# Neural Network Method for failure detection with skewed class distribution

*K. Carvajal Cuello*[1]*, M. Chacón*[1]*, D. Mery**[2] *and G. Acuña*[1]

[1]Departamento de Ingeniería Informática
Universidad de Santiago de Chile (USACH)
Avda. Ecuador 3659, PoBox 10233, Santiago, Chile.

* Corresponding author:
[2]Departamento de Ciencia de la Computación
Pontificia Universidad Católica de Chile
Avda. Vicuña Mackena 4860 (143), Santiago, Chile
e-mail: dmery@ieee.org

**Abstract**

The automatic detection of flaws through non-destructive testing uses pattern recognition methodology with binary classification. In this problem a decision is made about whether or not an initially segmented hypothetical flaw in an image is in fact a flaw. Neural classifiers are one among a number of different classifiers used in the recognition of patterns. Unfortunately, in real automatic flaw detection problems there are a reduced number of flaws in comparison with the large number of non-flaws. This seriously limits the application of classification techniques such as artificial neuronal networks due to the imbalance between classes. This work presents a new methodology for efficient training with imbalances in classes. The premise of the present work is that if there are sufficient cases of the smaller class, then it is possible to reduce the size of the larger class by using the correlation between cases of this latter class, with a minimum information loss. It is then possible to create a training set for a neuronal model that allows good classification. To test this hypothesis a problem of great interest to the automotive industry is used, which is the radioscopic inspection of cast aluminium pieces. The experiments resulted in perfect classification of 22936 hypothetical flaws, of which only 60 were real flaws and the rest were false alarms.

**Key Words:** automatic flaw detection, neuronal networks, classification, aluminium castings.

## 1. Introduction

The automatic detection of flaws in non-destructive testing based on image processing uses pattern recognition methodology for its implementation. This process has the following stages: image formation, pre-processing, segmentation, feature extraction, and their classification (Mery *et al*, 2003). Image formation is obtained by X-ray irradiation of the studied piece creating a digital image of the object on the basis of the original image obtained. The purpose of pre-processing is to improve the quality of the image for better recognition of possible flaws, reducing noise, enhancing contrast, and restoring. Segmentation consists of obtaining regions of the images that correspond to possible flaws. During feature extraction segmented regions are measured, and finally in classification, in accordance with the extracted features segmented regions are separated into two classes: "defects" (flaws) and "regular structures" (non-flaws). Because the classification is binary, it becomes a flaw detection problem.

There are different classifiers used in the recognition of patterns such as linear discriminators, (Schalkoff, 1992), those based on distances (Tou *et al*., 1974), Bayesian classifiers (Tou *et al*., 1974), Neural Networks (Bishop, 1995) and others.

Neuronal classifiers exhibit some significant advantages over other classifiers, such as their ability to perform non-linear discrimination, the possibility of including different types of variables (real, nominal, and binary) in the same model, and their processing and adaptive capacity (Mitchell,1997). This allows the inclusion of a great deal of features for discrimination. Nonetheless, the importance of these advantages could be diminished when there is a great deal of difference in the size of the classes.

Unfortunately, in real automatic flaw detection problems, the number of flaws is very small in comparison to the large number of non-flaws. This seriously limits the application of powerful classification techniques such as artificial neuronal networks due to the fact that training methods based on minimization of the mean quadratic error do not adequately weigh the smaller subset (Haykin, 1994).

This problem, known in the pattern recognition literature as a *skewed class distribution*, has been dealt with in other works, e.g. Chan *et al*, 1999 and Chih *et al*, 2002, which propose the creation of training subsets, combining some of the larger group's cases with the totality of the smaller group's cases, (the smaller group is replicated in all of the subsets). Subsequently, a classification model is created for each subset, and finally a meta-classifier which combines the predictions made by the individual classifiers. The ratio between cases in these works is 1:49 and 1:4, respectively.

The basis of the present work is that if there are sufficient examples of the smaller size class, it is possible to reduce the number of the other class, with a minimum loss of information, thus creating a representative subset of the larger class, which together with the smaller class make up a training set for a neuronal model that allows good classification.

The working hypothesis is that the training set of the neuronal network can be made up of the data from the smaller class size, and a reduced set of cases from the larger class size, which will allow increasing the discrimination capacity of the classifier, without modifying the error minimization algorithms used by neuronal networks. The reduction of the larger class is carried out by eliminating those cases that have a certain correlation with other cases in the same class.

In order to evaluate this hypothesis, we have used a problem which is of great interest in the automotive industry, the radioscopic inspection of cast pieces, which have already been tested with other methods of classification (Mery *et al*, 2003). The results of this inspection are 60 cases of flaws, and 22876 cases of non-flaws. The difference between these two groups is notable, and the ratio of flaws to non-flaws is 1:381.

The present work is divided into the following sections: Section 2 presents a brief description of the methods used to solve this problem, Section 3 presents our results, and Section 4 presents the conclusions of this work.

# 2. Description of Methods

## 2.1 Artificial Neuronal Networks

Artificial neuronal networks are mathematical tools derived from what is known about the mechanisms and physical structure of biological learning, based on the function of a neuron. They are parallel structures for distributed processing of information (Haykin, 1994). The basic processing unit is the neuron, made up of multiple inputs and only one output. This output is determined by an activation function that operates on input values, and a transfer function that operates on the activation value. In other words, if we consider $X_i$ inputs, $W_i$ weights, $A$ as the activation value and $Z$ as the output value of the neuron, the values of $A$ and $Z$ can be described by:

$$A = \sum_{i=1}^{n} W_i \cdot X_i + bias \, , \tag{1}$$

$$Z = g(A), \tag{2}$$

where *g(A)* is the so-called transfer function and is generally a sigmoid or linear function.

The structure of a neuronal network can have one or more neurons and depending on the type of problem and the training, these networks receive different names. They have the capacity to associate and classify patterns, compress data, perform process control and approximate non-linear functions (Mitchell, 1997).

The most often used type of neural network in classification is the Multi Layer Perceptron (MLP) which consists of sequential layers of neurons . The structure of an MLP is shown in Figure 1 where each neuron has equation (2) associated to it.

Backpropagation is the learning algorithm normally used to train this type of network. Its goal is to minimise the error function constructed from the difference between the desired and modelled output.

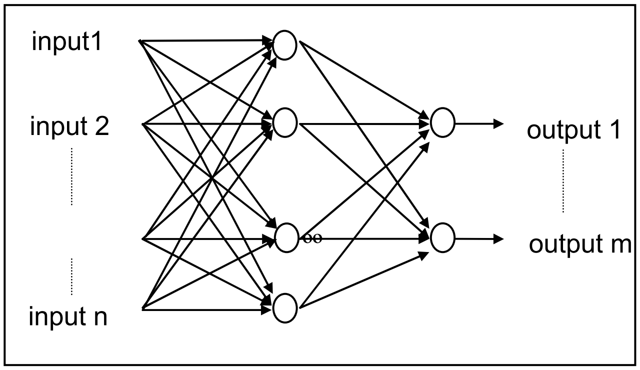

Figure 1: Multi Layer Perceptron

The initially developed backpropagation algorithm used a steepest descent first order method as learning rule. Nonetheless, other more powerful second order methods are in common use now. The method used in this paper is conjugate gradient, which consists of finding the gradient directions that satisfy:

$$\mathbf{d}^{(t+1)T}\mathbf{H}\mathbf{d}^{(t)} = 0, \tag{3}$$

where **d** is the slope direction and **H** is the Hessian matrix evaluated at point $\mathbf{w}^{(t+1)}$. Vector **w** is the network weight vector and it is updated by means of:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \lambda^{(t)}\mathbf{d}^{(t)}. \tag{4}$$

Parameter $\lambda^{(t)}$ is selected for minimising:

$$E(\lambda)=E(\mathbf{w}^{(t)} + \lambda\mathbf{d}^{(t)}). \tag{5}$$

where $E$ is the error function (Bishop, 1996).

### 2.2 Detecting flaws in cast pieces

As mentioned in the introduction, the automatic detection of flaws in non-destructive testing is commonly carried out by means of a pattern recognition method, which is made up of: image formation, pre-processing, segmentation, feature extraction, and classification (Mery *et al*, 2003).

Segmentation is carried out by means of an edge detection technique in which the digital image, obtained in the previous steps, is divided into disconnected regions with the purpose of separating those areas of interest from the rest of the image, thus detecting hypothetical flaws. In the feature extraction process the properties of each region obtained in segmentation are measured. The concept is that, on the basis of the extracted features it can be decided whether the “hypothetical flaw” corresponds to a “defect” (flaw) or to a “regular structure” (non-flaw).

The present study analyses the data presented in (Mery *et al*, 2003) generated on the basis of 50 radioscopic images cast aluminium pieces. These data correspond to a total of 22936 segmented regions of which 60 are defects and 22876 are regular structures.

The features extracted from the cast pieces correspond to two types: geometric features and intensity. The total number of features extracted from the pieces is, for this case, 405. Taking into consideration the computational cost required to process the 405 features only 28 were pre-selected. In order to carry out this pre-selection two methods were used, ROC analysis and the Fisher discriminant (Mery *et al*, 2003). Of the 405 features, those with an area under the ROC curve $A_z < 0.8$ and a Fisher discriminant $J < 0.2$ $J_{max}$, (where $J_{max}$ corresponds to the maximum Fisher discriminant obtained from evaluating all features), were eliminated. If two features had a correlation coefficient with an absolute value greater than or equal to 0.95, the one with the smaller $A_z$ was eliminated. Thus, of the 405 features 376 were eliminated, leaving only 28[1]. For details of the extracted and pre-selected features the reader is referred to Tables 1 and 2 of (Mery *et al*, 2003).

Figure 2 shows a representation, for the two first components (62.8 % of the information), of a Principal Component Analysis of the flaw and no-flaw cases which include the 28 features, in which the difficulty in separating these classes can be clearly observed.

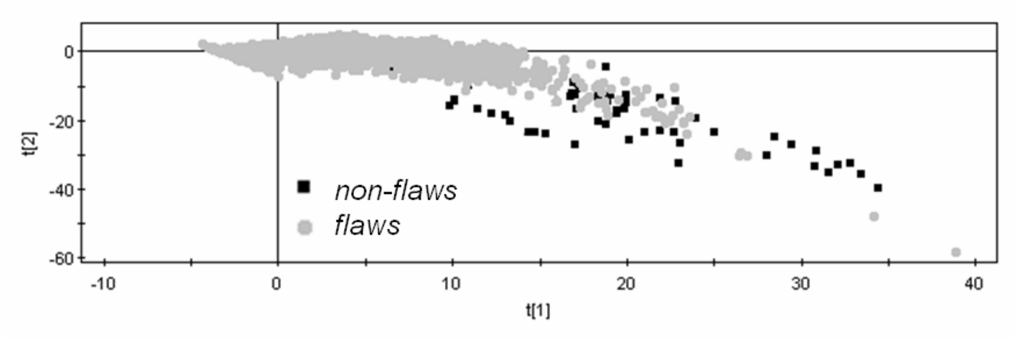

Figure 2: Principal Component Analysis of the 2936 cases of flaws and non-flaws

### 2.3. Neuronal Model for classifying unbalanced classes

In this problem there is a clear imbalance in the quantity of elements in each class, and the ratio of flaws to non-flaws is 1:381.

The hypothesis for this work is that by reducing the larger class and considering that there are sufficient examples or cases in the other class, a neuronal model can be constructed which uses the previously mentioned cases as a training set, and has the capacity to classify each case correctly, without having to modify the error minimization algorithms that the neuronal network use.

The general procedure for constructing a neuronal model that satisfactorily classifies cases into their respective classes is the following:

i. Normalise the data between 0 and 1 for both classes.

ii. Decrease the number of elements that belong to the larger class, leaving the group that is most representative of the class. This is achieved by eliminating highly correlated cases, the details of which will be explained later on.

iii. Create the neuronal model, considering an MLP with a hidden layer (Funahashi, 1989), a sigmoid transfer function for both the hidden layer and output, and a backpropagation learning method with conjugate gradient. The number of neurons in the hidden layer will depend on the number of cases available for training.

[1] The data are available at http://www.diinf.usach.cl/~dmery/papers/PANNDT2003a.htm

iv. Assemble the training set for the neuronal model, defining as elements of the set the subset of cases of the larger size class which are obtained after making the reduction referred to in point ii above, and all of the cases in the smaller class.

v. Train the network with the set assembled in step iv.

vi. Test the model on all cases, both flaws and non-flaws.

vii. Finally, use the same neuronal classifier to reduce even further the features used for classification. In this way considerable time is saved in the pre-processing and segmentation stages.

The procedure that allows the reduction of the larger class in such a way that the elements left represent the total set for that class, (point ii of the previous algorithm), is given below.

ii.1 Create groups of correlated cases. To this end, the quantity and type of cases with a correlation coefficient greater than 0.99 are calculated, and then for 0.98, and so forth until a correlation greater than 0 is reached. For each calculation the total set of cases is used. This process generates pairs (n,r) in which n represents the quantity of cases with a correlation greater than the valued indicated by r.

ii.2 Create a correlation curve placing the index of correlation r (where r $\in$ [0, 0.99]), on the ordinate in intervals of 0.01 and the on the abscissa the quantity of cases that are correlated with a correlation coefficient greater than r.

ii.3 Find a point on the correlation curve which indicates how many and what cases represent the larger class in order to use this point in the training of the neuronal network. In order to find the point it is necessary to progress as far as possible to the right of the correlation curve while maintaining a fixed height. Keeping a fixed height ensures that the information regarding the subset of cases is kept unaltered and moving to the right reduces the number of cases in the larger class. In order to find the cut-off point it is necessary to determine how much information about the cases one is willing to lose vs. the number of cases that represent the larger class. In this case for each 1% loss of information there must be a corresponding decrease in the number of cases of at least between 0.5 % and 1%. This point will indicate that the cases observed in the abscissas are correlated with a higher index of correlation than that indicated on the ordinate axis. In order to form the reduced set the complement of these cases is considered.

# 3. Analysis and discussion of results

In this section we will present the results obtained from various experiments carried out in which different sets are used for training the neuronal network. These results are analysed by looking at “sensitivity” ($S_n$) versus ‘1- specificity’ (1-$S_p$), defined as:

$$S_n = \frac{TP}{TP+FN}, \qquad 1-S_p = \frac{FP}{TN+FP} \tag{6}$$

where *TP* is the number of true positives (defects correctly classified), *TN* is the number of true negatives (regular structures classified correctly), *FP* is the number of false positives (false alarms or regular structures classified as defects) and *FN* is the number of false negatives (defects classified as regular structures). Ideally $S_n$=1 and 1-$S_p$= 0, i.e., all defects are detected without flagging false alarms.

### 3.1 Results obtained using all cases as a training set

The neuronal network created for this case was an MLP network, with a hidden layer and 76 neurons in that layer. A sigmoid transfer function was used, both for the hidden layer and the output.

The data set used was the set of 22936 cases of flaws and non-flaws used for training and testing. The results are shown in Table 1, Experiment 1. Although the training gave good results, it lasted three hours (considering that this job was carried out with an x86 Family 6, AT/AT Compatible machine with a Windows NT operating system and NeuroSolutions as the neuronal modeling tool) and was unable to recognise 100% of both flaws and regular structures. Additionally, when attempts are made to train the network with fewer features results decrease significantly.

Table 1. Performance of the Neuronal Network in Diverse Experiments

| **Experiment** | **Reduction of the number of cases in the larger class** | **Selected features** | *TP* | *FP* | $S_n$ | 1-$S_p$ |
|---|---|---|---|---|---|---|
| 1 | No | All | 59/60 | 2/22876 | 98.3% | 0.00% |
| 2 | Yes | All | 60/60 | 1/22876 | 100% | 0.00% |
| 3 | Yes | $Y_{DCT,}$ $Y_{DCT}$ (360, 376) | 60/60 | 88/22876 | 100% | 0.38% |
| 4 | Yes | $\Delta'_Q$, $F_1$, $Tx_5$ (33, 37, 186) | 60/60 | 0/22876 | 100% | 0.00% |
| 5 | Yes | K, $\Delta'_Q$, $F_1$ (31, 33, 37) | 60/60 | 14/22876 | 100% | 0.06% |
| 6 | Yes | $\Delta'_Q$, $F_1$, $Tx_3$ (33, 37, 128) | 59/60 | 3/22876 | 98.3% | 0.00% |
| 7 | Yes | K, $F_1$, $\sigma^2_g$ (31, 37, 59) | 59/60 | 30/22876 | 98.3% | 0.13% |
| 8 | Yes | $K_\sigma$, K, $Tx_5$ (30, 31, 179) | 59/60 | 36/22876 | 98.3% | 0.15% |
| 9 | Yes | C, $K_\sigma$, K (25, 30, 31) | 59/60 | 54/22876 | 98.3% | 0.24% |
| 10 | Yes | C, $F_1$, $Tx_5$ (25, 37, 186) | 59/60 | 56/22876 | 98.3% | 0.24% |

## 3.2 Results obtained using the reduced class as part of the training set

### 3.2.1 Results obtained reducing the larger size class

In order to reduce the larger size class, in this case the non-flaws class, the steps presented in Section 2.3 were followed. The correlation curve shown in Figure 3 was created with the data obtained after calculation of the correlation between cases. The ordinate axis represents the correlation index *r* and the abscissa represents the number of cases *n* with a correlation greater than *r*. The vertical lines on the graph represent the first eight points (*n, r*) calculated the values of which are tabulated in Table 2.

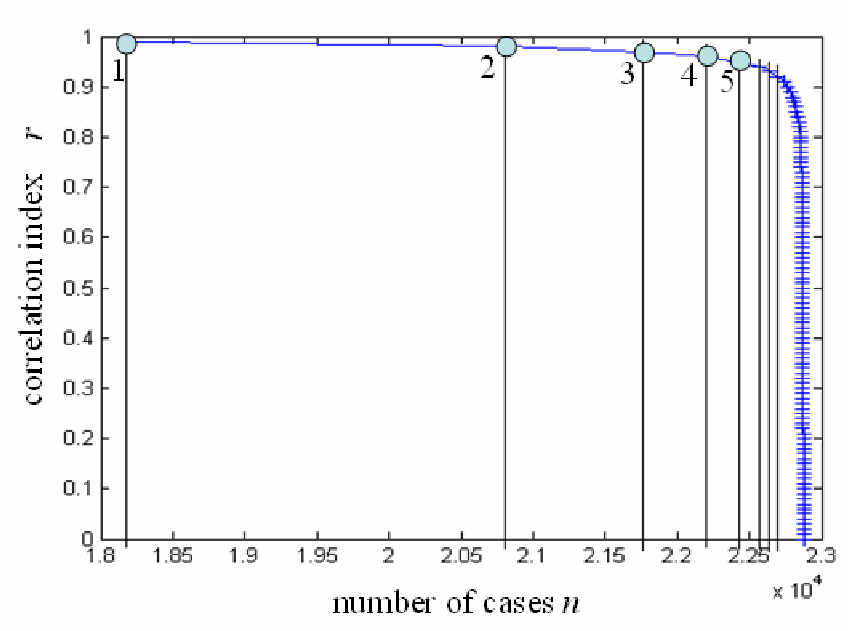

Figure 3. Correlation curve for non-flaws.

Table 2. Percentage of cases which could be eliminated from the training set

| **Points of Figure 3** | ***r*** | *N* | % cases to be eliminated (*n*/total_cases) | % cases in the reduced class |
|---|---|---|---|---|
| **1** | **0.99** | 18184 | 79.5% | 20.5% |
| **2** | **0.98** | 20806 | 91.0% | 9.0% |
| **3** | **0.97** | 21763 | 95.1% | 4.9% |
| **4** | **0.96** | 22198 | 97.0% | 3.0% |
| **5** | **0.95** | 22426 | 98.0% | 2.0% |
| **6** | **0.94** | 22569 | 98.7% | 1.3% |
| **7** | **0.93** | 22642 | 99.0% | 1.0% |
| **8** | **0.92** | 22700 | 99.2% | 0.8% |

Thus, in order to determine the cut-off point, the percentage of cases that would be eliminated was calculated for each point on the curve. Table 2 shows the results evaluated for the first points on the curve. As can be appreciated, the number of correlated cases increases as the correlation index decreases.

After the sixth point on the curve, for every drop of 0.01 correlation points, (1% of the total information), the number of cases eliminated is very few. At this sixth point, a significant reduction in cases is achieved while maintaining 94% of the information of all the cases in the original class in the remaining subset. Consequently, this point (22569, 0.94) was selected for effecting the reduction which means that 22569 cases have a correlation greater than 0.94. The complement of these cases, 307 non-flaws, are incorporated into the training set.

In this way a training set with 60 flaws and 307 non-flaws was created. A Multi Layer Perceptron network with a hidden layer with 4 neurons and a logarithmic sigmoid activation function was trained. The test data used comprised all the data available, including those used in the training, in other words these data included 60 flaws and 22876 non-flaws.

The training included 1000 iterations. The results are shown in Table 1, Experiment 2. As can be seen the results show a 100% recognition of flaws, and a 99.99% recognition of non-flaws. This shows that the reduced set stored sufficient information for recognising the majority of no-flaw cases without interfering with the recognition of flaws.

### 3.2.2 Results obtained using as a training set the reduced class with features reduced by means of a sensitivity analysis

In order to reduce pre-processing and segmenting times in the feature extraction, and thus be able to work with a smaller set of features, some tests were carried out and a few of these were selected using a sensitivity analysis (Principe, 2000), to determine the set of features that best defined the classes. Two of these gave the best results.

The features that define both classes well are feature 360 and feature 376 which correspond to components (3,3) and (5,3) of the transformed DCT ($Y_{DCT}$), respectively. The training set was then made up of 60 flaws and 307 non-flaws. The test was performed on the complete data set (like wise considering only those columns that corresponded to the features above).

The results are shown in Table 1, experiment 3. The network was able to recognise 100% of the flaws and 99.6% of the non-flaws, using the reduced class and all the data from the other class as a training set, while using only 2 of the 28 features.

### 3.3 Results obtained using as part of the training set the reduced class and selected features

Other tests were carried out with other data sets, considering features that can be computationally extracted easily and quickly in addition to being easy to interpret. Of the 28 features 10 were selected: features 25, 30, 31, 33, 37, 59, 100, 128, 179 and 186. These were combined among each other to form groups of three features. Table 1, experiments 4 to 9 show the best results obtained with reference to sensitivity and (1-specificity) of all the combinations tried.

As Table 1 shows, the results were very good. By reducing the larger class and selecting only three features the network is able to classify both classes with a 100% accuracy (experiment 4), thus surpassing the results obtained with the sensitivity analysis.

## 4. Conclusions

In flaw detection problems, there are many cases of unbalanced classes in which non-linear methods such as neuronal networks do not work if traditional methodologies are used in their application, principally because the smaller class is not adequately weighted.

In the present work we present the hypothesis that if there are sufficient cases of the smaller class, it is possible to reduce the larger class by carrying out a correlation analysis between cases, thus creating a more balanced training set and in this way achieving a neuronal model that can adequately classify both classes.

With this method, the non-flaw set was reduced from 22876 cases to just 307 cases that represented the larger size class. The network was trained with 307 non-flaws, 60 flaws, and 28 features. It was then tested on all the data resulting in 100% recognition of flaws, and over 99% of the non-flaws. Additionally good results were also achieved by selecting some of the features (see Table 1).

The best results for this work (Table 1, experiment 4) compared with Mery *et al.*, 2003, can be found in Table 3. Upon comparison of these results it can be seen that those of the present work surpass those obtained by Mery *et al.*, 2003, which used traditional classifiers and neuronal networks, the second case replicating the smaller size set.

Thus, it can be seen that the reduction of the larger size class is possible, and that excellent results are obtained in classification, even when training with two or three features. This work may be replicable in other cases with unbalanced classes.

Table 3. Comparison of performances

| **Method** | **Classifier** | **Selected features** | *TP* | *FP* | $S_n$ | $1\text{-}S_p$ |
|---|---|---|---|---|---|---|
| Mery *et al.* 2003 | Threshold | $K_{\sigma}$, $F_1$ (30, 37) | 57/60 | 230/22876 | 95% | 1.00% |
| Mery *et al.* 2003 | Neuronal Networks | $K_{\sigma}$, $F_1$ (30, 37) | 60/60 | 558/22876 | 100% | 2.44% |
| Proposed Method | Neuronal Networks, applying a reduction to the larger size class | $\Delta'_Q$, $F_1$, $Tx_5$ (33, 37, 186) | 60/60 | 0/22876 | 100% | 0.00% |

## Acknowledgment

This work was supported by FONDECYT -- Chile under grant no. 1040208.

# References

Bishop Christopher, 1995, “Neural Networks for Pattern Recognition”, Oxford University Press Inc., New York.

Chan, P. K., Fan W., Prodromidis, A. L., & Stolfo, S. J. , 1999. “Distributed data mining in credit card fraud detection”. IEEE Intelligent Systems, 14(6):67-74.

Chih-Ping W., I-Tang C., 2002, “Turning telecommunications call details to churn prediction: a data mining approach”, Expert Systems with Applications, Vol 23, Nº 2, pp. 103-112.

Funahashi, K.-I, 1989. "On the Approximate Realization of Continuous Mappings by Neural Networks," *Neural Networks*, 2(3), 183-192.

Haykin Simon, 1994, “Neural Networks A Comprehensive Foundation”, Macmillan College Publishing, Inc., USA.

Mery, D.,2003, “Crossing line profile: a new approach to detecting defects in aluminium castings”. Proceedings of the Scandinavian Conference on Image Analysis 2003 (SCIA 2003), J. Bigun and T. Gustavsson (Eds.), Lecture Notes in Computer Science LNCS 2749: 725-732.

Mery, D.; da Silva, R.; Caloba, L.P.; Rebello, J.M.A., 2003, “Pattern Recognition in the Automatic Inspection of Aluminium Castings”. Insight, 45(7):475-483.

Mitchell Tom, 1997, “Machine Learning”, McGraw-Hill, USA.

Principe José C., Euliano Neil R., Lefebvere W. Curt, 2000, “Neural Analysis Adaptive Systems”, John Wiley & Sons, Inc., USA.

Schalkoff Robert, 1992, “Pattern Recognition: Statistical, Structural and Neural Approaches”, John Wiley & Sons, Inc., USA.

Tou J. T.; González R. C., 1974, “Pattern Recognition Principles”, Addison-Wesley Publishing Company, Inc., USA.