning** is useful when you want to explore your data - **Clustering**, but **don't** yet have a specific goal or are not sure what information the data contains.
- It's also a good way to **reduce the dimensions** of your data.



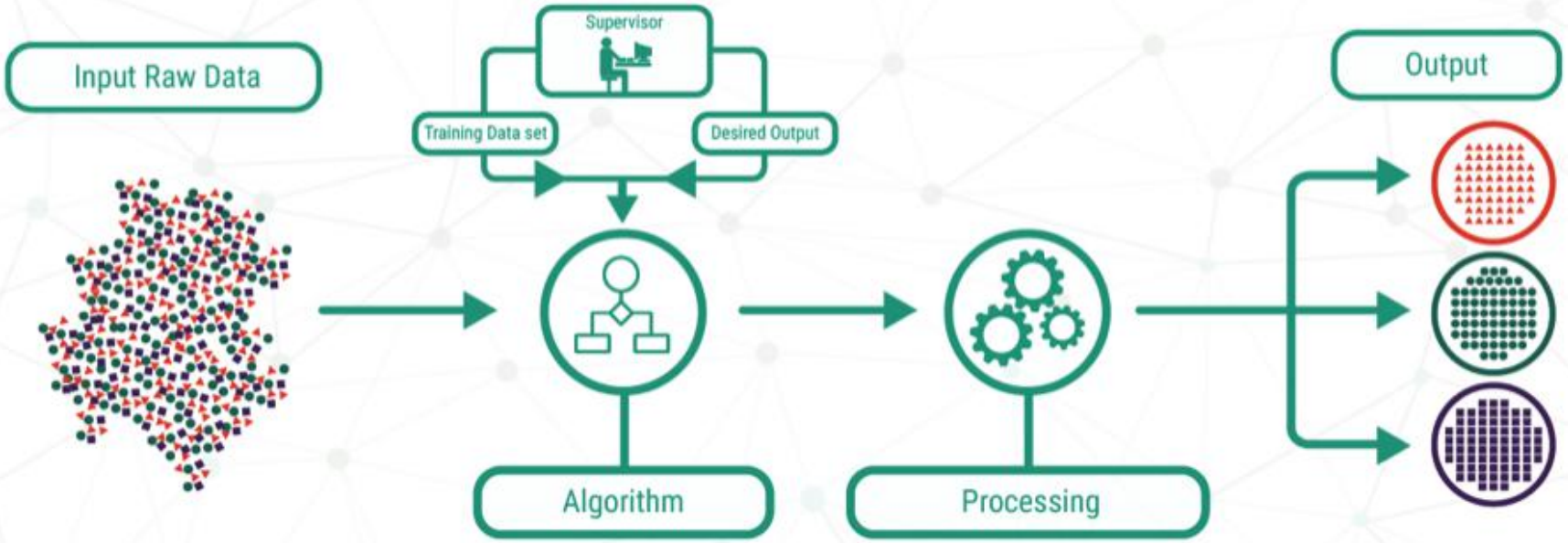

UNSUPERVISED LEARNING
Input Raw Data
Unknown Output
No Training Data Set
Interpretation
Algorithm
Processing
Output

When to Consider Supervised Learning

A supervised learning algorithm takes a **known set of input data** (the training set) and **known responses** to the data (output), and trains a model to generate reasonable predictions for the response to new input data. Use supervised learning if you have **existing data** for the output you are trying to **predict**.



SUPERVISED LEARNING



Semi-Supervised Learning

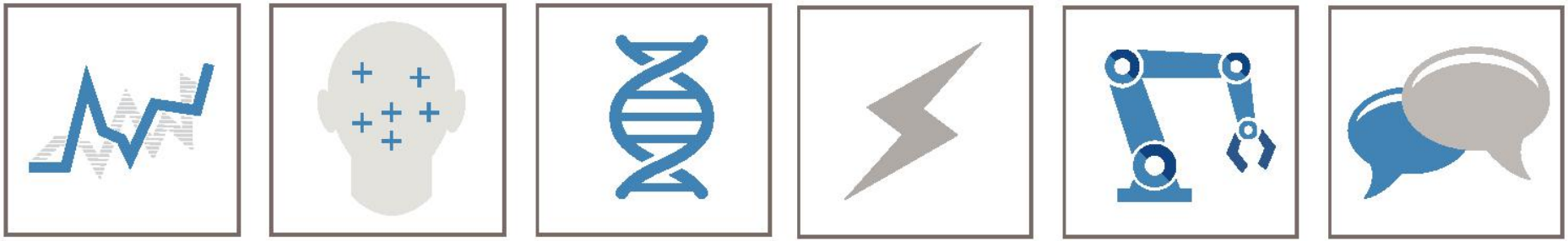
- Semi-supervised learning offers a happy **medium** between supervised and unsupervised learning.
- During training, it uses a **smaller labeled** data set to guide classification and feature extraction from a larger, **unlabeled** data set.
- Semi-supervised learning can solve the problem of having **not enough** labeled data (or not being able to afford to label enough data) **to train a supervised learning** algorithm.

Semi-Supervised Learning

- Semi-supervised learning is a **happy medium**, where you use a training dataset with **both labeled and unlabeled data**.
- It's useful when it’ s difficult to extract relevant features from data and when you have a **high volume of data**.
- Semi-supervised learning is ideal for **medical images,** where **a small amount** of training data can lead to a significant improvement in accuracy. For example, a radiologist can label a small subset of CT scans for tumors

Section Contents

1. What is Machine Learning
2. Why Use Machine Learning
3. Fundamentals of Machine Learning
4. Applications
5. Interpretability



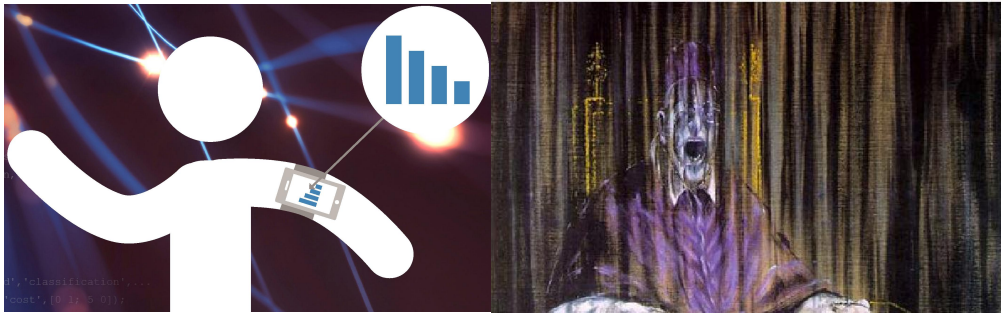
Applications

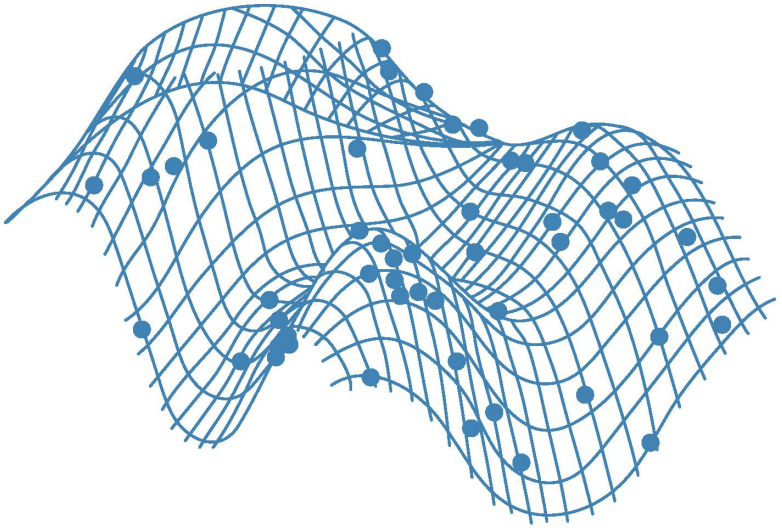
With the rise in big data, ML has become particularly important for solving problems in areas like these:

- Computational **finance**, for credit scoring and algorithmic trading
- **Image** processing and **computer vision**, for face recognition, motion detection, and object detection
- Computational **biology**, for tumor detection, drug discovery, and DNA sequencing
- **Energy** production, for price and load forecasting
- **Automotive**, aerospace, and manufacturing, for predictive maintenance
- Natural language processing

Case Studies [1,2]

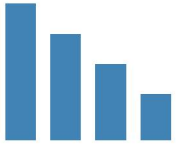
- Creating Algorithms that Can Analyze Works of Art
- Optimising HVAC Energy Usage in Large Buildings
- Detecting Low-Speed Car Crashes

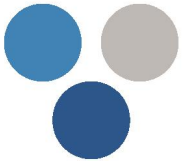




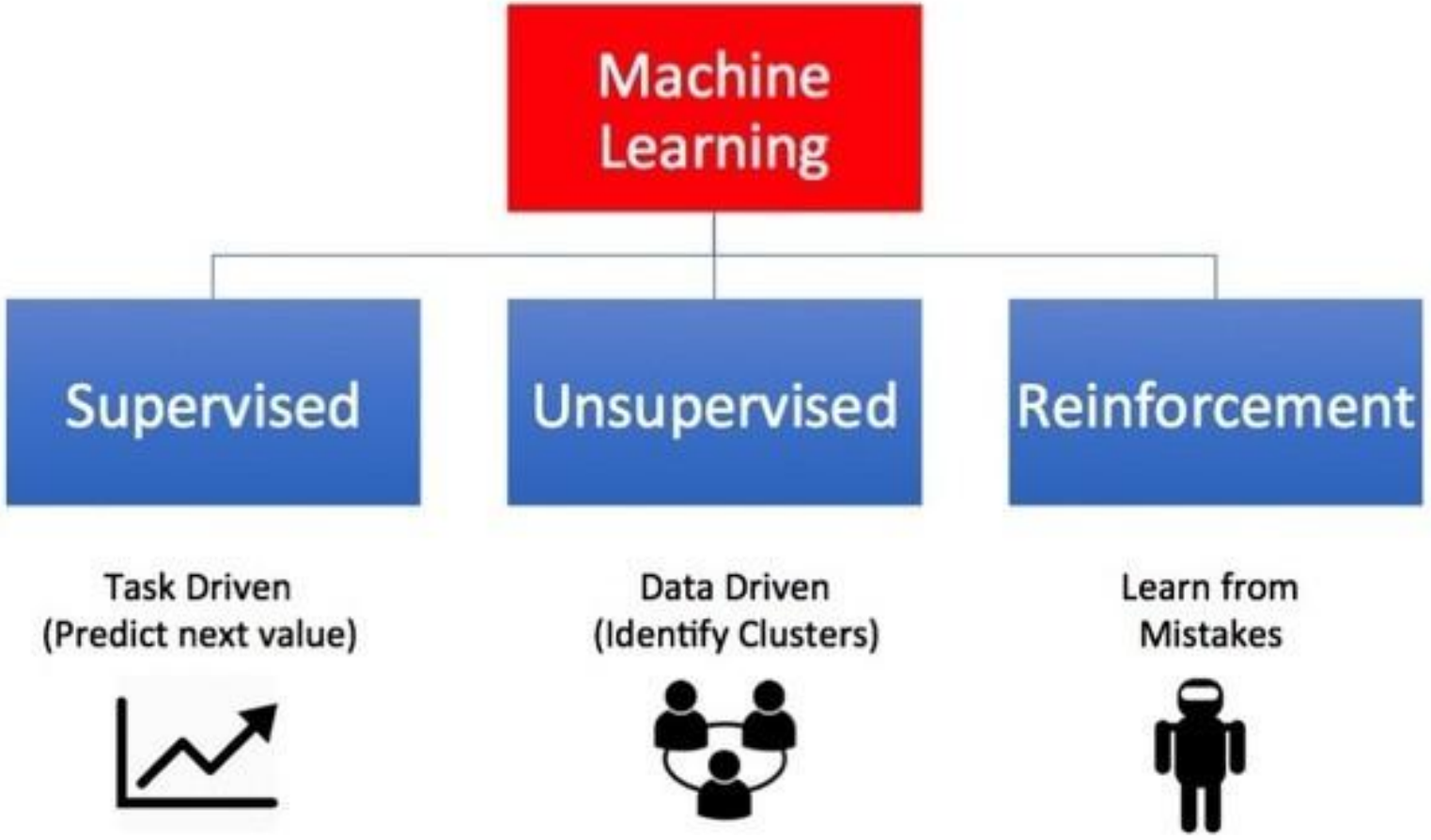
Machine Learning Challenges

- Data comes in **all shapes** and **sizes** (non-standard data)
- Preprocessing your data might require **specialised** knowledge and tools.
- It takes **time** to find the best **model** to fit the data.

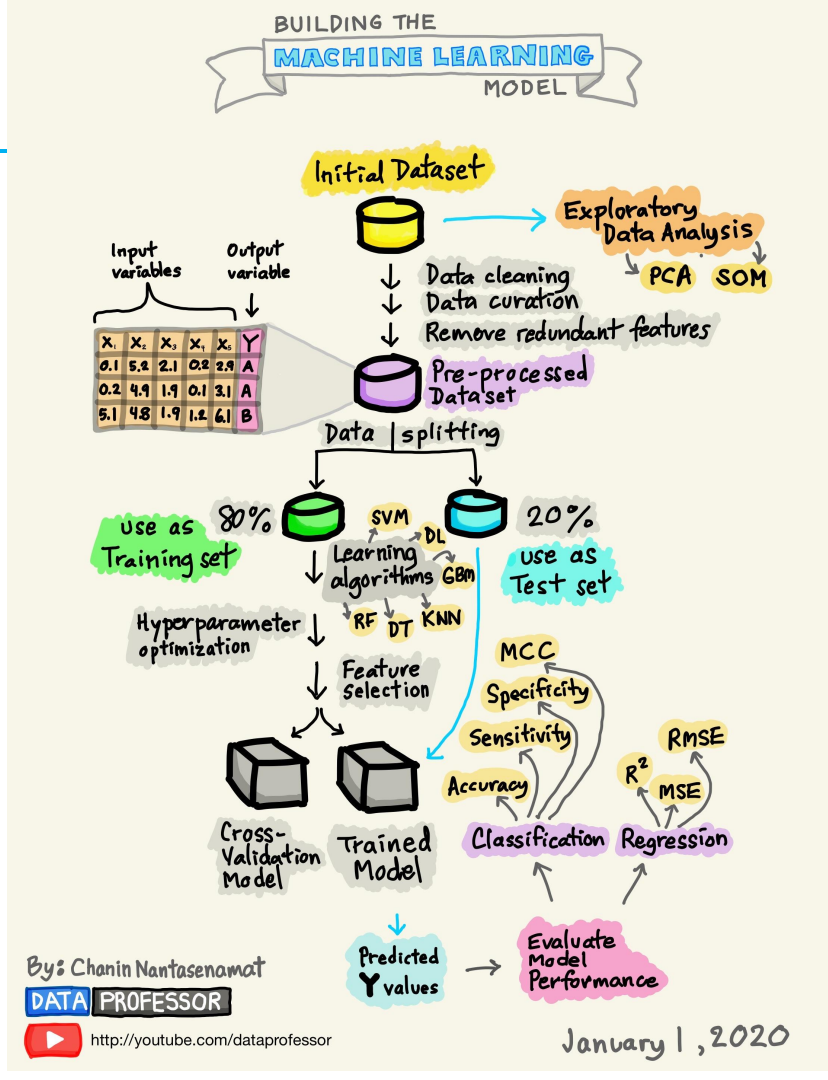


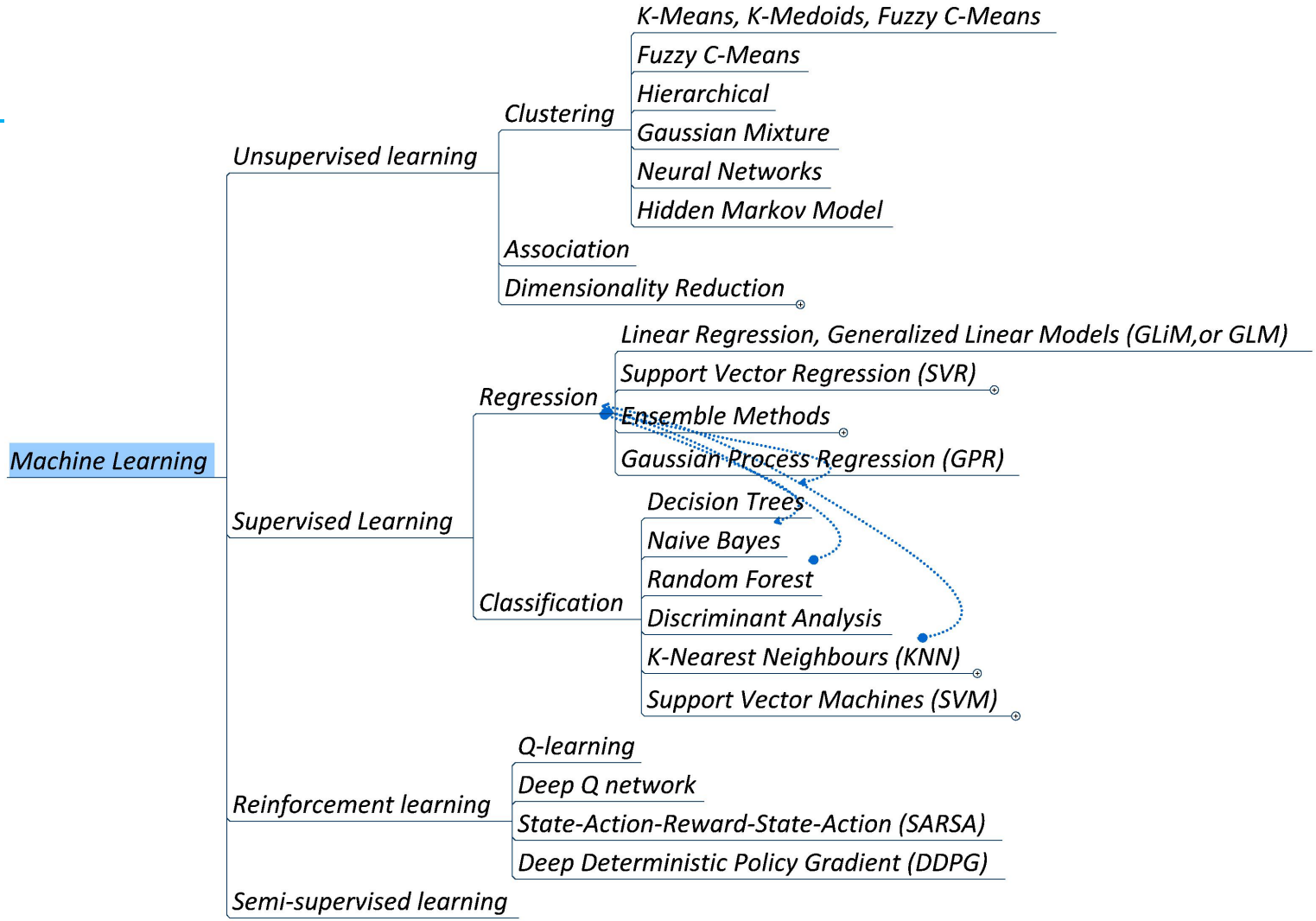


Types of Machine Learning



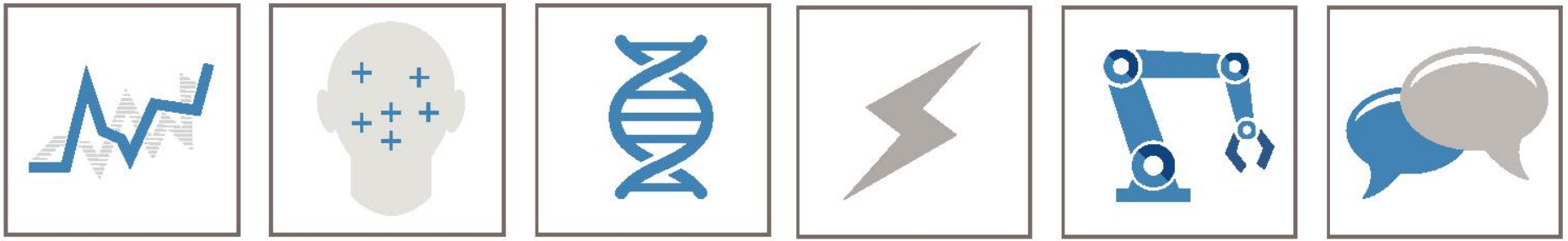
[4] Data Science Infographic
https://github.com/dataprofessor/infographic




Machine Learning
Unsupervised learning
Clustering
K-Means, K-Medoids, Fuzzy C-Means
Fuzzy C-Means
Hierarchical
Gaussian Mixture
Neural Networks
Hidden Markov Model
Association
Dimensionality Reduction
Supervised Learning
Regression
Linear Regression, Generalized Linear Models (GLiM,or GLM)
Support Vector Regression (SVR)
Ensemble Methods
Gaussian Process Regression (GPR)
Classification
Decision Trees
Naive Bayes
Random Forest
Discriminant Analysis
K-Nearest Neighbours (KNN)
Support Vector Machines (SVM)
Reinforcement learning
Q-learning
Deep Q network
State-Action-Reward-State-Action (SARSA)
Deep Deterministic Policy Gradient (DDPG)
Semi-supervised learning

Section Contents

1. What is Machine Learning
2. Why Use Machine Learning
3. Fundamentals of Machine Learning
4. Applications
5. Interpretability



Interpretability

- **ML** models can be astonishingly good at making **predictions**, but they often **can’ t yield explanations** for their forecasts in terms that humans can easily understand.
- The features from which they draw conclusions can be so **numerous**, and their **calculations** so **complex**, that researchers can find it impossible to establish exactly **why** an algorithm produces the **answers** it does.

Interpretability vs. Explainability

- "**interpretability**," is a very active area of investigation among AI researchers in both academia and industry.
- It differs *slightly* from "explainability" –answering why–in that it can reveal causes and effects of changes within a model, even if the model's internal workings remain opaque.

Interpretability Methods

- Partial Dependence Plot (PDP)；
- Individual Conditional Expectation (ICE)
- Permuted Feature Importance
- Global Surrogate
- Local Surrogate (LIME)
- Shapley Value (SHAP)

Videos

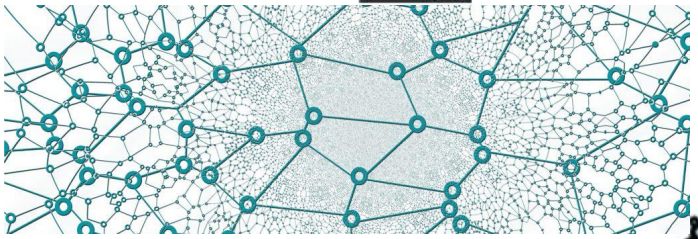
1. Machine Learning Basics (07:15)
2. The 7 steps of machine learning (10:35)

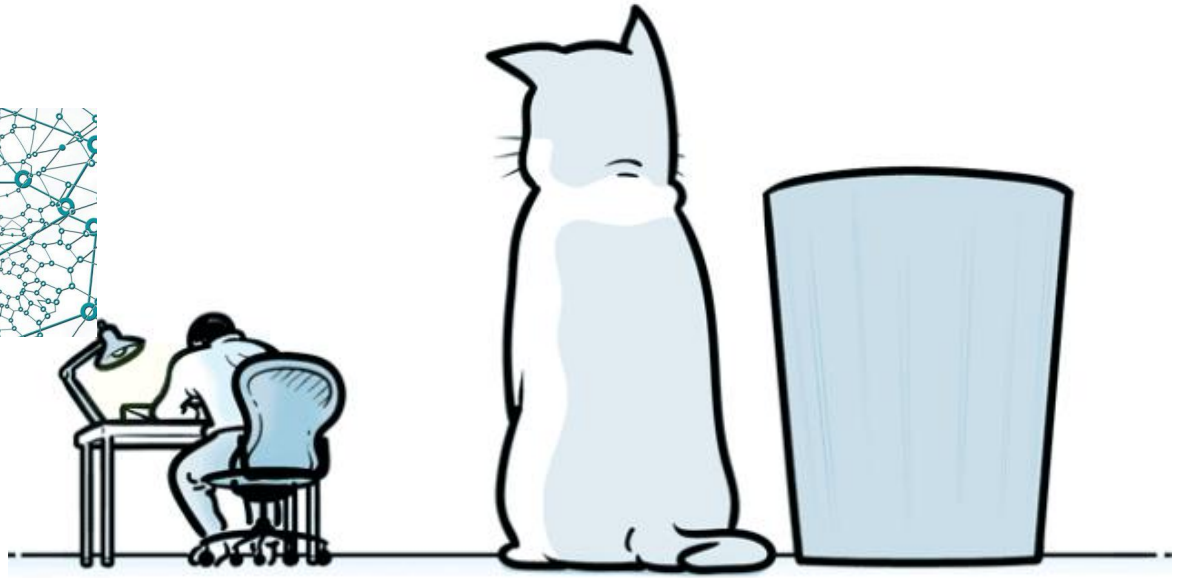
Class Discussion:

The Five Tribes of Machine Learning: [PedroDomingos2015]

- Symbolists
- Connectionists
- Evolutionaries
- Bayesians
- Analogizers

Thanks and Questions





Introduction to Artificial Intelligence
- 05-01 Machine Learning

Dr Leo Chen
leo.chen@ieee.org
06/Sep/2021

Thanks and Questions