DØNote 1606

Some Mathematical Comments on Feed-Forward Neural Networks

Harrison B. Prosper
Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510
(25-Jan-1993)

Abstract

In the past two years or so a clearer understanding of the correct interpretation of feed-forward neural networks has emerged. When properly understood, neural networks have a very clear meaning: their outputs are approximations to functions of certain probability densities. In this note, I review this fact and I show how a feed-forward neural network can be used to approximate likelihood functions and thereby provide a sound basis for multi-dimensional analysis.

I. INTRODUCTION

Neural networks are the basis of a powerful new paradigm for multi-dimensional data analysis and are being used routinely to solve very hard problems [1]. Unfortunately, they have the reputation of being mysterious "black boxes". The fact is, however, that neural networks (of the feed-forward type) are being used, with great success, in many areas of research; in particular, they are being used as an effective analysis tool in the field of high energy physics [2]. The purpose of this note is to review some mathematical aspects of neural networks in the hope that this will render this putative "black box", if not exactly transparent, then at least less opaque!

A general feed-forward neural network is a function which maps a vector of n real-valued inputs $\mathbf{X} = (x_1, \ldots, x_n)$ into a vector of m real-valued outputs. Here, I shall consider networks having one output only. The success of neural network techniques is based upon the fact that it is possible to train a network to encode mappings, whose functional form is unknown, by repeatedly presenting to the network known inputs associated with known outputs.

Let $F(\mathbf{X}, \mathbf{\Omega})$ represent the output value of a single-output feed-forward neural network. The symbol $\mathbf{\Omega}$ represents a vector $(\omega_1, \ldots, \omega_k)$ of k parameters called *weights* whose values are to be chosen so that $F(\mathbf{X}, \mathbf{\Omega})$ encodes the desired mapping. Much neural network research is aimed at trying to understand what is the optimal value of k and in devising strategies for finding the optimal values of the weights. For the feed-forward network the basic strategy is to minimise the function

$$E_N(F) = \frac{1}{N}\sum_{i=1}^{N}(F(\mathbf{X}_i, \mathbf{\Omega}) - D_i)^2 \tag{1.1}$$

with respect to the weights $\mathbf{\Omega} = (\omega_1, \ldots, \omega_k)$, where $\{\mathbf{X}_i\}$ is a set of N input vectors and the quantity D_i represents the known output associated with the vector $\mathbf{X}_i$. The vectors $\{\mathbf{X}_i\}$ are called *feature vectors*; they populate a vector space called, appropriately, *feature space*. In high energy physics research the feature vectors would be ntuples of quantities which characterise events. These vectors are distributed according to some n-dimensional probability distribution function, which, in general, is unknown.

An important first step in the analysis of high energy physics data is to separate the events of interest, that is, the signal S, from events which are considered to be the background B. This is the topic addressed in this note. The classes S and B are assumed to be mutually exclusive and their union equal to the complete sample space.

The standard technique is to apply cuts to the components of the feature vectors and to classify events according to whether they pass or fail the cuts. If one is fortunate enough to have found quantities which separate cleanly signal events from the background then the standard technique is simple and works well. In general, however, it is extremely difficult to find such quantities; there is usually considerable overlap between the 1-dimensional signal and background distributions. However, if one were to work directly in the n-dimensional feature space one anticipates being able to do a much better job of event classification because one can exploit the correlations between the feature vectors. Indeed, it may well be that the distributions in feature space can be adequately separated even if the problem seems hopeless using the standard technique. The problem, of course, is to construct a

hyper-surface (called a *decision boundary*) which partitions, adequately, the feature space into signal and background regions. This is a problem which is tailor-made for feed-forward neural networks.

II. BUILDING A DECISION BOUNDARY

In principle, given the probability density functions $P(\mathbf{X}|S)$ and $P(\mathbf{X}|B)$ for the signal and the background events, respectively, one can solve the event classification problem in an optimal way. Let $P(S)$ be the *prior* probability of an event being a signal event and let $P(B)$ be the corresponding quantity for the background events. The ratio $P(S)/P(B)$ is, typically, equal to the ratio of signal to background cross-sections. The prior probabilities satisfy the contraint: $P(S) + P(B) = 1$. From the likelihood functions $P(\mathbf{X}|S)$ and $P(\mathbf{X}|B)$ and the prior probabilities one can construct the new probability densities

$$P(S|\mathbf{X}) = P(\mathbf{X}|S)\, P(S) \,/\, P(\mathbf{X}), \tag{2.1}$$

and

$$P(B|\mathbf{X}) = P(\mathbf{X}|B)\, P(B) \,/\, P(\mathbf{X}), \tag{2.2}$$

where

$$P(\mathbf{X}) = P(\mathbf{X}|S)\, P(S) \,+\, P(\mathbf{X}|B)\, P(B), \tag{2.3}$$

which follow from Bayes' theorem on conditional probabilities. The densities $P(S|\mathbf{X})$ and $P(B|\mathbf{X})$ are sometimes referred to as *posterior densities*; $P(S|\mathbf{X})$ means the probability density that an event is a signal event S, given the measured feature vector $\mathbf{X}$. This is precisely the probability of interest for event classification.

An intuitively reasonable way to construct a decision boundary, and thereby partition the feature space, is to solve the equation

$$\begin{aligned} R &= \frac{P(S|\mathbf{X})}{P(B|\mathbf{X})}, \\ &= \frac{P(\mathbf{X}|S)\, P(S)}{P(\mathbf{X}|B)\, P(B)}, \end{aligned} \tag{2.4}$$

where R is called the *Bayes discriminant function*. It can be shown, in fact, that a cut on R is the optimal way to classify events: if $R > 1$ the event is called a signal event because that is more likely to be true, while if $R < 1$ it is classified as a background event. One notes that when $P(S) = P(B)$ R reduces to the likelihood ratio. It is often more convenient to use the quantity

$$\begin{aligned} f &= \log R, \\ &= \log P(\mathbf{X}|S) \,-\, \log P(\mathbf{X}|B) + \log\left[P(S)/P(B)\right], \end{aligned} \tag{2.5}$$

which can be interpreted as a generalisation of Fisher's linear discriminant function.

When the likelihood functions can be represented as n-dimensional gaussians one can write an explicit formula for f, namely:

$$\begin{aligned} f = & \frac{1}{2}(\mathbf{X} - <\mathbf{X}>)^T H_B (\mathbf{X} - <\mathbf{X}>) \\ & - \frac{1}{2}(\mathbf{X} - <\mathbf{X}>)^T H_S (\mathbf{X} - <\mathbf{X}>) \\ & + \frac{1}{2}\log\left(\frac{detH_B^{-1}}{detH_S^{-1}}\right) \\ & + \log\left(\frac{P(S)}{P(B}\right), \end{aligned} \tag{2.6}$$

where H_S and H_B are the H-matrices (that is, the inverse of the covariance matrices of the vectors $\mathbf{X}$) for the signal and background, respectively, and $detH_S^{-1}$ and $detH_B^{-1}$ are the determinants of the corresponding covariance matrices. Fisher's linear discriminant function is obtained by setting $P(S) = P(B)$ and $H_S = H_B = H$.

In general, the likelihood functions are very complicated (that is, non-gaussian) and, moreover, their functional forms are usually not known. In the next section I show how neural networks can be used to approximate these functional forms.

III. LEARNING LIKELIHOOD FUNCTIONS

It has been emphasised recently [3] that the output of a neural network can be interpreted as a Bayesian probability. This is a very significant finding, which turns out to be easy to prove. (The proof given here is a slight generalisation of that given in reference ([3]) for a single-output network.)

One can divide the function $E_N(F)$, given in Eq. (1.1), into two sums:

$$\begin{aligned} E_N(F) = & \frac{N_S}{N}\frac{1}{N_S}\sum_S^{N_S}(F - s)^2 \\ & + \frac{N_B}{N}\frac{1}{N_B}\sum_B^{N_B}(F - b)^2, \end{aligned}$$

that is, a sum over N_S signal events and a sum over N_B background events, where $N = N_S + N_B$. For signal events we have set $D = s$ and for the background D is set to b. One assumes that s and b are known. Consider, now the limit of $E_N(F)$ as $N \to \infty$ and call it $E(F)$. The fractions N_S/N and N_B/N go over to $P(S)$ and $P(B)$, respectively, while each sum goes over to an integral whose measure is determined by the distribution of feature vectors. That is, one can write

$$\begin{aligned} E(F) = & P(S)\int d\mathbf{X}\, P(\mathbf{X}|S)\,(F - s)^2 \\ & + P(B)\int d\mathbf{X}\, P(\mathbf{X}|B)\,(F - b)^2. \end{aligned} \tag{3.1}$$

Using equations (2.1) - (2.3) for the functions $P(S|\mathbf{X})$, $P(B|\mathbf{X})$ and $P(\mathbf{X})$ Eq. (3.1) can be transformed to

$$E(F) = \int d\mathbf{X}\, P(\mathbf{X})\, \left[F^2 \,-\, 2FG(\mathbf{X}, s, b)\right] + \int d\mathbf{X}\, P(\mathbf{X})\, \left[s^2\, P(S|\mathbf{X}) \,+\, b^2\, P(B|\mathbf{X})\right], \tag{3.2}$$

where the function $G(\mathbf{X}, s, b)$ is defined as

$$G(\mathbf{X}, s, b) \equiv s\, P(S|\mathbf{X}) \,+\, b\, P(B|\mathbf{X}). \tag{3.3}$$

By completing the square in the expression for $E(F)$ and re-arranging terms one obtains the important result

$$E(F) = \mathbf{E}\left[(F \,-\, G)^2\right] + \mathbf{E}\left[(s \,-\, G)(G \,-\, b)\right] \tag{3.4}$$

where the expectation operator, $\mathbf{E}\left[*\right]$, is defined by

$$\mathbf{E}\left[*\right] \equiv \mathbf{E}_S\left[*\right] \,+\, \mathbf{E}_B\left[*\right], \tag{3.5}$$

and

$$\mathbf{E}_S\left[Q\right] \equiv P(S) \int d\mathbf{X}\, P(\mathbf{X}|S)\, Q(\mathbf{X}),$$
$$\mathbf{E}_B\left[Q\right] \equiv P(B) \int d\mathbf{X}\, P(\mathbf{X}|B)\, Q(\mathbf{X}). \tag{3.6}$$

The results presented in reference ([3]) are obtained by setting $(s, b) = (1, 0)$ or $(s, b) = (1, -1)$ in Eq. (3.4).

The significance of Eq. (3.4) is that if a function $F(\mathbf{X}, \mathbf{\Omega})$ with sufficient functional flexibility can be found and given a large enough value for the number of signal and background events N, a minimisation of $E_N(F)$ leads to the solution

$$F(\mathbf{X}, \mathbf{\Omega}) = G(\mathbf{X}, s, b) \tag{3.7}$$

with the absolute minimum of $E(F)$, $min\, E(F)$, given by

$$min\, E(F) = \mathbf{E}\left[(s \,-\, G)(G \,-\, b)\right]. \tag{3.8}$$

This result shows that F is an approximation to a linear function of the densities $P(S|\mathbf{X})$ and $P(B|\mathbf{X})$. Noting that $P(S|\mathbf{X}) + P(B|\mathbf{X}) = 1$ one can write

$$P(S|\mathbf{X}) = (G \,-\, b)/(s \,-\, b), \tag{3.9}$$

$$P(B|\mathbf{X}) = (s \,-\, G)/(s \,-\, b), \tag{3.10}$$

while the Bayes discriminant function R is given by

$$R(G) = (G \,-\, b)/(s \,-\, G). \tag{3.11}$$

Approximations to $P(S|\mathbf{X})$, $P(B|\mathbf{X})$ and R are obtained by the replacement $G \rightarrow F$ in equations (3.9) to (3.11).

For some applications a knowledge of the quantities $P(S|\mathbf{X})$, $P(B|\mathbf{X})$ and R is sufficient. However, it is often useful to have an efficient way of summarizing a set of data. One way to do so is to specify the likelihood function, $P(\mathbf{X}|S)$, which describes the distribution of vectors $\mathbf{X}$. Given the likelihood function all probabilistic questions about the set of data can be answered. Unfortunately, it is not possible to extract the likelihood function from $P(S|\mathbf{X})$; at best one can extract the likelihood ratio: $P(\mathbf{X}|S)/P(\mathbf{X}|B)$. That fact, however, provides a clue as to how one might proceed.

Suppose that one were to set $P(B|\mathbf{X}) = 1/V$ where V is the volume of the feature space, that is, one takes $P(B|\mathbf{X})$ to be a *uniform* distribution over the feature space and one takes $P(S) = P(B)$, then the Bayes discriminant function reduces to

$$R = V\, P(\mathbf{X}|S). \tag{3.12}$$

If one now replaces G by the neural network output, F, one can approximate the likelihood function $P(\mathbf{X}|S)$ using the result

$$\begin{aligned} P(\mathbf{X}|S) &\approx R(F)/V, \\ &= \frac{1}{V}\frac{(F\,-\,b)}{(s\,-\,F)}. \end{aligned} \tag{3.13}$$

In summary: the pattern classification problem can be solved provided that 1) N is large enough and 2) a function F exists with the right property. An important question is: Do such functions exist? A very important theorem due to Kolmogorov [4] states that any real function of n real variables can be built from a superposition of several real functions of a single real variable. The importance of a feed-forward neural network is that it is an example of such a Kolmogorov function which by using special techniques, such as back-propagation [1], to minimise $E_N(F)$ can lead to good approximations to $G(\mathbf{X}, s, b)$.

IV. MEASURES OF GOODNESS-OF-FIT

How can one quantify the goodness-of-fit of $F(\mathbf{X}, \mathbf{\Omega})$ to $G(\mathbf{X}, s, b)$? At the moment I can give only a partial answer. Let us define the quantity

$$\epsilon \equiv F(\mathbf{X}, \mathbf{\Omega})\, -\, G(\mathbf{X}, s, b), \tag{4.1}$$

and write $E(F)$ as

$$E(F) = \mathbf{E}\,[\epsilon^2]\, +\, min\, E(F). \tag{4.2}$$

Now, since F is known one can compute all of its moments; in particular, one can compute $\mathbf{E}\,[F]$ and $\mathbf{E}\,[F^2]$. Many interesting, calculable, quantities can be derived; here are a few:

$$\mathbf{E}\,[G] = s\, P(S)\, +\, b\, P(B), \tag{4.3}$$

$$2\mathbf{E}\,[\epsilon\, F] = E(2F)\, -\, E(F)\, -\, \mathbf{E}\,[F^2], \tag{4.4}$$

$$\begin{aligned} Y &\equiv \mathbf{E}\,[G^2]\, -\, \mathbf{E}\,[\epsilon^2], \\ &= \mathbf{E}\,[F^2]\, -\, 2\mathbf{E}\,[\epsilon F], \end{aligned} \tag{4.5}$$

$$\begin{aligned} Z &\equiv \mathbf{E}\,[G^2]\, + min\, E(F), \\ &= Y\, +\, E(F). \end{aligned} \tag{4.6}$$

The quantity $E(2F)$ is calculated by the replacement $F \to 2F$ in the expression for $E(F)$.

A possible measure of goodness-of-fit might be

$$\begin{aligned} \eta' &= \mathbf{E}\,[\epsilon\, F]/\mathbf{E}\,[F^2], \\ &= 1 - \mathbf{E}\,[FG]/\mathbf{E}\,[F^2]. \end{aligned} \tag{4.7}$$

Unfortunately, that measure is not completely satisfactory: while $\eta' = 0$ is a necessary condition for a perfect fit it is not a sufficient condition because $\mathbf{E}\,[FG]$ close to $\mathbf{E}\,[F^2]$ does not imply that F is close to G. What is needed is a suitable function involving $\mathbf{E}\,[\epsilon^2]$. The reason is that $\mathbf{E}\,[\epsilon^2] = 0 \Rightarrow \epsilon = 0$ and, therefore, $F = G$. An ideal measure would be, for example,

$$\begin{aligned} \eta &= 1 - Y/\mathbf{E}\,[G^2], \\ &= \mathbf{E}\,[\epsilon^2]/\mathbf{E}\,[G^2]. \end{aligned} \tag{4.8}$$

The difficulty, however, is to find an adequate approximation to the quantity $\mathbf{E}\,[G^2]$.

To that end consider the expression for $G(\mathbf{X}, s, b)$ given in Eq. (3.3). From Eq. (3.9) one can write

$$G = (s\, -\, b)\, P(S|\mathbf{X})\, +\, b, \tag{4.9}$$

and, therefore,

$$\begin{aligned} \mathbf{E}\,[G^2] &= \int d\mathbf{X}\, P(\mathbf{X})\, G^2 \\ &= (s-b)^2 \int d\mathbf{X}\, P(S|\mathbf{X})\, P(S|\mathbf{X})\, P(\mathbf{X}) \\ &+ 2b(s-b) \int d\mathbf{X}\, P(S|\mathbf{X})\, P(\mathbf{X}) \\ &+ b^2 \int d\mathbf{X}\, P(\mathbf{X}). \end{aligned} \tag{4.10}$$

Now, using the basic relation $P(\mathbf{X}|S)P(S) = P(S|\mathbf{X})P(\mathbf{X})$ and the sum rules $\int d\mathbf{X} P(\mathbf{X}|S) = 1$ and $\int d\mathbf{X} P(\mathbf{X}) = 1$, the expression for $\mathbf{E}\,[G^2]$ can re-written as

$$\mathbf{E}\,[G^2] = (s-b)\, \mathbf{E}_S\,[G] + b(s-b)\, P(S) + b^2. \tag{4.11}$$

Unfortunately, one can not proceed unless the value of the quantity $\mathbf{E}_S\,[G]$ is known. One could try to replace G by F in the above. Let Q be defined by

$$Q \equiv (s-b)\, \mathbf{E}_S\,[F] + b(s-b)\, P(S) + b^2. \tag{4.12}$$

If one now writes $F = G + \epsilon$ one obtains the result

$$Q = \mathbf{E}\,[G^2]\, +\, (s-b)\, \mathbf{E}_S\,[\epsilon]. \tag{4.13}$$

It remains to be seen whether or not the quantity Q is an adequate approximation to $\mathbf{E}\,[G^2]$. That problem is being investigated. At any rate, substituting Q for $\mathbf{E}\,[G^2]$ in the expression for η leads to

$$\eta = ((s-b)\, \mathbf{E}_S\,[\epsilon]\, +\, \mathbf{E}\,[\epsilon^2])/((s-b)\, \mathbf{E}_S\,[\epsilon]\, +\, \mathbf{E}\,[G^2]). \tag{4.14}$$

V. CONCLUSIONS

I hope to have shown that feed-forward neural networks are useful mathematical entities whose interpretation is simple and unambiguous. The reputation of being a "black box" arises from the attempt, by some, to understand the meaning of the weights. While this may be a worthwhile effort for networks having simple structures, I would contend that trying to understand what the weights mean in what is, after all, a highly non-linear function of many variables is probably futile, and, is in any case, irrelevant to the purpose of fitting F to G. What is relevant is to understand how well the fit has been done so that one can estimate the systematic uncertainties.

REFERENCES

[1] For an excellent introduction to neural networks see: R. Beale and T. Jackson, *Neural Computing: An Introduction* (Adam Hilger, New York, 1991).

[2] A. De Angelis and P. Del Giudice, "Neural Networks for Physics Analysis in DELPHI", presented at the *Second Workshop on Artificial Neural Networks: From Biology to High Energy Physics*, Marciana Marina, 18-26 June 1992. To be published in the Proceedings.

[3] D.W. Ruck et. al., "The multilayer perceptron as an approximation to a Bayes optimal discriminant function", IEEE Trans. Neural Networks **1 (4)**, 296-298 (1990); E.A. Wan, "Neural network classification: a Bayesian interpretation", IEEE Trans. Neural Networks **1 (4)**, 303-305 (1990).

[4] A.N. Kolmogorov, "On the representation of continuous functions of several variables by a superposition of continuous functions of one variable and addition", Doklady Akademii Nauk, **SSSR 108**, 179-182 (1956). For a recent proof see, for example: E.K. Blum and L.K. Li, "Approximation theory and feedforward networks", Neural Networks, **4**, 511-515 (1991).