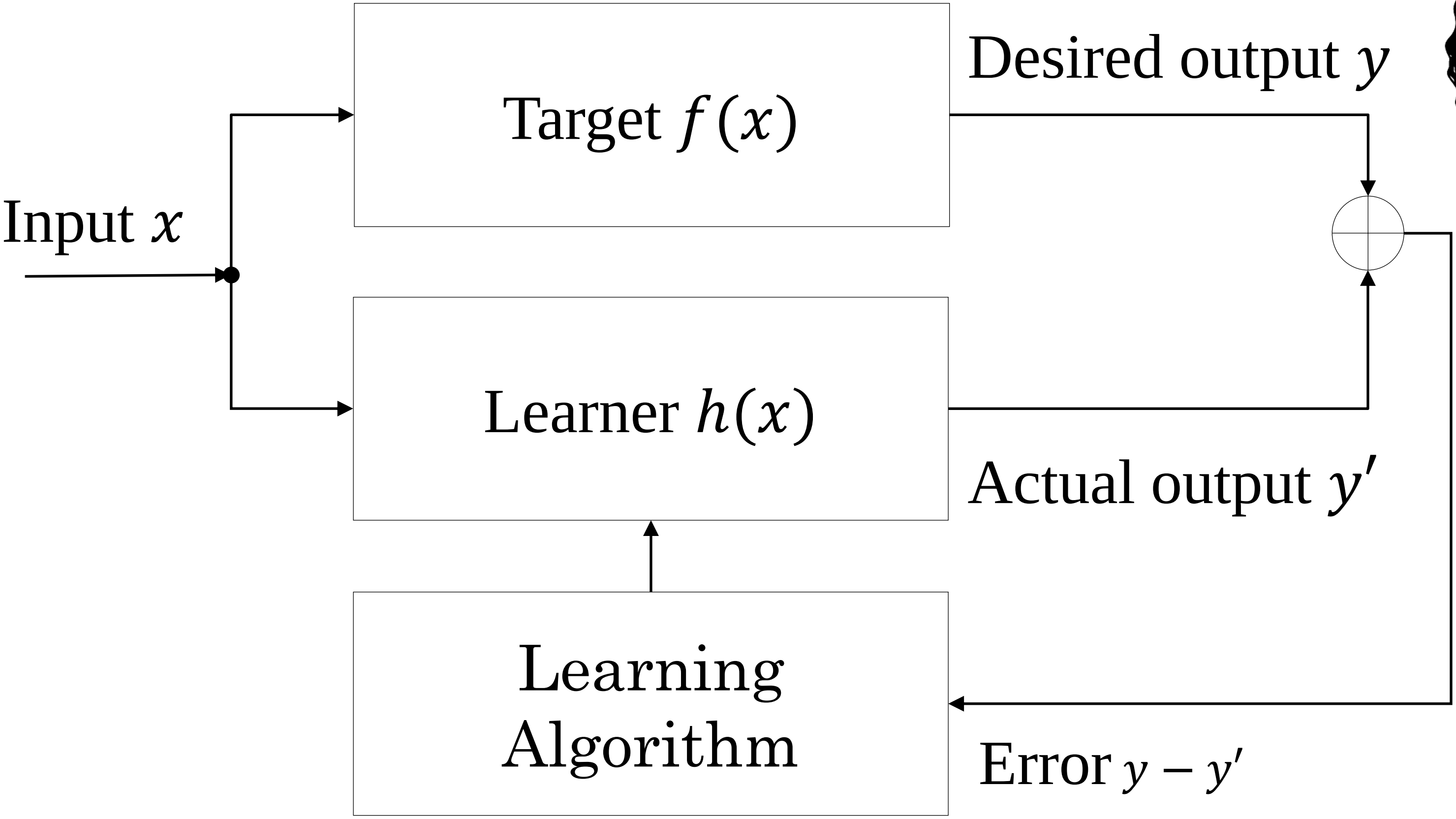
Lecture 2
Fundamentals of machine learning

Topics of this lecture

- Formulation of machine learning
- Taxonomy of learning algorithms
  - Supervised, semi-supervised, and unsupervised
  - Parametric and non-parametric
  - Online and offline
  - Evolutionary
  - Reinforcement
  - Deterministic and statistical

Formulation of machine learning (1)



Input x

Target $f(x)$

Desired output y

Learner $h(x)$

Actual output y'

Learning Algorithm

Error $y - y'$

Formulation of machine learning (2)

- Concepts to learn: $X_1, X_2, \dots, X_{N_c}$

$$X_i = \{x \in X | f(x) = y_i, \quad y_i \in Y\}$$

where $Y = \{y_1, y_2, \dots, y_{N_c}\}$ is the label set.

- A training datum is usually given as a pair (x, y), where x is the observation and y is the label given by a “teacher”.
- Learning is the process to find a good “learner” or learning model $h(x)$ to approximate the target function $f(x)$.

A concept is a set of patterns sharing some common properties (e.g. Student, Teacher, etc.)

Formulation of machine learning (3)

- In machine learning, we call $h(x)$ a **hypothesis (仮説)**. The set of all hypotheses $\mathcal{H}$ is called the **hypothesis space**.
- $\mathcal{H}$ is a set of functions (e.g. all linear functions defined in R^n) when the data are represented as points in R^n.
- Machine learning is an **optimization problem** for finding the best hypothesis $h(x)$ from $\mathcal{H}$, given an observed data set Ω.
- The goodness of a hypothesis can be evaluated by using the following “mean squared error” (MSE) function:

$$E = \frac{1}{|\Omega|} \sum_{x \in \Omega} |f(x) - h(x)|^2$$

- More theoretically, $\mathcal{H}$ is a Hilbert space, and the error can be defined using the norm $|f(x) - h(x)|$.

Formulation of machine learning (4)

- We may use a **loss function** instead of using the error function directly. The simplest loss function is 0-1 loss defined by

  $$L = \sum_{x \in \Omega} \mathbf{1}(f(x) \neq h(x))$$

  where $\mathbf{1}(P)$ is 1 if P is true, and 0 otherwise.
- The error or loss defined above is **empirical** **(経験的)** in the sense that they are defined based on the observed data only. The empirical cost or loss may not be the same as the **predictive value** **（予測的）** when we have more data.
- The best predictive error E^* or loss L^* is called the Bayes error or Bayes loss, and the hypothesis $h^*(x)$ that achieves the best error/loss is called the **Bayes Rule**. The goal of machine learning is to find $h^*(x)$ from $\mathcal{H}$.

Formulation of machine learning (5)

- To find the best hypothesis, however, we cannot use the MSE directly because the problem is **ill-posed**. That is, even if the obtained hypothesis is good for given data, it may not **generalize well** for unknown data.
- To avoid the problem, we usually introduce a **regularization factor** In the objective function.
- For example, if the hypothesis depends on a set of parameters $\theta = \{\theta_1, \theta_2, \ldots, \theta_m\}$, we may consider θ a m-dimension vector, and define the objective function as follows:

$$\min_{\theta} \sum_{x \in \Omega} |f(x) - h_\theta(x)|^2 + \lambda|\theta|$$

- where λ is a parameter for judging the balance between the error and regularization factor.

Formulation of machine learning (6)

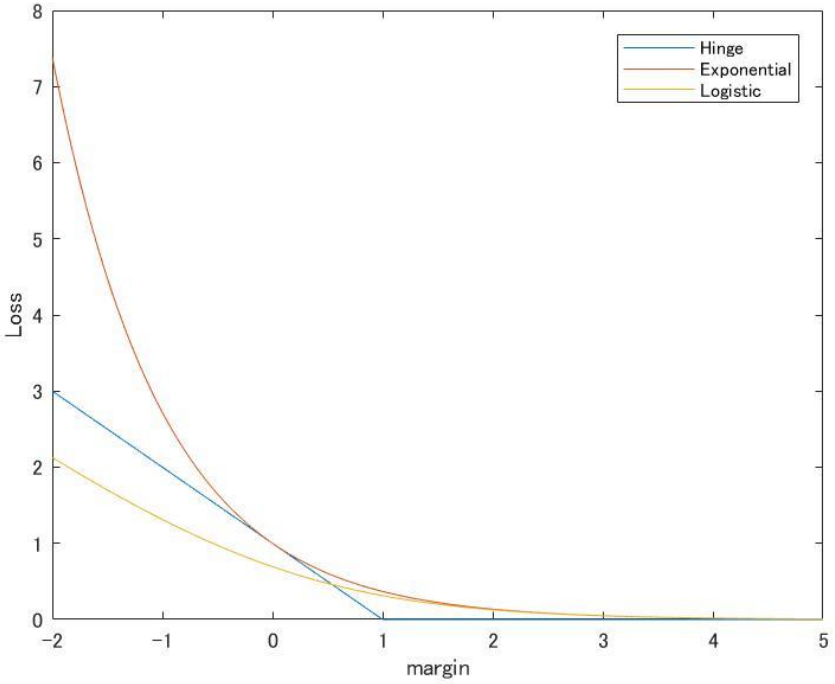
- The often used norm for the regularization factor is Euclidean norm. We may also use the norm of $h(x)$ defined in the Hilbert space $\mathcal{H}$.
- The physical meaning of regularization is to find the **most smooth solution** amount others, to improve the generalization ability.
- For *sparse learning*, we usually introduce a factor to encourage the learner *parsimony* (smaller model is better).
- If $f(x)$ takes values from R^{N_o}, the problem is called **regression**, where N_o is the number of output variables.
- For regression problem, 0-1 loss is not suitable because a good hypothesis $h(x)$ may not exactly equal to $f(x)$ for $x \in \Omega$.
- Instead, we can use other loss functions, such as
  - Hinge loss: $L(u) = \max\{1 - u, 0\}$
  - Exponential loss: $L(u) = e^{-u}$
  - Logistic loss: $L(u) = \log(1 + e^{-u})$

Formulation of machine learning (7)

- Note that u in the loss function can be defined as $f(x) \times h(x)$, which is called the ***margin***.

If the desired value is $f(x)$=1, and the actual output is $h(x)$=0.9, the Hinge loss is 0.1, the exponential loss is 0.41, and the logistic loss is 0.34;

If the desired value is $f(x)$=1, and the actual output is $h(x)$=-0.2, the Hinge loss is 1.2, the exponential loss is 1.22, and the logistic loss is 0.798.



We may also define u using the difference between$f(x)$ and $h(x)$.

Supervised, semi-supervised, and unsupervised learning (1)

- Supervised learning: If teacher signals or labels are available for training data.
- Un-supervised learning: If teacher signals are not available.
- Semi-supervised learning: If part of the signals are available.

Supervised, semi-supervised, and unsupervised learning (2)

- Teacher signals can be provided in different forms.
  - Correct answers for all input patterns.
    - Most informative, often used for pattern recognition.
  - Reward or penalty
    - The learner must learn what is the correct answer for each input pattern, to achieve a high score.
    - This is commonly known as **reinforcement learning**.
  - Goodness (fitness) of the current hypothesis
    - Each learner knows how good it is, and
    - Many learners can work together to find a good learner, through information exchange, or through self-improvement.
    - This is commonly known as **evolutionary learning, or meta-heuristic-based learning** in general.

Supervised, semi-supervised, and unsupervised learning (3)

- When there is no teacher signal at all, we need to partition the feature space into several disjoint clusters, and patterns in each cluster should share some common properties.
- This is in general a chicken-and-egg problem:
  - Define the clusters first, and then divide the space.
  - Divide the space first, and then define the clusters.
- The k-means algorithms is a heuristic algorithm for resolving the dilemma.
- Using different “similarity” measures, we can obtain different results → Some results may not be consistent with our expectation.

Supervised, semi-supervised, and unsupervised learning (4)

- When we have many un-labeled data, we can first define the “structure” of the feature space roughly based on un-supervised learning, and then use the labeled data to define (calibrate) the label of each cluster.
- This is also a heuristic based on the observation that “probability similar patterns have the same label”.
- In big data analytics, each datum may have multiple labels. Algorithms proposed for single label data are certainly not enough. → further study needed!

Parametric and non-parametric learning (1)

- Parametric learning: If each hypothesis in the hypothesis space is defined by a set of parameters.
  - Example 1: Similar data can be generated following a Gaussian distribution in the feature space. The mean and standard deviation can be used as parameters to determine this group of data.
  - Example 2: A neural network with a given structure is defined by its weights, and the weights are the parameters.
- The point is to find the best set of parameters to fit given training data.

Parametric and non-parametric learning (2)

- Non-parametric learning: If the hypotheses do not depend on a certain number of parameters.
  - Example 1: A nearest neighbor classifier using all training data cannot be defined by a “small” set of parameters, especially when the number of data is large, and changing.
  - Example 2: Support vector machine (SVM) is similar to a neural network in structure, but the number of support vectors depends on the training set size. So SVM is non-parametric approach.

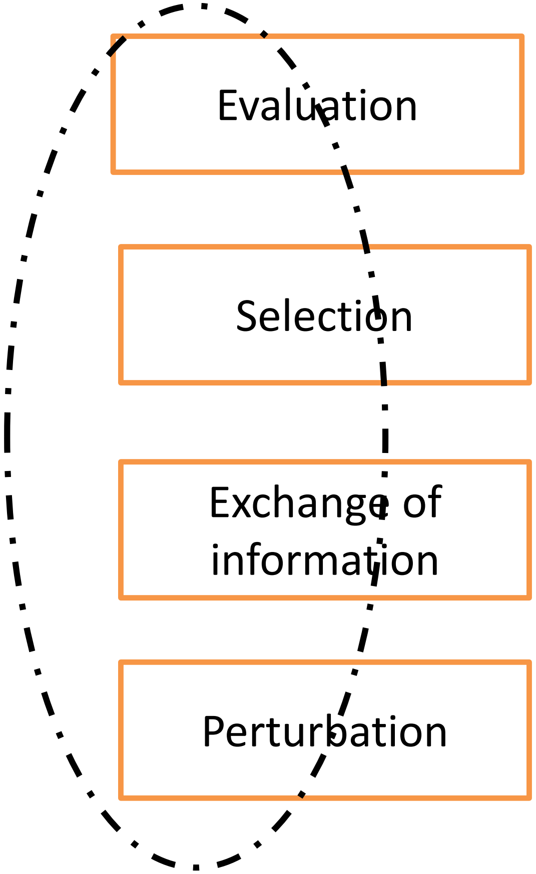
Online and off line learning

- Online learning:
  - Update the learner using newly observed data.
  - Do not use the data all at once.
  - → May obtain a good learner efficient by starting from a small training set.
  - → Suitable for learning with mobile devices.
- Offline learning:
  - Train the learner using all data.
  - Can obtain a better learner.
  - → Need more computing power for learning.
  - → Suitable for learning with strong platforms.

Mini-batch learning is a good compromise.

Evolutionary or population-based learning (1)

- Typical evolutionary algorithms include genetic algorithm (GA), evolutionary programing (EP), genetic programming (GP), evolution strategy (ES), etc.
- One important advantage of these algorithms is that they can find both structure and parameters together.



Evolutionary or population-based learning (2)

- In recent years, many other meta-heuristic algorithms have been proposed.
- Examples include particle swarm optimization (PSO), differential evolution (DE), etc.
- These algorithms can be adopted to machine learning because machine learning is nothing but an optimization problem.
- After finding a good solution, we may improve the search path (or the learning process) using some other meta-heuristic algorithm (e.g. ant colony optimization) → learning of learning.

Reinforcement learning (1)

- Reinforcement learning (RL) is important for “strategy learning”. It is useful for robotics, for playing games, etc.
- The well-known alpha-GO actually combined RL with deep learning, and was the first program that defeated human expert Go-players.

nature
THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE
At last — a computer program that can beat a champion Go player PAGE 484
ALL SYSTEMS GO
SONGBIRDS À LA CARTE
SAFEGUARD TRANSPARENCY
WHEN GENES GOT 'SELFISH'

Reinforcement learning (2)

- In RL, a learner is called an agent. The point is to take a correct “action” for each environment “situation”.
- If there is a teacher who can tell the correct actions for all situations, we can use supervised learning. In RL, we suppose that the teacher only “rewards” or “punishes” the agent under some (not all) situations.
- RL can find a “map” (a Q-table) that defines the relation between the situation set and the action set, so that the agent can get the largest reward by following this map.

Reinforcement learning (3)

- To play a game successfully, the computer can generate many different situations, and find a map between situation set and action set in such a way to win the game (with a high probability).
- Thus, even if there is no human opponent, a machine can improve its skill by playing with itself, using RL.
- Of course, if the machine has the honor to play many games with human experts, it can find the best strategy more "efficiently" without generating many "impossible" situations; or find good computer game players more acceptable to human.

Deterministic and statistic learning (1)

- Given a hypothesis space $\mathcal{H}$, we can find the best (in some given criterion) hypothesis deterministically or statistically.
- In deterministic learning, we usually assume that all functions are defined in a high dimensional Euclidean space, and do not use "probability" explicitly.
- For example, in the case we want to find a neural network, we can use some method proposed in the context of ***mathematical programming*** (e.g. the well known BP algorithm).
- Generally speaking, basis function-based methods are also deterministic.

Deterministic and statistic learning (2)

- In most cases, however, it is natural to assume that the data are generated by following some probability distribution (e.g. Gaussian, or combination of several Gaussians).
- Instead of finding a deterministic function, it is natural to find the probabilities such as
  - Given a pattern x, the probability that x belongs to a certain class,
  - given a class, the probability that x is observed,
  - and so on.
- Based on these probabilities, we may make some “recommended” decisions, instead of telling yes or no.

Homework

- Machine learning algorithms can also be divided into “multi-label” learning and “single-label” learning.
- Examples:
  - Given a street view image, we can assign many labels to this image (e.g. road, cars, human, …)
  - Given a piece of news, we can assign it into different categories (e.g. international, economic, trade war, etc.)
- Do you have any idea to conduct multi-label learning?