

 Open access • Journal Article • DOI:10.1093/JIGPAL/JZR023

Soft computing models to analyze atmospheric pollution issues — [Source link](#)

Ángel Arroyo, Emilio Corchado, Verónica Tricio

Institutions: University of Burgos, University of Salamanca

Published on: 01 Aug 2012 - Logic Journal of The Ijpl V Bulletin of The Ijpl (Oxford University Press)

Topics: Soft computing

Related papers:

- [Environmental Modelling Using Cloud Computing Tools: Case Studies and Examples](#)
- [Air pollution modelling, simulation and computational methods: a review](#)
- [Air Pollution Models As Tools To IntegrateScientific Results In Environmental Policy](#)
- [Air quality modeling for policy development.](#)
- [Multi-Agent Based Simulation of Environmental Pollution Issues: A Review](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/soft-computing-models-to-analyze-atmospheric-pollution-26ubng4w2b>

Soft Computing Models to Analyze Atmospheric Pollution issues

Ángel Arroyo¹, Emilio Corchado² and Verónica Tricio³

¹*Departamento de Ingeniería Civil, Universidad de Burgos, Burgos, Spain*

²*Departamento de Informática y Automática, Universidad de Salamanca, Salamanca, Spain*

³*Departamento de Física, Universidad de Burgos, Burgos, Spain*

Abstract. Multidisciplinary research into statistical and soft computing models is detailed that analyses data on inmissions of atmospheric pollution in urban areas. The research analyzes the impact on atmospheric pollution of an extended bank holiday weekend in Spain. Levels of atmospheric pollution are classified in relation to the days of the week, seeking to differentiate between working days and non-working days by taking account of such aspects as industrial activity and traffic levels. The case of study is based on data collected by a station at the city of Burgos, which forms part of the pollution measurement station network within the Spanish Autonomous Region of Castile-Leon.

Keywords. Artificial neural networks; soft computing; meteorology; atmospheric pollution; statistical model.

1. Introduction

It has been accepted for some years now that air pollution not only represents a health risk, but that it also, for example, reduces food production and vegetative growth due to its negative effects on photosynthesis.

Systematic measurements in Spain, which are usually taken within large cities, are fundamental, due to the health risks caused by high levels of atmospheric pollution. European legislation, will in the long term establish how and where such pollutants should be measured.

Thus, all efforts that are directed towards studying these phenomena [1, 2] may improve our understanding and help us to prevent the serious problematic nature of atmospheric pollution.

Several years ago, certain public, regional and municipal bodies adopted the principles of the V Programme Framework of the European Union and those of the Rio Conference concerning the application of community and international directives; in particular, those on environmental air quality in our cities. Previous directives were based on European legislation with which Spain, as a member of the European Community, was expected to comply.

The basis of this study is the application of a series of statistical and soft computing models for studying the evolution of air pollution throughout the week as a function of chosen variables.

Soft computing [3, 4, 5] consists of various technologies which are used to solve inexact and complex problems [6]. It is used to investigate, simulate, and analyze complex issues and phenomena in an attempt to solve real-world problems [7].

2. Soft Computing and Statistical Models

Several statistical and soft computing models are used in this study to analyze data taken from cases studies on aerosol pollutants in order to assess their performance.

Exploratory Projection Methods (EPP)

EPP [8, 9, 10] is a more recent statistical method aimed at solving the difficult problem of identifying structure in high dimensional data. It does this by projecting the data onto a low dimensional subspace in which we search for its structure by eye. However not all projections will reveal the data's structure equally well. EPP therefore defines an index that measures how "interesting" a given projection is, and it then represents the data in terms of projections that maximize that index. "Interesting" structure is usually defined with respect to the fact that most projections of high-dimensional data onto arbitrary lines through most multi-dimensional data give almost Gaussian distributions [11]. Therefore, in order to identify "interesting" features in data, it is necessary to look for those directions onto which the data-projections are as far from the Gaussian as possible.

A simple measure of deviation from a Gaussian distribution is based on the higher order moments of the distribution. Kurtosis is based on the normalized fourth moment of the distribution and measures the heaviness of the tails of a distribution. A bimodal distribution will often have a negative Kurtosis and therefore negative Kurtosis can signal that a particular distribution shows evidence of clustering.

Principal Component Analysis (PCA)

PCA [10] gives the best linear compression of the data in terms of least mean square error and can be implemented by several artificial neural networks [12, 13]. The basic PCA network [14] applied in this study is described by "Eq. (1)" and "Eq. (2)": an N -dimensional input vector at time t , $x(t)$, and an M -dimensional output vector, y , with W_{ij} being the weight linking input j to output i , and η being the learning rate. Its activation and learning may be described as follows:

Feedforward step, "Eq. (1)":

$$y_i = \sum_{j=1}^N W_{ij} x_j, \forall i \quad (1)$$

Feedback step, "Eq. (2)":

$$e_j = x_j - \sum_{i=1}^M W_{ij} y_i \quad (2)$$

Change weights, "Eq. (3)":

$$\Delta W_{ij} = \eta e_j y_i \quad (3)$$

This algorithm is equivalent to Oja's Subspace Algorithm [19], "Eq. (4)":

$$\Delta W_{ij} = \eta e_j y_i = \eta (x_j - \sum_k W_{kj} y_k) y_i \quad (4)$$

EPP [11, 15] projects the data onto a low dimensional subspace which allows its structure to be examined by eye. As explained before, this is done by means of an index that measures the "interestingness" of a given projection, the data for which is then represented by projections that maximize the most "interesting" vectors. "Interestingness" is usually defined by the fact that most projections of high-dimensional data onto arbitrary vectors through a majority of multi-

dimensional data give almost Gaussian distributions [11, 16], which allows a search to be made for those directions onto which the data-projections are as far from the Gaussian as possible.

ε - Insensitive Hebbian Learning

It has been shown [17] that the nonlinear PCA rule, “Eq. (5)”:

$$\Delta W_{ij} = \eta \left(x_j f(y_i) - f(y_i) \sum_k W_{kj} f(y_k) \right) \quad (5)$$

Can be derived as an approximation of the best non-linear compression of the data. Initially, therefore, there is a cost function, “Eq. (6)”:

$$J(W) = \frac{1}{2} E \left\{ \|\mathbf{x} - Wf(W^T \mathbf{x})\|^2 \right\} \quad (6)$$

Which may be minimized to give, “Eq. (5)” [18] using the residual in the linear version of, “Eq. (8)” to define a cost function of the residual, “Eq. (7)”:

$$J = f_1(\mathbf{e}) = f_1(\mathbf{x} - W\mathbf{y}) \quad (7)$$

Where $f_1 = \|\cdot\|^2$ is the (squared) Euclidean norm in the standard linear or nonlinear PCA rule. With this choice of $f_1(\cdot)$, the cost function is minimized with respect to any set of samples from the data set on the assumption that the residuals are chosen independently and identically distributed from a standard Gaussian distribution. The minimization of J is equivalent to minimizing the negative log probability of the residual, if \mathbf{e} is Gaussian, “Eq. (8)”:

$$\text{Let } p(\mathbf{e}) = \frac{1}{Z} \exp(-\mathbf{e}^2) \quad (8)$$

Then, p will denote a general cost function associated with this network as in, “Eq. (9)”:

$$J = -\log p(\mathbf{e}) = \mathbf{e}^2 + K \quad (9)$$

Where K is a constant. A gradient descent is performed on J , in “Eq. (10)”:

$$\Delta W \propto -\frac{\partial J}{\partial W} = -\frac{\partial J}{\partial \mathbf{e}} \frac{\partial \mathbf{e}}{\partial W} \approx \mathbf{y}(2\mathbf{e})^T \quad (10)$$

In which a less important term has been discarded [11].

In general [19], the minimization of such a cost function may be understood to increase the residual probability, depending on the Probability Density Function (*pdf*) of the residuals. Thus, if the *pdf* of the residuals is known, this knowledge could be used to determine the optimal cost function. [20] investigated this possibility using the (one dimensional) function, “Eq. (11)”:

$$p(\mathbf{e}) = \frac{1}{2 + \varepsilon} \exp(-|\mathbf{e}|_\varepsilon) \quad (11)$$

Where, “Eq. (12)”:

$$|\mathbf{e}|_\varepsilon = \begin{cases} 0 & \forall |\mathbf{e}| < \varepsilon \\ |\mathbf{e}| - \varepsilon & \text{otherwise} \end{cases} \quad (12)$$

With ε being a small scalar.

With this model of the residual *pdf*, the optimal $f_1(\cdot)$ function is the ε -insensitive cost function, “Eq. (13)”:

$$f_1(\mathbf{e}) = |\mathbf{e}|_\varepsilon \quad (13)$$

In the case of the negative feedback network, the learning rule is shown in “Eq. (14)”:

$$\Delta W \propto -\frac{\partial J}{\partial W} = -\frac{\partial f_1(\mathbf{e})}{\partial \mathbf{e}} \frac{\partial \mathbf{e}}{\partial W} \quad (14)$$

Which gives “Eq. (15)”:

$$\Delta W_{ij} = \begin{cases} 0 & \text{if } |e_j| < \varepsilon \\ \eta y(\text{sign}(e)) & \text{otherwise} \end{cases} \quad (15)$$

Applying Maximum Likelihood Hebbian Learning (MLHL)

The MLHL [21, 22] algorithm is defined as follows. Let the residual *pdf* after feedback be as in “Eq. (16)”:

$$p(\mathbf{e}) = \frac{1}{Z} \exp(-|\mathbf{e}|^p) \quad (16)$$

Where p is a parameter which relates the *pdf* to the learning rule “Eq. (20)”.

There is a general cost function associated with this network, “Eq. (17)”:

$$J = E(-\log p(\mathbf{e})) = E(|\mathbf{e}|^p + K) \quad (17)$$

Where K is a constant that is independent of W and the expectation is taken over the input data set. A gradient descent is performed on J : “Eq. (18)”

$$\begin{aligned} \Delta W &\propto -\frac{\partial J}{\partial W_{w(t-1)}} = -\frac{\partial J \partial e}{\partial e \partial w_{w(t-1)}} \approx \\ &\approx E\{y | e|^{p-1} \text{sig}(e)\}^T |_{w(t-1)} \end{aligned} \quad (18)$$

Where T denotes the transpose of a vector, and the operation of taking powers of the norm of e is on an element wise basis as it is taken from a derivative of a scalar with respect to a vector. If the conditions of stochastic approximation [23] are satisfied, it may be approximated with a difference equation. The function to be approximated is clearly sufficiently smooth, the learning rate is acceptable and the rule is as follows, “Eq. (19)”:

$$\Delta W_{ij} = \eta y_i \text{sign}(e_j) |e_j|^{p-1} \quad (19)$$

Researchers [9, 10] have shown that it is less important to obtain the exact distribution when searching for a specific source than it is to arrive at a distribution that is approximately correct. Therefore, the network operation is as follows: initially there is a feedforward step as in “Eq. (1)” and a feedback step as in “Eq. (2)” followed by a weights change, “Eq. (20)”:

$$\Delta W_{ij} = \eta y_i \text{sign}(e_j) |e_j|^{p-1} \quad (20)$$

By maximizing the likelihood of the residual with respect to the actual distribution, MLHL is matching the learning rule to the residual.

Cooperative Maximum Likelihood Hebbian Learning (CMLHL)

An extended version of the MLHL model is the Cooperative Maximum Likelihood Hebbian Learning (CMLHL) [24] model. CMLHL is based on MLHL [21, 22] adding lateral connections which have been derived from the Rectified Gaussian Distribution [22]. The resultant net can find the independent factors of a data set but does so in a way that captures some type of global ordering in the data set.

Consider an N -dimensional input vector (x), an M -dimensional output vector (y) and a weight matrix W , where the element W_{ij} represents the relationship between input x_j and output y_i , then as is shown in [22, 25], the CMLHL can be carried out as a four-step procedure:

Feed-forward step, outputs are calculated “Eq. (21)”:

$$y_i = \sum_{j=1}^N W_{ij} x_j, \forall i \quad (21)$$

Lateral activation passing step, “Eq. (22)”:

$$y_i(t+1) = [y_i(t) + \tau(b - Ay)]^+ \quad (22)$$

Feedback step, "Eq. (23)":

$$e_j = x_j - \sum_{i=1}^M W_{ij} y_i, \forall j \quad (23)$$

Weights update step, learn the neural network, "Eq. (24)":

$$\Delta W_{ij} = \eta \cdot y_i \cdot \text{sign}(e_j) |e_j|^{p-1} \quad (24)$$

Where t is the temperature, $[\]^+$ is necessary to ensure that the y -values remain in the positive quadrant, η is the learning rate, τ is the "strength" of the lateral connections, b the bias parameter, p a parameter related to the energy function, [22, 24, 25] and A is a symmetric matrix used to modify the response to the data [16]. The effect of this matrix is based on the relation between the distances separating the output neurons.

ISOMAP

Methods for nonlinear dimensionality reduction have proven successful in many applications, although the weakness of a method such as Multidimensional Scaling (MDS) [26] is that they are based on Euclidean distances and do not take the distribution of the neighboring datapoints into account. ISOMAP nonlinear dimensionality reduction [27] resolves this problem by attempting to preserve pairwise geodesic (or curvilinear) distance between datapoints. Geodesic distance is the distance between two points measured over the manifold. ISOMAP defines the geodesic distance as the sum of edge weights along the shortest path between two nodes (computed using Dijkstra's algorithm [26], for example). The doubly-centered geodesic distance matrix K in ISOMAP is of the form, "Eq. (25)":

$$K = \frac{1}{2} HD^2H \quad (25)$$

Where $D^2 = D_{ij}^2$ means the element wise square of the geodesic distance matrix $D = [D_{ij}]$, and H is the centring matrix, given by "Eq. (26)":

$$H = I_n - \frac{1}{N} e_N e_N^T \quad (26)$$

In which $e_N = [1 \dots 1]^T \in \mathbf{R}^N$

The top N eigenvectors of the geodesic distance matrix represent the coordinates in the new n -dimensional Euclidean space.

The complete isometric feature mapping, or ISOMAP, algorithm has three steps:

1. Construct neighborhood graph. Define the graph G over all data points by connecting points i and j if [as measured by $d_X(i, j)$] they are closer than e (e -Isomap), or if i is one of the K nearest neighbors of j (K -Isomap). Set edge lengths equal to $d_X(i, j)$.

2. Compute shortest paths. Initialize $d_G(i, j) = d_X(i, j)$ if i, j are linked by an edge; $d_G(i, j) = \infty$ otherwise. Then for each value of $k=1, 2, \dots, N$ in turn, replace all entries $d_G(i, j)$ by $\min\{d_G(i, j), d_G(i, k) + d_G(k, j)\}$. The matrix of final values $D_G = \{d_G(i, j)\}$ will contain the shortest path distances between all pairs of points in G .

3. Construct d -dimensional embedding. Let λ_p be the p -th eigenvalue (in decreasing order) of the matrix $\lambda(D_G)$, and v_p^i be the i -th component of the p -th eigenvector.

ISOMAP complements, and may be combined with linear extensions of PCA based on higher order statistics, such as independent component analysis (ICA) [10]. It may also lead to a better understanding of how the brain comes to represent the dynamic appearance of objects, where psychophysical studies of apparent motion point to the central role of geodesic transformations on nonlinear manifolds.

Curvilinear Component Analysis (CCA)

Curvilinear [28] is a non-linear projection method that preserves distance relationships in both input and output spaces. CCA is a useful method for redundant and non-linear data structure representation and can be used in dimensionality reduction. CCA is useful with highly non-linear data, where PCA or any other linear method fails to give suitable information [28].

CCA brings some improvements to other methods like Sammon's Mapping [29], although when unfolding a nonlinear structure, Sammon's Mapping cannot reproduce all distances. One way to get round this problem consists in favoring local topology: CCA tries to reproduce short distances firstly, long distances being secondary. Formally, this reasoning led to the following error function (without normalization), "Eq. (29)":

$$E_{CCA} = \sum_{i,j=1}^N (d_{i,j}^n - d_{i,j}^p)^2 F_{\lambda}(d_{i,j}^p) \quad (29)$$

By comparison with E_{Sammon} , E_{CCA} has an additional weighting function F depending on $d_{i,j}^p$ and on parameter λ . The F factor is a decreasing function of its argument, so it is used to favor local topology preservation. For example, F could be a step function of $(\lambda - d)$.

Another improvement brought by CCA to Sammon's mapping is the use of Vector Quantization (VQ). VQ decreases the computational load of the distance calculation (there $0.5 \cdot N \cdot (N-1)$ distance to compute). Unfortunately, the use of VQ implies the use of an interpolation in order to project vectors that are not prototypes of the codebook.

3. Case Study. Analysis of Atmospheric Pollution throughout the week using Soft Computing Models

This study presents interesting results related to the evolution of different pollution parameters using the records of an air quality control station (made available by the Department of the Environment-Directorate of Environmental Quality of the Government of the Spanish Autonomous Region of Castile-Leon) [30, 31]. The aforementioned station is situated in the urban area of the city of Burgos. The study was conducted over approximately half a year in 2007. After analyzing all the weeks in the period, a week with a national bank holiday weekend in December was selected because many people in Spain travel over this period, greatly decreasing traffic in large cities and almost completely curtails industrial activity.

In this study, the following variables were analyzed: SO_2 , NO , NO_2 , PM_{10} . These are standard variables used to monitor urban air quality. The variables express the concentration of the pollutant in $\mu g/m^3$. Regulations have been established for the admitted margins of tolerance, values limit and periods of time established. In the case of NO and NO_2 , the values limit come fixed on the Spanish Royal decree (R.D.) 1073/2003, transposed of the board 1999/30/CE. The limits of PM_{10} and SO_2 are the demanded for the R.D. 1073/2002, transposed of the board 1999/30/CE.

The general characteristics of the site where the measurement station used in the study was situated are as follows: Burgos, a city in the north-centre of Spain with a population of around 170,000 inhabitants and a total municipal area of approximately 107 km^2 . The city of Burgos is 854 masl (meters above sea level) at latitude (N) $42^{\circ}20'$ and longitude (W) $3^{\circ}42'$. The measurement station is located within the city and may be classified as an urban station.

This study examines the performance of several statistical and soft computing methods when analyzing the above-mentioned pollution variables, in order to track the evolution of air pollution over a significant time period. Thus, the main aim is to identify the extent to which air quality is affected on days with low industrial activity and reduced traffic.

Experiments and Results

As already mentioned, the aim of this study is to analyse the difference between working days and non-working days using four variables with information on pollution. The analysis was undertaken over six months.

The study, which forms part of a more ambitious project [32, 33], is based on a file containing meteorological and pollution data sets recorded at fifteen-minute intervals: a daily total of 96 records for all data in 2007, referring only to four variables. On this occasion, for presentation purpose, hourly averages are taken. Then, there is one sample per hour; 24 samples per day and 168 samples per week.

The information represented at each point is visually labelled from Fig. 1 to Fig. 9, which shows the time in 24h format and a weekday initial (*e.g.*, *6M means 6 am - Monday*). All data was normalized for the study.

The graphical results obtained in this study are presented and analyzed as follows.

Fig. 1. PCA - Projections of hourly pollution parameters.

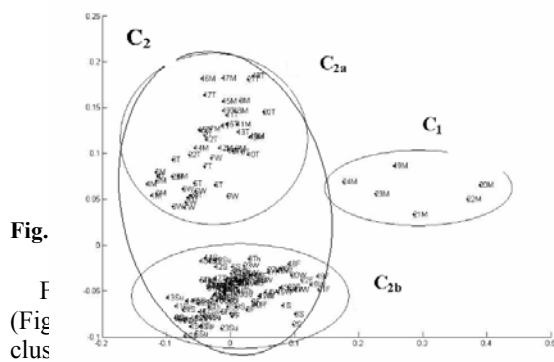
Fig. 1. PCA. This method identifies two main clusters (C_1 and C_2). Cluster C_2 groups most weekday samples and it is difficult to analyze its structure. C_1 is a group of scattered points that correspond in this case to Monday (M), which has the samples with the highest pollution throughout the entire week.

Fig. 2. MLHL - Projections of hourly pollution parameters.

Fig. 2. MLHL. This method provides improved results with respect to PCA, Fig. 1. Cluster C_1 represents the samples showing the highest pollution on Monday (M) evening, in the same way as was seen with PCA (Fig. 1). Cluster C_2 in Fig. 1 corresponds to clusters C_{2a} and C_{2b} in (Fig. 2). The samples for cluster C_{2a} have lower pollution levels than the samples for cluster C_1 . These samples belong to Tuesday (T) and Wednesday (W). Cluster C_{2b} is undoubtedly the one with the most samples. In this cluster, the points represent lower levels of pollution, corresponding to Thursday (Th), Friday (F), Saturday (S) and Sunday (Su). Thursday (Th) and Friday (F) were national bank holidays that correspond to Constitution Day and the Day of the Immaculate Virgin, respectively. On these days, industrial activity is greatly curtailed and many people traditionally travel long distances to visit relatives.

Fig. 3. CMLHL - Projections of hourly pollution parameters.

Fig. 3. CMLHL. The difference with MLHL, (Fig 2), is that CMLH is able to identify a new sub-cluster of samples. Cluster C_{2a} in (Fig. 2) corresponds to clusters C_{2a1} and C_{2a2} in (Fig. 3). Samples in C_{2a1} have higher levels of pollution than samples in C_{2a2} . Finally, Cluster C_{2b} contains the same samples as cluster C_{2b} in (Fig 2), but this time with sparser results. Applying CMLHL has made it possible to identify in a clearer and sparser way the internal structure based on pollution levels. C_{2b} contains the records with the lowest pollution levels recorded over the national bank holiday, Thursday (T), Friday (F), Saturday (S) and Sunday (Su).



lar to those obtained by applying MLHL, γ -clusters C_{2a} and C_{2b} are presented. The

Fig. 5. CCA - Projections of hourly pollution parameters.

Fig. 5. CCA. Once again, similar graphical results are obtained. CCA provides the same sample clusters as those that are obtained by applying MLHL and ISOMAP (Fig. 2, and Fig. 4).

4. Conclusions and future works

The aim of this study is to demonstrate the validity of different statistical and neural models for the analysis of air pollution variables over the week in a typical European city such as Burgos, using the information available on pollution. After applying nine different statistical and soft computing models to the data sets (only five results are shown here), it has been shown that a degree of internal structure can be identified. PCA provides an initial approximation to the internal structure of the data, but it does not offer interesting information on hourly pollution readings. Other soft computing methods applied are able to provide an improved response. MLHL, ISOMAP, and CCA provide a good response and very similar results among them. This good response is due to the fact that these methods offer a higher subdivision of the clusters; they group pollution values and classify them depending on the days of the week. CMLHL offers the best visual results, providing a very fine response. This is due to the fact that it is capable of differentiating new sub-clusters of data. Clearly, it was possible to fit each of the clusters to average hourly values of pollution. It is possible to affirm that the cluster with the lowest levels of pollution corresponds to the national bank holiday: Thursday (T), Friday (F), Saturday (S) and Sunday (Su).

In this study only pollutant variables are considered, but further clarification would be desirable on the important influence of atmospheric conditions on the evolution of atmospheric pollution. Future work will be based on the analysis of more complex data sets that combine pollution and meteorological data in the city of Burgos, using soft computing methods, in order to identify the relationship between pollution and meteorological conditions throughout the week and over different time periods.

Acknowledgments. This research has been partially supported through JCyL project BU035A08, and the Spanish Ministry of Education and Innovation project TIN2010-21272-C02-01 and CIT-020000-2009-12. The authors would also like to thank the vehicle interior manufacturer, Grupo Antolin Ingenieria, S.A., within the framework of the project MAGNO2008 - 1028.- CENIT Project funded by the Spanish Ministry.

References

1. R. San José, J. L. Pérez and R. M. Gonzalez, (2004), Air Quality real-Time Operational System for Large Industrial Plants: an application of MM5-CMAQ modelling system, Air pollution XII, pages: 245-253.
2. O. Cuesta-Santos, A. Collazo, A. Wallo, R. Labrador, M. Gonzalez and P. Ortiz, (2001), Deposition of atmospheric nitrogen compounds in humid tropical Cuba, The ScientificWorld e-journal (ISSN1532-2246), www.thescientificworld.com.

3. J. Sedano, L. Curiel, E. Corchado, E. de la Cal and J. R. Villar, A soft computing method for detecting lifetime building thermal insulation failures, April 2010, Integrated Computer-Aided Engineering, Volume 17 Issue 2 , 103-115 , IOS Press.
4. L. Lima, P. Novais, R. Costa, J. B. Cruz and J. Neves, Group decision making and Quality-of-Information in e-Health systems. Logic Jnl IGPL (2010) Oxford University Press doi:10.1093/jigpal/jzq029.
5. E. Corchado and A. Herrero, Neural visualization of network traffic data for intrusion detection, Applied Soft Computing, Elsevier, doi:10.1016/j.asoc.2010.07.002.
6. L.A. Zadeh, Fuzzy logic, neural networks, and soft computing, Communications of the ACM, volume 37 , issue 3 (March 1994) Pages: 77 – 84.
7. J. Sedano, E. Corchado, L. Curiel, J. R. Villar and P. M. Bravo, Modelling a Pneumatic Drill Process by a two-steps AI Model, International Journal of Computer Mathematics, Taylor & Francis, 2009.
8. J. H. Friedman and J. W. Tukey. 1974, A Projection Pursuit Algorithm for Exploratory Data-Analysis, IEEE Transactions on Computers 23 (9):881-890.
9. A. Hyvärinen, Complexity Pursuit: Separating interesting components from time series, Neural Computation 13 (2001), pp. 898-883.
10. A. Hyvärinen, J. Karhunen and E. Oja, Independent Component Analysis, Wiley, 2002.
11. P. Diaconis and D. Freedman. 1984, Asymptotics of Graphical Projection Pursuit, the Annals of Statistics, vol. 12, number 3 (1984), pp. 793-815.
12. E. Oja, Neural Networks, Principal Components and Subspaces, International Journal of Neural Systems, vol. 1 (1989), pp. 68-61.
13. E. Oja, H. Ogawa and J. Wangviwattana, Principal Components Analysis by Homogeneous Neural Networks, part 1, The Weighted Subspace Criterion, IEICE Transaction on Information and Systems E75D (1992), pp. 375-366.
14. C. Fyfe and R. Baddeley, Non-linear data structure extraction using simple Hebbian networks, Biological Cybernetics 72(6) (1995), pp. 541-533.
15. H. Hotelling, Analysis of a Complex of Statistical Variables into Principal Components, Journal of Education Psychology, 24:417-444. 1933.
16. S. Seung, N.D. Soccia and D. Lee, The Rectified Gaussian Distribution, Advances in Neural Information Processing Systems, 10 (1998), pp. 350-356.
17. L. Xu, Least Mean Square Error Reconstruction for Self-Organizing Nets, Neural Networks 6 (1993), pp. 648-627.
18. P. L. Lai, D. Charles and C. Fyfe, Seeking Independence using Biologically Inspired Artificial Neural Networks, in: Developments in Artificial Neural Network Theory: Independent Component Analysis and Blind Source Separation, M. A. Girolami (ed.), Springer Verlag, 2000.
19. J. Smola and B. Scholkopf, A Tutorial on Support Vector Regression, Technical Report NC2-TR-1998-030, NeuroCOLT2 Technical Report Series, 1998.
20. C. Fyfe and D. MacDonald, ϵ -Insensitive Hebbian learning, Neurocomputing 47(1-4) (2002), pp. 57-35.
21. E. Corchado and C. Fyfe, Connectionist Techniques for the Identification And Suppression Of Interfering Underlying Factors, IJPRAI 17(8): 1447-1466 (2003).
22. E. Corchado, D. MacDonald and C. Fyfe, Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit, Data Min. Knowl. Discov. 8(3) (2004), pp. 203-225.
23. R. L. Kashyap, C.C. Blaydon and K.S. Fu, Stochastic Approximation, in: A Prelude to Neural Networks, Adaptive and Learning Systems, Jerry M. Mendel (Ed), Prentice Hall, 1994.
24. E. Corchado, Y. Han and C. Fyfe, Structuring global responses of local filters using lateral connections, Journal of Experimental and Theoretical Artificial Intelligence 15(4): 473-487 (2003).
25. C. Fyfe and E. Corchado, Maximum Likelihood Hebbian Rules, Proc. of the 10th European Symposium on Artificial Neural Networks (ESANN 2002), (2002), pp. 143-148.
26. S. Roweis and K. Lawrence Nonlinear dimensionality reduction by locally linear embedding, Saul. Science, v.290 no.5500 , Dec.22, 2000. pp.2323--2326.
27. H. P. Chan, D. Y. Yeung and Y. Xiong, Super - resolution through neighbor embedding, In IEEE Computer Society Conference on computer vision and pattern recognition, volume 1, pages 275-282-, 2004.
28. P. Demartines and D. J. Héroult, Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets, IEEE Transactions on Neural Networks, 1997. 8(1): pp. 148-154.
29. J. W. Sammon, A nonlinear mapping for data structure analysis, IEEE Trans. On Computer, 18 (1969) 401-409.

- 30.V. Tricio, R. Vioria and A. Minguito, Ozone Measurements In Urban And Semi-Rural Sites At Burgos (Spain), IN Geophysical Research Abstracts, Volume 5, 2003. EGS-AGU-EUG Joint Assembly. EAE03-A-14249. ISSN: 1029-7006.
- 31.V. Tricio, R. Vioria and A. Minguito, (2007), Temporary evolution of ozone air quality in province of Burgos (Spain): regression and cluster techniques, Geophysical Research Abstracts, CD-ROM, (R) Volume 9. European Geosciences Union 2007. Sref-ID: 1607-7962/gra/EGU07-A-10951.
- 32.A. Arroyo, E. Corchado and V. Tricio, Atmospheric Pollution Analysis by Unsupervised Learning, Intelligent Data Engineering and Automated Learning - IDEAL 2009, pages: 767–772, 2009.
- 33.E. Corchado, A. Arroyo and V. Tricio, Soft computing models to identify typical meteorological days, Logic Journal of the IGPL, Oxford University Press July 21, 2010 doi:10.1093/jigpal/jzq035