Machine Learning for Problem Solving

B.H. Juang
Georgia Institute of Technology
9/12/2011

What is Machine Learning?

- *Wikipedia*: Machine learning, a branch of **artificial intelligence**, is a scientific discipline concerned with the design and development of algorithms that allow **computers to evolve behaviors based on empirical data**, such as from sensor data or databases. **Oh, really!?**
- Tom M. Mitchell: A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P*, if its performance at tasks in *T*, as measured by *P*, improves with experience *E*.

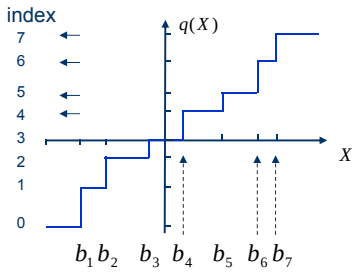
E, T and P

- Experience:
  - empirical data, e.g., labeled (or even un-labled) data, graded paper/HW, cause-effect relationship, etc.
- Task:
  - Understanding/characterization of observed data to allow, say, inference of properties – "knowledge discovery"
  - **Solving the problem directly** – for example, LMS algorithm, ..
- Performance:
  - Statistics: models/parameters for interpretation of properties
  - Various errors: error probability in recognition, type I & II in detection, mean squared error (MSE), …
  - Others

Data-driven Problem Solving

- Problem solving is an important mental and high-order cognitive process in parallel with "problem finding" and "problem shaping"
- More specific than "**knowledge discovery**"
- Premise and focus of this talk: machine learning from ***empirical data*** to solve problems beyond curve fitting
- Example:
  - Quantization of signal, an age old problem
  - Pattern recognition

Quantizer – A Simple Scalar Example



$$b_i \le X < b_{i+1} \Rightarrow q(X) = \hat{x}_i$$

$\hat{x}_i$ is the reconstruction value for X in region $[b_i, b_{i+1})$

Average distortion with known pdf $f_X(x)$:

$$D = \sum_{i=0}^{N-1} \int_{b_i}^{b_{i+1}} (x - \hat{x}_i)^2 f_X(x)\, dx$$

Problem Statement:

Given a source which puts out random values X, for a prescribed number of levels N, find the set of boundary values $\{b_i\}_{i=0}^{N}$ and the set of reconstruction values $\{\hat{x}_i\}_{i=0}^{N-1}$ so as to minimize the average distortion D.

Problem Analysis

- Surprisingly difficult
  - Cases of known distribution;
  - Cases of unknown distribution;
  - Mismatched/inaccurate estimate of distribution
- Known distribution:
  - Extremely low rate (only a few levels): use numerical method to find the solutions

$$D = \sum_{i=0}^{N-1} \int_{b_i}^{b_{i+1}} (x - \hat{x}_i)^2 f_X(x)\, dx$$

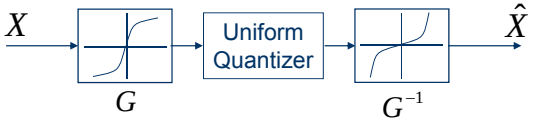
  - Extremely high rate: Bennett (1948), Lloyd (1957), Zador (1963), Gersho (1979)

Gersho's Conjecture

- Rate is high enough that there exists a continuous function $\lambda(x)$ that represents the density of quantizer points $\{\hat{x}_i\}_{i=0}^{N-1}$
- All cells are well approximated asymptotically by polytopes which are scaled, rotated or translated versions of a single tessellating (space filling) convex polytope with minimum normalized moment of inertia, conforming to the quantizer point density function

Impact of Data Distribution – 1 D Case

- If data distribution is known, it is "possible" to match the quantizer to the distribution



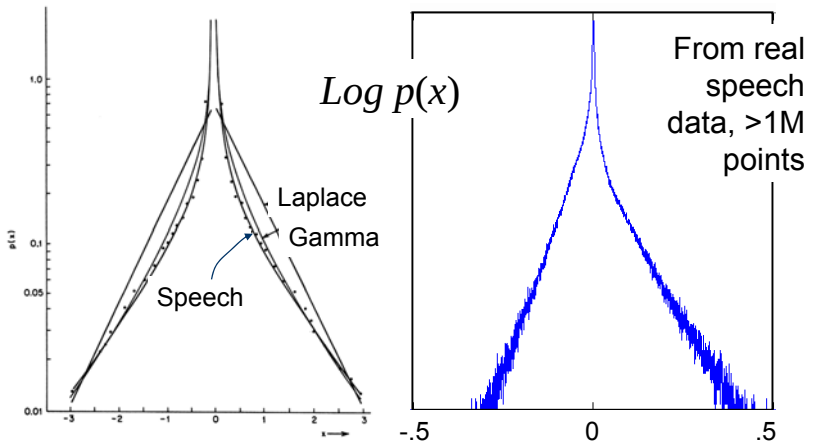
G: Monotonic non-linearity, a compressor

G^{-1}: Monotonic non-linearity, an expander

Gersho arrived at $\lambda(x) \propto G'(x) \propto f_X^{1/3}(x)$

Is it true then that the quantization problem is a distribution estimation problem?

Estimated Density of Speech



Bottomline

- Data distribution is not always simple in form, particularly in multi-dimensional space
- Estimate of distribution is not always accurate
- Almost impossible to relate quality of distribution estimate to quantizer performance
- Even if it is accurate, the conjecture does not entirely hold; so, quantization design based on distribution is interesting but hardly useful

Look at the paradigm of Stu Lloyd

Lloyd's Non-uniform Quantizer

- Lloyd's methods
  - Lloyd's method I (Lloyd algorithm) – use of data to "train" the quantizer, bypassing estimation of the probability density function (which is a daunting task itself)
  - Lloyd's method II
    - Reported in 1957
    - Rediscovered by Max in 1960; led to the name Lloyd-Max quantizer

Optimal Non-uniform Quantizer

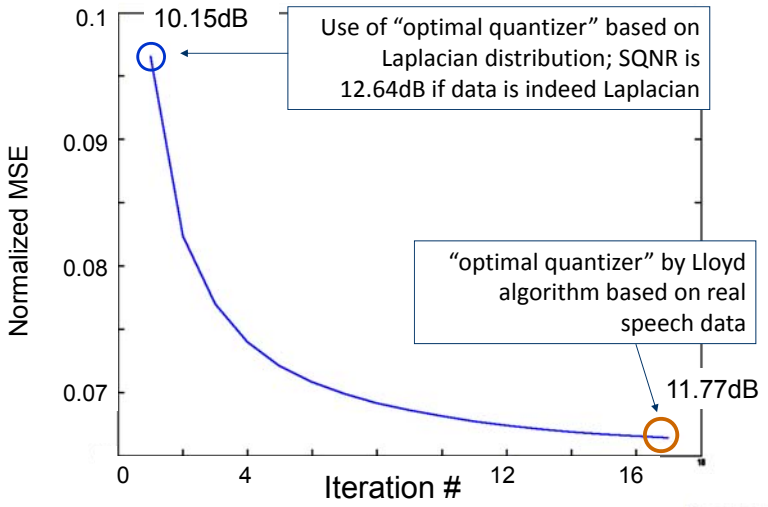
- Two coding principles embedded in empirical hill-climbing design – the Lloyd Algorithm

  #1: The best representation value in a partitioned region is one that minimizes average distortion in that region, i.e, its centroid

  #2: The best region an input value is assigned to is the one whose centroid is closest to the input value.

In every hill-climbing step, the average distortion is reduced.

Lloyd Algorithm – 3-bit Scalar



Questions to Ponder?

- Is the quantizer design problem the same as distribution estimation problem?
- Which problem are we trying to solve?
- What is the significance of distribution estimation in quantizer design in the real world?
- Can we say "the Lloyd quantizer is asymptotically optimal for any source"?
- By-product: Between the data set (as a non-parametric representation) and a parametric distribution, are we seeing an alternative?

Pattern Recognition – Bayes Theory

Problem Statement:

To identify/categorize an unknown observation X as one of M classes (of events or species) with minimum ***probability of error***

- **Conditional Error**: given X, the cost associated with deciding that it is an i^{th} class event

$$R(i \mid X) = \sum_{j=1}^{M} e_{ij} P(j \mid X)$$

- **Expected Error**:

$$\mathcal{E} = \int R(C(X) \mid X) p(X) dX$$

How do we design and implement $C(X)$?

Statistical Pattern Recognition

Essence of statistical methods (vs. heuristics/others):

- ***Learning from data***
- "Consistency" with formulation of error probability

With the need of posteriors, the problem is traditionally transformed into **distribution estimation**:

Given a set of design samples $\{X_i, y_i\}_{i=1}^{N}$ with known class identity, where y_i is the class label of X_i, estimate

$$P(j \mid X), \quad j = 1, 2, \cdots, M$$

or

$$P(X \mid j) \text{ and } P(j), \quad j = 1, 2, \cdots, M$$

to implement the Maximum a Posteriori (for 0-1 error) decision to achieve Bayes minimum error.

Issues in "Bayesian" Methods

- $P(j \mid X)$ is modeled/approximated by $\tilde{P}(j \mid X;\Lambda)$
  - ➔ choice of distribution form

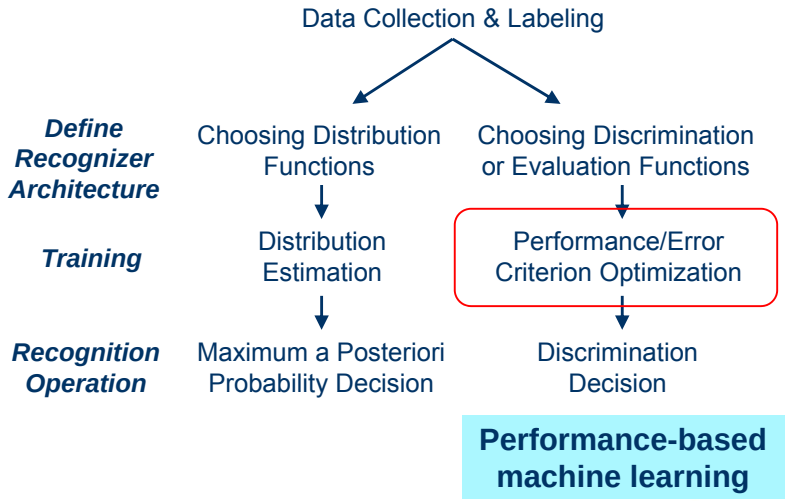
  (Wrong choice will not lead to optimal performance.)
- Availability of design sample
  - ➔ effort in data collection

  (Insufficient data means bad estimation results.)
- Estimation of parameter Λ
  - ➔ mathematical objective and method

  (How to do best when distribution form may be wrong and data may be insufficient?)

Is the pattern recognition problem the same as the distribution estimation problem?

Two Paths towards Pattern Recognition



Performance-based Approach

- Do not equate pattern recognition problem with distribution estimation problem
- Write out the performance measure explicitly for use as optimization objective (e.g., MCE)
- In pattern recognition, it is the empirical error rate

$$\mathcal{L}_0(\Lambda) = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} l_j(X_i;\Lambda)\mathbf{1}(X_i \in C_j) \quad \text{over} \quad \Omega = \{X_i\}_{i=1}^{N}$$

- If necessary, use a smooth approximation for ease in optimization
- Empirical error rate is an unbiased estimate of the error probability – i.e., **matched objective**

Back to Basics – Classification Error

Discrimination Function: $g_m(X;\Lambda)$ for class m

choose discriminant functions with form as close to the true *a posteriori* probability as possible

Error : ---------➔ $X \in C_j$ but $g_j(X;\Lambda) \neq \max_m g_m(X;\Lambda)$

$C_j = \{X \mid j_X = j\}$; $j_X = X$'s label

Empirical Error Rate :

$$\mathcal{L}_0(\Lambda) = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} l_j(X_i;\Lambda)\mathbf{1}(X_i \in C_j) \quad \text{over} \quad \Omega = \{X_i\}_{i=1}^{N}$$

Training set

where $l_j(X_i;\Lambda) = \begin{cases} 1, & \text{if } X_i \in C_j \text{ and } j \neq \arg\max_m g_m(X;\Lambda) \\ 0, & \text{otherwise} \end{cases}$

Expected Error Rate: $\mathcal{L}(\Lambda) = E_X\left\{\sum_{j \in I_M} l_j(X;\Lambda)\mathbf{1}(X \in C_j)\right\}$

Synthesizing a Smooth Error Function

- **Smoothed Misclassification Measure**

$$d_j(X) = -g_j(X;\Lambda) + \left\{ \frac{1}{M-1} \sum_{m,m\neq j} [g_m(X;\Lambda)]^\eta \right\}^{1/\eta}$$

where the η norm $\left\{ \frac{1}{M-1} \sum_{m,m\neq j} [g_m(X;\Lambda)]^\eta \right\}^{1/\eta} \approx \left\{ \int [g_m(X;\Lambda)]^\eta d\psi(j) \right\}^{1/\eta}$

is supported on integer set $\{m \mid m \neq j, m \in \{1,2,\cdots,M\}\}$ via discrete measure $\psi(j)$.

As η approaches infinity, i.e. $\| g_m(X;\Lambda) \|_\eta \to \| g_m(X;\Lambda) \|_\infty$

$$\| g_m(X;\Lambda) \|_\eta \to \| g_m(X;\Lambda) \|_\infty = \max g_m(X;\Lambda) \quad \text{over } \psi(j).$$

- **Smoothed Error Function**

$$l_j(X;\Lambda) = f_j\left(d_j(X;\Lambda)\right)$$

where $f_j\left(d_j(X;\Lambda)\right) = \frac{1}{1+e^{-\alpha_j d_j + \beta_j}}$ turns d_j into a smoothed error count.

Generalized Probabilistic Descent Algorithm

MCE/GPD

Empirical Error Rate:

$$\ell_0(\Lambda) = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} l_j(X_i;\Lambda) \mathbf{1}(X_i \in C_j)$$

$$\cong E_X\{l(X;\Lambda)\} = \ell(\Lambda)$$

Probabilistic Descent Algorithm:

$$\Lambda_{n+1} = \Lambda_n - \varepsilon_n U \nabla l(X, \Lambda_n)$$

where U is a positive definite matrix, ε_n is positive real, and

$$l(X;\Lambda) = \sum_{j=1}^{M} l_j(X;\Lambda) \mathbf{1}(X \in C_j)$$

Convergence $\Lambda_n \to \Lambda^*$ almost surely if $\sum \varepsilon_n = \infty$, and $\sum \varepsilon_n^2 < \infty$.

Performance Comparison

Connected Digit Recognition using hidden Markov Models

	Accuracy (%)			
	ML		MCE/GPD	
	Digit	String	Digit	String
Independent self test				
UCS 16-digit	97.8	78.3	99.8	98.0
Mall91 10-digit	95.0	76.3	99.4	97.0
TeleTravel 10-digit	95.1	81.9	99.9	99.4
combined	96.5	79.6	99.7	98.5
UCS-II 5-digit	98.0	91.3	99.1	96.5

Many similar results have been reported.

Let's Pause and Ponder …

- Data distribution and statistics give us insights about the source of information, but …
- Do they impose undue limitations in our search for problem solutions?
- Examples discussed:
  - In quantization problems, solutions based on distribution estimation, even if exist, do not quite imply minimum average distortion
  - In pattern recognition or decision problems, best curve fitting (or statistical modeling) does not necessarily lead to minimum error probability

MLSP TC Chair's (Adali) Slide

Cognitive information processing represents a major paradigm shift in learning

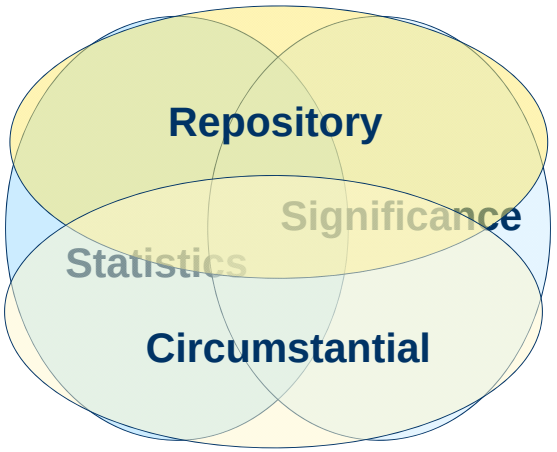
A dynamic system is called *cognitive* if it exhibits all four cognitive properties:

- Perception-action cycle, which produces information gain about the environment, obtained from one cycle to the next
- Memory, which predicts the consequences of action on/in the environment
- Attention, which is responsible for the allocation of available resources
- Finally, intelligence provides the basis for decision-making whereby intelligence choices are made in the face of environmental uncertainties

An Intelligent & Cognizant Mind Must

- Possess
  - Knowledge of statistics
  - Knowledge of significance
    Ex.: "I stop to rest." vs. "I stop the rest."
- Be able to maintain and retrieve knowledge
  - Repository: long-term, global and invariant
  - Circumstantial: references that need to be tracked
- Be able to construct hypothesis
  - An important element in reasoning; related to ***attention***
  - For exploration of solution or need for confirmation

Knowledge Support of Intelligence



Probability and Intelligence

- Statistics (or probability model) is organized information, but intelligence has more…
  - You don't solve a problem simply by giving the most "frequent" answer.
- Information has varying ***significance*** in each problem ***context***
- How do we incorporate ***significance*** into measure of information, beyond probability and uncertainty?

We believe

Frequency of occurrence $\neq$ Significance

Comprehension & Sense of Significance

- Consider the following three sentences:
  - **The storm, appearing as a giant white smudge over the Northeast on radar maps, seemed to land hardest in New York City**
  - **The storm, appears in giant sludge over the Northeast on radar, land hardest in New York City**
  - **The storm, appears as a giant white smudge on maps, seemed to land hardest in the sea.**

Storm over Northeast hardest in New York City

Incorporating Significance in Decision

- Hard to take it into account in the framework of axiomatic probability theory (σ-algebra)
- Error cost is one essential way and place to incorporate the notion of significance into the decision theory framework
- But what is its impact on statistical learning, particularly when we know we lack the precise knowledge of the distributions? (Further departure from statistics?)

Back to Performance Based Framework

- Machine learning for information identification needs an exact copy of the decision policy embedded and executed in the learning process
- Such embedding must be carried out on a per token basis, because what is important is **not just if the decision policy will commit an error but also how the particular error on the token is committed**
- MCE, as a design objective, can be extended to allow this embedding of significance (expressed via error analysis)

Modeling of Conditional Cost

Representing the conditional cost by approximation:

1. If we are confident that our knowledge of the *a posteriori* distribution is not seriously flawed:

$$\tilde{P}(j \mid X;\Lambda) \approx P(j \mid X) \;\Rightarrow\; R_i(X;\Lambda) \approx \sum_{j \in I_M} e_{ij} \tilde{P}(j \mid X;\Lambda)$$

2. If we are not so sure about the posteriors, one alternative is to model the conditional cost directly (as opposed to modeling the posteriors)

$$R_i(X;\Lambda) \approx R(i \mid X) = \sum_{j \in I_M} e_{ij} P(j \mid X)$$

To allow parameterization of system performance for optimization

Discriminant Function & Empirical Cost

$g_i(X;\Lambda) = \exp\{-R_i(X;\Lambda)\}$ monotonic function of R

The (approx.) MC rule: $C(X) = i_X = \arg\max_i g_i(X;\Lambda)$

i_X : recognizer decision on X

j_X : true category index of X

Token-based error cost: $l(X;\Lambda) = e_{i_X j_X}$ for all $X \in \Omega = \{X_n\}_{n=1}^N$

$$\min \mathcal{E} = \min \int R(C(X) \mid X) p(X) dX = \min \int e_{i_X j_X} p(X) dX$$

Empirical cost: $\int e_{i_X j_X} p(X) dX \rightarrow L = N^{-1} \sum_{X \in \Omega} e_{i_X j_X}$

or $L(\Lambda, \Omega) = N^{-1} \sum_{X \in \Omega} \sum_{i \in I_M} \sum_{j \in I_M} e_{ij} 1[j_X = j] 1\left\{ i = \arg\max_k g_k(X;\Lambda) \right\}$

Connected Digit ASR Results

Cost matrix:

$$e_{ij} = \begin{cases} 1, & i \neq j, i \neq 11, j \neq 4 \\ 0, & i = j \\ 10, & i = 11, j = 4 \end{cases}$$

$$[e_{ij}] = \begin{bmatrix} 0 & 1 & 1 & 10 & \cdots & 1 \\ 1 & 0 & 1 & 10 & \cdots & 1 \\ 1 & 1 & 0 & 10 & \cdots & 1 \\ \cdots & & \cdots & & \cdots & \\ 1 & 1 & 1 & 10 & \cdots & 1 \\ 10 & 10 & 10 & 10 & \cdots & 0 \end{bmatrix}$$

	Total Errors	"4" to others	Others to "oh"
Baseline	140	6	11
MCE (Uniform)	76	4	6
MCE (non-U)	67	1	4

Confusion Matrix (Baseline – ML)

	1	2	3	4	5	6	7	8	9	0	oh	DEL
One	*	0	0	0	0	0	0	0	0	0	**0**	0
Two	0	*	0	0	0	0	0	0	0	0	**4**	0
Three	0	3	*	0	0	0	0	0	0	0	**0**	0
Four	**1**	**0**	**0**	*	**0**	**0**	**0**	**0**	**0**	**0**	5	**0**
Five	0	0	0	0	*	0	0	0	0	0	**0**	0
Six	0	0	0	0	0	*	1	0	0	0	**0**	0
Seven	1	0	0	0	0	0	*	0	0	0	**1**	0
Eight	0	0	0	0	0	1	0	*	0	0	**0**	1
Nine	1	0	0	0	1	0	0	0	*	0	**1**	0
Zero	0	0	0	0	0	0	0	0	0	*	**0**	0
Oh	0	0	0	0	0	0	1	0	0	0	*	18
Ins	2	0	0	0	0	1	0	3	1	1	**92**	

Confusion Matrix (MCE)

	1	2	3	4	5	6	7	8	9	0	oh	DEL
One	*	0	0	0	0	0	0	0	0	0	**0**	0
Two	0	*	0	0	0	0	0	0	0	0	**4**	0
Three	0	3	*	0	0	0	0	0	0	0	**0**	0
Four	**2**	**0**	**0**	*	**0**	**0**	**0**	**0**	**0**	**0**	2	**0**
Five	0	0	0	0	*	0	0	0	0	0	**0**	0
Six	0	1	0	0	2	*	1	0	0	0	**0**	0
Seven	1	0	0	1	0	0	*	0	0	0	**0**	0
Eight	0	0	0	0	0	1	0	*	0	0	**0**	4
Nine	2	0	0	0	1	0	0	0	*	0	**0**	0
Zero	0	0	0	0	0	0	0	0	0	*	**0**	0
Oh	1	0	0	0	0	0	1	1	0	2	*	25
Ins	1	0	0	0	0	0	0	2	1	1	**18**	

Confusion Matrix (MCE – Non-U Cost)

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**0**	**oh**	DEL
One	*	0	0	0	0	0	0	0	0	0	**0**	0
Two	0	*	0	0	0	0	0	0	0	0	**4**	0
Three	0	3	*	0	0	0	0	0	0	0	**0**	0
Four	**1**	**0**	**0**	*	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Five	0	0	0	0	*	0	0	0	0	0	**0**	0
Six	0	0	0	0	0	*	1	0	0	0	**0**	0
Seven	1	0	0	0	0	0	*	0	0	0	**0**	0
Eight	0	0	0	0	0	1	0	*	0	0	**0**	4
Nine	1	0	0	0	1	0	0	0	*	0	**0**	0
Zero	0	0	0	0	0	0	0	0	0	*	**0**	0
Oh	0	0	0	0	0	0	2	4	1	0	*	23
Ins	1	0	0	0	0	1	0	2	1	1	**14**	

Summary – Key Messages

- "Learning from data" needs focus:
  - Formulate "learning" to **solve problem directly** rather than through surrogate
  - Get to the bottom of the "problem" and start with the right objective of learning
- In pattern recognition, performance based methods prove advantageous
- Intelligence processing is new dimension in machine learning, requiring handling of "significance" beyond Hartley and Shannon's "information"; the "problem solving" approach must apply.