Adjusting the balance between alpha and beta risks in NN classifiers

Philippe Thomas, Marie-Christine Suhner, Hind Bril El Haouzi

Athens Journal of Sciences 2024, 11: 1-12
https://doi.org/10.30958/ajs.X-Y-Z

Adjusting the Balance between Alpha and Beta Risks in NN Classifiers

By Philippe Thomas[*]*, Marie-Christine Suhner*[±] *&*
Hind Bril El Haouzi[°]

This paper delves into classification tasks, where data is categorized into binary classes, such as fraudulent/non-fraudulent or sick/not sick as example. Employing a statistical approach, this task entails utilizing hypothesis testing. Tuning this test involves selecting an acceptable risk alpha (associated with false positives), thereby implicating a beta risk (related to false negatives). In classification challenges, the principal aim is to mitigate the misclassification rate. However, the determination of these two risks is not be discretionary but rather enforced by the learning process, particularly evident when employing neural networks. This paper seeks to propose a modification of the learning algorithm for multilayer perceptron aimed at effectively balancing these risks. This adaptation hinges on leveraging a weighted criterion to minimize errors, accounting for the signs of different error types. This methodology is assessed across two benchmarks: a simulated dataset and a genuine medical dataset.

Keywords: *neural network, multilayer perceptron, learning, classification, hypothesis test*

Introduction

A common challenge in various domains is the ability to confidently classify data into two exclusive classes. This is particularly relevant in fields such as fraud detection (O'Kelly 2004), IT security (Hänisch & Karg 2019), medical diagnosis (Guyatt et al. 1995), and banking loan approval (Hidayah & Saptarini 2019). Historically, a statistical approach has been employed to address such problems using hypothesis testing.

Hypothesis testing entails the evaluation of competing hypotheses regarding population characteristics, typically denoted as H0 (null hypothesis) and H1 (alternative hypothesis). This methodology introduces two potential errors: Type I error (alpha), representing the risk of falsely rejecting the null hypothesis H0, and Type II error (beta), indicating the risk of falsely accepting H0. The interrelation between these risks, alpha and beta, is notable; an increase in alpha risk leads to a decrease in beta risk and vice versa. Thus, the fundamental principle of statistical studies involves selecting an acceptable alpha risk to construct the test, facilitating risk management.

[*]Associate Professor, University of Lorraine, CRAN, France.
[±]Assistant Professor, University of Lorraine, CRAN, France.
[°]Professor, University of Lorraine, CRAN, France.

Another, more recent approach is to use machine learning tools, in particular neural networks, to solve these classification problems (Bao et al. 2022, Chandrasekaran 1983, Mytnyk et al. 2023, Joolfoo & Hosany 2023). However, these tools work differently. It involves defining a criterion to be minimized (generally the quadratic criterion) and using a criterion minimization algorithm (gradient backpropagation) to minimize the misclassification rate. The risks of false positives and false negatives are then imposed. This point is particularly critical when the data is poorly balanced, as in reliability studies where few data correspond to defects. In this case, these approaches lead to models that are biased in favor of the most represented class (Castro & Braga 2013). In extreme cases, a classification model that always assigns data to the same class will perform very well in terms of misclassification rate. To address this problem, cost-sensitive approaches have been proposed (Thomas 2015, Zadrozny et al. 2003, Zadrozny & Elkan 2001). However, these approaches still do not allow to control first- and second-species risks.

The main objective of this paper is to propose a modification of the learning algorithm based on a particular choice of the criterion to be minimized, exploiting the fact that the errors associated with a false alarm or non-detection are not of the same sign. A second objective is to illustrate the use of this approach in a medical context, in order to differentiate the treatment of patients according to the degree of confidence we have in the classification result.

In the next section, the structure of the multilayer perceptron used is described and the proposed learning algorithm is presented. Part 3 is dedicated to the presentation of the simulation example used to illustrate the performance of the algorithm. An application to a medical field is proposed and discussed in the following section before concluding.

Multilayer Perceptron

Structure

According to Cybenko (1989) and Funahashi (1989), a multilayer neural network that includes only one hidden layer with a sigmoidal activation function and an output layer can accurately approximate all nonlinear functions. For the sake of simplicity, we will only focus on the single output case in this presentation. However, it is important to note that the multi-output case can be derived from this scenario with ease. The equation for the network output $\hat{y}$ is defined as follows:

$$\hat{y} = g_o\left(\sum_{h=1}^{n_o} w_h^2 . g_h\left(\sum_{i=1}^{n_i} w_{hi}^1 . x_i + b_h^1\right) + b\right) \quad (1)$$

where x_i represents the n_i inputs, w_{hi}^1 represents the connecting weights between the input and hidden layers, b_h^1 represents the hidden neuron biases, $g_h(.)$ represents the activation function of the hidden neurons (hyperbolic tangent), w_h^2 represents the connecting weights between the hidden and output layers, b represents the bias of the output neuron, and $g_o(.)$ represents the activation function of the output neuron.

Because the problem at hand is a classification task, the sigmoidal function $g_o(.)$ was chosen. The accuracy of the model is heavily influenced by the initial parameter set due to the local search for minimum during MLP learning. Various initialization algorithms have been proposed in the past (Thomas & Bloch 1997). The modification of the Nguyen and Widrow (NW) algorithm (Nguyen & Widrow 1990) utilized in this study allows for a random initialization of weights and biases while to be optimally placed in the input space (Demuth et al. 1994).

Proposed learning Algorithm

The primary objective of the learning algorithm in a classification problem is to devise a model capable of correctly associating each pattern with its respective class. This model is directly derived from a training dataset. To achieve this, the goal is to minimize the mean square error between the predicted output of the model and the actual desired output. Therefore, the classical quadratic criterion to minimize is expressed as:

$$V(\theta) = \frac{1}{2n}\sum_{k=1}^{n} \varepsilon^2(k,\theta) \qquad (2)$$

where θ encompasses all the unknown network parameters (weights and biases), n is the size of the training dataset, and ε represents the prediction error or residual given by:

$$\varepsilon(k,\theta) = y(k) - \hat{y}(k,\theta) \qquad (3)$$

where $y(k)$ is the actual desired class of pattern k and $\hat{y}(k,\theta)$ is the predicted class by the network.

This criterion does not allow for adjusting the model based on the selected acceptable risks (alpha and beta). However, the sign of the residual ε(k,θ) provides information about the type of error. If hypothesis H0 (or H1) suggests that the data under consideration, *k,* belongs to class 0 (or class 1), then a residual of $\varepsilon(k,\theta) < 0$ corresponds to a false positive error (type I or alpha error), while a residual of $\varepsilon(k,\theta) > 0$ corresponds to a false-negative error (type II or beta error).

To utilize this information, the criterion to be minimized (2) can be modified by assigning weight $C_{\alpha\beta}(k)$ to the residual k based on its sign, giving more or less importance as necessary:

$$V(\theta) = \frac{1}{2n}\sum_{k=1}^{n} C_{\alpha\beta}(k).\varepsilon^2(k,\theta) \qquad (4)$$

where the weight is determined by:

$$\begin{cases} C_{\alpha\beta}(k) = C_\alpha & if \quad \varepsilon(k,\theta) < 0 \\ C_{\alpha\beta}(k) = C_\beta & if \quad \varepsilon(k,\theta) > 0 \end{cases} \qquad (5)$$

Here, C_α (or C_β) is a positive integer value that adjusts the influence of the considered residual $\varepsilon(k,\theta)$ if it corresponds to a type I error (or type II). Choosing C_α and C_β such that $C_\alpha > C_\beta$ (or $C_\alpha < C_\beta$) implies that the learned model will reduce the alpha risk (or beta risk). Setting $C_\alpha = C_\beta$ corresponds to the classical quadratic criterion (2).

The classical Gauss-Newton algorithm is obtained by the 2nd-order Taylor series expansion of the minimization criterion (4):

$$\hat{\theta}^{i+1} = \hat{\theta}^i - (H(\hat{\theta}^i))^{-1} V'(\hat{\theta}^i) \qquad (6)$$

where $\hat{\theta}^i$ is the set of network parameters estimated at iteration i, $V'(\hat{\theta}^i)$ is the criterion gradient and $H(\hat{\theta}^i)$ is the Hessian Matrix. The gradient of the criterion is given by:

$$V'(\theta) = -\frac{1}{n}\sum_{k=1}^{n} \psi(k,\theta).C_{\alpha\beta}(k).\varepsilon(k,\theta) \qquad (7)$$

where $\psi(k,\theta)$ is the gradient of $\hat{y}(k,\theta)$ with respect to q.

The Levenberg-Marquardt (Levenberg, 1944; Marquardt, 1963) update rule can be used to estimate the Hessian matrix:

$$H(\theta) = \frac{1}{n}\sum_{k=1}^{n} \psi(k,\theta).C_{\alpha\beta}(k).\psi^T(k,\theta) + \beta I \qquad (8)$$

beta

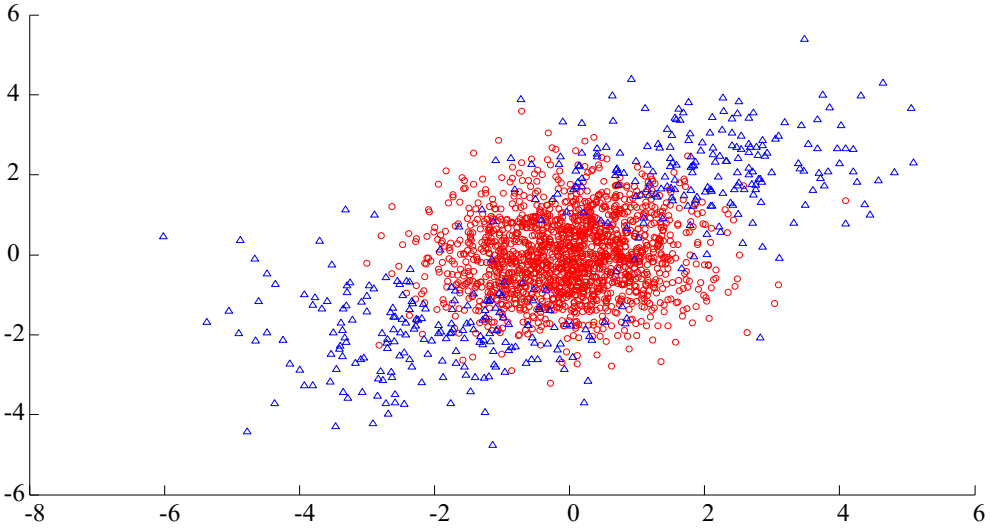
where I is the identity matrix and ~~b~~ a small non negative scalar which must be adapted during the learning process.

Simulation Example

To demonstrate the proposed learning algorithm, we use a simple simulation example derived from the one proposed by (Lin et al. 2002). The example involves a population comprising two subpopulations. The positive subpopulation follows a bivariate normal distribution with a mean of $(0, 0)^T$ and a covariance matrix of diag(1, 1). On the other hand, the negative subpopulation follows two bivariate normal distributions. The first subpopulation has a mean of $(2, 2)^T$ with a covariance of diag(2, 1), while the second subpopulation has a mean of $(-2, -2)^T$ with a covariance of diag(2, 1). The population consists of a positive subpopulation and a negative subpopulation. The positive and negative subpopulations account for 80% and 20% of the total population, respectively. The subpopulation with negative values is balanced and follows two distinct laws to prevent linear separability of the two classes.

Figure 1 displays the distribution of the two classes in the space of the two inputs. The red circles represent class0, while the blue triangles represent class1. It is evident that these two classes are partially overlapping. This fact implies that even the most accurate classifier is unable to perform its task without producing misclassifications.

Figure 1. *Representation of the Simulation Dataset*



A dataset of 2,000 data points is generated based on the aforementioned distribution. The dataset is then randomly split into a training set and a validation set, each containing 1,000 data points. The training set is utilized to construct a neural classifier (MLP) with a structure consisting of 2 inputs and 10 hidden neurons, which is deemed appropriate for the problem. The size of the hidden layer was determined through a trial-and-error approach.

To prevent the issue of local minimum trapping, the training process utilizes 20 distinct sets of initial weights for each experience.

The misclassification rate, also known as the error rate or 'zero-one' score function (Hand et al. 2001), is the classical criterion for evaluating classifiers:

$$S_{01} = \frac{1}{n}\sum_{k=1}^{n} I(y(k), \hat{y}(k,\theta)) \quad (9)$$

where $I(a,b) = 1$ when $a \neq b$ and 0 otherwise.

Two other indicators must be determined: the false alarm rate (*FA*) and the non-detection rate (*ND*). These indicators are:

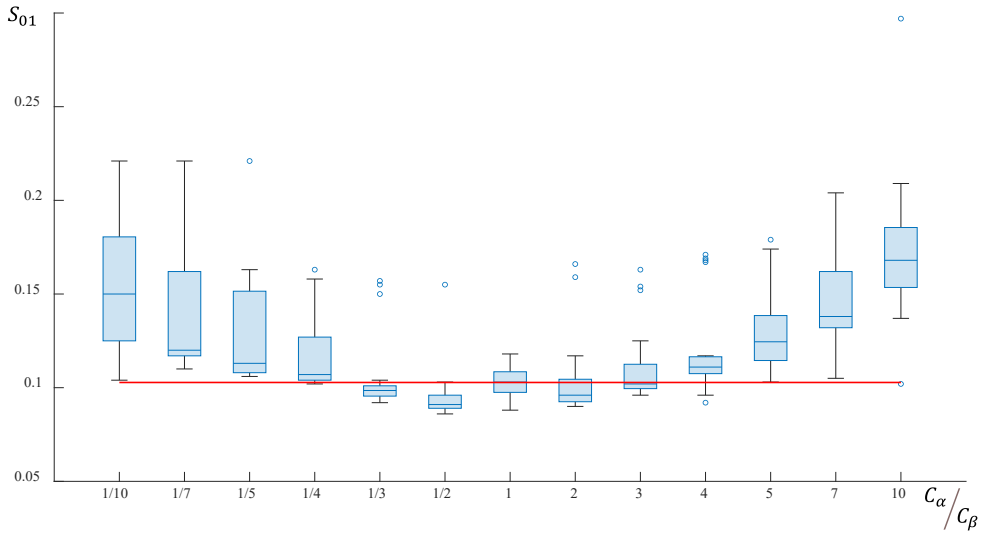
$$\begin{cases} FA = \frac{FP}{FP+TN} \\ ND = \frac{FN}{FN+TP} \end{cases} \quad (10)$$

where *FP* represents the number of false positives, *TN* represents the number of true negatives, *FN* represents the number of false negatives, and *TP* represents the number of true positives.

First, we will examine the effect of parameters choice on the misclassification rate. Specifically, we will focus on the impact of the C_α and C_β parameters (C_α =1 when $C_\beta \neq 1$ and vice versa) in equations 4 and 5. Figure 2 displays the dispersion

(boxplot) of misclassification rates obtained from 20 different sets of initial weights as the C_α/C_β ratio varies between 1/10 and 10.

Figure 2. *Boxplot of the S_{01} in Function of the C_α/C_β ratio for Simulation Example*



The reference model is the model that gives the lowest misclassification rate using the classical quadratic criterion ($C_\alpha/C_\beta = 1$). The results of all other models are compared with this model using a hypothesis test (proportion comparison) with a confidence level of 95%. The red line represents the acceptance limit of the test. Below this line, the considered model is statistically equivalent to the reference model; above this line, the considered model is statistically worse than the reference model. Figure 2 demonstrates that selecting a C_α/C_β ratio between 1/3 and 3 does not result in a decline in performance. In fact, for all boxplots analyzed, the median value remains below the red line, indicating that at least 50% of the models produce results that are statistically equivalent to those of the reference model. However, for values less than 1/3 or greater than 3, this statement no longer holds true. Therefore, it is important to make a reasonable choice for the C_α and C_β parameters and not exceed a value of 3.

Table 1 shows the rates (misclassification S_{01}, false alarm FA and non-detection ND) for the best model among the 20 trained with different C_α/C_β ratios obtained on the validation dataset. According to the results shown in figure 2, only ratios between 1/3 and 3 are shown.

The misclassification rates obtained with these five best models are statistically equivalent. Therefore, the proposed learning algorithm does not improve or degrade the misclassification rate. Additionally, a study of the false alarm rate (corresponding to the alpha risk) reveals that adjusting the C_α and C_β parameters can result in a range of rates between 1% and 7%. Similarly, adjusting these same

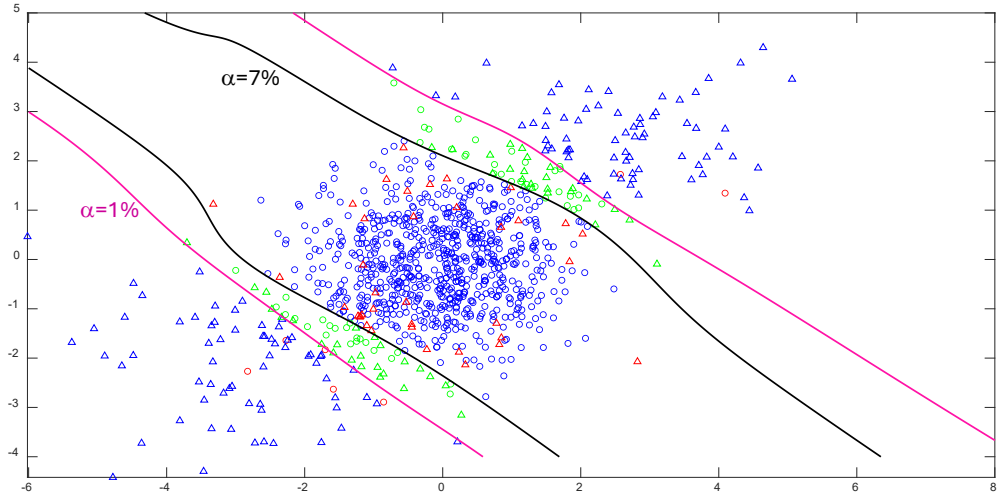
parameters can cause the non-detection rate (beta risk) to vary between 17% and 38%.

Table 1. *Misclassification, False Alarm and Non-detection Rates for the Best Models in Function of the* C_α/C_β *Ratio*

C_α/C_β	1/3	1/2	1	2	3
S_{01}	9.2%	8.6%	8.8%	9.0%	9.6%
FA (α risk)	0.9%	3.5%	3.7%	5.0%	7.6%
ND (β risk)	38.5%	26.7%	26.7%	23.1%	16.7%

Figure 3 displays the classification results obtained using two extreme models. The classification limits of the models are represented by a black curve for the model obtained with the ratio $C_\alpha/C_\beta = 3$ and a pink curve for the model obtained with the ratio $C_\alpha/C_\beta = 1/3$. Rounds represent data belonging to class 0, while triangles represent data belonging to class 1. The blue data points are well classified by both models, while the red data points are misclassified by both models. Data in green are those well classified by one model and misclassified by the other. It illustrates the usefulness of the proposed approach by defining three zones: a zone of high probability of belonging to class 0 (inside the black line), a zone of high probability of belonging to class 1 (outside the pink curve), and a zone of uncertainty between the two curves.

Figure 3. *Results of the Classification Models*



Medical Example

To illustrate the application of the approach, a medical example using a real dataset is presented. The dataset considered here concerns breast cancer diagnosis

(Mangasarian & Wolberg 1990) and can be downloaded from the UCI website[1]. This dataset includes 569 instances. For each instance, 11 variables are collected including the sample ID (not used here), the diagnosis (benign/malignant) which is the target of our model, and 9 features describing the tumor: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, mitoses, which are used as input of our model.

This dataset is subdivided into learning and validation datasets including 300 and 383 instances respectively (instances including missing data are removed).

The structure of the MLP used to learn this classification problem is composed of 9 inputs, 1 output and the size of the hidden layer is fitted to 5 neurons by using a trial-and-error strategy. As for the preceding example, to prevent the issue of local minimum trapping, the training process utilizes 20 distinct sets of initial weights for each experience.

Figure 4 is based on the same principle as figure 2. It shows the dispersion of misclassification rates obtained from 20 different sets of initial weights as the C_α / C_β ratio varies between 1/10 and 10. As for the figure 2, the reference model is the model that gives the lowest misclassification rate using the classical quadratic criterion ($C_\alpha / C_\beta = 1$). The results of all other models are compared with this model using a hypothesis test with a confidence level of 95% and the red line represents the acceptance limit of the test.

Figure 4 illustrates that the conventional quadratic criterion is not optimal in this context. In fact, the boxplot ($C_\alpha / C_\beta = 1$) reveals that over three-quarters of the models exhibit statistically inferior results relative to the reference model. Only the 1/4, 1/5, 1/7, and 1/10 ratios exhibit inferior performance in this regard. Furthermore, the impact of the C_α / C_β ratio is not symmetrical. Indeed, selecting $C_\alpha > C_\beta$ (alpha risk reduction) leads to markedly improved results (for ratios 2, 3, 4, and 5, more than half of the models are statistically equivalent to the reference model). Conversely, selecting the opposite choice leads to only about one quarter of the models being statistically equivalent to the reference model. This may be attributed to the fact that the data are slightly unbalanced (2/3 class 0; 1/3 class 1).

[1]https://archive.ics.uci.edu/dataset/15/breast+cancer+wisconsin+original.

Figure 4. *Boxplot of the S_{01} in Function of the C_α/C_β Ratio for Breast Cancer Problem*

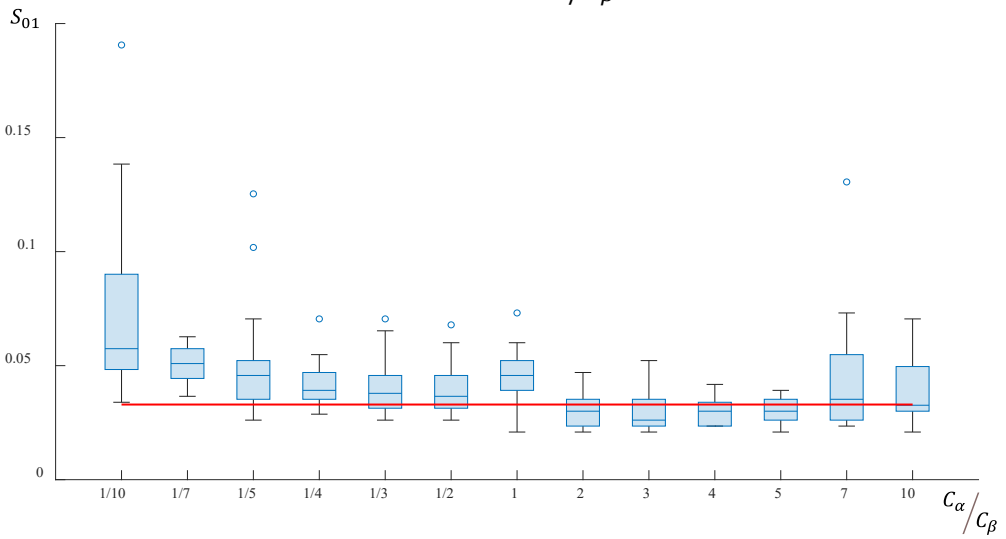


Table 2 shows the rates (misclassification S_{01}, false alarm FA and non-detection ND) for the best model among the 20 trained with different C_α/C_β ratios obtained on the validation dataset. As for the preceding case, only ratios between 1/3 and 3 are shown.

The misclassification rates obtained with these five best models are statistically equivalent. Therefore, the proposed learning algorithm does not improve or degrade the misclassification rate. Additionally, a study of the false alarm rate (corresponding to the alpha risk) reveals that adjusting the C_α and C_β parameters can result in a range of rates between 1.2% and 3%. Similarly, adjusting these same parameters can cause the non-detection rate (beta risk) to vary between 0% and 5%.

Table 2. *Misclassification, False Alarm and Non-detection Rates for the Best Models in Function of the C_α/C_β Ratio for Breast Cancer Problem*

C_α/C_β	1/3	1/2	1	2	3
S_{01}	2.6%	2.6%	2.1%	2.1%	2.1%
FA (α risk)	1.6%	1.2%	2.8%	2.8%	3.1%
ND (β risk)	4.7%	5.5%	0.8%	0.8%	0.0%

In a similar vein as illustrated Figure 3, it is possible to select two distinct models in order to differentiate treatments according to the cancer predictions provided by the two models. The two models selected here are the optimal models obtained with ratios of ½ and 3, respectively, and indicated in yellow in Table 2. The use of the ratio 3 model is particularly advantageous, as it offers a risk of non-detection of 0%. With this model, there is a high probability of correctly diagnosing all sick patients. Nevertheless, with this model, over 3% of cases are false positives. In contrast, the model obtained with the ratio ½ leads to a non-detection rate of 5.5%

but reduces the false positive rate to 1.2%. It is then possible to easily construct three groups of patients according to the results of these two models:

- Group 1: Patients declared healthy by both models. In this case, it can be assumed with a high degree of certainty that the patients are indeed healthy, and that regular follow-up will be sufficient.
- Group 2: Patients diagnosed with cancer by both models. In this case, the probability is very high that these patients do in fact have cancer, and treatment can begin very quickly.
- Group 3: Patients whose diagnose differ between the two models. In this instance, the risk of error is high, necessitating further investigations to refine the diagnosis.

Table 3 is derived from the confusion matrices. It presents the results obtained on the validation set by mapping predicted classes to actual classes. In comparison to a conventional confusion matrix, an additional column entitled "undetermined" is added, indicating the number of instances that are well classified by one model and poorly classified by the other. The "class 0" column corresponds to Group 1 of patients reported as not ill. Here, we find our non-detection rate of 0. The class 1 column corresponds to Group 2 of patients who were declared to have cancer. We note that only three of these patients were in fact healthy. The undetermined column corresponds to Group 3, which requires further investigation. This group includes almost as many healthy patients as cancer patients.

Table 3. *Modified Confusion Matrix*

		Predicted class		
		healthy	undetermined	cancer
real class	healthy	245	7	3
	cancer	0	10	118

Conclusion

This paper addresses the issue of two-class classification using neural networks, while controlling for first- and second-species risks. The primary concept is based on the use of a criterion to be minimized, including a weight that allows for the reduction of type I or type II errors to be given priority.

The proposed approach was tested on a simulation example where the impact of the choice of weight on the performance of the model learned in terms of misclassification, false alarm, and non-detection rates was evaluated. The results demonstrated that this weight favored a reduction in the number of false alarms or, conversely, non-detections, without compromising the misclassification rate. The proposed algorithm was then tested on a medical dataset to illustrate the value of the approach in differentiating patient treatment according to diagnostic results.

One limitation of this approach that has not been addressed here is the potential impact of data pollution by outliers. Future work will be directed towards investigating the impact of outliers on the robustness of learned models.

Acknowledgments

This work was financially supported by the ANR-22-CE46-0010 JUNEAU.

References

Bao Y, Hilary G, Ke B (2022) Artificial Intelligence and Fraud Detection. In V Babich, JR Birge, G Hilary (eds.), *Innovative Technology at the Interface of Finance and Operations: Volume I*, 223–247. Springer International Publishing.

Castro CL, Braga AP (2013) Novel Cost-Sensitive Approach to Improve the Multilayer Perceptron Performance on Imbalanced Data. *IEEE Transactions on Neural Networks and Learning Systems* 24(6): 888–899.

Chandrasekaran B (1983) On Evaluating Artificial Intelligence Systems for Medical Diagnosis. *AI Magazine* 4(2): Article 2.

Cybenko G (1989) Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems* 2(4): 303–314.

Demuth H, Beale M, Hagan M (1994) *Neural network toolbox*. Mathworks.

Funahashi K-I (1989) On the approximate realization of continuous mappings by neural networks. *Neural Networks* 2(3): 183–192.

Guyatt G, Jaeschke R, Heddle N, Cook D, Shannon H, Walter S (1995) Basic statistics for clinicians: 1. Hypothesis testing. *CMAJ: Canadian Medical Association Journal* 152(1): 27–32.

Hand D, Mannila H, Smyth P (2001) Principles of data mining. 2001. *MIT Press. Sections* 6(3): 2–6.

Hänisch T, Karg C (2019) Estimating the success of it security measures in industry 4.0 environments using monte carlo simulation on attack defense trees. *Athens Journal of Technology & Engineering* 6(4): 211–222.

Hidayah E, Saptarini GD (2019) Pentagon Fraud Analysis in Detecting Potential Financial Statement Fraud of Banking Companies in Indonesia. In *Proceeding UII-ICABE*, 89–102.

Joolfoo MBA, Hosany MA (2023) Machine learning solutions in combating COVID-19: State of the art and challenges. *Athens Journal of Technology* 10(1): 71–88.

Levenberg K (1944) A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*.

Lin Y, Lee Y, Wahba G (2002) Support Vector Machines for Classification in Nonstandard Situations. *Machine Learning* 46(1): 191–202.

Mangasarian OL, Wolberg WH (1990) *Cancer diagnosis via linear programming*. University of Wisconsin-Madison Department of Computer Sciences.

Marquardt DW (1963) An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *Journal of the Society for Industrial and Applied Mathematics* 11(2): 431–441.

Mytnyk B, Tkachyk O, Shakhovska N, Fedushko S, Syerov Y (2023) Application of Artificial Intelligence for Fraudulent Banking Operations Recognition. *Big Data and Cognitive Computing* 7(2): Article 2.

Nguyen D, Widrow B (1990) Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In *1990 IJCNN International Joint Conference on Neural Networks*, 21–26 volume 3.

O'Kelly M (2004) Using statistical techniques to detect fraud: A test case. *Pharmaceutical Statistics* 3(4): 237–246.

Thomas P (2015) Perceptron learning for classification problems: Impact of cost-sensitivity and outliers robustness | IEEE Conference Publication | IEEE Xplore. In *IJCCI'15 7th Int. Joint Conf. on Computational Intelligence*.

Thomas P, Bloch G (1997) *Initialization of one hidden layer feedforward neural networks for non-linear system identification* 4: 295–300.

Zadrozny B, Elkan C (2001) Learning and making decisions when costs and probabilities are both unknown. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 204–213.

Zadrozny B, Langford J, Abe N (2003) Cost-sensitive learning by cost-proportionate example weighting. In *Third IEEE International Conference on Data Mining*, 435–442.