6.891 Machine learning and neural networks

Instructor: Prof. Tommi Jaakkola
tommi@ai.mit.edu
NE43-735

TA: Jason Rennie
jrennie@ai.mit.edu
NE43-733

Course administrivia

- Lectures TR 2.30-4pm (in this room)
- Recitations WF 1-2pm (in this room)
- Midterm (15%) and a final (30%)
- 6 bi-weekly problem sets (30%)
- final project (25%)
- The course satisfies the AI requirement for Area II graduate students

  Additional information in the handout...

Topics

Classification, regression
Neural networks, back-propagation
Generalization, regularization
Feature selection, feature induction
Combining multiple models, additive models, boosting
Support vector machines, kernels
Complexity, VC-dimension
Model selection, structural risk minimization
Density estimation, parametric, non-parametric
Clustering, agglomerative, mixture models
Hidden markov models, forward-backward algorithm, EM
Bayesian networks, graphs, estimation, model selection
Markov random fields, dynamic Bayesian networks
Time series prediction, independent component analysis

Learning

- Some terminology: *learning* vs. *adaptation*
  - *Adaptation* is temporary and "resets" itself
    - e.g., brightness adaption of eyes
  - *Learning* involves a *persistent* change

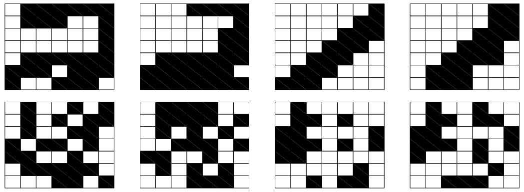
Learning cont'd

- No learning takes place without **assumptions!**
- Assumptions/biases come in many flavors
  - representation: how information is stored in the observed examples such as images
  - qualitative relations: independence

    etc.

Representation and assumptions

- Example: image labeling



- A change in the representation that nevertheless preserves the relevant information can preclude learning

Machine learning

- Machine learning is everywhere...

  speech recognition, medical diagnosis, image/text retrieval, commercial software, ...
- It is not just *deduction* but *induction* of rules and properties from examples to facilitate prediction and decision making
- Research in this area tries to understand why, how, and under what conditions learning takes place
- Strong ties to statistical inference/estimation (Why?)

Boldly goes where angels fear to tread...

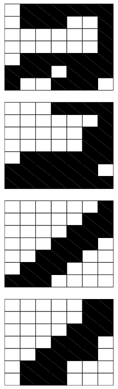
Machine learning problems

- Types of learning problems:
  1. Supervised learning (with a teacher)
  2. Unsupervised learning (no teacher, objective)
  3. Reinforcement learning (receive a cost associated with an action)
- This is not a complete nor "precise" list
  - e.g., active learning

Supervised learning

Example: digit recognition (8x8 binary digits)

binary digit | target label



"2"

"2"

"1"

"1"

. . .   . . .

- We wish to learn the mapping from digits to labels

Supervised learning

- We are given set of *input examples* $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, where

$$\mathbf{x}_i = (x_{i1} \ldots x_{id})^T, \;\; d = dim(\mathbf{x}_i),$$

  and corresponding *outputs* $\{y_1, \ldots, y_n\}$.
- Our task is to learn a *mapping or function* $f : \mathcal{X} \to \mathcal{Y}$ from inputs $\mathbf{x} \in \mathcal{X}$ to outputs $y \in \mathcal{Y}$ such that

$$y_i \approx f(\mathbf{x}_i), \;\; i = 1, \ldots, n \qquad (1)$$

  n is the number of *training examples*.

  For any problem there is a simple solution that is wrong

Supervised learning

The input/output spaces may be discrete or continuous

- **Classification** (qualitative)

  Continous/discrete inputs, discrete outputs
  - (binary pixel values) $\to$ digit label
  - (gray level pixel values) $\to$ digit label
- **Regression** (quantitative)

  Continous/discrete inputs, continuous outputs
  - (interest rate) $\to$ stock price
  - (stock($t-1$), stock(t)) $\to$ stock($t+1$)

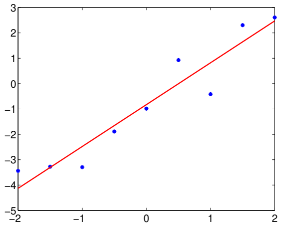
Supervised learning

- Given a set of training examples $\{(\mathbf{x}_1, y_1) \ldots, (\mathbf{x}_n, y_n)\}$, we want to learn a mapping $f : \mathcal{X} \to \mathcal{Y}$ such that

$$y_i \approx f(\mathbf{x}_i), \;\; i = 1, \ldots, n \qquad (2)$$

- We need to *parameterize* the set of functions (Why?)

  For example: $f(x) = ax + b$, where we learn the parameters a and b.



- BUT: how did we "fit" the function to the training examples?

Supervised learning

- We define a *loss function* $\text{Loss}(y, f(x))$ that measures how much our predictions deviate from the given answers

$$y_i \approx f(\mathbf{x}_i), \;\; \text{Loss}(y_i, f(\mathbf{x}_i)), \;\; i = 1, \ldots, n \qquad (3)$$

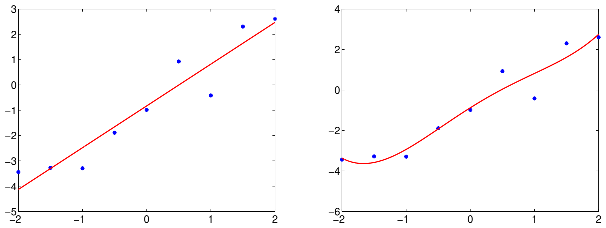
For example: $\text{Loss}(y_i, f(\mathbf{x}_i)) = (y_i - f(\mathbf{x}_i))^2$.

- To "fit" the parameters such as the linear coefficients a and b in $f(x) = ax + b$, we can minimize the *training error/loss*:

$$\sum_{i=1}^{n} \text{Loss}(y_i, f(\mathbf{x}_i)) \qquad (4)$$

Supervised learning

- Of course, the more parameters we have in the function, the smaller training error that we can achieve

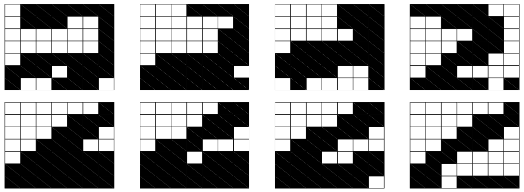


f = linear $\qquad$ $f = 4^{th}$ order polynomial

- Overfitting
- Generalization
- Model selection

Unsupervised learning

The digits again...



- **Clustering**: group together "similar" digits
- **Density estimation**: model the population of digits

Clustering

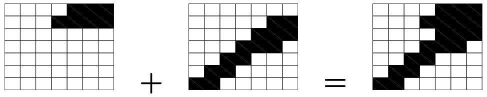
- Pairwise clustering: for any two examples $\mathbf{x}_i$ and $\mathbf{x}_j$ we need a *similarity* or *dissimilarity* measure

$$D_{ij} = D(\mathbf{x}_i, \mathbf{x}_j), \;\; i, j = 1, \ldots, n \qquad (5)$$

- There are several problems here:
  1. How to choose the dissimilarity measure?
  2. How to exploit the pairwise measure to group examples together?
  3. How to assess the quality of the resulting grouping?

Density estimation

- We wish to construct a *generative* probability model of the examples in a given population
  For example:



- A closely related problem is how to best *compress* a given set of examples.

Reinforcement learning

- There's no teacher but we receive (possibly noisy) feed-back about how good our actions are

  Take action $a \quad \rightarrow \quad$ receive $Cost(a, \omega)$

  Often we do not know the cost function $Cost(a, \omega)$ but only observe its values after taking actions.

- The goal is to select the action that minimizes the expected cost

$$E_{\omega}\left\{ Cost(a, \omega) \right\} \approx \frac{1}{n}\sum_{i=1}^{n} Cost(a, \omega_i) \tag{6}$$

- This becomes much more interesting when our actions actually *affect* the state of the environment... games, planning, etc.

Summary

Some of the key concepts we have discussed:

- Assumptions & representation in learning
- The role of probability in machine learning
- Overfitting, generalization, model selection
- Pairwise clustering, generative density estimation
- Minimizing expected cost