To appear in Proceedings IEEE Workshop on Neural Networks for Signal Processing '95

ACTIVE LEARNING THE WEIGHTS OF A RBF NETWORK

Kah-Kay Sung and Partha Niyogi

Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139, USA

Abstract

We describe a principled strategy to sample functions optimally for function approximation tasks. The strategy works within a Bayesian framework and uses ideas from optimal experiment design to evaluate the potential utility of new data points. We consider an application of this general framework for active learning the weight coefficients of a Gaussian Radial Basis Function (RBF) network. We also derive some sufficiency conditions on the learning problem for which there are analytical solutions to the data sampling procedure.

1 Introduction

In most classical formulations of learning from examples, the data (examples) are assumed to be randomly drawn and presented to the learner. This is the case for a variety of situations ranging from network models (Rumelhart et. al. [5], Poggio and Girosi [4]), PAC (Valiant [8]) frameworks, and classical pattern recognition. In this sense, the learner is a *passive* recipient of information about the target function. In contrast, one could consider a learner that plays a more *active* role in collecting its examples. By judiciously selecting examples instead of allowing for possible random sampling, *active learning* techniques can conceivably have faster learning rates and better approximation results than passive learning methods.

Using ideas from Optimal Experiment Design (Fedorov [2]), we have, in an earlier work (Sung and Niyogi [7]) formulated a general procedure for sampling an unknown target function at points in its domain. In this paper, we consider an application of this general framework to encompass the active learning of the coefficients of a Radial Basis Function (RBF) network (Poggio and Girosi [4]). More generally, our solution allows us to handle parameter estimation problems for function classes that are linear in their parameters. In this paper, our contributions are:

1. The application of an Optimal Experiment Design framework for estimating the weights of RBF-type function classes, and the analytical derivation of an optimal sampling strategy for parameter estimation of such classes from data. This represents a significant improvement over existing work on toy function classes, like step functions considered in

(Sung and Niyogi [7]). Recently, Sollich [6] has applied a similar optimal experiment design framework for estimating the weights of a single perceptron unit. To the best of our knowledge, this analysis has not been done before on RBF-type networks.

2. Empirical comparisons of the data requirements for active and passive learning of RBF-type function classes.

3. Development of sufficiency conditions on the learning problem for which an active data selection sequence can be analytically derived and precomputed within our optimal experiment design framework.

We now provide a brief statement of our problem, our solution for the active sampling strategy, and a glimpse of the empirical simulations.

2 The General Formulation

We adopt a Bayesian formulation for active example selection. Specifically, we can pose the problem as follows: Let $\mathcal{D}_n = \{(x_i, y_i)|i = 1, \ldots, n\}$ be a set of n data points sampled from an unknown target function, $g_{\mathbf{t}}$, possibly in the presence of noise. Given a class of approximation functions, $\mathcal{F}$, where each $f \in \mathcal{F}$ has prior probability $P_{\mathcal{F}}(f)$, one can use regularization techniques to approximate $g_{\mathbf{t}}$ from $\mathcal{D}_n$ (in a MAP sense) by means of a function $\hat{g} \in \mathcal{F}$. We want a strategy to determine at what input location one should sample the next data point, (x_{n+1}, y_{n+1}), in order to obtain the "best" possible Bayes optimal approximation of the unknown target function $g_{\mathbf{t}}$ with our concept class $\mathcal{F}$.

We approach the active example selection problem in two stages:

1. **Define the notion of a "best" possible Maximum A-Posteriori (MAP) approximation for an unknown target function.** We propose an optimality criterion for evaluating how well a solution approximates an *unknown* target function. Our active learning goal is to find a solution $\hat{g} \in \mathcal{F}$ that best approximates the unknown target function in terms of the optimality criterion.

2. **Formalize mathematically the task of determining the best input location to sample next.** We express the active learning optimality criterion as a cost function to be minimized. The task of choosing the next data point becomes one of minimizing the cost function with respect to the next sample's input location.

Earlier work by Cohn [1] and MacKay [3] have tried using similar optimal experiment design techniques [2] to collect data that maximizes information about an unknown target function. Our work differs from theirs in two respects. First, we use a different, and perhaps more general, optimality criterion for evaluating solutions to an unknown target function, based on a measure of function uncertainty that incorporates both bias and variance components of the total *output* generalization error. In contrast, MacKay and Cohn consider only the variance component in *model parameter* space. Second, we also show that our active example selection strategy requires fewer training examples than passive methods to learn a target function to a given degree of accuracy.

2.1 The Formulation

We begin by establishing an optimality criterion for measuring the quality of an approximation function with respect to an unknown target. Recall that our active learning goal is to find solutions that best approximate an unknown target according to our proposed optimality criterion. From the optimality criterion, we can derive a scheme for measuring a new example's utility in terms of how well the example steers the learner toward the goal, and an accompanying computational procedure for selecting the next training example.

Let $g_{\mathbf{t}}$ be a target function that we want to estimate by means of an approximation function $\hat{g} \in \mathcal{F}$. If the target function $g_{\mathbf{t}}$ were known, then one popular measure of how well (or badly) $\hat{g}$ approximates $g_{\mathbf{t}}$ is their *Integrated Squared Difference* (ISD) over an appropriate region of interest $\mathcal{X}$:

$$\delta(\hat{g}, g_{\mathbf{t}}) = \int_{x \in \mathcal{X}} (g_{\mathbf{t}}(x) - \hat{g}(x))^2 dx. \tag{1}$$

In most function approximation tasks, the target $g_{\mathbf{t}}$ is unknown, so we clearly cannot express the quality of a learning result, $\hat{g}$, in terms of $g_{\mathbf{t}}$. We can, however, compute an *expected* integrated squared difference (EISD) between $\hat{g}$ and its *unknown* target, $g_{\mathbf{t}}$, by treating the unknown target $g_{\mathbf{t}}$ as a random variable in the approximation function concept class $\mathcal{F}$. Notice that the distribution of $g_{\mathbf{t}}$ is simply its a-posteriori likelihood given $\mathcal{D}_n$, the n data points seen so far: i.e. $P(g_{\mathbf{t}}|\mathcal{D}_n) \propto P_{\mathcal{F}}(g_{\mathbf{t}})P(\mathcal{D}_n|g_{\mathbf{t}})$. We shall use the EISD as a criterion for evaluating the quality of an approximation result, $\hat{g}$, for an *unknown* target function $g_{\mathbf{t}}$:

$$\begin{aligned}
\mathtt{EISD}(\hat{g}, g_{\mathbf{t}}) &= E_{g_{\mathbf{t}} \in \mathcal{F}}[\delta(\hat{g}, g_{\mathbf{t}})|\mathcal{D}_n] \\
&= \int_{g_{\mathbf{t}} \in \mathcal{F}} P(g_{\mathbf{t}}|\mathcal{D}_n)\delta(\hat{g}, g_{\mathbf{t}}) dg_{\mathbf{t}} \\
&= \int_{g_{\mathbf{t}} \in \mathcal{F}} P_{\mathcal{F}}(g_{\mathbf{t}})P(\mathcal{D}_n|g_{\mathbf{t}})\delta(\hat{g}, g_{\mathbf{t}}) dg_{\mathbf{t}}.
\end{aligned}$$

Let $\hat{g}_n$ be the approximation result given n data points, $\mathcal{D}_n$. Our learning goal is to minimize $EISD(\hat{g}_n, g_{\mathbf{t}})$ for each new n, and so a reasonable active example selection strategy would be to choose the next input location, x_{n+1}, that minimizes $EISD(\hat{g}_{n+1}, g_{\mathbf{t}})$. For a given x_{n+1}, we can predict the new EISD that results from sampling our next data point there as follows:

Suppose we also know the target output value y_{n+1} at the given x_{n+1}. Our new data set would then be $\mathcal{D}_{n+1} = \mathcal{D}_n \cup \{(x_{n+1}, y_{n+1})\}$, and the new EISD between $g_{\mathbf{t}}$ and its new estimate $\hat{g}_{n+1}$ is given by $E_{\mathcal{F}}[\delta(\hat{g}_{n+1}, g_{\mathbf{t}})|\mathcal{D}_{n+1}]$, where the new estimate, $\hat{g}_{n+1}$, can be derived from the new data set, $\mathcal{D}_{n+1}$ via regularization. In reality, we do not know the actual value of y_{n+1}, but we can derive for it the following conditional probability distribution:

$$\begin{aligned} P(y_{n+1}|x_{n+1}, \mathcal{D}_n) &= \int_{f \in \mathcal{F}} P(y_{n+1}|x_{n+1}, f) P(f|\mathcal{D}_n) df \\ &\propto \int_{f \in \mathcal{F}} P(\mathcal{D}_n \cup \{(x_{n+1}, y_{n+1})\}|f) P_{\mathcal{F}}(f) df. \end{aligned}$$

So, for a given next location to sample, x_{n+1}, we can compute the *expected* value of the resulting EISD measure:

$$\begin{aligned} \mathcal{U}(\hat{g}_{n+1}|x_{n+1}, \mathcal{D}_n) &= \mathrm{E}_{y_{n+1}}[\mathrm{E}_{\mathcal{F}}[\delta(\hat{g}_{n+1}, g_{\mathbf{t}})|\mathcal{D}_{n+1} \cup (x_{n+1}, y_{n+1})]|x_{n+1}, \mathcal{D}_n] \\ &= \int_{y_{n+1} \in \Re} P(y_{n+1}|x_{n+1}, \mathcal{D}_n) \\ &\qquad E_{\mathcal{F}}[\delta(\hat{g}_{n+1}, g_{\mathbf{t}})|\mathcal{D}_n \cup (x_{n+1}, y_{n+1})] dy_{n+1}. \qquad (2) \end{aligned}$$

Clearly, the optimal location to sample next is the location that minimizes the above cost function:

$$\hat{x}_{n+1} = \arg\min_{x_{n+1}} \mathcal{U}(\hat{g}_{n+1}|x_{n+1}, \mathcal{D}_n).$$

The important questions at this stage are:

1. Can the equations above be analytically solved to yield a specific sampling procedure?
2. Does the sampling procedure allow one to learn the target function with fewer examples? Does it reduce the sample complexity?

We answer these questions in the context of estimating the weight parameters of a Radial Basis Function network. In particular, we are able to derive analytical solutions to the equations above and compare the sample complexities of active and passive learning.

3 Radial Basis Function Networks

3.1 The Function Class $\mathcal{F}$

We consider a class of d-dimensional Gaussian Radial Basis Functions with K fixed centers where the ith basis function, $\mathcal{G}_i$, has a fixed covariance of $\mathcal{S}_i$, and a center $\mathbf{c}_i$. The coefficients to be learned are denoted by $\mathbf{a} = [a_1 \; a_2 \; \ldots \; a_K]^{\mathtt{T}}$. Specifically, an arbitrary function in this class can be represented as:

$$\begin{aligned} g_{\mathbf{a}}(\mathbf{x}) &= \sum_{i=1}^{K} a_i \mathcal{G}_i(\mathbf{x}) \\ &= \sum_{i=1}^{K} a_i \frac{1}{(2\pi)^{d/2}|\mathcal{S}_i|^{1/2}} \exp(-\frac{1}{2}(\mathbf{x}-\mathbf{c}_i)^{\mathtt{T}} \mathcal{S}_i^{-1}(\mathbf{x}-\mathbf{c}_i)) \end{aligned}$$

The priors on the class $\mathcal{F}$ are obtained by putting a normal distribution with covariance $\Sigma_{\mathcal{F}}$ on the coefficients $\mathbf{a}$. Thus, for an arbitrary $g_{\mathbf{a}}$:

$$P(g_{\mathbf{a}}) = P(\mathbf{a}) = \frac{1}{(2\pi)^{K/2}|\Sigma_{\mathcal{F}}|^{1/2}} \exp(-\frac{1}{2}\mathbf{a}^{\mathtt{T}}\Sigma_{\mathcal{F}}^{-1}\mathbf{a}).$$

Finally, the learner has access to noisy data of the form $\mathcal{D}_n = \{(\mathbf{x}_i, y_i = g_{\mathbf{t}}(\mathbf{x}_i) + \eta) : i = 1, \ldots, n\}$, where $g_{\mathbf{t}}$ is the unknown target function, and η is zero-mean additive Gaussian noise with variance σ_η^2. This leads to an expression for $P(\mathcal{D}_n|g_{\mathbf{a}})$ for every candidate $g_{\mathbf{a}} \in \mathcal{F}$ that is of the form:

$$P(\mathcal{D}_n|g_{\mathbf{a}}) \propto \exp\left(-\sum_{i=1}^{n} \frac{(y_i - g_{\mathbf{a}}(\mathbf{x}_i))^2}{2\sigma_\eta^2}\right).$$

The weight parameters, $\hat{\mathbf{a}}$, of the MAP solution ($g_{\hat{\mathbf{a}}}$) to the learning problem can then be recovered as follows:

$$\hat{\mathbf{a}} = \mathcal{C}_n \mathbf{b}$$

where:

$$\begin{aligned} \mathbf{b} &= \frac{1}{\sigma_\eta^2}[\sum_{i=1}^{n} y_i \mathcal{G}_1(\mathbf{x}) \; \sum_{i=1}^{n} y_i \mathcal{G}_2(\mathbf{x}_i) \; \ldots \; \sum_{i=1}^{n} y_i \mathcal{G}_K(\mathbf{x}_i)]^{\mathtt{T}} \\ \mathcal{C}_n^{-1} &= \Sigma_{\mathcal{F}}^{-1} + \Sigma_n^{-1} \end{aligned} \tag{3}$$

and Σ_n^{-1} is a $K \times K$ matrix whose $(i,j)^{\mathtt{th}}$ element is: $\frac{1}{\sigma_\eta^2}\sum_{k=1}^{n} \mathcal{G}_i(\mathbf{x}_k)\mathcal{G}_j(\mathbf{x}_k)$.

This is the solution the learner proposes on the basis of the data set *irrespective* of how the data points are selected. We now provide an active strategy for selecting the data optimally.

3.2 Active Sampling

Recall that we are interested in answering the two questions at the end of Section 2.

1. For the RBF class considered here, it is possible to show that an analytical expression exists for Equation 2, the cost function to minimize that yields the optimal location to sample next:

$$\mathcal{U}(\hat{g}_{n+1}|\mathbf{x}_{n+1}, \mathcal{D}_n) \propto |\mathcal{C}_{n+1}|$$

$\mathcal{C}_{n+1}$ has the same form as $\mathcal{C}_n$ of Equation 3, and depends on the previous data sample locations $\{\mathbf{x}_i : i = 1, \ldots, n\}$, the weight priors $\Sigma_{\mathcal{F}}$, σ_η and $\mathbf{x}_{n+1}$. When minimized over $\mathbf{x}_{n+1}$, we get $\hat{\mathbf{x}}_{n+1}$ as the maximum utility location where the active learner should next sample the unknown target function.

2. Is it indeed the case that the active strategy outperforms the passive one? To investigate this we randomly generated 5000 target RBF functions with 8 fixed, arbitrarily chosen centers. The priors on the weights $\mathbf{a}$ are provided by $\Sigma_{\mathcal{F}} = \mathbf{I}_K$. For each target RBF function, we collected data according to the active strategy as well as the passive (uniformly random) strategy for varying number of data points. Figure 1 shows the mean error rate (i.e., the integrated squared difference, Equation 1, between the *actual* target function and its MAP *estimate*, averaged over the 5000 different target functions) as a function of the number of data points. Notice that the active strategy has a lower mean error rate than the passive for the same number of examples and is particularly true for small number of data. The left graph shows a situation where the learner has knowledge of the true priors on $\mathcal{F}$. The right graph is for a case where the learner has the following incorrect prior assumptions: (a) the true centers of the target RBF's are slightly different from the values that the learner assumes them to be, and (b) the learner's priors on the weights ($\Sigma_{\mathcal{F}} = 0.9\mathbf{I}_K$) are different from that on the target class ($\Sigma_{\mathcal{F}} = \mathbf{I}_K$). Despite these incorrect prior assumptions, the active strategy still outperforms the passive case.

4 Sufficiency Conditions for Pre-Computing a Data Sampling Sequence

It is noteworthy that for learning RBF weights, the new optimal sample location, $\hat{\mathbf{x}_{n+1}}$, does not depend on the y_i data values previously observed but only on the $\mathbf{x}_i$ values sampled. Thus, if the learner were to collect n data points, it can pre-compute the exact sequence of n points at which to sample from the start, even before receiving any data from the target function. This behavior has been observed by MacKay [3] for an active example selection strategy that minimizes only a model parameter variance cost function. For such cost functions, any class of approximation functions that are linear in their model parameters would exhibit such behavior.

In our framework, we minimize an output uncertainty cost function that includes both bias and variance terms. The following theorem provides sufficiency conditions on the learning problem for which our active learning for-

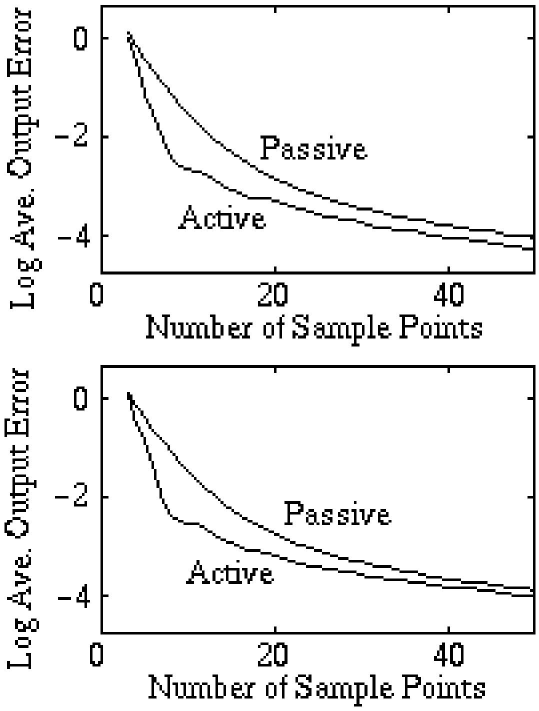

Figure 1: Comparing the active and passive mean error rates for learning an unknown RBF target function with 8 fixed centers. **Top:** The learner uses the same priors on model parameters as the process that randomly generates the unknown target functions. **Bottom:** The learner uses slightly different weight priors and has centers that are slightly displaced from the true centers.

mulation leads to a data selection strategy that does not depend on previously observed y_i data values.

Theorem 1 *Suppose $\mathcal{F}$ is a class of real-valued functions parameterized by $\mathbf{a} \in \Re^k$. On the basis of a data set $\mathcal{D}_n = \{(\mathbf{x}_i, y_i) : i = 1, \ldots, n\}$, let the MAP solution to the learning problem be given by $\hat{\mathbf{a}} = \arg\min_{\mathbf{a} \in \Re^k} P(g(\mathbf{a}|\mathcal{D}_n))$. Then the following three conditions guarantee that the choice of $\hat{\mathbf{x}}_{n+1}$ will be independent of the previously observed y_i's in $\mathcal{D}_n$.*

1. *$P(g(\mathbf{a})|D_n)$ can be expressed as $Q((\mathbf{a} - \hat{\mathbf{a}}(\mathcal{D}_n)), \{\mathbf{x}_i : i = 1 \ldots n\})$ where Q is some arbitrary function that does not depend on the data, $\mathcal{D}_n$. In other words, the y_i terms of $\mathcal{D}_n$ do not appear anywhere else in $P(g(\mathbf{a})|D_n) = Q((\mathbf{a} - \hat{\mathbf{a}}(\mathcal{D}_n)), \{\mathbf{x}_i : i = 1 \ldots n\})$ other than in $\hat{\mathbf{a}}$.*

2. $\mathcal{F}$ is linear in its parameters $\mathbf{a}$*, i.e.:* $g_{\mathbf{a}_1+\mathbf{a}_2}(\mathbf{x}) = g_{\mathbf{a}_1}(\mathbf{x}) + g_{\mathbf{a}_2}(\mathbf{x})$.

3. The prior distribution on model parameters $\mathbf{a}$ *has support* $\Re^k$.

5 Remarks and Conclusions

We have extended a previously developed Bayesian framework for active learning to the case of learning the weights of an RBF network. We derive a specific sampling strategy, and show how this strategy allows us to learn with fewer examples (alternatively, make smaller error with the same number of examples) than random (*passive*) sampling. This is an application of the optimal experiment design paradigm to function approximation and seems to bear promise for the design of more efficient learning algorithms.

References

[1] D. Cohn. A Local Approach to Optimal Queries. In D. Touretzky, editor, *Proc. of 1990 Connectionist Summer School*, San Mateo, CA, 1991. Morgan Kaufmann Publishers.

[2] V. Fedorov. *Theory of Optimal Experiments*, page 35. Academic Press, New York, 1972.

[3] D. MacKay. *Bayesian Methods for Adaptive Models.* PhD thesis, California Institute of Technology, Pasadena, CA, 1992.

[4] T. Poggio and F. Girosi. A Theory of Networks for Approximation and Learning. Technical Report AIM–1140, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1989.

[5] D. Rumelhart and J. McClelland. *Parallel Distributed Processing*, volume 1. MIT Press, Cambridge, Massachusetts, 1986.

[6] P. Sollich. Query Construction, Entropy, Generalization in Neural Network Models. *Physical Review E*, 49:4637–4651, 1994.

[7] K. Sung and P. Niyogi. Active Learning for Function Approximation. In *Advances in Neural Information Processings Systems 7*, San Mateo, CA, 1995. Morgan Kaufman.

[8] L.G. Valiant. A theory of learnable. *Proc. of the 1984 STOC*, pages 436–445, 1984.