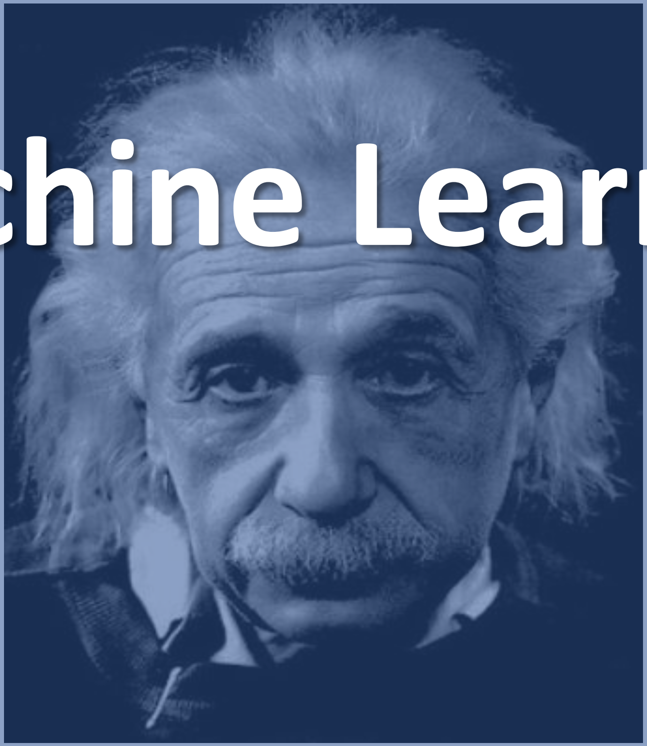
Computers are incredibly fast, accurate, and stupid; humans are incredibly slow, inaccurate and brilliant; together they are powerful beyond imagination.

Machine Learning



Albert Einstein

Course Information

- **Class homepage:** http://www.ccs.neu.edu/home/yzsun/classes/2013Spring_CS6220/index.htm
  - Slides
  - Announcement
  - Assignments

- **Prerequisites**

  The course assumes some basic knowledge of
  - Probabilistic and Statistics
    - Joint and marginal probability distributions
    - Normal (Gaussian) distribution
    - Expectation and variance
    - Statistical correlation and statistical independence
  - Linear algebra,
    - Matrices, vectors, and their multiplication
    - Matrix inverse and transformation
    - Eigen value decomposition

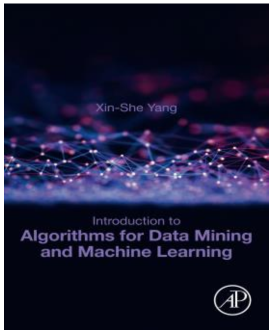
Course Objectives

- Introduce central approaches of machine learning
- Point out relations to human learning
- Define a class of problems that encompasses interesting forms of learning
- Explore algorithms that solve such problems
- Provide understanding of the fundamental structure of learning problems and processes

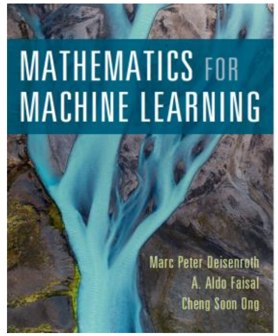
Grading

- Homework: 25%
- Course project: 25%
- Class participation: 10%
- Exam: 40%

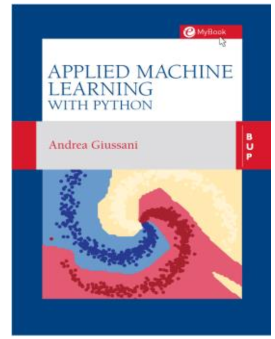
Source Materials



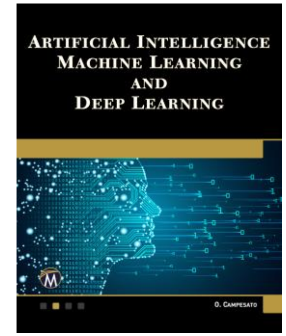
Introduction to Algorithms for Data Mining and Machine Learning by Xin-She Yang (2019)



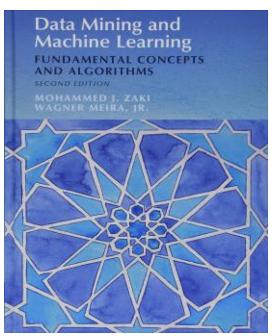
MATHEMATICS FOR MACHINE LEARNING 2021 by Cheng Soon Ong



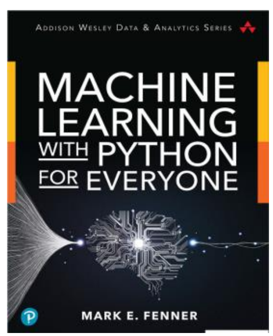
Applied Machine Learning with Python by Andrea Giussani (2020)



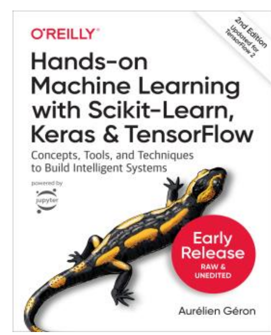
Artificial Intelligence Machine Learning and Deep Learning by Oswald Campesato (2020)



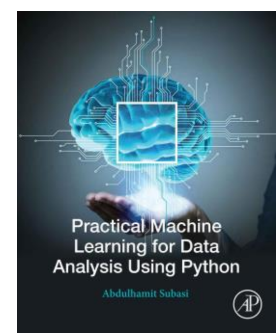
DATA MINING AND MACHINE LEARNING Fundamental Concepts and Algorithms by MOHAMMED J. ZAKI 2020



Machine Learning With Python For Everyone by Mark E. Fenner (2020)



Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow 2019 by Aurélien Géron



Practical Machine Learning for Data Analysis Using Python by Abdulhamit Subasi (2020)

Outline of the Course

1. Introduction
2. Dimensionality reduction
3. Neural Network
4. Regression
5. Clustering algorithms
6. Bayesian Learning
7. DT3 and random forest
8. Support Vector Machines
9. Model evaluation
10. Ensemble learning

What is machine learning?

According to Wikipedia

- "**Learning is** acquiring new knowledge, behaviors, skills, values, preferences or understanding, and may involve synthesizing different types of information. The ability to learn is possessed by humans, animals and some machines. Progress over time tends to follow learning curves."
- "**Machine learning is** a scientific discipline that is concerned with the design and development of algorithms that allow computers to change behavior based on data, such as from sensor data or databases. A major focus of machine learning research is to automatically learn to recognize complex patterns and make intelligent decisions based on data. Hence, machine learning is closely related to fields such as statistics, probability theory, data mining, pattern recognition, artificial intelligence, adaptive control, and theoretical computer science."

What is Machine Learning?

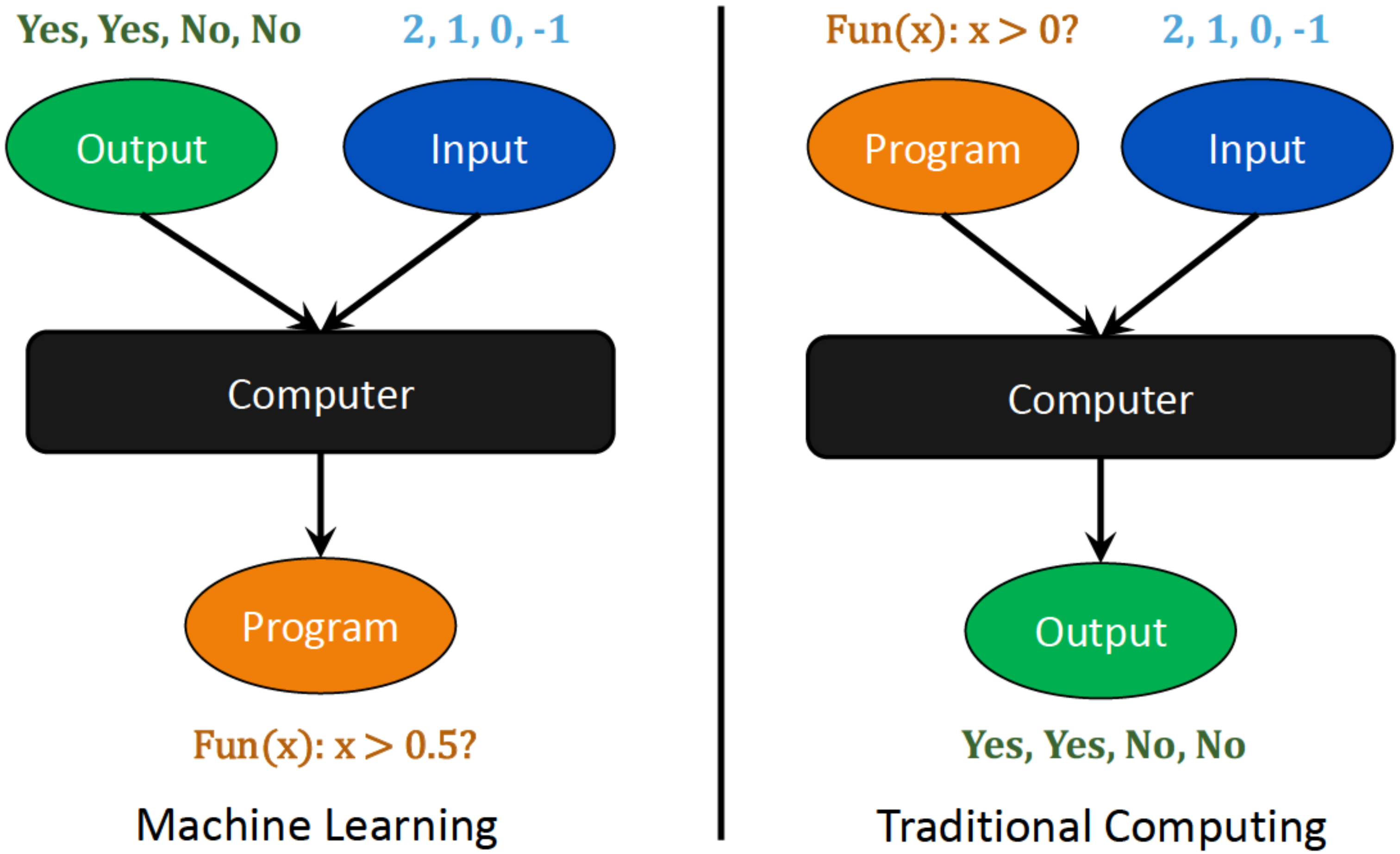
Adapt to / learn from data

- To optimize a performance function

Can be used to:

- Extract knowledge from data
- Learn tasks that are difficult to formalise
- Create software that improves over time

Machine learning vs Traditional computing



A Bit of History

1940s

Advances in mathematical logic, information theory, concept of neural computation

1943: McCulloch & Pitts Neuron

1948: Shannon: Information Theory

1949: Hebbian Learning

cells that fire together, wire together

1950s

Early computers. Dartmouth conference coins the phrase “artificial intelligence” and Lisp is proposed as the AI programming language

Alan Turing

The Turing Test, A machine is intelligent if its answers are indistinguishable from a human’s.

1952 Checkers Program

Created a Checkers-playing program that got better overtime. Also introduced the term “Machine Learning”.

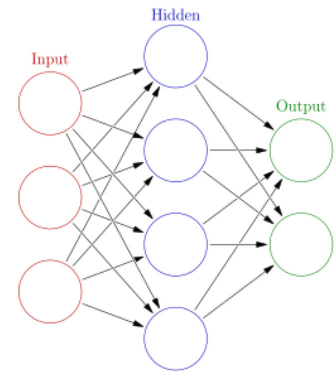
Arthur Samuel

1956: Dartmouth Conference

Frank Rosenblatt

1957 Perceptron

Predecessor of deep networks. Separating two classes of objects using a linear threshold classifier. Provable learning and convergence guarantees followed later by Albert Novikoff.



1958: Friedberg: Learn Assembly Code
1959: Samuel: Learning Checkers

1960s: Lots of hope for AI to solve everything!
A.I. funding increased (mainly military). Famous quote: "Within a generation ... the problem of creating 'artificial intelligence' will substantially be solved."

Early symbolic reasoning approaches.
Logic Theorist, GPS, Perceptrons

Ax. 1. $P(\varphi) \wedge \Box\, \forall x[\varphi(x) \to \psi(x)] \to P(\psi)$
Ax. 2. $P(\neg\varphi) \leftrightarrow \neg P(\varphi)$
Th. 1. $P(\varphi) \to \Diamond\, \exists x\, [\varphi(x)]$
Df. 1. $G(x) \iff \forall\varphi[P(\varphi) \to \varphi(x)]$
Ax. 3. $P(G)$
Th. 2. $\Diamond\, \exists x\, G(x)$
Df. 2. $\varphi \text{ ess } x \iff \varphi(x) \wedge \forall\psi\{\psi(x) \to \Box\, \forall x[\varphi(x) \to \psi(x)]\}$
Ax. 4. $P(\varphi) \to \Box\, P(\varphi)$
Th. 3. $G(x) \to G \text{ ess } x$
Df. 3. $E(x) \iff \forall\varphi[\varphi \text{ ess } x \to \Box\, \exists x\, \varphi(x)]$
Ax. 5. $P(E)$
Th. 4. $\Box\, \exists x\, G(x)$

1969: Minsky & Papert "Perceptrons"

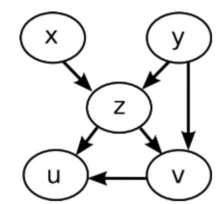
1970s
A.I. "winter" – Funding dries up as people realize this is a hard problem!
Limited computing power and dead-end frameworks lead to failures.

AI didn’t live up to the hype!

1966: Machine Translation failed.
1970: Minsky and Papert argued against Perceptron.
1971: Speech Understanding failed.
1973: Lighthill report torn apart AI.
1974: The UK and US stopped funding AI research.

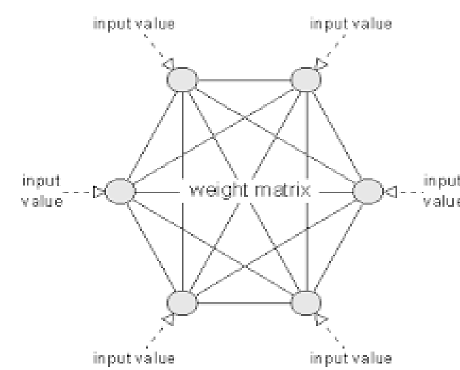
The AI Winter, 1974-1980

1980s
Rule based “expert systems” used in medical / legal professions.
Bio-inspired algorithms (Neural networks, Genetic Algorithms).
Again: A.I. promises the world – lots of commercial investment



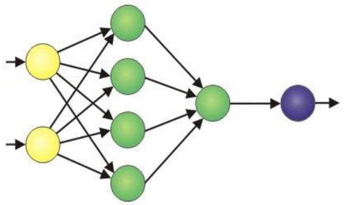
$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}.$$

Expert Systems (Mycin, Dendral, EMYCIN)
Knowledge Representation and reasoning:
    Frames, Eurisko, Cyc, NMR, fuzzy logic Speech
    Recognition (HEARSAY, HARPY, HWIM)
ML:
  1982: Hopfield Nets, Decision Trees, GA & GP.
  1986: Backpropagation, Explanation-Based Learning



1990s

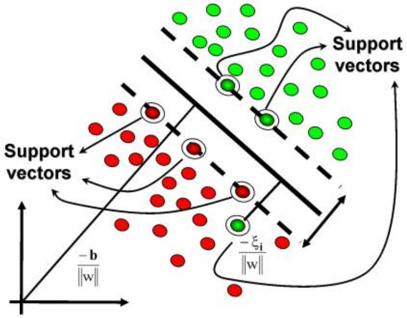
Some concrete successes begin to emerge. AI diverges into separate fields: Computer Vision, Automated Reasoning, Planning systems, Natural Language processing, **Machine Learning**…



…Machine Learning begins to overlap with statistics / probability theory.

1992: Koza & Genetic Programming

1995: Vapnik: Support Vector Machines



Rebirth as Machine Learning

Machine Learning: Originally, a bit of a name game to get funding. Fundamentally a different approach to intelligence:

Machine Learning	Artificial Intelligence
Data-driven	Knowledge-based
Bottom-up approach	Heavy use of logic
	Top-down approach

Foundations of ML, 1980s-present

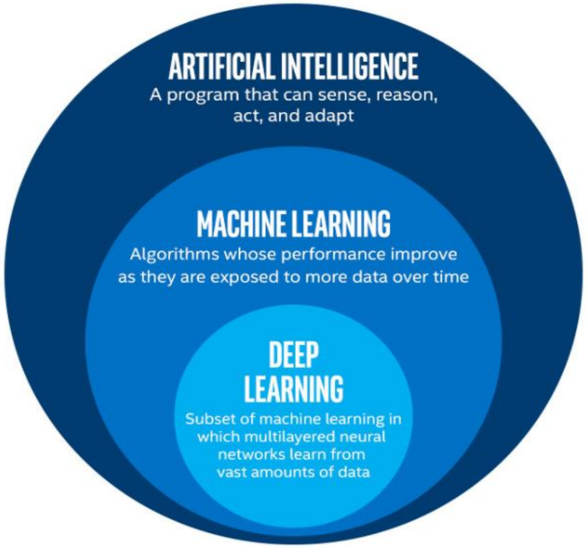
Formal notions of learnability from Data.

- When data-driven learning is possible?

  → Probably Approximately Correct Learning (PAC) by Valiant.

  → How much data is required?

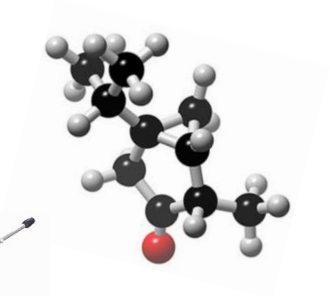
- What’s the difference between great and mediocre learners?
  → Improving the performance of a learning algorithm.
  → Boosting algorithm of Freund and Schapire.
- How to deal with difficult and noisy learning problems?
  → (Soft Margin) Support Vector Machines by Cortes and Vapnik
- What to do when the learning task evolves over time?
  → Online learning framework.

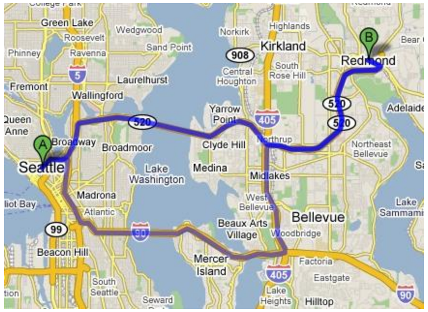


2000s: **Machine learning became profitable!**

First commercial-strength applications: Google, Amazon, computer games, route-finding, credit card fraud detection, spam filters, etc…

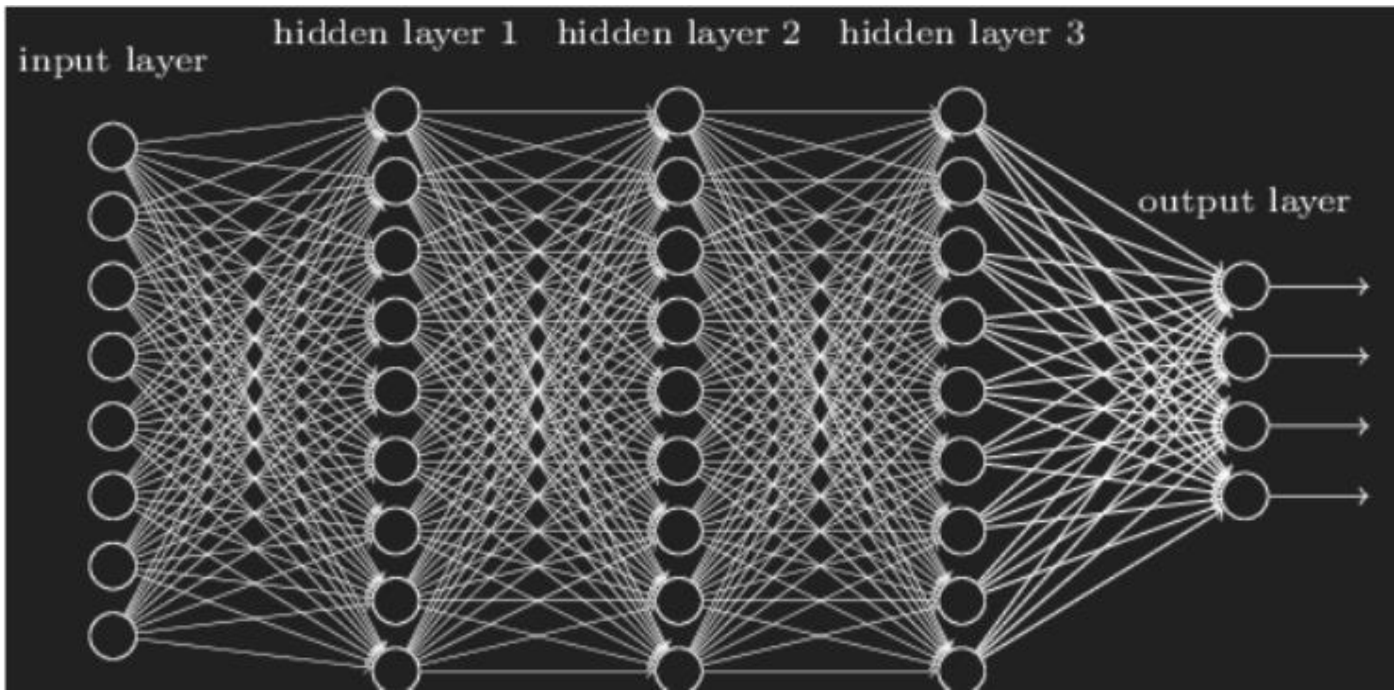
Tools adopted as standard by other fields e.g. biology





2010s: Return of Neural Networks

Neural networks return and excel at image recognition, speech recognition, … The 2018 Turing award was given to Yoshua Bengio, Geoff Hinton, and Yann LeCun.

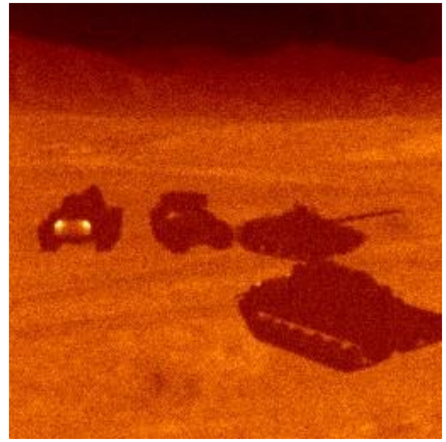


Machine Learning and AI Applications

- Autonomous vehicles have sensors and cameras that continuously scan the surrounding area and capture information.
- Machine learning algorithms employ deep learning to learn from the behaviour of human drivers. Many hours on the road are needed to train the self-driving brain how to act near pedestrians, when to slow down, to drive in the middle of a lane, and to stop at a red light.

Robotics vision

Target Recognition



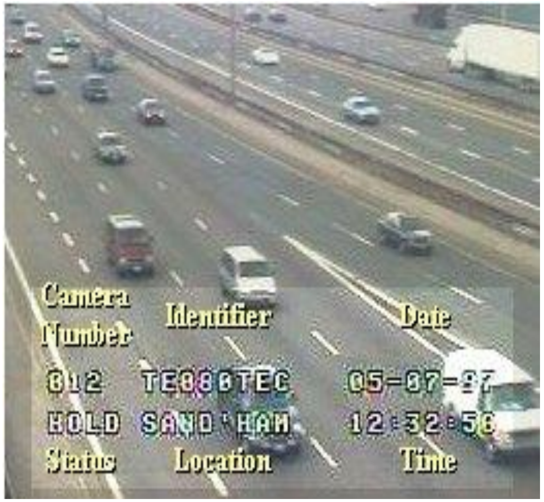
Netflix

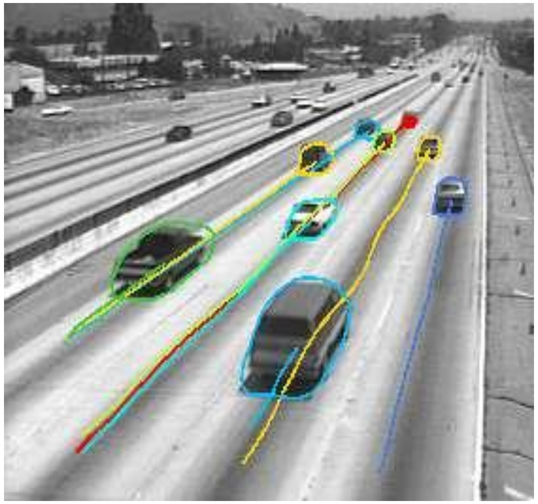
- Machine learning is integral to Netflix's video recommendation engine.

UBER

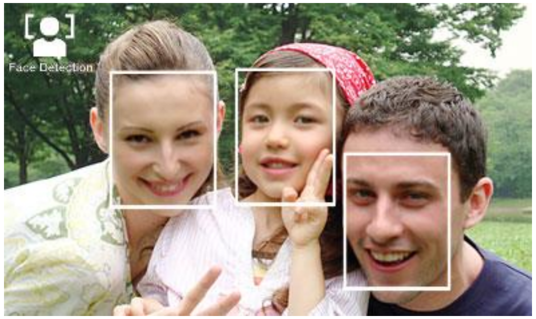
- The tech giant uses machine learning to determine arrival times, picks up and locations and Uber Eats food deliverables

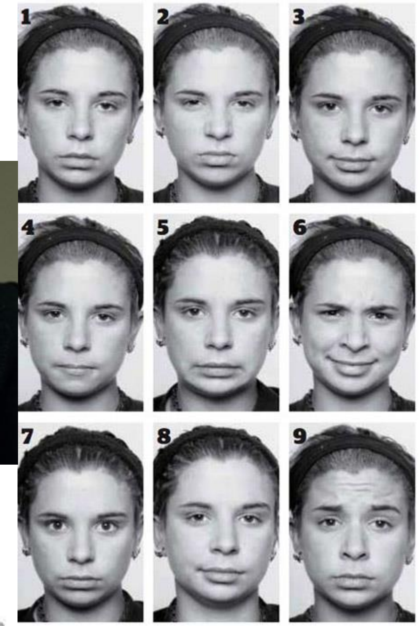
Traffic Monitoring

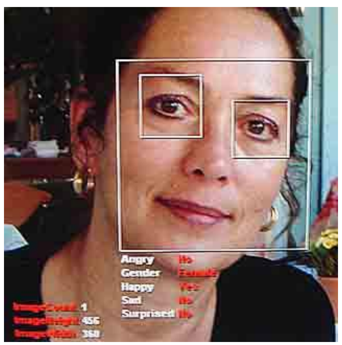




Identify faces and expressions

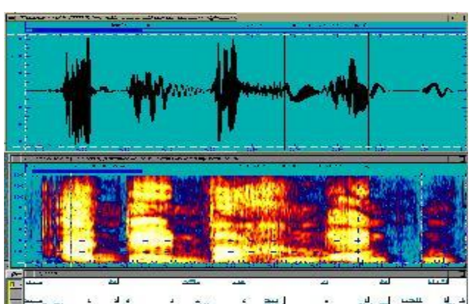


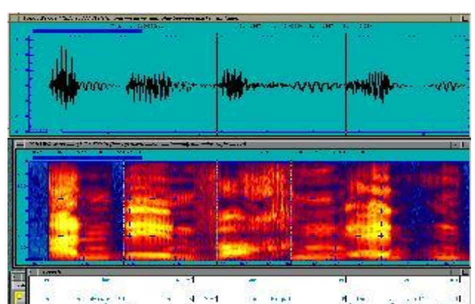




ALGORITHMS
Decision Trees
Adaboost

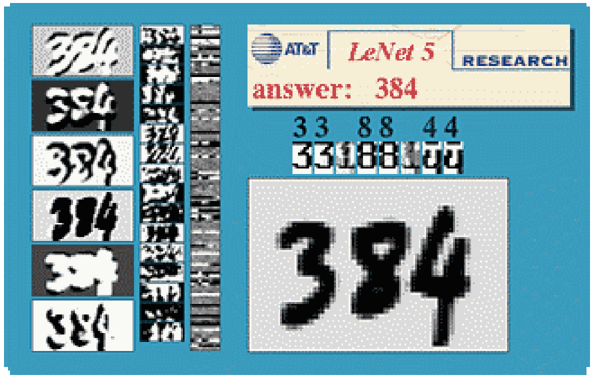
Identify vocal patterns



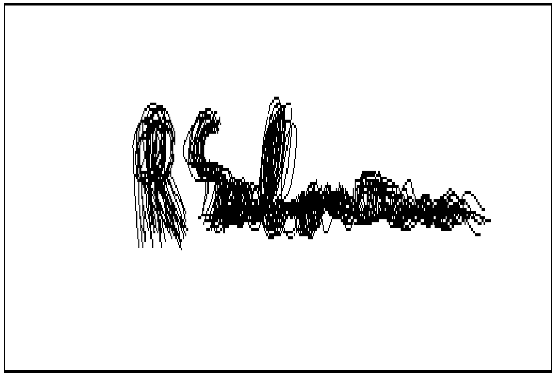


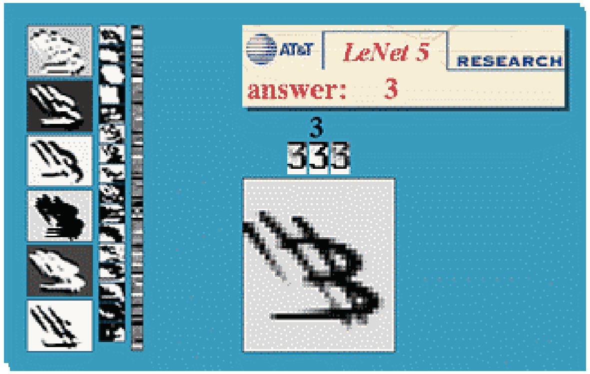
ALGORITHMS
Feature Extraction
Probabilistic Classifiers
Support Vector Machines
+ many more….

Hand-written digits

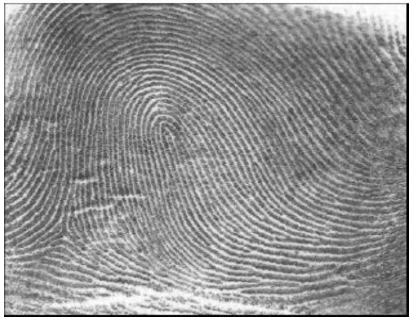


Signature Verification





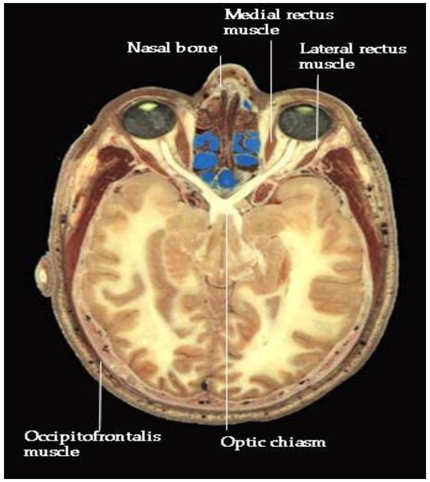
Fingerprint



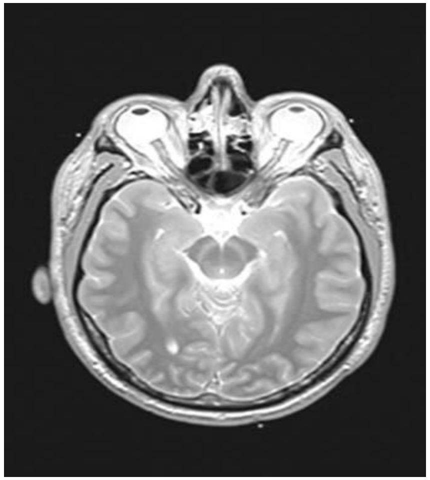
Healthcare

- Disease Identification/Diagnosis e.g. IBM Watson Genomics
- Personalized Treatment/Behavioural Modification
- Clinical Trial Research
- Radiology and Radiotherapy
- Smart Electronic Health Records

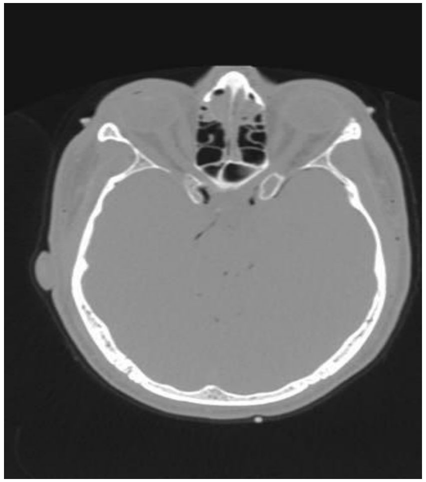
Medical diagnosis



Photo



MRI

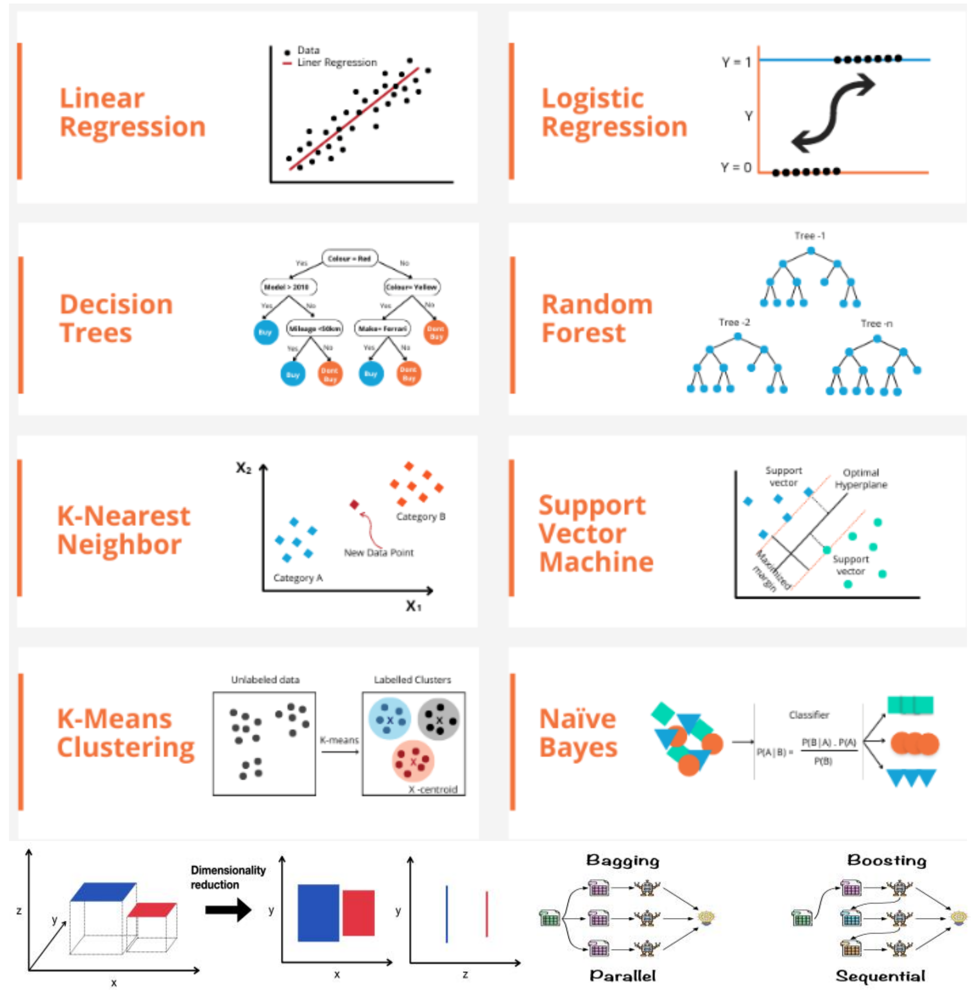


CT

Top 10 Machine Learning Algorithms For Beginners

List of Common Machine Learning Algorithms

- Linear regression
- Logistic regression
- Decision tree
- SVM algorithm
- Naive Bayes algorithm
- KNN algorithm
- K-means
- Random forest algorithm
- Dimensionality reduction algorithms
- Gradient boosting algorithm and AdaBoosting algorithm



How Do We Learn?

Human	Machine
Memorize	k-Nearest Neighbors, Case-based learning
Observe someone else, then repeat	Supervised Learning, Learning by Demonstration
Keep trying until it works (riding a bike)	Reinforcement Learning
20 Questions	Decision Tree
Pattern matching (faces, voices, languages)	Pattern Recognition
Guess that current trend will continue (stock market, real estate prices)	Regression

Probability & Statistics in Learning

- Many learning methods formulated as a probabilistic model of data
  - Can deal with uncertainty in the data
  - Missing values for some data can be handled
  - Provides a unified framework to combine many different models for different types of data
- Statistics are used to analyze the behavior of learning algorithms
  - Does the learning algorithm recover the underlying model given enough data: "consistency"
  - How fast does is do so: rate of convergence
- Common important assumption
  - Training data sampled from the true data distribution
  - The test data is sampled from the same distribution

Parametric & Nonparametric Methods

- **Nonparametric** methods:
  - No explicit "model" of the concept being learned
  - Key: keep all the data (memorize)
  - "lazy" or "memory-based" or "instance-based" or "case-based" learning
- **Parametric** methods:
  - Concept model is specified with one or more parameters
  - Key: keep a compact model, throw away individual data points
  - E.g., a Gaussian distribution; params = mean, std dev

Features and labels

- **Features** (aka covariates) are the variables that describe the data.
- **Label** (target variable/covariate): The feature we want to predict in supervised learning. So, a label is kind of like a feature

Examples:

- **Supervised** - Predict housing prices using:
  - Features: number of bathrooms, floor, number of windows, kitchen size
  - Label: House prices
- **Unsupervised** - Find patterns in books using:
  - # of pages, # of printed copies, language, # of pictures, average words per page
  - Label: there is none
    - Maybe we will ‘cluster’ the books into genres? Maybe into authors? Etc.

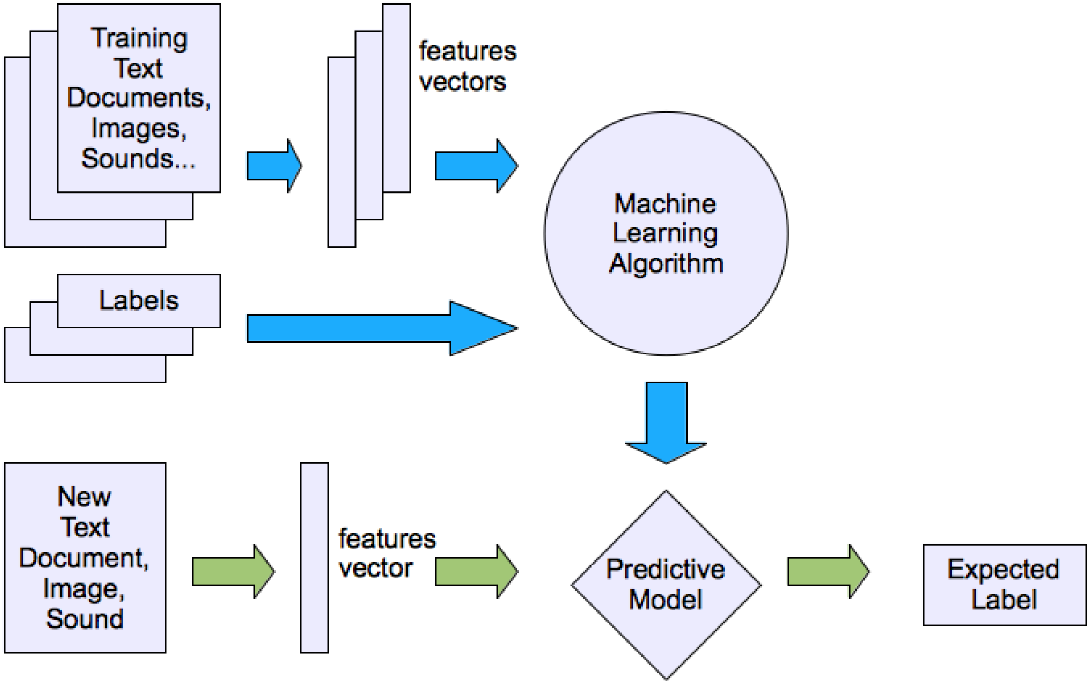
Types of Machine Learning

- **Supervised Learning.**
  - The computer is presented with example inputs and their desired outputs, given by a "**teacher**", and the goal is to learn a general rule that maps inputs to outputs. Making predictions from **labeled**
- **Unsupervised Learning.**
  - No example outputs are given to the learning algorithm, leaving it on its own to find structure in its input. Detecting patterns from **unlabeled**
- **Reinforcement Learning.**
  - A computer program interacts with a dynamic environment and must perform a certain goal (such as driving a car or playing chess). The program is provided feedback (rewards and punishments).
- **Semi-Supervised Learning.**
  - Semi-supervised learning is a branch of machine learning that combines supervised and unsupervised learning by using both labeled and unlabeled data to train artificial intelligence (AI) models for classification and regression tasks.

Learning methods

Supervised Learning:

- kNN (k Nearest Neighbors)
- Naïve Bayes
- Linear + Logistic Regression
- Support Vector Machines
- Random Forests
- Neural Networks



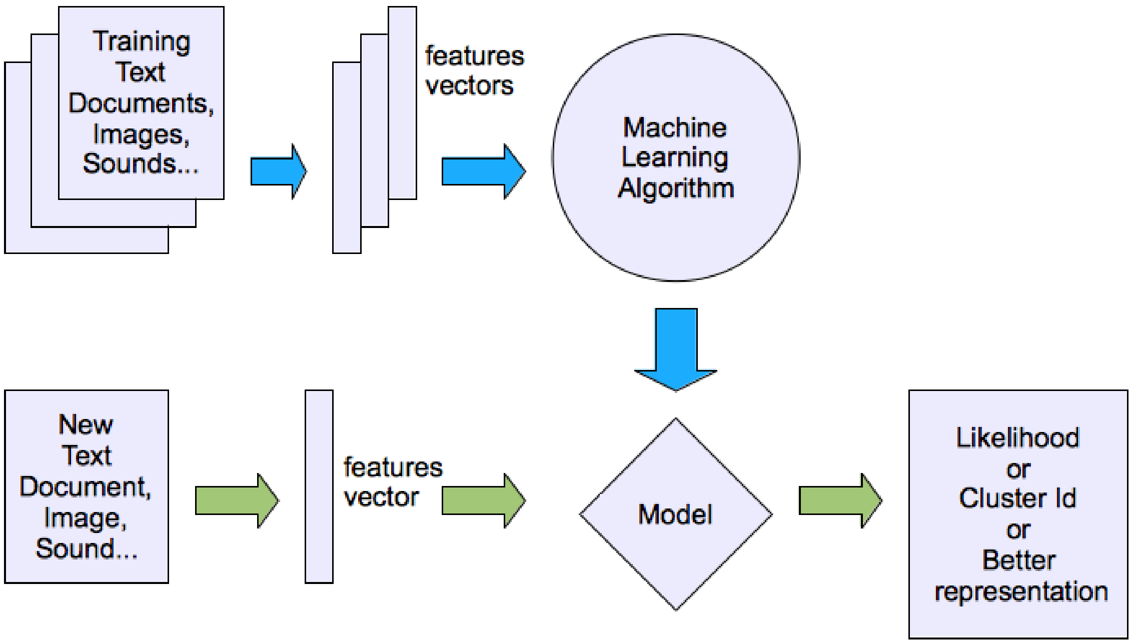
Supervised learning techniques

- **Linear classifier** (numerical functions)
- **Parametric** (Probabilistic functions)
  - Naïve Bayes, Gaussian discriminant analysis (GDA), Hidden Markov models (HMM), Probabilistic graphical models
- **Non-parametric** (Instance-based functions)
  - *K*-nearest neighbors, Kernel regression, Kernel density estimation, Local regression
- **Non-metric** (Symbolic functions)
  - Classification and regression tree (CART), decision tree
- **Aggregation**
  - Bagging (bootstrap + aggregation), Adaboost, Random forest

Learning methods

Unsupervised Learning:

- Clustering
- Topic Models
- HMMs (Hidden Markov Models)
- Neural Networks



Unsupervised learning techniques

- **Clustering**
  - K-means clustering
  - Spectral clustering
- **Density Estimation**
  - Gaussian mixture model (GMM)
  - Graphical models
- **Dimensionality reduction**
  - Principal component analysis (PCA)
  - Factor analysis

More examples

Supervised:

1. **Regression**: Predict housing prices given the prices of 1,000 previously sold houses, with various **features** describing the houses (size, # of windows, …)
2. **Classification**: Predict if a gift is a book, a toy, or something else (three discrete options only)

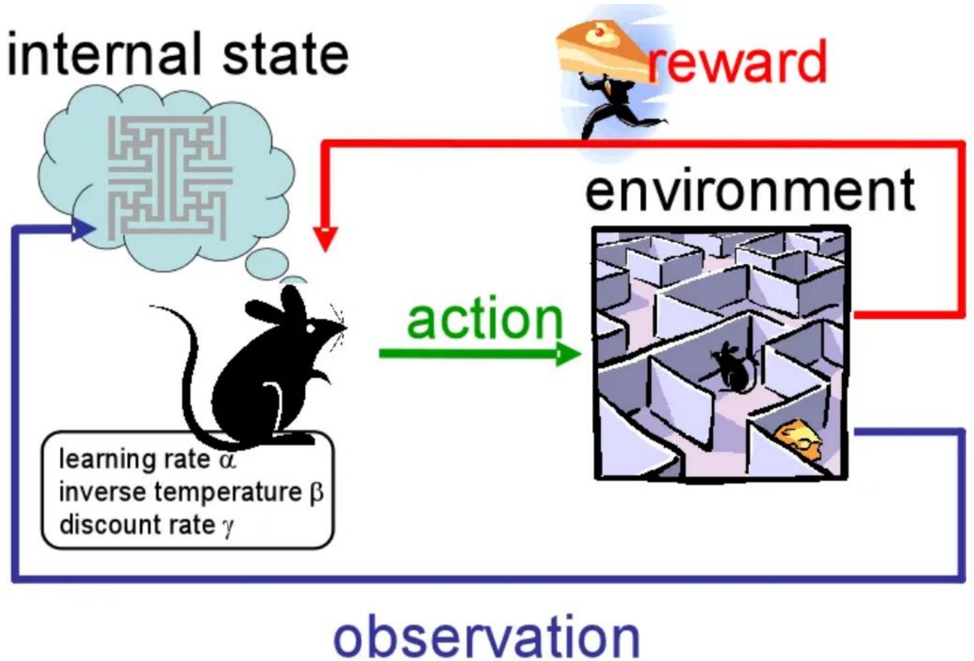
Unsupervised:

3. Get data on sports players and see if there are specific patterns.
   - Maybe the data has 3 'clusters' that mean something– basketball players, soccer players, and sumo wrestlers?
   - Maybe its clusters into categories nobody has thought about?

Learning methods

Reinforcement Learning:

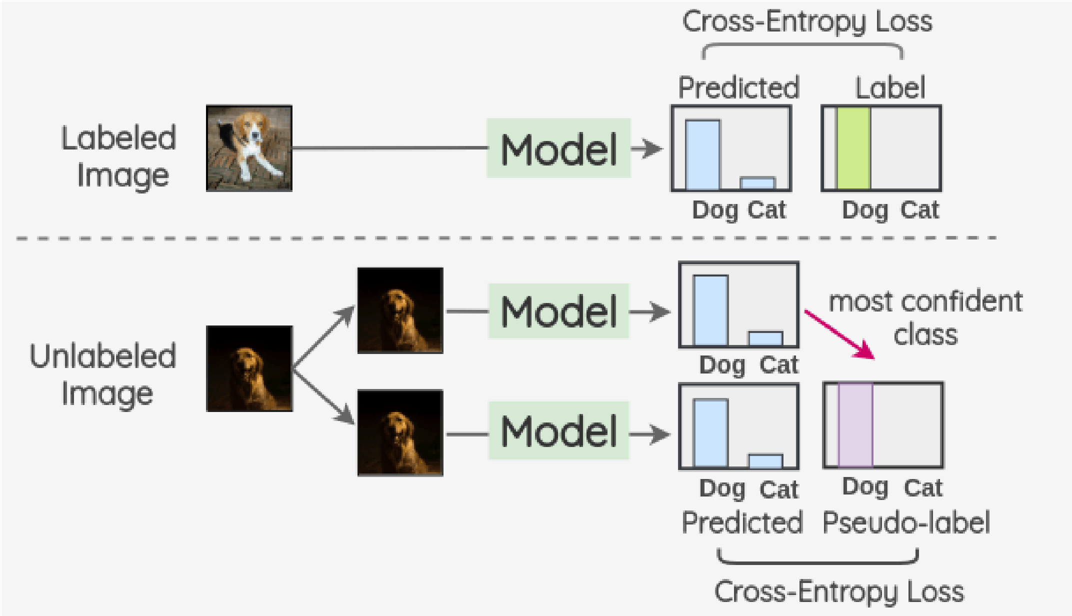
- Q-learning
- DQN
- DDPG
- A3C
- SAC



Learning methods

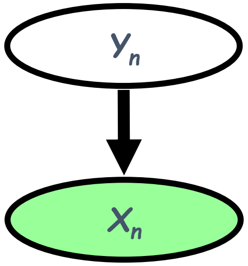
Semi-Supervised Learning

- Co-training
- Graph-based
- Self-training
- Contrastive learning
- Few shots learning
- Siamese networks



Generative vs. Discriminative Classifiers

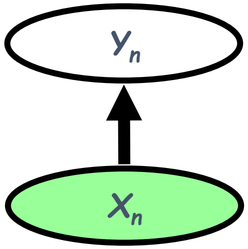
Goal: Wish to learn f : X → Y



Generative classifiers (e.g., Naïve Bayes):

- Assume some functional form for P(X|Y), P(Y)
  - This is a ***'generative'*** model of the data!
- Estimate parameters of P(X|Y), P(Y) directly from training data.
- We can model the input of the system and apply Bayes' rule:

$$P(\mathbf{x}, \mathbf{y}|W) = \underbrace{P(\mathbf{x}|\mathbf{y}, W)}_{\text{modelling inputs}}P(\mathbf{y})$$

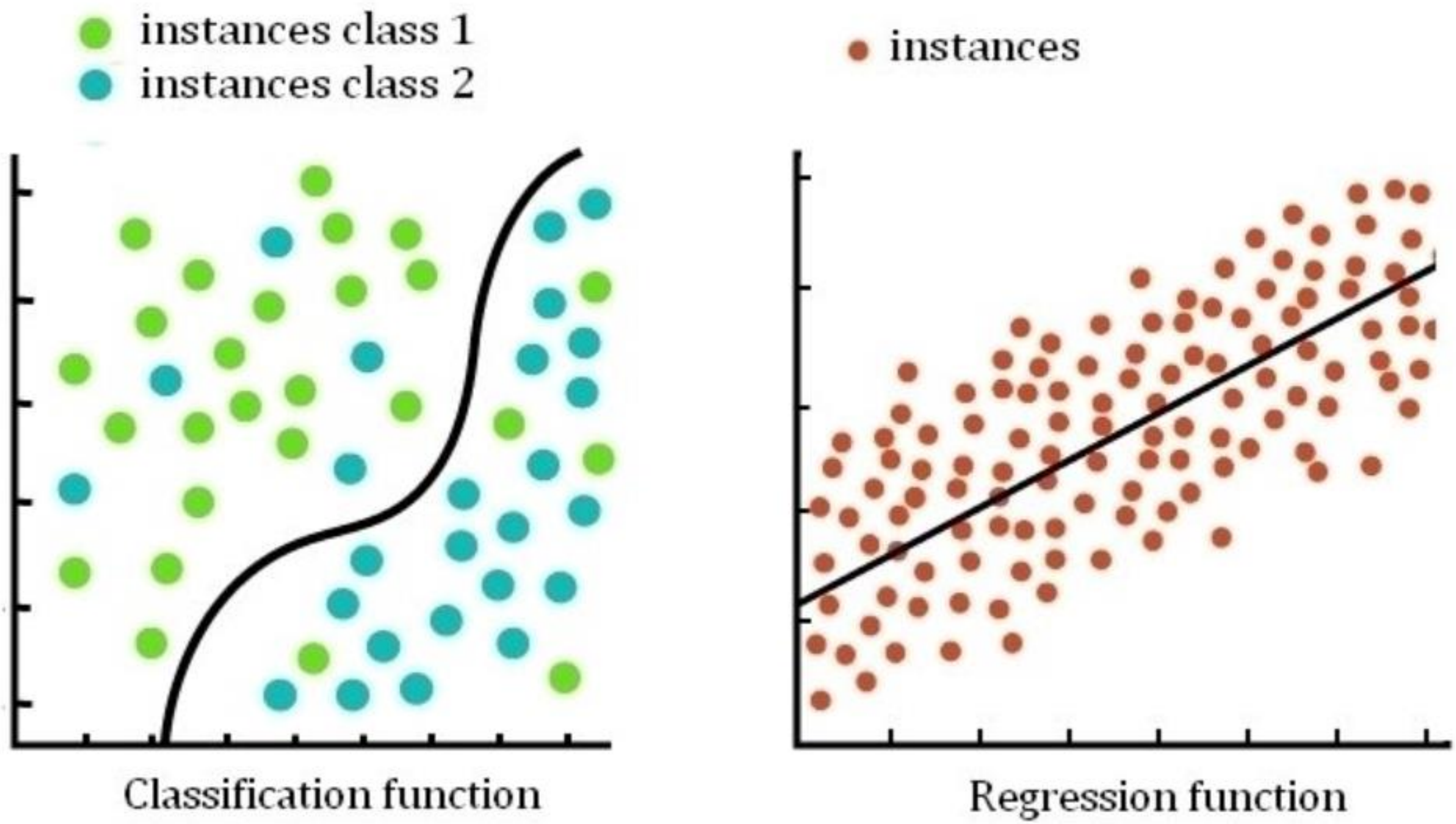


Discriminative classifiers (e.g., logistic regression)

- Directly assume some functional form for P(Y|X)
  - This is a ***'discriminative'*** model of the data!
- Estimate parameters of P(Y|X) directly from training data

$$P(\mathbf{x}, \mathbf{y}|W) = \underbrace{P(\mathbf{y}|\mathbf{x}, W)}_{\text{modelling outputs}}P(\mathbf{x})$$

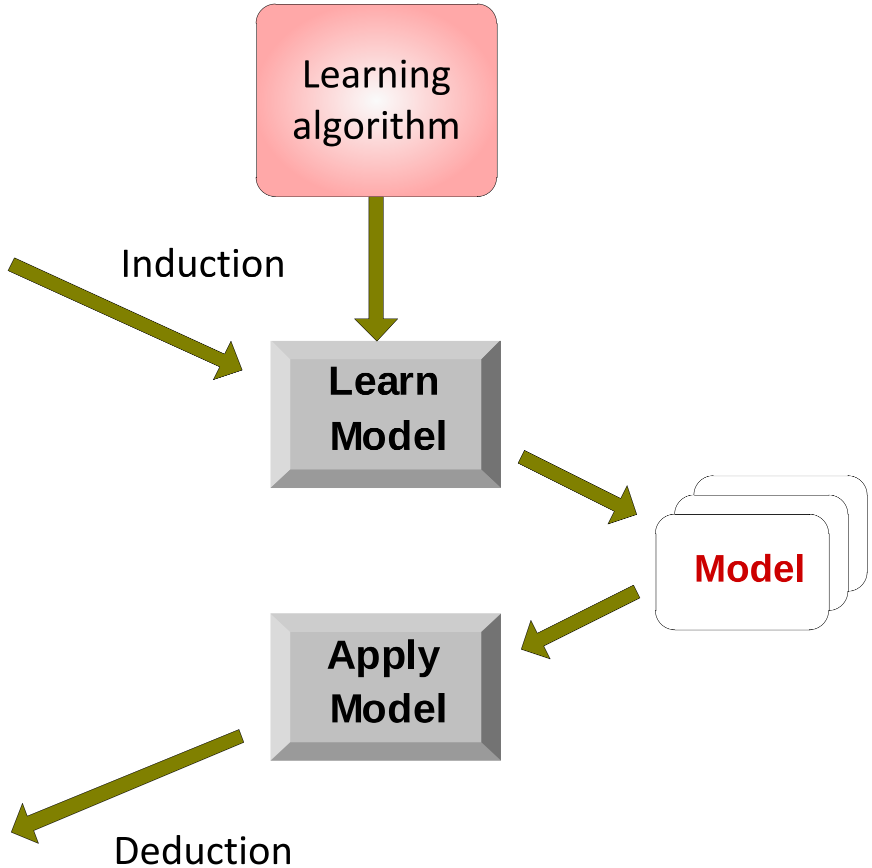
Classification vs Regression



Classification Task

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

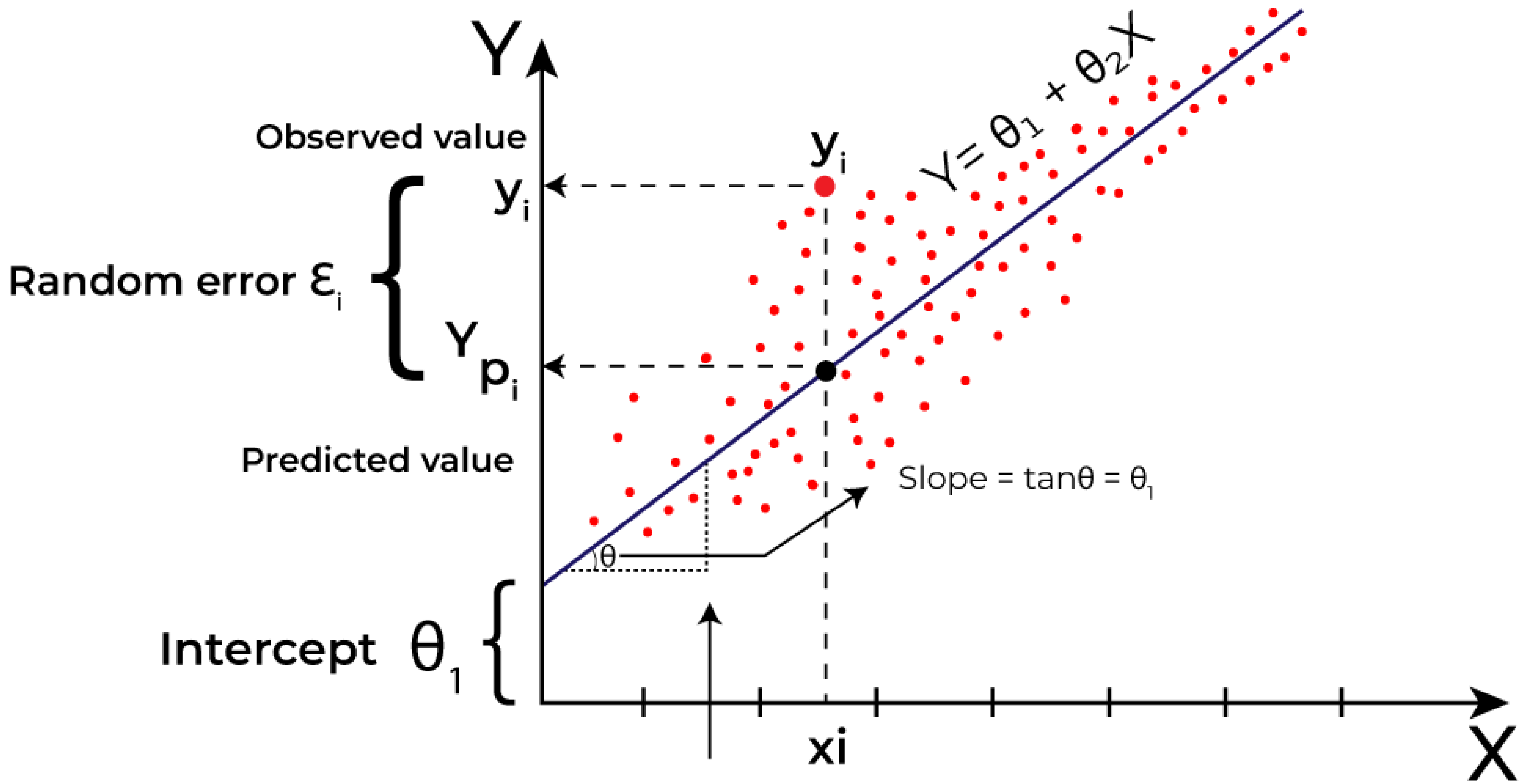
Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

Regression Task



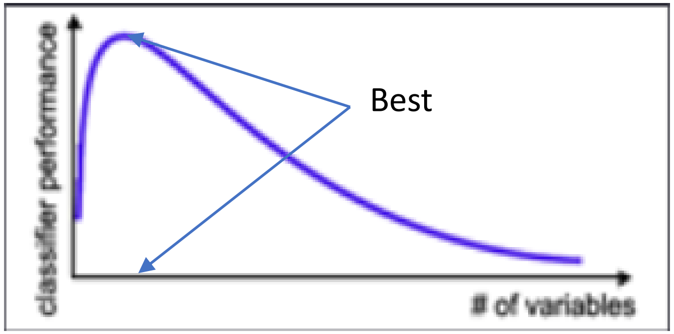
Machine Learning and Pattern Recognition

- Machine learning and pattern recognition are **<u>not</u>** the same thing.
  - This is a point that confuses many people.
- You can use machine learning to learn things that are not classifiers. For example:
  - Learn how to walk on two feet.
  - Learn how to grasp a medical tool.
- You can construct classifiers without machine learning.
  - You can hardcode a bunch of rules that the classifier applies to each pattern to estimate its class.
- However, machine learning and pattern recognition are heavily related.
  - A big part of machine learning research focuses on pattern recognition.
  - Modern pattern recognition systems are usually exclusively based on machine learning.

Dimensional reduction

- Reduce dimensions
  - **Feature selection** - Choose only a subset of features
  - **Feature transformation** - Use algorithms that transform the data into a lower dimensional space (example – PCA)

  ***Both methods often result in information loss**
- Less is More
  - In many cases the information that is lost by discarding variables is made up for by a more accurate mapping/sampling in the lower-dimensional space



Why Reduce Dimensionality?

1. Reduces time complexity: Less computation
2. Reduces space complexity: Less parameters
3. Saves the cost of acquiring the feature
4. Simpler models are more robust
5. Easier to interpret; simpler explanation
6. Data visualization (structure, groups, outliers, etc) if plotted in 2 or 3 dimensions

Steps in machine learning

- **Data collection**
  - “**training data**”, optionally with “**labels**” provided by a “teacher”.
- **Representation**
  - how the data are encoded into “**features**” when presented to learning algorithm.
- **Modeling**
  - choose the class of models that the learning algorithm will choose from.
- **Estimation**
  - find the model that best explains the data: simple and fits well.
- **Validation**
  - evaluate the learned model and compare to solution found using other model classes.
- Apply learned model to new “**test**” data