

Finite Sample Performance Guarantees of Fusers for Function Estimators

Nageswara S.V. Rao *

Computer Science and Mathematics Division

Oak Ridge National Laboratory

Oak Ridge, Tennessee 37831, USA

Information Fusion, vol. 1, no. 2, 2000, pp. 101-102.

March 6, 2002

Abstract

An independent and identically distributed sample of an unknown function generated according to an unknown distribution is given. Several function estimators are computed based on the sample by minimizing the empirical error over function families. These estimators provide performance guarantees based on best available cover sizes for the respective function families. Traditionally, the estimator that performs the best on the training data or provides the best performance guarantee is often selected and others are discarded when no other information – such as additional examples – is available. We consider a fuser trained with the outputs of the individual estimators by minimizing empirical error over a fuser class. If the fuser class satisfies a simple isolation property and has a smaller cover size compared to individual estimators, we show that the performance guarantee of the fuser is at least as good as that of the empirical best estimator. Several well-known fusers such as linear combinations, special potential functions, and certain feedforward piecewise-linear networks satisfy the isolation property. In the first two cases the fuser class forms a vector space for which we derive more detailed conditions. We also derive conditions in terms of the natural parameters when the estimators are feedforward sigmoidal networks with bounded weights. We present simulation results to show the effectiveness of the fuser.

Index terms— Function estimation, multiple models, PAC learning, sample size estimation

*This research is sponsored by the Engineering Research Program of the Office of Basic Energy Sciences, U.S. Department of Energy, under Contract No. DE-AC05-96OR22464 with Lockheed Martin Energy Research Corp., the Seed Money Program of Oak Ridge National Laboratory, and the Office of Naval Research under order N00014-96-F-0415.

1 Introduction

Function estimation problems based on empirical data arise in a number of disciplines. Some of the widely-used methods for function estimation are from the areas of statistics [30], systems theory [24], and more recently from computer science [16]. In general, the performance of the function estimators from the statistics and system theory literature is characterized in terms of asymptotic convergence results, whereas many of the results from computer science literature are in terms of finite sample guarantees. Furthermore, the results from the former are typically stated in terms of smoothness properties whereas those from the latter are typically stated in terms of combinatorial parameters. The performance conditions in either case could be quite involved and beyond the expertise of an average applications person [40, 41]. At present, a practitioner is provided a wide variety of function estimators, each of which is characterized by (a) empirical performance based on a sample, and (b) performance guarantees provided through analysis. In terms of finite sample results, the available guarantees are in terms of the best available upperbounds, which are often utilized in comparing the estimators. Nevertheless, several of these estimators are based on considerable practical and theoretical insights, and it would be most desirable to retain some of their strengths. This paper is an attempt at such approach. Informally speaking, we show that the various estimators can be fused without losing the performance guarantee of the best individual estimator, and this can be done without the knowledge of the relative performances of the estimators. For the practitioner our results suggest a simple recipe: utilize available methods to compute a number of function estimators, and then combine them using a fuser from a class that satisfies the isolation property and has small cover size. These properties are satisfied by linear combinations and certain feedforward networks, and are fairly easy to verify in general.

Problems dealing with multiple methods for solving a problem have been studied as early as 1818 by Laplace [10] (see [26] for a brief historical account of information/decision fusion methods). In engineering systems, von Neumann [42] showed that unreliable components can be combined to produce a more reliable computer. In a ground-breaking paper in the area of forecasting, Bates and Granger [5] showed that “better” forecasts can be produced by (linearly) combining several different forecast modules. Throughout the long history of the problems of this nature, it is generally accepted that “better” results can be produced by “suitably” combining the methods rather than picking the best based on the test data. Roughly speaking, we show that such result holds for function estimation problems within the Probably Approximately Correct (PAC) framework of Valiant [38] under fairly mild conditions. Such problems are being extensively studied by the PAC learning community [43, 13, 7]; we subsequently discuss the relationship between our formulation and the existing ones.

We are required to estimate a function $f : \mathcal{R}^d \mapsto [0, 1]$, based on a sample $(X_1, f(X_1)), (X_2, f(X_2)), \dots, (X_l, f(X_l))$, where X_1, X_2, \dots, X_l ¹ is independently and identically distributed (iid) according to an *unknown* distribution P_X defined on \mathcal{R}^d . For an estimator \hat{f}

¹Throughout the paper random variables are denoted by upper case letters, e.g. X, Y , etc., and their deterministic versions are denoted by the corresponding lower case letters, e.g. x, y , etc.

of f we consider the *expected square error* given by ²

$$I(\hat{f}) = \int (f(X) - \hat{f}(X))^2 dP_X.$$

Typically \hat{f} is chosen from a function class \mathcal{F} , e. g. a feedforward sigmoid networks of certain architecture [35], potential functions [25], or radial basis functions [21]. A precise estimation of $f^* \in \mathcal{F}$ that minimizes the expected error over the function class is not possible since P_X is unknown. We adopt the popular PAC framework [38] in quantifying the generalization property in terms of $I(\cdot)$. Given a sufficiently large sample, we require that

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

for $\epsilon > 0$, $0 < \delta < 1$, where the sample size is a function of ϵ , δ , and certain parameters of the function class. As per the convention, $P = P_X^l$ denotes the distribution of X_1, X_2, \dots, X_l , and ϵ and δ denote *precision* and *confidence*, respectively. Note that if the function class contains f then $I(f^*) = 0$. The main attraction of this paradigm is that the sample size can be specified entirely in terms of ϵ , δ and the function class \mathcal{F} . In particular, no knowledge of the distributions is needed.

We are given N function estimators each of which is obtained by *deterministically* selecting a function that minimizes the empirical error over a class of functions. Let $\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_N$ correspond to the function classes of the estimators (each function class contains functions of the form $h : \mathbb{R}^d \mapsto [0, 1]$). For example, \hat{f}_1, \hat{f}_2 , and \hat{f}_3 could be a neural network, a potential function and a linear estimator, respectively. The performance bounds of these estimators are specified in terms of their corresponding parameters. For the i th estimator, let $I(f_i^*) = \min_{f_i \in \mathcal{F}_i} I(f_i)$ and

$I(\hat{f}_i) = \min_{f_i \in \mathcal{F}_i} \hat{I}(f_i)$, where the *empirical error* $\hat{I}(\cdot)$ is given by

$$\hat{I}(\hat{f}) = \frac{1}{l} \sum_{i=1}^l (f(X_i) - \hat{f}(X_i))^2.$$

If \mathcal{F}_i satisfies certain properties, e. g. finiteness of psuedo-dimension [16] or scale sensitive dimension [3], we have $P[I(\hat{f}_i) - I(f_i^*) > \epsilon] < \delta_i$, where δ_i depends on the sample size, ϵ and the parameters of \mathcal{F}_i . For each estimator the best available bound for δ_i is utilized here, which may or may not be the best possible one. The weakest characterization of \mathcal{F}_i for which such results are established is the finiteness of scale-sensitive dimension [3]. There are a number of stronger characterization such as the finiteness of L_1 -cover [39], pseudo-dimension [29], and capacity [40] that are sufficient to yield such results (see [16] for a comprehensive treatment). The actual characterization is not very critical for the discussion of our results, and we choose to present our results in terms of L_∞ - or L_1 -cover for ease of presentation. These cover sizes are utilized in providing finite sample performance bounds for a number of practical methods such as potential functions and neural networks (Sections 3 and 4).

We address the question of choosing the “best” estimator versus “fusing” the estimators under the conditions that no additional examples are available. We consider that each \hat{f}_i is

²Certain measurability properties are assumed to be satisfied for the existence of various expected errors in this paper (see [28] for details).

computed as a deterministic function of the sample. In terms of finite sample guarantees, we show that the fuser is at least as efficient as the empirical best estimator (in a precise sense defined in the next section) if the function class of the fuser: (a) satisfies a simple isolation property (defined in Section 2), and (b) has the cover size smaller than the sum of the cover sizes of individual estimators. The isolation property is satisfied for feedforward piecewise-linear networks, special potential functions, and linear combinations. When function estimators are based on the popular feedforward sigmoid feedforward networks, and potential function methods, we derive more detailed conditions in terms of the natural parameters.

We then consider the case where the fuser function class forms a vector space of which the linear combinations are a special case. In this case, the computation of fuser can be performed in polynomial (in l) time using well-known least square methods. Our result provides an analytical justification for the success of linear combination fusers in a number of areas such as forecasting [5], neural networks [15], regression estimation [?], and pattern recognition.

There are a large number of fusion methods studied in various disciplines – the analytical results concerning the fuser are rather specific to the individual areas. Within the framework of PAC learning, there are a number of works dealing with “combining learners” (for example see [19, 12, 6]). The most distinguishing features of our framework include: (a) fixed set of potentially different estimators, (b) fixed iid sample, and (c) off-line learning formulation. This framework is motivated by the applications areas of automatic target detection [9], and sensor-based robotics [1]. In these applications, the sample is expensive to collect and is typically gathered in a batch mode, for example, by flying an airplane to collect data corresponding to known targets or by conducting indoor mobile robot experiments. Also, several domain-specific learning methods have been developed based on different approaches which are very difficult to compare (not unlike in the area of forecasting [11]) mainly due to the technical conditions under which their performance is characterized.

We now briefly summarize the existing formulations from PAC learning literature that are very closely related to ours. The methods of Freund [12] and Schapire [37] rely on utilizing different sets of examples to generate a large number of hypotheses using a single learner. In the method of Kearns and Seung [19], the number of hypotheses is a variable, and the hypotheses are chosen to be independent to meet the performance criterion. In spirit, our formulation is the closest to that of combining experts to achieve the performance of the best expert. The literature on this problem is rather extensive (see for example Cesa-Bianchi *et al* [7, 6], Haussler *et al* [17], Littlestone and Warmuth [23], Vovk [43], for more details), but the most of these results deal with on-line framework. In addition to a less stringent requirement due to off-line criterion, we deal with function estimation; most results on combining experts deal with indicators functions, with the exceptions such as the works of Kivinen and Warmuth [20], and Freund and Schapire [13].

The organization of the paper is as follows. In Section 2 we show that the isolation property and smaller cover size of the fuser class are sufficient to ensure that the fuser performs at least as efficiently as the best of the estimators. In Section 3, we show an implementation of a fuser for a set of sigmoid neural networks with bounded weights. In Section 4, we discuss the vector space fusers which include the popular linear combination fusers and special potential functions. We describe a simulation example in Section 5.

2 Fused System Versus Best Estimator

Given the set of N estimators, for the i th estimator, we have

$$P \left[\sup_{g \in \mathcal{F}_i} |I(g) - \hat{I}(g)| > \epsilon/2 \right] < \delta_i,$$

which implies

$$P[I(\hat{f}_i) - I(f_i^*) > \epsilon] < \delta_i,$$

where δ_i can be specified in terms of a “suitable” cover size of \mathcal{F}_i (see Appendix). Recall that we utilize the best available value for δ_i . Let $\hat{f}_{\min} = \arg \min_{i=1}^N \hat{I}(\hat{f}_i)$ denote the *empirical best* estimator, and $f_{\min}^* = \arg \min_{i=1}^N I(f_i^*)$ denote the *expected best* estimator. Then with probability $1 - (\delta_1 + \dots + \delta_N)$ we have $I(\hat{f}_i) - I(f_i^*) \leq \epsilon$ for all $i = 1, 2, \dots, N$, which implies $I(\hat{f}_{\min}) - I(f_{\min}^*) \leq \epsilon$ or equivalently

$$P[I(\hat{f}_{\min}) - I(f_{\min}^*) > \epsilon] < \delta_1 + \dots + \delta_N. \quad (2.1)$$

This condition specifies the performance guarantees that can be provided if the empirical best estimator is chosen, compared to the best possible estimator from $\mathcal{F}_1 \cup \mathcal{F}_2 \cup \dots \cup \mathcal{F}_N$.

The following definition³ enables us to compare two estimators in terms of the performance guarantees.

Definition 2.1 Consider two function estimates \hat{g}_1 and \hat{g}_2 , for a function g such that

$$P[I(\hat{g}_1) - I(g) > \epsilon] < \delta_1(\epsilon, l)$$

$$P[I(\hat{g}_2) - I(g) > \epsilon] < \delta_2(\epsilon, l).$$

The performance guarantee of the estimate \hat{g}_1 is at least as good as that of \hat{g}_2 with respect to function g , if $\delta_1(\epsilon, l) \leq \delta_2(\epsilon, l)$, for all ϵ .

The motivation of this definition is pragmatic: it relates what is *known* about the performance guarantees rather than the inherent properties of the estimators, and thus, enables us to compare estimators in practical scenarios. A definition based on actual values of δ_i 's might be more satisfying from a mathematical perspective, but, on the other hand, will not be of much practical value (since such values are known only in very limited cases).

The outputs of the estimators on the sample are utilized to compute the fuser as follows. Given the estimators $\hat{f}_1, \hat{f}_2, \dots, \hat{f}_N$, the fuser is computed based on the sample $(\hat{f}(X_1), f(X_1)), (\hat{f}(X_2), f(X_2)), \dots, (\hat{f}(X_l), f(X_l))$, generated from the original sample such that $\hat{f}(X_i) = (\hat{f}_1(X_i), \dots, \hat{f}_N(X_i))$, by choosing a function from the fuser class $\mathcal{F}_F = \{f_F : [0, 1]^N \mapsto [0, 1]\}$. The expected and empirical errors of the fuser are given by

$$I_F(f_F) = \int [f(X) - f_F(\hat{f}_1(X), \hat{f}_2(X), \dots, \hat{f}_N(X))]^2 dP_X.$$

³This definition may not yield yes/no answer for certain pairs of estimators if $\delta_1 < \delta_2$ is not satisfied for the entire range of ϵ . This can be modified to be sensitive to certain range of values of ϵ .

$$\hat{I}_F(f_F) = \frac{1}{l} \sum_{i=1}^l [f(X_i) - f_F(\hat{f}_1(X_i), \hat{f}_2(X_i), \dots, \hat{f}_N(X_i))]^2,$$

respectively. Let f_F^* and \hat{f}_F minimize $I_F(\cdot)$ and $\hat{I}_F(\cdot)$ over \mathcal{F}_F , respectively. To compare the fused system with the best of the individual estimators, we use the following obvious extension of Definition 2.1.

Definition 2.2 Consider the system of N estimators and the fuser such that

$$P[I(\hat{f}_{\min}) - I(f_{\min}^*) > \epsilon] < \delta_1(\epsilon, l) + \dots + \delta_N(\epsilon, l)$$

$$P[I_F(\hat{f}_F) - I(f_{\min}^*) > \epsilon] < \delta_F(\epsilon, l).$$

The performance guarantee of the fuser \hat{f}_F is at least as good as that of the empirical best learner \hat{f}_{\min} if for all ϵ we have

$$\delta_F(\epsilon, l) \leq \delta_1(\epsilon, l) + \dots + \delta_N(\epsilon, l).$$

Informally speaking, the condition requires that the δ of the fuser be smaller than that of the empirical best estimator for the same value of ϵ and l , i.e. fuser has higher confidence of achieving the same precision as the empirical best estimator for the same sample size.

The following definition specifies a simple but a critical constituent property that yields an efficient fuser.

Definition 2.3 A function class $\mathcal{F} = \{f : [0, \tau]^k \mapsto [0, \tau]\}$ has the isolation property if it contains the functions $f^i(y_1, y_2, \dots, y_k) = y_i$ for all $i = 1, 2, \dots, k$.

The isolation property was first proposed in [34, 31] for concept and sensor fusion problems. If \mathcal{F} consists of linear combinations, i. e. $f(y_1, y_2, \dots, y_k) = w_1 y_1 + w_2 y_2 + \dots + w_k y_k$, for $w_i \in \mathfrak{R}$, this property is trivially satisfied. If \mathcal{F} consists of the potential functions [2] this property is not satisfied in general. This property is also not satisfied for feedforward sigmoid networks, but is satisfied by certain feedforward networks with piecewise linear units (see Section 3). In the special case of Boolean valued functions, the isolation property is satisfied when \mathcal{F} corresponds to set of all Boolean functions on k variables.

If \mathcal{F}_F has isolation property, it is direct to note that $I_F(f_F^*) \leq I(f_{\min}^*)$, and $\hat{I}_F(\hat{f}_F) \leq \hat{I}(\hat{f}_{\min})$. But, for the fuser to be better, $I_F(\hat{f}_F)$ must be better than $I(f_{\min}^*)$ as per the Definition 2.2. The other parameters of the fuser must be accounted for to ensure that $\delta_F \leq \delta_1 + \dots + \delta_N$. While such guarantees are not possible in all cases, very general conditions are sufficient. We now present two such condition in terms of cover sizes, which are used in a number of practical estimators.

Let S be a set equipped with a pseudometric ρ . The covering number $N(\epsilon, \rho, S)$ under metric ρ is defined as the smallest number of closed balls of radius ϵ , and centers in S , whose union covers S . For a set of functions $\mathcal{G} = \{g : \mathfrak{R}^M \mapsto [0, 1]\}$, we consider two metrics defined as follows: for $g_1, g_2 \in \mathcal{G}$ we have

$$d_P(g_1, g_2) = \int_{z \in \mathfrak{R}^M} |g_1(z) - g_2(z)| dP,$$

for the probability distribution P defined on \mathfrak{R}^M , and

$$d_\infty(g_1, g_2) = \sup_{z \in \mathfrak{R}^M} |g_1(z) - g_2(z)|.$$

This definition is applied to functions defined on $A \subseteq \mathfrak{R}^M$ by extending them to take value 0 on $\mathfrak{R}^M \setminus A$.

Theorem 2.1 *Given N function estimators such that i th estimator is chosen to minimize empirical error over the class $\mathcal{F}_i = \{f_i : \mathfrak{R}^d \mapsto [0, 1]\}$. If the fuser class $\mathcal{F}_F = \{f_F : [0, 1]^N \mapsto [0, 1]\}$ has the isolation property, the performance guarantee of the fuser is at least as good as that of the empirical best estimator under the condition*

$$N(\epsilon, \mathcal{F}_F) \leq N(\epsilon, \mathcal{F}_1) + \dots + N(\epsilon, \mathcal{F}_N)$$

for the cases: (i) $N(\epsilon, \mathcal{G}) = N(\epsilon, d_\infty, \mathcal{G})$ for $\mathcal{G} = \mathcal{F}_1, \dots, \mathcal{F}_N, \mathcal{F}_F$, and (ii) $N(\epsilon, \mathcal{G}) = N(\epsilon, d_P, \mathcal{G})$, for $\mathcal{G} = \mathcal{F}_1, \dots, \mathcal{F}_N, \mathcal{F}_F$ such that the bound $N(\epsilon, d_P, \mathcal{G})$ is valid for all distributions P .

Proof: Since $I_F(f_F^*) \leq I(f_{\min}^*)$, we have

$$P_X^l \left[I_F(\hat{f}_F) - I(f_{\min}^*) > \epsilon \right] \leq P_X^l \left[I_F(\hat{f}_F) - I_F(f_F^*) > \epsilon \right].$$

For $\mathcal{X} = \{X_1, X_2, \dots, X_l\}$, let \mathcal{E}_i denote the function (defined on set of all \mathcal{X} and taking values in \mathcal{F}_i) corresponding to the i th estimator such that $\mathcal{E}_i(\mathcal{X}) = \hat{f}_i$. Thus, there exists a function \mathcal{E} such that $\mathcal{E}(\mathcal{X}) = \hat{f} = (\hat{f}_1, \hat{f}_2, \dots, \hat{f}_N)$. For $f_F \in \mathcal{F}_F$, and estimator \hat{f} , we define $f_F \circ \hat{f}$ such that $(f_F \circ \hat{f})(X) = f_F(\hat{f}(X))$, for $\hat{f}(X) = (\hat{f}_1(X), \hat{f}_2(X), \dots, \hat{f}_N(X))$. Then, the cost functions realized by all possible $f_F \circ \hat{f}$ is given by

$$\mathcal{G}_F = \left\{ (f - f_F \circ \hat{f})^2 \mid f_F \in \mathcal{F}_F, \text{ and all corresponding } \hat{f} \right\}.$$

By noting that \hat{f} is fixed for any given $\mathcal{X} = \{X_1, X_2, \dots, X_l\}$, we decompose \mathcal{G}_F such that

$$\mathcal{G}_F = \bigcup_{\mathcal{X}} \left\{ (f - \mathcal{E}(\mathcal{X}))^2 : f_F \in \mathcal{F}_F \right\} = \bigcup_{\hat{f}} \left\{ (f - f_F \circ \hat{f})^2 : f_F \in \mathcal{F}_F \right\},$$

where $\mathcal{E}(\mathcal{X})(X) = (f_F \circ \hat{f})(X)$ for all X . From Vapnik [39] we have

$$P \left[I_F(\hat{f}_F) - I_F(f_F^*) > \epsilon \right] \leq P \left[\sup_{g \in \mathcal{G}_F} |P_n g - P g| > \epsilon/2 \right]$$

where $P g = \int g(X) dP_X$, $P_l g = \frac{1}{l} \sum_{i=1}^l g(X_i)$. Based on Pollard [28] (see Lemma 6.1 of the Appendix which states the formulation from [25]), the right hand side is upperbounded by $4E[N(\epsilon/32, d_l^g, \mathcal{G}_F)] e^{-n\epsilon^2/512}$, where d_l^g is defined as: for any $g', g'' \in \mathcal{G}_F$, and \mathcal{X}

$$d_l^g(g', g'') = \frac{1}{l} \sum_{i=1}^l |g'(X_i) - g''(X_i)|$$

which is a random variable.

We now bound $E[N(\epsilon/32, d_l^g, \mathcal{G}_F)]$ by using the bounds corresponding to the simpler family \mathcal{F}_F . Consider Part (i). For any fixed X_1, X_2, \dots, X_l , and any $g', g'' \in \mathcal{G}_F$, there exists a single \hat{f} , and $f'_F, f''_F \in \mathcal{F}_F$ such that $g' = (f - f'_F \circ \hat{f})^2$ and $g'' = (f - f''_F \circ \hat{f})^2$. Then, for this sample \mathcal{X} , we have

$$\begin{aligned}
d_l^g(g', g'') &= \frac{1}{l} \sum_{i=1}^l |g'(X_i) - g''(X_i)| \\
&\leq \sup_{i=1, \dots, l} |g'(X_i) - g''(X_i)| \\
&\leq \sup_{i=1, \dots, l} \left| [f'_F(\hat{f}(X_i)) - f''_F(\hat{f}(X_i))] [2f - f'_F(\hat{f}(X_i)) - f''_F(\hat{f}(X_i))] \right| \\
&\leq \sup_{i=1, \dots, l} 2 \left| f'_F(\hat{f}(X_i)) - f''_F(\hat{f}(X_i)) \right| \\
&\leq 2 \sup_y |f'_F(y) - f''_F(y)|,
\end{aligned}$$

where the third inequality follows since each $f_F \in \mathcal{F}_F$ maps to $[0, 1]$ and $f : \mathfrak{X}^d \mapsto [0, 1]$. This condition implies $d_l^g(g', g'') \leq 2d_\infty(f'_F, f''_F)$, which in turn implies

$$N(\epsilon/32, d_l^g, \mathcal{G}_F) \leq N(\epsilon/64, d_\infty, \mathcal{F}_F)$$

for all \mathcal{X} . Furthermore, since d_∞ is independent of X_1, X_2, \dots, X_l , we have

$$E[N(\epsilon/32, d_l^g, \mathcal{G}_F) | X_1, X_2, \dots, X_l] \leq N(\epsilon/64, d_\infty, \mathcal{F}_F).$$

Thus, applying Lemma 6.1 of Appendix to individual estimators, we have

$$\delta_i = 4N(\epsilon/64, d_\infty, \mathcal{F}_i) e^{-n\epsilon^2/512}.$$

For the fuser, we have

$$\begin{aligned}
E[N(\epsilon/32, d_l^g, \mathcal{G}_F)] &= \int E[N(\epsilon/32, d_l^g, \mathcal{G}_F) | X_1, X_2, \dots, X_l] dP_{X_1} dP_{X_2} \dots dP_{X_l} \\
&\leq \int E[N(\epsilon/64, d_\infty, \mathcal{F}_F) | X_1, X_2, \dots, X_l] dP_{X_1} dP_{X_2} \dots dP_{X_l} \\
&\leq \int N(\epsilon/64, d_\infty, \mathcal{F}_F) dP_{X_1} dP_{X_2} \dots dP_{X_l} \\
&\leq N(\epsilon/64, d_\infty, \mathcal{F}_F),
\end{aligned}$$

and thus we have $\delta_F = 4N(\epsilon/64, d_\infty, \mathcal{F}_F) e^{-n\epsilon^2/512}$. Then, the condition in Definition 2.2 is satisfied if

$$N(\epsilon, d_\infty, \mathcal{F}_F) \leq N(\epsilon, d_\infty, \mathcal{F}_1) + \dots + N(\epsilon, d_\infty, \mathcal{F}_N),$$

which proves Part (i).

For Part (ii), we first note that for any X_1, X_2, \dots, X_l , we have

$$N(\epsilon/32, d_l^g, \mathcal{G}_F) \leq N(\epsilon/64, \mathcal{F}_F)$$

since the bound is valid for the particular distribution generated by $\hat{f}(X_1), \hat{f}(X_2), \dots, \hat{f}(X_l)$. Furthermore, different samples correspond to different distributions, but the same bound is valid for all cases. Thus, $E [N(\epsilon/32, d_l^g, \mathcal{G}_F)] \leq N(\epsilon/64, \mathcal{F}_F)$. The rest of the proof is identical to Part (i). \square

This theorem specifies a simple recipe for the fuser design: choose a fuser class with the isolation property and a small cover size. In particular, linear combinations are a good choice if the number of estimators is small (see Section 4).

In essence, this theorem enables us to conclude the performance guarantee of the fuser by simply examining the cover sizes. The condition of Theorem 2.1 is trivially satisfied if the fuser has a smaller cover size compared to the minimum cover size among the estimators. In general, this condition can be expressed to that in terms of other parameters in specific cases. Such parameters include the basis size of the potential function method, number of terms in the linear combination fusers, and the number of nodes or bounds on the weights ⁴. Such derivations are illustrated in the next two sections.

3 Feedforward Networks

In this section, we consider the cases where sigmoid feedforward networks are employed as estimators. We show that in these cases the fuser design based on Theorem 2.1 is particularly simple, such as specifying a bound on the weights of a linearized class of networks.

We now consider a class of feedforward neural networks with a single hidden layer of h nodes and a single output node. The output of the network corresponding to input $x \in \mathbb{R}^d$ is given by $f_w(x) = \sum_{j=1}^h a_j \sigma(b_j^T x + t_j)$ where $w = (w_1, w_2, \dots, w_{h(d+2)})$ is the *weight vector* of the network consisting of $a_1, a_2, \dots, a_h, b_{11}, b_{12}, \dots, b_{1d}, \dots, b_{h1}, \dots, b_{hd}$ and t_1, t_2, \dots, t_h , and $\sigma(\cdot)$ is a given function. We denote the set of *sigmoidal networks* with bounded weights by

$$\mathcal{FS}_B = \{f_w : w \in [-B, B]^{h(d+2)}, \sigma(z) = \tanh^{-1}(\gamma z)\},$$

where $\gamma > 0$ is called the gain parameter. The sigmoidal neural networks are very popular and are extensively applied in function estimation problems [36, 22]. Since the sigmoidal neural networks do not have the isolation property, they do not readily yield a fuser specified by Theorem 2.1. They, however, form good candidates for the individual function estimators. We now consider that different network classes are used for the individual function estimators by specifying different values for h, γ and B .

For the fuser, we employ feedforward *piecewise linear networks* where $\sigma(z)$ is given by

$$\sigma^\tau(z) = \begin{cases} z & \text{if } z \in [0, \tau] \\ \tau & \text{if } z > \tau \\ 0 & \text{if } z < 0 \end{cases}$$

⁴Similar results can be derived for the case of unbounded weights based on the VC dimension bounds of Karpinski and Macintyre [18]. In all our motivating practical examples, the neural networks have bounded weights, for which simpler arguments yield the required cover sizes for function estimation problems [25, 4, 35, 32, 33].

for some bounded $\tau > 1$. We denote the set of these piecewise linear networks with bounded weights by \mathcal{FL}_B . These networks satisfy the isolation property since f^i in Definition 2.3 can be realized by choosing: $a_i = 1$, $a_j = 0$ for $j \neq i$; $t_j = 0$, for all j ; $b_{jk} = 1$, for $j = i$ and $k = i$, and $b_{jk} = 0$, otherwise. More general networks of this type have been studied by a number of researchers (for example, see [14]).

Consider that i th estimator minimizes empirical error over \mathcal{FS}_{B_i} , and let h_{\min} be the minimum number of hidden nodes among all estimators with bound $B = \min_{i=1}^N B_i$. Let the fuser be chosen from the class of piecewise linear networks with h_{\min} hidden nodes. For $\gamma = 1$, the fuser class can be obtained by computing the single parameter B_F as follows. From [25] we have

$$N(\epsilon, d_P, \mathcal{FS}_B) \leq \left(\frac{4e(h_{\min} + 1)h_{\min}}{\epsilon} \right)^{h(2d+3)+1}.$$

The same bound applies to \mathcal{FL}_B with d and B replaced by N and B_F , respectively (since σ^τ is a nondecreasing function with bounded variation; see [25, 27] for details). Then the condition in Theorem 2.1 can be satisfied by ensuring $N(\epsilon, \mathcal{FL}_{B_F}) \leq N(\epsilon, \mathcal{FS}_B)$ which specifies B_F as follows

$$B_F \leq \left(\frac{4e(h_{\min} + 1)}{\epsilon} \right)^{\frac{2h_{\min}(d-N)}{h_{\min}(2N+3)+1}} B^{\frac{h_{\min}(2d+3)+1}{h_{\min}(2N+3)+1}}.$$

Thus the fuser design simply involves satisfying a bound on the weights of the fuser class \mathcal{FL}_{B_F} .

We now consider a more restrictive case when X is chosen from $[0, 1]^d$. Similar computation can be carried out by using the bound [35], for $X \in [0, 1]^d$,

$$N(\epsilon, d_\infty, \mathcal{FS}_B) \leq \frac{2\gamma B^2 h}{\epsilon} e^{\left\{ \frac{\gamma B^2 h}{\epsilon} \left[\left(\frac{\gamma B^2 h}{\epsilon} \right)^{d-1} + 1 \right] \right\}},$$

and the bound for $N(\epsilon, d_\infty, \mathcal{FL}_{B_F})$ is obtained by using $d = N$, $\gamma = 1$, and $B = B_F$ in the above expression. The condition in this case can be shown to be

$$B_F \leq \left\{ \sqrt{\gamma} B, \epsilon / h_{\max} (\gamma B h_{\max} - 1)^{(d-1)/(N-1)} \right\}.$$

For $X \in [0, 1]^d$, we have another bound given by

$$N(\epsilon, d_\infty, \mathcal{FS}_B) \leq L_B^{h(d+2)} (1/\epsilon)^{h(d+2)},$$

where $L_B = \max(1, B\gamma^2/4)$ [32]. As in the previous case let $\gamma B = \min_{i=1}^N \gamma_i B_i$. By using estimators such that $\gamma \geq 2/\sqrt{B}$, we have $L_B = B\gamma^2/4$. Then for the fuser, we have $N(\epsilon, d_\infty, \mathcal{FS}_B) \leq (B_F/\epsilon)^{h_{\max}(N+2)}$. Then the condition of Theorem 2.1 is satisfied by the following condition

$$B_F \leq (B\gamma/4)^{\frac{d+2}{N+2}} (1/\epsilon)^{\frac{d-n}{N+2}}.$$

Since $\epsilon \leq 1$, if $d \geq N$, this condition is implied by the simpler condition $B_F \leq (B\gamma/4)^{\frac{d+2}{N+2}}$.

4 Vector Space Methods and Linear Combinations

The potential functions are extensively applied in the function estimation problems [2]. We can employ them as individual function estimators here, where each estimation is of the form $w_1\phi_1(x) + \dots + w_s\phi_s(x)$, for $w_i \in \mathfrak{R}$, $x \in \mathfrak{R}^d$, $\phi_i : \mathfrak{R}^d \mapsto \mathfrak{R}$, corresponding to the basis $\{\phi_1, \dots, \phi_s\}$. Different estimators can be obtained by varying the basis functions as well as the basis size. In each case, the estimator \hat{f}_i corresponds to the weight vector (w_1, \dots, w_s) that minimizes the empirical error. One of the attractive features of this method is that \mathcal{F}_i forms a vector space of dimensionality s , which yields a simple bound $N(\epsilon, d_P, \mathcal{F}_i) \leq 2 \left(\frac{2e}{\epsilon} \ln \frac{2e}{\epsilon}\right)^s$ (see Lemma 2 of Appendix, and also see [25]) which makes it very convenient to check the conditions of Theorem 2.1.

The general potential functions do not satisfy the isolation property, but a suitable fuser class can be obtained by choosing functions of the form $w_1\phi_1(b_1^T y) + \dots + w_N\phi_N(b_N^T y)$, for $w_i \in \mathfrak{R}$, $y \in [0, \tau]^N$, $\tau \geq 1$, and $b_i \in \mathfrak{R}^N$. We choose piecewise linear version of ϕ_i 's along the lines of $\sigma^T(\cdot)$ of the last section. Another way is to augment the basis of the potential functions with N additional linear functions, which ensures the isolation property at the cost of adding a factor of N to the dimensionality.

Linear combinations have been extensively used as fusers in various applications. For example, recent applications include combining neural network estimators [15], and regression estimators [?]. Since the linear combinations form a vector space, the bound in Lemma 2 of Appendix is also applicable here. Thus one of the interesting and useful cases is when individual estimators form vector spaces, and the fuser is a linear combination. Then the condition of Theorem 2.1 takes the following form

$$\left(\frac{2e}{\epsilon} \ln \frac{2e}{\epsilon}\right)^N \leq \sum_{i=1}^N \left(\frac{2e}{\epsilon} \ln \frac{2e}{\epsilon}\right)^{s_i},$$

where s_i is the dimensionality of the i th estimator. Thus we have the following direct but useful result.

Proposition 4.1 *For a linear combination fuser of N vector space estimators, the fuser is at least as efficient as the empirical best estimator if at least one of the learners has a dimensionality N .*

5 Simulation Results

We consider the special potential functions of the form

$$\left\{ \sum_{i=1}^s a_i e^{-b_i^T x} \mid a_i \in \mathfrak{R}, b_i \in \mathfrak{R}^5, x \in \mathfrak{R}^5 \right\}$$

to illustrate the performance of the fuser. The training and testing samples are generated by a particular function f of this type obtained by randomly generating the parameters s , a_i 's, and b_i 's from the integer ranges $[1, 11]$, $[1, 1000]$, and $[1, 1000]$, respectively. Then the inputs X_i 's are generated according to a uniform distribution on $[0, 1]^5$, and the function values are generated using f .

estimator	s_i	error on training data	error on testing data	fuser coefficients
Estimator #1	11	0.008156	0.006913	0.428581
Estimator #2	5	2.230803	1.958127	0.000937
Estimator #3	2	179.517059	167.371048	0.000003
Estimator #4	8	0.065237	0.055423	0.032205
Estimator #5	7	0.851890	0.796250	-0.008831
Estimator #6	6	0.017759	0.014399	0.547107
Fuser	–	0.000816	0.001155	–

Table 1: Percentage of data sets on which fuser has lower error than best individual estimator.

A number of estimators based on this special potential functions are computed to estimate the unknown function. For each individual estimator, s_i and b_i 's are randomly initialized (using same ranges as above), and then a_i 's are computed using the least squares method based on the sample. Fuser is a linear combination of the estimators computed using the outputs of the estimators corresponding to the training sample.

We now describe a simple data set and the corresponding results of our simulation (Table 1). Based on 1000 learning examples, the best estimator (Estimator #1) achieved mean square training error of 0.008156, and the fuser achieved a lower error given by 0.000816; note that the error achieved by the worst estimator is 179.517059. Based on 1000 testing examples, the mean square testing errors are 0.006913 and 0.001155, respectively, for the empirical best estimator and fuser. It is interesting to note that the magnitude of coefficients of the fuser are larger for the two of the estimators with low empirical error and lowest for the one with highest empirical error.

To understand the performance of the fuser at a higher-level we varied the number of learners and repeated the simulation for a number of data sets. There are 10,000 training examples and 10,000 testing examples in each data set. There are altogether 20 data sets. For each data set the errors committed by the fuser and the best individual estimator are computed based on the test set. The percentage of data sets for which the fuser achieved no more error than the best estimator is computed (see Table 2) as the number of learners is varied from 2 to 10. The fuser performed at least as good as the best estimator at 100% for $N \geq 6$.

Then a second set of individual estimators is obtained such that the basis size for each estimator is at least as high as the number of estimators, which ensures the condition of Proposition 4.1. The simulation was repeated on 20 data sets containing 10,000 training examples and 10,000 testing examples. The corresponding results are shown in Table 3. Notice the improved performance of the fuser in this case compared to the previous case, namely that the fuser performed better 100% for $N \geq 4$.

6 Conclusions

We considered the problem of fuser design for a set of function estimators each computed by minimizing empirical error over a sample. We showed that a fuser, trained with the estimator

number of estimators	percentage
2	81
3	72
4	90
5	84
6	100
7	100
8	100
9	100
10	100

Table 2: Percentage of data sets on which fuser has lower error than best individual estimator.

number of estimators	percentage
2	81
3	86
4	100
5	100
6	100
7	100
8	100
9	100
10	100

Table 3: Percentage of data sets on which fuser has lower error than best individual estimator. Each estimator has at least N basis elements.

outputs corresponding to the training sample, provides performance guarantee at least as good as the empirical best estimator, if it possesses isolation property and a smaller cover size. We derived sufficiency conditions in terms of two types of cover sizes. For the specific cases of feedforward neural networks, potential functions, and linear combinations, we derived simpler conditions in terms of the natural parameters such as weight bounds, basis sizes, etc. The main motivation for this formulation is its occurrence in practical applications in the areas of automatic target detection and robot sensor fusion.

Consider that we employ a fuser based on linear combinations of the available function estimators of the type considered here. Then the fused system is guaranteed to be at least as good as the best of the methods under the conditions considered here. If at a latter point, another function estimation method is devised, it can be easily integrated into the fused system by retraining the fuser. As a result, we have a system that whose guarantee is at least as good as the best available method at all times. Also, the computational problem of updating the fuser is a simple least squares estimation that can be solved using a number of available methods.

The results of this paper can be extended to the regression estimation problem. The main ideas of this paper can be applied for characterizations based on scale sensitive dimension, etc., which specify weaker conditions for function learning compared to the cover sizes considered here. The treatment of this paper is valid for deterministic function estimators only. It would be interesting to see if isolation property can be extended or is sufficient for the case of probabilistic estimators. It would be also of future interest to see if stronger results than just performing better than best learner can be shown in the present formulation. Note that here additional samples required for typical boosting algorithms [13] or the access to the estimator algorithms⁵ themselves are not available in our formulation. But methods such as bootstrapping and cross-validation can be carried out in our formulation. In certain cases, a fuser obtained by these methods might provide better guarantees compared to our method which just minimizes empirical error. Other areas of future research include the formulations of sufficient conditions in terms of properties alternative to the isolation property, and also the necessary conditions for the fuser to be better than the best estimator.

Appendix

We restate several known results here to facilitate the proof of Theorem 2.1 and discussion in Section 4.

Let Y_1, Y_2, \dots, Y_l denote the iid sample such that $Y_i \in A \subseteq \mathfrak{R}^M$, and $\vec{Y} = \{Y_1, Y_2, \dots, Y_l\}$. Let \mathcal{F} denote a class of functions defined on the domain A . We define $\mathcal{F}_{\vec{Y}} = \{ (f(Y_1), f(Y_2), \dots, f(Y_l)) : f \in \mathcal{F} \} \subseteq [0, 1]^l$. Consider the random variable representing the covering number $N(\epsilon, \hat{d}_l, \mathcal{F}_{\vec{Y}})$ where $\hat{d}_l : [0, 1]^l \times [0, 1]^l \mapsto [0, 1]$ is defined as $\hat{d}_l(x, z) = \frac{1}{l} \sum_{i=1}^l |x_i - z_i|$, for $x = (x_1, x_2, \dots, x_l)$ and $z = (z_1, z_2, \dots, z_l)$. This covering number plays an important role in the convergence of empirical means of functions to their expectations. We now state a result which is an adaptation of Pollard's result [28] by Lugosi and Zeger [25].

Lemma 6.1 *Let \mathcal{F} be a class of measurable functions from A into $[0, 1]$, and P be a probability*

⁵We are only given the function estimates which are outputs of these algorithms.

measure defined on A . Let Y_1, Y_2, \dots, Y_l be an iid sample according to P . Then

$$P \left\{ \sup_{f \in \mathcal{F}} |P_n f - P f| > \epsilon \right\} \leq 4E \left[N(\epsilon/16, \hat{d}_l, \mathcal{F}_{\vec{Y}}) \right] e^{-\epsilon^2 l / 128}$$

where $\vec{Y} = \{Y_1, Y_2, \dots, Y_l\}$, $P f = \int f(y) dP$ and $P_n f = \frac{1}{l} \sum_{i=1}^l f(Y_i)$.

In deriving finite sample results, one often utilizes a distribution-free bounds on $N(\epsilon, \hat{d}_l, \mathcal{F}_{\vec{Y}})$. One such method utilizes a cover size based on d_∞ ; since $\hat{d}_l(f_1, f_2) \leq d_\infty(f_1, f_2)$ for any $f_1, f_2 \in \mathcal{F}$, we have $N(\epsilon, \hat{d}_l, \mathcal{F}_{\vec{Y}}) \leq N(\epsilon, d_\infty, \mathcal{F})$. A second method utilizes a bound $N(\epsilon, d_P, \mathcal{F})$ that is valid for any distribution P on A ; since such bound is valid in particular for the specific distribution that places a mass of $1/l$ at each of Y_i 's, we have $N(\epsilon, \hat{d}_l, \mathcal{F}_{\vec{Y}}) \leq N(\epsilon, d_P, \mathcal{F})$.

If \mathcal{F} forms a vector space of dimensionality d , then its covering number can be upper-bounded as follows as a direct consequence of results of Cover [8] and Haussler [16].

Lemma 6.2 *Let \mathcal{F} denote d dimensional vector space of functions defined on A with range $[0, 1]$. Then for any probability measure P defined on A , we have*

$$N(\epsilon, d_P, \mathcal{F}) \leq 2 \left(\frac{2e}{\epsilon} \ln \frac{2e}{\epsilon} \right)^d \quad \text{and} \quad N(\epsilon, \hat{d}_l, \mathcal{F}_{\vec{Y}}) \leq 2 \left(\frac{2e}{\epsilon} \ln \frac{2e}{\epsilon} \right)^d.$$

References

- [1] M. A. Abidi and R. C. Gonzalez, editors. *Data Fusion in Robotics and Machine Intelligence*. Academic Press, New York, 1992.
- [2] M. A. Aizerman, E. M. Braverman, and L. I. Rozonoer. *Extrapolative problems in automatic control and method of potential functions*, volume 87 of *American Mathematical Society Translations*, pages 281–303. 1970.
- [3] N. Alon, S. Ben-David, N. Cesa-Bianchi, and D. Haussler. Scale-sensitive dimensions, uniform convergence, and learnability. In *Proc. of 1993 IEEE Symp. on Foundations of Computer Science*, 1993.
- [4] P. L. Bartlett. The sample complexity of pattern classification with neural networks: the sample size of the weights is more important than the size of the network. Technical report, Department of Systems Engineering, Australian National University, 1996.
- [5] J. M. Bates and C. W. J. Granger. The combination of forecasts. *Operations Research Quarterly*, 20:451–468, 1969.
- [6] N. Cesa-Bianchi, Y. Freund, D. P. Helmbold, D. Haussler, R. E. Schipire, and M. K. Warmuth. How to use expert advice. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on the Theory of Computing*, pages 382–391, 1993.
- [7] N. Cesa-Bianchi, Y. Freund, D. P. Helmbold, and M. K. Warmuth. On-line prediction and conversion strategies. *Machine Learning*, 25:71–110, 1996.
- [8] T. Cover. Geometric and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers*, EC-14:326–334, 1965.

- [9] B. V. Dasarathy. *Decision Fusion*. IEEE Computer Society Press, Los Alamitos, California, 1994.
- [10] P. S. de Laplace. Deuxième supplément à la théorie analytique des probabilités. 1818. Reprinted (1847) in *Oeuvres Complètes de Laplace*, vol. 7 (Paris, Gauthier-Villars) 531-580.
- [11] D. P. Foster and R. V. Vohra. A randomization rule for selecting forecasts. *Operations Research*, 41(4):704–709, 1993.
- [12] Y. Freund. Boosting a weak learning algorithm by majority. *Information and Computation*, 121:256–285, 1995.
- [13] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of Second European Conference on Computational Learning Theory*, pages 23–37. 1995.
- [14] P. W. Goldberg and M. R. Jerrum. Bounding the Vapnik-Chervonenkis dimension of concept classes parametrized by real numbers. *Machine Learning*, 18:131–148, 1995.
- [15] S. Hashem. Optimal linear combinations of neural networks. *Neural Networks*, 10(4):599–614, 1997.
- [16] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, 100:78–150, 1992.
- [17] D. Haussler, J. Kivinen, and M. K. Warmuth. Tight worst-case loss bounds for predicting with expert advice. In *Proceedings of Second European Conference on Computational Learning Theory*, pages 69–83. 1995.
- [18] M. Karpinski and A. Macintyre. Polynomial bounds for VC dimension of sigmoidal and general Pfaffian neural networks. *Journal of Computer and System Science*, 54:169–176, 1997.
- [19] M. J. Kearns and H. S. Seung. Learning from a population of hypotheses. *Machine Learning*, 18:255–276, 1995.
- [20] J. Kivinen and M. K. Warmuth. Using experts for predicting continuous outcomes. In *Computational Learning Theory: EuroCOLT'93*, pages 109–120, 1993.
- [21] A. Krzyzak, T. Linder, and G. Lugosi. Nonparametric estimation and classification using radial basis function nets and empirical risk minimization. *IEEE Transactions on Neural Networks*, 7(2):475–487, 1996.
- [22] W. S. Lee. *Agnostic learning and single hidden layer neural networks*. PhD thesis, Australian National University, 1996.
- [23] N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.
- [24] L. Ljung. *System Identification*. Prentice Hall, Engelwood Cliffs, NJ, 1987.
- [25] G. Lugosi and K. Zeger. Nonparametric estimation via empirical risk minimization. *IEEE Transactions on Information Theory*, 41(3):677–687, 1995.
- [26] R. N. Madan and N. S. V. Rao. Guest editorial on information/decision fusion with engineering applications. *Journal of Franklin Institute*, 336B(2), 1999. 199-204.
- [27] D. Nolan and D. Pollard. U-Processes: Rates of convergence. *Annals of Statistics*, 15(2):780–799, 1987.
- [28] D. Pollard. *Convergence of Stochastic Processes*. Springer-Verlag, New York, 1984.
- [29] D. Pollard. Asyptotics via empirical processes (with discussion). *Statistical Science*, 4:341–366, 1989.

- [30] B. L. S. Prakasa Rao. *Nonparametric Functional Estimation*. Academic Press, New York, 1983.
- [31] N. S. V. Rao. Fusion methods for multiple sensor systems with unknown error densities. *Journal of Franklin Institute*, 331B(5):509–530, 1994.
- [32] N. S. V. Rao. Fusion methods in multiple sensor systems using feedforward neural networks. *Intelligent Automation and Soft Computing*, 5(1):21–30, 1998.
- [33] N. S. V. Rao. Simple sample bound for feedforward sigmoid networks with bounded weights. *Neurocomputing*, 29:115–122, 1999.
- [34] 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.
- [35] N. S. V. Rao and V. Protopopescu. Function estimation by feedforward sigmoidal networks with bounded weights. *Neural Processing Letters*, 7:125–131, 1998.
- [36] V. Roychowdhury, K. Siu, and A. Orlicsky, editors. *Theoretical Advances in Neural Computation and Learning*. Kluwer Academic Pub., Boston, 1994.
- [37] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5:197–227, 1990.
- [38] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.
- [39] V. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, New York, 1982.
- [40] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.
- [41] M. Vidyasagar. *A theory of Learning and Generalization*. Springer-Verlag, New York, 1997.
- [42] 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.
- [43] V. G. Vovk. Aggregating strategies. In *Proceedings of Workshop on Computational Learning Theory*, pages 371–383, 1990.