# Lecture 1

# Introduction

# The 3W of Machine Learning

- **W**hat is Machine Learning (ML)?

  What is **machine**?

  What is **learning**?

  What is **machine learning**?

- **W**hy do we need Machine Learning?

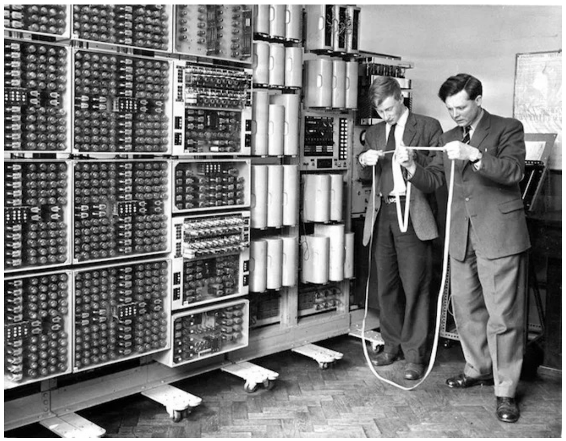
  The **necessity and importance** for machine learning

- Ho**W** to perform Machine Learning?

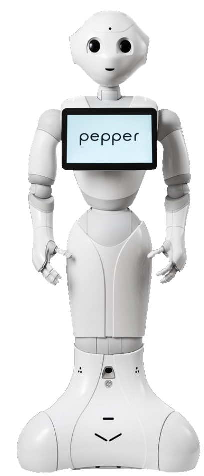
  The **building blocks** of a machine learning system

# What is Machine?

## Some examples……

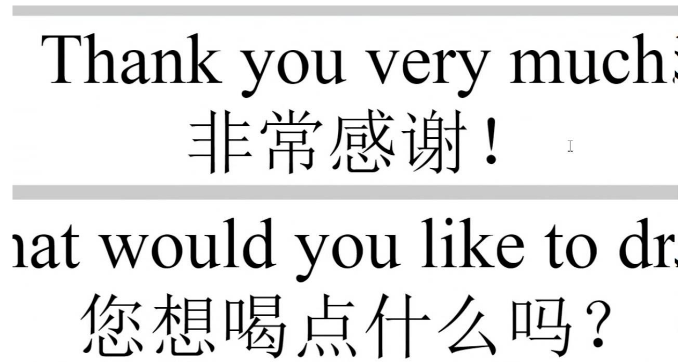




# What is Learning?

## Some examples……


# What is Machine Learning?

## An example……

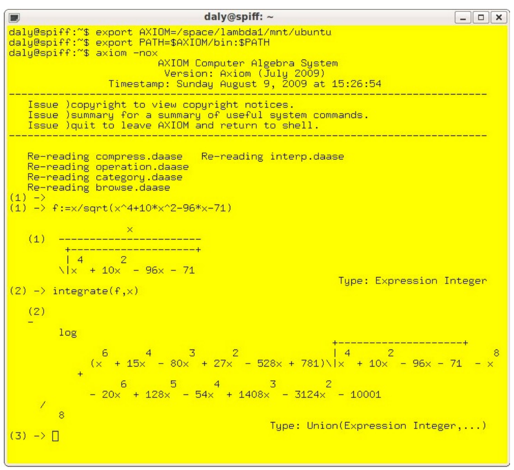
# What Can Machine Learning Do?

## Some examples……

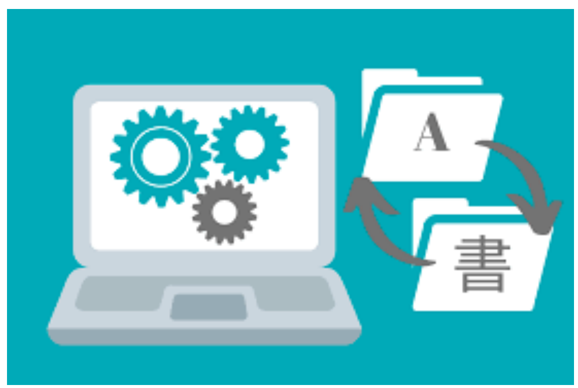
Deep Blue

Unmanned Air Vehicle

Autonomous Driving




Symbolic Computation




Machine Translation

Digit Recognition

# What Can Machine Learning Do?

## OpenAI Sora: Creating video from text

Prompt: A young man at his 20s is sitting on a piece of cloud in the sky, reading a book.

# Definition of (Machine) Learning

Herbert Simon
(1916—2001)

Learning is any process by which a system improves performance from experience.

Nobel Prize for Economics (1978)
Turing Award (1975)

Tom Mitchell

Machine Learning is the study of algorithms that

- improve their performance $P$
- at some task $T$
- with experience $E$.

Founders University Professor at CMU

# Definition by ChatGPT

SU 什么是机器学习?

机器学习是一种人工智能的分支，它使用算法和统计模型使计算机系统能够从数据中自动学习，$\boxed{\text{而不需要显式地编程}}$。机器学习的目标是通过$\boxed{\text{让计算机系统从数据中学习来改进其性能和准确性}}$，以便它们可以自动完成复杂的任务。

机器学习的基本原理是从历史数据中发现模式和规律，以便在未来的新数据上进行预测或决策。这种学习可以通过各种算法和技术来实现，例如神经网络、决策树、支持向量机和贝叶斯分类器等。

机器学习应用广泛，包括语音识别、图像识别、自然语言处理、推荐系统、金融风险管理等领域。它已经成为当今人工智能发展的重要组成部分，具有广泛的应用前景和深远的影响。

Note that the output of ChatGPT may vary with each interaction

# What Can Machine Learning Do?

## ChatGPT (OpenA

USER what is the fermat's littl

CHATGPT Fermat's Little Theorem
divisible by $p$, then $a^p$ ($a$ t
divide $a^p$ by $p$, the remain

For example, if $p = 5$ and
Therefore, $2^5$ is congruen
this case.

Fermat's Little Theorem
is necessary to perform
a useful tool for proving

; Given an array of integers nums and an integer target, return indices of the
; two numbers such that they add up to target. You may assume that each input
; would have exactly one solution, and you may not use the same element twice.
; You can return the answer in any order.

section .data
nums dd 1, 2, 3, 4, 5 ; array of integers
target dd 6 ; target integer

section .text
global _start

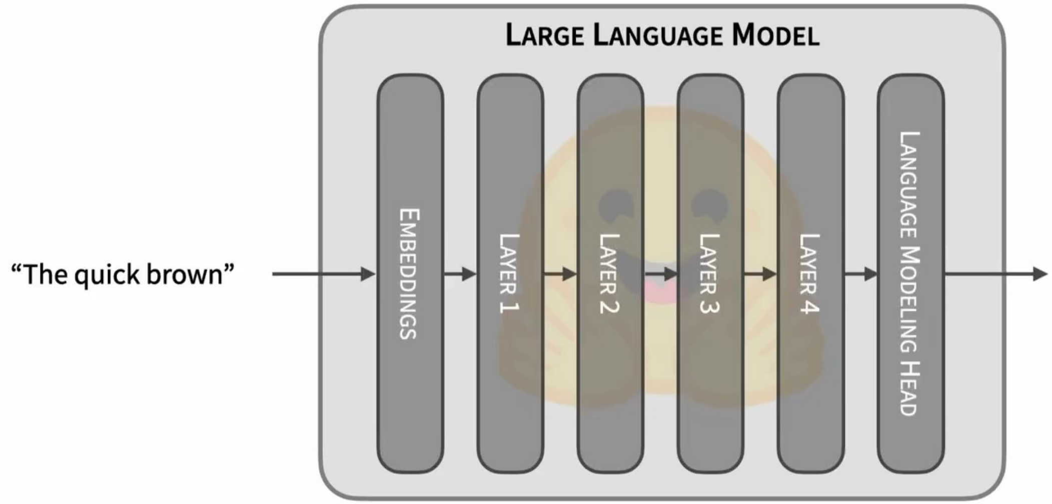
_start:
mov ebp, esp ; set up stack frame
push ebp
mov ebp, esp

Copy code

```
; initialize variables
mov eax, 0 ; first index
mov ebx, 0 ; second index
mov ecx, 0 ; sum of two numbers

; loop through nums array
mov edx, [nums] ; edx points to first element of nums
mov esi, 0 ; esi is index counter
```

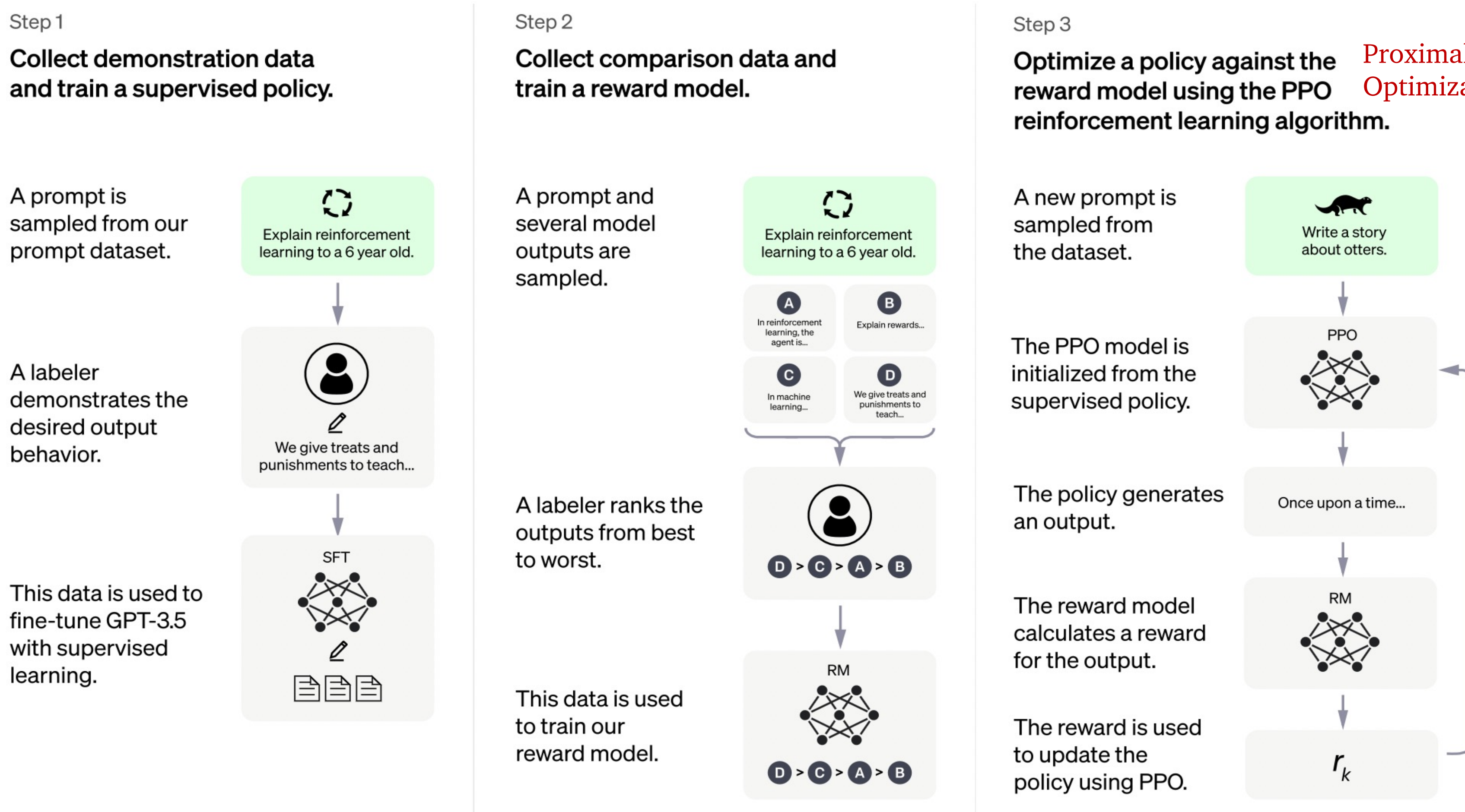
# ChatGPT (OpenAI) - Text Generation



# ChatGPT (OpenAI) - Text Generation

# ChatGPT (OpenAI) - RLHF

## RLHF: Reinforcement Learning from Human Feedback



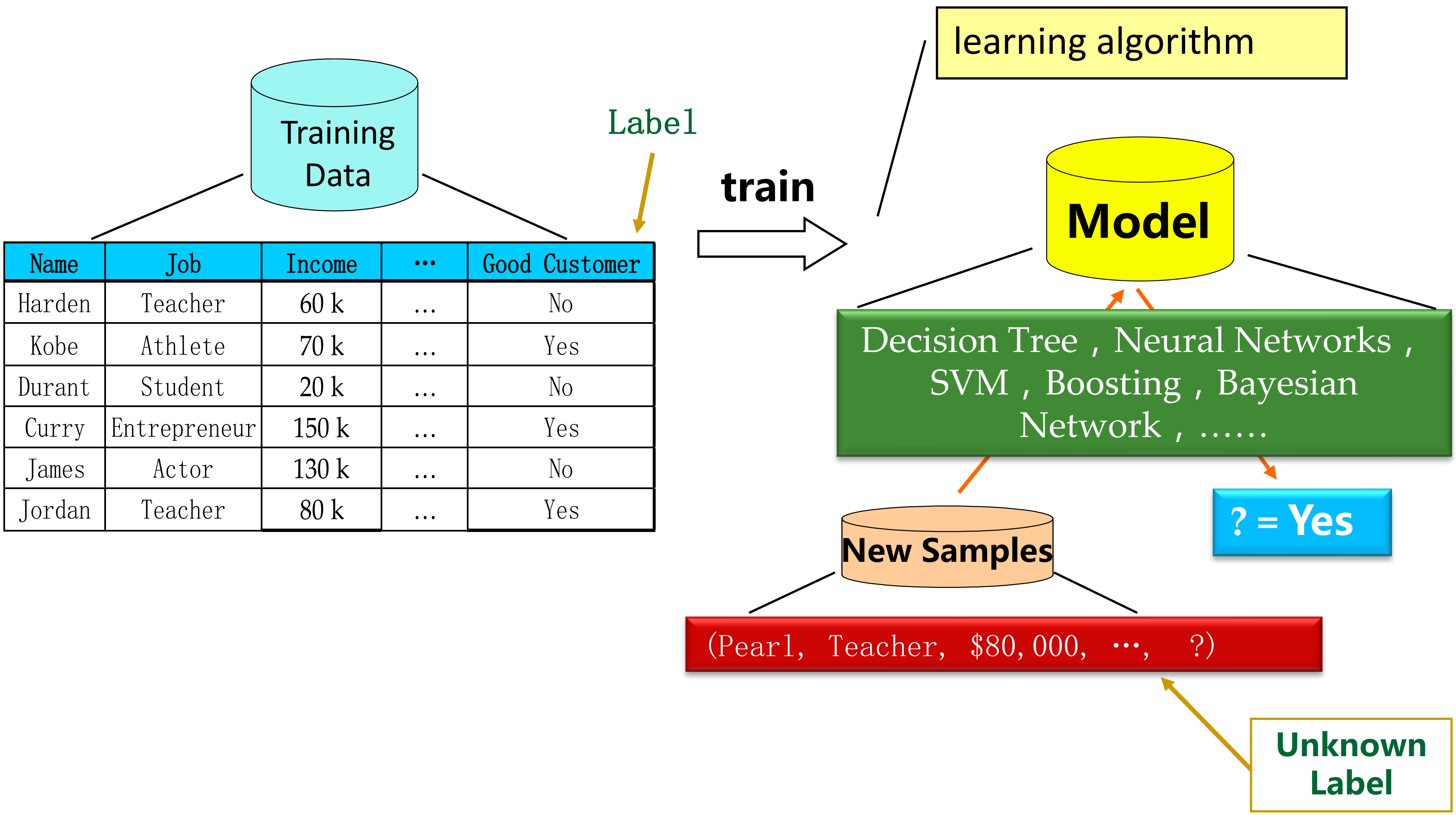

Proximal Policy Optimization

# Types of Machine Learning Algorithm

- **Supervised Learning**
  - Given: training data + desired outputs (labels)
- **Semi-Supervised Learning**
  - Given: training data + a few desired outputs
- **Unsupervised Learning**
  - Given: training data (without desired outputs)
- **Reinforcement learning**
  - Rewards from sequence of actions

$$\text{GPT-4} \stackrel{?}{=} \text{UL} + \text{SL} + \text{RL}$$

# Supervised Learning

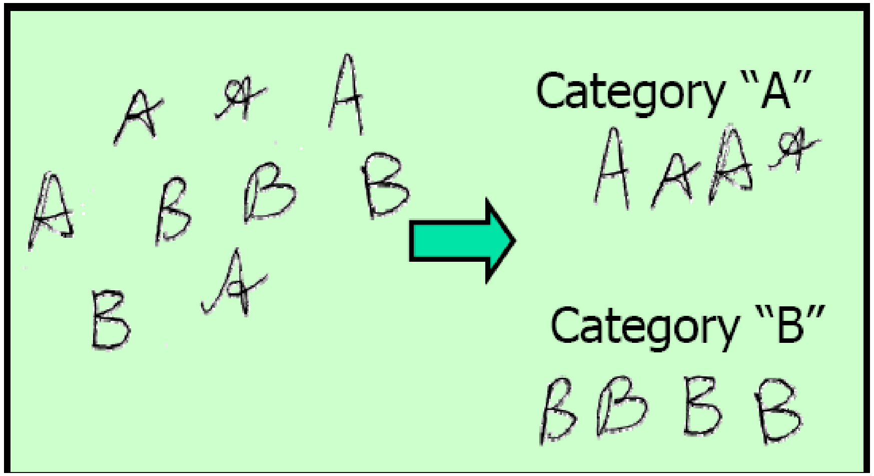



| Name | Job | Income | … | Good Customer |
|---|---|---|---|---|
| Harden | Teacher | 60 k | … | No |
| Kobe | Athlete | 70 k | … | Yes |
| Durant | Student | 20 k | … | No |
| Curry | Entrepreneur | 150 k | … | Yes |
| James | Actor | 130 k | … | No |
| Jordan | Teacher | 80 k | … | Yes |

# Supervised vs. Unsupervised Learning

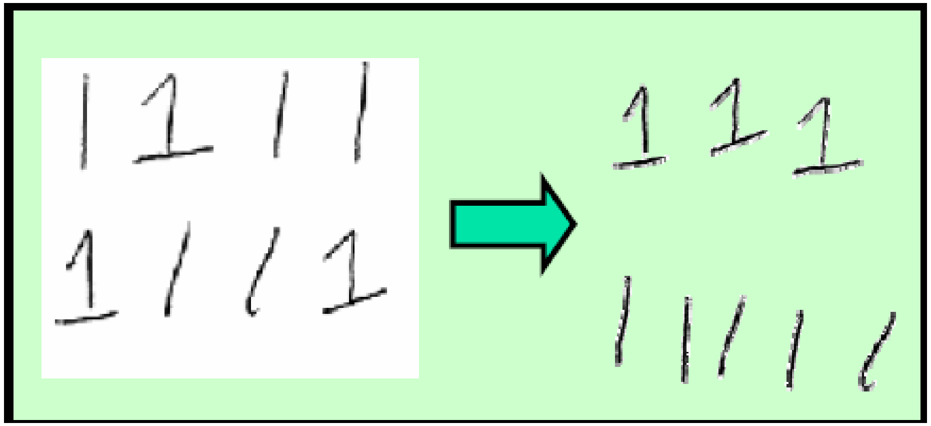
Classification: An example

We already know the categories of characters, and then classify the handwritten ones into category "A" and category "B"




Clustering: An example

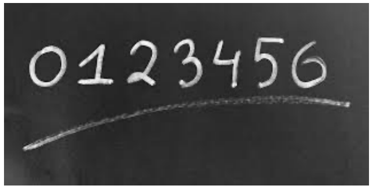
We do not know the categories of symbols, and then learn the categories and group the symbols accordingly



# Applications of Machine Learning

## 1) Character Recognition [字符识别]

**Input:**
images with characters




**Output:**
the identified character strings (0123456)

Useful in scenarios such as automatic license plate recognition (ALPR), optical character recognition (OCR), etc.

# Applications of Machine Learning (Cont.)

2) Fingerprint Recognition

[指纹识别]

**Input:**
fingerprints of some person

**Output:**

the person's identity

Useful in scenarios such as computerized access control, criminal pursuit, etc.

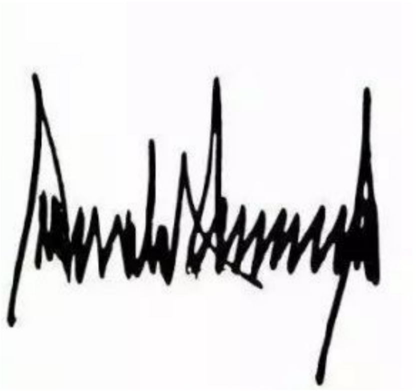
# Applications of Machine Learning (Cont.)

## 3) Signature Identification

[签名验证]

**Input:**

signature of some person
(sequence of dots)



**Output:**

the signatory's identity

Useful in scenarios such as digital signature verification, credit card anti-fraud, etc.

# Applications of Machine Learning (Cont.)

## 4) Speech Recognition [语音识别]

**Input:**
acoustic signal
(e.g. sound waves)

**Output:**

contents of the speech

"Imagine all the people.
Living life in peace"
——John Lennon

# Applications of Machine Learning (Cont.)

5) Face Detection
[人脸检测]

Useful in scenarios such as digital camera capturing, video surveillance, etc.

**Input:**
images with several people

**Output:**
locations of the peoples' faces in the image

# Applications of Machine Learning (Cont.)

## 6) Text Categorization

[文档分类]

**Input:**

document, web pages, etc.

A general view shows the Independence Monument in central Kyiv, Ukraine February 25, 2022. REUTERS/Valentyn Ogirenko

KYIV, Feb 26 (Reuters) - Ukraine on Saturday denied suggestions that it was refusing to negotiate a ceasefire with Russia but said it was also not ready to accept ultimatums or unacceptable conditions.

Mykhailo Podolyak, an adviser to the office of Ukrainian President Volodymyr Zelenskiy, told Reuters Ukraine has prepared a negotiating position but was faced with impractical negotiating conditions from Russia.

"It was yesterday that the aggressive actions of the armed forces of the Russian Federation escalated, up to evening and night mass air and missile strikes on Ukrainian cities," he said in a message. "We consider such actions only an attempt to break Ukraine and force it to accept categorically unacceptable conditions."

Reporting by Pavel Polityuk; writing by Matthias Williams; Editing by Catherine Evans

Our Standards: **The Thomson Reuters Trust Principles.**

**Output:**

category of the text, such as political, economic, military, sports, etc.

Useful in scenarios such as information retrieval, document organization, etc.

# Applications of Machine Learning - More

| Problem | Input | Output |
|---|---|---|
| Detection and diagnosis of disease | Electrocardiogram (ECG) waveforms, Electroencephalogram (EEG) waveforms | Types of cardiac conditions, classes of brain conditions |
| Natural resource identification | Multispectral images | Terrain forms, vegetation cover |
| Aerial reconnaissance | Visual, infrared, radar images | Tanks, airfields |
| Identification and counting of cells | Slides of blood samples, micro-sections of tissues | Type of cells |
| Inspection (PC boards, IC masks, textiles) | Scanned image (visible, infrared) | Acceptable/unacceptable |
| Manufacturing | 3-D images (structured light, laser, stereo) | Identify objects, pose, assembly |
| Web search | Key words specified by a user | Text relevant to the user |
| …… | …… | …… |

# How to perform Machine Learning?

For humans, learning is natural & easy

recognize a face    understand spoken words

read handwritten characters    identify items by feel

decide whether an apple is ripe by its smell    ……

For computers, learning is never easy

All in all, machine learning is important, useful, attractive, but rather challenging

Challenges ➔ Opportunities

# Basic Concepts

## Model (模型)

Descriptions which are typically mathematical in form
[以数学形式表达的性质]

*e.g. image → matrix; sound waves → frequency vector*

## Sample (样本)

Representatives of the patterns we want to classify
[分类的基本对象，学习的实例]

*e.g. fingerprint of a suspect; Electrocardiogram of a patient*

## Training Set (训练集)

A set of samples used to train classifiers
[用于训练模型的样本集合]

# Basic Concepts (Cont.)

## Test Set (测试集)

A set of samples to be classified, usually being mutually exclusive to training set

[用于测试分类器的样本集合,通常与训练集无交集]

*"Training set" vs. "Test set"* ⇔ *"Homeworks" vs. "Exams"*

## Feature (特征)

Attributes which characterize properties of the samples

[用于刻画样本性质的属性]

*e.g. to characterize a person, we may use features such as height, weight, age, salary, occupation, etc.*

# Basic Concepts (Cont.)

## Feature Vector (特征向量)

Vector formed by a group of features, usually in column form

[由一组特征组成的向量，通常表示为列向量]

$$\boxed{\mathbf{x}} = \begin{pmatrix} x_1 \\ \boxed{x_2} \\ \vdots \\ x_{\boxed{d}} \end{pmatrix}$$

vector component (in *italic*)

feature vector (in **boldface**)

dimensionality (number of features)

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{pmatrix}$$

$$\mathbf{x} = (x_1, x_2, \ldots, x_d)^{\boxed{T}}$$

transpose operator

# Basic Concepts (Cont.)

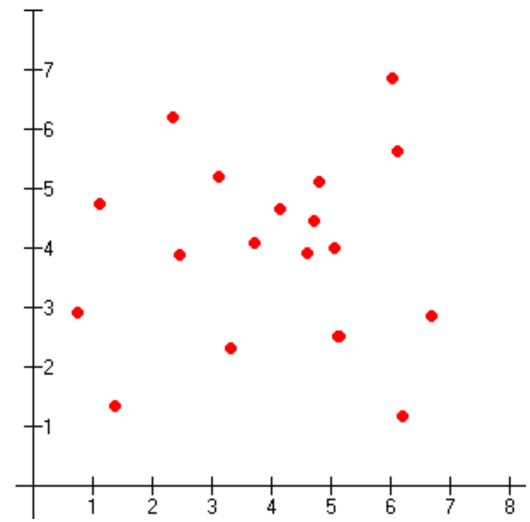
## Feature Space (特征空间)

Space containing all the possible feature vectors (由所有可能的特征向量组成的数据空间)

*e.g. the d-dimensional Euclidean space* $\mathbb{R}^d$

## Scatter Plot (散布图)

Each sample is plotted as a point in the feature space (将每个样本表示为特征空间中的一个点)

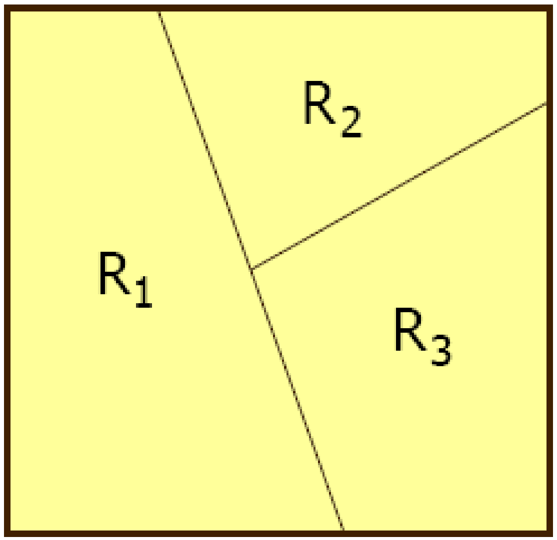



*a 2D scatter plot*

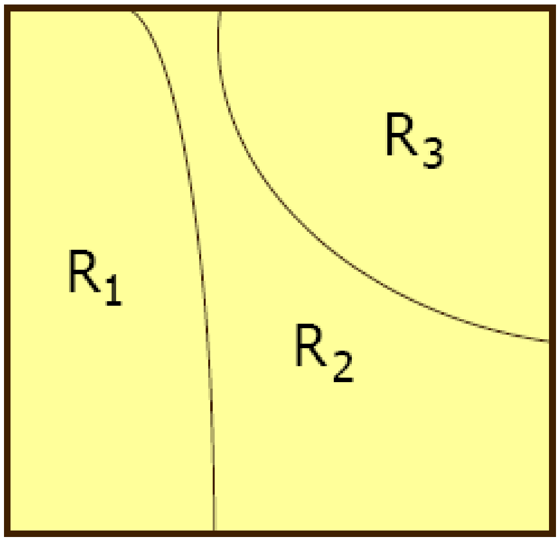
# Basic Concepts (Cont.)

## Decision Boundary (决策边界)

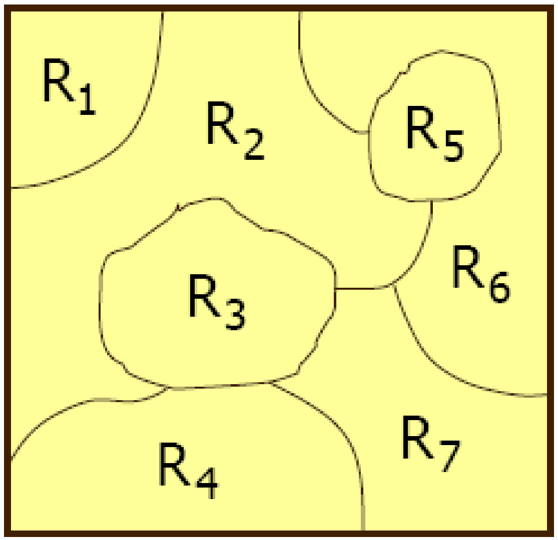
Boundaries in feature space which separate different categories (特征空间中区分各个类别的边界)




linear boundary




quadratic boundary



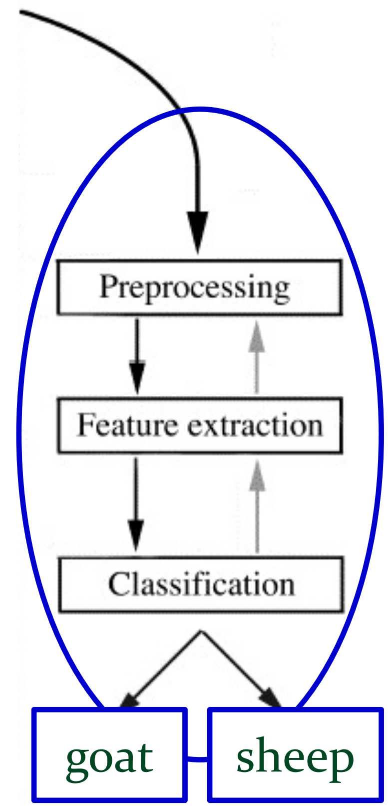

complex boundary

# How to do Machine Learning?

## An Example

**The task:** Automate the process of separating animals according to species

Separate *goat* from *sheep* [山羊 vs. 绵羊]




Three basic steps

# Example: “goat” vs. “sheep” (Cont.)

**Step I:** Preprocessing (预处理)

**Goal**: Preprocess the image captured by the camera, such that subsequent operations could be simplified without losing relevant information

Routine image processing

- Adjust the level of illumination
- Denoising
- Enhance the level of contrast

segmentation

- Isolate different objects from one another
- Isolate objects from the background

……

# Example: "goat" vs. "sheep" (Cont.)

**Step II:** Feature Extraction (特征抽取)

**Goal**: Extract features (with good distinguishing ability) from the preprocessed image to be used for subsequent classification

Goat is usually larger than sheep → "size" could be a good candidate for features

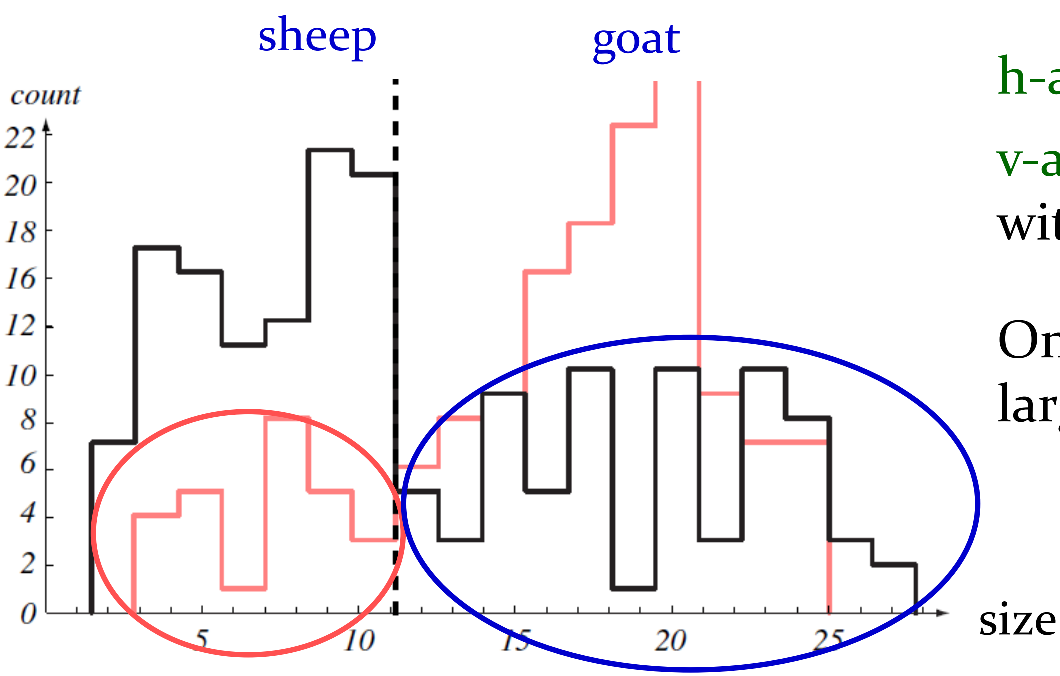
Sheep wool is usually more curly than goat → "curliness" could be another good candidate for features

……

# Example: "Goat" vs. "Sheep" (Cont.)

**Step III:** Classification (分类)

**Goal**: To distinguish different types of objects (in this case, *Goat* vs. *Sheep*) based on the extracted features




histogram for *size*

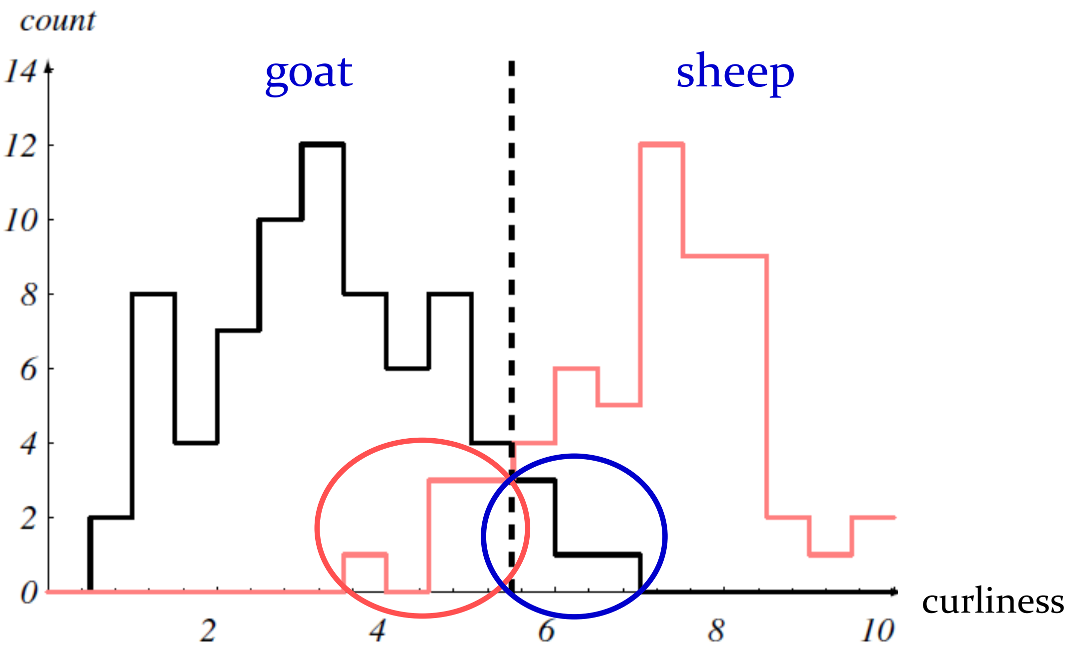
h-axis: size of the animals

v-axis: number of animals with a certain size

On average, goat is somewhat larger than sheep

Too much overlaps ➔ poor separation with the length feature

# Example: "Goat" vs. "Sheep" (Cont.)




histogram for *curliness*

h-axis: curliness of wool

v-axis: number of animals with a certain curliness

On average, sheep wool is much curlier than goat

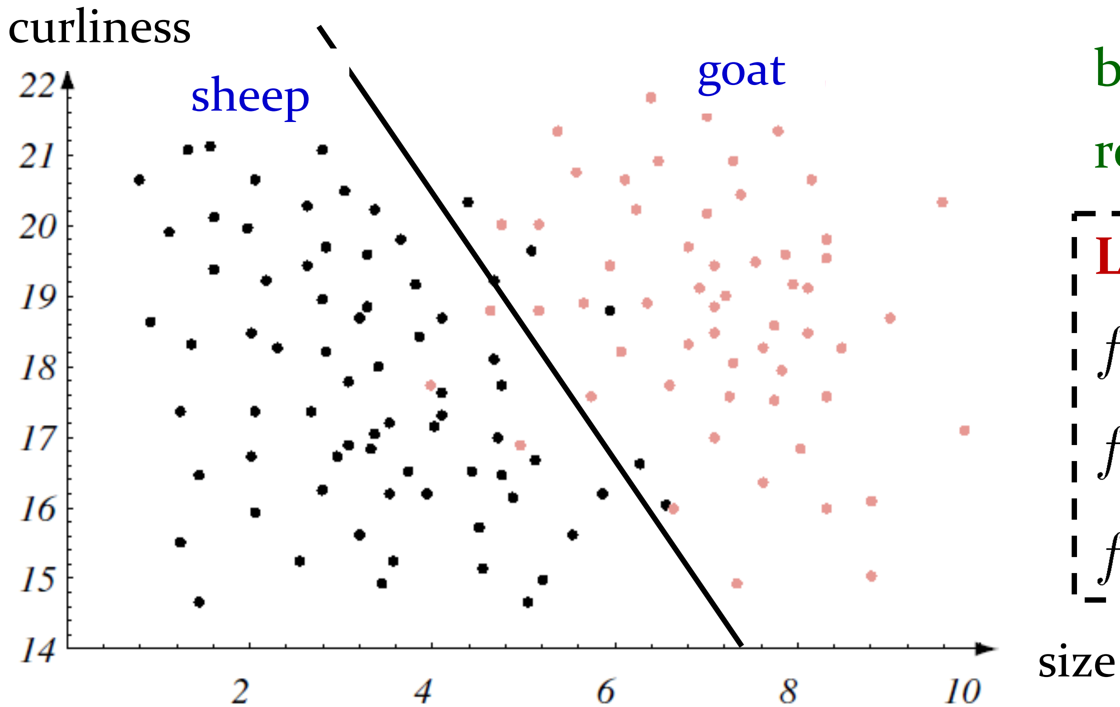
Less overlaps ➔ better separation with the lightness feature, but still a bit unsatisfactory

What if no other single feature yields better performance? ➔ Use more features at the same time!

# Example: "Goat" vs. "Sheep" (Cont.)

Using two features simultaneously

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$




scatter plot for the feature vectors

black dots: sheep samples

red dots: goat samples

**Linear** decision boundary:

$$f(x_1, x_2) = a \cdot x_1 + b \cdot x_2 + c$$

$$f(x_1, x_2) > 0 \implies \text{sheep}$$

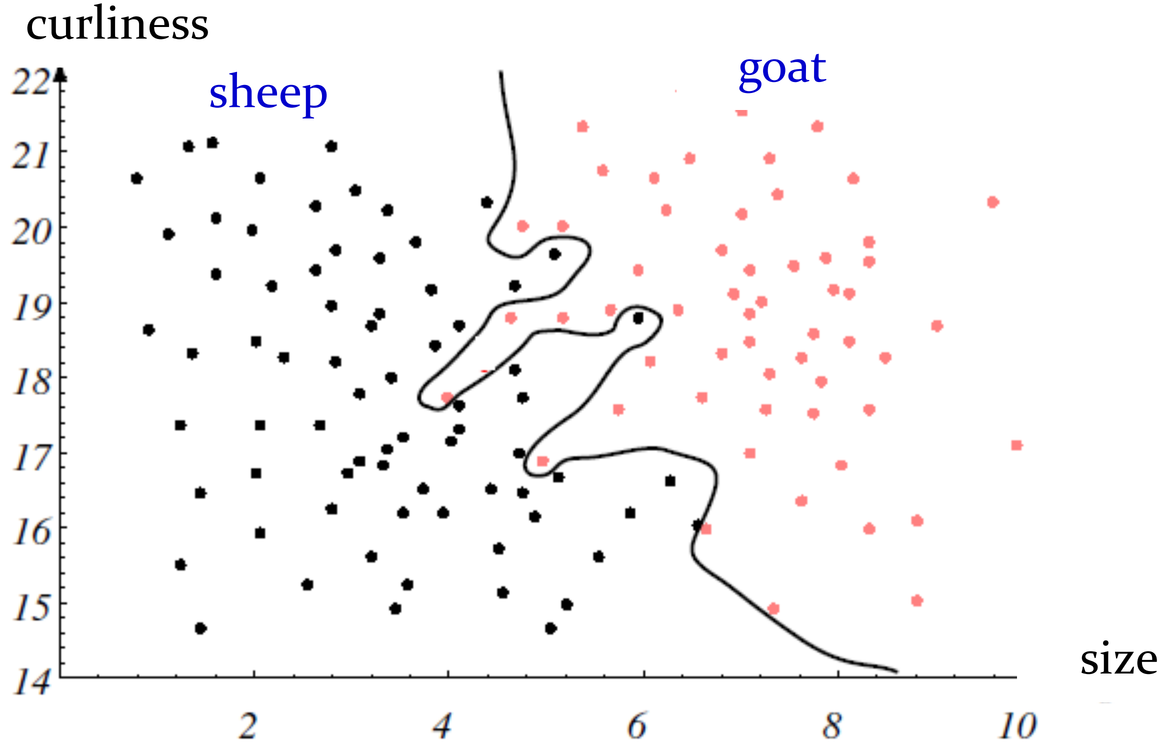
$$f(x_1, x_2) \leq 0 \implies \text{goat}$$

Much better than single feature

# Example: "Goat" vs. "Sheep" (Cont.)

**Linear** decision boundary:

**Complex** decision boundary




All the training samples (i.e. known patterns) have been separated perfectly

Linear vs non-linear, which one is better?

# Example: "Sheep" vs. "Goat" (Cont.)

Generalization

[泛化能力/推广能力]

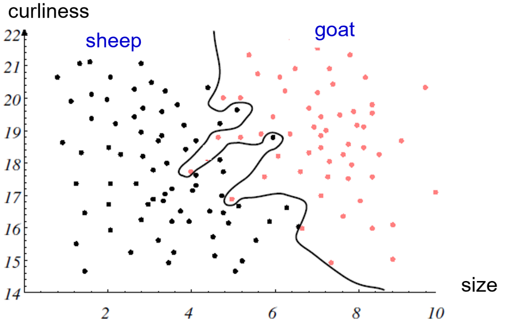
**The ultimate goal!**

The central aim of designing a classifier is to make correct decisions when presented with *novel* (*unseen/test*) patterns, not on training patterns whose labels are already known

Performance on the training set

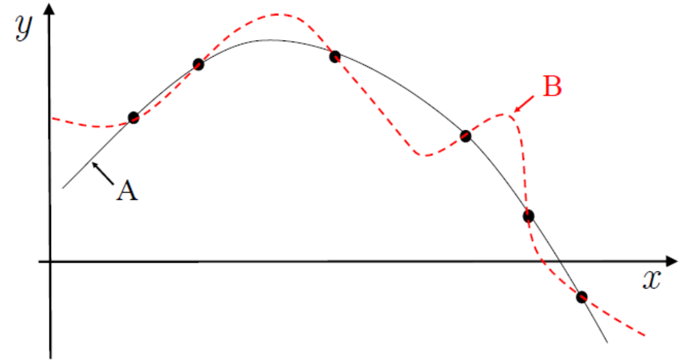
Simplicity of the classifier

Tradeoff



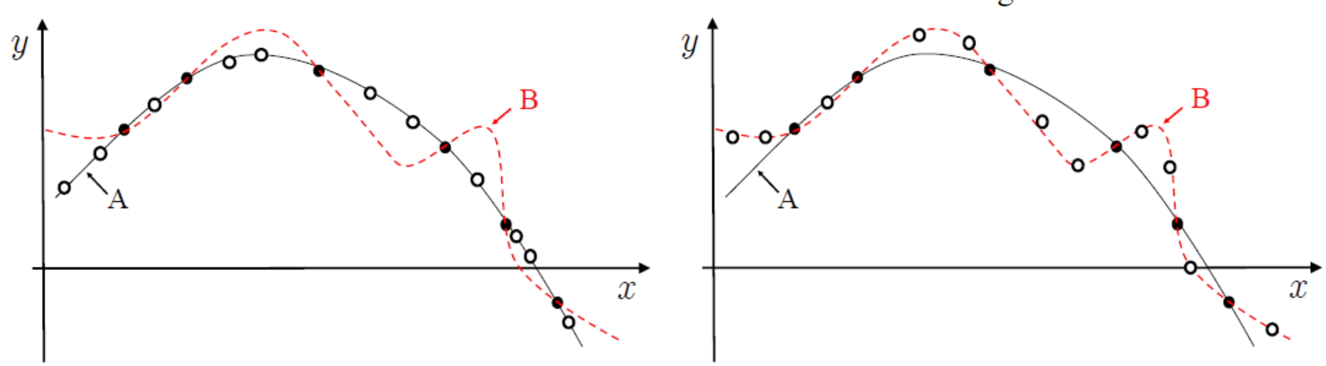
# Inductive Bias

The bias of a learning algorithm towards a particular class of hypotheses is called the inductive bias.




A or B?

There are infinite curves that are consistent with the finite training set.




(a) A is better. (b) B is better.

Fig. 1.4: There is no free lunch. (● are training samples; ○ are testing samples)

# Inductive Bias

- We can interpret inductive bias as the heuristic or value philosophy of learning algorithms for search in potentially huge *hypothesis space*.

- A fundamental and widely used principle for this question in natural science is the *Occam's razor principle*, which says that we should choose the simplest hypothesis when there is more than one hypothesis consistent with the observations.

- In practice, whether this hypothesis matches the specific problem or not usually determines the performance of the model.

# No Free Lunch

## The No Free Lunch (NFL) theorem

If a learning algorithm $\boldsymbol{H}_a$ outperforms another learning algorithm $\boldsymbol{H}_b$ in some situations, then $\boldsymbol{H}_b$ will outperform $\boldsymbol{H}_a$ in some other situations.

All learning algorithms are equally good considering all contexts. Debating "which learning algorithm is better" is meaningless without considering the specific task,

# No Free Lunch

In binary classification problems, the target function could be any functions with a function space of $\mathcal{X} \mapsto \{0,1\}$ with a function space of $\{0,1\}^{|\mathcal{X}|}$. Summing the errors of $f$ with respect to uniform distribution gives：

$$
\begin{aligned}
\sum_f E_{ote}(\mathfrak{L}_a|X,f) &= \sum_f \sum_h \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x})\ \mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x}))\ P(h \mid X, \mathfrak{L}_a) \\
&= \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \sum_h P(h \mid X, \mathfrak{L}_a) \sum_f \mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x})) \\
&= \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \sum_h P(h \mid X, \mathfrak{L}_a) \frac{1}{2} 2^{|\mathcal{X}|} \\
&= \frac{1}{2} 2^{|\mathcal{X}|} \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \sum_h P(h \mid X, \mathfrak{L}_a) \\
&= 2^{|\mathcal{X}|-1} \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \cdot 1\ .
\end{aligned}
$$

**The sum of errors is independent of the learning algorithm！**

- All learning algorithms are equally good considering all contexts.
- Debating "which learning algorithm is better" is meaningless without considering the specific task,

# Generalization Error

Definitions of the generalization error and empirical error from *"Foundations of Machine Learning"*

**Definition 2.1 (Generalization error)** *Given a hypothesis $h \in \mathcal{H}$, a target concept $c \in \mathcal{C}$, and an underlying distribution $\mathcal{D}$, the* generalization error *or* risk *of $h$ is defined by*

$$R(h) = \mathop{\mathbb{P}}_{x \sim \mathcal{D}}[h(x) \neq c(x)] = \mathop{\mathbb{E}}_{x \sim \mathcal{D}}\left[1_{h(x) \neq c(x)}\right],$$

*where $1_\omega$ is the indicator function of the event $\omega$.*[2]

**The generalization error of a hypothesis is not directly accessible**

**Definition 2.2 (Empirical error)** *Given a hypothesis $h \in \mathcal{H}$, a target concept $c \in \mathcal{C}$, and a sample $S = (x_1, \ldots, x_m)$, the* empirical error *or* empirical risk *of $h$ is defined by*

$$\widehat{R}_S(h) = \frac{1}{m} \sum_{i=1}^{m} 1_{h(x_i) \neq c(x_i)}.$$

# Generalization Error

For a fixed $h \in \mathcal{H}$, the expectation of the empirical error based on an *i.i.d.* sample $S$ is equal to the generalization error: (independent and identically distributed)

$$\underset{S\sim\mathcal{D}^m}{\mathbb{E}}[\widehat{R}_S(h)] = R(h).$$

**Proof:** by the linearity of the expectation and the fact that the sample is drawn *i.i.d.*, we can write

$$\underset{S\sim\mathcal{D}^m}{\mathbb{E}}[\widehat{R}_S(h)] = \frac{1}{m}\sum_{i=1}^{m}\underset{S\sim\mathcal{D}^m}{\mathbb{E}}[1_{h(x_i)\neq c(x_i)}] = \frac{1}{m}\sum_{i=1}^{m}\underset{S\sim\mathcal{D}^m}{\mathbb{E}}[1_{h(x)\neq c(x)}],$$

for any $x$ in sample $S$. Thus,

$$\underset{S\sim\mathcal{D}^m}{\mathbb{E}}[\widehat{R}_S(h)] = \underset{S\sim\mathcal{D}^m}{\mathbb{E}}[1_{h(x)\neq c(x)}] = \underset{x\sim\mathcal{D}}{\mathbb{E}}[1_{h(x)\neq c(x)}] = R(h).$$

# Generalization Error Bounds

The generalization error can be upper bounded by the empirical error

**Corollary 3.19 (VC-dimension generalization bounds)** *Let $\mathcal{H}$ be a family of functions taking values in $\{-1,+1\}$ with VC-dimension $d$. Then, for any $\delta > 0$, with probability at least $1-\delta$, the following holds for all $h \in \mathcal{H}$:*

$$R(h) \leq \widehat{R}_S(h) + \sqrt{\frac{2d \log \frac{em}{d}}{m}} + \sqrt{\frac{\log \frac{1}{\delta}}{2m}}.$$

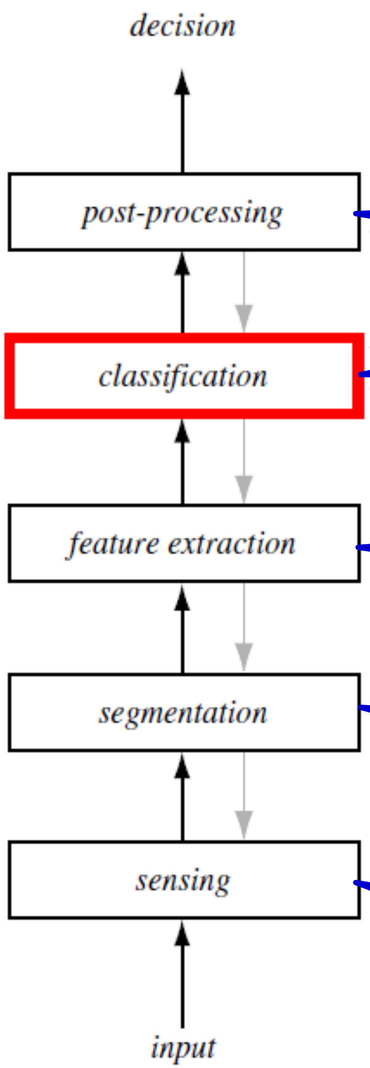
**Theorem 3.5 (Rademacher complexity bounds – binary classification )** *Let $\mathcal{H}$ be a family of functions taking values in $\{-1,+1\}$ and let $\mathcal{D}$ be the distribution over the input space $\mathcal{X}$. Then, for any $\delta > 0$, with probability at least $1-\delta$ over a sample $S$ of size $m$ drawn according to $\mathcal{D}$, each of the following holds for any $h \in \mathcal{H}$:*

$$R(h) \leq \widehat{R}_S(h) + \mathfrak{R}_m(\mathcal{H}) + \sqrt{\frac{\log \frac{1}{\delta}}{2m}}$$

$$\textit{and} \quad R(h) \leq \widehat{R}_S(h) + \widehat{\mathfrak{R}}_S(\mathcal{H}) + 3\sqrt{\frac{\log \frac{2}{\delta}}{2m}}.$$

# Machine Learning System

In addition to the usual "bottom-up" flow of data, some systems also employ feedback from higher levels back down to lower levels (gray arrows)




A post processor decide on the appropriate action based on the classification

A classifier uses extracted features to assign the sensed object to a category

A feature extractor measures object properties that are useful for classification

A segmentor isolates sensed objects from the background or from other object

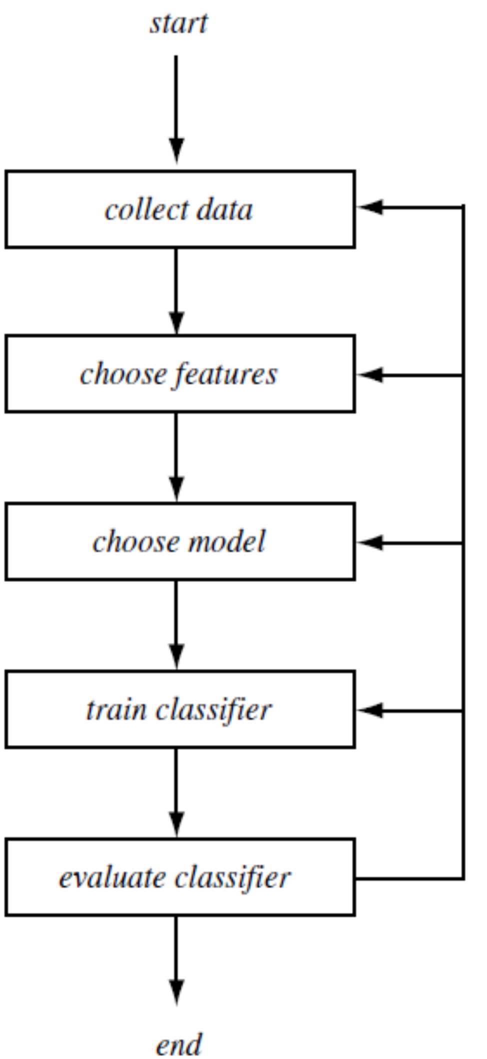
A sensor converts physical inputs (e.g. images, sounds) into digital signal data

# Design Cycle of ML System

The design of a ML system usually entails a number of different activities, such as *data collection, feature choice, model choice, classifier training, classifier evaluation.*

- Data collection accounts for a large part of the cost of developing a ML system
- Feature choice and model choice are highly domain-dependent, where *prior knowledge* (先验知识) plays very important role

  *e.g.: the orientation of the tails are the most reliable feature to distinguish sheep vs goat!*



# Important Issues in Machine Learning

- Noise
  (噪声)
- Segmentation
  (分割)
- Data Collection
  (数据采集)
- Domain Knowledge
  (领域知识)
- Feature Extraction
  (特征抽取)
- Feature Representation
  (特征表示)
- Missing Features
  (特征缺失)
- Model Selection
  (模型选择)
- Overfitting
  (过拟合)
- Context
  (上下文)
- Classifier Ensemble
  (分类器集成)
- Costs and Risks
  (代价与风险)
- Computational Complexity
  (计算复杂度)
- ......

# Noise

- General definition
  - Any property of the sensed pattern which is not due to the true underlying model but instead to intrinsic randomness of the world or the sensors
- Various types of noise exist
  - shadows, conveyor belt might shake, etc.
- Noise can reduce the reliability of the feature values measured
- Knowledge of the noise process can help improve performance

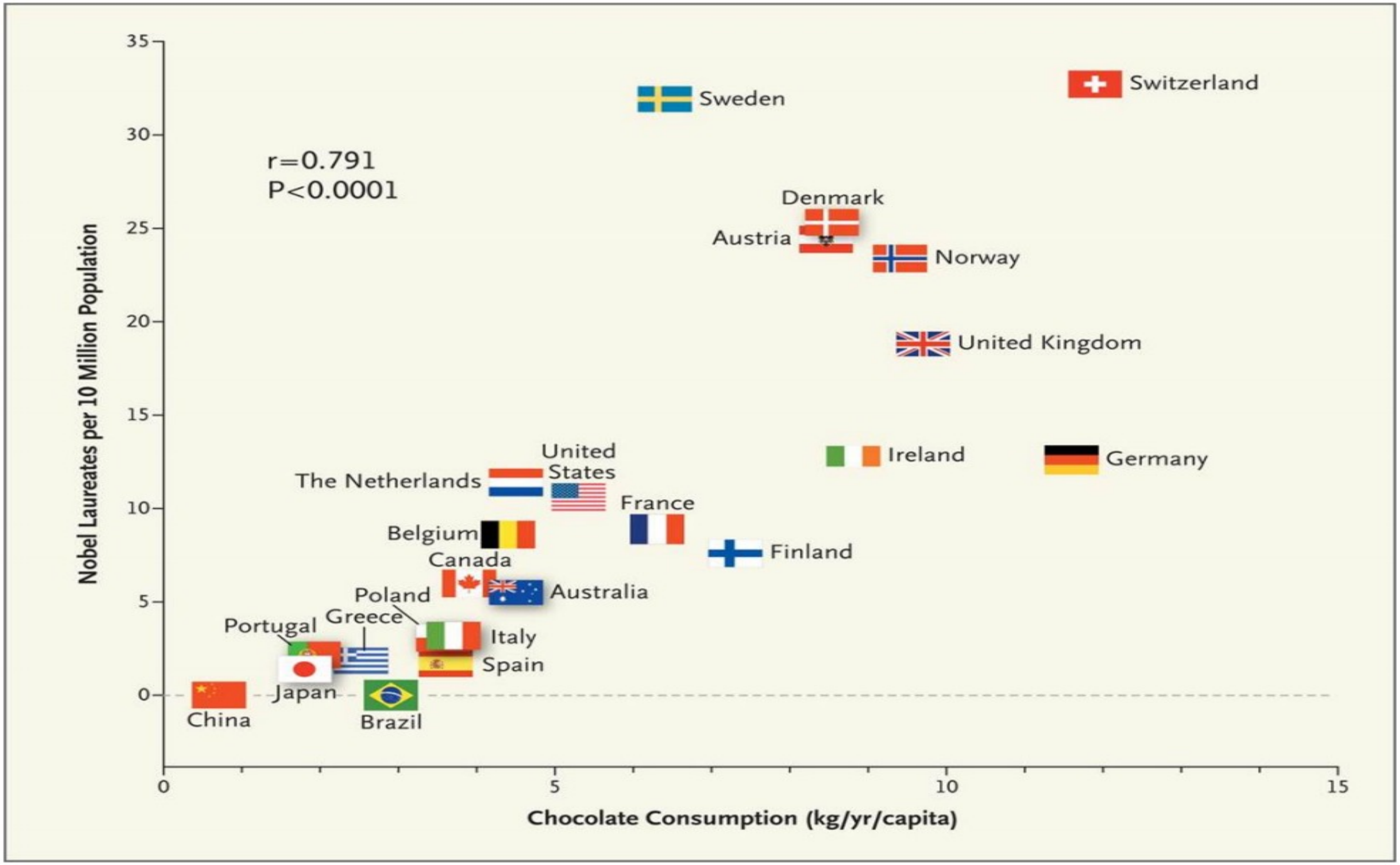
# Data Collection

- A small set of "typical" examples ➔ Preliminary study of system feasibility
- Much more data ➔ Assure good performance in the fielded system
- How do we know that we have collected:
  - Adequately large set of examples for training and testing the system?
  - Representative set of examples for training and testing the system?
- The efforts of data collection could be rather demanding

# Domain Knowledge

- There is not sufficient data for training ➔ Incorporate domain knowledge (a.k.a. prior knowledge)
- **Type I:** Incorporate domain knowledge on the patterns themselves – Difficult!
  - To recognize all types of chairs
  - Astounding variety in *number of legs*, *material*, *shape*, and *so on* ➔ What is the commonness for chairs which could be regarded as domain knowledge?
- **Type II:** Incorporate domain knowledge on the pattern generation procedure
  - Optical character recognition ➔ Assume handwritten characters are written as a sequence of strokes
  - First try to recover stroke representations ➔ deduce the character from the identified strokes

# Feature Extraction




F. H. Messerli: *Chocolate Consumption, Cognitive Function, and Nobel Laureates*, N Engl J Med 2012

# Data Representation

- Various ways for feature representation
  - **Statistical:** *feature vector* (the most popular)
  - **Template Matching:** *prototype templates*
  - **Syntactic:** *rules or grammars*
- Desired Properties
  - Samples from the same classes should have similar representations
  - Samples from different classes should have dissimilar representations
  - Feature representations should be invariant to transformations such as *translations, rotations, resizes, reflections, non-rigid deformations*
  - Intra-class variation should be small
  - Inter-class variation should be large
  - ……

# Missing Features

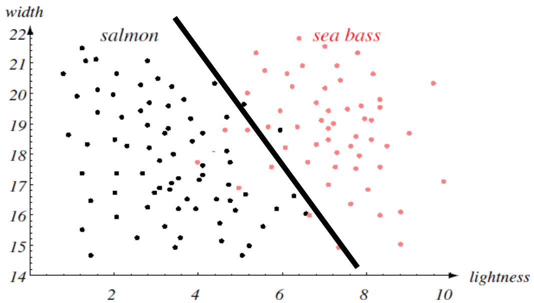
- In practical problems, values for certain features may be missing
  - Occlusion between fishes ➔ fish width can't be measured
- How could we train classifiers with missing features?
  - Naïve method could be used, but may not be optimal
    - Assuming the value of missing features is zero
    - Assigning  the average value of patterns already seen for the missing feature
  - Sophisticated method might be better, but requires extra efforts in terms of storage and time
    - Fill in the missing values with regression techniques

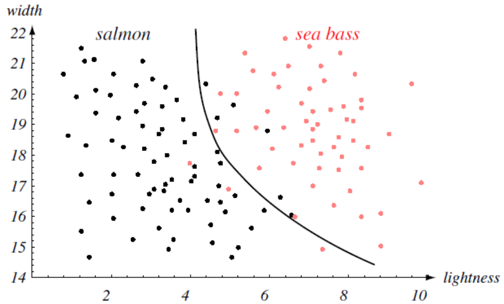
# Model Selection

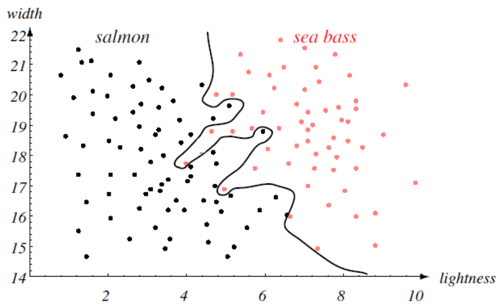
- Each machine learning method employs certain *model hypothesis*
- Every machine learning problem has its own *underlying true model*
- Fundamental questions on model selection
  - How do we know whether the hypothesized model is (relatively) consistent with the underlying true model?
  - How are we to know to reject a class of models and try another one?
  - Can we automate the process of model selection, instead of trial and error which is random and tedious?

# Overfitting

- We can get perfect classification performance on the training data by choosing complex models
  - Complex models are tuned to the particular training samples, rather than the characteristics of the true model
- Models overly complex than necessary lead to overfitting
  - Good performance on the training data, but poor performance on novel data
- How can we find principled ways to obtain best complexity?



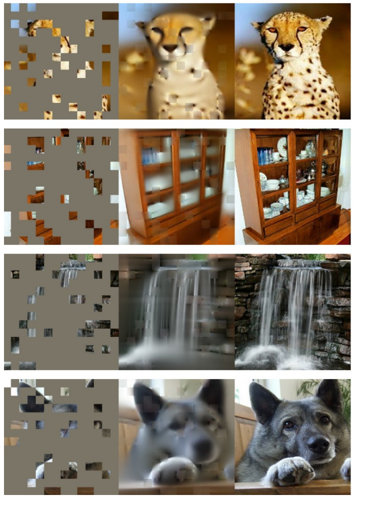



# Context

- **Context:** Input-dependent information, other than from the pattern itself
  - context of language, context of videos, etc.
- The same pattern within different contexts might have different meanings
  - Use the context of a conversation to infer the meaning of the speaker
- Context is very helpful!

*How much information are you missing*



# Classifier Ensemble

- Classifier ensemble aims to improve generalization performance by employing a number of classifiers for the same task
  - To improve the performance of speech recognizer: combine the results of *acoustic recognition* and *lip reading*
  - a.k.a. *Multi-classifier System*, *Mixture of Experts*, *Classifier Fusion*, etc.
  - Diverse ensemble techniques: *Bagging*, *Boosting*, *Random subspace.*
- How to combine different classifiers?
  - **Majority voting:** vote for the category where most classifiers agree
  - **Weighted voting:** weight each vote by classifier's confidence
  - **Stacking:** learn the rule of combination (more complicated)

# Costs and Risks

- Cost is the loss after making incorrect decisions
  - **Equal cost:** In OCR, the cost of mistaking "6" as "9" might be equal to that of mistaking "9" as "6"
  - **Unequal cost:** In COVID-19 diagnosis, the cost of mistaking "positive (阳性)" as "negative (阴性)" would be much higher than that of mistaking "negative" as "positive" (cost-sensitive learning)
- Risk is total expected cost which we want to optimize
  - Error rate (percentages of test patterns being wrongly classified)
  - Precision, Recall, Area under the ROC curve (AUC), etc.
- Questions on costs and risks
  - How do we incorporate knowledge of costs, e.g. unequal cost?
  - Can we estimate the *lowest* possible risk of any classifier?
  - ……

# Computational Complexity

- How does an algorithm scale with
  - The number of features (dimensionality)
  - The number of training patterns
  - The number of possible categories
- Brute force (蛮力) approaches might lead to perfect classification, but with impractical time and storage requirements
  - In OCR, label all possible 20x20 binary pixel images with a category ➔ use simple table lookup (查表) to classify incoming patterns
  - Labeling each of the $2^{20\times20}$ ($\approx 10^{120}$) patterns is prohibitive
- How can we find a good tradeoff between computational ease and classifier performance?

# The essence of machine learning

- A pattern exists
- We cannot pin it down mathematically
- We have data on it

# Summary

- What is Machine Learning?
  - Machine
    - Computers
  - Learning
    - Identification of a pattern as a member of a category
    - Classification: categories known ➔ assign proper class label for each pattern
    - Clustering: categories unknown ➔ learn categories and group patterns
  - Machine Learning
    - Perceive: observe the environment (i.e. interact with the real-world)
    - Process: learn to distinguish patterns of interest
    - Prediction: make sound and reasonable decisions about the categories

# Summary (Cont.)

- Why Machine Learning?
  - Machine learning is needed in designing almost all automated and intelligent systems
  - Applications of machine learning are ubiquitous
    - Character recognition (images ➔ characters)
    - Speech recognition (speech ➔ text)
    - Fingerprint recognition (fingerprints ➔ person's identity)
    - Signature identification (signature ➔ signatory's identity)
    - Face detection (images ➔ face locations)
    - Text categorization (documents ➔ semantic categories)
    - ……

# Summary (Cont.)

- How Machine Learning?
  - Basic concepts
    - model, sample, training set, test set, feature, feature vector, feature space, scatter plot, decision boundary
    - An illustrative example: “sheep” vs. “goat”
    - **Generalization**: Make correct decisions given novel patterns
  - Related fields
    - hypothesis testing, image processing, associative memory, regression, interpolation, density estimation
  - Components of Machine Learning System
    - sensing ➔ segmentation ➔ feature extraction ➔ classification ➔ post-processing ➔ ……

# Summary (Cont.)

- How Machine Learning?
  - Design Cycle of Machine Learning System
    - collect data ➔ choose features ➔ choose model ➔ train classifier ➔ evaluate classifier ➔ ......
  - Important Issues
    - Noise
    - Segmentation
    - Data Collection
    - Domain Knowledge
    - Feature Extraction
    - Feature Representation
    - Missing Features
    - Model Selection
    - Overfitting
    - Context
    - Classifier Ensemble
    - Costs and Risks
    - Computational Complexity
    - ......

# Machine Learning Conference & Journal

# Machine Learning Journals

- **JMLR** 《Journal of Machine Learning Research》
- **TPAMI**《IEEE Transactions on Pattern Analysis and Machine Intelligence》
- **TKDE**《IEEE Transactions on Knowledge and Data Engineering》
- **TNNLS** 《IEEE Transactions on Neural Network and Learning Systems》
- 国内：《中国科学 信息科学》、FCS
- …

# Machine Learning Conferences

- **ICML** (International Conference on Machine Learning)
- **NeurIPS** (Neural Information Processing Systems)
- **ICLR** (International Conference on Learning Representations)
- **KDD** (ACM SIGKDD Conference on Knowledge Discovery and Data Mining)
- **AAAI** (AAAI conference on Artificial Intelligence)
- **IJCAI** (International Joint Conference on Artificial Intelligence)
- 国内：MLA/CCML/CCDM/CCFAI

# Machine Learning History & Future

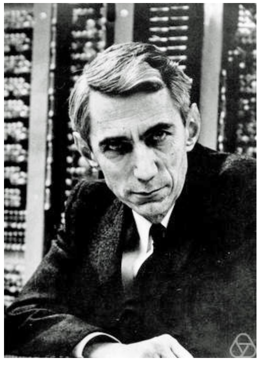
# History of AI

Marvin Minsky

John McCarthy

Julian Bigelow

Claude Shannon

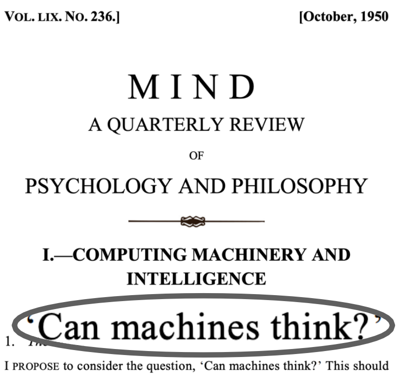


Allen Newell

Herbert Simon

**Dartmouth Workshop 1956**

# History of AI


VOL. LIX. NO. 236.] [October, 1950

MIND

A QUARTERLY REVIEW

OF

PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

‘Can machines think?’

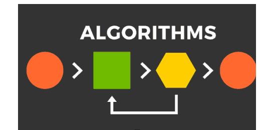
I PROPOSE to consider the question, ‘Can machines think?’ This should

First AI winter
Mid 1970’s–1980

Logic

Second AI winter
Late 1980’s–Early 1990’s

SVMs


Deep Learning

AlphaGo
AlphaFold
ChatGPT

1950
Turing Test

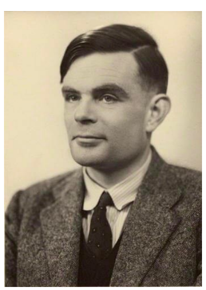
1956
The Birth of AI

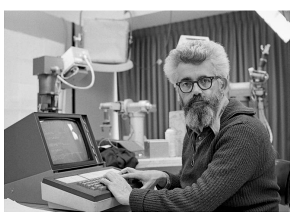
1997
Deep Blue

2006
DNNs

2012
ImageNet

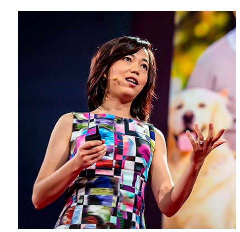
Expert System


Alan Turing


John McCarthy

Deep Blue


Geoffrey Hinton


Fei-Fei Li

# Further Reading

- *[**Mitchell, 1997**]* is the first textbook dedicated to machine learning. *[**Duda et al., 2001; Alpaydin, 2004; Flach, 2012**]* are also excellent introductory books. *[**Hastie et al., 2009**]* is a good intermediate level book, *[**Bishop, 2006**]* is a great book for reference, particularly for readers who favor Bayesian learning. *[**Shalev-Shwartz and Ben-David, 2014**]* is suitable for readers who wish to understand more about the underlying theories.

- *Machine Learning — an Artificial Intelligence Approach* *[**Michalski et al., 1983**]* which collected 16 articles contributed by 20 scholars, was the most important literature in the early days of machine learning. *[**Dietterich, 1997**]* provided a review and envision of the development of machine learning.

# The Future of AI

*PROCEEDINGS OF THE IRE*

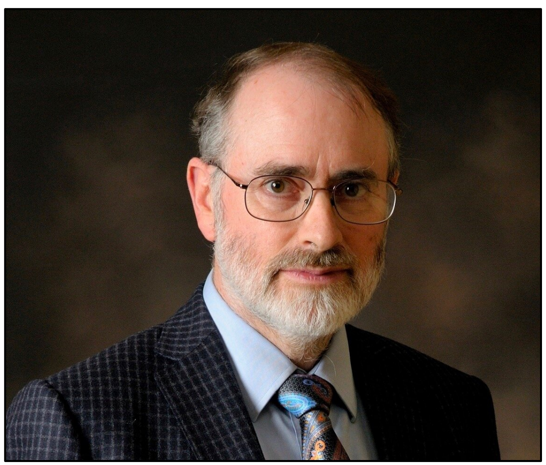
**Steps Toward Artificial Intelligence***

MARVIN MINSKY†, MEMBER, IRE

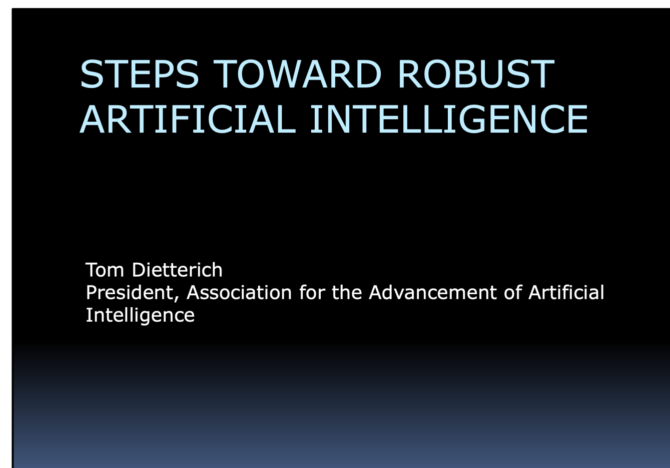
1961

Marvin Minsky (1927-2016)
人工智能先驱

2016



Tom Dietterich
**AAAI/ACM Fellow**
**AAAI Chair (2016)**

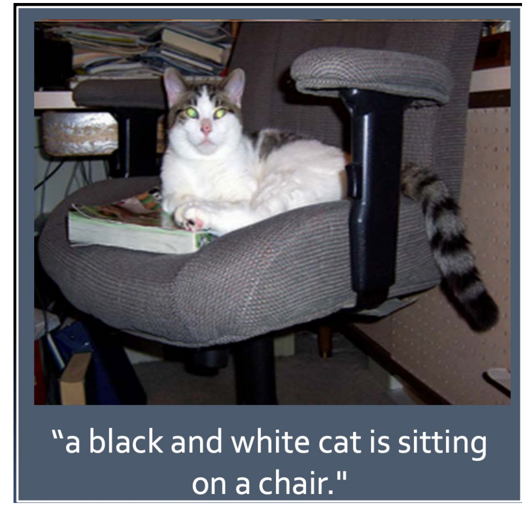


# Minsky: Difference between Computer Program and Human

"almost any error will completely paralyze ***a typical computer program***, whereas a person whose brain has failed at some attempt will find some other way to proceed. We rarely depend upon any one method. We usually know ***several different ways*** to do something, so that if one of them fails, there's always another."

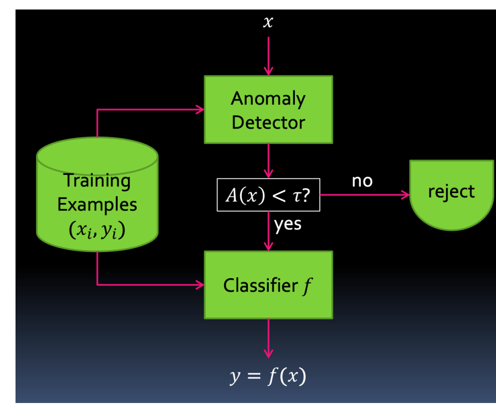
# Robust AI

A bad case



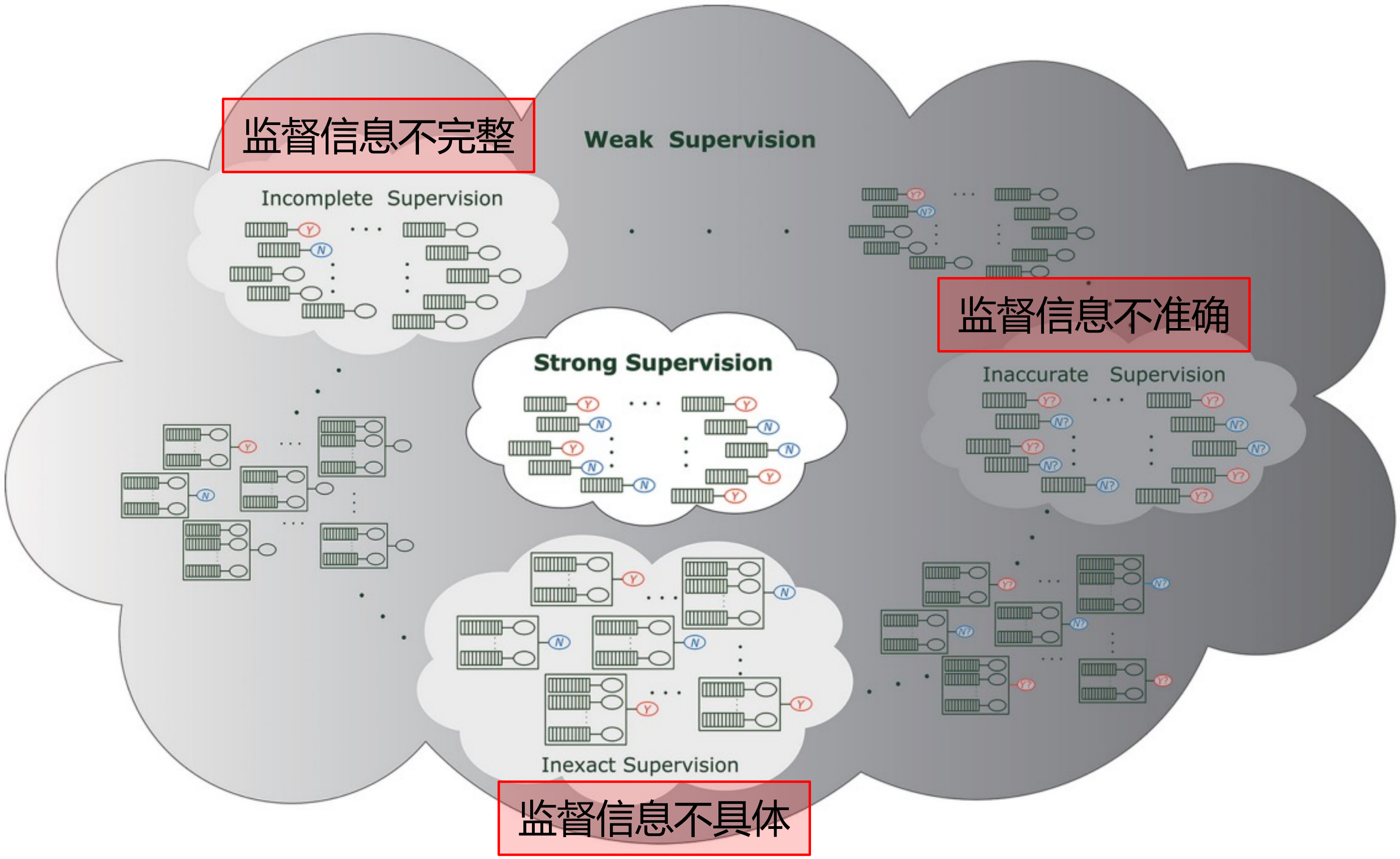

- The Need for Robust AI
  - High Stakes Applications
  - Need to Act in the face of Unknown Unknowns

- Approaches toward Robust AI
  - Robustness to Known Unknowns
  - Robustness to Unknown Unknowns




An example: Anomaly Detection

# An Example: Weakly Supervised Learning




**弱监督学习研究的是当监督信息“不完美”时，如何提升学习效果**

Zhi-Hua Zhou. A brief introduction to weakly supervised learning. National Science Review, 2018.

# An Example: Weakly Supervised Learning

**弱监督学习**
**有哪些研究内容？**

| 监督信息不完整 | 监督信息不准确 | 监督信息不具体 | … |
|---|---|---|---|
| 半监督学习 | 标记噪声学习 | 多示例学习 | 类别不平衡学习 |
| PU 学习 | 众包 | 偏标记学习 | 异常检测 |
| 主动学习 | OOD 检测 | 标记分布学习 | 增量学习 |

# PAttern Learning and Mining Lab

Introduction News Members Academics 中文首页 Application

## PAttern Learning and Mining(PALM) Lab

# Research@PALM

- Machine Learning & Data Mining
  - Multi-instance and multi-label learning
  - Semi-supervised and active learning
  - Cost-sensitive and class-imbalance learning
  - Structure learning and clustering
  - Label-distribution learning
  - Partial label learning
- Computer Vision
- Natural Language Processing