Machine Learning

Copernicus Master in Digital Earth

Charlotte Pelletier

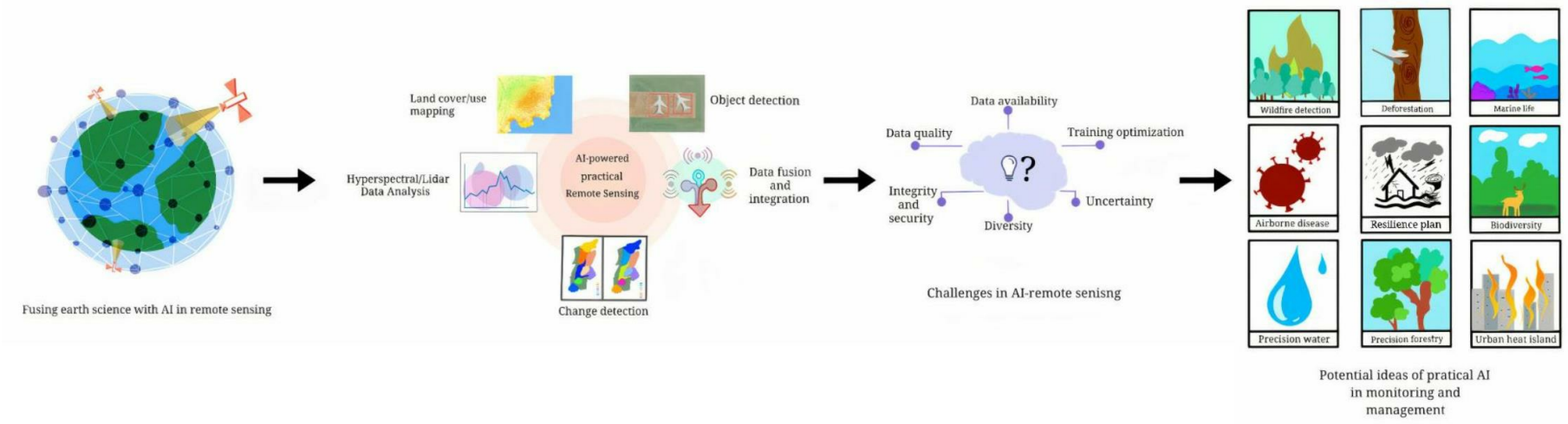
A deluge of Earth Observation data

Source: European Space Agency (ESA)

Estimated Daily Copernicus EO Data Acquisition $\sim$ 20 TB of data acquired by Sentinel satellites every day

Introduction

This course is motivated by the abundant presence of Earth observation data, and the need to analyse them automatically and continuously.



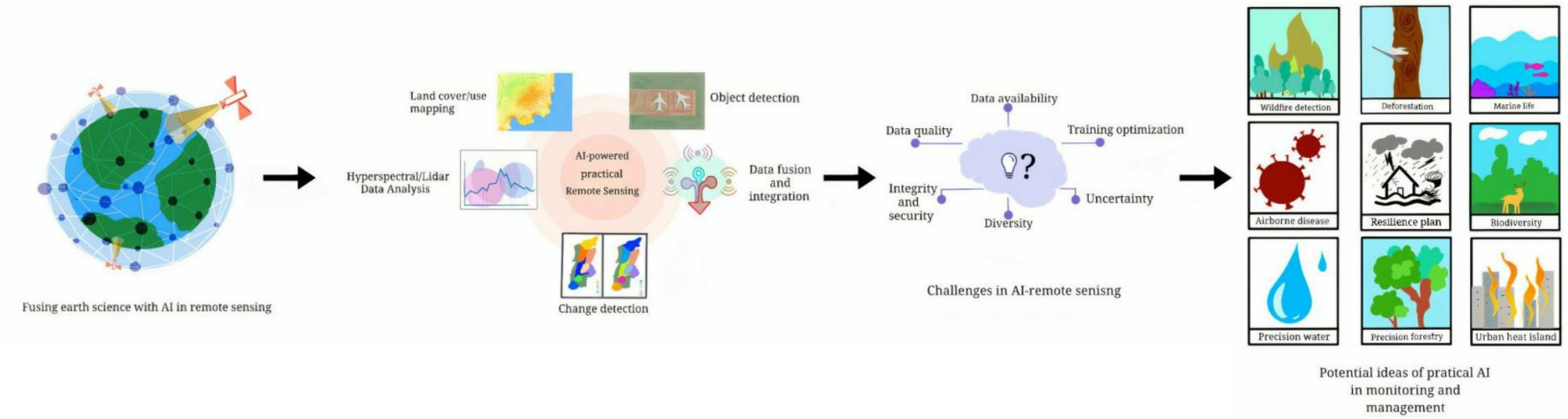
Challenges

- data complexity and volume
- model generalization and transferability
- operational and real-world deployment
- interpretability and trust

Janga, B., Asamani, G. P., Sun, Z., & Cristea, N. (2023). A review of practical ai for remote sensing in earth sciences. Remote Sensing, 15(16), 4112.

Introduction

This course is motivated by the abundant presence of Earth observation data, and the need to analyse them automatically and continuously.



Challenges

- data complexity and volume
- model generalization and transferability
- operational and real-world deployment
- interpretability and trust

Janga, B., Asamani, G. P., Sun, Z., & Cristea, N. (2023). A review of practical ai for remote sensing in earth sciences. Remote Sensing, 15(16), 4112.

Introduction

This is an introductory course in machine learning with applications to remote sensing and geoscience data. It contains 2 parts:

- **Part I.** Machine Learning Foundations
  - machine learning definition
  - machine learning pipeline
  - data exploration and visualisation
  - data clustering
- **Part II.** Machine Learning Algorithms
  - linear models
  - decision trees
  - support vector machine
  - ensemble methods
  - kernel methods

Charlotte Pelletier

Thomas Corpetti

Audrey Poterie

This course is not exhaustive, but it should enable you to acquire good habits for your future machine learning projects, and also strong foundations to understand unseen machine learning concepts.

Introduction

This is an introductory course in machine learning with applications to remote sensing and geoscience data. It contains 2 parts:

- **Part I.** Machine Learning Foundations
  - machine learning definition
  - machine learning pipeline
  - data exploration and visualisation
  - data clustering
- **Part II.** Machine Learning Algorithms
  - linear models
  - decision trees
  - support vector machine
  - ensemble methods
  - kernel methods

Charlotte Pelletier

Thomas Corpetti

Audrey Poterie

This course is not exhaustive, but it should enable you to acquire good habits for your future machine learning projects, and also strong foundations to understand unseen machine learning concepts.

Syllabus

Each bloc consists of about 48 hours with a mix of

- lectures
- in-class practical sessions in **Python**
- MOOC "Machine learning in Python with Scikit-learn"[1]
- project

Syllabus Part I.: Machine Learnning Foundation

1. Introduction
2. Model selection and hyperparamter tuning
3. Data visualisation and dimensionality reduction
4. Data clustering

Evaluation: more information during the next session.

[1] **Source** : `https://www.fun-mooc.fr/en/courses/machine-learning-python-scikit-learn/`

Syllabus

Each bloc consists of about 48 hours with a mix of

- lectures
- in-class practical sessions in **Python**
- MOOC "Machine learning in Python with Scikit-learn"[1]
- project

Syllabus Part I.: Machine Learnning Foundation

1. Introduction
2. Model selection and hyperparamter tuning
3. Data visualisation and dimensionality reduction
4. Data clustering

Evaluation: more information during the next session.

[1] **Source** : `https://www.fun-mooc.fr/en/courses/machine-learning-python-scikit-learn/`

Part I. Learning Objectives

By the end of Part I, students will be able to:

- Define machine learning and identify its main types and applications in Earth observation.
- Describe the machine learning pipeline and its key steps.
- Explore, clean, and visualise datasets using Python tools.
- Apply dimensionality reduction techniques and interpret results.
- Perform clustering on data and evaluate unsupervised learning outcomes.
- Adopt good practices in experimentation.

Content

Source : https://xkcd.com/1838/

What is Machine Learning (ML)?

Machine learning is a branch of computer science and statistics that enables computers to **learn patterns from data** and make predictions or decisions.

It is often confused and mixed with other concepts:

- **Artificial Intelligence (AI)** – general term, started in 1940, without a strict definition, which includes machine learning, computer vision, NLP, robotics, intelligent agents; often misused as a "hype" term for ML or basic data analysis.
- **Deep Learning (DL)** – a subset of ML using neural networks.
- **Pattern Recognition (PR)** – term often used in signal and image processing.
- **Statistical Learning** – term used in statistics emphasizing the mathematical modeling of data.

Different fields, different names: the underlying idea is the same – learning from data to extract meaningful information and make predictions.

Terminology

Statistical Models	Machine Learning
data points	samples
variables	features
parameters	weights
estimation/fitting	learning
regression/classification	supervised learning
clustering/density estimation	unsupervised learning
response	label
performance	generalization

Not Just a Matter of Terminology?

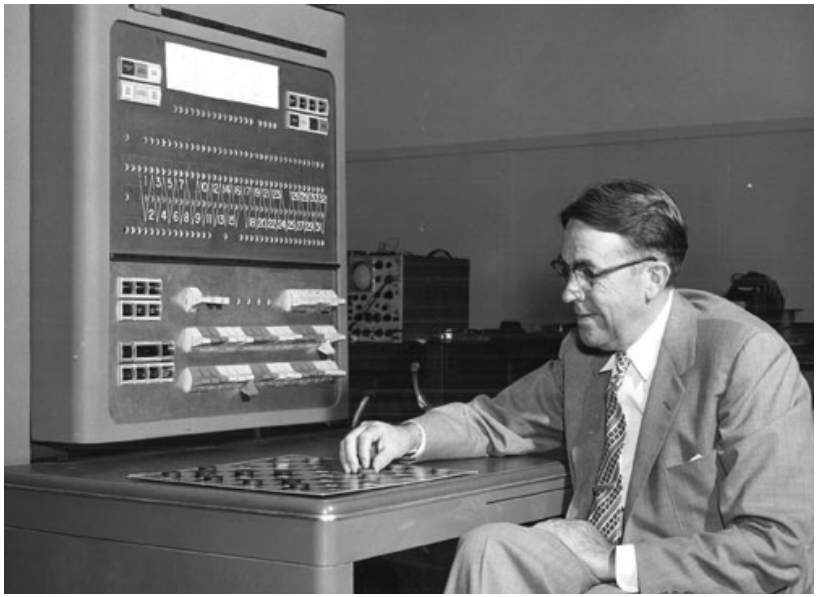
- historically developed as different fields, but many methods and concepts are pretty much the same
- ML: rather accurate predictions with more complex models. *versus*
  Stats: more interpreting relationships and sound inference
- Now: both basically work on same problems with same tools, but communities are still divided

To delve deeper: `https://towardsdatascience.com/the-actual-difference-between-statistics-and-machine-learning-64b49f07ea3`

What is machine learning?

Some definitions provided in the literature

- "Field of study that gives computers the ability to learn without being explicitly programmed" (*Arthur Samuel, 1959*).
- A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E" (*Tom Mitchel, 1997*).



A. Samuel

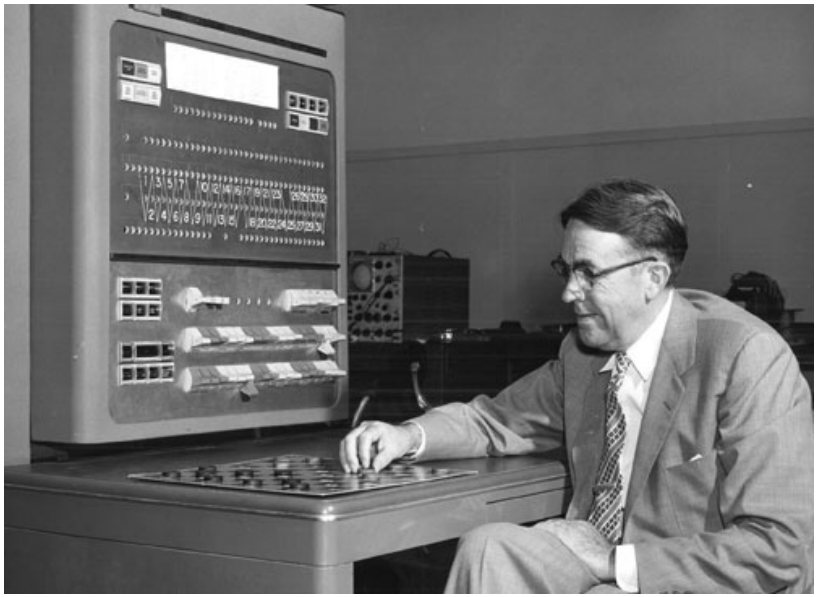
It usually refers to a technique that is

- mathematically well-defined
- solves reasonably narrow tasks
- construct predictive models from data, instead of explicitly programming them

What is machine learning?

Some definitions provided in the literature

- "Field of study that gives computers the ability to learn without being explicitly programmed" (*Arthur Samuel, 1959*).
- A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E" (*Tom Mitchel, 1997*).



A. Samuel

It usually refers to a technique that is

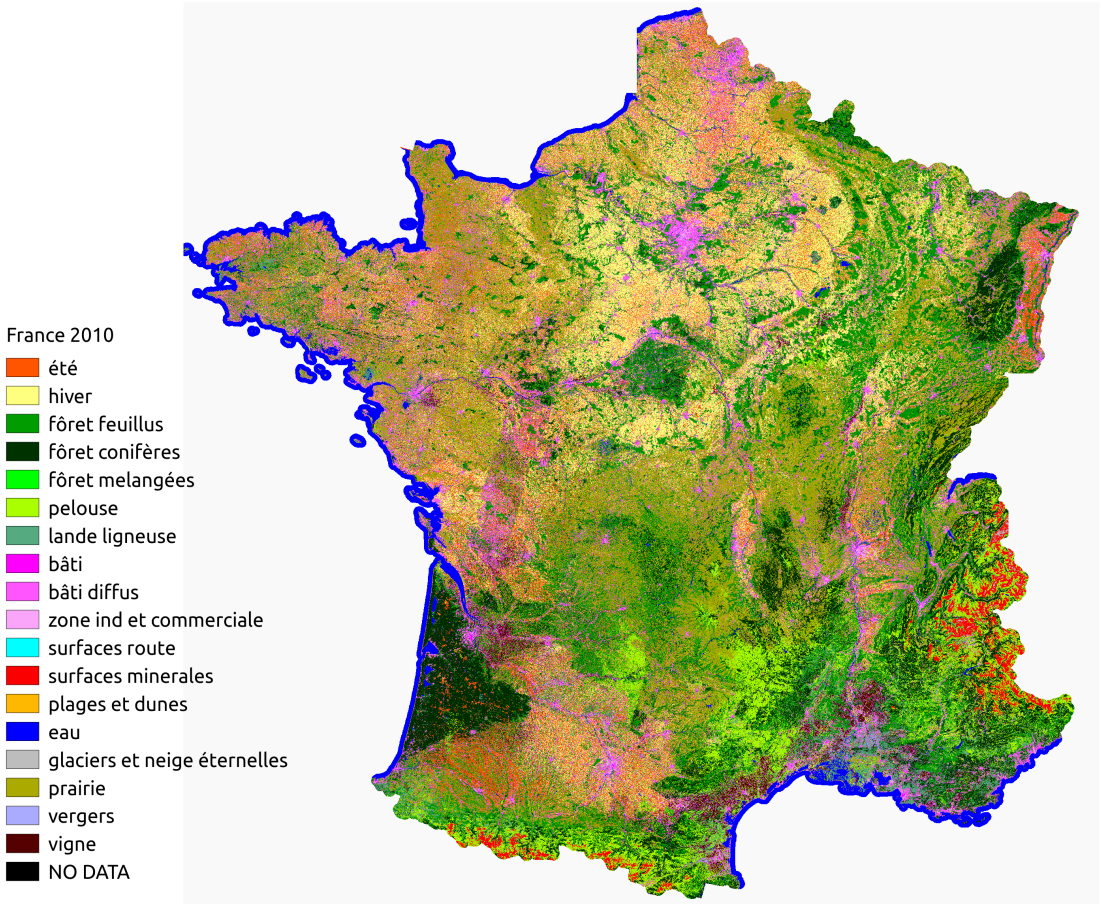
- mathematically well-defined
- solves reasonably narrow tasks
- construct predictive models from data, instead of explicitly programming them

Machine learning is transforming how we interact with the world and how science advances:

- Search engines adapt to the user preferences.
- Recommendation systems understand our tastes in books, music, movies, and more.
- Translation tools bridge numerous languages accurately.
- AI, like DeepMind, surpasses humans in complex games such as Go.
- Large language models (LLMs) revolutionize many domains, currently excelling in coding.
- Data-driven discoveries push the frontiers of physics, biology, genetics, astronomy, chemistry, and beyond.

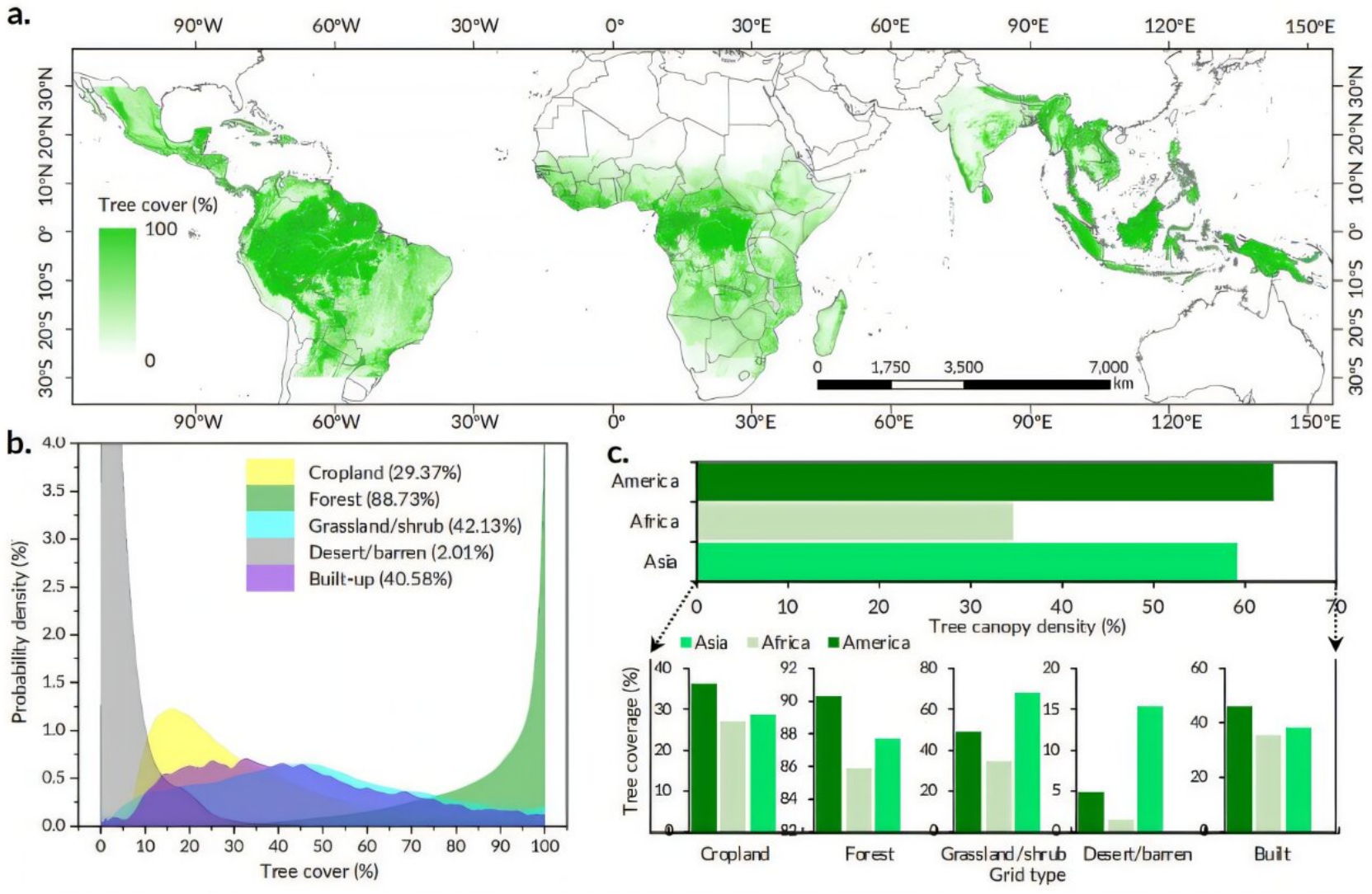
Land cover mapping

French land cover land use map: produced yearly applying Random Forests to Sentinel-2 image time series[2]



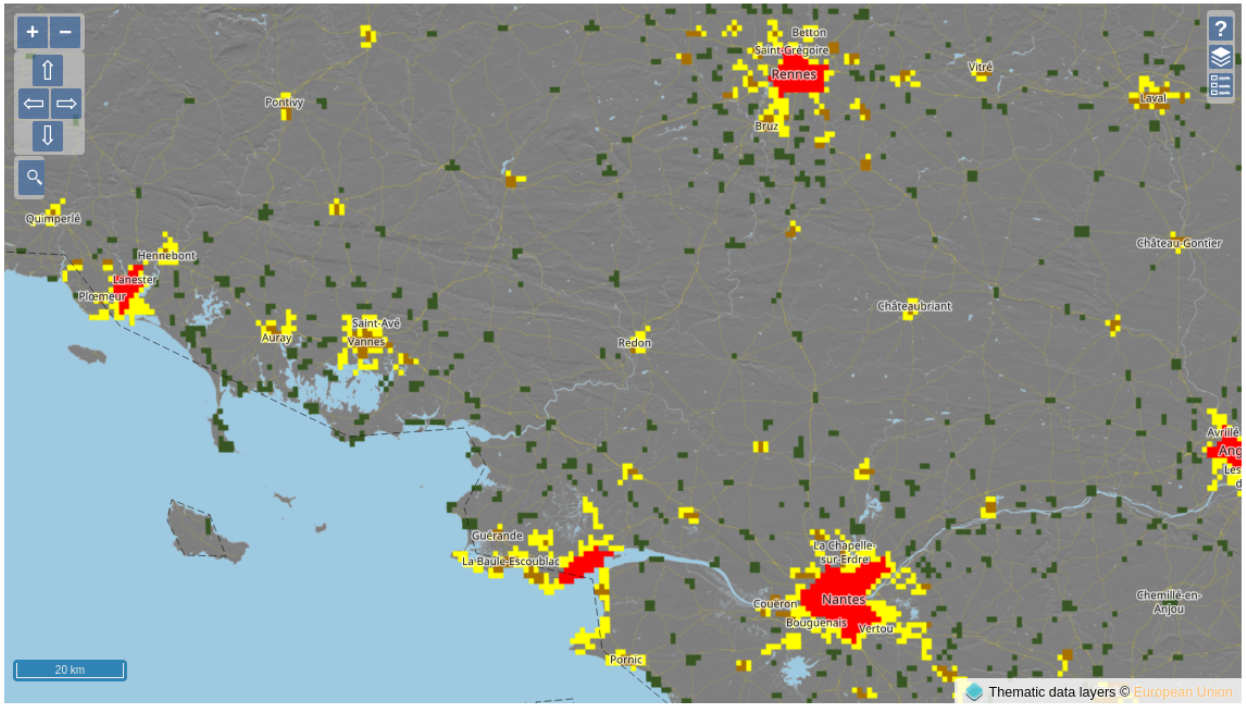
[2] Credit: CESBIO THEIA CES OSO

Pan-tropical tree cover



Source: Liu, S., Zhang, J., Wang, L., Ciais, P., Zhang, J., Penuelas, J., ... & Niu, Z. (2025). Mapping previously undetected trees reveals overlooked changes in pan-tropical tree cover. *Nature Communications*, 16(1), 5561.

Global Human Settlement Mapping



Degree of Urbanisation

Urban centre (City):

- Urban centre (City)

Urban cluster (Town & suburb):

- Dense and semi-dense urban cluster (Town)
- Suburban or peri-urban cells (Suburb)

Rural grid cells (Rural area):

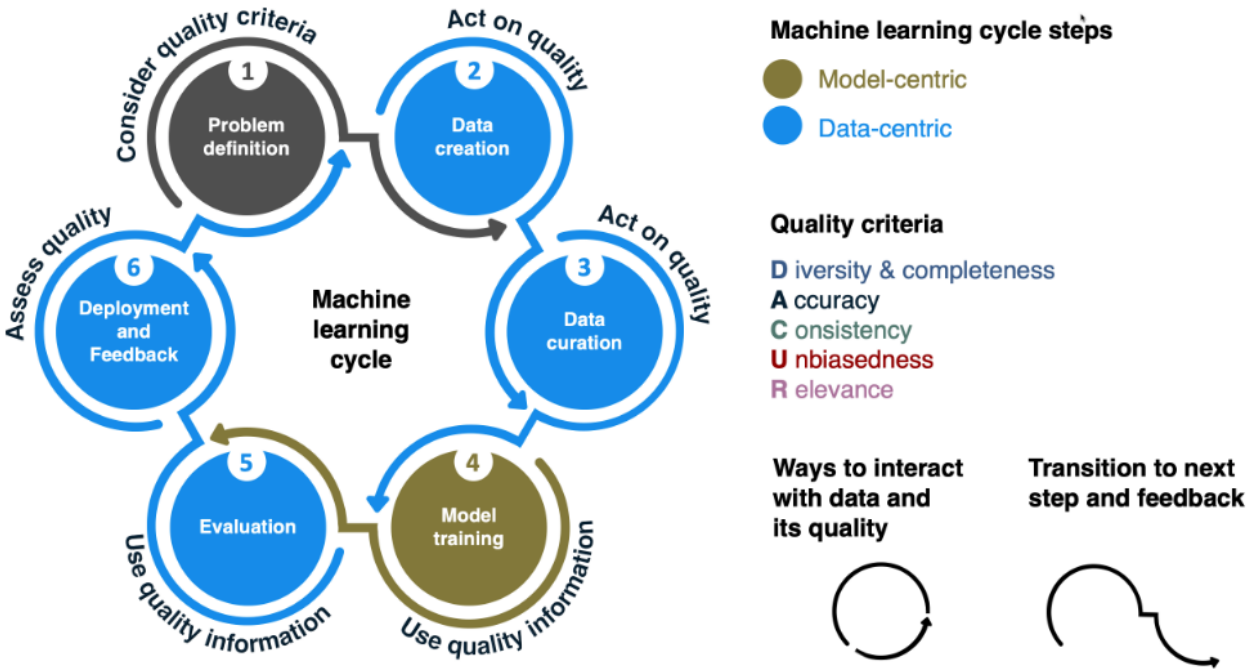
- Rural cluster (Village)
- Low density rural grid cells (Dispersed rural area) - transparent

Source: `https://human-settlement.emergency.copernicus.eu/visualisation.php`

Components of a Machine Learning System

In practice, a typical system consists of

- an acquisition system (sensor, database, manual or automatic labeling
- a set of data preprocessing (cleaning, formatting, conversion, normalization)
- a feature extraction system (manual or automatic; reduction, extraction or selection)
- an algorithm (clustering, classification, regression)
- an evaluation protocol (metrics, generalization, transferability)
- a deployment stage



Source: Roscher, R., Russwurm, M., Gevaert, C., Kampffmeyer, M., Dos Santos, J. A., Vakalopoulou, M., ... & Tuia, D. (2024). Better, not just more: Data-centric machine learning for Earth Observation. *IEEE Geoscience and Remote Sensing Magazine*.

Machine learning objective

Goal: explain or predict an **output** $y_i \in \mathcal{Y}$ from an **input data** $\mathbf{x}_i \in \mathcal{X}$

- $\mathcal{X}$ is the input space, *e.g.*, $\mathcal{X} = \mathbb{R}^d$
- $\mathcal{Y}$ is the output space

A sample (an observed dataset): $\mathcal{D}_m = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_m, y_m)\}$ where $y_i \in \mathcal{Y}$ denotes the outputs and $\mathbf{x}_i \in \mathcal{X}$ the inputs. A pair $(\mathbf{x}_i, y_i)$ is one **observation** (or a sample point).

Terminology

- $\mathbf{x}_i = [x_{i1}, \cdots, x_{id}]^T$ is the input, written as a vector of d features
  $\rightarrow$ **bold** emphasizes that $\mathbf{x}_i$ is a **vector**
  - The d features can be numerical, categorical, or more complex (*e.g.*, images or curves).
  - In this course, we will deal only with numerical or categorical features.
- y_i is the output, also called an answer, a response, or a target

To go further: *Probabilistic Machine Learning: An Introduction* by Kevin Patrick Murphy. Source: `https://probml.github.io/pml-book/book1.html`

Data structure

Input Data:
a matrix $\mathbf{X} \in \mathbb{R}^{m \times d}$ with m observations as rows and d features as columns.
This input matrix can also be written as $\mathbf{X} = [\mathbf{x}_1^T; \ldots; \mathbf{x}_m^T]$. The variables m and d represent the sample size and the feature dimensionality. Their value has an influence on the learning problem.

Features $\rightarrow$ Observations $\downarrow$	X_1	$\cdots$	X_j	$\cdots$	X_d
$\mathbf{x}_1$	x_{11}	$\cdots$	x_{1j}	$\cdots$	x_{1d}
$\vdots$	$\vdots$		$\vdots$		$\vdots$
$\mathbf{x}_i$	x_{i1}	$\cdots$	x_{ij}	$\cdots$	x_{id}
$\vdots$	$\vdots$		$\vdots$		$\vdots$
$\mathbf{x}_m$	x_{m1}	$\cdots$	x_{mj}	$\cdots$	x_{md}

In this course, we will deal only with **tabular data**, or "transformed" complex data.

More terms

Population: we assume that data is generated from an unknown **joint probability distribution** $\mathbb{P}_{XY}$ (sometimes written $P(X, Y)$) defined on $(\mathcal{X} \times \mathcal{Y})$.
Random variables $(\mathbf{x}, y) \sim \mathbb{P}_{XY}$ represent generic draws from this population.

Sample, also known as the **(observed) dataset**:

$$\mathcal{D}_m = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\} \sim (\mathbb{P}_{XY})^m$$

is an independent and identically distributed (i.i.d) sample of size m.

This means

- all observations are drawn from the same distribution, and
- are mutually independent. The i-th realization does not depend on the other $m - 1$ ones.

Other terms

- $\mathcal{D}_m$ is the whole **sample** (the dataset), and $(\mathcal{X} \times \mathcal{Y})^m$ is the set of all datasets of size m
- $(\mathbf{x}_i, y_i)$ is the i-th observation (sample point). Each $(\mathbf{x}_i, y_i)$ is a **realization** of the random variables $(\mathbf{x}, y) \sim \mathbb{P}_{XY}$
- $\mathbb{P}_{XY}$ is the true but unknown joint distribution on $(\mathcal{X} \times \mathcal{Y})$.

The ML's objective is to learn from a sample $\mathcal{D}_m$ to approximate the structure of the true distribution $\mathbb{P}_{XY}$ to predict y.

In ML, the terminology is often blurred: an observation $(\mathbf{x}_i, y_i)$ is sometimes called a sample.

More terms

Population: we assume that data is generated from an unknown **joint probability distribution** $\mathbb{P}_{XY}$ (sometimes written $P(X, Y)$) defined on $(\mathcal{X} \times \mathcal{Y})$.
Random variables $(\mathbf{x}, y) \sim \mathbb{P}_{XY}$ represent generic draws from this population.

Sample, also known as the **(observed) dataset**:

$$\mathcal{D}_m = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\} \sim (\mathbb{P}_{XY})^m$$

is an independent and identically distributed (i.i.d) sample of size m.

This means

- all observations are drawn from the same distribution, and
- are mutually independent. The i-th realization does not depend on the other $m - 1$ ones.

Other terms

- $\mathcal{D}_m$ is the whole **sample** (the dataset), and $(\mathcal{X} \times \mathcal{Y})^m$ is the set of all datasets of size m
- $(\mathbf{x}_i, y_i)$ is the i-th observation (sample point). Each $(\mathbf{x}_i, y_i)$ is a **realization** of the random variables $(\mathbf{x}, y) \sim \mathbb{P}_{XY}$
- $\mathbb{P}_{XY}$ is the true but unknown joint distribution on $(\mathcal{X} \times \mathcal{Y})$.

The ML's objective is to learn from a sample $\mathcal{D}_m$ to approximate the structure of the true distribution $\mathbb{P}_{XY}$ to predict y.

In ML, the terminology is often blurred: an observation $(\mathbf{x}_i, y_i)$ is sometimes called a sample.

Types of Problems

Unsupervised Learning: Understanding Data

- **Clustering**: Organizing objects into groups with a certain similarity (taxonomy of animal species).
- **Probability Density Estimation**: Estimating the probability distribution of training data (estimating the distribution of noise).
- **Dimensionality Reduction**: Reducing the dimensionality of data to better interpret/visualize it (recommendation).

Types of Problems

Unsupervised Learning: Understanding Data

- **Clustering**: Organizing objects into groups with a certain similarity (taxonomy of animal species).
- **Probability Density Estimation**: Estimating the probability distribution of training data (estimating the distribution of noise).
- **Dimensionality Reduction**: Reducing the dimensionality of data to better interpret/visualize it (recommendation).

Supervised Learning: Learning to Predict

- **Classification**: Assigning a class to an observation (character recognition, weather forecasting for rain).
- **Regression**: Predicting a real value based on an observation (temperature forecasting in weather).

Types of Problems

Unsupervised Learning: Understanding Data

- **Clustering**: Organizing objects into groups with a certain similarity (taxonomy of animal species).
- **Probability Density Estimation**: Estimating the probability distribution of training data (estimating the distribution of noise).
- **Dimensionality Reduction**: Reducing the dimensionality of data to better interpret/visualize it (recommendation).

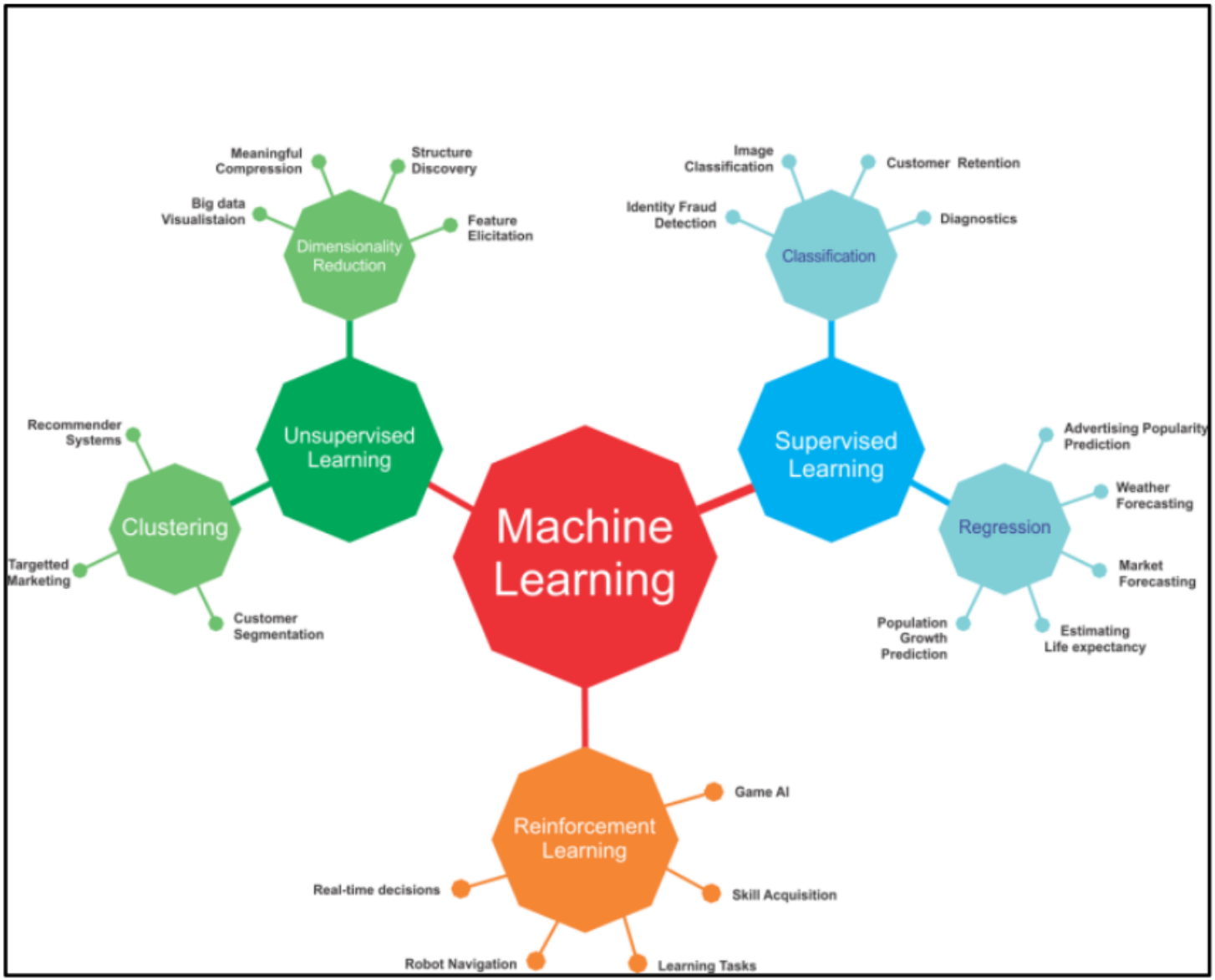
Supervised Learning: Learning to Predict

- **Classification**: Assigning a class to an observation (character recognition, weather forecasting for rain).
- **Regression**: Predicting a real value based on an observation (temperature forecasting in weather).

Reinforcement Learning: Learning Through Play

- Learning to maximize a reward (autonomous driving, games, control systems).

Types of Problems and Application



Source: Skilllx (2020) *List of Machine Learning Algorithms*.

Types of Problems and Associated Data

Unsupervised Learning

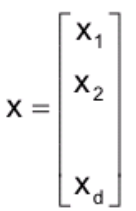
- An observation $\mathbf{x}$ is a predictor with d variables.
- A dataset is defined by m observations $\{\mathbf{x}_i\}_{i=1}^m$.

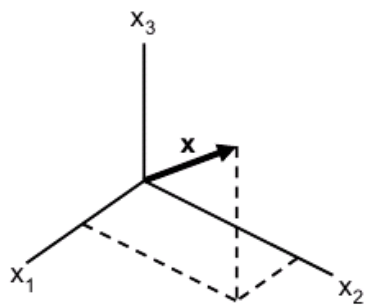
Supervised Learning

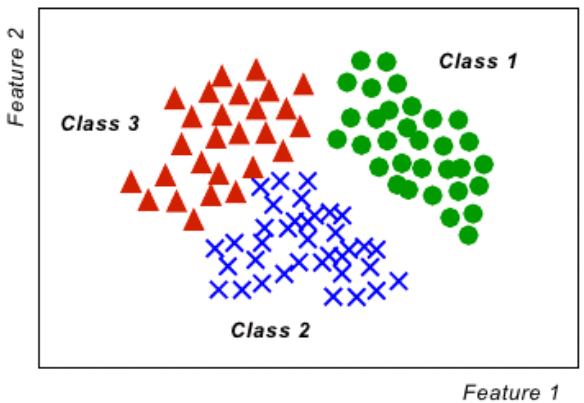
- An observation is a couple $(\mathbf{x}, y)$ where $\mathbf{x}$ is already associated with a value to predict $y \in \mathcal{Y}$.
- The space of values to predict $\mathcal{Y}$ is:
  - $\mathcal{Y} = \{-1, 1\}$ for binary classification or $\mathcal{Y} = \{1, \ldots, K\}$ for multi-class classification (K classes).
  - $\mathcal{Y} = \mathbb{R}$ for regression.
- A dataset is defined by m observations $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$ The values to predict (labels) can be concatenated into a vector $\mathbf{y} \in \mathcal{Y}^m$.

Feature

- A variable, also named a feature, is a distinctive trait or characteristic of an object. It can be symbolic (e.g., a color) or numeric (e.g., size).
- **Definition**
  - A combination of variables is represented using a vector $\mathbf{x}$ of dimension d.
  - The d-dimensional space is called the feature space (*e.g.*, $\mathbb{R}^d$).
  - Objects are represented as points in the feature space. This representation is called a scatterplot.





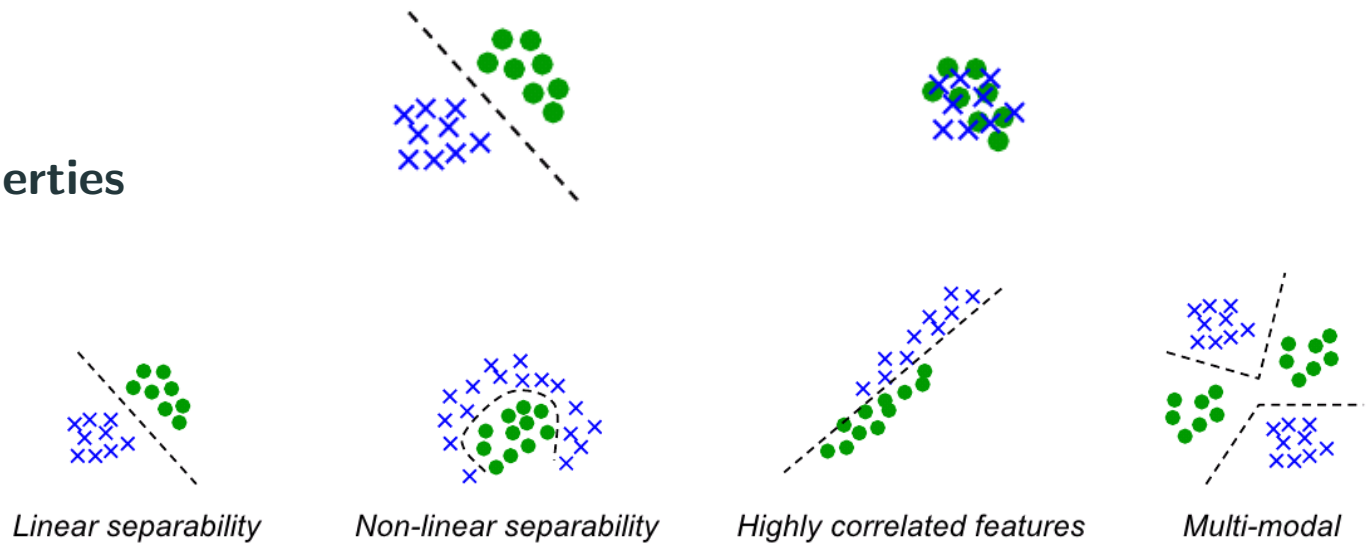


What Makes a "Good" Variable?

The quality of a variable depends on the learning problem.

- **Classification**: examples from the same class should have similar features, while examples from different classes should have different features.
- **Regression**: The feature should help in better predicting the value (it should be correlated with the values to predict).

Other Properties



Model

A model

$$f_\Theta : \mathcal{X} \to \mathcal{Y}$$

is a function parameterized by Θ that maps feature vectors to the predicted output.

- f_Θ is meant to capture intrinsic patterns of the data, the underlying assumption being that these hold true for all data drawn from $\mathbb{P}_{XY}$
- in ML, we constrain f to a certain type of functions (when choosing the algorithm).

Hypothesis space

- the task of finding a "good" model among all available models is impossible to solve
- we choose a category of model *a priori* to narrow the search space
- the set of functions defining a specific model class is called a hypothesis space $\mathcal{H} = \{f_\Theta$, where f_Θ belongs to a certain famiy of model$\}$
- finding an optimal model $\Rightarrow$ finding optimal parameters

Model

A model

$$f_\Theta : \mathcal{X} \to \mathcal{Y}$$

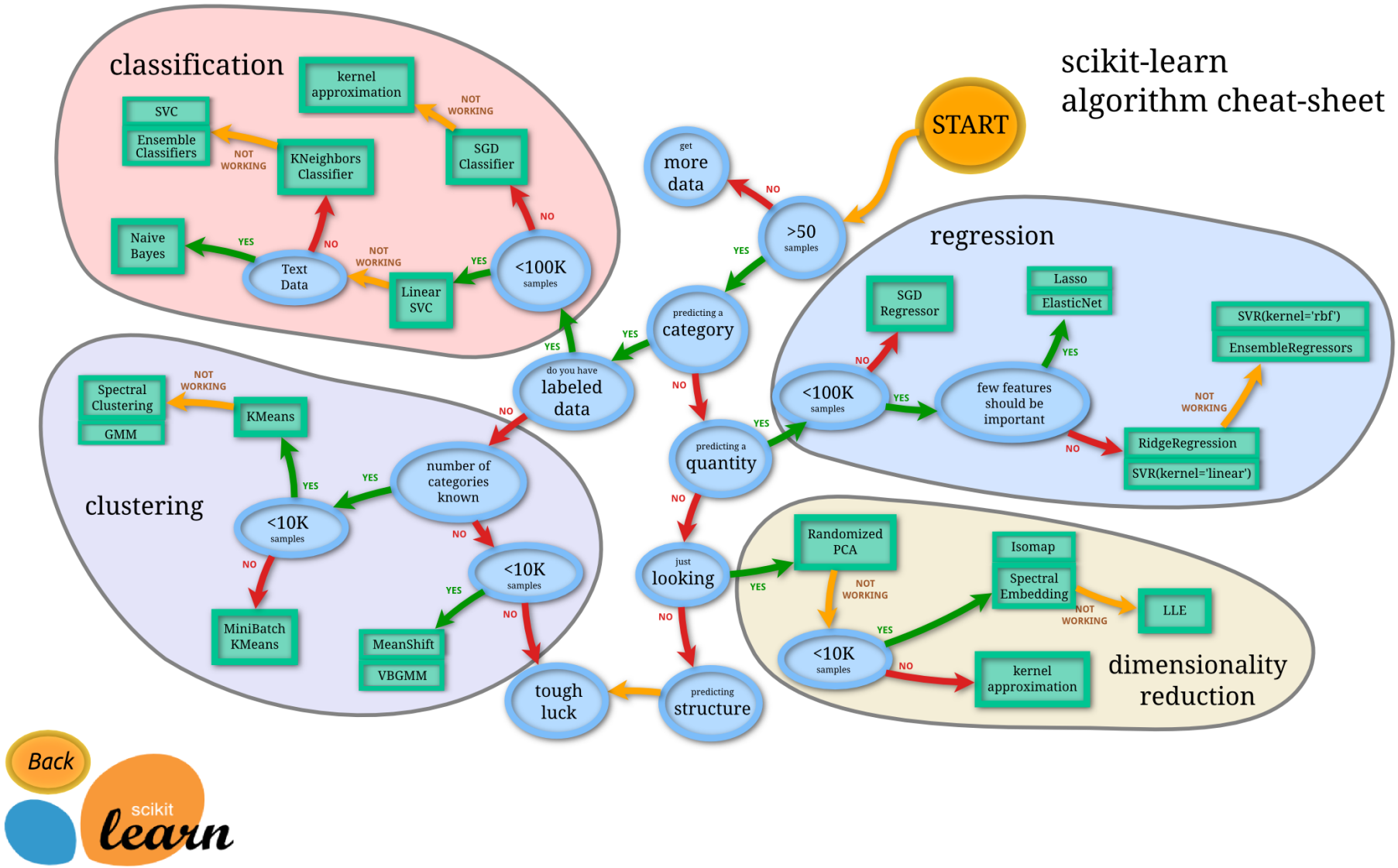
is a function parameterized by Θ that maps feature vectors to the predicted output.

- f_Θ is meant to capture intrinsic patterns of the data, the underlying assumption being that these hold true for all data drawn from $\mathbb{P}_{XY}$
- in ML, we constrain f to a certain type of functions (when choosing the algorithm).

Hypothesis space

- the task of finding a "good" model among all available models is impossible to solve
- we choose a category of model *a priori* to narrow the search space
- the set of functions defining a specific model class is called a hypothesis space $\mathcal{H} = \{f_\Theta$, where f_Θ belongs to a certain famiy of model$\}$
- finding an optimal model $\Rightarrow$ finding optimal parameters

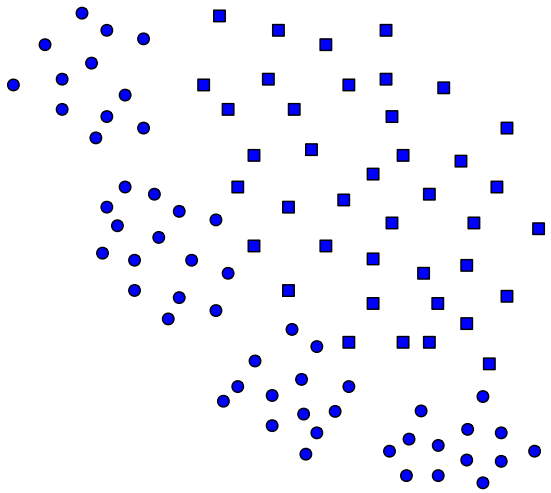
Finding Your Algorithm



Back

Content

Data Description/Exploration



Consider a training set $\{\mathbf{x}_i\}_{i=1}^m$ composed of examples of dimension d.

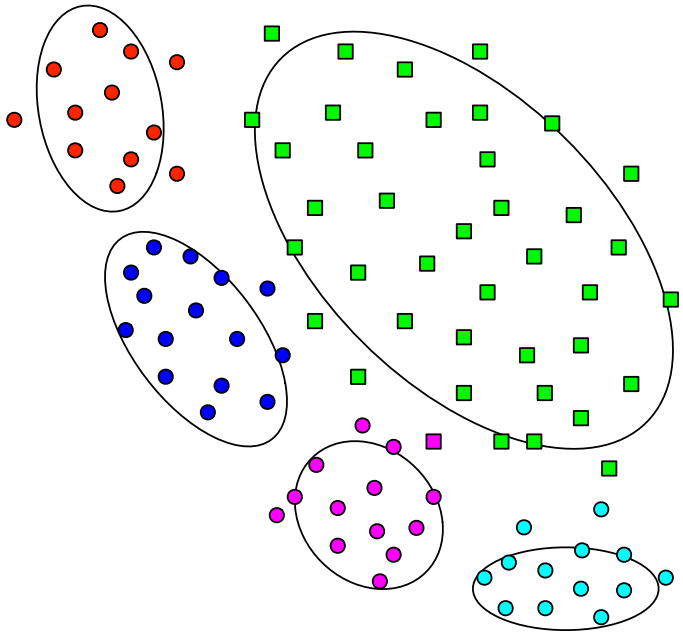
Objectives

- **Clustering** $\{\mathbf{x}_i\}_{i=1}^m \Rightarrow \{\hat{y}_i\}_{i=1}^m$ where $\hat{y}$ represents membership in a group.
- **Probability Density Estimation** $\{\mathbf{x}_i\}_{i=1}^m \Rightarrow p(\mathbf{x})$.
- **Dimensionality Reduction** $\{\mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^m \Rightarrow \{\tilde{\mathbf{x}}_i \in \mathbb{R}^{d'}\}_{i=1}^m$ with $d' \ll d$.

Clustering

Objective

- Organize the training examples into groups.
- $\{\mathbf{x}_i\}_{i=1}^n \Rightarrow \{\hat{y}_i\}_{i=1}^n$ where $\hat{y} \in \mathcal{Y}$ represents a group (cluster) $\{1, \ldots, m\}$
- Parameters:
  - m number of groups
  - Similarity measure (characterizing similarities between observations)



Methods

- k-means clustering.
- Gaussian mixture models.
- Hierarchical clustering.

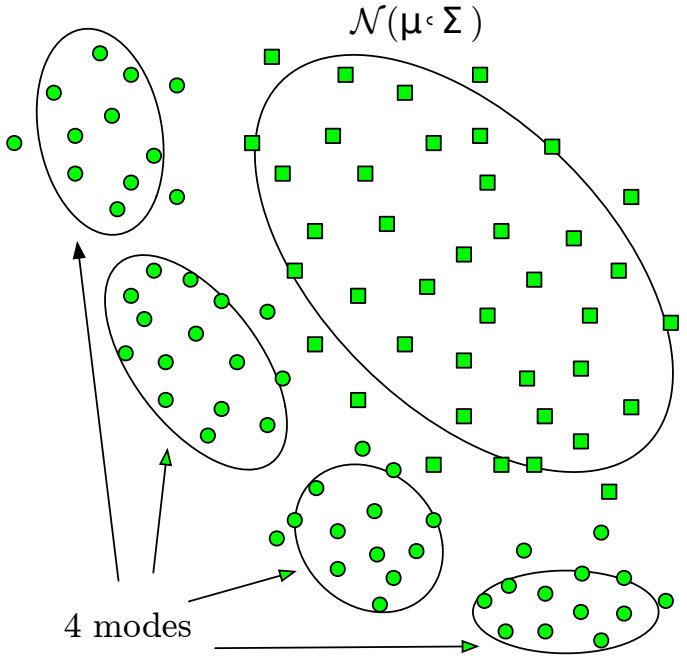
Examples

- Taxonomy of animals.
- Gene clustering.
- Social networks.

Probability Density Estimation

Objective

- Estimate the probability distribution of the data.
- $\{\mathbf{x}_i\}_{i=1}^m \Rightarrow p(\mathbf{x})$ where $p(\mathbf{x})$ is a probability density ($\int p(\mathbf{x})d\mathbf{x} = 1$)
- Model can be generative.
- Parameters:
  - Type of distribution (Gaussian, ...)
  - Distribution parameters (μ, Σ)



Methods

- parzen windows
- histogram
- Gaussian Mixture Models

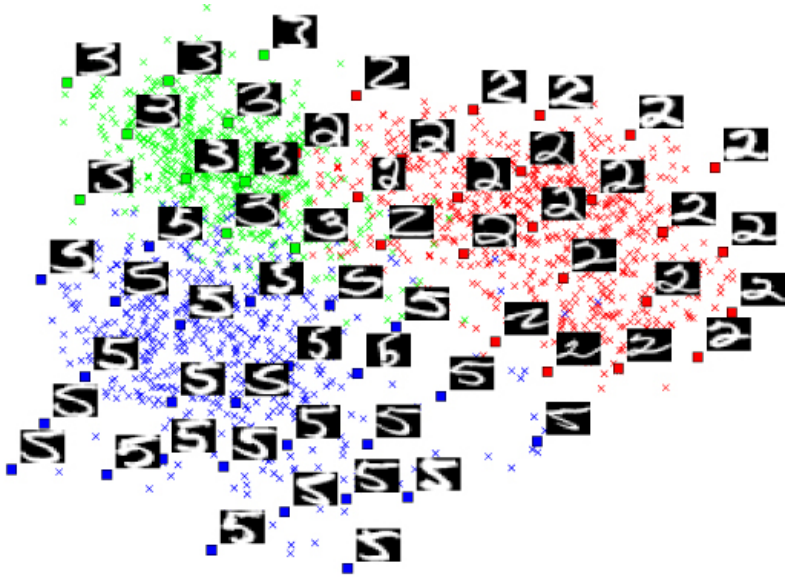
Examples

- noise estimation
- data generation (faces, ...)
- novelty detection

Dimensionality Reduction

Objective

- Project the data into a low-dimensional space.
- $\{\mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^n \Rightarrow \{\tilde{\mathbf{x}}_i \in \mathbb{R}^{d'}\}_{i=1}^m$ with $d' \ll d$ (often $d' = 2$).
- Parameters:
  - Type of projection.
  - Similarity measure.



Methods

- Variable selection.
- Principal Component Analysis (PCA).
- Non-linear reduction.

Examples

- Data preprocessing.
- Vector visualization.
- Data interpretation.
- Recommendation systems.

Content

Prediction

Consider a training set $\{\mathbf{x}_i, y_i\}_{i=1}^n$ composed of n observations $\mathbf{x}_i \in \mathbb{R}^d$ of dimension d and target values $y_i \in \mathcal{Y}$.

Objective

- We aim to learn from the training data a prediction function $f(\cdot) : \mathbb{R}^d \to \mathcal{Y}$.
- Types of predictions:
  - **Classification**
    $f(\cdot)$ predicts a class / category (discrete output), either in binary classification $\mathcal{Y} = \{-1, 1\}$ or multiclass $\mathcal{Y} = \{1, \ldots, m\}$.
  - **Regression**
    $f(\cdot)$ predicts a real value ($\mathcal{Y} = \mathbb{R}$).

Linear Function

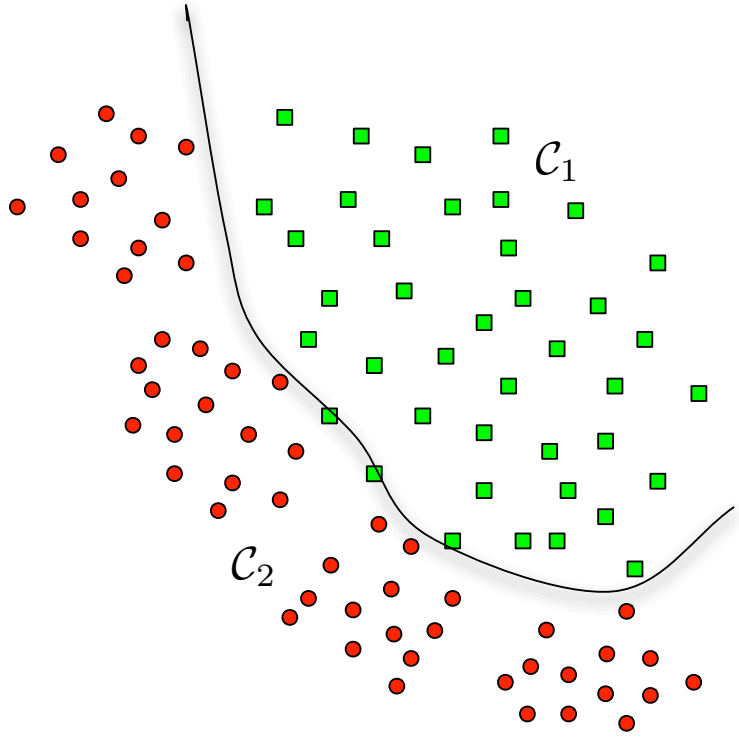
$$f(\mathbf{x}) = \sum_{j=1}^{d} w_j x_j + b = \mathbf{w}^\top \mathbf{x} + b$$

parameterized by $\mathbf{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$

Binary Classification

Objective

- Learn a function that predicts either class -1 or 1.
- $\{\mathbf{x}_i, y_i\}_{i=1}^m \Rightarrow f(\mathbf{x})$.
- Prediction: sign of $f(\cdot)$
- $f(\mathbf{x}) = 0$: decision boundary.
- Parameters:
  - Types of functions
  - Performance measurement



Methods

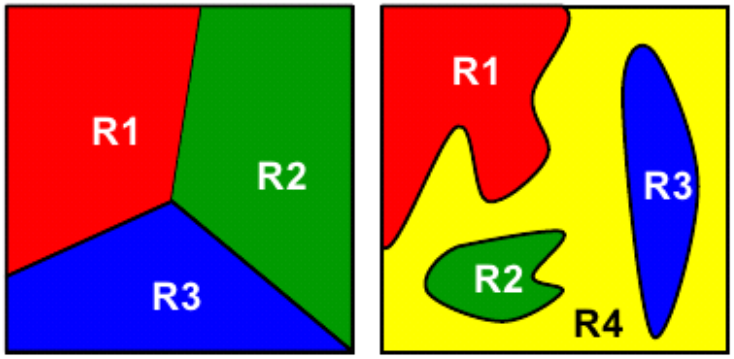
- Bayesian Methods
- Linear Discriminant Classifier
- Support Vector Machine (SVM)
- Decision Trees

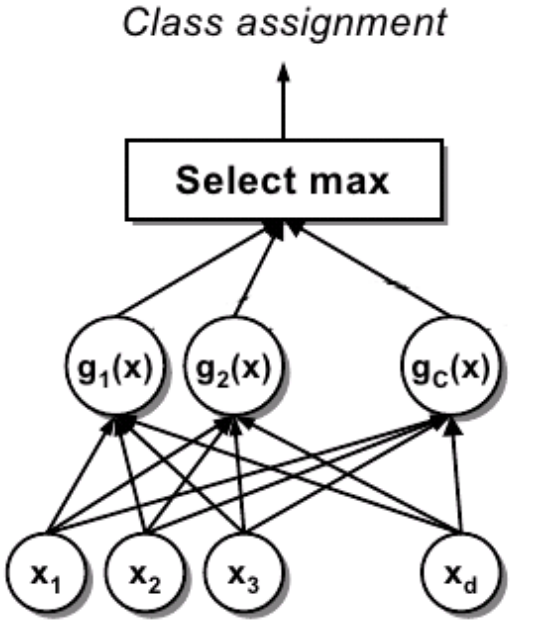
Examples

- Character Recognition.
- Diagnostic Assistance.
- Parts Inspection.
- Weather (Rain) Prediction.

Multiclass Classification

- The role of a classifier is to partition the space of variables into multiple regions to which classes are assigned.
  - The boundaries are called decision boundaries.
  - Classifying a vector of variables $\mathbf{x}$ involves determining which region it belongs to and assigning it the label of that region.
- The classifier can be represented by a set of discriminant functions: the classifier assigns $\mathbf{x}$ to class j if $g_j(x) > g_i(x)$ for all $i \neq j$.

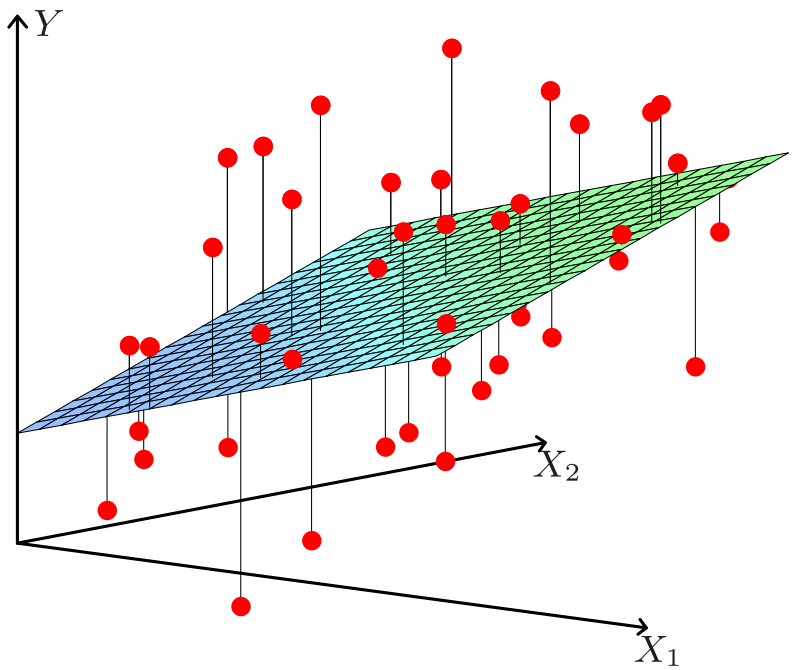




Regression

Objective

- Learn a function that predicts a real value.
- $\{\mathbf{x}_i, y_i\}_{i=1}^{n} \Rightarrow f(\mathbf{x})$.
- Parameters:
  - Type of function.
  - Performance measurement.
  - Prediction error.



Methods

- Least Squares.
- Ridge Regression.
- Kernel Regression.

Examples

- Motion Prediction.
- Cholesterol Level Prediction.
- Weather (Temperature) Prediction.