SOME APPLICATIONS OF UNSUPERVISED NEURAL NETWORKS IN RATE MAKING PROCEDURE

RENATO PELESSONI
LIVIANA PICECH

1998 GENERAL INSURANCE CONVENTION
AND
ASTIN COLLOQUIUM

GLASGOW, SCOTLAND: 7-10 OCTOBER 1998

SOME APPLICATIONS OF UNSUPERVISED NEURAL NETWORKS IN RATE MAKING PROCEDURE[1]

Renato Pelessoni

Liviana Picech

Dipartimento di Matematica Applicata alle
Scienze Economiche Statistiche ed Attuariali "Bruno de Finetti"
Università degli Studi di Trieste
P.le Europa n.1
34127 TRIESTE – ITALY

Telephone: +39 040 6767062

Facsimile: +39 040 54209

e-mail: renatop@econ.univ.trieste.it - livianap@econ.univ.trieste.it

Summary

In recent years, neural networks have been having a wide range of applications. In particular, the unsupervised neural networks are designed to implement clustering techniques. In this paper we apply a two-stage Kohonen Self-Organising Map to collect the basic classes of one tariff variable in clusters. In this procedure we take advantage of the topology preservation property of the Self-Organising Maps in order to build tariff classes containing contiguous values of the tariff variable.

Keywords

Tariff classes, unsupervised neural networks, competitive learning, Kohonen self-organising maps.

[1] This research work was partially supported by C.N.R. (Research Project n. 97.01047.CT10115.21688: "Modelli e metodi per valutazioni in assicurazioni non-vita").

1. Introduction

In recent years Neural Networks (NN) have been having a wide spread of applications in many different fields: from signal and image processing in engineering to the exchange rates forecast and other financial applications in economics. Also the actuaries are taking interest into possible applications of NN in actuarial practice. J.A. Lowe and L.M. Pryor [16] have reported on the application of supervised NN in underwriting, since this type of NN is specifically designed to deal with models representing a set of information from which some sort of predictions are derived.

This paper is concerned with unsupervised neural networks. As the supervised NN are connected to statistical models for predictions, the unsupervised NN are connected to cluster analysis techniques. In [8] some types of unsupervised NN have been applied to collect the values (basic classes) of one tariff variable into tariff classes; it has been showed how these techniques allow the implementation of partitioning methods of cluster analysis, which produce appreciable results in the applications and in comparison with the traditional actuarial methods.

In this paper, we still deal with the clustering problem of collecting the basic classes of one tariff variable in clusters. In particular we investigate the possibility of taking advantage of a topological property of Kohonen Self-Organising Maps in order to build tariff classes containing contiguous values of the tariff variables.

An outline of the paper is the following.

In Section 2 we recall, following the discussion in [7], the methods proposed by H. Dickmann and by K. Loimaranta *et al.* to determine the tariff classes by means of a hierarchical clustering method and a non-hierarchical method of mixtures respectively.

In Section 3 we briefly describe two neural network algorithms frequently used in clustering problems: Simple Competitive Learning and Kohonen Self-Organising Map.

Sections 4 is devoted to an application of the algorithms described in Section 3 to collect in clusters the values of the tariff variable "age of the insured" in a motor vehicle insurance portfolio.

In Section 5 the topology preservation property of the Self-Organising Maps is exploited to collect the basic classes described by the age of the insured in a motor insurance portfolio in clusters formed by contiguous values of the tariff variable.

In Section 6 some final remarks and suggestions for further investigations are resumed.

2. Clustering methods proposed for the determination of tariff classes.

The basic classes can be seen as objects that have to be joined together according to the values of the characteristic variable; from this point of view the problem of determining the tariff classes can be seen as a clustering problem.

Since the observed values of the characteristic variable in each basic class arise from observations on risks with different exposures, these values are not immediately comparable by means of the similarity or dissimilarity measures considered in traditional clustering procedures. Therefore, in the actuarial literature some clustering techniques have been implemented in order to take account of the exposures of the

basic classes as well (see [7] for a review).
In particular, the method proposed by H. Dickmann is a hierarchical agglomerative clustering method in which, at the beginning, each basic class is viewed as a group containing one object and, at each stage, the merging of two groups is done if it minimises the increase of the total within-cluster variance. The procedure is repeated until all basic classes are located in one cluster.
For a short description of the algorithm, let us consider a single stage with the basic classes joined together in K clusters. Let

m_k be the number of basic classes located in cluster k;

x_{ik} be the observation of the characteristic variable with respect to the i-th basic class located in cluster k;

t_{ik} be the value which reflects the exposure of the i-th basic class located in cluster k (e.g. the number of observed policy-years);

$n_k = \sum_{i=1}^{m_k} t_{ik}$ be the total exposure in cluster k.

Define the within-cluster variance for cluster k as:

$$_k\sigma_W^2 = \sum_{i=1}^{m_k} \left(x_{ik} - \bar{x}_k\right)^2 \frac{t_{ik}}{n_k}$$

where

$$\bar{x}_k = \sum_{i=1}^{m_k} x_{ik} \frac{t_{ik}}{n_k}.$$

It is important to note how the definition of within-cluster variance allows to take account of the different exposures of the basic classes.
Then the total within-cluster variance with K clusters is defined as:

$$^{(K)}\sigma_W^2 = \sum_{k=1}^{K} {}_k\sigma_W^2 \frac{n_k}{n}$$

where $n = \sum_{k=1}^{K} n_k$

and we pass from K to K-1 clusters by merging two of the existing clusters so that the increase of the within-clusters variance $^{(K-1)}\sigma_W^2 - {}^{(K)}\sigma_W^2$ is minimum.
Another method has been proposed by K. Loimaranta, J. Jacobsson & H. Lonka and it consists in a non-hierarchical method of mixtures.
It is assumed that the N basic classes belong to K clusters and that the characteristic variables are independent random variables with probability distribution a mixture of K distributions, one for each cluster.
More precisely, let $X_1,\ldots,X_N$ be the random characteristic variables of the N basic classes and $x_1,\ldots,x_N$ their observations. $X_1,\ldots,X_N$ are assumed to be independent and X_i ($i=1,\ldots,N$) is distributed as:

$$P_{mix}\left(x;t_i,\vartheta^{(1)},...,\vartheta^{(K)}\right)=\sum_{k=1}^{K}p_k P\left(x;t_i,\vartheta^{(k)}\right)$$

where

t_i	is a value which reflects the exposure of the basic class i;
$\vartheta^{(1)},...,\vartheta^{(K)}$	are parameters to be estimated;
p_k	is the k-th weight in the mixture and can be seen as "a priori" probability that a basic class is located in cluster k; $\sum_{k=1}^{K} p_k = 1$;
$P\left(x;t_i,\vartheta^{(k)}\right)$	is the probability distribution of the characteristic variable conditionally to the belonging of the basic class i to the cluster k and dependent on the exposure t_i.

The posterior probability $P(k|x_i)$ for the i-th basic class to belong to the k-th cluster can be derived:

$$P(k|x_i)=\frac{p_k P\left(x_i;t_i,\vartheta^{(k)}\right)}{\sum_{h=1}^{K} p_h P\left(x_i;t_i,\vartheta^{(h)}\right)} \quad (k=1,...,K).$$

After assigning the "a priori" probabilities p_k $(k=1,...,K)$ and estimating the parameters $\vartheta^{(1)},...,\vartheta^{(K)}$ by the maximum likelihood method, the posterior probabilities $\hat{P}(k|x_i)$ ($k=1,...,K$; $i=1,...,N$) can be estimated. As long as the probability distribution $\hat{P}(k|x_i)$ $k=1,...,K$ is "sufficiently" concentrated on the value $\bar{k}$, then the i-th basic class will be clearly allocated in cluster $\bar{k}$.

It is interesting to note how, in this method, the different exposures have been dealt with by means of the probability distributions.

Another important class of cluster analysis techniques is known as partitioning methods (among which the well-known k-means algorithms). In these methods the number of the clusters K is fixed in advance or, in some variants, determined through the procedure. Moreover, unlike the hierarchical techniques, they allow the relocation of the objects. In this way, bad initial partitions can be improved. Most of these techniques consist of two distinct procedures:

- the determination of an initial allocation of the objects into the clusters;
- the relocation of some or all of the objects in the clusters.

An essential feature of these methods is the calculation of the centroids of the clusters. Many clustering algorithms have been proposed; among them those proposed by E.W. Forgy, by J.B. MacQueen and a variant of the latter method (see [1]) are reported in [8].

K. Loimaranta *et al.* ([15]) stated that, in their opinion, as far as the determination of tariff classes is concerned, a method that searches for the optimal partition could be preferred to a hierarchical clustering technique. However, partitioning methods face the

difficulty of considering the exposures of the basic classes.
In [8], the Authors discussed some techniques, in a NN framework, by which some partitioning algorithms can be implemented in a more flexible environment, allowing the exposures to be managed as well.

3. Self-Organising Map and Simple Competitive Learning.

In [12] T. Kohonen introduced an unsupervised technique to construct topology-preserving mappings from an input space into a low dimensional lattice (usually a one- or two-dimensional array of units). This algorithm is implemented by means of a neural network, called Self-Organising Map (SOM), whose vertices (units or neurons) are disposed into two layers. The first layer consists of the input units, the second one of the output units and there are connections linking each input unit with each output unit.
Let, in a general case, n be the number of input units and K be the number of output units. Each input unit represents a real number, so that the array of input units represents an input vector in $\Re^n$. Let $O=\{1,\dots,K\}$ be the set of the output units forming the lattice. To each connection between the input units and the output units a real number called weight is associated.
Let

- d be a distance in $\Re^n$;
- d' be a distance in the lattice;
- λ_t be a family of positive non-increasing real functions defined on $\Re^+$, where t is a real non-negative number and $\lambda_t(0)=1\ \forall t$;
- μ_{ji} be the weight corresponding to the connection between the input unit i and the output unit j.

Each output unit can be represented by the weight vector $m_j=(\mu_{j1},\dots,\mu_{jn})$.
The network is used to classify a set of data in clusters. We will suppose to have a set S (input space) of N input vectors (real vectors of $\Re^n$) denoted by x_h $(h=1,\dots,N)$, which have to be classified in clusters. If we present a vector of data $x\in S$ to the network, it can be compared with all the weight vectors. We call winner unit, $c=c(x)$, the output unit satisfying the condition

(3.1) $d(x,m_c)\le d(x,m_j)\ \forall j=1,\dots,K$.

The SOM algorithm ([10], [12], [13]) carries out a vector classifier according to the criterion (3.1). In order to minimise the number of misclassifications, the algorithm updates the weights of the network by means of a learning rule (step 4 in the following description of the algorithm).
More precisely, denoting by $m_j(t)$ and $x(t)$ the weight vectors at time t and the input vector presented at the same time respectively, the SOM algorithm consists of the

following steps.

Self-Organising Map algorithm

1. put $t = 0$ and initialise the vectors $m_j(0)$ $(j = 1,\ldots K)$;
2. choose an input vector $x(t) \in S$;
3. find the index c such that $d(x(t), m_c(t)) = \min_{j=1,\ldots,K} \{d(x(t), m_j(t))\}$;
4. update the weight vectors according to the rule
   $m_j(t+1) = m_j(t) + \alpha(t)\lambda_t(d'(c,j))(x(t) - m_j(t)) \quad \forall j \in O$, with $\alpha(t) > 0$;
5. stop if the stopping rule is satisfied;
   otherwise replace t with $t+1$, go back to step 2 and repeat for the next input vector.

Usually, the term $\alpha(t)$ (learning rate factor) is a positive non-increasing function of t and $\alpha(0)$ is chosen not too far from 1 (typically 0.8).
Also λ_t is a non-increasing function and, as a consequence, the weights of the units of the lattice close to the winner unit and those of the winner unit itself are changed significantly. On the other hand, weights of units placed further away from the winner unit are not updated appreciably.
At the end of the learning process the network is able to classify the input vectors: input vectors that make winner the same output unit belong to the same cluster and the corresponding weight vector can be chosen as "representative" of the cluster itself. Clearly, different runs of the algorithm can produce different results.
The choice of the functions λ_t is crucial for another important feature of the SOM algorithm, namely the property of topology preservation. Essentially, after the convergence of the algorithm, the input vectors that are close in the input space are assigned to clusters represented by output units which are close in the lattice (see [6] for a formal definition of this property). A well-known function, widely used in the applications, is the gaussian function:

$$\lambda_t(r) = exp\left(-\frac{r^2}{2\sigma^2(t)}\right)$$

where σ is a decreasing function and $\sigma(0)$ is large enough.
In the very specific case when

$$\lambda_t(r) = \begin{cases} 1 & \text{if } r = 0 \\ 0 & \text{if } r \neq 0 \end{cases}$$

that is if, at each time t, only the weight vector of the winner unit is updated, the so called Simple Competitive Learning (SCL) algorithm is implemented. It is important to note that in this case the topology preservation property is no more in force. A thorough discussion on SCL can be found in [10], where also the strong relationship between SCL and k-means is pointed out.
In the SCL algorithm, the training is continuous, since the weights are updated after the presentation of each pattern (see step 4 of the SOM algorithm). Nevertheless, there

exists also a batch version of the same algorithm, known as Linde-Buzo-Gray (LBG) algorithm of Vector Quantisation ([14]), where the weights are updated after all patterns have been presented.

The Vector Quantisation algorithm has been originally designed for encoding/decoding processes in data compression. An unified framework of this algorithm and of most of its variants can be found in [3]. As pointed out in [3] and [17], if we denote by p_h $(h = 1,\dots,N)$ a probability distribution over the input space S and the input vectors are selected according to this probability distribution, the LBG algorithm converges to a local minimum of the quantity (average distortion)

$$D = \sum_{h=1}^{N} d(x_h , m_{c(x_h)})^2 p_h .$$

Note that, from an essentially practical point of view, continuous training is frequently preferred to batch training, because the random presentation order of the input vectors can help to avoid poor local minima (see [9] at page 168).

In [8] it has been emphasised how a suitable definition of the probability distribution p_h $(h = 1,\dots,N)$ over the input space S of the basic classes allows to implement, by means of these algorithms, a partitioning method of cluster analysis which allows to take account of the different exposures of the basic classes. This can be done by defining

$$(3.2) \qquad p_h = \frac{t_h}{\sum_{h=1}^{N} t_h} \qquad (h = 1,\dots,N)$$

where t_h is the exposure of the basic class h.

It must be noted that, despite the extensive use of SOMs, the mathematical theory of Kohonen's algorithm is so far unsatisfactory. A review on main results can be found in [4]. See also [2] and [19] for a wide investigation of the connections between neural networks and pattern recognition.

4. An example

In this Section we present an application of the algorithms described in Section 3 to the data in Table 4.1, where the claim frequencies in a motor vehicle insurance portfolio are reported.

The basic classes are the ages of the policyholders and we want them to be collected in clusters according to their claim frequencies. We assume the relative exposures (3.2) as probability distribution on the basic classes and we apply both SCL and SOM algorithms.

In Figure 1, the claim frequencies and the relative exposures of the basic classes are depicted.

Table 4.1: Policy-years (exposure) and relative and absolute claim frequencies in automobile insurance for different policyholder's ages.

AGE	No. OF CLAIMS	EXPOSURE	CLAIM FREQUENCY	AGE	No. OF CLAIMS	EXPOSURE	CLAIM FREQUENCY
18	23	91,01	0,252706	57	219	2073,21	0,105633
19	113	593,24	0,190480	58	187	1746,44	0,107075
20	263	1266,33	0,207686	59	174	1714,64	0,101479
21	306	1939,81	0,157748	60	168	1637,98	0,102565
22	376	2156,65	0,174345	61	132	1498,30	0,088100
23	362	2566,41	0,141053	62	146	1450,78	0,100636
24	391	2724,07	0,143535	63	133	1442,04	0,092230
25	365	2832,83	0,128847	64	111	1390,82	0,079809
26	384	2974,93	0,129078	65	122	1329,39	0,091771
27	339	3132,37	0,108225	66	107	1135,92	0,094197
28	343	3177,86	0,107934	67	98	1035,29	0,094660
29	334	3311,80	0,100851	68	89	990,80	0,089827
30	327	3431,88	0,095283	69	80	922,72	0,086700
31	307	3418,68	0,089801	70	82	838,55	0,097787
32	302	3317,40	0,091035	71	82	787,63	0,104110
33	280	3087,49	0,090688	72	64	690,74	0,092655
34	300	3168,45	0,094683	73	60	590,56	0,101599
35	241	3016,52	0,079893	74	57	539,82	0,105591
36	254	2968,79	0,085557	75	44	434,87	0,101179
37	244	2860,14	0,085311	76	24	234,06	0,102537
38	225	2794,10	0,080527	77	15	177,68	0,084421
39	235	2831,11	0,083006	78	15	150,75	0,099502
40	229	2727,02	0,083974	79	16	171,77	0,093149
41	227	2819,13	0,080521	80	15	160,90	0,093229
42	245	2772,79	0,088359	81	10	122,52	0,081620
43	207	2582,16	0,080166	82	14	89,25	0,156859
44	263	2605,99	0,100921	83	8	66,34	0,120589
45	291	2737,60	0,106297	84	6	57,26	0,104794
46	247	2660,50	0,092840	85	1	44,51	0,022465
47	272	2764,65	0,098385	86	5	21,96	0,227728
48	264	2656,79	0,099368	87	3	22,47	0,133523
49	309	2728,08	0,113267	88	0	12,52	0,000000
50	199	2099,79	0,094771	89	1	12,71	0,078647
51	211	2103,36	0,100316	90	1	11,26	0,088826
52	250	2171,34	0,115136	91	0	7,00	0,000000
53	222	2068,38	0,107331	92	1	4,84	0,206782
54	215	2056,44	0,104550	93	0	2,90	0,000000
55	230	2221,64	0,103527	94	0	6,86	0,000000
56	203	2183,59	0,092966	95	3	33,14	0,090528

(Data provided by an Italian Insurance Company)

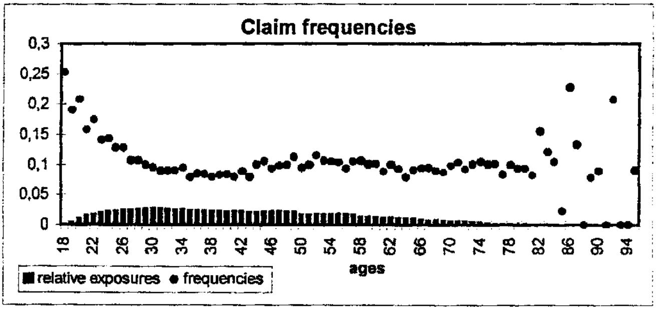

Figure 1

To perform the experiments, we used Matlab version 4.2c.1 and the Neural Network Toolbox version 2.0b ([5]). For this purpose we had to modify the programs provided in the Toolbox. Among the main modifications, we mention here the implementation, in the SCL procedure, of the following recursive formula to assign an individual learning rate to each weight vector

$$\alpha_j(t+1)=\begin{cases}\dfrac{\alpha_j(t)}{1+\alpha_j(t)} & \text{if } j=c\\ \alpha_j(t) & \text{if } j\neq c\end{cases}\qquad (j=1,\ldots,K)$$

where $\alpha(t)$ is the learning rate at time t and c is the winner unit at the same time; in this way, in every training cycle, only the learning rate corresponding to the winner unit c is updated. A discussion on this topic and on the choice of an "optimal" learning rate can be found in [13].

Other substantial modifications of the original SOM procedure are the provision of the gaussian function and of the following formulas for α and σ, as suggested by H. Ritter and K. Schulten (see [9] at page 114)

$$\alpha(t)=\alpha_0\left(\frac{\alpha_{t_{max}}}{\alpha_0}\right)^{\frac{t}{t_{max}}}\qquad\qquad \sigma(t)=\sigma_0\left(\frac{\sigma_{t_{max}}}{\sigma_0}\right)^{\frac{t}{t_{max}}}$$

where t_{max} is the maximum value for t (fixed in advance) and α_0, $\alpha_{t_{max}}$, σ_0, $\sigma_{t_{max}}$ are the fixed initial and final values of α and σ respectively.

In all the experiments proposed here, both the distances d and d' are Euclidean. The initial values of the weight vectors were assigned by means of the "random guess method", that is by choosing them randomly in the "right" domain, according to the values of the input vectors. Besides, as suggested by several authors, the algorithms were stopped after a quite large number of iterations.

Analogously to the traditional partitioning methods of cluster analysis (e.g. k-means)

we obtain different partitions of the basic classes depending on the stated number of clusters (output units) K. Therefore, a problem arises: a criterion to decide how many clusters should be considered.
In [18] it has been applied for this purpose, the method proposed by H. Schmitter and E. Straub ([20]) (S-S method) to find the "best" subdivision of an insurance portfolio in tariff classes. They assumed the existence of a "natural subdivision" and derived two statistics to single out this subdivision, or possibly the "closest" one from a set of "admissible subdivisions" (the "admissible subdivisions" are a subset of all the subdivisions of the portfolio, which can be actually considered for practical and commercial reasons).
From the two statistics a practical decision rule is derived but, to be applied, observations of the characteristic variables of the basic classes over a certain number of years are required.
Since we have observations over one year only, following [7] we calculate for each of our "admissible subdivisions" g $(g = 1,\dots, L)$ the following statistics:

$$W^{(g)} = \frac{1}{K_g - 1} \sum_{k=1}^{K_g} \frac{t_{\cdot k}^{(g)}}{t} \left(X_{\cdot k}^{(g)} - X\right)^2 , \quad T^{(g)} = \left(K_g - 1\right) W^{(g)}$$

where

K_g $(g = 1,\dots,L)$	is the number of clusters of the g-th subdivision,
$t_{\cdot k}^{(g)}$	is the total exposure of the k-th cluster of the g-th subdivision,
$t = \sum_{k=1}^{K} t_{\cdot k}^{(g)}$	is the total exposure,
$X_{\cdot k}^{(g)}$	is the mean of the characteristic variable of the basic classes located in the k-th cluster of the g-th subdivision, weighted with their exposures,

and

$$X = \frac{\sum_{k=1}^{K} X_{\cdot k}^{(g)} \, t_{\cdot k}^{(g)}}{t}$$

The practical decision rule, as reported in [7], is:

> choose the subdivision g that shows the highest value of $W^{(g)}$ among those with the highest $T^{(g)}$ values.

However, the subdivision with the highest value of $T^{(g)}$ will be discarded if another subdivision with a slightly lower value of $T^{(g)}$ and with a higher value of $W^{(g)}$ can be formed joining some clusters of the former subdivision.
In [7] it is also noted that, although the method is clear and valuable, since a good subdivision of the basic classes in clusters should reflect the heterogeneity of the

portfolio, the decision rule cannot ensure to find the natural subdivision. In fact it could not belong to the family of "admissible subdivisions" which are tested. Moreover, since in practical situations the boundary among the clusters may be rather vague, it could not be identified by the decision rule.
In Table 4.3 we report the best results obtained by means of the SCL algorithm in several trials carried out with different numbers of output units and various initial learning rates.
To simplify the description of the obtained clusters, the basic classes and the claim frequencies, ordered by the latter, are reported in Table 4.2
The subdivisions reported in Table 4.3 (and in the following analogous tables) refer to the order in the data: e.g. (5 3 6 26 33 5) characterises the subdivision where the first cluster contains the first five elements in Table 4.2 (ages: 18, 86, 20, 92 and 19), the second cluster contains the following three elements (ages: 22, 21 and 82), etc.

Table 4.2: Policyholder's ages and relative claim frequencies (ordered by claim frequencies).

AGE	CLAIM FREQUENCY	AGE	CLAIM FREQUENCY	AGE	CLAIM FREQUENCY	AGE	CLAIM FREQUENCY
18	0,252706	45	0,106297	30	0,095283	69	0,086700
86	0,227728	57	0,105633	50	0,094771	36	0,085557
20	0,207686	74	0,105591	34	0,094683	37	0,085311
92	0,206782	84	0,104794	67	0,094660	77	0,084421
19	0,190480	54	0,104550	66	0,094197	40	0,083974
22	0,174345	71	0,104110	80	0,093229	39	0,083006
21	0,157748	55	0,103527	79	0,093149	81	0,081620
82	0,156859	60	0,102565	56	0,092966	38	0,080527
24	0,143535	76	0,102537	46	0,092840	41	0,080521
23	0,141053	73	0,101599	72	0,092655	43	0,080166
87	0,133523	59	0,101479	63	0,092230	35	0,079893
26	0,129078	75	0,101179	65	0,091771	64	0,079809
25	0,128847	44	0,100921	32	0,091035	89	0,078647
83	0,120589	29	0,100851	33	0,090688	85	0,022465
52	0,115136	62	0,100636	95	0,090528	88	0,000000
49	0,113267	51	0,100316	68	0,089827	91	0,000000
27	0,108225	78	0,099502	31	0,089801	93	0,000000
28	0,107934	48	0,099368	90	0,088826	94	0,000000
53	0,107331	47	0,098385	42	0,088359		
58	0,107075	70	0,097787	61	0,088100		

Looking at Table 4.3 we note that the subdivisions in 7 and respectively 8 clusters show quite comparable values of T, whereas the T value of the subdivision in 6 clusters is sensibly lower. Therefore, according to the S-S criterion, the subdivision in 7 clusters seems to be the best one (see the corresponding value of W). The details on this subdivision are reported in Table 4.4. We note that the weights of the neurons and the centroids of the clusters are approximately equal (except those in the 7^{th} cluster).

We can deduce that the algorithm has converged to a possibly local minimum of the average distortion and the remarkable difference between weight and centroid of the 7th cluster can be explained by the very low exposures of the basic classes belonging to this cluster (approx. 74 policy-years).

Table 4.3: Best subdivisions in clusters obtained by SCL.

No. of clusters	Clusters	$W \times 10^{-5}$	$T \times 10^{-4}$	$D \times 10^{-5}$
8	4 1 3 5 16 26 18 5	7.4694	5.2285	1.9423
7	5 3 6 24 22 13 5	8.7101	5.2261	1.9692
6	5 3 6 26 33 5	10.2257	5.1129	3.1042

Table 4.4: Details on a subdivision in 7 clusters obtained by SCL.

7 clusters - W=8.7101 x 10^{-5} T=5.2261 x 10^{-4}					
Cluster	No. of elements	Policyholder's ages	Weights	Centroids	Exposures
1	5	18 19 20 86 92	0.2048	0.2048	1977.38
2	3	21 22 82	0.1662	0.1663	4185.70
3	6	23-26 83 87	0.1351	0.1353	11187.06
4	24	27-29 44 45 48 49 51-55 57-60 62 71 73-76 78 84	0.1050	0.1051	42389.66
5	22	30-34 42 46 47 50 56 61 63 65-68 70 72 79 80 90 95	0.0928	0.0927	38243.33
6	13	35-41 43 64 69 77 81 89	0.0824	0.0824	25225.41
7	5	85 88 91 93 94	0.0205	0.0136	73.78

In Table 4.5 the best results obtained by means of a Kohonen SOM are reported. A Kohonen network with a one-dimensional array of output units has been considered. The parameters of the Ritter and Schulten formulas have been set to the values $\alpha_0 = 0.8$, $\alpha_{t_{max}} = 0.01$, $\sigma_0 = 0.75$, $\sigma_{t_{max}} = 0.25$.

Table 4.5: Best subdivisions in clusters obtained by SOM.

No. of clusters	Clusters	$W \times 10^{-5}$	$T \times 10^{-4}$	$D \times 10^{-5}$
8	6 4 4 9 17 17 9 12	7.4610	5.2227	2.0020
7	6 4 4 9 17 20 18	8.6809	5.2085	2.1489
6	6 7 10 17 20 18	10.2531	5.1265	2.9643

We note that the T values are quite close to those obtained by SCL (Table 4.3) and, except the subdivision into 6 clusters, the SCL clusters seem preferable (compare also the distortion *D*). Following S-S method the partition in 7 clusters should be chosen as in the SCL case. The details on this subdivision are reported in Table 4.6.

Table 4.6: Details on a subdivision in seven clusters obtained by SOM.

7 clusters - W=8.6809 x 10^{-5} T=5.2085 x 10^{-4}					
Cluster	No. of elements	Policyholder's ages	Weights	Centroids	Exposures
1	6	18 19 20 22 86 92	0.1880	0.1889	4134.02
2	4	21 23 24 82	0.1464	0.1466	7319.55
3	4	25 26 83 87	0.1288	0.1289	5896.57
4	9	27 28 45 49 52 53 57 58 74	0.1088	0.1088	20375.10
5	17	29 44 47 48 51 54 55 59 60 62 70 71 73 75 76 78 84	0.1012	0.1011	25617.76
6	20	30-34 42 46 50 56 61 63 65-68 72 79 80 90 95	0.0922	0.0921	34640.12
7	18	35-41 43 64 69 77 81 85 88 89 91 93 94	0.0827	0.0822	25299.19

It is interesting to note that the resulting clusters are rather different from those obtained by SCL. In particular, the small group (7^{th} cluster in Table 4.4) containing 5 basic classes characterised by very low claim frequencies, which emerged through the SCL algorithm, has not been isolated by applying SOM. As a consequence, the SCL clusters are more differentiated than the SOM ones and, in fact, the distortion is lower in the former case. Therefore, if we want to get a subdivision with low distortion, the SCL clusters should be preferred.

However, if we look at the resulting groups of basic classes (Table 4.4), we realise that this subdivision could be unsatisfactory for actual rate making purposes.

In particular, the basic classes are not contiguously grouped. Moreover, we observe that basic classes characterised by low exposures are anyhow classified according to their claim frequencies. For instance, the basic class "age 85" is classified in the 7^{th} cluster, whereas the basic classes "age 84" and "age 86" are classified in the 4^{th} and in the 1^{st} cluster respectively, since these basic classes show very different claim experiences (see Table 4.1), even though their exposures are very low.

This inconvenience could be avoided if, when grouping the basic classes, the information "age of the insurer" would be considered as substantial information and not only as a label attached to the basic classes just to identify them. An example of a procedure where values labelling data are directly employed as source of information in clustering can be found in [11].

5. Clustering under a constraint of contiguous grouping

A way to take account of the actual value "age of the insured", in addition to the observed claim frequency, is to apply the clustering techniques to the objects (basic classes) described by two characteristic variables: the claim frequency and the age of the insured. Therefore, since the objects are described by $\Re^2$ vectors, a suitable

distance in $\Re^2$ should be considered. Clearly, it determines the strength of the information "age" with respect to the observed claim frequency. In addition, this distance should have the appreciable property of enforcing this strength when the exposure (that is to say the number of observations) of the basic class is very low.
In this Section we follow a different approach and develop a procedure in which the clustering of the basic classes "age of the insured" is performed under a sort of constraint of contiguous grouping. More precisely, it consists of a successive application of Kohonen SOMs in which the property of topology preservation plays the substantial role of inducing the contiguous grouping in a natural way, even though it is *not generally granted.*
The procedure develops in two stages in which the basic classes are actually described by the two characteristic variables "age of the insured" and "claim frequency".

I Stage

In the first stage two parallel SOMs are trained: one concerns, as objects to be collected in clusters, the "age of the insured" relative to the basic classes and the other the "claim frequency". As a result we obtain classes of "ages" and classes of "claim frequencies". Thanks to the topology preservation property both the classes of ages and the classes of claim frequencies are ordered by age and by claim frequency respectively.

II Stage

In the second stage the outputs of the first stage become the input of another SOM.
To each initial basic class (described by the two characteristic variables "age of the insured" and "claim frequency") the corresponding indexes of age class and of claim-frequency class are associated. In this way, the new basic classes are now described by two characteristic variables, the index of age class and the index of claim frequency class, and these objects form the input space of another SOM. More precisely, the input space is now a subset of $\Re^2$ whose elements are the couples (index of age class, index of claim frequency class) to which at least one initial basic class has been associated. Since both the characteristic variables are indexes of clusters resulting from the first stage, the usual Euclidean distance can be considered. Moreover, we assume, for each couple of indexes, the total amount of the relative exposures of the initial basic classes associated to such couple as probability distribution over the input space.

In the following, we report the results of an application of this procedure to the data in Table 4.1. We have used the SOM algorithm implemented in Matlab and the parameters of the Ritter and Schulten formulas have been set to the values $\alpha_0 = 1.5$, $\alpha_{t_{max}} = 0.1$, $\sigma_0 = 2$, $\sigma_{t_{max}} = 0.5$.

In the First Stage one SOM is trained to collect the objects "age of the insured" into 20 clusters. The input space is the set of the ages in Table 4.1 and the probability distribution is defined as the relative exposures (3.2). Therefore, no evidence is given to the claim frequency, but only the actual age values are considered. The resulting clusters are reported in Table 5.1. It has to be stressed that the topology preservation property of the SOM algorithm makes the indexes of the clusters ordered according to the order on the input space. Moreover, the relative exposures of the resulting clusters are quite flat.

At the same stage another SOM is trained to collect the objects "claim frequencies" into 9 clusters. The input space is the set of claim-frequency values from Table 4.1 and the probability distribution is defined, again, by means of the relative exposures (3.2). The resulting clusters are reported in Table 5.2 and their indexes are ordered in accordance to the claim-frequency values, owing to the topological property of the SOM algorithm (cf. Table 4.2). In Table 5.2 are reported the labels "age of the insured" in order to identify the objects collected in the same cluster.

Table 5.1: I stage – Clusters of ages.

Age index	Ages	Relative exposures
1	18 - 22	0.049050
2	23 - 25	0.065892
3	26 - 27	0.049539
4	28 - 29	0.052641
5	30 - 31	0.055568
6	32 - 33	0.051953
7	34 - 35	0.050169
8	36 - 38	0.069945
9	39 - 40	0.045085
10	41 - 42	0.045359
11	43 - 45	0.064289
12	46 - 47	0.044006
13	48 - 50	0.060712
14	51 - 53	0.051452
15	54 - 56	0.052414
16	57 - 59	0.044891
17	60 - 62	0.037208
18	63 - 66	0.042976
19	67 - 71	0.037110
20	72 -	0.029742

Table 5.2: I stage – Clusters of claim frequencies.

Claim-frequency index	Ages	Centroids	Relative exposures
1	35-41 43 64 77 81 85 88 89 91 93 94	0.082046	0.197729
2	31-33 42 61 68 69 90 95	0.089643	0.130210
3	30 34 46 50 56 63 65-67 72 79 80	0.093848	0.158257
4	29 44 47 48 51 59 60 62 70 73 75 76 78	0.100367	0.166243
5	27 28 45 53-55 57 58 71 74 84	0.106366	0.167085
6	49 52 83	0.114182	0.040280
7	23-26 87	0.135333	0.090205
8	21 82	0.157709	0.016459
9	18-20 22 86 92	0.188920	0.033533

In the Second Stage one SOM is trained to collect objects described by the couple of indexes (age index, claim-frequency index) into 7 clusters. The elements of the input space are represented in Figure 2.

The results are reported in Table 5.3, where we note that the final groups contain age values which are actually contiguous.

To appreciate the features of the resulting groups we compare in Figure 3 the original claim frequencies with the centroids of the clusters.

Incidentally, observe that young drivers show a quite high risk level and in fact the claim frequency progressively decreases in clusters 2, 3 and 4, whereas clusters 5 and 6 show a higher risk level again. This is a well-known phenomenon present in the Italian market and it is explained by the fact that in the age classes of insured 44-59 we find the insured whose young sons or daughters get their driving licence and begin to drive their parents'car.

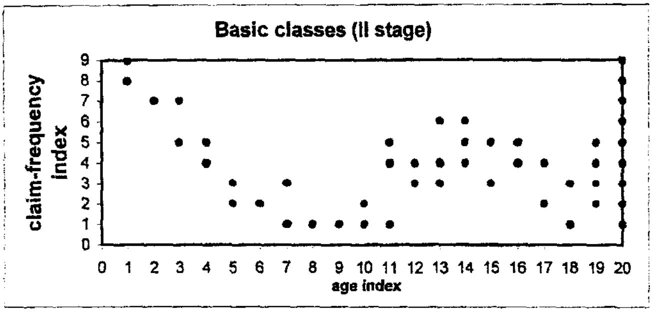

Figure 2

Table 5.3: II stage – Clusters of ages of the insured.

Cluster	Ages	Claim-frequency centroids	Relative exposure
1	18 - 26	0.1507	0.139073
2	27 - 29	0.1056	0.078049
3	30 - 35	0.0904	0.157690
4	36 - 43	0.0835	0.181334
5	44 - 50	0.1011	0.148062
6	51 - 59	0.1042	0.148756
7	60 -	0.0945	0.147036

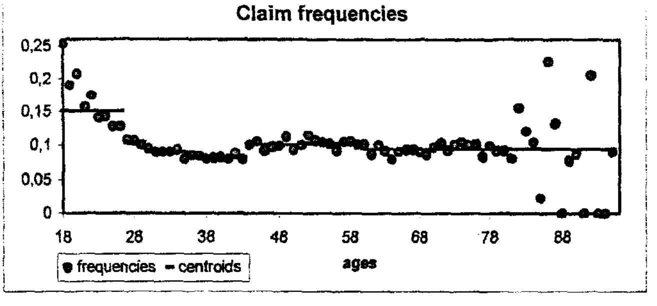

Figure 3

As far as the goodness of the clustering is concerned, the distortion and therefore also the S-S method are no more acceptable criteria to choose among different groupings. In fact, in this application the distortion calculated from the centroids of the clusters is 1.2026×10^{-4}, an extremely high level when compared with the results in Section 4. Clearly, the continuity of the elements in the groups is a valuable result but it cannot be evaluated by means of these traditional measures.

6. Closing remarks

In this paper we were concerned with the problem of determining the tariff classes by means of some unsupervised neural networks. In particular the SCL algorithm and the SOM algorithm have been applied to collect the age of the insured in clusters. The results have shown that these methods do not consider the information "age" itself. In order to get groups of basic classes formed by contiguous values, a two-stage Kohonen SOM algorithm has been applied.

In the example reported in Section 5, a suitable choice of the parameters has produced interesting results thanks to the topology preservation property of SOM. In fact, the final clusters contain contiguous values of the considered tariff variable, as it is desired and often pursued in the actuarial practice.

However, since such type of results is not generally granted, the role played by the various parameters seems worthwhile of further investigations.

References

[1] Anderberg M. R., *Cluster Analysis for applications*, Academic Press, New York, N.Y., 1973.

[2] Bishop C.M., *Neural networks for pattern recognition*, Clarendon Press, 1995.

[3] Black J.V., "A Unified Framework for Vector Quantisers", *Defence Research Agency Malvern Memorandum 4670*, 1992.

[4] Cottrell M., J. C. Fort, G. Pagès, "Two or three things that we know about the Kohonen algorithm", *Proceedings of ESANN'94*, 235-244, Bruxelles, 1994.

[5] Demuth H., M. Beale, *Neural Network Toolbox For Use with MATLAB - User's Guide*, The Mathworks Inc., 1994.

[6] Der R., M. Herrmann, T.M. Martinetz, T. Villmann, "Topology preservation in self-organizing feature maps: exact definition and measurement", *IEEE Transactions on Neural Networks*, 8 No.2, 256-266, 1997.

[7] van Eeghen J., E. K. Greup, J. A. Nijssen, "Rate making", *Surveys of Actuarial Studies*, 2, Nationale-Nederlanden N.V., 1983.

[8] Giulini S., R. Pelessoni, L. Picech, "Determination of tariff classes: cluster analysis methods and unsupervised neural networks", *Proceedings of the XXVIII International Astin Colloquium*, 129-150, Cairns (Australia), 1997.

[9] Hassoun M.H., *Fundamentals of artificial neural networks*, The MIT Press, 1995.

[10] Hertz J., A. Krogh, R. G. Palmer, *Introduction to the theory of neural*

computation, Addison-Wesley Publishing Company, 1991.

[11] Jain A.K., F. Farrokhnia, "Unsupervised Texture Segmentation Using Gabor Filters", *Patter Recognition*, 24 No.12, 1167-1186, 1991.

[12] Kohonen T., *Self-Organization and Associative Memory*, Springer, 1984.

[13] Kohonen T., *Self-Organizing Maps*, Springer, 1995.

[14] Linde Y., A. Buzo, R. Gray, "An Algorithm for Vector Quantizer Design", *IEEE Transactions on Communications*, COM-28, 84-99, 1980.

[15] Loimaranta K., J. Jacobsson, H. Lonka, "On the use of mixture models in clustering multivariate frequency data", *Transactions of the 21st International Congress of Actuaries*, 2, 147-161, 1980.

[16] Lowe J.A., L.M. Pryor, "Neural Networks v. GLMs in pricing general insurance", *General Insurance Convention*, 2, 417-438, 1996.

[17] Luttrell S.P., "Derivation of a Class of Training Algorithms", *IEEE Transactions on Neural Networks*, 1 No.2, 229-232, 1990.

[18] Pelessoni R., L. Picech, "Simple Competitive Learning, Kohonen S.O.M. and Schmitter-Straub's Method: a Proposal in Finding "Good" Tariff Classes", *Atti della Giornata di Studio su "Aspetti Scientifici e Didattici nella Teoria del Rischio"* (to appear), Campobasso, 1997.

[19] Ripley B.D., *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, 1996.

[20] Schmitter H., E. Straub, "How to find the right subdivision into tariff classes", *Astin Bulletin*, 8 No.2, 257-263, 1975.