CONF-9608144--1

Multiple Sensor Fusion Under Unknown Distributions†

Nageswara S. V. Rao
Center for Engineering Systems Advanced Research
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831-6364

Submitted to *Workshop on Information/Decision Fusion: Applications to Engineering Problems*, August 7-9, 1996, Washington, D.C.

†Research sponsored by the Engineering Research Program of the Office of Basic Energy Sciences, of the U.S. Department of Energy, under Contract No. DE-AC05-96OR22464 with Lockheed Martin Energy Research Corp.

DISCLAIMER

Multiple Sensor Fusion Under Unknown Distributions

Nageswara S. V. Rao
Center for Engineering Systems Advanced Research
Oak Ridge National Laboratory
Oak Ridge, TN 37831
raons@ornl.gov

Abstract

In a system of N sensors, the sensor S_i, $i = 1, 2 \ldots, N$, outputs $Y^{(i)} \in \Re$, according to an unknown probability distribution $P_{Y^{(i)}|X}$, corresponding to input $X \in \Re$. A training n-sample (X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n) is given where $Y_i = (Y_i^{(1)}, Y_i^{(2)}, \ldots, Y_i^{(N)})$ such that $Y_i^{(j)}$ is the output of S_j in response to input X_i. The problem is to design a fusion rule $f : \Re^N \mapsto \Re$, based on the sample, such that the expected square error

$$I(f) = \int [X - f(Y))]^2 dP_{Y|X} dP_X,$$

where $Y = (Y^{(1)}, Y^{(2)}, \ldots, Y^{(N)})$, is minimized over a family of functions $\mathcal{F}$. Let f^* minimize $I(.)$ over $\mathcal{F}$; in general, f^* cannot be computed since the underlying distributions are unknown. We consider sufficient conditions based on smoothness and/or combinatorial dimensions of $\mathcal{F}$ to ensure that an estimator $\hat{f}$ satisfies

$$P[I(\hat{f}) - I(f^*) > \epsilon] < \delta$$

for any $\epsilon > 0$ and $0 < \delta < 1$. We present two methods for computing $\hat{f}$ based on feedforward sigmoidal networks and Nadaraya-Watson estimator. Design and performance characteristics of the two methods are discussed, based both on theoretical and simulation results.

1 Introduction

Over the past decade, the area of sensor fusion has witnessed a tremendous growth due to: (a) an expanding application base that requires solutions to difficult fusion problems, and (b) advances in computational systems and methods that make it possible to process large volumes of data. The sensor fusion problems have particular relevance to engineering applications, where researchers realized fundamental limitations of single sensor systems. By employing multiple sensors: (i) replicated sensors can be employed for fault tolerance, and (ii) sensors of different modalities can be used to achieve tasks that cannot be performed by a single sensor. In either case, the fusion method must be designed carefully, since an inappropriate fuser can make the system worse than the worst individual sensor.

Several existing sensor fusion methods require either independence of sensor errors or closed-form analytical expressions for error densities. In the former case, a general majority rule suffices, while in the latter a fusion rule can be computed using Bayesian methods. Several popular distributed decision fusion methods belong to the latter class [5]. In engineering systems, however, independence can seldom be assured and, in fact, may not be satisfied. Also, the problem of obtaining the probability densities required by Bayesian methods can be more difficult than the fusion problem itself. Thus practical solutions to fusion problems must exploit the empirical data available from observation and/or experimentation. Recently, such "learning" methods that estimate fusion rules based on recent advances in empirical estimation and non-linear computational methods have been developed [18] within the framework of Probably and Approximately Correct (PAC) learning [31, 29]. These methods are suited for engineering systems where the sensor system is available for operation/experimentation.

Consider a system of N sensors such that corresponding to input $X \in \Re$, the sensor S_i, $i = 1, 2, \ldots, N$, outputs $Y^{(i)} \in \Re$ according to an *unknown* distribution $P_{Y^{(i)}|X}$. A training n-sample $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$ is given where $Y_i = (Y_i^{(1)}, Y_i^{(2)}, \ldots, Y_i^{(N)})$ and $Y_i^{(j)}$ is the output of S_j in response to input X_i. We consider the expected

square error

$$I(f) = \int [X - f(Y))]^2 p(Y|X)p(X)dYdX, \quad (1.1)$$

where $Y = (Y^{(1)}, Y^{(2)}, \ldots, Y^{(N)})$, to be minimized over a family of fusion rules $\mathcal{F}$, based on the given n-sample. For simplicity, we considered quadratic cost, but the approach is valid for general costs if suitable boundedness or smoothness conditions are satisfied (see Section 3). For convenience, in some parts of this paper, we may denote $\mathcal{F}$ by $\{f^\alpha\}_{\alpha\in\Lambda}$, where Λ is an abstract index set.

Let $f^* \in \mathcal{F}$ minimize $I(.)$. In general, f^* cannot be computed since the underlying distributions are unknown. Furthermore, since no restrictions are placed on the distributions, it will not be possible to infer f^* (with probability one) based on *only* a finite sample. We show that in several cases an estimator $\hat{f}$ can be computed, based on a sufficiently large sample, which satisfies

$$P[I(\hat{f}) - I(f^*) > \epsilon] < \delta, \quad (1.2)$$

where $\epsilon > 0$ and $0 < \delta < 1$. Eq. (1.2) states that the "error" of $\hat{f}$ is within ϵ of the optimal error (of f^*) with arbitrary high probability $1 - \delta$, given a sufficiently large sample. Such criteria have been extensively used in a number of machine learning and empirical estimation problems (see Vapnik [31] for more details). We estimate the sample size required to ensure (1.2) as a function of ϵ, δ, and the parameters of $\mathcal{F}$. We consider two types of conditions on $\mathcal{F}$ that enable us to ensure (1.2). The first type are based on geometric and combinatorial properties and the second type are based on smoothness properties. The geometric and combinatorial properties are based on recent developments in empirical processes [12, 13] and their applications to computational learning theory [31]. The smoothness conditions are the traditional ones used in nonlinear statistical estimators [15] coupled with computational properties of sigmoidal neural networks and Haar wavelets.

To put the above formulation in perspective, we now briefly discuss some related existing results. If the sensor error densities are known, several cases of the fusion rule estimation problem have been solved by methods not requiring the samples. Earlier work in this direction was done in the areas of pattern recognition (Chow [3]), political economy (Grofman and Owen [8]), and reliability (von Neumann [32]). The distributed detection problem based on probabilistic formulations has been extensively studied; see Dasarathy [5] (also the recent special issue [6]) for a comprehensive treatment. Most existing sensor fusion methods are based on maximizing a posteriori probabilities of hypotheses under a suitable probabilistic model. However, when the probability densities are unknown (or difficult to estimate) such methods are ineffective. One alternative is to estimate the density based on a finite sample. But, as illustrated in general by Vapnik [30], the density estimation is more difficult than the subsequent problem of estimating a function chosen from a family with bounded capacity or a suitable ϵ-cover.

The sensor fusion problem (1.1) under criterion (1.2) was first formulated in Rao [18] and was further developed in Rao [17, 20, 21]. The special case of decision fusion where $Y_i \in \{0, 1\}^N$ has been solved using majority rules [25, 23], empirical Bayesian rules [16, 24], and nearest neighbor rules [22].

The paper is organized as follows. Preliminaries are summarized in Section 2. In Section 3, we show that for a sufficiently large sample, the bound (1.2) can be satisfied under fairly general conditions. We then consider two computationally viable methods for fuser design based on neural networks and Nadaraya-Watson estimator in Sections 4 and 5, respectively. We present simulation examples and discussion of performance in Sections 6 and 7, respectively.

2 Preliminaries

We first review some basic definitions of smoothness of functions and their consequences. Let Q denote the unit cube $[0, 1]^N$ and $C(Q)$ denote the set of all continuous functions defined on Q. The modulus of smoothness of $f \in C(Q)$ is defined as

$$\omega_\infty(f; r) = \sup_{\|y-z\|_\infty < r,\ y,z\in Q} |f(y) - f(z)|$$

where $\| y - z \|_\infty = \max_{i=1}^{M} |y_i - z_i|$.

For $m = 0, 1, \ldots$, let Q_m denote a family of diadic cubes (Haar system) such that $Q = \bigcup_{J\in Q_m} J$, $J \cap J' = \emptyset$ for $J \neq J'$, and the N-dimensional volume of J, denoted by $|J|$, is 2^{-Nm}. Let $1_J(y)$ denote the indicator function of $J \in Q_m$: $1_J(y) = 1$ if $y \in J$, and $1_J(y) = 0$ otherwise. For given m, we define the map P_m on $C(Q)$ as follows: for $f \in C(Q)$, we have $P_m(f) = P_m f$ defined by

$$P_m f(y) = \frac{1}{|J|} \int_J f(z)dz$$

for $y \in J$ and $J \in Q_m$ [4]. Note that $P_m f : Q \mapsto [0, 1]$ is a discontinuous (in general) function which

takes constant values on each $J \in Q_m$. Consider the *Haar kernel* given by $P_m(y,z) = \frac{1}{|J|} \sum_{J \in Q_m} 1_J(y) 1_J(z)$ for $y, z \in Q$. Then an estimator for a density $p \in C(Q)$ based on n-sample is given by [4]

$$\hat{p}_{m,n}(y) = \frac{1}{n} \sum_{j=1}^{n} P_m(y, Y_j)$$

which can also be written in the form $\hat{p}_{m,n}(y) = \sum_{J \in Q_m} n(J) h_J(y)$ with $n(J) = \frac{1}{n} |\{j : Y_j \in J\}|$ and $h_J(y) = \frac{1}{|J|} 1_J(y)$. Note that a random variable is denoted by an uppercase letter (e. g. Y) and its deterministic version is denoted by the corresponding lowercase letter (e. g. y).

We now consider the covering properties of $\mathcal{F}$. Let S be a set equipped with a pseudometric d. The *covering number* $N(\epsilon, d, S)$ is defined as the smallest number of closed balls of radius ϵ, and centers in S, whose union covers S. Let $N_\infty(\epsilon, \mathcal{F}) = N(\epsilon, \| \cdot \|_\infty, \mathcal{F})$, where $\| f(y) \|_\infty = \sup_{y \in [0,1]^N} |f(y)|$.

The following cover size for the class of Lipschitz functions will be used in our sample size estimates.

Lemma 2.1 [27] *Let $\mathcal{F}_k = \{f_k : [0,1]^N \mapsto \Re\}$ denote the set of Lipschitz functions with Lipschitz constant k, i. e. for every $f \in \mathcal{F}_k$, we have $|f(y) - f(z)| \leq k \| y - z \|_\infty$. Then $N_\infty(\epsilon, \mathcal{F}_k) \leq \frac{2k}{\epsilon} 2^{\left\{\frac{k}{\epsilon}\left[\left(\frac{k}{\epsilon}-1\right)^{N-1}+1\right]\right\}}$.* □

We now present some basic definitions from Vapnik [30]. For family $\{A_\gamma\}_{\gamma \in \Gamma}$, $A_\gamma \subseteq A$, and for a finite set $\{a_1, a_2, \ldots, a_n\} \subseteq A$ we define

$$\Pi_{\{A_\gamma\}}(\{a_1, a_2, \ldots, a_n\}) = \{\{a_1, a_2, \ldots, a_n\} \cap A_\gamma\}_{\gamma \in \Gamma}.$$

We maximize this quantity with respect to the set $\{a_1, a_2, \ldots, a_n\}$ to obtain

$$\Pi_{\{A_\gamma\}}(n) = \max_{a_1, a_2, \ldots, a_n} |\Pi_{\{A_\gamma\}}(\{a_1, a_2, \ldots, a_n\})|.$$

The following critical identity is established in [30].

$$\Pi_{\{A_\gamma\}}(n) = \begin{cases} 2^n & \text{if } n \leq h \\ < 1.5 \frac{n^h}{h!} & \text{if } n > h. \end{cases}$$

Notice that for a fixed h, the right hand side increases exponentially with n until it reaches h and then varies as a polynomial in n with fixed power h. This quantity h is called the *VC dimension* of A_γ.

For a set of functions, the *capacity* is defined as the largest number h of pairs (x_i, y_i) that can be subdivided in all possible ways into two classes by means of rules of the form

$$\{\Theta[(x - f^\alpha(y))^2 + \beta]\}_{\alpha,\beta}$$

where

$$\Theta(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0. \end{cases}$$

Formally, the capacity of $\{f^\alpha(y)\}_{\alpha \in \Lambda}$ is the Vapnik-Chervonenkis dimension of the set of indicator functions

$$\{\Theta[(x - f^\alpha(y))^2 + \beta]\}_{(\alpha,\beta) \in \Lambda \times \Re}.$$

The following identity yields useful bounds on the simultaneous occurrence of events that may not be independent.

Lemma 2.2 [23] *Consider events A_i, $i = 1, 2, \ldots, N$ such that $P(A_i) \geq 1 - \delta_i$. Then we have*

$$P(A_1 \cap A_2 \cap \ldots \cap A_N) \geq \frac{1 - \delta_N}{2^{N-1}} - \sum_{i=1}^{N-1} \frac{\delta_i}{2^i}.$$

It is assumed that we consider very small values of δ_i's such that the right hand side of the equation in the above lemma is positive.

3 General Solutions for Fuser Design

In this section, we consider general conditions under which criterion (1.2) is met. Consider the empirical cost given by for any $f \in \mathcal{F}$

$$I_{emp}(f) = \frac{1}{l} \sum_{i=1}^{n} [X_i - f(Y_i)]^2 \qquad (3.1)$$

based on the sample $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$. To approximate $f^* \in \mathcal{F}$ that minimizes the expected error in (1.1), we minimize instead the empirical error in (3.1) to obtain a best empirical estimate $\hat{f}$. In order to ensure the (ϵ, δ)-condition in (1.2), two types of conditions are sufficient [30]:

(a) the capacity of $\mathcal{F} = \{f^\alpha\}_{\alpha \in \Lambda}$ is bounded;

(b) the error $I(.)$ is bounded, i.e., $\sup_{x,y,\alpha} (x - f^\alpha(y))^2 \leq \tau$ or the relative error is bounded as follows for some $p > 1$

$$\sup_\alpha \frac{\left[\int (x - f^\alpha(x))^{2p} P(x,y) dx dy\right]^{1/p}}{\int (x - f^\alpha(x))^2 P(x,y) dx dy} < \tau.$$

First we illustrate a very simple case.

Theorem 3.1 [18] *Consider that x and f^α take values from $\{0, 1\}$.*

(i) Given an n-sample, we have

$$P\left[I(\hat{f}) - I(f^*) > 2\kappa\right] < 9\frac{(2n)^h}{h!}e^{-\kappa^2 n/4}$$

where h is the capacity of $\mathcal{F}$.

(ii) If the hypothesis space is finite in that $\mathcal{F} = \{f^{\alpha_1}(y), f^{\alpha_2}(y), \ldots f^{\alpha_M}(y)\}$, given an n-sample, we have

$$P\left[I(\hat{f}) - I(f^*) > 2\kappa\right] < 2Me^{-2\kappa^2 n}.$$

In Part (i), notice that the upperbounds on the right hand side are products of two main factors: first one is n^h and the second one is $e^{-\kappa^2 n/4}$. For a fixed value of h, the latter decreases with the sample size n, and thus if n is chosen large enough the right hand side can be made equal to δ.

An example of infinite hypothesis class can be given by the set of all neural networks with a fixed number of nodes, where $f^\alpha(y)$ stands for a feedforward neural network with connection weight vector α (a more precise discussion is provided in the next section).

The following two theorems for the general case $f : \Re^N \mapsto \Re$ [18] follow from the results of [30].

Theorem 3.2 [18] *Consider that the error is bounded as* $\sup_{x,y,\alpha}(x - f^\alpha(y))^2 \leq \tau$.

(i) Then given an n-sample, we have

$$P[I(\hat{f}) - I(f^*) \geq 2\tau\kappa] \leq 9\frac{(2n)^h}{h!}e^{-\kappa^2 n/4}.$$

(ii) If the hypothesis space is finite in that $\mathcal{F} = \{f^{\alpha_1}(y), f^{\alpha_2}(y), \ldots, f^{\alpha_M}(y)\}$. Then given an n-sample, we have

$$P\left[I(\hat{f}) - I(f^*) > 2\tau\kappa\right] < 18Mle^{-\kappa^2 n/4}.$$

Theorem 3.3 [18] *Consider that the relative error be bounded such that for some $p > 1$ we have*

$$\sup_\alpha \frac{\left[\int (x - f^\alpha(x))^{2p} P(x,y)dxdy\right]^{1/p}}{\int (x - f^\alpha(y))^2 P(x,y)dxdy} < \tau.$$

(i) If $p > 2$, we have

$$P\left\{\frac{I(\hat{f}) - I(f^*)}{I(f^*)} > \frac{2\tau a(p)\kappa}{1 - \tau a(p)\kappa}\right\} < 24ne^{-\kappa^2 n/4}$$

where $a(p) = \left[\frac{(p-1)^{p-1}}{2(p-2)^{p-1}}\right]^{1/p}$.

(ii) If $1 < p \leq 2$, we have

$$P\left\{\frac{I(\hat{f}) - I(f^*)}{I(f^*)} > \frac{2\tau V_p(\kappa)}{1 - \tau V_p(\kappa)}\right\} < 24ne^{-\kappa^2 n^{2-(2/p)}/4}$$

where $V_p(\kappa) = \kappa\left[1 - \frac{\ln\kappa}{p^{1/(p-1)}(p-1)}\right]^{\frac{p-1}{p}}$.

The above theorems are derived based on uniform convergence of empirical measures to their expectations, which are available from the empirical process theory [12, 13] and its applications to machine learning [31, 9]. Results of this kind are available based on a number of characterizations of $\mathcal{F}$ such as pseudo-dimension, fat VC-dimension, etc., which can be used to obtain results along the lines of Theorems 3.1-3.3.

We now illustrate a well-known argument due to Vapnik [30] to facilitate the discussion of performance in Section 7. Consider a set of functions $\mathcal{G}$ such that the uniform convergence holds in the following manner:

$$P\left[\sup_{g\in\mathcal{G}} |I(g) - I_{emp}(g)| \geq \epsilon\right] < \delta(\epsilon, n, \mathcal{G}).$$

such that $\lim_{n\to\infty} \delta(\epsilon, n, \mathcal{G}) = 0$. Notice here that we explicitly show the dependence of δ on the precision ϵ, sample size l and the family of functions $\mathcal{G}$. Recall that g^* and $\hat{g}$ minimize $I(.)$ and $I_{emp}(.)$ respectively over $\mathcal{G}$. With probability $1 - \delta(\epsilon, n, \mathcal{G})$ we have $I_{emp}(g) \leq I(g) + \epsilon$ and $I(g) \leq I_{emp}(g) + \epsilon$ for all $g \in \mathcal{G}$. In particular we have $I(\hat{g}) \leq I_{emp}(\hat{g}) + \epsilon$ and $I_{emp}(g^*) \leq I(g^*) + \epsilon$. Noting that $I_{emp}(\hat{g}) \leq I_{emp}(g^*)$, we have

$$I(\hat{g}) \leq I_{emp}(\hat{g}) + \epsilon \leq I_{emp}(g^*) + \epsilon \leq I(g^*) + 2\epsilon$$

with probability $1 - \delta(\epsilon, n, \mathcal{G})$ or, equivalently, we have

$$P[I(\hat{g}) - I(g^*) > 2\epsilon] < \delta(\epsilon, n, \mathcal{G}).$$

Thus, the uniform convergence of empirical measures to their expectations implies the proximity of $\hat{g}$ to g^* in the above sense.

The results of this section do not directly yield methods to *compute* the required $\hat{f}$. However, they provide very useful guidelines for the conditions under which this empirical estimation procedure is a viable option. The problem of computing $\hat{f}$ in this general framework is computationally intractable; for example in the special case that $\mathcal{F}$ is set of feedforward neural networks with threshold hidden units, this problem is NP-complete even for simple architectures [2]. In

the next sections, we consider more restrictive cases where computational problems are easier to handle. We wish to emphasize that to be practically viable the solutions to the fusion rule must be computable with a low computational complexity.

4 Fusers Based on Feedforward Neural Networks

In this section we consider that $\mathcal{F}$ is given by feedforward neural networks with sigmoidal hidden nodes. These networks have been found to perform well in a number of difficult non-linear function estimation problems [28]. The results of this section are valid under the boundedness assumption that $x \in [-A, A]$, for $0 < A < \infty$, and $y \in [-B, B]$, for $0 < B < \infty$ (see [20] for details).

We consider a feedforward network with a single hidden layer of l nodes and a single output node. The output of the jth hidden node is $\sigma(b_j^T y + t_j)$, where $y \in [-B, B]^d$, $b_j \in \Re^d$, $t_j \in \Re$, and the nondecreasing $\sigma : \Re \mapsto [-1, +1]$ is called the *activation function*. The output of the network corresponding to input y is given by

$$f_w(y) = \sum_{j=1}^{l} a_j \sigma(b_j^T y + t_j)$$

where $w = (w_1, w_2, \ldots, w_{l(d+2)})$ is the *weight vector* of the network consisting of a_1, a_2, ..., a_l, $b_{11}, b_{12}, \ldots, b_{1d}$, ..., $b_{l1}, \ldots b_{ld}$, and $t_1, t_2, \ldots, t_l$. Let the set of *sigmoidal feedforward networks* with *bounded weights* be denoted by

$$\mathcal{F}_{\mathcal{W}}^{\gamma} = \{f_w : w \in [-W, W]^{l(d+2)}\} \qquad (4.1)$$

where $0 < \gamma < \infty$, and $\sigma(z) = \tanh(\gamma z), 0 < W < \infty$.

The function class $\mathcal{F}$ has an *envelope* F if $f(y) \leq F(y)$ for all y and every $f \in \mathcal{F}$. Let μ be a probability measure on $[-B, B]^d$, and $\mu(f^1) = \int\limits_{y \in [-B,B]^d} |f(y)| d\mu$ for a measurable function f. For a measure μ such that $\mu(F^1) < \infty$, we define the *covering number* $N_1(\epsilon, \mu, \mathcal{F}, F)$ to be the smallest cardinality for a subclass $\mathcal{F}^*$ of $\mathcal{F}$ such that

$$\min_{f^* \in \mathcal{F}^*} \mu(|f - f^*|^1) \leq \epsilon \mu(F^1)$$

for each $f \in \mathcal{F}$. Due to the boundedness of $\mathcal{F}$ we have $N_1(\epsilon/\mu(F), \mu, \mathcal{F}, F) \leq N_\infty(\epsilon/(2B)^d, \mathcal{F})$ since $\mu(|f - f^*|^1) \leq \int |f(X) - f^*(X)| d\mu \leq (2B)^N \parallel f - f^* \parallel_\infty$.

We show that the solutions to problem (1.1) can be found under requirement (1.2) by obtaining estimates for the required sample size. These estimates are based on three different parameters of the neural network. The first and second bounds are based on the Lipschitz properties of $f_w(y)$ with respect to w and y respectively. The third bound is based on the cover size estimate for $\mathcal{F}_{\mathcal{W}}^1$ derived by Lugosi and Zeger [10].

Lemma 4.1 [20] *For the class of feedforward neural networks $\mathcal{F}_{\mathcal{W}}^{\gamma}$ of Eq. (4.1), we have*

$$N_\infty(\epsilon, \mathcal{F}_{\mathcal{W}}^{\gamma}) \leq \frac{2\gamma W^2 l}{\epsilon} e^{\left\{ \frac{\gamma W^2 l}{\epsilon} \left[\left(\frac{\gamma W^2 l}{\epsilon} - 1 \right)^{d-1} + 1 \right] \right\}}.$$

If $y \in [-B, B]^d$ for $0 < B < \infty$, then we have

$$N_\infty(\epsilon, \mathcal{F}_{\mathcal{W}}^{\gamma}) \leq L_w^{l(d+2)} (1/\epsilon)^{l(d+2)}$$

where $L_w = \max(1, WB\gamma^2/4, W\gamma^2/4)$. For $\gamma = 1$, we have

$$N_1(\epsilon, \mu, \mathcal{F}_{\mathcal{W}}^{\gamma}, lW) \leq \left(\frac{4e(l+1)lW}{\epsilon} \right)^{l(2d+3)+1}.$$

Since $f_w(y) \leq lW$ for all $f_w \in \mathcal{F}_{\mathcal{W}}^{\gamma}$, we have

$$\sup_{x,y} |x - f(y)| \leq A + lW$$

which enables us to convert a cover for $\mathcal{F}_{\mathcal{W}}^{\gamma}$ into a cover for the class functions of the form $(x - f(y))^2$, for $f \in \mathcal{F}_{\mathcal{W}}^{\gamma}$. Based on these cover sizes, we can estimate the sample sizes required to ensure condition (1.2). Here f_w^* and $\hat{f}_w$ denote a neural network that minimizes $I(.)$ and $I_{emp}(.)$, respectively, over the set $\mathcal{F}_{\mathcal{W}}^{\gamma}$.

Theorem 4.1 [20] *Consider the class of feedforward neural networks $\mathcal{F}_{\mathcal{W}}^{\gamma}$ of Eq (4.1) Let $\mathcal{G}_{\mathcal{W}}^{\gamma} = \{(x - f_w(y))^2 : f_w \in \mathcal{F}_{\mathcal{W}}\}$ and $R = 8(A + lW)^2$. Given a sample of size at least*

$$\frac{16R}{\epsilon^2} \left(\ln(18/\delta) + 2\ln(8R/\epsilon^2) + \ln(2\gamma^2 W^2 lR/\epsilon) + \frac{\gamma W^2 lR}{\epsilon} \left[\left(\frac{\gamma W^2 lR}{\epsilon} - 1 \right)^{d-1} + 1 \right] \right),$$

the empirically best neural network $\hat{f}_w$ in $\mathcal{F}_{\mathcal{W}}$ approximates the best expected f_w^* in $\mathcal{F}_{\mathcal{W}}$ such that

$$P\left[I(\hat{f}_w) - I(f_w^*) > \epsilon \right] < \delta.$$

The same condition can also be ensured under the sample size

$$\frac{16R}{\epsilon^2} \left(\ln(18/\delta) + 2\ln(8R/\epsilon^2) + l(d+2)\ln(L_w R/\epsilon) \right)$$

where $L_w = \max(1, WB\gamma^2/4, W\gamma^2/4)$, *or, for* $\gamma = 1$,

$$\frac{128R}{\epsilon^2}\max\left\{\ln\left(\frac{8}{\delta}\right), \ln\left(\frac{16e(l+1)R}{\epsilon}\right)\right\}.$$

The three estimates in Theorem 4.1 provide three different means for controlling the sample size depending on the available information and intrinsic characteristics of the neural network class $\mathcal{F}_{\mathcal{W}}^{\gamma}$. For example, the sample size in the first bound is easier to modify by changing the parameter γ. In practice, it could be useful to compute all three bounds and choose the smallest one.

In statistics and control theory literature dealing with general function estimation problems (to which the present sensor fusion problem is closely related), asymptotic results are more common. The results in Theorem 4.1 can be used in Borel-Cantelli Lemma [1] to show that $I(\hat{f}) - I(f^*) \to 0$ as $n \to 0$ almost surely, thereby providing the asymptotic consistency result for the sensor fusion design problem.

5 Fusers Based on Nadaraya-Watson Estimator

We now present a polynomial-time (in sample size n) estimator which guarantees the criterion (1.2) under additional conditions listed in Theorem 5.1.

Given an n-sample, the Nadaraya-Watson estimator based on Haar kernels is defined by

$$\hat{f}_{m,n}(y) = \frac{\sum_{j=1}^{n} X_j P_m(y, Y_j)}{\sum_{j=1}^{n} P_m(y, Y_j)} = \frac{\sum_{Y_j \in J} X_j}{\sum_{Y_j \in J} 1_J(Y_j)} \qquad (3.4)$$

for $y \in J$ [15] (see also Engel [7]) [1]. The second expression indicates that $\hat{f}_{m,n}(y)$ is the mean of the function values corresponding to Y_j's in J that contains y. This property is the key to efficient computation of the estimate [26].

The Nadaraya-Watson estimator based on more general kernels is classical in statistics literature [11]. Since its introduction in the early sixties, this estimator was successfully employed in a number of applications involving nonlinear regression estimation. The classical analysis of this estimator was restricted to asymptotic results, and is not particularly directed towards linear-time computation. This computationally efficient version based on Haar kernels is due to Engel [7], which was subsequently shown to yield finite sample guarantees by Rao and Protopopescu [26]. The result of [26] requires finiteness of capacity of $\mathcal{F}$ in addition to smoothness, and here we require only the latter. The following theorem specifies the sample size needed to ensure the condition (1.2).

Theorem 5.1 *Consider a family of functions* $\mathcal{F} \subseteq C(Q)$ *with range* $[0,1]$ *such that* $\omega_\infty(f;r) \le kr$ *for some* $0 < k < \infty$. *We assume that: (i) there exists a family of densities* $\mathcal{P} \subseteq C(Q)$; *(ii) for each* $p \in \mathcal{P}$, $\omega_\infty(p;r) \le kr$; *and (iii) there exists* $\mu > 0$ *such that for each* $p \in \mathcal{P}$, $p(y) > \mu$ *for all* $y \in [0,1]^N$. *Suppose that the sample size, n, is larger than*

$$\frac{2^{m+4}}{\epsilon_1^2}\left[\left(\frac{k2^m}{\epsilon_1}\left[\left(\frac{k2^m}{\epsilon_1}-1\right)^{N-1}+1\right]+m\right)\ln\left(2^{m+1}k/\epsilon_1\right)+\ln\left(\frac{2^{2m+6}}{(\delta-\lambda)\epsilon_1^4}\right)\right]$$

where $\epsilon_1 = \epsilon(\mu-\epsilon)/4$, $0 < \beta < \frac{N}{2(N+1)}$, $m = \lceil \frac{\log n\beta}{N} \rceil$ *and* $\lambda = b\left(\frac{2}{\epsilon}\right)^{1/N+1-1/2\beta} + b\left(\frac{2}{\epsilon_1}\right)^{1/N+1-1/2\beta}$. *Then for any* $f \in \mathcal{F}$, *we have* $P\left[|I(\hat{f}_{m,n}) - I(f^*)| > \epsilon\right] < \delta$.

The computation of $\hat{f}_{m,n}(y)$ at a given y involves obtaining the local sum of X_i's in J that contains y. The range-tree (see Preparata and Shamos [14]) can be constructed to store the cells J that contain at least one Y_i; with each such cell, we store the number of the Y_i's that are contained in J and the sum of the corresponding X_i's. This computation can be achieved by known methods [14] in $O(n(\log n)^{N-1})$ time , and the values of J containing y can be retrieved in $O((\log n)^N)$ time. Thus $\hat{f}_{m,n}(y)$ can be computed in $O((\log n)^N)$ time after a preprocessing step in $O(n(\log n)^{N-1})$ time (see [26]).

6 Simulation Results

We present two examples to illustrate the performance of neural network and Nadaraya-Watson estimator. For both examples we also provide results obtained with the nearest neighbor rule, which is analyzed elsewhere [18]. In the second example, we also consider another estimate, namely, the empirical decision rule described in [22].

Example 1: *Fusion of Function Estimators:* [26] We consider five function estimators each of which

[1] Conventionally this estimator is used to fit functions of the form $f(X) = Y$ (or its regression version). Due to the form of the present sensor fusion problem, namely fitting functions of the form $f(Y) = X$, the conventional notational roles of the variables X_i and Y_i are switched in this expression.

Training Set	Testing Set	Nadaraya-Watson	Nearest Neighbor	Neural Network
100	10	0.000902	0.002430	0.048654
1000	100	0.001955	0.003538	0.049281
10000	1000	0.001948	0.003743	0.050942

(a) $d = 3$

Training Set	Testing Set	Nadaraya-Watson	Nearest Neighbor	Neural Network
100	10	0.004421	0.014400	0.018042
1000	100	0.002944	0.003737	0.021447
10000	1000	0.001949	0.003490	0.023953

(b) $d = 5$

Table 1. Comparative performance.

outputs the value of an unknown function $g(X) \in [0,1]$ at the input $X \in [0,1]^d$. In particular S_i outputs a corrupted value $g_i(X)$ of $g(X)$ when presented with input $X \in [0,1]^d$. The fusion problem is to compute a function $f : [0,1]^5 \mapsto [0,1]$ such that $f(g_1(X), \ldots, g_5(X))$ closely approximates $g(X)$. Here g is realized by a feedforward neural network, and, for $i = 1, 2, \ldots, 5$, $g_i(X) = g(X)(1/2 + iZ/10)$ where Z is uniformly distributed over $[-1, 1]$; note that $1/2 - i/10 \leq g_i(X)/g(X) \leq 1/2 + i/10$. Table 1 corresponds to the mean square error in the estimation of f for $d = 3$ and $d = 5$, respectively, using the Nadaraya-Watson estimator, nearest neighbor rule, and a feedforward neural network with backpropagation learning algorithm. Note the superior performance of the Nadaraya-Watson estimator. □

Example 2: *Decision Fusion:* [22, 20] We consider a system with 5 sensors such that $Y \in \{H_0, H_1\}^5$. To each X there corresponds a "correct" decision; in the training data the correct decision (H_1 or H_0) is generated with equal probabilities for each X_i, i. e., $P(H_0|X) = P(H_1|X) = 1/2$. The sensor S_i, $i = 1, 2, \ldots, 5$, introduces an error as follows: the output corresponds to the correct decision with probability of $1 - i/10$, and with probability $i/10$ output is the opposite. The individual sensor behavior is implemented by generating a uniform random variable in the range $[0, D]$ and checking whether it falls within the interval $[0, iD/10]$. The sensor fusion problem is to compute a rule that combines the outputs of the sensors to predict the correct decision. The percentage error of the individual detectors and the fused system based on the Nadaraya-Watson estimator is presented in Table 2. Note that the fuser is consistently better than the best sensor S_1 beyond the sample sizes of the order of 1000. Thus this example illustrates that the performances exceeding the best of the individual sensors can be achieved through fusion methods. A comparative performance of the Nadaraya-Watson estimator, empirical decision rule, nearest neighbor rule, and the Bayesian rule based on the analytical formulae is presented in Table 3. The Bayesian rule is computed based on the formulae used in the data generation and is provided for comparison only (note that such formulae are assumed to be not available in computing the other estimators). □

Sample Size	Test set size	S_1	S_2	S_3	S_4	S_5	Nadaraya-Watson
100	100	7.0	20.0	33.0	35.0	55.0	12.0
1000	1000	11.3	18.5	29.8	38.7	51.6	10.6
10000	10000	9.56	20.19	30.38	39.82	49.68	8.58
50000	50000	10.038	20.136	29.854	39.904	50.050	8.860

Table 2: Performance of Nadaraya-Watson estimator.

Sample Size	Test set size	Bayesian Fuser	Empirical Decision	Nearest Neighbor	Nadaraya-Watson
100	100	91.91	23.00	82.83	88.00
1000	1000	91.99	82.58	90.39	89.40
10000	10000	91.11	90.15	90.81	91.42
50000	50000	91.19	90.99	91.13	91.14

Table 3: Comparative Performance.

7 Discussion of Performance

The results obtained in Sections 3-5 guarantee only a PAC fuser design. The performance of the fuser compared to its individual components has not been addressed. We now investigate the conditions under which the composite system is at least as good as the best of the individual sensors.

Consider that we use the empirical data to obtain a function $f_i \in \mathcal{F}_i$ that maps the output of the sensor S_i to $\Re^d$. The performance of f_i can be measured by using (with abuse of notation) the following cost function

$$I(f_i) = \int [X - f_i(Y)]^2 dP_{(Y|X)} dP_X$$

as opposed to (1.2).

The success of the fuser design is determined by comparing the performance of the composite system $\langle f, S_1, \ldots, S_N \rangle$ to the individual sensor system $\langle f_i, S_i \rangle$. The relative performance here depends on $\mathcal{F}$ and $\mathcal{F}_i$'s, and also on $\hat{f}$ and $\hat{f}_i$'s. We consider that the empirical estimation methods of Section 3 or 4 are used in obtaining $\hat{f}$ and $\hat{f}_i$. In particular, we normalize the performance parameters by recomputing $\delta_i(\epsilon, \mathcal{F}_i, l)$ corresponding to $\epsilon_i = \epsilon$ for each sensor and we also compute $\delta_F(\epsilon, \mathcal{F}, l)$. Then a single sensor system with least value for δ, denoted by $\langle f_{\min}, S_{\min} \rangle$, is called *best* sensor system, i.e., $\delta_{\min}(\epsilon, \mathcal{F}_{\min}, l) = \min_i \delta_i(\epsilon, \mathcal{F}_i, l)$. Then the composite system will be at least as good as the best of the individual sensor systems under the condition [18]

$$\delta_F(\epsilon, \mathcal{F}, l) \leq \delta_{\min}(\epsilon, \mathcal{F}_{\min}, l).$$

In the present case, the error and its expectation are related by the following equations:

$$E[I(\hat{f}_i) - I(f_i^*)] = \int P[I(\hat{f}_i) - I^*(f_i^*) > \epsilon] d\epsilon \quad (8.1)$$

$$P[I(\hat{f}_i) - I^*(f_i^*) > \epsilon] \leq \frac{1}{\epsilon} E[I(\hat{f}_i) - I(f_i^*)] \quad (8.2)$$

where the former follows from the definition of expectation and the latter is the well-known Chebyshev's inequality. Based on (8.2), the smaller the expected error, the smaller will be the corresponding δ for fixed ϵ.

We now consider a specific class of fusion rules $\mathcal{F} = \{w_1 f_1 + \ldots + w_N f_N\}$, for $(w_1, \ldots, w_N) \in \Re^N$ such that $\sum_{i=1}^{N} w_i = 1$, and $0 \leq w_i \leq 1$. The first consequence is that $f_i \in \mathcal{F}$, which implies that

$$\min_{f \in \mathcal{F}} I_{emp}(f) \leq \min_i \min_{f_i \in \mathcal{F}_i} I_{emp}(f_i).$$

Thus the empirical error of the fuser is no greater than that of the best sensor. An analogous property is valid for $I(.)$ as stated in the next theorem. Let Π be a permutation of $\{1, 2, \ldots, N\}$, and define

$$\Upsilon(\delta_1, \delta_2, \ldots, \delta_N) = \max_{\Pi} \left(\frac{1 - \delta_{\Pi(N)}}{2^{N-1}} - \sum_{i=1}^{N-1} \frac{\delta_{\Pi(i)}}{2^i} \right).$$

Let A_i denote the event $I(\hat{f}_i) \leq \epsilon$ such that $P(A_i) > 1 - \delta_i$. Then by Lemma 2.1 we have

$$P(A_1 \cap A_2 \cap \ldots \cap A_N) \geq \Upsilon(\delta_1, \delta_2, \ldots, \delta_N).$$

Theorem 7.1 [18] *With probability* $\Upsilon(\delta_1, \ldots, \delta_N)$, *we have* $\delta_F(\epsilon, \mathcal{F}, l) \leq \min_i \delta_i(\epsilon, \mathcal{F}_i, l)$, *where*

$$I_{emp}(\hat{f}) = \min_{f \in \mathcal{F}} I_{emp}(f), \; I_{emp}(\hat{f}_i) = \min_{f_i \in \mathcal{F}_i} I_{emp}(f_i)$$

and $\mathcal{F} = \{w_1 f_1 + w_2 f_2 + \ldots + w_N f_N\}$, *such that* $w_i \in [0, 1]$ *and* $f_i \in \mathcal{F}_i$, $i = 1, 2, \ldots, N$, *and* $\sum_{i=1}^{N} w_i = 1$.

The above theorem illustrates that if no other information is available, the fuser hypothesis space can be easily constructed using weighted sums of functions from individual hypotheses classes.

There are a number of related methods for fusion rule estimation under different types of typically stronger conditions. Three methods based on classical Robbins-Monroe algorithms, potential functions, and kernel regression methods are proposed in [17] for the fusion rule estimation. These methods are algorithmic (in contrast with general solutions of Section 3) but are guaranteed to satisfy criterion (1.2) under various smoothness and martingale conditions. These conditions are very difficult to verify in typical applications.

From a computational point of view, the class of linearly separable systems of [19] constitute a non-trivial example where the empirical risk minimization of Section 3 can be solved in polynomial time (using quadratic programming methods).

8 Conclusions

We presented a review of solutions to the general sensor fusion problem, where the underlying sensor error distributions are not known but a sample is available. Based on the smoothness and/or combinatorial properties of the class of fusion rules, general solutions to the problem are provided based on empirical risk minimization. Two computationally viable methods are

presented based on feedforward sigmoidal networks and Nadaraya-Watson estimator. An assessment of these methods was carried out as to their intrinsic characteristics and overall performance.

Several computational issues of the fusion rule estimation are open problems. It would be interesting to obtain general conditions under which polynomial-time algorithms can be used to solve the fusion rule estimation problem under the criterion (1.2). It would also be interesting to investigate the utility of computational methods based on bootstrap and cross-validation in the fusion rule estimation problem. Also, conditions under which the composite system is "significantly" better than best sensor would be extremely useful. Finally, lower bound estimates for various sample sizes will be very important in judging the optimality of sample size estimates.

Acknowledgements

This research is sponsored by the Engineering Research Program of the Office of Basic Energy Sciences, of the U.S. Department of Energy, under Contract No. DE-AC05-96OR22464 with Lockheed Martin Energy Research Corp. We deeply appreciate the constructive comments of Vladimir Protopopescu which greatly improved the presentation of this paper.

References

[1] P. Billingsley. *Probability and Measure.* John Wiley and Sons, New York, second edition, 1986.

[2] A. L. Blum and R. L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5:117–127, 1992.

[3] C. K. Chow. Statistical independence and threshold functions. *IEEE Trans. Electronic Computers*, EC-16:66–68, 1965.

[4] R. Cole. Parallel merge sort. *SIAM Journal on Computing*, 17(4):770–785, 1988.

[5] B. V. Dasarathy. *Decision Fusion.* IEEE Computer Society Press, Los Alamitos, California, 1994.

[6] B. V. Dasarathy. Special issue on sensor fusion. *Optical Engineering*, 35(3), 1996.

[7] J. Engel. A simple wavelet approach to nonparametric regression from recursive partitioning schemes. *Journal of Multivariate Analysis*, 49:242–254, 1994.

[8] B. Grofman and G. Owen, editors. *Information Pooling and Group Decision Making.* Jai Press Inc., Greenwich, Connecticut, 1986.

[9] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, 100:78–150, 1992.

[10] G. Lugosi and K. Zeger. Nonparametric estimation via empirical risk minimization. *IEEE Transactions on Information Theory*, 41(3):677–687, 1995.

[11] E. A. Nadaraya. *Nonparametric Estimation of Probability Densities and Regression Curves.* Kluwer Academic Publishers, Dordrecht, 1989.

[12] D. Pollard. *Convergence of Stochastic Processes.* Springer-Verlag, New York, 1984.

[13] D. Pollard. *Empirical Processes: Theory and Applications.* Institute of Mathematical Statistics, Haywood, California, 1990.

[14] F. P. Preparata and I. A. Shamos. *Computational Geometry: An Introduction.* Springer-Verlag, New York, 1985.

[15] B. L. S. Prakasa Rao. *Nonparametric Functional Estimation.* Academic Press, New York, 1983.

[16] N. S. V. Rao. Distributed decision fusion using empirical estimation. *IEEE Transations on Aerospace and Electronic Sysetms*, 1995. under revision.

[17] N. S. V. Rao. Fusion rule estimation in multiple sensor systems using training. In H. Bunke, T. Kanade, and H. Noltemeier, editors, *Modelling and Planning for Sensor Based Intelligent Robot Systems*, pages 179–190. World Scientific Pub., 1995.

[18] N. S. V. Rao. Fusion rule estimation in multiple sensor systems with unknown noise densities. *Journal of Franklin Institute*, 331B(5):509–530, 1995.

[19] N. S. V. Rao. Fusion rule estimation in multiple sensor systems with unknown noise distributions. In R. N. Madan, N. S. V. Rao, V. P. Bhatkar, and L. M. Patnaik, editors, *Parallel and Distributed Signal and Image Integration Problems*, pages 263–279. World Scientific Pub., 1995.

[20] N. S. V. Rao. Fusion methods in multiple sensor systems using feedforward neural networks. *Intelligent Automation and Soft Computing*, 1996. submitted.

[21] N. S. V. Rao. Nadaraya-Watson estimator for sensor fusion. *Optical Engineering*, 1996. submitted.

[22] N. S. V. Rao and S. S. Iyengar. Distributed decision fusion under unknown distributions. *Optical Engineering*, 35(3), 1996.

[23] N. S. V. Rao and E. M. Oblow. Majority and location-based fusers for PAC concept learners. *IEEE Trans. on Syst., Man and Cybernetics*, 24(5):i713–727, 1994.

[24] N. S. V. Rao and E. M. Oblow. N-learners problem: System of PAC learners. In *Computational Learning Theory and Natural Learning Systems, Vol IV: Making Learning Practical.* MIT Press, 1996. in press.

[25] N. S. V. Rao, E. M. Oblow, C. W. Glover, and G. E. Liepins. N-learners problem: Fusion of concepts. *IEEE Transactions on Systems, Man and Cybernetics*, 24(2):319–327, 1994.

[26] N. S. V. Rao and V. Protopopescu. On PAC learning of functions with smoothness properties using feedforward sigmoidal networks. *Proceedings of the IEEE*, 1996. to appear.

[27] N. S. V. Rao, V. A. Protopopescu, and H. Qiao. Function estimation by feedforward sigmoidal networks with bounded weights. Oak Ridge National Laboratory, Oak Ridge, TN, 1996. manuscript.

[28] V. Roychowdhury, K. Siu, and A. Orlitsky, editors. *Theoretical Advances in Neural Computation and Learning*. Kluwer Academmic Pub., 1994.

[29] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[30] V. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, New York, 1982.

[31] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

[32] J. von Neumann. Probabilistic logics and the synthesis of reliable organisms from unreliable components. In C. E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 43–98, 1956. Princeton University Press.