Importance Sampling for Objective Function Estimations in Neural Detector Training Driven by Genetic Algorithms

Raúl Vicen-Bueno · M. Pilar Jarabo-Amores · Manuel Rosa-Zurera · José L. Sanz-González · Saturnino Maldonado-Bascón

Abstract To train Neural Networks (NNs) in a supervised way, estimations of an objective function must be carried out. The value of this function decreases as the training progresses and so, the number of test observations necessary for an accurate estimation has to be increased. Consequently, the training computational cost is unaffordable for very low objective function value estimations, and the use of Importance Sampling (IS) techniques becomes convenient. The study of three different objective functions is considered, which implies the proposal of estimators of the objective function using IS techniques: the Mean-Square error, the Cross Entropy error and the Misclassification error criteria. The values of these functions are estimated by IS techniques, and the results are used to train NNs by the application of Genetic Algorithms. Results for a binary detection in Gaussian noise are provided. These results show the evolution of the parameters during the training and the performances of the proposed detectors in terms of error probability and Receiver Operating Characteristics curves. At the end of the study, the obtained results justify the convenience of using IS in the training.

R. Vicen-Bueno · M. P. Jarabo-Amores · M. Rosa-Zurera (✉) · S. Maldonado-Bascón
Signal Theory and Communications Department, Superior Polytechnic School,
University of Alcalá , Ctra. Madrid-Barcelona, km. 33.1, 28805 Alcalá de Henares, Madrid, Spain
e-mail: manuel.rosa@uah.es

R. Vicen-Bueno
e-mail: raul.vicen@uah.es

M. P. Jarabo-Amores
e-mail: mpilar.jarabo@uah.es

S. Maldonado-Bascón
e-mail: saturnino.maldonado@uah.es

J. L. Sanz-González
Signals, Systems and Radiocommunications Department, ETSI de Telecomunicación,
Polytechnic University of Madrid, Ciudad Universitaria, 28040 Madrid, Spain
e-mail: jlsanz@gcs.ssr.upm.es

Keywords Genetic algorithms · Neural networks · Monte Carlo · Importance sampling · Mean-Square error · Cross Entropy error · Misclassification error

1 Introduction

Standard Monte Carlo simulations present several limitations to accurately evaluate a statistical process or an objective/error function, especially when rare events are present. In this way, some limitations of the Monte Carlo simulation for some functional spaces that define the regularity of the input data are exposed in [1]. Nevertheless, an accurate estimation of a process or function can be achieved by the use of importance sampling (IS) techniques [2,3]. These techniques have been successfully applied in different kind of learning structures. For instance, in [4], the way IS is used in modern Markov chain machine learning is introduced in a general way. On the other hand, in [5,6] the use of IS in probabilistic neural networks (NNs) is exposed. Both papers present the main advantages of using IS for estimating the error during training in probabilistic NNs, where the improvement of training speed against traditional training methods is studied. IS techniques have also been successfully used for improving both the performance and the design speed of Bayesian networks [7]. The previous works used IS in supervised training/design of systems, but IS can also be used in unsupervised training. Thus, in [8], a stochastic gradient learning algorithm based on IS techniques for unsupervised learning of over-complete dictionaries is presented. As in the previous works [1–7], it is shown in [8] that the proposed algorithm is faster and more efficient than classical ones. Moreover, its great efficiency allows the treatment of large-scale problems in a statistically sound framework, as demonstrated for the extraction of individual piano notes from a polyphonic piano recording. Finally and independently of the kind of used system, in [9] a system dimensionality study is done when IS is used, paying special attention to the IS influence in the system performance and training speed. Moreover, it is shown how the training convergence time in NN training by IS for learners (NNs) with many parameters (weights) can be reduced by competitive, cooperative and concurrent reinforcement strategies. Finally, it is important to note that, in all the previously exposed studies, the computational cost needed to implement the IS technique is lower than the computational cost of using a huge quantity of data to design systems that are able to process rare events.

Due to the advantages of the use of IS in system design, this paper is focused on the comparative study of the application of IS techniques to train NNs in a supervised way using different objective functions. As IS is a modified Monte Carlo technique, it can be applied to rare event computation in different applications: performance analysis of radar and communications detectors [10–18], or rare event simulation in finance [19,20]. In communication detectors [10–15], the error probability (P_e) can be estimated by IS techniques for very low P_e values. In radar detectors [16–18], very low false-alarm probabilities (P_{fa}) can also be estimated by IS techniques. Considering NN-based detectors, IS has been applied in [16] to estimate the performance in terms of P_{fa} estimations in the testing phase, but without taking into consideration the application of IS techniques for training. In financial applications, an adaptive search algorithm has been developed recently [19] for parameter estimation of a mixture of Student's t density, in order to be used as IS probability density function. This algorithm, named Adaptive Mixture of t(AdMit) has been extended in [20] for efficient computation of the risk measures Value at Risk (VaR) and Expected Shortfall (ES).

Other works [21–23], which considered the use of NNs to approximate communication and radar detectors, have highlighted the poor performance of such detectors for low P_e and P_{fa} values. NN-based detectors have been compared to the Neyman–Pearson optimum

detector in [22]. The theoretical explanation has been recently published in [23], demonstrating that supervised adaptive learning machines trained to minimize the sum of squares error approximates the optimum Neyman–Pearson detector. The poor performance for very low P_{fa} can be explained from the point of view of training. Rare events hardly influence the estimation of the error function for training. So, the approximation of the boundary of the decision regions is not good enough where rare events occur. Therefore, IS techniques can be used to estimate with high accuracy the error functions during the NN training.

The application of IS techniques to NN training has been previously applied in the design of NN-based detectors in communication applications [24,25]. Both works considered the use of the Mean-Square (MS) error criterion for NN training. In [24], a suboptimal probability density function for IS is proposed. A genetic algorithm is used for training, and MS error is estimated during training using IS. A more advanced approach appears in [25], where the suboptimal probability density function depends on a parameter, the value of which is adapted during training with a genetic algorithm to minimize the variance of MS error estimation. On the other hand, the application of IS techniques in NN-based radar detectors is proposed in [26], using both the MS and CE error criteria, but without considering adaptive algorithms during training in order to optimize the considered probability density function (pdf) for IS.

As exposed in the above mentioned works [24–26] in order to apply IS techniques in NN training, the error function must be adequately modified in the training phase by finding an appropriate pdf. This pdf needs to ensure the error estimator is unbiased and has a reduced variance, which should vanish as the number of training samples tends to infinity (consistency). In [27], the problem of not taking into account the variance of the value function estimators in existing off-policy methods is studied. This absence makes their performance tend to be unstable. To cope with this problem, the use of an adaptive IS technique is proposed, which allows them to actively control the trade-off between bias and variance. They further provide a method for optimally determining the trade-off parameter based on a variant of cross-validation.

Our paper deals with the application of IS techniques to train NN-based detectors. Three different error criteria are considered and compared: MS, Cross Entropy (CE), and Misclassification (MC). An important novelty is the proposal of the optimal pdf for each error criterion that are not realistic, unfortunately. In order to avoid it, a suboptimal parametric pdf for IS is proposed. The parameters are adapted during training with a genetic algorithm (GA) in order to avoid bias and minimizing the variance of the estimated error. The algorithm for parameter optimization is called Adaptive Search Algorithm (ASA) and is described in this paper with detail for the first time. Computer simulations have been carried out, which show the convenience of using IS techniques for training NNs, which allows stopping the training when both the error of the estimated error function and the variance of its estimation are low. In order to assess the validity of the proposed technique, different experiments have been carried out: training neural networks with the three error functions and the proposed technique, varying the number of patterns considered in the training set (Sect. 5.1) and varying the NN size (Sect. 5.2).

In order to define the notation, we refer to Fig. 1, where $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ is the input vector of the R^n-space, $y = g(\mathbf{x})$ is the scalar output, $g(\cdot)$ is a nonlinear system (e.g. a NN), T_0 is the detection threshold and $z = u(g(\mathbf{x}) - T_0)$ is the detector output, where $u(\cdot)$ is the unit-step function, (i.e., $u(t) = 1$ if $t > 0$ and $u(t) = 0$ if $t < 0$). We denote $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ as a random vector and $f_{\mathbf{X}}(\mathbf{x}|H_i)$ as the pdf of $\mathbf{X}$ under a hypothesis H_i, $i = 1, 0$ (binary hypotheses), where H_0 is the null hypothesis or symbol "0" and H_1 is the alternative hypothesis or symbol "1". $P(H_i)$ is the "a priori" probability of the hypothesis

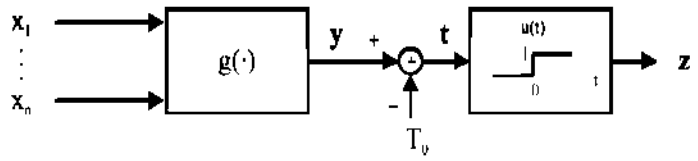

Fig. 1 The binary detector structure, where $g(\cdot)$ is a NN

H_i $(i = 1, 0)$, and $P\left(D_j|H_i\right)$ is the conditional probability of deciding H_j $(j = 1, 0)$ under the true hypothesis H_i $(i = 1, 0)$. If $g(\mathbf{x}) > T_0$ (or $z = 1$), the decision is H_1, but if $g(\mathbf{x}) < T_0$ (or $z = 0$), the decision is H_0. Moreover, for binary hypothesis the probability $P(D_1|H_0)$ is defined as the false-alarm probability (P_{fa}), and the probability $P(D_1|H_1)$ is defined as the probability of detection (P_d). Finally, $\mathrm{E}\{Z|H_i\}$ is the expectation of the random variable Z conditioned by H_i $(i = 1, 0)$, and $\mathrm{E}\{g(\mathbf{X})\}$ is the expectation of $g(\cdot)$ with respect to the pdf of $\mathbf{X}$ (i.e. $f_{\mathbf{X}}(\mathbf{x})$).

The paper is organized as follows. Section 2 presents the used IS technique for parameter estimation. Section 3 contains the algorithm proposed to adaptively estimate the best parameters of the parametric suboptimal pdf used in the NN training. Section 4 exposes the three different objective functions used in the IS-based NN training. Moreover, the use of the above-mentioned parametric suboptimal pdf is used for sampling the process according to these objective functions during the NN training. Section 5 presents the results of the NN training driven by GA and adaptive IS parameter estimation when the three different objective functions are used. According to them, the performance of the proposed detectors, in terms of P_e and Receiver Operating Characteristics (ROC) curves, are presented and discussed. Finally, Sect. 6 summarises the main conclusions of the study of the proposed way to train NNs by the use of IS techniques and GAs.

2 Importance Sampling Technique

In order to introduce the basic concepts of the IS technique used in this paper, and considering the notation exposed in Sect. 1, suppose that a parameter E can be expressed as follows

$$E = \int_{R^n} e(\mathbf{x})\, d\mathbf{x} \tag{1}$$

where $e(\mathbf{x}) \geq 0, \forall \mathbf{x} \in R^n$.

Now, consider a pdf $f_{\mathbf{x}}^*(\mathbf{x})$, in such a way that $f_{\mathbf{x}}^*(\mathbf{x}) \neq 0$ wherever $e(\mathbf{x}) \neq 0, \forall \mathbf{x} \in R^n$. Then from (1), we can write the following identities

$$E = \int_{R^n} e(\mathbf{x})\, d\mathbf{x} = \int_{R^n} \frac{e(\mathbf{x})}{f_{\mathbf{x}}^*(\mathbf{x})} f_{\mathbf{x}}^*(\mathbf{x})\, d\mathbf{x} = \mathrm{E}^*\left\{\frac{e(\mathbf{X})}{f_{\mathbf{x}}^*(\mathbf{X})}\right\} \tag{2}$$

where $\mathrm{E}^*\{\cdot\}$ stands for mathematical expectation with respect to the pdf $f_{\mathbf{x}}^*(\mathbf{x})$.

As is well known, a good estimator of the statistical mean is the sample mean. Then, from the last equality in (2), an estimator of E is given by

$$\hat{E}^* = \frac{1}{N} \sum_{k=1}^{N} \frac{e\left(\mathbf{x}_k^*\right)}{f_{\mathbf{x}}^*\left(\mathbf{x}_k^*\right)} \tag{3}$$

where $\mathbf{x}_k^*, k = 1, 2, \ldots, N$, are independent sample vectors whose pdf's are $f_{\mathbf{x}}^*(\mathbf{x})$ (this pdf is known as the Importance Sampling pdf or the biasing pdf). The estimator $\hat{E}^*$, given by (3), must be computed in order to perform the NN training, i.e., in order to find the NN parameters that minimize the objective function E.

As the mean $\mu_{\hat{E}^*}$ of the estimator $\hat{E}^*$ is

$$\mu_{\hat{E}^*} = \mathrm{E}^*\left\{\hat{E}^*\right\} = \mathrm{E}^*\left\{\frac{e\left(\mathbf{X}_k^*\right)}{f_{\mathbf{x}}^*\left(\mathbf{X}_k^*\right)}\right\} = \int_{R^n} e(\mathbf{x})\, d\mathbf{x} = E \tag{4}$$

then $\hat{E}^*$ is an unbiased estimator of E.

On the other hand, the variance $\sigma_{\hat{E}^*}^2$ of the estimator $\hat{E}^*$ is

$$\begin{aligned}\sigma_{\hat{E}^*}^2 &= \mathrm{E}^*\left\{\left(\hat{E}^* - \mu_{\hat{E}^*}\right)^2\right\} = \mathrm{E}^*\left\{(\hat{E}^*)^2\right\} + \mu_{\hat{E}^*}^2 - 2\mu_{\hat{E}^*}\mathrm{E}^*\left\{\hat{E}^*\right\} \\ &= \frac{1}{N}\left[\mathrm{E}^*\left\{\left(\frac{e\left(\mathbf{X}_k^*\right)}{f_{\mathbf{x}}^*\left(\mathbf{X}_k^*\right)}\right)^2\right\} - E^2\right].\end{aligned} \tag{5}$$

Because of $\sigma_{\hat{E}^*}^2 \to 0$ as $N \to \infty$, we have that $\hat{E}^*$ is a consistent estimator of E, i.e., $\hat{E}^* \to E$ as $N \to \infty$. On the other hand, from Jensen's inequality (see [28, p. 88]) applied to (5), we have

$$\mathrm{E}^*\left\{\left(\frac{e\left(\mathbf{X}_k^*\right)}{f_{\mathbf{x}}^*\left(\mathbf{X}_k^*\right)}\right)^2\right\} \geq E^2. \tag{6}$$

The eqnality case is hold in (6), if and only if

$$\Pr\left\{\frac{e\left(\mathbf{X}_k^*\right)}{f_{\mathbf{x}}^*\left(\mathbf{X}_k^*\right)} = E\right\} = 1 \tag{7}$$

In other words, if the pdf of the sampling process exposed in (2) is given by

$$f_{\mathbf{x}}^*(\mathbf{x}) = \frac{1}{E} e(\mathbf{x}), \mathbf{x} \in R^n \tag{8}$$

then (6) is hold with equality aud, cousequently, from (5) $\sigma_{\hat{E}^*}^2 = 0$ for auy value of N, i.e. the pdf of $\hat{E}^*$ is a Dirac delta function centred on E.

Equation (8) is the unconstrained optimal solution for $f_{\mathbf{x}}^*(\mathbf{x})$. Note that $f_{\mathbf{x}}^*(\mathbf{x})$ must be a probability density function, so it is reqnired that $e(\mathbf{x}) \geq 0, \forall \mathbf{x} \in R^n$, and E mnst be defined in accordance with (1). The optimal solntion for $f_{\mathbf{x}}^*(\mathbf{x})$ given in (8) is not realistic, becanse E is not known a priori (it has to be estimated by (3)). Consequently, we have to find a snboptimum $f_{\mathbf{x}}^*(\mathbf{x})$ that resembles (8), providing a good estimator of E, what is discussed in the following sectious.

The standard deviation $\sigma_{\hat{E}^*}$ of the estimator $\hat{E}^*$ is defined by

$$\sigma_{\hat{E}^*} = \sqrt{\frac{1}{N}\left[\mathrm{E}^*\left\{\left(\frac{e\left(\mathbf{X}_k^*\right)}{f_{\mathbf{x}}^*\left(\mathbf{X}_k^*\right)}\right)^2\right\} - E^2\right]} \tag{9}$$

where (4) and (5) has been considered in (9).

3 Adaptive Search Algorithm for Parameter Estimation of IS Probability Density Function

Frequently, $f_{\mathbf{x}}^{*}(\mathbf{x})$ belongs to a parametric family of pdf's, denoted as $f_{\mathbf{x}}^{*}(\mathbf{x};\boldsymbol{\theta})$, where the parameter vector $\boldsymbol{\theta}=(\theta_1,\theta_1,\ldots,\theta_n)$ belongs to the corresponding parameter space [29]. In general, for some $\boldsymbol{\theta}=(\theta_1,\theta_1,\ldots,\theta_n)$, $f_{\mathbf{x}}^{*}(\mathbf{x};\boldsymbol{\theta})$ accomplishes $f_{\mathbf{x}}^{*}(\mathbf{x};\boldsymbol{\theta})=0$ as $e(\mathbf{x})\neq 0$ for some $\mathbf{x}\in R^n$. Consequently, the estimator $\hat{E}^{*}$ given by (3) is biased; in fact, it is an underestimate of E, as shown below by computing its mean value:

$$\mu_{\hat{E}^{*}}=\mathrm{E}_{\boldsymbol{\theta}}^{*}\left\{\hat{E}^{*}\right\}=\mathrm{E}_{\boldsymbol{\theta}}^{*}\left\{\frac{e\left(\mathbf{X}_k^{*}\right)}{f_{\mathbf{x}}^{*}\left(\mathbf{X}_k^{*};\boldsymbol{\theta}\right)}\right\}<\int_{R^n}e(\mathbf{x})\,d\mathbf{x}=E. \tag{10}$$

For Monte Carlo sampling, the best $f_{\mathbf{x}}^{*}(\mathbf{x};q)$ is the pdf that maximizes (10) in order to avoid the underestimation of E and, simultaneously, minimize (9). We are interested in adaptive importance sampling techniques that, starting from $\boldsymbol{\theta}^{*}=\boldsymbol{\theta}_0$ and using an optimisation algorithm, find a new adequate $\boldsymbol{\theta}^{*}$ in each iteration of the training. It is not necessary for $\boldsymbol{\theta}^{*}$ to be the optimum in each iteration, but it must be the optimum at the end of the training.

Suppose that $\boldsymbol{\theta}=\boldsymbol{\theta}^{*}$ minimizes a combined functional of (9) and (10), as follows:

$$\boldsymbol{\theta}^{*}=\arg\min_{\boldsymbol{\theta}\in\Theta}\left\{\alpha\cdot\log_{10}\left(\mu_{\hat{E}^{*}}\right)+(1-\alpha)\cdot\log_{10}\left(\sigma_{E^{*}}\right)\right\} \tag{11}$$

where the parameter α is a positive real number that balances the importance of each term. Therefore, $\boldsymbol{\theta}=\boldsymbol{\theta}^{*}$ is a good value to be used in $f_{\mathbf{x}}^{*}(\mathbf{x};\boldsymbol{\theta})$ for sampling.

As the theoretical analysis of (11) is difficult, we can also use an estimator and compute it by Monte Carlo simulations. For this purpose, a good estimator of (11) is

$$\hat{\boldsymbol{\theta}}^{*}=\arg\min_{\boldsymbol{\theta}\in\Theta}\left\{\alpha\log_{10}\left(\hat{E}^{*}\right)+(1-\alpha)\cdot\log_{10}\left(\hat{\sigma}_{E^{*}}\right)\right\} \tag{12}$$

where

$$\hat{\sigma}_{\hat{E}^{*}}=\sqrt{\frac{1}{N}\left[\frac{1}{N}\sum_{k=1}^{N}\left(\frac{e\left(\mathbf{x}_k^{*}\right)}{f_{\mathbf{x}}^{*}\left(\mathbf{x}_k^{*};\boldsymbol{\theta}\right)}\right)^2-\left(\hat{E}^{*}\right)^2\right]} \tag{13}$$

and $\hat{E}^{*}$ is redefined from (3) for parameterization purposes as

$$\hat{E}^{*}=\frac{1}{N}\sum_{k=1}^{N}\frac{e\left(\mathbf{x}_k^{*}\right)}{f_{\mathbf{x}}^{*}\left(\mathbf{x}_k^{*};\boldsymbol{\theta}\right)}. \tag{14}$$

Finally, the Monte Carlo-based algorithm proposed for selecting the best $\boldsymbol{\theta}^{*}$, is called *Adaptive Search Algorithm* (ASA). It is applied each iteration of the NN training in order to find the corresponding optimum value $\boldsymbol{\theta}^{*}$, which is necessary to estimate E in each iteration. The steps of the algorithm are:

1. Consider a block of random values for the parameter of the suboptimal IS pdf, $\boldsymbol{\theta}_i\in\Theta, i=1,2,\ldots,L$, where L is large enough (e.g. $L=10^2$). The parameters are uniformly distributed. In the experimental results presented in Sect. 5, in the first iteration of the algorithm, the uniform distribution is centred in the estimated mean of the true hypothesis, and the width of the distribution is 0.4. In the remaining iterations, the distribution is centred in the best value obtained in the previous iteration.

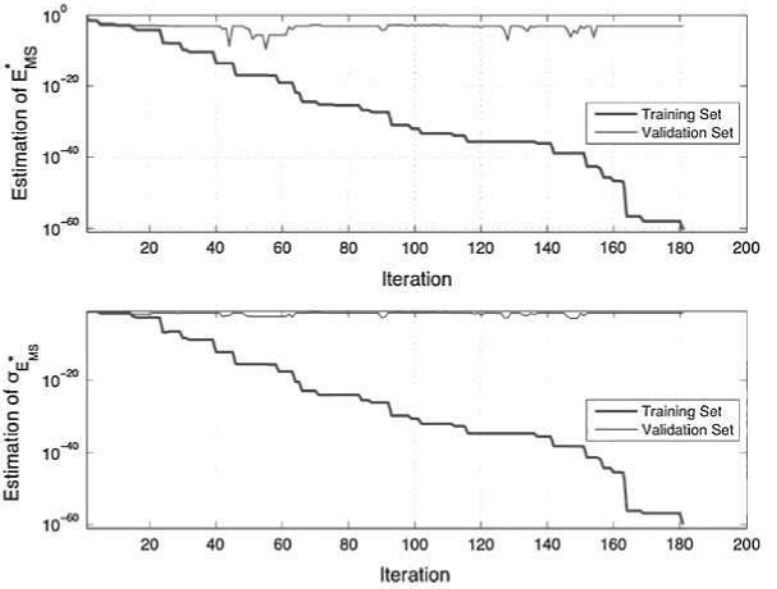

Fig. 2 Evolution of the estimation error ($\hat{E}^*_{MS}$) and its standard deviation ($\hat{\sigma}_{\hat{E}^*_{MS}}$) during the training of an MLP $5 \times 5 \times 1$ with the MS error criterion (without IS technique)

2. Compute (14) and (13) for each $\theta_i \in \Theta$, $i = 1, 2, \ldots, L$, and each NN in the population, with an appropriate number of patterns or observations N (e.g. $N = 10^3$), generated with each suboptimal IS pdf.
3. Select the value $\theta = \theta^*$ that minimizes (12) among the L possible values.
4. Compute again (14) and (13) for $\theta = \theta^*$, with an appropriate number of patterns N (e.g. $N = 10^4$, ten times the number of patterns used in the step 2), for each NN in the population. Note that these estimations are obtained by using IS techniques with the suboptimal pdf $f^*_{\mathbf{x}}(\mathbf{x}; \theta^*)$, and are used to apply the GA for NN training, as described in Sect. 5.

In the next sections, this algorithm is iteratively applied to train NNs by using GAs with different objective functions. The objective functions considered in our study are explained in Sect. 4. Also, computer simulations have been carried out in order to compare their training performances, basing this comparison in their estimated P_e and ROC curves.

In order to illustrate ASA, some figures have been added with the results of the experiments in Sect. 5. Figures 2, 3 and 4 illustrate the big differences in the variance of the estimator $\hat{E}^*$ using the training and validation sets, demonstrating that NNs are not properly trained, even using IS techniques. In order to avoid it, ASA is applied. Figures 5, 6 and 7 illustrate the evolution of the estimation error ($\hat{E}^*_{MS}$), its estimated standard deviation ($\hat{\sigma}_{\hat{E}^*_{MS}}$) and the optimum parameter (θ^*), demonstrating that using ASA, NNs are trained properly using IS techniques.

4 Objective Functions for Training by Importance Sampling

In this section, the use of IS techniques, as previously exposed, in the NN training and testing considering objective functions commonly used in communication systems [13] is proposed.

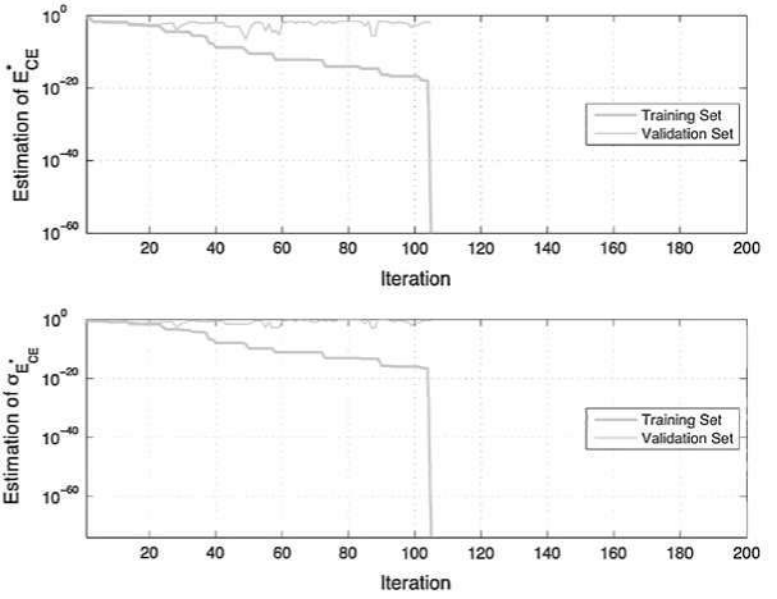

Fig. 3 Evolution of the estimation error ($\hat{E}^*_{CE}$) and its standard deviation ($\hat{\sigma}_{\hat{E}^*_{CE}}$) during the training of an MLP $5 \times 5 \times 1$ with the CE error criterion (without IS technique)

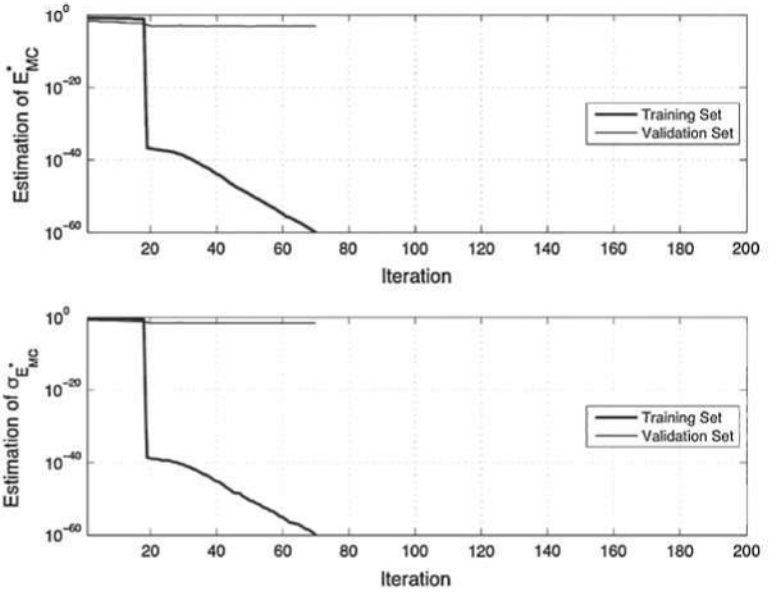

Fig. 4 Evolution of the estimation error ($\hat{E}^*_{MC}$) and its standard deviation ($\hat{\sigma}_{\hat{E}^*_{MC}}$) during the training of an MLP $5 \times 5 \times 1$ with the MC error criterion (without IS technique)

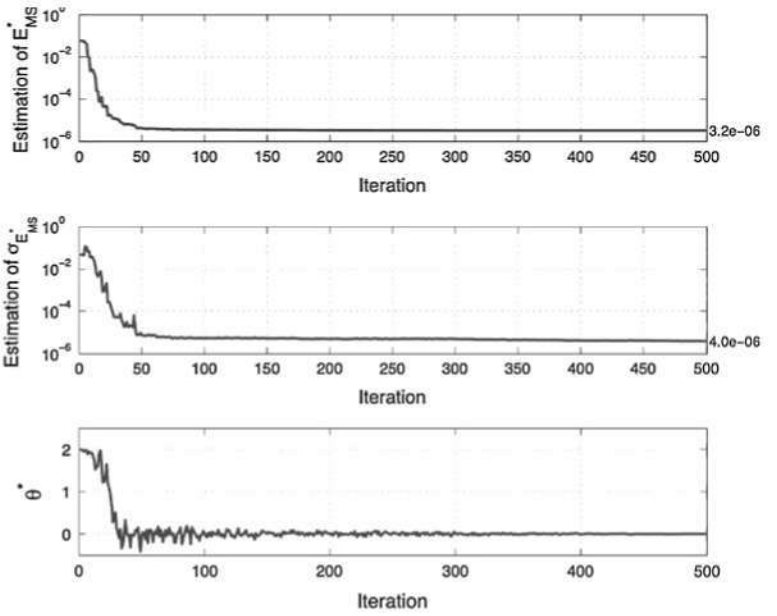

Fig. 5 Evolution of the estimation error ($\hat{E}^*_{MS}$), its estimated standard deviation ($\hat{\sigma}_{\hat{E}^*_{MS}}$) and the optimum parameter ($\boldsymbol{\theta}^*$) during the training of an MLP $5 \times 5 \times 1$ with IS and the MS error criterion

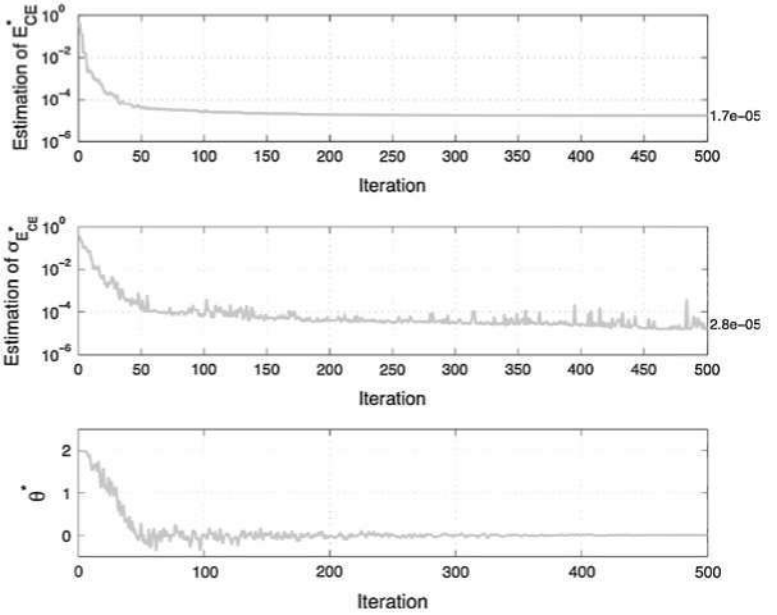

Fig. 6 Evolution of the estimation error ($\hat{E}^*_{CE}$), its estimated standard deviation ($\hat{\sigma}_{\hat{E}^*_{CE}}$) and the optimum parameter ($\boldsymbol{\theta}^*$) during the training of a MLP $5 \times 5 \times 1$ with IS and the CE error criterion

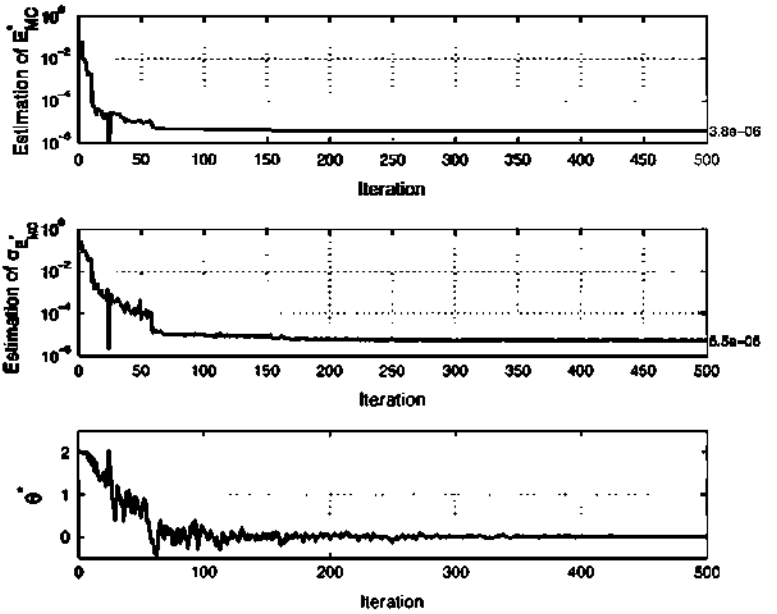

Fig. 7 Evolution of the estimation error ($\hat{E}^*_{MC}$), its estimated standard deviation ($\hat{\sigma}_{\hat{E}^*_{MC}}$) and the optimum parameter (θ^*) during the training of an MLP $5 \times 5 \times 1$ with IS and the MC error criterion

4.1 Mean-Square Error Objective Function

First, we consider the MS error as objective function [25,29]. According to the notation given above, the MS error is defined by

$$E_{MS} = \mathrm{E}\left\{(Y - Y_d)^2\right\} = \int_{R^n} e\left(\mathbf{x}\right) d\mathbf{x} \tag{15}$$

where the random variable $Y = g\left(\mathbf{X}\right)$ is the NN output, Y_d is the desired output ($Y_d = 1$ for H_1 and $Y_d = 0$ for H_0), and $e\left(\mathbf{x}\right)$ is given by

$$e\left(\mathbf{x}\right) = P\left(H_0\right)\left[g\left(\mathbf{x}\right)\right]^2 f_{\mathbf{X}}\left(\mathbf{x}|H_0\right) + P\left(H_1\right)\left[g\left(\mathbf{x}\right) - 1\right]^2 f_{\mathbf{X}}\left(\mathbf{x}|H_1\right), \quad \mathbf{x} \in R^n \tag{16}$$

being $e\left(\mathbf{x}\right) \geq 0, \forall \mathbf{x} \in R^n$. Then, (15) with (16) is a particular case of (1). In this way, all the conclusions of Sect. 2 are applied here. In particular, taking (16) into (8), the unconstrained optimal solution of $f^*_{\mathbf{X}}\left(\mathbf{x}\right)$ can be expressed for future refereuces as follows

$$f^*_{\mathbf{X}}\left(\mathbf{x}\right) = \frac{1}{E_{MS}}\left[P\left(H_0\right)\left[g\left(\mathbf{x}\right)\right]^2 f_{\mathbf{X}}\left(\mathbf{x}|H_0\right) + P\left(H_1\right)\left[g\left(\mathbf{x}\right) - 1\right]^2 f_{\mathbf{X}}\left(\mathbf{x}|H_1\right)\right], \quad \mathbf{x} \in R^n. \tag{17}$$

The optimal solution for $f^*_{\mathbf{X}}\left(\mathbf{x}\right)$ given in (17) is not realistic [25] because E_{MS} is not known a priori (it has to be estimated by (3)). Furthermore, in the training stage, $g\left(\cdot\right)$ is modified from one iteration to another in order to minimize E_{MS}.

4.2 Cross Entropy Error Objective Function

Now, we consider the CE error as objective functiou, which is defined by

$$E_{CE} = \mathrm{E}\{-\ln|Y - Y_d|\} = \int_{R^n} e(\mathbf{x})\, d\mathbf{x} \tag{18}$$

where $e(\mathbf{x})$ is given by [29]

$$e(\mathbf{x}) = -P(H_0)\ln|g(\mathbf{x}) - 1|\, f_{\mathbf{X}}(\mathbf{x}|H_0) - P(H_1)\ln|g(\mathbf{x})|\, f_{\mathbf{X}}(\mathbf{x}|H_1), \quad \mathbf{x} \in R^n \tag{19}$$

being $e(\mathbf{x}) \geq 0, \forall \mathbf{x} \in R^n$. As previously occurred, (18) with (19) is a particular case of (1). So, all the conclusions of Sect. 2 are applied in this case. Taking (19) into (8), the unconstrained optimal solution of $f_{\mathbf{X}}^*(\mathbf{x})$ can be expressed as follows

$$f_{\mathbf{X}}^*(\mathbf{x}) = \frac{1}{E_{CE}}\left[-P(H_0)\ln|g(\mathbf{x}) - 1|\, f_{\mathbf{X}}(\mathbf{x}|H_0) - P(H_1)\ln|g(\mathbf{x})|\, f_{\mathbf{X}}(\mathbf{x}|H_1)\right], \quad \mathbf{x} \in R^n. \tag{20}$$

Once again, the optimal solution for $f_{\mathbf{X}}^*(\mathbf{x})$ given in (20) is not realistic, as in case (a).

4.3 Misclassification Error Objective Function

Also, we cousider the Error Probability or Misclassificatiou error (E_{MC}) as objective fuuction, which is defined by [30]

$$E_{P_e} = E_{MC} = P(H_0)\, P(D_1|H_0) + P(H_1)\, P(D_0|H_1) = \int_{R^n} e(\mathbf{x})\, d\mathbf{x} \tag{21}$$

where $e(\mathbf{x})$ is given by

$$e(\mathbf{x}) = P(H_0)\, f_{\mathbf{X}}(\mathbf{x}|H_0)\, u(g(\mathbf{x}) - T_0) + P(H_1)\, f_{\mathbf{X}}(\mathbf{x}|H_1)\, u(T_0 - g(\mathbf{x})), \quad \mathbf{x} \in R^n \tag{22}$$

being $e(\mathbf{x}) \geq 0, \forall \mathbf{x} \in R^n$. Then, (21) with (22) is a particular case of (1). Taking (22) into (8), the unconstrained optimal solution of $f_{\mathbf{X}}^*(\mathbf{x})$ can be expressed as follows

$$f_{\mathbf{X}}^*(\mathbf{x}) = \frac{1}{E_{MC}}\left[P(H_0)\, f_{\mathbf{X}}(\mathbf{x}|H_0)\, u(g(\mathbf{x}) - T_0) + P(H_1)\, f_{\mathbf{X}}(\mathbf{x}|H_1)\, u(T_0 - g(\mathbf{x}))\right], \quad \mathbf{x} \in R^n. \tag{23}$$

Finally, the optimal solution for $f_{\mathbf{X}}^*(\mathbf{x})$ given in (23) is not realistic, as in cases (a) and (b).

As can be observed from the previous analysis of the used objective functions, suboptimal solutions of $f_{\mathbf{X}}^*(\mathbf{x})$ are proposed in this paper for (17), (20) and (23). $f_{\mathbf{X}}(\mathbf{x}|H_i)$, $i = 1, 0$, usually depends ou a parameter $\boldsymbol{\theta}$. Iu our case of study, this parameter is related to the Sigual-to-Noise Ratio (SNR) and it cau be written as $f_{\mathbf{X}}(\mathbf{x}; \boldsymbol{\theta}|H_i)$, $i = 1, 0$, i.e., if $\boldsymbol{\theta} = 0$ (zero vector), there is only noise (both pdf's, $f_{\mathbf{X}}(\mathbf{x}|H_0)$ and $f_{\mathbf{X}}(\mathbf{x}|H_1)$, are identical to the noise pdf), and if $\boldsymbol{\theta}$ is too large, both hypotheses are highly separate, where very low values of objective functions are achieved. Consequently, we propose the following family of density functions $f_{\mathbf{X}}^*(\mathbf{x}; \boldsymbol{\theta})$, $\boldsymbol{\theta} \in \Theta$, as the IS pdf's

$$f_{\mathbf{X}}^*(\mathbf{x}; \boldsymbol{\theta}) = P(H_0)\, f_{\mathbf{X}}(\mathbf{x}; \boldsymbol{\theta}|H_0) + P(H_1)\, f_{\mathbf{X}}(\mathbf{x}; \boldsymbol{\theta}|H_1), \quad \mathbf{x} \in R^n. \tag{24}$$

The value $\boldsymbol{\theta}^*$ is obtained by the ASA algorithm proposed in Sect. 3.

Because it is not guaranteed in (24) that $f_{\mathbf{x}}^{*}(\mathbf{x};\theta) \neq 0$ wherever $e(\mathbf{x}) \neq 0, \forall \mathbf{x} \in R^n$, we shall have $\mathrm{E}^{*}\left\{\hat{E}\right\} \leq E$, i.e., $\hat{E}$ is an uuderestimation of E. Note that $\hat{E}$ is also an underestimator of E if $f_{\mathbf{x}}^{*}(\mathbf{x};\theta) \approx 0$ and $f_{\mathbf{x}}^{*}(\mathbf{x};\theta) << e(\mathbf{x})$ for some $\mathbf{x} \in R^n$, i.e., $\hat{E} < E$ with high probability. Therefore, we have to assure that the solution θ^{*} produces minimum bias in $\hat{E}$.

Finally, we point out that ASA is computed at each iteration of the NN training in order to find the corresponding optimum value θ^{*}, which is necessary to estimate E in each iteration. At the end of the training, $\hat{E}^{*}$ and $\hat{\sigma}_{\hat{E}^{*}}$ are good estimations of E and $\sigma_{\hat{E}^{*}}$, respectively, as shown in the following sections.

5 Results

Let's consider the detection of binary symbols in Gaussian noise, by means of NNs trained without and with IS using the three objective functions defined above. The hypotheses are $H_1 : \mathbf{x} = \eta + \mathbf{a}$ and $H_0 : \mathbf{x} = \eta - \mathbf{a}$, where $\mathbf{a} = (a_1, a_2, \ldots, a_n)$, $a_i = \mu, i = 1, 2, \ldots, n, \mu$ is a real constant (for simulations $\mu = 2.0$) and $h = (\eta_1, \eta_2, \ldots, \eta_n)$ is a Gaussian noise vector of independent and identically distributed zero-mean samples of unit variance. Then, their pdf's are normal distributed with means $+\mu$ and $-\mu$, respectively, i.e.

$$f_{\mathbf{x}}(\mathbf{x}|H_0) = \frac{1}{\sqrt{(2\pi)^n}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(x_i + \mu)^2\right) \tag{25}$$

$$f_{\mathbf{x}}(\mathbf{x}|H_1) = \frac{1}{\sqrt{(2\pi)^n}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(x_i - \mu)^2\right). \tag{26}$$

Also, let's suppose that, for simulations purposes, $P(H_1) = P(H_0) = \frac{1}{2}$, i.e., the symbols are equally likely.

The binary detector used in our experiments is based on a feedforward NN of type Multi-Layer Perceptron (MLP), which is used as the nonlinear system $g(\mathbf{x})$ of Fig. 1. The parameters that define an MLP are its size: $5 \times 5 \times 1$ (i.e. number of input nodes: 5; number of neurons in the hidden layer: 5; and number of outputs: 1) and the activation function for each neuron of the MLP: a sigmoid function [31] for our case of study. A GA [25] is used for training the MLP. GAs for optimisation are very close to Monte Carlo techniques, therefore, IS is well tailored for training NNs by means of GA. In fact, IS and GA are stochastic algorithms and they can interact very easily.

Although our GA with elitism and real-number genes in the chromosomes is not the subject of this paper, because it is considered as a tool, the parameters used in the training are supplied. A standard set of parameters has been defined based on the results of several experiments, and the ideas found in the technical literature [32, 33]:

- Number of individuals that composed the population: 20 MLPs randomly initialized using the Nguyen–Widrow method [34] at the beginniug of the GA computation. The uumber of iudividuals in the populatiou has beeu usually set to 50 iu other works [32], but in this application it seems to be very big to ensure convergeuce, furthermore cousidering that ASA is simultaueously applied. In this paper we have used the same number of individuals as in [25], i.e. 20 MLPs.
- Scaling of the MLP estimated errors: scaling by rank, which sets the rank of an individual with its position in the sorted scores, rather than its score.

- Type and quantity of selection in the population: the best (elitism) 10 MLPs survive to the next population.
- Crossover: it is not applied between individuals.
- Type and quantity of mutation: random values from a Gaussian distribution with zero mean and a variance of 0.20 are added to randomly selected genes of the worst 10 MLPs, i.e. soft modifications to the genes of the remaining population after the selection are applied.
- Fitness function: $\log_{10}\left(\hat{E}^{*}_{MS}\right)$, $\log_{10}\left(\hat{E}^{*}_{CE}\right)$ and $\log_{10}\left(\hat{E}^{*}_{MC}\right)$, depending on the objective function used in the NN training.
- Stopping criteria is based on the achievement of one of the following limitations:
  - a maximum of 500 iterations;
  - a maximum of 10 generations where the error does not decrease (stall);
  - and a goal lower than or equal to 10^{-60} in the estimated error.

In the case of NN training without IS, an external validation in the training is done. It is done by a validation set, which contains different observation vectors with the same pdf as the training set. This external validation is not done when the NN training is done with IS because the algorithm proposed changes the virtual training set each step of the algorithm, so it is impossible that the MLP memorizes the training set and no validation set is needed to stop the training process before the MLP is specialized in the training set. So, first, all the NNs are trained using the three objective functions under study without using IS in order to highlight the main problems this way of training has, and after the NNs are trained with IS in order to emphasize its advantages. The evolution of the training processes without IS are shown in Figs. 2, 3, and 4 for the MS, CE and MC errors, respectively. As can be observed, at the beginning of the trainings, the errors in the estimation of the objective functions are high. This fact makes unreliable the estimation of the objective function, but with the progress of the training they evolve to better results in terms of the estimated errors and their standard deviations. As can be observed, all the training processes using the different objective functions (estimated errors with the training set) are stopped because the estimated errors are lower than 10^{-60}. After their trainings, the three NNs seem to be well trained when the error is estimated with the training set. But the results achieved when the validation set is considered show that the MS, CE and MC error estimates are approximately 10^{-3}, 10^{-2} and 10^{-3}, respectively, which are greater than 10^{-60}. On the other hand, the standard deviations of these errors estimated with the validation set are approximately 3×10^{-2}, 3×10^{-1} and 3×10^{-2}, respectively. The standard deviations achieved show us that the error estimation accuracy is not as good as we could hope, so the IS technique is necessary during the training to estimate the errors and their standard deviations. In order to compare the use or not of IS in the training, the P_e obtained for the NN-based detectors designed with the three objective criteria considered in our studies and a decision threshold of $T_0 = 0.5$ are given in Table 1. Moreover, the ROC curves of these detectors are also given in Fig. 8.

In order to improve the NN training, IS is used to estimate the objective functions as described in previous sections. The number of observation vectors for estimating the three types of errors by the IS technique is $N = 10^4$ once the θ^* parameter is selected for the suboptimal pdf, as shown in the ASA specifications. This amount of observations is used to have enough precision in the estimations, where the error in the estimations is very low using IS techniques for the case under study.

Figure 5 shows comparatively the three main curves of the training progress with ASA when the MS error criterion is used. The balance parameter of (12) is set to $\alpha = 0.20$

Table 1 Probability of error of the binary detector based on MLPs $5 \times 5 \times 1$ trained without and with IS to minimize the MS, CE and MC error criteria and tested with IS

Binary detector based on	Probability of error (P_e)	
	Trained without IS	Trained with IS
$\text{MLP}^{(\text{MS})}$	6.07×10^{-5}	3.03×10^{-5}
$\text{MLP}^{(\text{CE})}$	6.93×10^{-5}	3.98×10^{-5}
$\text{MLP}^{(\text{MC})}$	5.22×10^{-5}	2.12×10^{-5}

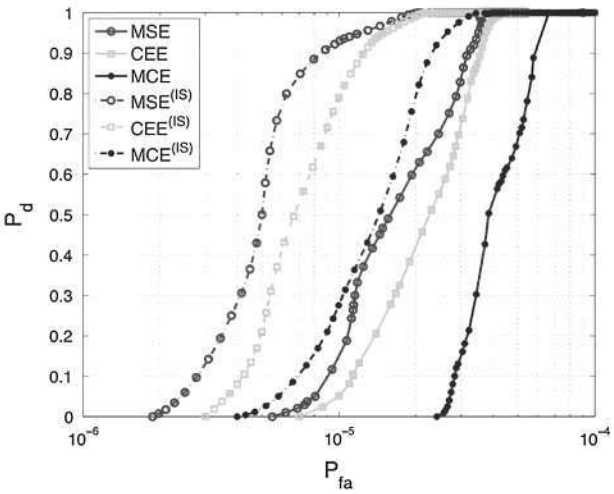

Fig. 8 ROC curves for the binary detectors based on MLPs $5 \times 5 \times 1$ trained to minimize the MS, CE and MC error criteria without IS (MSE, CEE and MCE in *grey solid lines*) and with IS ($\text{MSE}^{(\text{IS})}$, $\text{CEE}^{(\text{IS})}$ and $\text{MCE}^{(\text{IS})}$ in *grey dashed lines*)

(empirically obtained) and the mean values of the Gaussian symbols are $+2.0$ and -2.0 ($a_i = \mu = 2.0, i = 1, 2, \ldots, n$), i.e., a single parameter for the family of admissible IS pdf's is considered because the means of each sample a_i are the same. Considering $a_i = \mu = 2.0, i = 1, 2, \ldots, n$ in (25) and (26) we have the family

$$f_{\mathbf{X}}^{*}(\mathbf{x}; \theta) = \frac{1}{2} \frac{1}{\sqrt{(2\pi)^n}} \left[\exp\left(-\frac{1}{2} \sum_{i=1}^{n} \left(x_i + \theta^* \right)^2 \right) + \exp\left(-\frac{1}{2} \sum_{i=1}^{n} \left(x_i - \theta^* \right)^2 \right) \right]. \quad (27)$$

Focusing on the Fig. 5, the upper, middle and bottom plots correspond to the evolution of the estimation of the MS error ($\hat{E}_{MS}^{*}$) by (14), the estimation of the standard deviation ($\hat{\sigma}_{\hat{E}^*}$) of $\hat{E}_{MS}^{*}$ by (13) and the evolution of the optimum parameter θ^* for $f_{\mathbf{X}}^{*}(\mathbf{x}; \theta)$, respectively, when the training of the NN is based on IS. Several aspects can be emphasized. First, the estimated error and its estimated standard deviation during the training and especially at its end are always greater than the estimations obtained without IS training (for comparison, see Fig. 2). Second, the best parameter θ^* adapts automatically from the starting value ($\theta^* = 2.0$) to a null value ($\theta^* = 0$). It involves that the best $f_{\mathbf{X}}^{*}(\mathbf{x}; \theta)$ at the end of the training is a normal pdf with zero mean and unity variance, i.e., the noise pdf is the suboptimal solution for the

IS pdf when the NN is quasi-trained, which is

$$f_{\mathbf{x}}^{*}(\mathbf{x};\theta) = \frac{1}{\sqrt{(2\pi)^n}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n} x_i^2\right). \tag{28}$$

Third, at the beginning of the training, $\hat{E}_{MS}^{*}$ is high (poor behaviour) with high standard deviation ($\hat{\sigma}_{E_{MS}^{*}}$) in its estimation, but lower than the equivalent training without IS, so the estimations of E_{MS} and $\sigma_{E_{MS}}$ are inaccurate. When the training progresses, $\hat{E}_{MS}^{*}$ decreases to a minimum value of 3.2×10^{-6} and makes more reliable this estimation because the $\hat{\sigma}_{\hat{E}_{MS}^{*}}$ decreases to a valne of 4×10^{-6}. As can be observed, the estimations obtained here are better than those obtained in the case of training withont IS.

Fignres 6 and 7 show the progresses of the MLP training with the CE and MC error criteria, respectively. As occurred when the MS error criterion is used, the same aspects related with the estimatiou of the error, its standard deviation and the optimum value of the pdf can be obtained for both. Because the error criteria are different, the estimation errors $\hat{E}_{CE}^{*}$ and $\hat{E}_{MC}^{*}$ tend antomatically to 1.7×10^{-5} and 3.8×10^{-6} at the end of the training, respectively. Whereas, their standard deviations $\hat{\sigma}_{E_{CE}^{*}}$ and $\hat{\sigma}_{E_{MC}^{*}}$ are 2.8×10^{-5} and 5.5×10^{-6}, respectively. As occnrs when the MS error criterion is used, both estimations are better than those obtained in their corresponding cases of training without IS (see Figs. 3, 4).

Because different error criteria in the NN training are used, the minimum error achieved at the end of the trainiug cannot be used to compare each other. So, two different measurements are used to compare their performances iu the testiug stage. The first one corresponds to the estimated P_e and the second one to the estimation of the ROC cnrves. Each point of the ROC curves (P_{fa}, P_d) is obtained for a given threshold T_0 by IS nsing 10^5 observations of each hypothesis. This process of estimation lets to obtain a maximnm error in each estimation of 5%, even in the worst cases (low P_{fa} values).

Table 1 shows the P_e of the binary detector that uses an MLP trained without IS to minimize the MS ($\text{MLP}^{(\text{MS})}$), CE ($\text{MLP}^{(\text{CE})}$) and MC ($\text{MLP}^{(\text{MC})}$) errors and a threshold value of $T_0 = 0.5$. The results show how the binary detector using the $\text{MLP}^{(\text{MC})}$ is better than the others for the considered threshold. Moreover, the worst one is the binary detector using an $\text{MLP}^{(\text{CE})}$. These resnlts show that $\text{MLP}^{(\text{MC})}$ is the best option to implement a binary detector in commnnications applications where the probability of error is the parameter that defines the quality of the system. Moreover, the same results are shown for MLPs trained with IS. From these results, the same conclnsions as previonsly given can be achieved, bnt with more accurate measurements because the training is based on IS.

On the other hand, the ROC curves of the previous binary detectors are exposed in Fig. 8. The results show how the performaunces of the detectors based ou MLPs trained with MS and CE error criteria are better thau the one trained usiug the MC error criteriou. This is due to MC error criterion minimizes the error of classification, instead of the overall error as do the previons ones. Moreover, as can also be observed, this behaviour is independent whether the training is done nsing or not IS. The differences between both kind of trainings (without and with IS) is due to the more accurate estimation of the objective fnnctiou in trainiug with IS, where those eveuts that rarely happens are artificially cousidered. As cau be observed, the best error criterion to traiu MLPs with ASA in terms of ROC curves is the MS oue, followed by the CE one, whereas the worst oue is the MC.

The experiments presented in this section, which are taken as a reference here on, show the performance improvement of training MLPs with IS against the training without IS. But it is necessary to check if this behavior is robust against changes in the parameters of the ASA

Table 2 Probability of error of the binary detectors based on MLPs $5 \times 5 \times 1$ trained without and with IS to minimize the MS, CE and MC error criteria and tested with IS when 10^4 patterns are used during the training of the GA

Binary detector based on	Probability of error (P_e)	
	Trained without IS	Trained with IS
$MLP^{(MS)}$	5.31×10^{-5}	2.98×10^{-5}
$MLP^{(CE)}$	6.24×10^{-5}	3.92×10^{-5}
$MLP^{(MC)}$	4.21×10^{-5}	2.07×10^{-5}

training algorithm or in the MLP size. The next two subsections are dedicated to study the influence of the variation of parameters like the number of observations used in the training and the influence of the MLP size in the binary detector. Other parameter studies like the initial value of θ being greater than 2.0 or the limitation of the GA iterations to 200 has shown no relevant variations in the performances (P_e and ROC curves) of the designed detectors.

5.1 Influence of the Number of Observations Considered in the NN Training

One aspect to take into account is the number of observations used in the NN training, when both IS is used or not. It is important because the higher the number of observations considered in the training, the higher the accuracy of the performances achieved and the higher the computational cost. But, its accuracy and computational cost are not the only magnitude that increases with the number of observations, the memory requirements increase too. So, in this subsection, the affects of an increase in the number of observations in the training are exposed and analyzed.

Table 2 shows the P_e of the binary detectors based on MLPs trained to minimize the MS, CE and MC error criteria without and with IS when $N = 10^4$ observations for step 2 in ASA and $N = 10^5$ observations for step 4 in ASA are used. Note that 10^4 observations are also used in the training and validations sets when no IS is used in the NN training. So, if both cases are compared with the reference case exposed in Sect. 5 ($N = 10^3$ and $N = 10^4$ patterns for steps 2 and 4 in ASA, respectively), which results are exposed in Table 1, several conclusions can be obtained. First, changes are only relevant when no IS is used in the training, because the detector performances remain practically constant when IS is used. Second, the P_e estimations decrease for all the cases when no IS is used. The reason why it happens is due to an increase of the number of observations involves an improvement of the accuracy of the error estimation in this kind of training, what finally improves the detector performances. And third, the relative behaviours between error criteria continue as in the reference case, i.e., the best error criterion is the MC one and the worst is the CE one, independently whether IS is used or not.

On the other hand, the ROC curves of the binary detectors exposed in the previous paragraph are shown in Fig. 9. If they are compared with the performances of the binary detectors obtained in the reference case (see Fig. 8), the same conclusions can be achieved. In this case, first, the best performances are always achieved when IS is used in the training. Second, the best performances for low P_{fa} values are obtained when the MS error criterion is used instead of the CE and MC ones. This behaviour is independent whether the training is based on IS or not. And third, the performances of these detectors are always better than in the reference case when no IS is used and are practically the same when IS is used. The reasons of this behaviour are exactly the same as the ones showed previously for the estimation of the P_e. So, the accuracy in the estimation during the training is better when no IS is used and

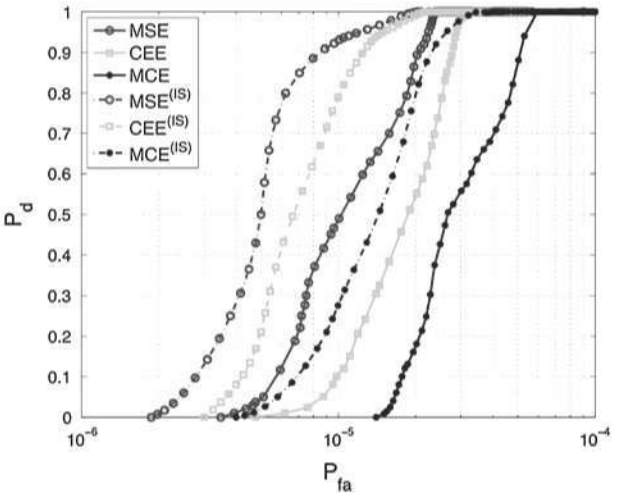

Fig. 9 ROC curves for the binary detectors based on MLPs $5 \times 5 \times 1$ trained to minimize the MS, CE and MC error criteria without IS (*grey solid lines*) and with IS (*grey dashed lines*) when 10^4 observations are used during the training of the GA

Table 3 Probability of error of the binary detector based on MLPs $5 \times 10 \times 1$ trained without and with IS to minimize the MS, CE and MC error criteria and tested with IS

Binary detector based on	Probability of error (P_e)	
	Trained without IS	Trained with IS
$\text{MLP}^{(\text{MS})}$	4.01×10^{-5}	2.10×10^{-5}
$\text{MLP}^{(\text{CE})}$	4.78×10^{-5}	2.93×10^{-5}
$\text{MLP}^{(\text{MC})}$	3.06×10^{-5}	1.15×10^{-5}

practically the same when IS is used, what involves that this number of observations increase is not justified when IS is used.

5.2 Influence of the MLP Size

Other parameter that should be studied is the MLP size, which is related with the intelligence of the binary detector. So, an increase in the MLP size, normally involves an improvement of the detector performance. In this subsection, this effect is studied when an increase from 5 hidden neurons to 10 hidden neurons is done. Table 3 shows the results of the binary detectors based on MLPs of size $5 \times 10 \times 1$ trained with MS, CE and MC error criteria when IS and no IS are used. The results show that the best error criterion is the MC one, whereas the worst one is the MS one. This behavior is independent of training without or with IS. Moreover, these conclusions are the same as those obtained for the case of MLP sizes of $5 \times 5 \times 1$ (reference case). The only difference between this case and the reference one is related with the achieved estimated errors. In this way, for high MLP sizes, the error is lower than for the case of MLP sizes of $5 \times 5 \times 1$, as can be observed in the obtained results.

On the other hand, the ROC curves for the previous binary detectors are shown in Fig. 10. As can be observed, the best performances are achieved when the MS error criterion is used and the worst ones are achieved when the MC one is used. Again, this behavior is independent of training without or with IS. A comparison between these results and those obtained for the

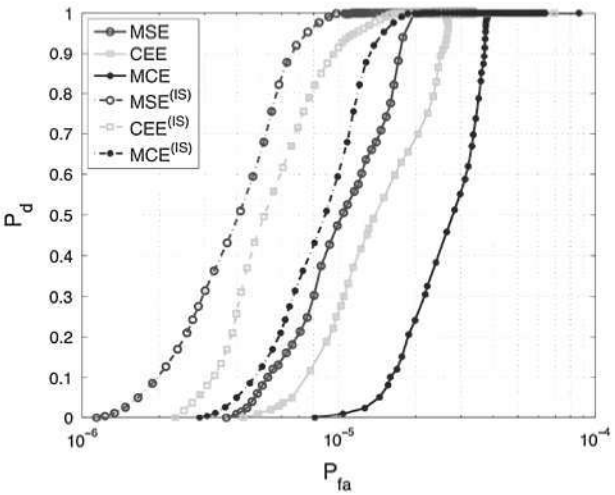

Fig. 10 ROC curves for the binary detectors based on MLPs $5 \times 10 \times 1$ trained to minimize the MS, CE and MC error criteria without IS (MSE, CEE and MCE in *grey solid lines*) and with IS ($\text{MSE}^{(\text{IS})}$, $\text{CEE}^{(\text{IS})}$ and $\text{MCE}^{(\text{IS})}$ in *grey dashed lines*)

lowest MLP size ($5 \times 5 \times 1$) taken as reference (Fig. 8) gives the same kind of conclusions. But the main difference is based on the performances achieved, because they are better than those obtained in the reference case, specially when no IS is used.

6 Conclusions

Estimations of three objective functions (Mean-Square, Cross Entropy and Misclassification error criteria) have been performed to supervise NN training. The value of these functions decreases as the training progresses and so, the number of test observations necessary for an accurate estimation has to be increased. Consequently, the training computational cost is unaffordable for very low objective function value estimations, and the use of IS techniques is applied to drastically accelerate its computation and to improve the final performance. The optimal IS pdf for each objective function is presented in this paper, becoming evident the impossibility of implementing them, because they depend on the error to be estimated. Because of that, suboptimal IS probability density functions are proposed in order to estimate error during training. The proposed suboptimal IS probability density functions belong to a family of parametric functions. The optimum parameter is determined during training by applying an adaptive search algorithm (ASA). The ASA is based on Genetic Algorithms and is performed in order to find a good IS probability density function (biasing pdf) in each iteration of the training to compute the estimation of the objective function and the error in this estimation. For our specific application, it is found that, at the end of the training, the best pdf for parameter estimation in ASA is the normal distribution with zero mean and unity variance because the parameter θ^* vanishes at the end of the training.

In order to determine which error criterion is better to implement detectors based on the Neyman–Pearson criterion, or classifiers to minimize the classification error, some experiments have been carried out. The comparison of the results for the binary detectors based on MLPs trained with the three objective functions allows us to extract four important

conclusions. The first one is related to the P_e of the binary detectors (equivalent to the Misclassification error), where the lowest P_e (better performance) is achieved by the use of the Misclassification error criterion in the training process, whereas the worst objective function is the Cross Entropy one.

The second conclusion is obtained from the ROC curves. This conclusion establishes that the best objective function is the Mean-Square error, what makes it the best option for radar applications, where the maximization of P_d for P_{fa} remaining constrained to low values is required (Neyman–Pearson criterion). In this kind of applications, the worst performances are achieved when the Misclassification error criterion is used, whereas the Cross Entropy error criterion is between them.

The third conclusion is obtained according to the study of the results obtained during the training without and with IS. In this sense, the first two conclusions are independently obtained whether IS is used or not during the trainings. But, note that the performances of the NN-based detectors are always better when IS is used in the training than when it is not used.

Finally, the fourth conclusion let us to know that some parameters, like the number of observations, make more reliable the training without IS but do not improve the performance of the NNs trained using IS, whereas this improvement increases the computational cost of the training. On the other hand, other parameters, like the NN size, let us improve the NN-based detector performances, whereas others, like an increase in the number of iterations by more than 200, do not improve them. Moreover, other parameters of the ASA algorithm, such as the initial value of the parameter q, do not produce relevance changes in the detector performances.

Concluding, we have studied in detail the use of IS techniques to train NN in order to implement NN-based detectors. Three objective functions have been considered in the paper, demonstrating that the Mean Square Error is the better one. An adaptive search algorithm (ASA) of the best parameter of the IS probability density function has been proposed, that is applied simultaneously with a GA for NN training.

References

1. Atanassov E, Dimov IT (2008) What Monte Carlo models can do and cannot do efficiently? Appl Math Model 32:1477–1500
2. Borcherds PH (2000) Importance sampling: an illustrative introduction. Eur J Phys 21:405–411
3. Denny M (2001) Introduction to importance sampling in rare-event simulations. Eur J Phys 22:403–411
4. Andrieu C, Freitas N, Doucet A, Jordan MI (2003) An introduction to MCMC for machine learning. Mach Learn 50:5–43
5. Bengio Y, Senécal JS (2003) Quick training of probabilistic neural nets by importance sampling. In: Proceedings of artificial intelligence statistics 2003 (AIStats 2003)
6. Bengio Y, Senécal JS (2008) Adaptive importance sampling to accelerate training of a neural probabilistic language model. IEEE Trans Neural Netw 19:713–722
7. Yuan C, Druzdzel MJ (2006) Importance sampling algorithms for Bayesian networks: principles and performance. Math Comput Model 43:1189–1207
8. Blumensath T, Davies M (2005) A fast importance sampling algorithm for unsupervised learning of over-complete dictionaries. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing, 2005 (ICASSP '05), March 18–23, vol 5, pp 213–216
9. Uchibe E, Doya K (2004) Competitive-cooperative-concurrent reinforcement learning with importance sampling. In: Proceedings of the 8th international conference on simulation of adaptive behavior: from animals and animats, July 13–17, pp 287–296
10. Chen JC, Lu D, Sadowsky JS, Yao K (1993) On importance sampling in digital communications. Part I: fundamentals. IEEE J Sel Areas Commun 11:289–299

11. Wolfe RJ, Jeruchim MC, Hahn PM (1990) On optimum and suboptimum biasing procedures for importance sampling in communication simulation. IEEE Trans Commun 38:639–647
12. Al-Qaq WA, Devetsikiotis M, Townsend JK (1995) Stochastic gradient optimization of importance sampling for the efficient simulation of digital communication systems. IEEE Trans Commun 43:2975–2985
13. Smith PJ, Shafi M, Gao H (1997) Quick simulation: a review of importance sampling techniques in communications systems. IEEE J Sel Areas Commun 15:597–613
14. Al-Qaq WA, Townsend JK (1997) A stochastic importance sampling methodology for the efficient simulation of adaptive systems in frequency nonselective Rayleigh fading channels. IEEE J Sel Areas Commun 15:614–625
15. Orsak GC, Aazhang B (1991) Constrained solutions in importance sampling via robust statistics. IEEE Trans Inf Theory 37:307–316
16. Sanz-González JL, Andina D (1999) Performance analysis of neural network detectors by importance sampling techniques. Neural Process Lett 9:257–269
17. Gerlach K (1999) New results in importance sampling. IEEE Trans Aerosp Electron Syst 35:917–925
18. Orsak GC (1993) A note on estimating false alarm rates via importance sampling. IEEE Trans Commun 41:1275–1277
19. Hoogerheide LF, Kaashoek JF, van Dijk HK (2007) On the shape of posterior densities and credible sets in instrumental variable regression models with reduced rank: an application of flexible sampling methods using neural networks. J Econom 139:154–180
20. Hoogerheide LF, van Dijk HK (2010) Bayesian forecasting of value at risk and expected shortfall using adaptive importance sampling. Int J Forecast 26(2):231–247
21. Gandhi PP, Ramamurti V (1997) Neural networks for signal detection in non-Gaussian noise. IEEE Trans Signal Process 45:2846–2851
22. Andina D, Sanz-González JL (1996) Comparison of a neural network detector vs. Neyman–Pearson optimal detector. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing (ICASSP'96), May 7–10, vol 6, pp 3573–3576
23. Jarabo-Amores MP, Rosa-Zurera M, Gil-Pita R, López-Ferreras F (2009) Study of two error functions to approximate the Neyman–Pearson detector using supervised learning machines. IEEE Trans Signal Process 57:4175–4181
24. Sanz-González JL, Andina D (2001) Importance sampling techniques in neural detector training. Lect Notes Comput Sci 2167:431–436
25. Sanz-González JL, Andina D, Seijas J (2002) Importance sampling and mean-square error in neural detector training. Neural Process Lett 16:259–276
26. Rosa-Zurera M, Jarabo-Amores P, López-Ferreras F, Sanz-González JL (2005) Comparative analysis of importance sampling techniques to estimate error functions for training neural networks. In: Proceedings of the IEEE 13th workshop on statistical signal processing, July 17–20, pp 121–125
27. Hachiya H, Akiyama T, Sugiayma S, Peters J (2009) Adaptive importance sampling for value function approximation in off-policy reinforcement learning. Neural Netw 22:1399–1410
28. Poor HV (1994) An introduction to signal detection and estimation, 2nd edn. Springer, Berlin
29. Hampshire JB, Pearlmutter B (1990) Equivalence proofs for multilayer perceptron classifiers and the Bayesian discriminant function. In: Proceedings of the 1990 Connectionist Models Summer School, pp 159–172
30. Van Trees HL (2003) Detection, estimation and modulation theory. Part I. Wiley, New York
31. Haykin S (2008) Neural networks and learning machines, 3rd edn. Prentice-Hall, London
32. Seiffer U (2001) Multiple layer perceptron training using genetic algorithms. In: Proceedings of the 2001 European symposium on artificial neural networks, pp 159–164
33. Montana DJ, Davis L (1989) Training feedforward neural networks using genetic algorithms. In: Proceedings of 11th international joing conference on artificial intelligence, pp 762–767
34. Nguyen D, Widrow B (1990) Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In: Proceedings of the 1990 international joint conference on neural networks (IJCN 1990), June 17–21, vol 3, pp 21–26