Introduction to Machine Learning

Concepts

Joaquin Vanschoren, Eindhoven Univeristy of Technology

Artificial Intelligence

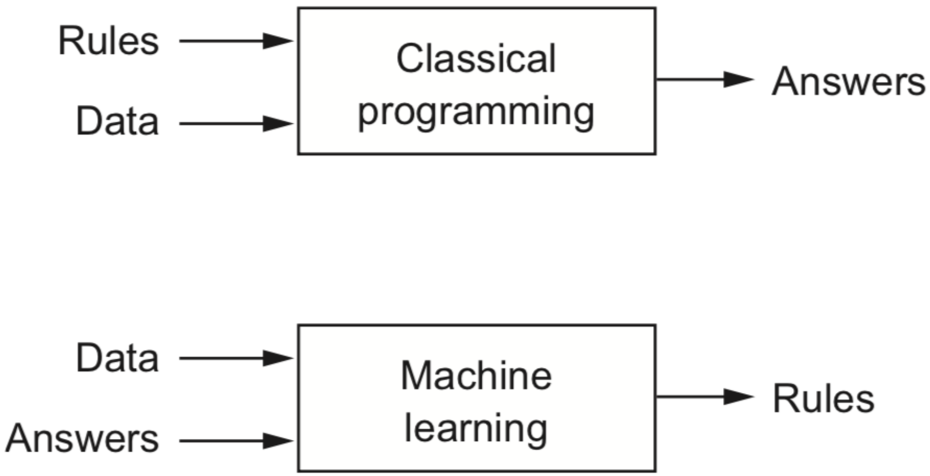
1950s: Can computers be made to 'think'?

- automate intellectual tasks normally performed by humans
- encompasses learning, but also many other tasks (e.g. logic, planning,...)
- *symbolic AI*: programmed rules/algorithms for manipulating knowledge
  - Great for well-defined problems: chess, expert systems,...
  - Pervasively used today (e.g. chip design)
  - Hard for complex, fuzzy problems (e.g. images, text)

Machine Learning

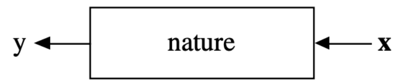
Are computers capable of learning and originality? Alan Turing: Yes!

- Learn to perform a task T given experience E, always improving according to some metric M
- New programming paradigm
  - System is *trained* rather than explictly programmed
  - Finds rules or functions (models) to act/predict

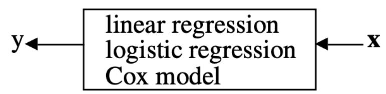


Machine learning vs Statistics

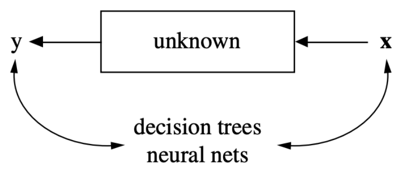
- Both aim to make predictions of natural phenomena:



- Statistics:
  - Help humans understand the world
  - Parametric: assume that data is generated according to parametric model



- Machine learning:
  - Automate a task entirely (replace the human)
  - Non-parametric: assume that data generation process is unknown
  - Engineering-oriented, less (too little?) mathematical theory



See Breiman (2001): Statical modelling: The two cultures

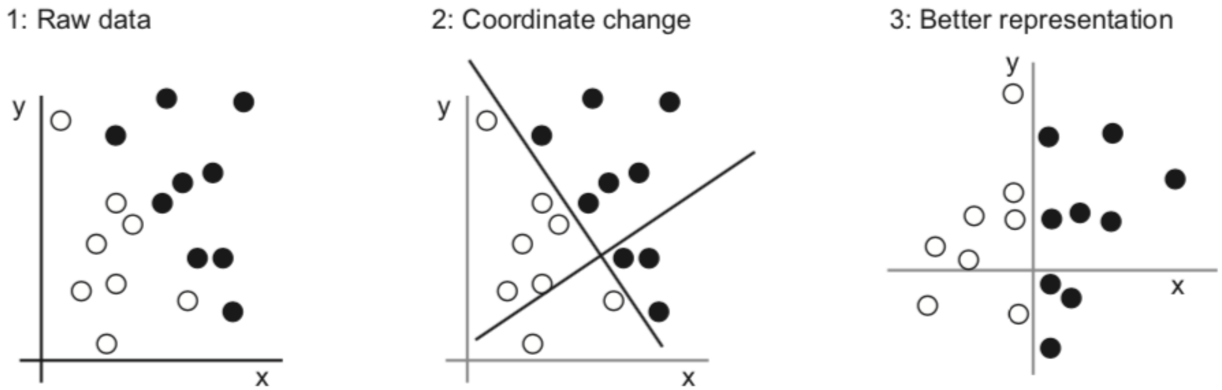
How to represent learning?

All machine learning algorithms consist of 3 components:

- Representation: A model must be represented in a formal language that the computer can handle
  - Defines the 'concepts' it can learn, the _hypothesis space-
  - E.g. a decision tree, neural network, set of annotated data points
- Evaluation: An *internal* way to choose one hypothesis over the other
  - Objective function, scoring/loss function
  - E.g. Difference between correct output and predictions
- Optimization: An *efficient* way to search the hypothesis space
  - Start from simple hypothesis, extend (relax) if it doesn't fit the data
  - Defines speed of learning, number of optima,...
  - E.g. Gradient descent

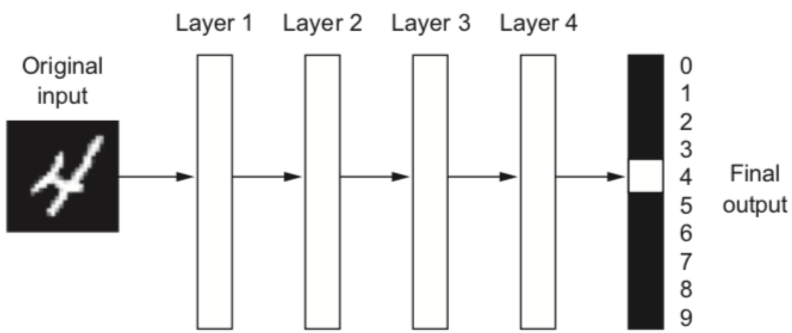
How to represent the problem?

- We need 3 inputs:
  - Input data, e.g. measurements, images, text
  - Expected output: e.g. correct labels produced by humans
  - Performance measure: feedback signal, are we learning the right thing?
- Algorithm needs to correctly transform the inputs to the right outputs
- Often includes transforming the data to a more useful representation (or encoding)
  - Can be done end-to-end (e.g. deep learning) or by first 'preprocessing' the data



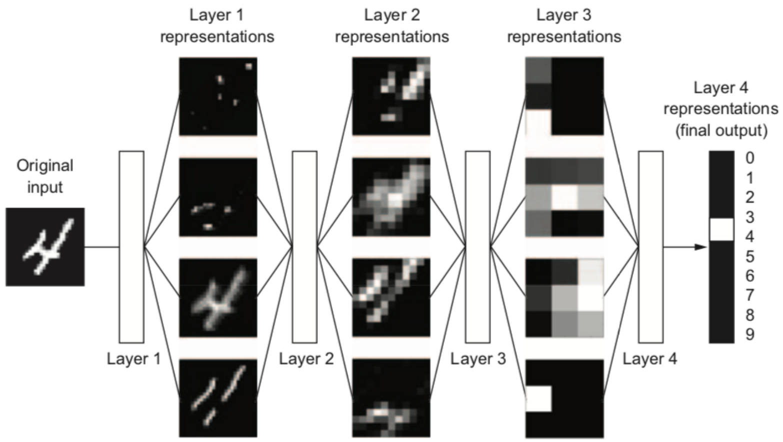
Deep Learning

- Most machine learning techniques require humans to build a good representation of the data
  - Sometimes data is naturally structured (e.g. medical tests)
  - Sometimes not (e.g. images) -> extract features
- Deep learning: learn your own representation of the data
  - Through multiple layers of representation (e.g. layers of neurons)
  - Each layer transforms the data a bit, based on what reduces the error

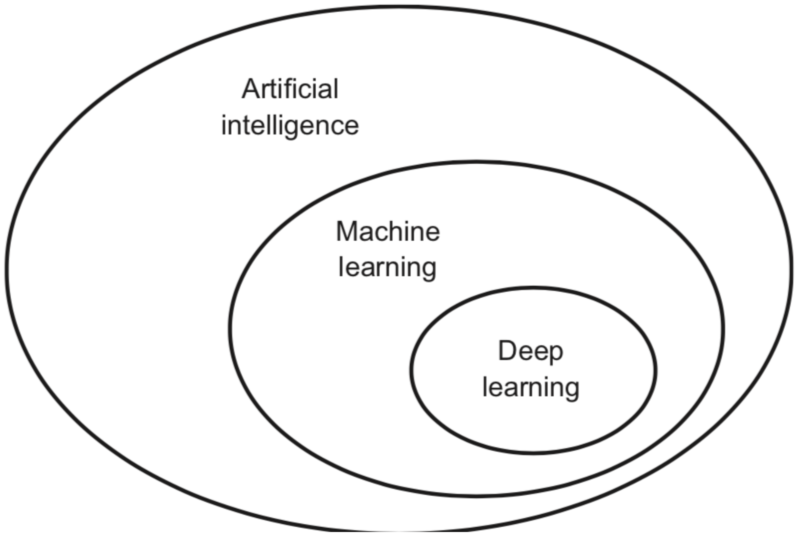


Example: digit classification

- Input pixels go in, each layer transforms them to an increasingly informative representation for the given task
- Often less intuitive for humans



Overview



Success stories:

- Search engines (e.g. Google)
- Recommender systems (e.g. Netflix)
- Automatic translation (e.g. Google Translate)
- Speech understanding (e.g. Siri, Alexa)
- Game playing (e.g. AlphaGo)
- Self-driving cars
- Personalized medicine
- Progress in all sciences: Genetics, astronomy, chemistry, neurology, physics,..

Example: dating

Nr	Day of Week	Type of Date	Weather	TV Tonight	Date?
1	Weekday	Dinner	Warm	Bad	No
2	Weekend	Club	Warm	Bad	Yes
3	Weekend	Club	Warm	Bad	Yes
4	Weekend	Club	Cold	Good	No
Now	Weekend	Club	Cold	Bad	?

- Is there a combination of factor that works? Is one better than others?
- What can we assume about the future? Nothing?
- What if there is noise / errors?
- What if there are factor you don't know about?

Types of machine learning

We often distinguish 3 `types` of machine learning:

- **Supervised Learning**: learn a model from labeled *training data*, then make predictions
- **Unsupervised Learning**: explore the structure of the data to extract meaningful information
- **Reinforcement Learning**: develop an agent that improves its performance based on interactions with the environment

Note:

- Semi-supervised methods combine the first two.
- ML systems can combine many types in one system.

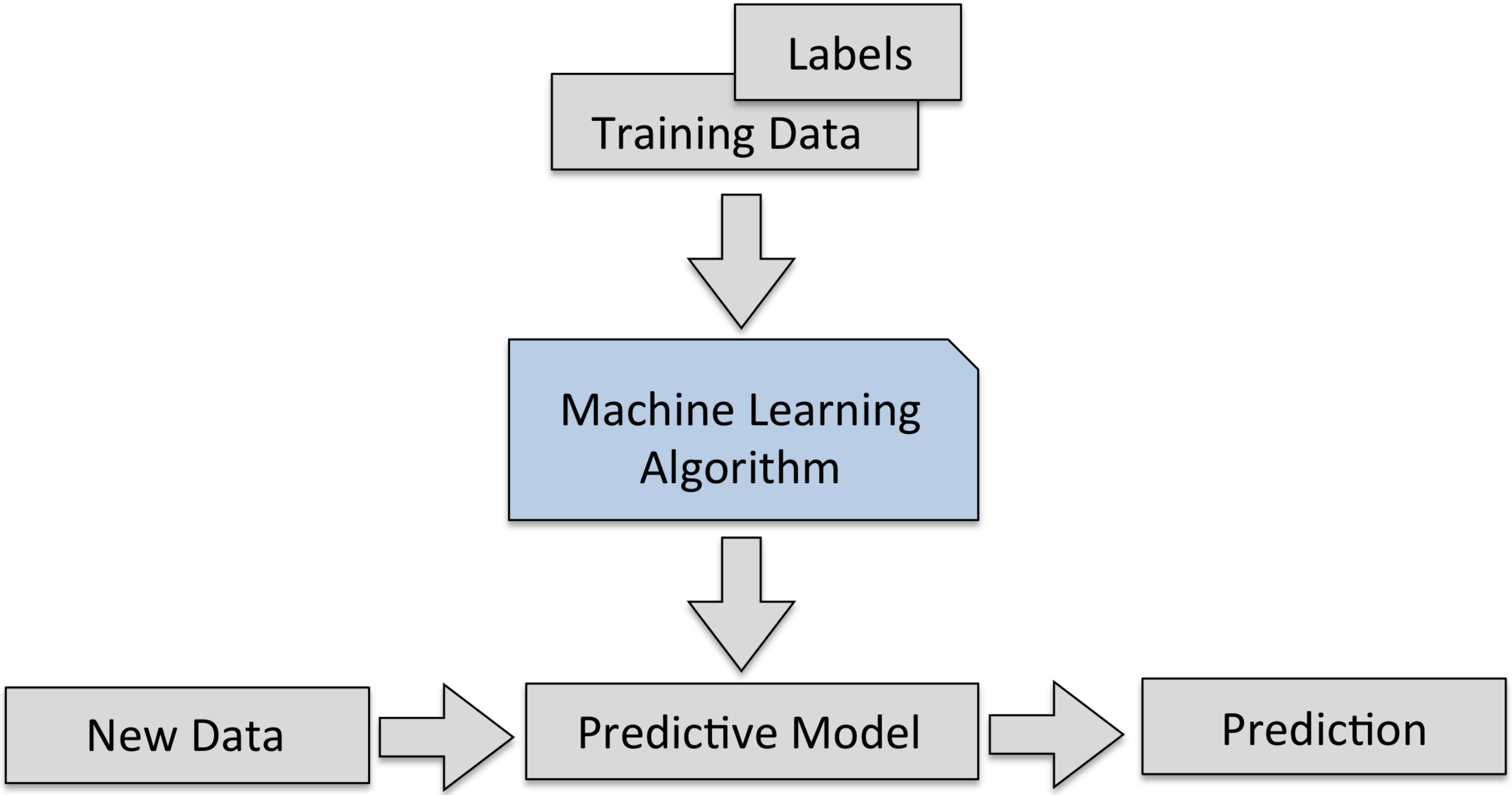
Supervised Machine Learning

- Learn a model from labeled training data, then make predictions
- Supervised: we know the correct/desired outcome (label)

2 subtypes:

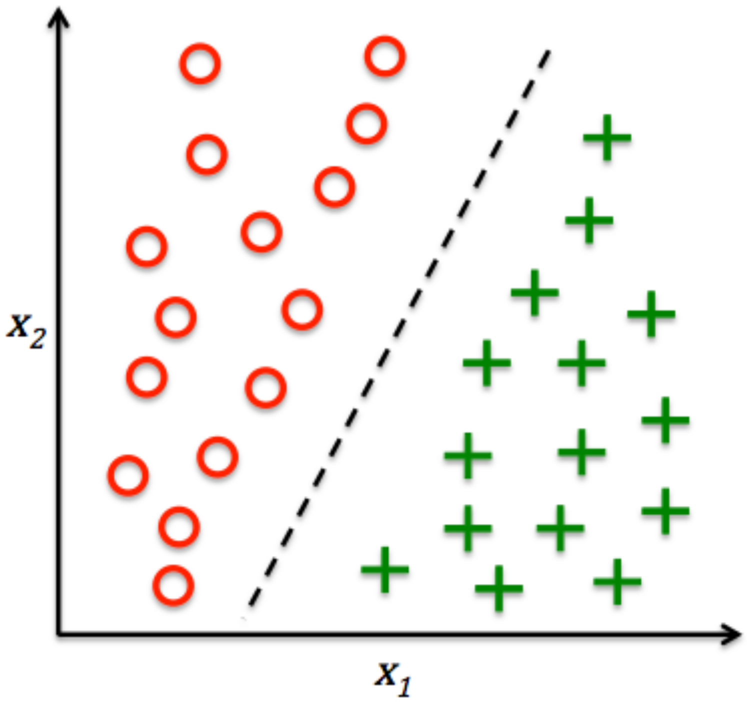
- Classification: predict a *class label* (category), e.g. spam/not spam
  - Many classifiers can also return a *confidence* per class
- Regression: predict a continuous value, e.g. temperature
  - Some algorithms can return a *confidence interval*

Most supervised algorithms that we will see can do both.


Labels
Training Data
Machine Learning Algorithm
New Data
Predictive Model
Prediction

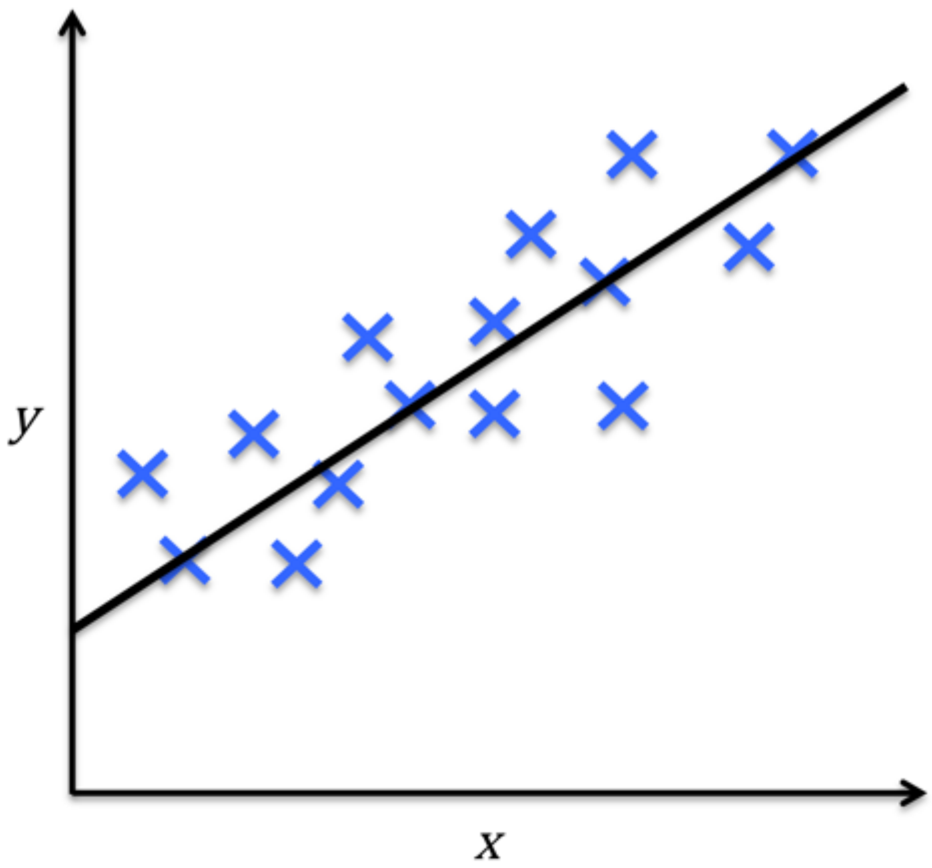
Classification

- Class labels are discrete, unordered
- Can be *binary* (2 classes) or *multi-class* (e.g. letter recognition)
- Dataset can have any number of predictive variables (predictors)
  - Also known as the dimensionality of the dataset
- The predictions of the model yield a *decision boundary* separating the classes



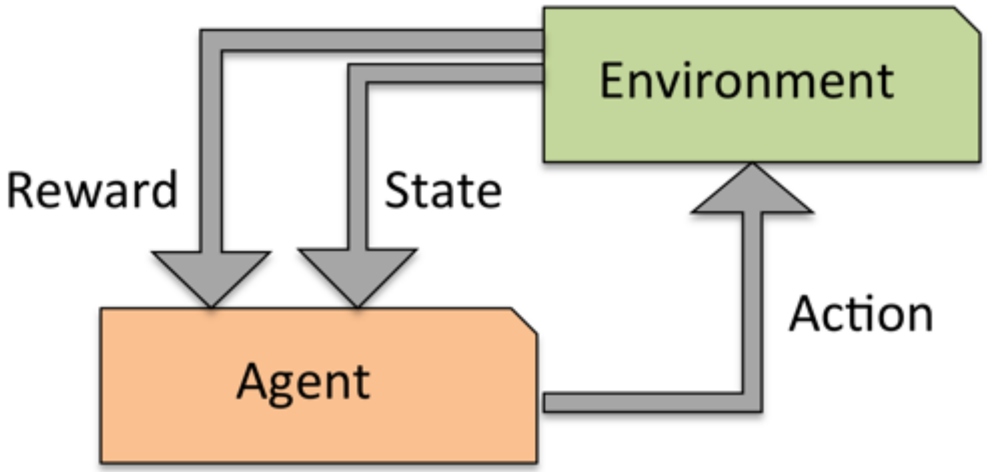
Regression

- Target variable is numeric
- Find the relationship between predictors and the target.
  - E.g. relationship between hours studied and final grade
- Example: Linear regression (fits a straight line)



Reinforcement learning

- Develop an agent that improves its performance based on interactions with the environment
  - Example: games like Chess, Go,...
- *Reward function* defines how well a (series of) actions works
- Learn a series of actions that maximizes reward through exploration

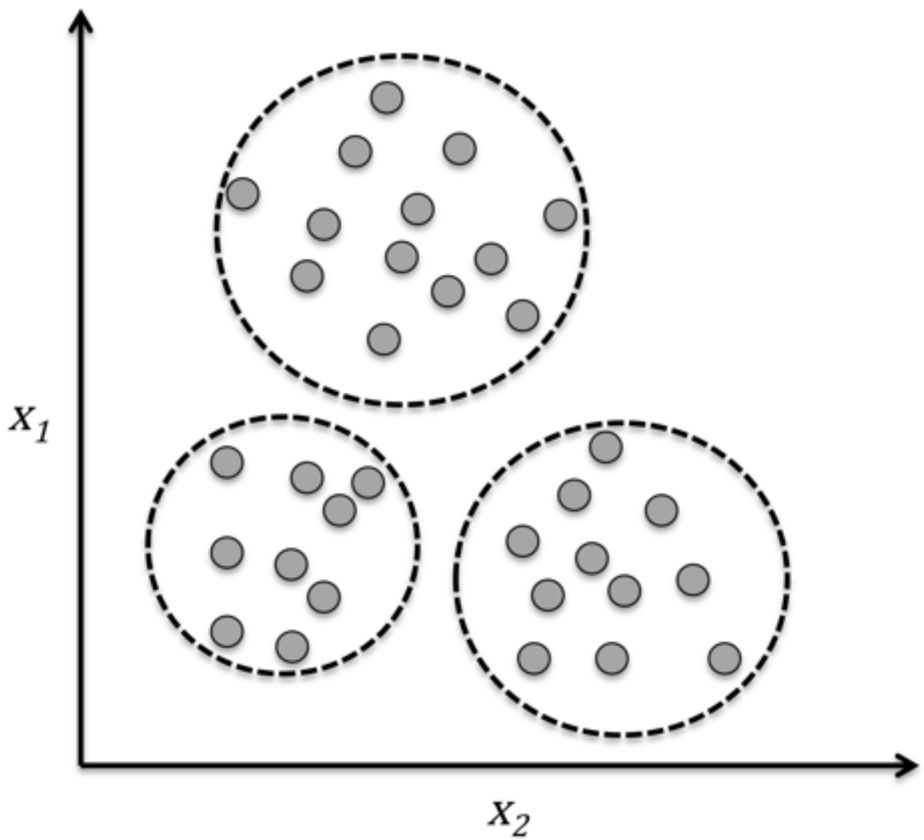


Unsupervised Machine Learning

- Unlabeled data, or data with unknown structure
- Explore the structure of the data to extract information
- Many types, we'll just discuss two.

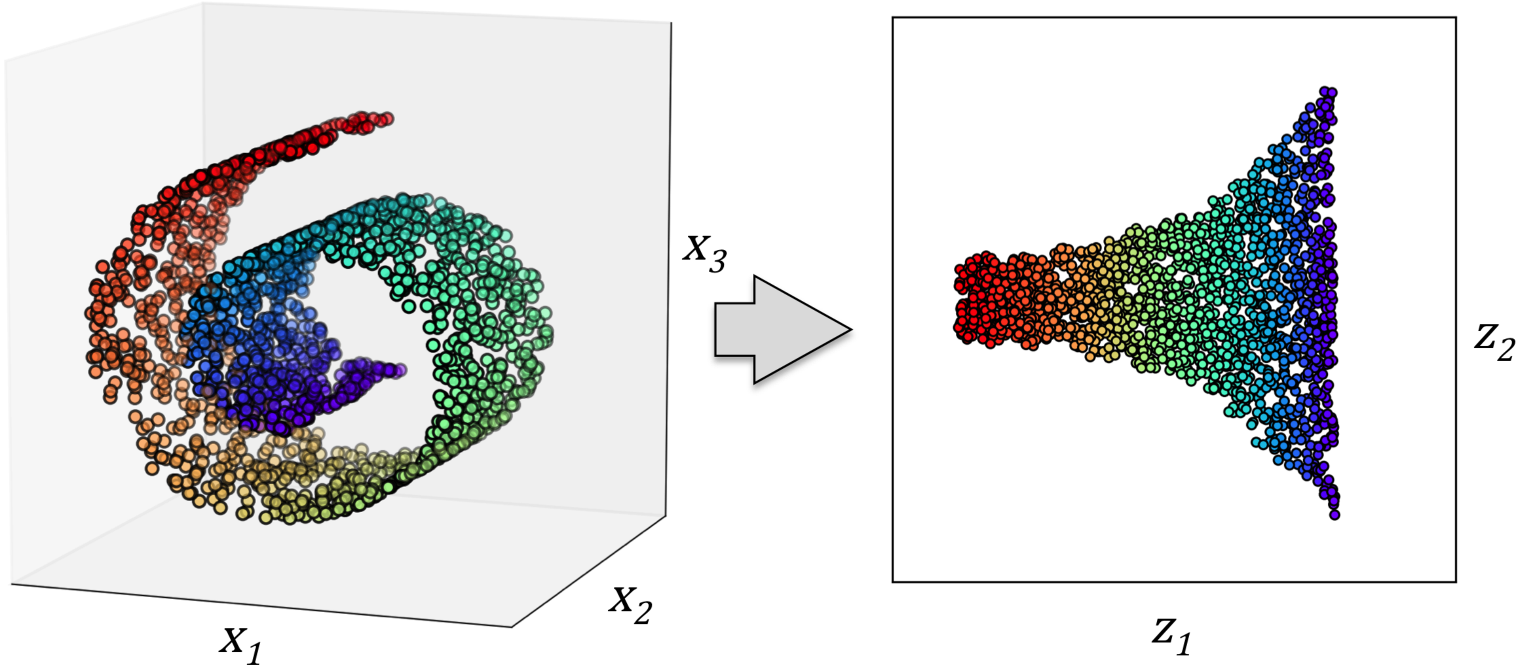
Clustering

- Organize information into meaningful subgroups (clusters)
- Objects in cluster share certain degree of similarity (and dissimilarity to other clusters)
- Example: distinguish different types of customers



Dimensionality reduction

- Data can be very high-dimensional and difficult to understand, learn from, store,...
- Dimensionality reduction can compress the data into fewer dimensions, while retaining most of the information
- Contrary to feature selection, the new features lose their (original) meaning
- Is often useful for visualization (e.g. compress to 2D)

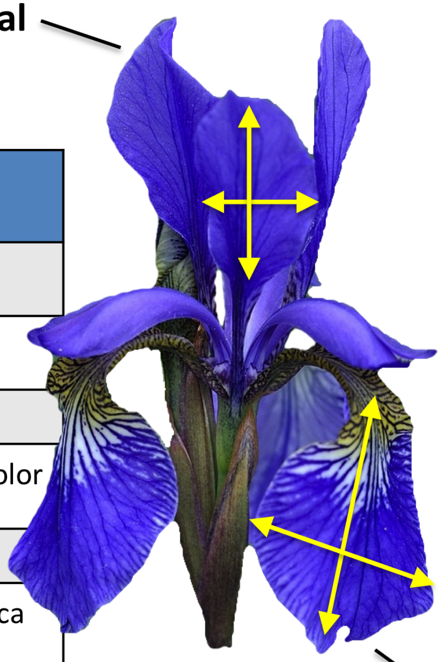


x_3
x_2
x_1
z_2
z_1

Basic Terminology (on Iris dataset)

Samples
(instances, observations)

	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
			...		
50	6.4	3.5	4.5	1.2	Versicolor
			...		
150	5.9	3.0	5.0	1.8	Virginica



Features
(attributes, measurements, dimensions)

Class labels
(targets)

Building machine learning systems

A typical machine learning system has multiple components:

- Preprocessing: Raw data is rarely ideal for learning
    - Feature scaling: bring values in same range
    - Encoding: make categorical features numeric
    - Discretization: make numeric features categorical
    - Feature selection: remove uninteresting/correlated features
    - Dimensionality reduction can also make data easier to learn

- Learning and model selection
  - Every algorithm has its own biases
  - No single algorithm is always best (No Free Lunch)
  - *Model selection* compares and selects the best models
    - Different algorithms
    - Every algorithm has different options (hyperparameters)
  - Split data in training and test sets

- Together they form a *workflow* of *pipeline*