23

Neural Computation Methods in the Determination of Fungicides

Carmen Paz Suárez Araujo[1], Patricio García Báez[2]
and Yaridé Hernández Trujillo[1]
[1]*Universidad de Las Palmas de Gran Canaria,*
[2]*Universidad de La Laguna*
Spain

1. Introduction

Fungicides are a specific type of pesticides that control diseases produced by fungi, inhibiting specifically or directly killing these parasite organisms. It has been used during more than forty years for the protection of harvests and farming lands.

There are several treatments with fungicides: protector or preventative from the germination of the spores and follow up infection, or curing or eradicating, when mycelium has been formed and must be controlled. Treatments are applied in the soil and on stored vegetable products, seeds or plants. In this case two types of fungicides are classified: a) those with contact: unable to penetrate the inside of the vegetable and control epiphytic fungi, and b) those that are systemic and control endophyte fungi.

The large variety of fungicides that exist makes a classification absolutely necessary. We find different taxonomies that can be studied not only for their structural aspects, but such as chemical composition, and also in action modes. The guidelines that regulate the managed use and classification of these are defined in Spanish, European and American Environmental Agencies.

The use of chemical products in farming activities has produced important benefits in agricultural production, increasing profitability of harvests while simultaneously raising the quality levels of the food products. Nevertheless, there are other considerations with regard to these benefits that result in the systematic destruction of parasites, that affect the health of the plants, animals and human health, and require consideration of the interaction of the different chemical main components with animal species and with humans themselves (Rivas, 2004). To begin with, the form of the administración of fungicides favors their accumulation in the sediments and in drainage waters. Also, in cases where plaguicides have been used indiscriminately, the species of plagues have become resistent and difficult to control.

The main source of exposition of the general population to fungicides is through food, a fact that has forced the establishment of regulations of its maximum daily ingestion allowance. There are studies that relate the exposure to pesticides with damaging effects on human health: neurological damage, hormonal and reproductive disorders, dermatological or carcinogenic reactions (Alavanja, 2004).

Thus, in European countries the residues from plaguicides in commercial foods are subject to international standards and are strictly monitored. As a result, it is necessary to develop technologies that allow the treatment of residues and polluted environments. However the development of new methods with precise and sensitive detection are also essential, especially those that detect, quantify and control those damaging compounds for the ecosystem and as a consequence for human health.

Some of the most commonly used instrumental techniques for the detection of these compounds is found in thin layer chromatography (TLC), gas chromatography (GC) (Vassilakis, 1998) high performance liquid chromatography (HPLC) (Halko, 2006), gas chromatography - mass spectrometry (GC-MS) (Martínez Vidal, 2000) and high performance liquid chromatography – mass spectrometry (HPLC-MS) (Taguchi, 1998). The characteristics and properties of these substances make use of specified techniques more difficult, or perhaps even necessary to use preprocessing and/or a combination of techniques. Due to the unstable character of some of the pesticides it is not easy to perform GC analysis. In this case HPLC is an appropriate technique for its determination. A similar and very common step is to carry out some type of derivatization of the compounds, such as in the case where they are transformed into fluorescent compounds (Vassilakis, 1998). These pre-treatments require a large consumption of dissolvents and reactants in addition to the cost, the time and the analytical complexity of the cited techniques (Chiron, 1995).

Other developed chemical methods are spectrophotometric methods (Sastry, 1987), however one of its main drawbacks is the degree of overlap in the spectra, which complicates the analysis of complex mixtures of these compounds. The cynetic methodology based on the difference of reaction velocities is another effective way to analyze several analytes at the same time and has improved with the use of chemometric procedures. Some of the most commonly used are those obtained from the application of multivariate methods (PLS, PCR, CLS) and the analysis of variance ANOVA(Galeano-Diaz, 1997).

Chemometric methods are an important alternative to the purely instrumental ones (Olivieri, 2008), finding a high application of the same in the determination, not only of fungicides but also of pesticides in general, applying one or a combination with the instrumental ones or the neural ones. Among these we point out the MCR-ALS method (multivariate curve resolution-alternating- least-squares) which has been applied to the study of the contamination of sediments and waters and the measurement of air quality (Salau, 1997) and parallel factor analysis or PARAFAC (Bro, 1997). Other methods needing mention are those based on partial least squares approach (PLS) in the setting of environmental analysis (Piccirilli, 2006). Principal component analysis or PCA is a multivariate statistical method commonly used in data analysis (Jolliffe, 1986) but with some limitations and attempts to solve models based on these same methods such as UNMIX (Henry, 2003) and the PMF (positive matrix factorization) (Paatero, 1994). Both have been adopted by the United States Environmental Protection Agency (U.S. EPA) because they are robust methods to measure air quality. Alternative methods have also been developed based on tri-linear and multi-linear models such as TUCKER-3 (Tucker, 1966) and the ME (Paatero, 1999). Shao-Hua Zhu (Zhu, 2007) use fluorescence detection combined with chemometrics for the detection of Carbendazim in banana samples. They specifically use the excitation-emission matrix (EEM) fluorescence spectrum and three second order calibration methods: PARAFAC, SWATLD and APTLD, obtaining satisfactory results in the quantification of the fungicides in complex samples.

QSAR (quantitative structureactivity relationships) data analysis (quantitative structure activity relationships) describes the mathematical relationship between the structural properties of a series of chemical compounds and their possible toxicological activities, hence it is very useful in the determination of toxicity of synthetic fungicides.

A complementary approach to the cited methods, with high capacity and efficiency, which tackles the identification of fungicides, and pesticides in general, in complex samples is one based on a different computational approach to symbolic computation, as is the case with neural computation. The basic processing element of this computational approach is found in artificial neural networks (ANNs). ANNs can be defined as a cognitive information processing structure (massively parallel dynamical system) based upon models of the brain function. They are composed of highly interconnected computational elements with graph topology. Its most appealing property is its learning capability. Its behaviour emerges from structural changes driven by local learning rules, with generalization capability (Suárez Araujo, 1999)(García Báez, 2010).

The suitability of ANNs has been extensively demonstrated for its use in a wide variety of applications where real-time data analysis and information extraction are required in different settings related to pesticides in general and fungicides specifically, such as chemistry, environmental and agriculture fields (Yang, 1997). Applications in cited papers include those of Suárez Araujo and García Báez (Suárez Araujo, 2006)(Suárez Araujo, 2009)(García Báez, 2010) in which they identify fungicides in mixtures of up to three and four different fungicides using ANNs and neural network ensembles. Research by Guiberteau and collaborators have solved ternary mixtures of pesticides with similar molecular structure (Guiberteau, 2001), Baoxin Li (Li, 2006) identify three organophosphorated pesticides, Istamboulie solves insecticides (Georges, 2009), and Wan Chuanhao (Wan, 2000), who proposes the use of neural computation to identify pesticides from the family of carbamates, or the developments by Yongnian Ni and collaborators (Ni, 2004), that describe the RBF-ANN method and the PC-RBF-ANN method as the best at detecting mixtures of three pesticides of this same family. On the other hand, work by Karl-Heinz Ott (Karl-Heinz, 2003) shows an example of neural networks that determine the action mode for a large number of herbicides.

In this chapter, a study of neural computation has been presented where theory and architecture of paradigmatic supervised and unsupervised ANNs, and a new hierarchical unsupervised modular adaptive neural network (HUMANN) is reported. The neural network ensemble approach is introduced. Finally, we will discuss the ability of ANNs and a neural network ensemble to address this issue, describing the outcomes of implementations of such approaches for the determination of fungicides in different kind of samples.

2. Neural computation

Researchers from different and varied fields have studied the neural processing and control of biological systems and have attempted to develop synthetic systems, that is artificial systems, which possess similar capacities (Suárez Araujo, 1996), as well as formulating theories regarding how computation in biological systems really occurs. The first ANN model is attributed to Warren McCulloch and Walter Pitts and had universal computational capacities, a formal neural network, based on knowledge that was available on nervous system functions (McCulloch, 1943). Since then there have been a series of advances that have placed neural computation as a key element in the solution of a great variety of

problems in a wide domain of applications and as a capable computational paradigm for the advanced knowledge in the function and structure of the brain as well as its computation style.

Neural computation can be understood as parallel, distributed, and adaptive computing carried out by three dimensional modular architectures organized in layers and made up of a large number of processing elements, with a high degree of connectivity, with a learning capacity that may be supervised or not (Suárez Araujo, 1996). Its main structure of information processing is artificial neural network, where the adjective neural suggests that they are inspired in biological neural networks. ANNs study and use some strategies from the methods used by biological neurons to process information. Neural computation is just one of many different computational models that have been defined throughout history, where ANNs are considered as another approach to computational problems (García Báez, 2005).

2.1 Artificial Neural Networks

ANNs can be defined as large, parallel and interconnected networks that are made up of simple elements (generally adaptative) and with a hierarchical organization that attempt to interact with the objects of the real world by imitating the biological nervous system (Kohonen, 1989). ANNs found their inspiration in biological neural networks. Thus, there is a biological feel for the models and systems based on ANNs or neural computation. Nevertheless, the task of generating an ANN comparable to the biological neural network is not easy at all, since the human brain has on order 10^{11} neurons and each one of these receives an average of 10^3 to 10^4 connections, all of which are integrated to obtain a unique output with a complex structure of connections, not completely determined, providing a high and complex processing capacity. If we compare biological neurons to the logical gates in silicon, they are smaller but need a longer time to generate output, even though there is a lack of slantedness by the high number of neurons that exist as well as the large capacity of interconnectivity and work in parallel that these possess. From a computational perspective there is much research to do before obtaining a solid method such as the biological one to implement our neural models. Given that the human brain is much more complex in order of magnitude than any existing ANN, it is impossible to approach this process capacity, even with today´s technology.

The general framework of an ANN includes eight components (Rumelhart, 1986): a *set of processing elements* (*units* or *neurons*), each one self-contained, with local memory; *activation state* of each processing element; *output function* for each processing element; *connectivity pattern* between the processing units, where each connection has an associated *synaptic weight*; *propagation rule* or *network function*, to propagate the activity patterns based on the connections network; *activation rule* or *activation function*, to combine the inputs that arrive to a unit with the actual state of the unit and can produce a new activation level in it; *learning rule*, through which it can modify the connectivity patterns based on experience and *representation of the environment*, there are two information settings, one local and another global.

Following this framework we characterize ANNs in the three following levels: *connectivity topology* (covering the neural structure), *neurodynamics* and *learning*.

Connectivity topology is an essential part of the neural structure of an ANN and indicates the shape in which the different processing elements of a network are interconnected amongst themselves (Hecht Neilsen, 1990). The structural organization of the processing

elements that make up the ANN normally uses layers, taking into account the individual units that form it with similar characteristics. There are ANNs with a flat neural structure, with only one layer of processing elements (the input layer is excluded), also called *single-layer* ANNs. In general form, ANNs can have different degrees of depth, which is given by the number of layers that make it up and are referred to as *multi-layer* networks. In this last case we can identify: the *input layer*, where the elements that are not usually considered as making up a layer, since they do not carry out any of their processing, but instead simply distribute the input information to the rest of the processing elements with which they connect; the *output layer*, which represents the units that provide the output of the network and the *hidden layers*, all of those that generate the connectivity of the network between the input layer and the output layer without direct contact with the environment.

The connectivity density of ANNs as well as its direction, where direction is understood to be the information flow direction between two interconnected processing units, namely inter-layers and intra-layers, create a structural taxonomy of the ANNs (Simpson, 1990). Hence we can find networks with total connectivity, where all of the units of the network are connected with each other, *full-connected* networks and the dispersed or partial connectivity, where each neuron is connected to only one set of neurons of the network, called *partial-connected*. This full or partial connectivity can also refer to the relations that are established between the connections of the two layers. One particular case of this last category is *one-to-one* connectivity. With respect to the direction of connections between the layers we have *feed-forward* connected networks, where the connections go from the input to the output, *recurrent* networks, where connections can exist that go from earlier layers (*backwards*) or other networks where the connections tie neighboring units from their own layer, which are called *lateral connections*.

Neurodynamics cover of the local information processing that the units carry out. This process is given by the local computation model of the neuron, Fig. 1. Mathematically this is expressed with the so-called network function *net* that integrates the inputs **x** and the corresponding synaptic weights **w** (normally by means of a weighted sum, equation (1), when working with a linear computation model, McCulloch-Pitts model (McCulloch, 1943), although non-linear models also exist (Suárez Araujo, 1997)).

$$net_i(\mathbf{x}) = \sum_j w_{ij} x_j \tag{1}$$

The activation function f_{act} (normally a non-linear function of any type), acting on the function value of the network which allows the dynamics of the activation states of the units and finally an output function f_{out} to be obtained, that are applied to the activation state of the unit provides the output values y_i of the units. It is also possible to store local values inside the processing units, θ_i, that can be used in the calculation of the outputs, which is known as local memory:

$$y_i = f_{out}\left(f_{act}\left(net_i(\mathbf{x}) - \theta_i\right)\right) \tag{2}$$

Finally, an essential and differentiating part of ANNs is learning. We can define learning as the capacity of a system of absorbing information from the environment, without a need for the system to be programmed externally. Learning in ANNs follows a compound model using two stages: *load mode* (learning) and *retrieving mode* (execution). Load mode is where

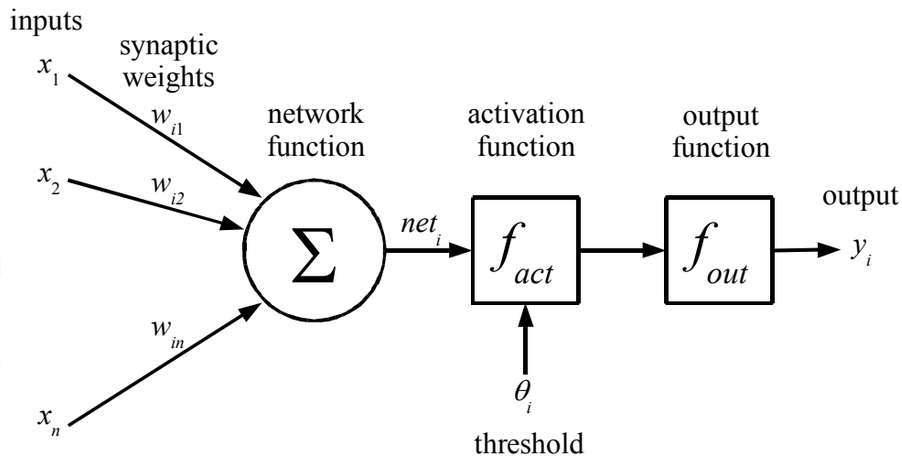


Fig. 1. General Building-Block of an Artificial Neuron

the learning takes place. Starting with the data that are received in the setting the ANNs are capable of processing them and storing the extracted information from them, in their synaptic connections. On the other hand, the retrieving mode allows responses to be obtained based on established questions from the setting, by means of adequate processing using the knowledge that has been acquired by the ANNs.

The learning processes produce changes in the network in order to try and achieve a new way to respond more efficiently to the specifies task. These changes can be gathered by modifications in the connectivity topology and/or by modifications in the synaptic weights:

$$\tau \frac{d\mathbf{w}}{dt} = -\frac{\partial}{\partial \mathbf{w}} R\left(\mathbf{x}(t), \mathbf{d}(t), \mathbf{w}(t)\right), \tag{3}$$

where the function R is called the *instantaneous learning potential* and **d** can or not be present, depend on kind of learning, and is defined as the *teacher signal*, and the synaptic weight vector **w** changes in the direction of decreasing R (Amari, 1990). We can reformulate and simplify the equation (3) for discrete case as:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t). \tag{4}$$

In general terms, the ANNs carry out the learning process by itself based on a set of training data sets, which is called learning based on examples, in these cases learning algorithms are used, which are iteratively ordered to update the weighted values and/or necessary topological changes. As opposed to neurodynamics, the performance of non-local processes in the definition of the learning algorithms is allowed. There is also a dependent designer modality, which can establish the needed changes by means of an appropriate formulation to solve the problem.

With regard to the type of learning that can be computed we identify two types of different learning experiences (Rumelhart, 1986). *Associative learning*, where the experience learns to produce a pattern of specific activation in a set of units when another specific pattern occurs in another set of units. It allows, consequently, to map arbitrary activation patterns in the input, into other activation patterns into another set of units, normally output, and *Feature detection*, where the processing units learn to answer when faced with *interesting patterns* in

their input. It is the basis for the development of characteristic detectors and consequently the basis for knowledge representation in ANNs.

Based on the guided learning process we can identify three different paradigms (Haykin, 1994):

- *Supervised learning*: for each input pattern to the network there is a signal (correct response for it) which serves as a guide for the adjustments in the synaptic weights
- *Reinforcement learning*: with the response of the network given an input it facilitates a scalar evaluation of the response. It indicates whether it is correct or incorrect.
- *Unsupervised learning (self-organized learning)*: The network does not receive any tutoring, but has to organize its output based on the redundancy and structures that can be detected in the input.

The entire learning process, independent of its type and according to the taxonomy carried out entails a way in which the synaptic weights are modified and updated. This way consists in what we call *learning rules*, which are mathematically expressed, according to the type of systems (continuous or discrete), and by means of differential equations or difference equations. When dealing with the different types of rules we can speak of another classification for learning processes, where we identify four different types (Haykin, 1994)(Jain, 1996):

- *Hebbian*: It can be considered as a learning support, biological as well as computational, by constructing the basis of many follow up learning rules, essentially unsupervised ones. It is included in the coincidence learning category. In neurocomputation the Hebbian rule has its origin in a mathematical point of view the affirmations stated by Donald Hebb in 1949 (Hebb, 1949) based on neurobiological observations.
- *Competitive*: this type also has high neurobiological bases since its experiments have shown their use in the formation of topographic maps in the brain and in the orientation of cell nerves that are sensitive from the striate cortex. It is based on a competition process among all units to assign the exclusive representation in the face of a group of input patterns
- *Error correction*: focuses on rules based on error correction attempt to correct the error that is produced in the network when comparing the desired output with the actual output from the network. It is normally applied in the supervised paradigm. Biological plausibility is not so evident as in the earlier cases
- *Energy optimization*: one of the best known of these is the *Bolzmann learning*, and is characterized by its use of an energy function, determined by the states of the individual neurons, which attempts to be optimized

ANNs have many architecture properties and functionalities that make it especially appropriate to tackle highly complex problems based on behaviour, non-linear, etc. in real time, among the most relevant are (Haykin, 1994) *Generalization*, *Treatment of Contextual Information*, *Fault Tolerance*, *Evidential Response* and *Uniformity of Analysis and Design*. In general, the applications which have been most useful to ANNs are characterized by their capacity to handle the following tasks (Haykin, 1994)(Jain, 1996): *Pattern Classification*, *Function Approximation*, *Associative Memory*, *Prediction*, *Control* and *Optimization*.

Finally we must consider the capacities of the ANNs saturated in its structural organization. Two types of structural organization are generally considered: *monolithic* and *modular.* Monolithic organization, where ANNs are frequently considered as *black-boxes,* is often used, and not only with the level of construction of models such as an application level. This structural organization begins to present serious difficulties as the neural networks grow, or

when the applications become more complex. In specific cases, the complex problems, the efficiency usually decays until a point where it is impossible to solve using monolithic systems. The term *modular neural network* is used to identify many different types of neural structures, in general it can even be said that any network that is not monolithic is normally considered to be a modular one (Rojas, 1996). Likewise, in the literature we can find other terms that form part, include, or are related to modular neural networks, such as *multiple neural networks, neural ensembles, mixture of experts*, *hybrid systems* or *multi-sensorial fusion* (Gallinari, 1995)(Haykin, 1994).

The main idea underlying neural modular networks is the possibility of solving complex problems in a simpler, more flexible, and faster way. This objective is achieved based on the construction of parts or modules, that would probably be neural networks, and at the same time are simpler and smaller that an equivalent monolithic network. This restructuring allows the task learning to be carried out by each module, normally in a simpler way than the global task and the different training that can be carried out in an independent and parallel way (García Báez, 2005).

2.2 Neural architectures

2.2.1 Simple perceptron

The perceptron model (Rosenblatt, 1961) reflects the beginnings of machine pattern recognition. It is a feed-forward single-layer architecture with an input layer, which are made up of setting sensors, and an output layer, with responses sent by the network. Hidden layers are not needed. Its neurodynamics are made up of a network function that uses the classical weight summation for inputs, equation (1), and a step, bipolar or binary, equation (5), for the activation function, with a transition point which is determined by a threshold value θ_i stored locally in the neuron.

$$y_i = \begin{cases} 0 & \text{if } net_i < \theta_i \\ 1 & \text{if } net_i \geq \theta_i \end{cases} \tag{5}$$

The learning model that it follows is capable of adapting its weights and thresholds by means of a supervised paradigm using the so-called *perceptron rule*, based on the correction of the produced error in the output layer:

$$\Delta w_{ij} = a\left(x_i - d_i\right)x_j \tag{6}$$

where α is the learning rate and d_i is the desired output.

One of the advantages of this model is that it uses the *Perceptron Convergence Theorem* which guarantees the learning convergence in finite time and that the architecture always allows the solution to be represented. Precisely it is in this capacity of representation where the greatest limitation of the model resides, for instance in (Minsky, 1969) they discuss its inability to solve non linear separable problems, for example in the case of the exclusive-or problem (XOR).

2.2.2 Backpropagation

One way to overcome the representation limitations mentioned in the Simple Perceptron is through the use of Multi-layer Perceptrons (MLP). A MLP constitutes a topology with one or several hidden layers, and feed-forward connections among its successive layers, either in a

total or partial way. In order to represent any boolean function it is necessary that some of the neurons use non linear activation functions (thresholding function or sigmoid function, equation (7)), maintaining the rest of the neurodynamics the same as in the Simple Perceptron.

$$f_{act}\left(net,\theta\right)=\left(1+e^{-net+\theta}\right)^{-1} \tag{7}$$

The most popular algorithm for training MLPs is *backpropagation* (Werbos, 1974). It is based on a supervised correction of the squared error generated in the output layer using a gradient descent method. This method forces the used activation functions to be differentiable and monotonic. It begins with the output layer and adjusts the weights of the connections that are affected there, producing a backpropagation of errors of the previous layer to occur which successively corrects the weights until reaching the first hidden layer.

$$\Delta w_{ij}=a\delta_i x_j \tag{8}$$

$$\delta_i=\begin{cases} f'_{act}\left(net_i,\theta_i\right)\left(d_i-x_i\right) & \text{if } i\in Output \\ f'_{act}\left(net_i,\theta_i\right)\sum_k \delta_k w_{ki} & \text{otherwise} \end{cases} \tag{9}$$

Some of the most noteworthy problems are studied in backpropagation, since the descent gradient descent does not insure reaching the global minimum error, as opposed to the Simple Perceptron (Minsky, 1969). Many variations of backpropagation have been proposed to over this obstacle, such as generalization, learning speed and fault tolerance.

2.2.3 Simple competitive learning

Together with Hebbian learning, competitive learning is one of the main approaches of unsupervised learning. It also makes up the basis for most neural systems with unsupervised pattern recognition. It uses one type of neuron called the *winner-take-all* or *grandmother cells* (Hertz, 1991) that compete against each other to act as the triggers. Consequently, the idea is to cluster or categorize the input data, that is, similar input will be classified as belonging to the same category and will trigger the same neuron.

Its single-layer architecture presents an input layer that has a full-connectivity with the output layer by means *excitatory connections* (weights greater than or equal to zero). The output layer simultaneously presents *inhibitory lateral connections* (weights less than or equal to zero) among neural neighbors as well as excitatory self-connections. Said connections are those that facilitate the competitive process while searching for the winning neuron from maximum activation.

Their neurodynamics in practice are usually simplified by carrying out the weighted sum of the inputs and the neuron with the largest value in it, considering it the winning unit, sending it in one of its output while the remaining units would send a zero, equation (10). It is also possible to substitute the weighted sum by euclidean distance, in such a case the winning unit will be that which is closest (higher similarity) to the input vector.

$$y_i=\begin{cases} 1 & \text{if } i=\arg\max_k\left(net_k\right) \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

There are several alternatives with respect to the learning rule to use (Hertz, 1991). One of these is the so-called *standard competitive learning rule*, which tries to move the incident weights in the winning neuron directly towards the input pattern.

$$\Delta w_{ij} = \begin{cases} \alpha\left(x_i - w_{ij}\right) & \text{if } i = \arg\max_k\left(net_k\right) \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

This model presents some problems which are difficult to resolve, amongst others we point out the selection of the appropriate number of neurons in the output layer, the potential presence of *dead units* that never end up being winners, the sensitivity with regard to the initial values that are assigned to the weights and the possible need to normalize inputs and/or weights.

2.2.4 Radial Basis Function Network

Radial Basis Function Network (RBFN) architecture can be considered from the point of view of function approximation as a system capable of obtaining the appropriate parameters for a linear combination of basic localized functions that are adjusted to a specific desired function (Musavi, 1992). It uses a modular architecture made up of a two layer set with feed-forward connections between its successive layers, where the first of these is totally connected with the inputs.

The hidden layer use radial basis functions, usually Gaussian kernels, as an activation function, equation (13), hence its name. The weights of the incident connections in each neuron of the hidden layer or *prototypes* makes up a point in the input space of the network. The network function in this layer is to obtain the distance between the input and each prototype, equation (12). Consequently the hidden neurons will be activated in the next inputs to its prototype, thus generating hyperspheres in the input space, which is different from the hyperplanes generated by other activation functions. The output layer makes up a Simple Perceptron or similar linear network that implements a linear combination of these radial basis functions.

$$net_i\left(\mathbf{x}\right) = \left(\sum_j \left(x_j - w_{ij}\right)^2\right)^{1/2} \tag{12}$$

$$f_{act}\left(net, \sigma\right) = e^{\frac{-net^2}{2\sigma^2}} \tag{13}$$

It follows a hybrid learning, where the hidden layer is adjusted with Simple Competitive Learning and the output layer by means of the perceptron rule or the backpropagation rule.

Among the advantages of this model is the faster convergence than the MLP with backpropagation. On the other hand it usually needs a greater number of hidden neurons than the others, which produces a slower retrieving mode.

2.2.5 Kohonen's Self-Organizing Maps

Self-Organizing Maps (SOMs) (Kohonen, 1989)(Kohonen, 1997) describe the idea that topographic maps, such as those that exist in the cortex of highly developed animal brains,

extract the features of the input space preserving its topology. On one hand they combine characteristics of competitive systems, quantifying the input space in different regions represented by a specific number of output neurons. On the other hand, they maintain a neighborhood relationship between the units of the output space, that is, two neighboring neurons will represent close regions in the input space. Thus these methods will generate a discrete map, possibly with reduced dimensionality that the input space, and will preserve the existing topology in this one. Among the most commonly used methods we find the Kohonen SOMs, although there is a great diversity in their variations. These exhibit a similar topology than Simple Competitive Learning, but the output layer is organized in an *m*-dimensional space in agreement with the form that we desire for the map, and the most common is a two dimensional matrix.
The neurodynamics that follow the Kohonen SOMs accept the same possibilities as those in Simple Competitive Learning. There is also unsupervised and competitive training paradigm which follows. The main variations are seen in the modification of the synaptic weights, equation (14), which not only affects the winning neuron but also to a lesser degree the set of neurons in the winners neighborhood *N*, and consequently being able to generate topological relations. The neighborhood relationship between nodes is normally given by a hexagonal or squared type lattice, whose size decreases during the training period.

$$\Delta w_{ij} = \begin{cases} \alpha\left(x_i - w_{ij}\right) & \text{if } i \in N\left(\arg\max_k\left(net_k\right)\right) \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

Their uses range from the projection of multivariate data, density approximation to clustering, having been successfully applied in a multitude of fields (Kohonen, 1997). This method acquired the problems previously mentioned of Simple Competitive Learning, added by the fact that a larger number of parameters that can be adjusted are available.

2.2.6 HUMANN

HUMANN (García Báez, 2003) is a modular neural network that can implement the general approach of the classification process, which has three stages: a) feature extraction, b) template generation, c) discrimination (labeling), in a transparent and efficient way. HUMANN uses a multi-layer neural structure with three modules and with different neurodynamics, connectivity topologies and learning rules, Fig. 2.
The first neural module of our HUMANN is a Kohonen's s SOM module. The second module is the Tolerance layer. It is the main module responsible for the robustness of HUMANN against noise. Its topology is a two-dimensional array which has the same dimension as the Kohonen layer and a one-to-one interconnection scheme with that previous layer. The main objective of this layer is to compare the fitting between the input patterns and the Kohonen detectors. If the goodness of the fit is not sufficient the pattern is regarded as an outlier and is discarded. The weights of this layer are responsible for storing the mean ($\mathbf{w}^0$) and standard deviation ($\mathbf{w}^1$) of the fits between the inputs and the Kohonen detectors when this neuron is the winner. This is a new concept called the *Tolerance margin*, equation (15). The goodness of the representation of a pattern by a detector will be a function of the ratio of the scalar product or euclidean distance between both of them and the Tolerance margin of the detector. The needed learning rule to obtain the weights of the global variance in the degree of the pairing, is based on a differential equation that

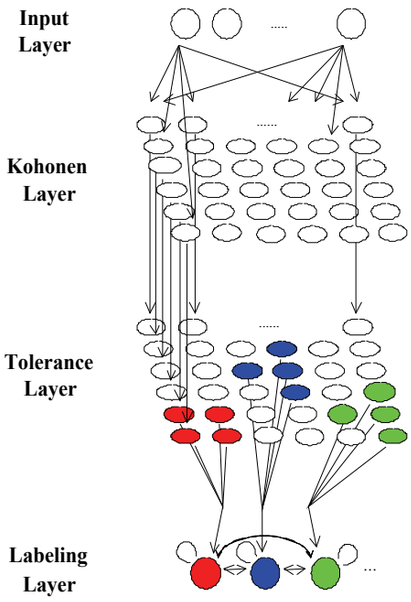


Fig. 2. HUMANN architecture

converges towards said average, to which must be added a decay term to make the finalt inputs to the system more relevant, in addition to avoiding possible pernicious effects of artifacts or outliers patterns, equations (16) and (17).

The labeling module (García Báez, 2003) implements the discrimination task. It maps the outputs of a neural assembly belongs to the Tolerance layer which have been activated by a category, into different clusters represented by labeling units. This module exhibits a full-connection topology and a dynamic dimension, which is fitted to the number of clusters detected in the data set. In addition it presents a group of lateral connection that connect each unit with all of the units in this layer, including itself, to this type of connection and is called *silent synapses*, since they do not directly influence the output of this layer, as opposed to the rest of the connections that we will call active connections.

$$y_i = \begin{cases} 0 & \text{if } x_i \geq (w_i^0 - \lambda w_i^1) \vee w_i^1 = 0 \\ 1 - \dfrac{(w_i^0 - x_i)}{\lambda w_i^1} & \text{otherwise} \end{cases} \tag{15}$$

$$\Delta w_i^0 = \begin{cases} \alpha\left(x_i - w_i^0\right) - \beta w_i^0 & \text{if } i = \arg\max_k \left(x_k\right) \\ -\beta w_i^0 & \text{otherwise} \end{cases} \tag{16}$$

$$\Delta w_i^1 = \begin{cases} \alpha\left(\left|x_i - w_i^0\right| - w_i^1\right) - \beta w_i^1 & \text{if } i = \arg\max_k \left(x_k\right) \\ -\beta w_i^1 & \text{otherwise} \end{cases} \tag{17}$$

The adaptive character of HUMANN is essentially an embodiment in the labeling module, because of its dynamic dimension (García Báez, 2001)(García Báez, 2003). This characteristic

is implemented by two neuronal mechanisms, a) neuronal elimination, b) neuronal generation. They perform the refinement processes in the neuronal circuits, and they are present in the human brain. Silent synapses are in charge of controlling the dimension of the output layer, which is dynamic and adjusts to the number of detected clusters in the set of training data. In this way, the dimension increases when a new cluster is detected and the cluster can be reduced when the patterns belong to a class that are poorly classified by two or more different units, or in the case when, upon detecting units that are practically not activated by any pattern. Thus, we can consider this system as an *ontogenic network* (Fritzke, 1997), able to adapt its dimension to the characteristics of its environment. The neurodynamics of this layer consist in an activation function that inhibits the output in the case where this value does not exceed a specific trigger threshold:

$$y_i = \begin{cases} net_i & \text{if } net_i \geq \rho \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

In the case where no one exceeds the threshold value, all of the units will have an output value of 0, which indicates that the pattern could not be classified. If this type of event is occurs in the learning stage it will trigger the generation process of a new unit, which has its biological counterpart in the *neurogenesis process* (Alvarez Buylla, 1990), with the following increase in the output layer dimension of this module. This would indicate that an input pattern has arrived that has not been considered an outlier and no unit in the labeled (cluster) layer actually exists that it represents. The general training rule used for the adjustment of the active weights $\mathbf{w}^{act}$ would try to achieve that the winning neuron obtained as an output would have a value close to ,1, for this the weights are updated that lead to the winning unit, leaving the remaining ones unmodified, that is, a competitive learning, equation (19). The weights of the silent connections $\mathbf{w}^{sil}$ are symmetric and try to represent the existing time correlation between both units when one of them is the winner. In this way, if one wins against another it is normally activated in a simultaneous way, the weight of its connection is kept at a high level, while if the opposite occurs the connection will have values approximately equal to zero, that is, a Hebbian learning, equation (20).

$$\Delta w_{ij}^{act} = \begin{cases} \alpha\left(1 - net_i\right)x_j & \text{if } i = \arg\max_k\left(net_k\right) \wedge \left(net_i \geq \rho\right) \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

$$\Delta w_{ij}^{sil} = \Delta w_{ji}^{sil} = \begin{cases} \alpha\, net_i\left(net_j - w_{ji}^{sil}\right) & \text{if}\left(i = \arg\max_k\left(net_k\right) \wedge \left(net_i \geq \rho\right)\right) \vee j = i \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

If a group of two or more units from the labeling layer are connected with weighted values that exceed a threshold value, indicate that normally these units are being treated as activated before the same patterns and indicates which units are represented *coincidental clusters*. This event will trigger a process of neural reorganization that will reduce the dimension of this layer. This way, each set of units of coincidental clusters be transformed into a unique unit, producing a *synaptic reorganization*.

HUMANN is available in variants in order to process data with outlier noise and generate not exclusive clusterings, where the input can belong to several clusters at the same time. It

is scalable, hence it is adequate to use with large data sets, allowing clusters with arbitrary shapes and those with high tolerances to be found.

3. Neural Network Ensembles

The ANNs generate relationships between the inputs and outputs from a training set. It will be able to have several network configurations close to the optimal one, according to the initial conditions of the network and the ones typical of the environment. These configurations correspond to different ways of forming generalizations about the patterns inherent in the training set. As each network makes generalization errors on different subsets of the input space, it is possible to argue that the collective decision produced by the complete set, or a *screened* subset, of networks, with an appropriate collective decision strategy, is less likely to be in error than the decision made by any of the individual networks (Hansen, 1990). This has given rise to the use of groups of neural networks, in a trial to improve the accuracy and the generalization skills of them. In analogy with physical theory, it has been referred to the set of neural networks used as an *ensemble*, *neural network ensemble* (NNE) (Hansen, 1990).

Being able to tackle complex tasks in an efficient way has been another proof feature of the NNE approach. A NNE combines a set of neural networks which learn to subdivide the task and thereby solve it more efficiently and elegantly. In a sense, the NNE follows a divide-and-conquer approach by dividing the data space into smaller and easier-to learn partitions, where each ANN learns only one of the simpler partitions. The underlying complex decision boundary can then be approximated by an appropriate combination of different ANNs. NNEs are also very appropriate in applications where large volumes of data must be analyzed. It is necessary partitioning the data into smaller subsets, training different ANNs with different partitions of data, and combining their outputs using an intelligent combination rule. The situation of having too little data can also be handled using ensemble systems. (Polikar, 2007). Data fusion scheme, where several sets of data are obtained from various sources, is another scenario for using NNE. This last case is the typical one concerning with the subject studied in this chapter.

A NNE offers several advantages over a monolithic ANN: It can perform more complex tasks than any of its components. It is more robust than a monolithic neural network. It can produce a reduction of variance and increase in confidence of the decision, and can show graceful performance degradation in situations where only a subset of neural networks in the ensemble are performing correctly (Liu, 2003).

This ensemble based approach has shown to be very useful for a broad range of applications and under a variety of scenarios, (Polikar, 2007) essentially in data analysis and automated decision making applications. The idea of designing ensemble learning systems can be traced back to as early as 1958 (Liu, et al., 2003) and 1979 with Dasarathy and Sheela's paper (Polikar, 2006). Then, and since the early 1990s, algorithms based on similar ideas have been developed, appearing often in the literature under various other names, such as ensemble systems (Polikar, 2007), classifier fusion (Kuncheva, 2001) committees of neural networks (Drucker, 1994), mixture of experts, (Jacobs, 1991),(Jordan, 1994); boosting and bagging methods (Schapire, 1990),(Drucker, 1993) among others.The paradigms of these approaches differ from each other with respect to the procedure used for generating ensemble members, and/or the strategy for combining them. Or even some differences, from statistical point of view, can be found (Opitz, 1996).

Two strategies are needed to build an ensemble system:
a) Strategy for generating the ensemble members. This must seek to improve ensemble´s diversity. The other main stone in order for an ensemble to generalize well is the accuracy. Brown et al. (Brown, 2005) suggest that two ANNs are diverse if they make different errors on the same data points/inputs. Accuracy refers to how good the learning model is in comparison to random guessing on a new input (Brown, 2005). Two common methods to maintain the diversity within an ensemble are bagging and boosting, and its successor *AdaBoost* (Polikar, 2006)(Polikar, 2007). This group of techniques belongs to the sequential training methods of designing NNEs (Liu, 2003). The bagging method will randomly generate a new training set with an uniform distribution for each network member from the original data set (Polikar, 2006). The boosting approach (Schapire, 1990), resamples the data set with a non-uniform distribution for each ensemble member. The whole idea of boosting and bagging is to improve the performance by creating some weak and biased classifiers (Dam, 2008).
There exists some NNE where the training data is inherently resampled by classifier conditions, as the neural-based learning classifier system, NLCS, (Dam, 2008) or HUMANN based ensembles (Suárez Araujo, 2009)(García Báez, 2010). These NNEs implicitly maintain diversity in the population without using bagging or boosting. The stacked generalization (SG) and mixture of experts (ME) use different designing strategy to bagging or boosting techniques as well (Polikar, 2006)(Polikar, 2007). They generate two levels of ensemble members. A first level where individual classifiers are experts in some portion of the feature space. A second level classifier, which is used for assigning weights for the consecutive combiner, which is usually not a classifier, in the ME, and as a meta classifier for final decision in the SG (Polikar, 2006). These last ensemble schemes belong to the simultaneous training methods (Liu, 2003).
b) Combination strategy. It is necessary to combine the outputs of individual ANNs that make up the ensemble in such a way that the correct decisions are amplified, and incorrect ones are cancelled out. Two taxonomies can be considered, a) trainable vs. non-trainable combination strategies, b) combination strategies that apply to class labels vs. to class-specific continuous outputs.
In trainable combination rules, the parameters of the combiner, *weights*, are determined through a separate training algorithm, an example is the EM algorithm used by the ME model. There is no separate training involved in non-trainable rules, since the parameters become immediately available as the classifiers are generated. Weighted majority voting is an example of such non-trainable schemes (Polikar, 2006). In the second grouping several choices are available. For combining class labels we have majority voting, weighted majority voting, behavior knowledge space (BKS), and borda count schemes. For combining continuous outputs we can find some other schemes such as, algebraic combiners, decision templates and dempster-shafer based combination (Polikar, 2006).
Whereas there is no single ensemble generation algorithm or combination rule that is universally better than others, all of the approaches discussed above have been shown to be effective on a wide range of real world and benchmark datasets, provided that the classifiers can be made as diverse as possible. In the absence of any other prior information, the best ones are usually the simplest and least complicated ones that can learn the underlying data distribution (Polikar, 2006). Such an approach, does not guarantee the optimal performance, (Roli, 2002). For a small number of classifiers optimal ensembles can be found exhaustively, but the burden of exponential complexity of such search limits its practical applicability for

larger systems (Ruta, 2005). An appropriate design of NNE is where selection and fusion are recurrently applied to a population of best combinations of classifiers. rather than the individual best (Ruta, 2005).

4. Application of neural computation approach to identify fungicides

This section is dedicated to the application of the neural computation approach to face the resolution of difficult multi-component systems (overlapping) of fungicides with fluorescence detection. We present a study of automated decision making systems based on ANNs, data fusion and NNE to identify benzimidazoles fungicides (BFs). All these systems are modular neural computation systems, with a structure of pre-processing and processing stage, where their main element is the HUMANN neural architecture. This ANN is an unsupervised neural architecture which performs blind clustering. Because of this it can identify, in an automatic way, the different existing compounds in mixture samples without prior knowledge of which (and how many) of theses possible compounds can be found in the analysed sample. HUMANN has shown to be also suitable for identification of organochlorinated compounds with environmental interest (García Báez, 2003)(Suárez Araujo, 2003).

4.1 HUMANN-based systems to identify benzimidazoles fungicides with fluorescence detection

There are three HUMANN-based systems to identify the BFs Carbendazim 99.7% (methyl (1H-benzimidazol-2-yl) carbamate) (MBC), Benomyl 99.3% (methyl 1-(butylcarbomayl) benzimidazole-2-yl carbamate) (BM), Thiobendazole 99.6% (2-(4-thiazol)benzimidazole) (TBZ) and Furberidazole 99.6% (2-(2 furanyl)-1h-benzimidazole) (FB): a) HUMANN-based simple detector system, (Suárez Araujo, 2006), b) Multi-input HUMANN-based system, c) HUMANN-ensemble system (Suárez Araujo, 2009).

To gather the used data sets for each one of these HUMANN-based systems, an experiment design with chemical and computational requirements was performed (Suárez Araujo, 2009). These data sets, Table 1, are formed by emission, excitation and synchronous fluorescence spectra of each BFs, in six different concentrations, Fig. 3, which are adapted to the luminescent features of each compound. There were made mixtures for each one of them, with one, two, three and four compounds to obtain spectra at optimum, mean and median fluorescence features. To guarantee the measurements and define the margins of error in measurements, all the spectra were repeated three times.

The three HUMANN-based systems presented in the section share a common part, the feature extraction stage. This stage is implemented by the pre-processing module.

During the learning process, ANNs create internal representations of the characteristics of the training pattern set. The goal of the pre-processing stage is to prepare the information environment of HUMANN in such a way that it can adequately extract the information required. The pre-processing task constructs a set of feature vectors of the real signals which will be used in the processing stage. In the analysis of real fluorescence spectra two stages are followed: 1. modelling of the spectra. 2. Determination of the features vector which corresponds to the fluorescence spectrum model of mixture (García Báez, 2005)(Suárez Araujo, 2009).

	Characteristics	Benzimidazole family
	Compounds	4: Benomyl (BM), Carbendazim (MBC), Fuberidazole (FB), Thiabendazole (TBZ)
	Concentrations / compound:	6 (1 to 6)
	BM:	Interval = 250-1500 µg/l, Δc = 250 µg/l
	MBC:	Interval = 250-1500 µg/l, Δc = 250 µg/l
	FB:	Interval = 25-150 µg/l, Δc = 25 µg/l
	TBZ:	Interval = 2.5-15 µg/l, Δc = 2.5 µg/l
	Types of spectra:	4 synchronous, 1 excitation, 1 emission
	Mean / TBZ optimum (Δλ1):	Δλ = 47 nm, Interval = 200-400 nm
	Median (Δλ2):	Δλ = 53 nm, Interval = 200-400 nm
	BM/MBC optimum (Δλ3):	Δλ = 59 nm, Interval = 200-400 nm
	FB optimum (Δλ4):	Δλ = 29 nm, Interval = 200-400 nm
	Mean (λ_{em}):	λ_{em} = 327 nm, Interval = 200-315 nm
	Mean/Median (λ_{ex}):	λ_{ex} = 277 nm, Interval = 300-400 nm
Individual compounds	**Number of samples**	24 (4 compounds x 6 concentrations)
Individual compounds	**Number of spectra / sample**	3 per type = 18
Individual compounds	**Number of spectra of compounds**	72 per type = 432
Mixtures	**Number of mixtures**	100
Mixtures	**Number of spectra / mixture**	3 per type = 18
Mixtures	**Number of spectra of mixtures**	300 per type = 1800

Table 1. General characteristics of the sets of data of the benzimidazole fungicides. (From Chemical &Environmental Analysis Research Group (ULPGC))

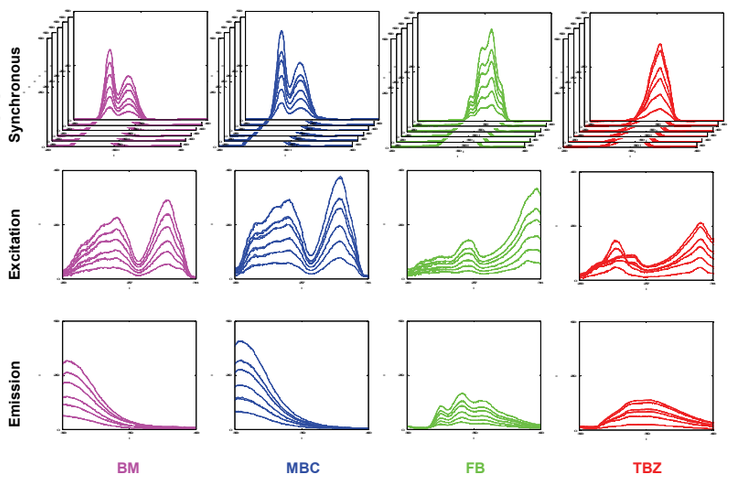


Fig. 3. Set of spectra of solutions of compounds belonging to the family of benzimidazole at different concentrations

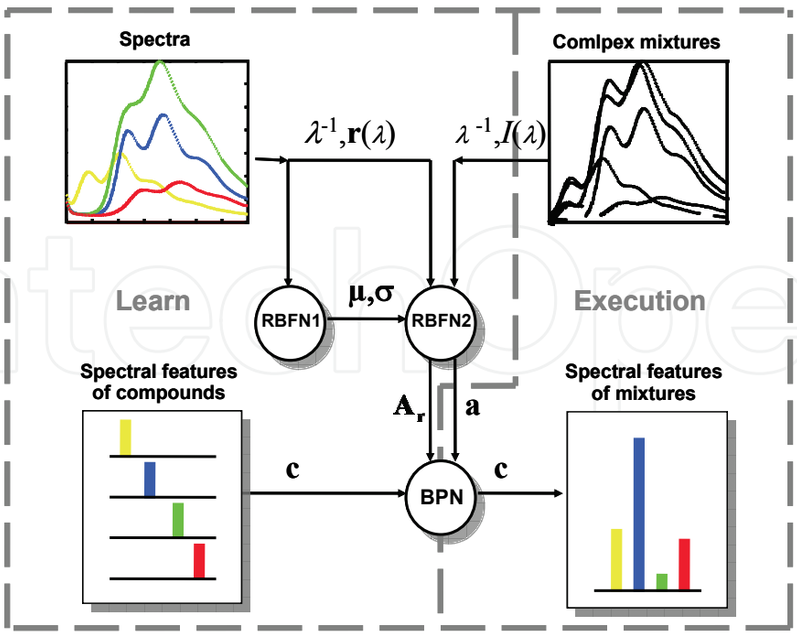


Fig. 4. Scheme of pre-processing stage

Using as basis the developments and later experimental studies made by Lloyd and Evett (Lloyd, 1977) and Cabaniss (Cabaniss, 1991) the fluorescence spectra can be modelled by a Gaussian distribution of intensity versus reciprocal wavelength (frequency). All spectra used in our developments have been previously normalized to the unit. The spectrum can be represented as equation:

$$I(\lambda) \approx a \cdot \mathbf{gaus}\left(\lambda^{-1}\right). \tag{21}$$

We will work with a linear approximation for the mixture model, such that the spectrum of a mixture will be represented by a linear combination of reference spectra. Reference spectra are the spectra of the compounds which can be identified in a mixture (García Báez, 2005)(Suárez Araujo, 2009):

$$I(\lambda) \approx c \cdot r(\lambda) = \sum_i c_i r_i(\lambda) \quad I(\lambda) \approx c \cdot A_r \cdot \mathbf{gaus}\left(\lambda^{-1}\right) \tag{22}$$

where $\mathbf{r}(\lambda)$ are the reference spectra and $\mathbf{c}$ is a vector with the contributions of each of the spectra of $\mathbf{r}(\lambda)$ in the mixture. $\mathbf{c}$ is then a vector which characterises a mixture and which is ideal for use as a feature vector for HUMANN-based systems. Spectral representation via Gaussian distribution will be carried out using RBFNs, and the approximation of concentration coefficients vector ($\mathbf{c}$) through a backpropagation network. Our pre-processing module is therefore made up of a complex neural structure of RBFNs + BPN, Fig. 4 (García Báez, 2005)(Suárez Araujo, 2009). The feature vectors finally obtained will be the input data to the processing stage for each HUMANN-based system studied at the last part of this section.

a) HUMANN-based simple detector system. It is the HUMANN, Fig. 2, version for simultaneous classifications. It will allow the firing of several labelling neurons

simultaneously. Each mixture (input pattern) will therefore be classified as belonging to several classes. In the output layer, the necessary adjustments of parameters have been made during learning process to generate 3 neurons corresponding to the three categories of BFs, which have been defined as BM/MBC, FB and TBZ, to be detected. These classes have been defined because of high BM and MBC spectral correlation. The training set is represented by conventional (emission and excitation) and synchronous fluorescence spectra (Suárez Araujo, 2006), Fig. 3.

To evaluate the overall efficiency of the used systems, we defined a function E of mixture error with respect to the categories detected in any one mixture:

$$E_{FN} = \frac{NCND}{NCIM}, E_{FP} = \frac{NCBD}{NCIM}, E = E_{FN} + E_{FP} \tag{23}$$

where $NCIM$ is the total number of categories in the mixture, $NCND$ is the number of categories non detected, $NCBD$ is the number of categories bad detected, and E_{FN} is the false negative error and E_{FP} the false positive error in order to evaluate the sensitivity and specificity.

Highly satisfactory results for fluorescence detection of these fungicides, using mixtures of up to four, have been obtained, being observed that the most sensitive to BFs determination type of spectra are the synchronous ones, Fig. 6(a). This will be the type of spectra which will constitute the information environment of the two next neural architectures, which improve the obtained results by a single HUMANN, Fig. 6 and Fig. 7.

b) Multi-input HUMANN-based system. One of the systems proposed for the processing module is a multi-input system based on HUMANN (HUMANN1–4). This system uses simultaneously the feature vectors (dimension 3) of four types of synchronous spectra explained in Table 1, as an input, Fig. 5(a). Via this multi-fluorescence spectra system, HUMANN has information from the luminescent sources whose Δλ has been tuned to the optimum values for each compound to be detected. The dimension of the optimum SOM layer for this system has been found to be 3x3.

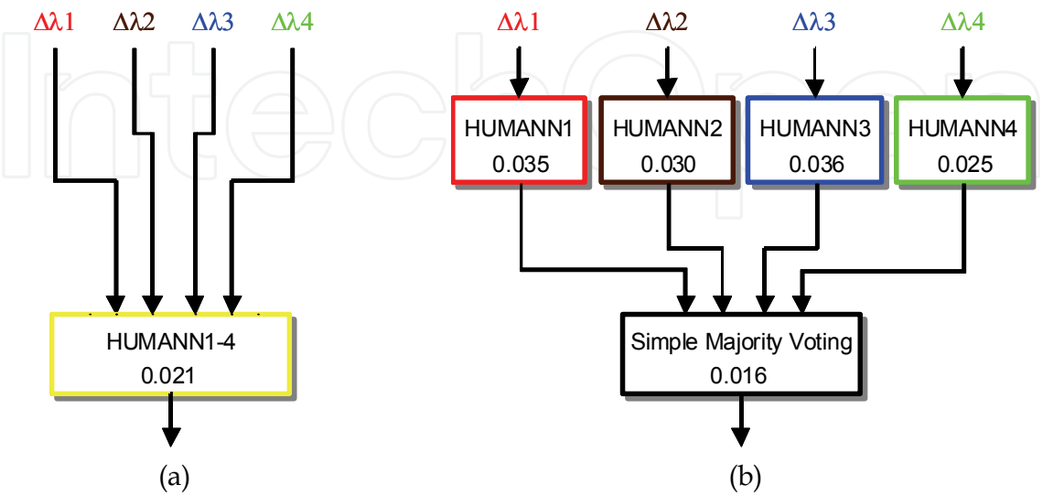
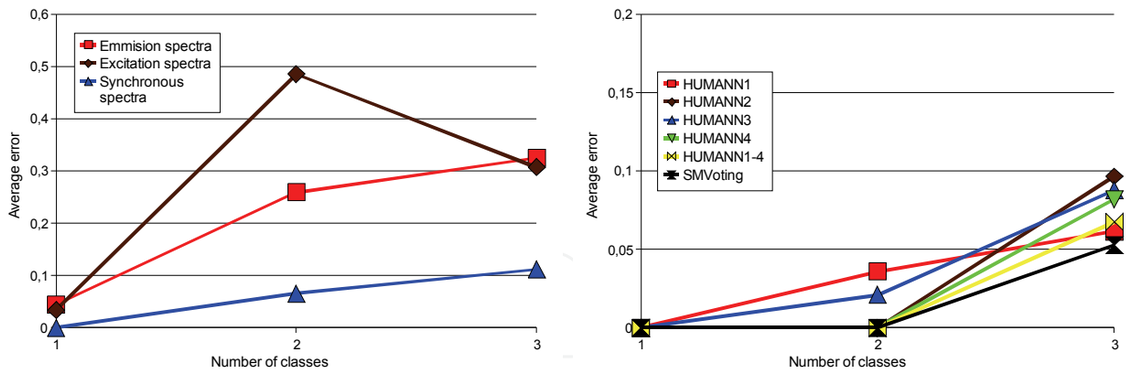
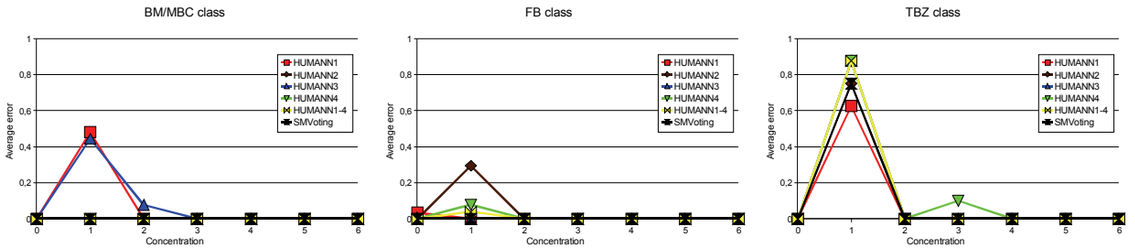


Fig. 5. HUMANN1-4 scheme (a). HUMANN-ensemble scheme (b)



(a) (b)

Fig. 6. Influence of the number of categories from the mixture in the average of the errors of the mixtures. Average error for each type of spectra (a). Average error for each HUMANN-based system (b)



(a) (b) (c)

Fig. 7. Influence of the concentration of the analytes BM/MBC (a), FB (b) and TBZ (c) in the mixture over the average of the errors of the mixtures. SMVoting is HUMANN-ensemble system with a simple majority voting combination strategy. The concentration "0" means absence of the analyte

c) HUMANN-ensemble system. The last automated decision making system to determine pesticides in a difficult multi-component system presented, is a NNE (HUMANN-ensemble). It will improve the obtained results with simple HUMANN fungicide detectors. The main aspect in the strategy for generating the ensemble members is diversity. The diversity of HUMANN ensemble depends on two aspects, correlation of HUMANN performance, which is a pair-wise measure of diversity, and the networks should have their errors in different parts of the input space. This effect is reached using four HUMANN architectures as ensemble members. Each one is working in a different local region of input space, which will be determined by the fluorescence characteristics of synchronous spectra: HUMANN1, HUMANN2, HUMANN3 and HUMANN4 respectively for the types of spectra $\Delta\lambda_1$, $\Delta\lambda_2$, $\Delta\lambda_3$ and $\Delta\lambda_4$ (Table 1). Each one of these HUMANNs has the same architecture of HUMANN1-4. Finally, a combination strategy is needed in order to obtain an overall NNE. This strategy must be such that the correct decisions may be amplified and the incorrect decisions cancelled or decreased (Polikar, 2006). Our HUMANN-ensemble system uses a non-trainable combination strategy that applies labels to class, the simple majority voting (SMV), Fig. 5(b). Each module polls a vote for each category, indicating whether it considers that the said category is present in the mixture or not. A later module is

responsible for the overall count, considering whether the compound belongs to one category or another, depending on whether more than half of the HUMANN modules consider this category is present in the mixture. The combination of the outputs of several ensemble members does not guarantee a superior performance to the best ensemble module, but it clearly reduces the risk of making a particularly poor selection (Polikar, 2006).
In modules of Fig. 5(b) we show the average value of the mixture errors over all the mixtures studied, taking into account the various systems deployed. As it can be seen, the most precise simple detectors are HUMANN4 and HUMANN2. Once again, HUMANN1-4 is more precise than any of the simple detectors, with a margin of *E* error 0.004 less than HUMANN4. HUMANN-ensemble is superior even to HUMANN1-4, with its margin of *E* error 0.005 less than the latter and 0.009 less than HUMANN4. Very successful results have then been obtained with the proposed neural computation systems having a high specificity and a good sensitivity in detection of benzimidazole fungicides. Furthermore, these systems are suitable for studying such figures of merit as precision, sensitivity and limit of detection of the method, for each of the fungicides, Fig. 7. The studied systems are also capable to indicate the existence of some other analyte different to BF in the analysed sample, or to face the resolution of multi-analyte mixtures (whatever analyte, even a clean sample). In this case, it is needed to use an appropriate training set in which there exists spectral characteristics of possible compounds that can be present in the difficult multi-analyte systems analysed. This is easily possible because of high adaptive character of HUMANN, concretely; this skill is embodied by the neurogenesis mechanism (García Báez, 2005).

5. Conclusion

This chapter is dedicated to the area of qualitative and quantitative chemical analysis of fungicides using neural computation methods. The study of this area will allow introducing the intelligent environmental monitoring area.
We have showed that the determination of fungicides is essentially carried out using chromatographic techniques combined with specific detection schemes. However, these methodologies are relatively expensive, time consuming and often require laborious sample treatment before analysis. There are other efficient alternatives to overcome these limitations, the chemometry methods.
A complementary approach to face these problems is using a computational model which has been biologically inspired, "the artificial neural network". This subject is the focus of the chapter. In this chapter we describe the resolution of difficult multi-component systems (overlapping) of fungicides, using ANNs, and also data fusion and NNE.
We present a general study of neural computation. This study covers theory and architecture of the main, and most used for chemical data processing, classical supervised and unsupervised ANNs. A new hierarchical unsupervised modular adaptive neural network (HUMANN) is also reported as very important ANNs in the identification of fungicides. The NNE has been analysed as a successful technique where outputs of a set of separately trained neural networks are combined to form one unified prediction. We have shown it to be a very appropriate approach to the scope of our application domain, noisy, outliers and overlapping data distributions.
Finally, this chapter presents the ability of ANN and NNE, to address the determination of benzimidazoles fungicides in complex mixtures, describing the outcomes of three implementations of such approaches: HUMANN-based simple detector system, multi-input

HUMANN-based system (HUMANN1-4), and HUMANN-ensemble. It has been demonstrated that these systems are very appropriate methods to face the extremely restricted scope of application of fluorescence spectrometry in the analysis of multi-component mixtures. This can be seen in the highly satisfactory results for fluorescence detection of these compounds of environmental interest, using mixtures of up to four different fungicides per mixture, and without prior knowledge of which (and how many) of these possible compounds can be found in the analysed sample. The comparative studies conclude the HUMANN-ensemble system provides better performance than simple detector system. This confirms the advantage to extract complementary pieces of information from different or/and diverse data sources.

The study presented in this chapter can be an important contribution in the environmental analytical chemistry field. This importance is based on that these neural computational methods use only spectral fluorescence data, are very simple, fast and economic method for monitoring of the environment. Furthermore these developments can consist on the first steps forward designing of an on line intelligent environmental monitoring system.

6. References

Alavanja, M.; Hoppin, J. & Kamel, F. (2004). Health effects of chronic pesticide exposure: cancer and neurotoxicity. *Annu Rev Public Health*, Vol. 25, pp. 155-197

Alvarez Buylla, A.; Kirn, J.R. & Nottebohm, F. (1990). Birth of projection neurons in adult avian brain may be related to perceptual or motor learning. *Science*, Vol. 249, No. 4975, pp. 1444-1446

Amari, S. (1990). Mathematical Foundations of Neurocomputing. *Proceedings of the IEEE*, Vol. 78, No. 9, pp. 1443-1463

Bro, R. (1997). PARAFAC. Tutorial and applications. *Chemom. Intell. Lab. Syst*, Vol. 38, No. 2, pp. 149-171

Brown, G.; Wyatt. J. & Harris, R. (2005). Diversity Creation Methods: A Survey and Categorisation. *Information Fusion*, Vol. 6, No. 1, pp. 5-20

Cabaniss, S. E. (1991). Theory of Variable-Angle Synchronous Fluorescence Spectra. *Analytical Chemistry*, Vol. 63, No. 13, pp. 1323-1327

Chiron, S.; Valverde, A.; Fernandez-Alba, A. & Barcelo, D. (1995). Automated sample preparation for monitoring groundwater pollution by carbamate insecticides and their transformation products. *AOAC Int.*, Vol. 78, No. 6, pp. 1346-1352

Dam, H.H.; Abbass, H.A.; Lokan, C. & Yao, X. (2008). Neural-Based Learning Classifier Systems. *IEEE Trans on Knowledge and Data Engineering*, Vol. 20, No. 1, pp. 26-39

Drucker, H.; Schapire, R. & Simard, P. (1993). Boosting performance in neural networks. *Int. J. Pattern Recognition Artif. Intelligence*, Vol. 7, No. 4, pp. 704-709

Drucker, H.; Cortes, C.; Jackel, L.D.; LeCun, Y. & Vapnik, V. (1994). Boosting and other ensemble methods. *Neural Computation*, Vol. 6, No. 6, pp. 1289-1301

Fritzke, B. (1997). Unsupervised ontogenic networks, In: *Handbook of Neural Computation*, Fiesler, E. & Beale, R. (Ed.), pp. C2.4:1-C2.4:16, IOP Publishing Ltd and Oxford University Press, ISBN 0750304138, Oxford, UK

Galeano-Diaz, T.; Guiberteau, A.; Ortiz-Burguillos, J.M. & Salinas, F. (1997). Comparison of chemometric methods: Derivative ratio spectra and multivariate methods (CLS, PCR and PLS) for the resolution of ternary mixtures. *Analyst*, Vol. 122, pp. 513-517

Gallinari, P. (1995). Training of Modular Neural Net Systems, In: *The Handbook of Brain Theory and Neural Networks*, Arbib, M. A. (Ed.), pp. 582-585, MIT Press, ISBN 0-262-51102-9, Cambridge, MA

García Báez, P.; Fernández López, P. & Suárez Araujo, C.P. (2001). Extension of HUMANN for Dealing with Noise and with Classes of Different Shape and Size: A Parametric Study, In: *Bio-Inpired Applications of Connectionism*, Mira, J. & Prieto, A. (Ed.), pp. 96-103, Springer-Verlag, ISBN 3-540-42237-4, Berlin, GE

García Báez, P.; Fernández López, P. & Suárez Araujo, C. P. (2003). A Parametric Study of HUMANN in Relation to the Noise. Application to the Identification of Compounds of Environmental Interest. *Systems Analysis Modelling and Simulation*, Vol. 43, No. 9, pp. 1213-1228

García Báez, P. (2005). HUMANN: una nueva red neuronal artificial adaptativa, no supervisada, modular y jerárquica: aplicaciones en neurociencia y medioambiente. Departamento de Informática y Sistemas, Universidad de Las Palmas de GC

García Báez, P.; Suárez Araujo, C.P.; Sánchez Rodríguez, Á. & Santana Rodríguez, J.J. (2010). Towards an efficient computational method for fluorescence identification of fungicides using data fusion and neural ensemble tchns.. *Luminescence*, in press

Istamboulie, G.; Cortina-Puig, M.; Marty, J.L. & Noguer, T. (2009). The use of Artificial Neural Networks for the selective detection of two organophosphate insecticides: Chlorpyrifos and chlorfenvinfos. *Talanta*, Vol. 79, No. 2, pp. 507-511

Guiberteau, A.; Galeano, T.; Mora, N.; Salinas, F.; Ortíz, J.M. & Vire, J.C. (2001). Resolution by polarographic techniques of the ternary mixture of captan, captafol and folpet by using PLS calibration and artificial neuronal networks. *Computers and Chemistry*, Vol. 25, No. 5, pp. 459-473

Halko, R.; Padron Sanz, C.; Sosa Ferrera, Z. & Santana Rodríguez, J.J. (2006). . *J AOAC Int*, Vol. 89, pp. 1403-1409

Hansen, L.K. & Salamon, P. (1990). Neural Network Ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 10, pp. 993-1001

Haykin, S. (1994). *Neural Networks. A Comprhensive Foundation*, Macmillan College Publishing, ISBN 0-02-352761-7, New York, NY

Hebb, D.O. (1949). *The organization of behavior: A neuropsychological theory*, John Wiley and Sons, ISBN , New York, NY

Hecht Neilsen, R. (1990). *Neurocomputing*, Addison-Wesley Publishing, ISBN 0201093553, Reading, MA

Henry, R.C. (2003). Multivariate receptor modeling by N-dimensional edge detection. *Chemom. Intell. Lab. Syst.*, Vol. 65, No. 2, pp. 179-189

Hertz, J.; Krogh, A. & Palmer, R.G. (1991). *Introduction to the Theory of Neural Computation*, Addison-Wesley Publishing, ISBN 0-201-50395-6, Reading, MA

Jacobs, R.A.; Jordan, M.I.; Nowlan, S.J. & Hinton, G.E. (1991). Adaptive mixtures of local experts. *Neural Computation*, Vol. 3, No. 1, pp. 79-87

Jain, A.K.; Mao, J. & Mohiuddin, K. (1996). Artificial Neural Networks: A Tutorial. *IEEE Computer*, Vol. 29, No. 3, pp. 31-44

Jolliffe, I.T (1986). *Principal Component Analysis*, Springer-Verlag, ISBN 0387962697, New York, NY

Jordan, M.J. & Jacobs, R.A. (1994). Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, Vol. 6, No. 2, pp. 181-214

Karl-Heinz, O.; Araníbar, N.; Singh, B. & Stockton, G.W. (2003). Metabonomics classifies pathways affected by bioactive compounds. Artificial neural network classification of NMR spectra of plant extracts BASF. *Phytochemistry*, Vol. 62, No. 6, pp. 971-985

Kohonen, T. (1989). *Self-Organization and Associative Memory, 3rd ed*, Springer Series in Information Sciences, ISBN 3-540-51387-6, Berlin, GE

Kohonen, T. (1997). *Self-Organizing Maps, 2nd ed*, Springer Series in Information Sciences, ISBN 3-540-62017-6, Berlin, GE

Kuncheva, L.I.; Bezdek, J.C. & Duin, R. (2001). Decision templates for multiple classifier fusion: An experimental comparison. *Pattern Recognition*, Vol. 34, No. 2, pp. 299-314

Li, B.; He, Y. & Xu, C. (2006). Simultaneous determination of three organophosphorus pesticides residues in vegetables using continuous-flow chemiluminescence with ANN calibration. *Talanta*, Vol. 72, No. 1, pp. 223-230

Liu, Y, Yao, X. & Higuchi, T. (2003). Designing Neural Network Ensembles by Minimising Mutual Information, In: *Computational Intelligence in Control*, Mohammadian, M.; Sarker, R.A. & Yao, X. (Ed.), pp. 1-21, Idea Group Inc., ISBN 1-59140-037-6, London

Lloyd, J.B.F. & Evett, I.W. (1977). Prediction of Peak Wavelengths and Intensities in Synchronously Excited Fluorescence. *Anal. Chemistry*, Vol. 49, No. 12, pp. 1710-1715

Martínez Vidal, J.L.; Pablos Espada, M.C.; Garrido Frenich, A. & Arrebola, F.J. (2000). *J Chromatogr A*, Vol. 867, pp. 235-245

McCulloch, W.S. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, Vol. 5, pp. 115-133

Minsky, M. L. & Papert, S. A. (1969). *Perceptrons*, MIT Press, ISBN , Cambridge, MA

Musavi, M.T.; Ahmed, W.; Chan, K.H.; Faris, K.B. & Hummels, D.M. (1992). On the Training of Radial Basis Function Classifiers. *Neural Networks*, Vol. 5, No. 4, pp. 595-603

Ni, Y.; Huanga, C. & Kokot, S. (2004). Application of multivariate calibration and artificial neural networks to simultaneous kinetic-spectrophotometric. *Chemometrics and Intelligent Laboratory Systems*, Vol. 71, No. 2, pp. 177-193

Olivieri, A.C. (2008). Analytical advantages of multivariate data processing. One, two, three, infinity?. *Anal. Chem.*, Vol. , pp. 5713-5720

Opitz, D.W. & Shavlik, J.W. (1996). Actively Searching for an Effective Neural-Network Ensemble. *Connection Science*, Vol. 8, No. 3, pp. 337-353

Paatero, P. & Tapper, U. (1994). Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, Vol. 5, No. 2, pp. 111-126

Paatero, P. (1999). The multilinear engine - a table-driven, least squares program for solving multilinear problems, including the n-way parallel factor analysis model. *J Comput Graph Stat*, Vol. 8, No. 4, pp. 854-888

Piccirilli, G.N. & Escandar, G.M. (2006). Partial least-squares with residual bilinearization for the spectrofluorimetric det. of pesticides. A solution of the problems of inner-filter eff.. *Analyst*, Vol. 131, pp. 1012-1020

Polikar, R. (2006). Ensemble Based Systems in Decision Making. *IEEE Circuits and Systems Magazine*, Vol. 6, No. 3, pp. 21-45

Polikar, R.; Topalis, A.; Green, D.; Kounios, J. & Clark, C.M. (2007). Comparative multiresolution wavelet analysis of ERP spectral bands using an ensemble of classifiers app. for early diagnosis of Alzheimer's disease. *Computers in Biology and Medicine*, Vol. 37, No. 4, pp. 542-558

Rivas, A. et al. (2004). Exposición humana a disruptores endocrinos. *Monográfico Revista Ecosistemas*, Vol. 13, No. 3

Rojas, R. (1996). *Neural Networks. A Systematic Introduction*, Springer-Verlag, ISBN 3540605053, Berlin, GE

Roli, F. & Giadnto, G. (2002). Design of multiple classifier systems, In: *Hybrid Methods in Pattern Recognition*, Bunke, H. & Kandel, A. (Ed.), pp. 199-226, World Scientific Publishing, ISBN 9789812778147, Singapore

Rosenblatt, F. (1961). *Principles of Neurodynamics*, Spartan Books, ISBN , Washington, WA

Rumelhart, D.E.; Hinton, G.E. & McClelland, J.L. (1986). A general framework for parallel distributed processing, In: *Parallel distributed processing: explorations in the microstr. of cog.*, (Ed.), pp. 45-76, MIT Press, ISBN 0-262-68053-X, Cambridge, MA

Ruta, D. & Gabrys, D. (2005). Classifier selection for majority voting. *Information Fusion*, Vol. 6, No. 1, pp. 63-81

Salau, J.S.; Tauler, R.; Bayona, J.M. & Tolosa, I. (1997). Input characterization of sedimentary organic contaminants and molecular markers in the Northwestern Mediterranean Sea by exploratory data analysis. *Environ. Sci. Technol*, Vol. 31, No. 12, pp. 3482-3490

Sastry, C.C.S.P. & Vijaya, D. (1987). Spectrophotometric determination of some insecticides with 3-methyl-2-benzothiazolinone hydrazone hydrochloride. *Talanta*, Vol. 34, No. 3, pp. 372-374

Schapire, R.E. (1990). The strength of weak learnability. *Machine Learning*, Vol. 5, No. 2, pp. 197-227

Simpson, P.K. (1990). *Artificial Neural Systems. Foundations, paradigms, applications, and impl.*, Pergamon Press, ISBN 0080378943, New York, NY

Suárez Araujo, C.P. (1996). Neurociencia y computación neuronal: una perspectiva, In: *Neurociencia y Computación Neuronal*, Suárez Araujo, C.P. & Regidor García, J. (Ed.), pp. 251-266, Servicio de publicaciones de la ULPGC, ISBN 84-89728-11-9, Las Palmas de GC, ES

Suárez Araujo, C.P (1997). Novel Neural Network Models for Computing Homothetic Invariances: An Image Algebra Notation. *Journal of Mathematical Imaging and Vision*, Vol. 7, pp. 69-83

Suárez Araujo, C.P. (1999). Neural computation approach in luminescence spectrometry. *Biomedical Chromatography*, Vol. 13, pp. 187-188

Suárez Araujo, C.P.; Santana Rodríguez, J.J.; García Báez, P. & Betancort Rodríguez, J.R. (2003). HUMANN-Based Computational Neural System for the Determination of Pollutants using Fluorescence Measurements, *Proc. of the Int. Conf. on Engineering Applications of Neural Networks*, pp. 290-297, ISBN 84-930984-1-8Dpto. de Ingeniería de Sistemas y Automática, Málaga, ES

Suárez Araujo, C.P.; García Báez, P.; Sánchez Rodríguez, A. & Santana Rodríguez, J.J. (2006). Design of a HUMANN-based method for the determination of benzimidazole fungicides with fluorescence. *Luminescence*, Vol. 21, pp. 342-344

Suárez Araujo, C.P.; García Báez, P.; Sánchez Rodríguez , A. & Santana Rodríguez, J.J. (2009). HUMANN-based system to determine pesticides using multifluorescence spectra: An ensemble approach. *Analytical and Bioanalytical Chemistry*, Vol. 394, No. 4, pp. 1059-1072

Taguchi, V.Y.; Jenkins, S.W.A.; Crozier, P.W. & Wang, AT. (1998). *J Am Soc Mass Spectrom*, Vol. 9, pp. 830-839

Tucker, L.R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika*, Vol. 31, No. 3, pp. 279-311

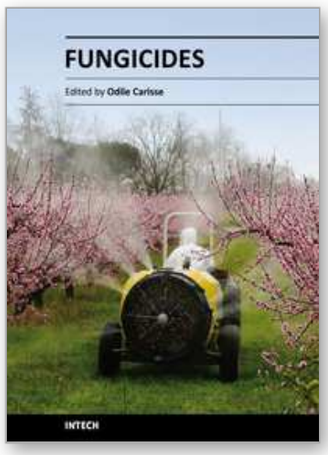
Vassilakis, Y.; Tipi, D. & Scoullos, M. (1998). *J Chromatogr A*, Vol. 823, pp. 49-58

Wan, C. & Harrington, P.B. (2000). Screening GC-MS data for carbamate pesticides with temperature-constrained-cascade correlation neural networks. *Analytica Chimica Acta*, Vol. 408, No. 1-2, pp. 1-12

Werbos, P.J. (1974) Beyond regression: new tools for prediction and analysis in the behavioral sciences.Doctoral ThesisHarvard University.

Yang, C.C.; Prasher, S.O.; Sreekanth, S.; Patni, N.K. & Masse, L. (1997). An artificial neural network model for simulating pesticide concentrations in soil. *Transactions of the American Society of Agricultural Engineers*, Vol. , pp. 1285-1294

Zhu, S.H.; Wu, H.L.; Xia, A.L.; Ha,. Q.J. & Zhang, Y. (2007). Determination of Carbendazim in Bananas by Excitation- Emission Matrix Fluorescence with Three Second-order Calibration Methods. *Analytical Science*, Vol. 23, No. 10, pp. 1173-1177



Fungicides

Edited by Odile Carisse

ISBN 978-953-307-266-1

Hard cover, 538 pages

Publisher InTech

Published online 14, December, 2010

Published in print edition December, 2010

Plant and plant products are affected by a large number of plant pathogens among which fungal pathogens. These diseases play a major role in the current deficit of food supply worldwide. Various control strategies were developed to reduce the negative effects of diseases on food, fiber, and forest crops products. For the past fifty years fungicides have played a major role in the increased productivity of several crops in most parts of the world. Although fungicide treatments are a key component of disease management, the emergence of resistance, their introduction into the environment and their toxic effect on human, animal, non-target microorganisms and beneficial organisms has become an important factor in limiting the durability of fungicide effectiveness and usefulness. This book contains 25 chapters on various aspects of fungicide science from efficacy to resistance, toxicology and development of new fungicides that provides a comprehensive and authoritative account for the role of fungicides in modern agriculture.

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen