On-Line Learning for Active Pattern Recognition

Jong-Min Park, *Student Member, IEEE,* and Yu Hen Hu, *Senior Member, IEEE*

***Abstract*—An adaptive on-line learning method is presented to facilitate pattern classification using active sampling to identify the optimal decision boundary for a stochastic oracle with a minimum number of training samples. The strategy of sampling at the *current estimate* of the decision boundary is shown to be optimal compared to random sampling in the sense that the probability of convergence toward the true decision boundary at each step is maximized, offering theoretical justification on the popular strategy of category boundary sampling used by many query learning algorithms.**

I. Introduction

PATTERN recognition via active sampling can trace its roots to statistical experiment design where performing an experiment (acquiring one training sample) may incur significant cost.

A number of active learning strategies, based on the concept of optimal experimental design, as well as importance sampling have been reported [1]–[5]. References [1] and [2] focused on active learning for pattern classification applications, with a common heuristic to sample at or near the present estimate of the category boundary, using a justification that [1] and [2] the function approximation of the posterior probability is most uncertain near the category boundary.

In this letter, we examine the validity of this argument using a two-class pattern classification problem as an example. We show that the variance of the approximation error reaches its maximum at the *true* category boundary. However, the *present estimate* of the category boundary may not coincide with the true boundary unless convergence is reached.

Based on a stochastic oracle model, we show that the strategy of sampling at the *present estimate* of category boundary is optimal by using a perceptron-like learning algorithm. This result offers a direct theoretical justification of the "sample-at-current-boundary" strategy. Preliminary simulation results are consistent with what the algorithm predicts.

II. Problem Formulation

In a two-class pattern recognition problem, the feature vector $x \in \Re$ and the class label $C \in \{0, 1\}$ are random variables with conditional probability density function $f_{x|C}(x|C = i) = f_i(x)$, and prior probability $P(C = i) = \pi_i$, where $i = \{0, 1\}$.

Manuscript received March 27, 1996. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. V. J. Mathews.

The authors are with the Department of Electrical and Computer Engineering, University of Wisconsin, Madison, WI 53706-1691 USA (e-mail: jong-min@engr.wisc.edu).

Publisher Item Identifier S 1070-9908(96)07928-X.

We also denote the posterior probability that $C = i$ given x is

$$q_i(x) = P\{C = i|x\} = \frac{\pi_i f_i(x)}{\pi_0 f_0(x) + \pi_1 f_1(x)}, \quad i = 0, 1.$$

Since $q_0(x) = 1 - q_1(x)$, for simplicity we shall denote $q_1(x)$ by $q(x)$ in the rest of this letter.

A maximum *a posteriori* probability (MAP) classifier is a decision rule that determines that x is drawn from class $C = 1$ if $q(x) > 0.5$ and from class $C = 0$ if $q(x) < 0.5$.

The set of points $B = \{x|q_0(x) = q_1(x) = 1/2\}$ is called the decision boundary. In general, B may contain more than one point. In this work, we are mainly interested in applications where B contains exactly one point. This will be the case if, say, $q(x)$ is a monotonically increasing function of x, as we will assume in the sequel.

In conventional pattern recognition problem, a set of "training samples" are given so that the decision boundary w^* where $q(w^*) = 0.5$ can be estimated from these samples. In an active learning (also known as *query learning* [6]) problem formulation, the set of training samples are not given. Instead, a "learner" (the classification algorithm) will sample a feature vector x and present to an oracle (by performing an experiment or running a simulation) to learn the corresponding class label of x. In a two-class pattern recognition problem, this is equivalent to the evaluation of a function $y(x)$ at a specific value of x. The oracle will return $y(x) = 1$ or $y(x) = 0$ as the class label associated with x according to the posterior probability $P\{C = 1|x\} = q(x)$ and $P\{C = 0|x\} = 1 - q(x)$.

For $x \gg w^*$ (w^* is unknown to the learner) $q(x) \to 1$ and the oracle will most likely return $y(x) = 1$, while for $x \ll w^*$, it will most likely return $y(x) = 0$. For $x \approx w^*$, it is equally likely for the oracle to return $y(x) = 0$ or $y(x) = 1$. In other words, $y(x)$ is an oracle whose answer is probabilistic in nature. This property is quite distinct from previous works where an oracle is assumed to be deterministic [2], [5].

The goal of active learning in the pattern recognition problem setting is to find w^* with minimum number of queries to the oracle. Minimizing the number of queries not only reduces the cost associated with each experiment (query), but also expedites the convergence of the algorithm.

III. Minimum Error Active Learning

To devise a learning rule that learns the optimal decision boundary w^* using active learning, let us define a 0–1 loss function $L[y(x), f(w, x)] = [y(x) - f(w, x)]^2 = [y(x) - u(x - w)]^2$, where $u(x) = 1$ if $x > 0$ and $u(x) = 0$ if $x < 0$. That is, if $f(w, x)$ and $y(x)$ have the same value for a given

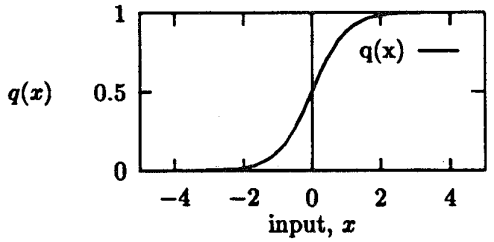

Fig. 1. Two-class problem, $q(x) = P\{Y = 1|z\}$.

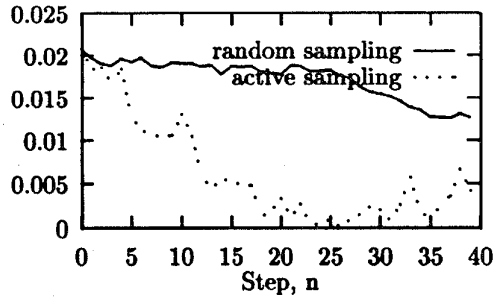

Fig. 2. $|w_n - w^*|$, average mean of active and random sampling.

x, then the loss is 0. Otherwise, the loss is 1. Then, a cost function as the conditional risk given x can be defined as

$$\begin{aligned} Cost(x) &= E\{[y(x) - f(w, x)]^2|x\} \\ &= P\{y(x) = 1|x\} \cdot [1 - u(x - w)]^2 \\ &\quad + P\{y(x) = 0|x\} \cdot [0 - u(x - w)]^2 \\ &= q(x) \cdot [1 - u(x - w)] + [1 - q(x)] \cdot u(x - w) \\ &= \begin{cases} 1 - q(x) & x > w \\ q(x) & x < w \end{cases} \end{aligned} \tag{1}$$

Integrate $Cost(x)$ over x, and we have the expected risk

$$\begin{aligned} R &= \int Cost(x)p(x)\,dx = \int_w^\infty P\{x \in C_0|x\}p(x)\,dx \\ &\quad + \int_{-\infty}^w P\{x \in C_1|x\}p(x)\,dx \\ &= P\{x \in C_0\} \int_w^\infty p\{x|x \in C_0\}p(x)\,dx \\ &\quad + P\{x \in C_1\} \int_{-\infty}^w p\{x|x \in C_1\}p(x)\,dx \\ &= P\{x \in C_0\}P\{x > w|x \in C_0\} \\ &\quad + P\{x \in C_1\}P\{x < w|x \in C_1\} \end{aligned} \tag{2}$$

This shows that the expected risk R is the probability of misclassification. Thus, for any fixed x, minimizing $Cost(x)$ will minimize the probability of misclassification.

Since $Cost(x)$ is not differentiable with respect to the decision boundary w, we use an adaptive formula similar to that of the classical perceptron learning algorithm

$$w_{n+1} = w_n - \{\epsilon[y(x_{n+1}) - 0.5]\} \tag{3}$$

where ϵ is the learning rate, also called step-size, and the new estimate of the boundary moves to the left or right by $\epsilon/2$ depending on the sample output $y(x_{n+1})$ at the next sampling of x_{n+1}. Note that $E[y(w^*)] = 0.5$.

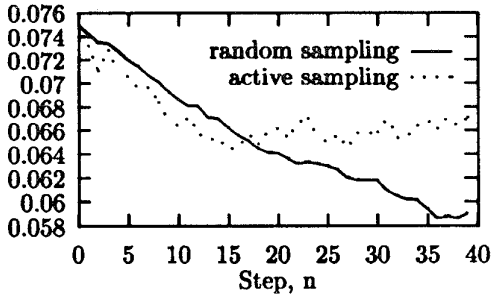

Fig. 3. $|w_n - w^*|^2$, average variance of active and random sampling.

From (3)

$$|w_{n+1} - w^*| = |w_n - w^*| \pm \frac{\epsilon}{2}. \tag{4}$$

The algorithm will move toward convergence in the present step if $|w_{n+1} - w^*| = |w_n - w^*| - \epsilon/2$.

Theorem 1: Given the *a posteriori* probability $q(x)$ of a two-class problem, the probability of moving away from the true boundary, $P\{|w_{n+1} - w^*| = |w_n - w^*| + (\epsilon/2)|x_{n+1}\}$, is minimized when $x_{n+1} \to w_n$. Moreover

$$P\left\{|w_{n+1} - w^*| = |w_n - w^*| - \frac{\epsilon}{2}\middle|x_{n+1} = w_n\right\} > 0.5. \tag{5}$$

Proof: Let

$$\begin{aligned} P_e &= P\left\{|w_{n+1} - w^*| = |w_n - w^*| + \frac{\epsilon}{2}\right\} \\ &= P\{y(x_{n+1}) = 0|w_n > w^*\} \cdot P\{w_n > w^*\} \\ &\quad + P\{y(x_{n+1}) = 1|w_n < w^*\} \cdot P\{w_n < w^*\}. \end{aligned} \tag{6}$$

The event $y(x_{n+1}) = 0$ given x_{n+1} and the event $w_n > w^*$ are independent, hence

$$\begin{aligned} P\{y(x_{n+1}) = 0|w_n > w^*\} &= P\{y(x_{n+1}) = 0\} \\ &= 1 - q(x_{n+1}) \end{aligned} \tag{7}$$

$$\begin{aligned} P\{y(x_{n+1}) = 1|w_n < w^*\} &= P\{y(x_{n+1}) = 1\} \\ &= q(x_{n+1}). \end{aligned} \tag{8}$$

Thus, (6) becomes

$$\begin{aligned} P_e &= [1 - q(x_{n+1})] \cdot P\{w_n > w^*\} \\ &\quad + q(x_{n+1})P\{w_n < w^*\}. \end{aligned} \tag{9}$$

If $w_n > w^*$, one would want to minimize $1 - q(x_{n+1})$ term by choosing $x_{n+1} \geq w_n$, since $1 - q(x_{n+1}) < 0.5$ for $w^* < w_n \leq x_{n+1}$. If we chose $x_{n+1} < w_n$, it may be $x_{n+1} < w^*$, in which case $1 - q(x_{n+1}) > 0.5$, thus actually increasing P_e and not being optimal.

On the other hand, if $w_n < w^*$, one would choose $x_{n+1} \leq w_n$ to minimize the term $q(x_{n+1})$, since $q(x_{n+1}) < 0.5$ for $x_{n+1} \leq w_n < w^*$. If we chose $x_{n+1} > w_n$, it may be $x_{n+1} > w^*$, in which case $q(x_{n+1}) > 0.5$, again increasing P_e.

Since one has no prior knowledge of whether $w_n > w^*$ or $w_n < w^*$, we opt to use the min-max criterion to minimize the maximum probability value of P_e regardless of whether $w_n > w^*$ or $w_n < w^*$. In particular, we note that when $w_n < w^*$, choosing $x_{n+1} < w_n$ will run into the risk of

$x_{n+1} < w^*$, which implies $P_e = 1 - q(x_{n+1}) > 0.5$. Only for $x_{n+1} \geq w_n$ is it guaranteed that $P_e < 0.5$. Similarly, when $w_n < w^*$, only for $x_{n+1} \leq w_n$ does it guarantee that $P_e < 0.5$. Taking the intersection of the two sets, $\{x_{n+1} \geq w_n\}$and $\{x_{n+1} \leq w_n\}$, one concludes that $x_{n+1} = w_n$ is the only solution that guarantees that $P_e < 0.5$.

The (6) now becomes

$$\begin{aligned} P\left\{|w_{n+1} - w^*| = |w_n - w^*| + \frac{\epsilon}{2}\right\} \\ = [1 - q(w_n)] \cdot P\{w_n > w^*\} \\ + q(w_n) \cdot P\{w_n < w^*\} \\ < [1 - q(w^*)] \cdot P\{w_n > w^*\} \\ + q(w^*) \cdot P\{w_n < w^*\} = 0.5. \end{aligned} \tag{10}$$

Since, for any fixed x_{n+1}

$$\begin{aligned} P\left\{|w_{n+1} - w^*| = |w_n - w^*| - \frac{\epsilon}{2}\right\} \\ = 1 - P\left\{|w_{n+1} - w^*| = |w_n - w^*| + \frac{\epsilon}{2}\right\} > 0.5 \end{aligned} \tag{11}$$

this proves (5). ■

This theorem establishes that, with a min-max criterion, the optimal active learning strategy for the two-class pattern classification problem is to sample at the *current estimate* of the category boundary w_n, and not the unknown theoretic decision boundary w^*.

Due to space limitation, the convergence of this proposed learning algorithm will be discussed formally in a separate paper. Briefly, we note that the update formula (3) is Markovian in nature because the next state w_{n+1} depends only on the previous state w_n and current input x_{n+1}. Since the probability of moving toward the decision boundary (see (7) and (8)) is a function of x_{n+1} only, it is a time-invariant Markov chain with a continuous state space of w. As the Markov chain converges to its limiting distribution $P_{w_\infty}(w)$ $\lim_{t\to\infty} \Pr\{W(t) \leq w\}$, so will the proposed algorithm. Furthermore, the corresponding limiting density function will peak at $w = w^*$.

IV. Simulation

We have conducted a simulation to validate the theorem developed in this letter. We assume that the data sampled are drawn from two Gaussian distribution with means equal to 1 ($C0$) and -1 ($C1$), respectively, and the variance equal to 1. This yields a posterior probability (Fig. 1)

$$q(x) = \frac{e^{-(x-1)^2/2}}{e^{-(x-1)^2/2} + e^{-(x+1)^2/2}}.$$

For each x, a random variable r, uniformly distributed over [0, 1], is generated. If $r < q(x)$, then $y(x) = 1$, otherwise $y(0) = 0$. The initial estimate of w^*, w_0, is chosen randomly over the interval $[-0.5, 0.5]$.

Two methods are compared. One chooses $x_{n+1} = w_n$, the sample-at-the-boundary method discussed in this letter. The other, a random sampling method, draws x_{n+1} randomly from the interval $[-0.5, 0.5]$. The step-size is $\epsilon = 0.05$ in both methods.

One-hundred random initial estimates from $[0.5, -0.5]$ have been drawn. For each of them, 100 trial runs are performed, with each trial lasting for 100 steps.

Figs. 2 and 3 show the average of the mean $|w_n - w^*|$ and the average of the variance of the boundary estimates $|w_n - w^*|^2$ at each step.

Fig. 2 shows that the active sampling converges faster than the random sampling with the same update step ϵ.

However, when the estimate approaches near the true boundary (the mean nears zero), the variance of the active sampling becomes higher than that of the random sampling. This is expected, since ϵ now plays a crucial role of oscillation when $|w_{n+1} - w^*| < \epsilon$; that is, the update step-size ϵ is too large for w_{n+1} to fall at the true boundary exactly. This can be seen in Fig. 2, where $|w_n - w^*|$ approaches zero near the point where the variance becomes higher.

V. Conclusion

Active learning in a stochastic environment reflects the method of estimating the learning model given existing samples, querying new samples that may optimize the estimation process, and then iterating this process.

It is theoretically shown that sampling near the boundary is the optimal way for active learning in a stochastic environment. An example is shown to demonstrate the formulation for a two-class problem with active querying near the boundary points.

References

[1] D. A. Cohn, "Neural network exploration using optimal experiment design," *Adv. Neural Inform. Processing Syst.*, vol. 6, 1994.

[2] J.-N. Hwang, J. J. Choi, S. Oh, and R. J. Marks, II, "Query-based learning applied to partially trained multilayer perceptrons," *IEEE Trans. Neural Networks*, vol. 2, no. 1, pp. 131–136, Jan. 1991.

[3] Y. Kabashima and S. Shinomoto, "Incremental learning with and without queries in binary choice problems," in *Proc. Int. Joint Conf. Neural Networks*, vol. 2, 1993, pp. 1637–1640.

[4] D. MacKay, "Information-based objective functions for active data selection," *Neural Computat.*, vol. 4, no. 4, pp. 590–604, 1992.

[5] M. Plutowski and H. White, "Active selection of training examples for network learning in noiseless environments," Tech. Rep. CS91-180, Univ. California, San Diego, CA, Feb. 1990.

[6] D. Angluin, "Queries and concept learning," *Machine Learn.* vol. 2, pp. 319–342, 1988.