



ELSEVIER

European Journal of Operational Research 114 (1999) 346–353

EUROPEAN  
JOURNAL  
OF OPERATIONAL  
RESEARCH

Theory and Methodology

# Comparing performance of feedforward neural nets and K-means for cluster-based market segmentation

Harald Hruschka<sup>a,\*</sup>, Martin Natter<sup>b</sup>

<sup>a</sup> *Department of Marketing, University of Regensburg, Universitätsstraße 31, D-93053 Regensburg, Germany*

<sup>b</sup> *Department of Industrial Information Processing, University of Economics, A-1200 Vienna, Austria*

Received 12 June 1997; accepted 28 April 1998

---

## Abstract

We compare the performance of a specifically designed feedforward artificial neural network with one layer of hidden units to the K-means clustering technique in solving the problem of cluster-based market segmentation. The data set analyzed consists of usages of brands (product category: household cleaners) in different usage situations. The proposed feedforward neural network model results in a two segment solution that is confirmed by appropriate tests. On the other hand, the K-means algorithm fails in discovering any somewhat stronger cluster structure. Classification of respondents on the basis of external criteria is better for the neural network solution. We also demonstrate the managerial interpretability of the network results. © 1999 Elsevier Science B.V. All rights reserved.

*Keywords:* Neural networks; Marketing; K-means; Cluster analysis; Market segmentation

---

## 1. Introduction

The problem of cluster-based or post hoc market segmentation consists of determining segments by partitioning buyers according to their similarities across several selected (behavioral, psychographic or socio-demographic) segmentation criteria (Green, 1971; Wind, 1978). The number of segments (clusters), their size and description are not known before completing the analysis.

We compare the performance of two approaches to cluster analysis using a real life data set.

1. K-means which is one of the most widespread algorithms, especially in marketing research (Green and Krieger, 1995).
2. A specifically designed feedforward artificial neural network with one layer of hidden units.

Sketching the relevant literature shows that many artificial neural networks may be seen as alternatives or extensions of somewhat more traditional data-analytic methods for regression, discriminant analysis, clustering or data compression (Hertz et al., 1991; Cheng and Titterington, 1994; Bishop, 1995; Haykin, 1994; Ripley, 1996).

---

\* Corresponding author.

Although the main problem category of feedforward networks is supervised learning (i.e. problems with dependent and independent variables), such networks can also be used for non-supervised learning (i.e. clustering and data reduction problems), if they are specified in an appropriate manner.

There are a few publications which compare artificial neural networks to the K-means algorithm. Balakrishnan et al. (1994) study self-organizing maps introduced by Kohonen (1984). Their main result is that self-organizing maps perform significantly worse than K-means when applied to simulated data. In another paper Balakrishnan et al. (1996) deal with the frequency-sensitive competitive learning algorithm of Krishnamurthi et al. (1990). Though this artificial neural net did not perform better than K-means, the authors finally recommend to combine both approaches.

**2. Clustering methods used**

Both clustering methods used in our study try to minimize the square-error objective  $E$  for a fixed number of segments (clusters):

$$E = \sum_p \sum_o (\hat{y}_{op} - y_{op})^2. \tag{1}$$

This objective equals the sum of quadratic differences between the theoretical value  $\hat{y}_{op}$  according to a cluster analysis model and the observed value  $y_{op}$  of each segmentation criterion  $o$  for each person  $p$ . For example, the theoretical value for K-means is the average value of the segmentation criterion in the cluster to which person  $p$  is assigned.

*2.1. The artificial neural network*

The artificial neural network model is a feedforward neural network using segmentation criteria both as input variables (units) and output variables (units). Between input and output we put a layer of hidden units whose values can be interpreted as membership values of a person for different segments. The networks are fully connected, i.e. each input variable is linked to every hidden unit, each hidden unit to every output unit (see Fig. 1).

Using segmentation criteria  $y_{op}, o = 1, O$  of person  $p$  as inputs the membership value  $s_{jp}$  with regard to segment  $j$  is computed by means of a multinomial logit function which is usually called softmax in the artificial neural network literature (Bridle, 1990):

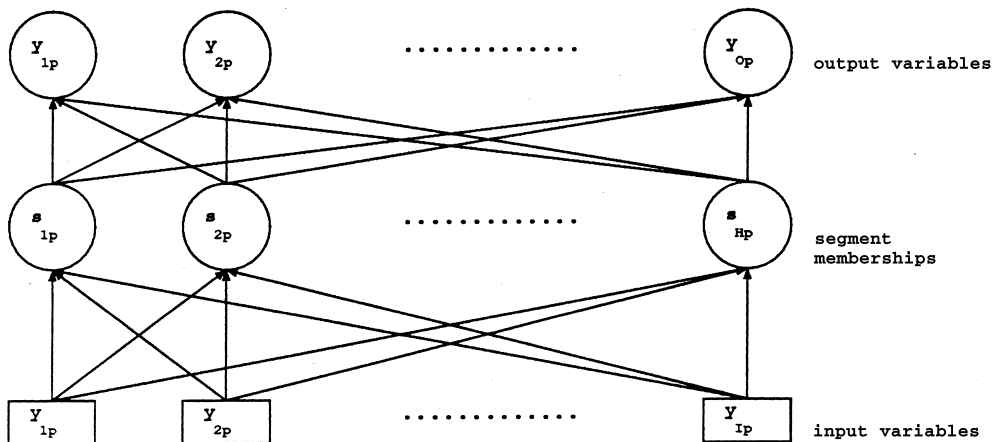


Fig. 1. Feedforward neural network for clustering.

$$s_{jp} = \frac{\exp(\sum_o \alpha_{oj} y_{op})}{\sum_h \exp(\sum_o \alpha_{oh} y_{op})}. \tag{2}$$

The multinomial logit formulation guarantees that membership values of any person lie between zero and one and sum to one:

$$0 < s_{hp} < 1, \quad h = 1, H, \quad p = 1, P, \\ \sum_h s_{hp} = 1, \quad p = 1, P.$$

The weights  $\alpha_{oh}$  measure the importance of a segmentation criterion with regard to membership in segment  $h$ . High positive (negative) values of these weights indicate that the  $o$ th segmentation criterion is associated with high (low) probability of membership in segment  $h$ .

In the output layer of the network model theoretical values of segmentation criterion  $o$  for respondent  $p$  are calculated in the following way:

$$\hat{y}_{op} = 1 / \left( 1 + \exp \left( - \sum_h \beta_{ho} s_{hp} \right) \right). \tag{3}$$

Segment memberships  $s_{hp}$  are weighted by criterion-specific weights  $\beta_{ho}$ . The sum of this weighted memberships over all segments transformed by a binomial logit function gives the theoretical value of segmentation criterion  $o$  for respondent  $p$ . High positive (negative) values of the  $\beta_{ho}$  show that membership to segment  $h$  goes with high (low) probability for segmentation criterion  $o$ .

We use a variant of backpropagation which is the most popular method to determine parameters (weights) in feedforward networks (Rumelhart et al., 1986; Haykin, 1994; Ripley, 1996). In each of several iterations adjustment of weights starts with the output units. Errors between actual and estimated output values are propagated layerwise backwards. Backpropagation tries to minimize the error measure  $E$  of Eq. (1).

The backpropagation algorithm runs for a number of iterations  $t = 1, 2, \dots$  each with a forward and a backward pass. For a network with parameters  $w_{ij}$  the first partial derivatives of the error measure  $E$  can be written as:

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial x_j} \frac{\partial x_j}{\partial w_{ij}} = z_i \frac{\partial E}{\partial x_j} = z_i f'_j(x_j) \frac{\partial E}{\partial z_j} = z_i \delta_j, \\ \delta_j = f'_j(x_j) \frac{\partial E}{\partial z_j}, \tag{4}$$

where  $x_j$  is the total input to unit  $j$  given by the weighted sum of individual inputs  $\sum_i w_{ij} z_i$ .  $z_j$  denotes unit's  $j$  output after transformation of  $x_j$  by function  $f_j$ .

For output units  $\partial E / \partial z_j$  can be calculated directly starting with Eq. (1). For network models with binomial logit functions to compute segmentation criteria we arrive at the following expression for  $\delta_j$  which we call  $\delta_{y_{op}}$  for better identification:

$$\delta_{y_{op}} = \hat{y}_{op}(1 - \hat{y}_{op})(\hat{y}_{op} - y_{op}). \tag{5}$$

The following expressions for  $\delta_j$  are valid for units in hidden layers (the summation runs over units  $k$  that have unit  $j$  as input):

$$f'_j(x_j) \frac{\partial E}{\partial z_j} = f'_j(x_j) \sum_{k:j \rightarrow k} w_{jk} \frac{\partial E}{\partial x_k} \\ = f'_j(x_j) \sum_{k:j \rightarrow k} w_{jk} \delta_k. \tag{6}$$

For network models with multinomial logistic functions of the membership values in the hidden layer this leads to

$$\delta_{s_{hp}} = s_{hp}(1 - s_{hp}) \sum_o \beta_{ho} \delta_{y_{op}}. \tag{7}$$

During the forward pass values of hidden units or output variables are determined layer after layer starting with the input units on the basis of the weighted summation and transforming functions (here: multinomial logit and binomial logit functions). During the backward pass the  $\delta_j$  and the  $\partial E / \partial w_{ij}$  are calculated beginning with the output units.

The different stages of the backpropagation algorithm are:

1. Initialize the iteration counter  $t = 1$ .
2. Initialize the learning constant  $\eta = 0.1$  and the momentum parameter  $\theta = 0.6$ .
3. Initialize  $E(0)$  to a very high value.
4. Initialize coefficients  $\alpha_{oh}, \beta_{ho}$  ( $o = 1, O, h = 1, H$ ) randomly to values in the interval

$[-0.1, +0.1]$ .

5. Set the observation counter  $p = 0$ .
6. Increase the observation counter  $p = p + 1$ .
7. Compute membership values  $s_{hp}$  ( $h = 1, H$ ) of observation  $p$  by Eq. (2).
8. Compute theoretical values  $\hat{y}_{op}$  ( $o = 1, O$ ) of the segmentation criteria of observation  $p$  by Eq. (3).
9. Compute  $\delta_{y_{op}}$  ( $o = 1, O$ ) by Eq. (5).
10. Compute  $\delta_{s_{hp}}$  ( $h = 1, H$ ) by Eq. (7).
11. Change coefficient values by subtracting from  $\alpha_{oh}$  and  $\beta_{ho}$  respectively:

$$\Delta\alpha_{oh}(t) = \eta\delta_{s_{hp}y_{op}} + \theta\Delta\alpha_{oh}(t - 1), \quad o = 1, O, \quad h = 1, H,$$

$$\Delta\beta_{ho}(t) = \eta\delta_{y_{op}s_{hp}} + \theta\Delta\beta_{ho}(t - 1), \quad h = 1, H, \quad o = 1, O.$$

12. If  $p < P$ , goto 6.
13. Compute the error measure  $E(t)$  by Eq. (1).
14. If the error measure  $E(t)$  has changed essentially compared to  $E(t - 1)$ , increase the iteration counter ( $t = t + 1$ ) and goto 5.

In step 11 we enlarged the basic backpropagation algorithm by considering momentum terms  $\theta\Delta\alpha_{oh}(t - 1)$  or  $\theta\Delta\beta_{ho}(t - 1)$ , which depend on the modification of a parameter in the previous iteration  $t - 1$ . This way the danger of oscillating parameters during estimation is reduced, as momentum terms prevent that changing directions of the gradient have a full effect on new parameter values.

Moreover we adaptively determine step size by varying the learning constant  $\eta$ . If during every 50 iterations  $E$  does not decrease,  $\eta$  is multiplied by 1.2, otherwise by 0.7.

After about 2000 iterations this extended backpropagation algorithm usually converges.

## 2.2. K-means

As K-means is well known we only give a short pseudo-algorithmic description of the implementation used (Jain and Dubes, 1988):

1. Set the iteration counter  $t = 1$ .
2. Generate randomly an initial partition with K clusters.

3. Compute cluster centers (i.e. vectors of average criterion values for each cluster).
4. Generate a new partition by assigning each pattern to its closest cluster center in terms of Euclidean distance.
5. Compute new cluster centers.
6. If cluster memberships change compared to the last iteration, increase the iteration counter ( $t = t + 1$ ) and goto 4.
7. Stop.

## 3. Evaluation of cluster analysis results

It might seem obvious to use the square-error objective in order to evaluate results obtained by the cluster analysis methods considered here. But  $E$  (or similar fit indices) come with a serious disadvantage: in most cases they improve (i.e. decrease) with larger number of segments. This behavior of fit indices makes the decision on the number of segments hard, if not impossible. What is worse, this behavior could be caused by the lack of a cluster structure of the data studied. In this situation application of any cluster analysis algorithm clearly does not make sense.

We use a relative index of cluster validity, the Davies–Bouldin index  $DB(H)$  which can be computed for  $H > 1$  clusters (Davies and Bouldin, 1979):

$$DB(H) = 1 / \left( H \sum_{h=1}^H R_h \right) \quad (8)$$

$R_h$  is defined as follows for any segment  $h$ :

$$R_h = \max_{j \neq h} (e_h + e_j) / d_{hj},$$

where  $e_h$  is the square root of the average square error of segment  $h$ ,  $d_{hj}$  the Euclidian distance of the centers of clusters  $h$  and  $j$ .

The smaller  $DB(H)$  the better the clustering. Small values of  $DB(H)$  occur for a solution with low variance within segments and high variance between segments. Therefore one chooses the number of segments at which this index attains its minimum value.

If one obtains the minimum value for a two segment solution, this could also reflect the fact that there are not clusters in the data as  $DB(H)$  is not defined for  $H=1$ . In this situation a procedure to test against the hypothesis of no-clusters or randomness should be additionally used.

We follow recommendations of Jain and Dubes (1988) in developing the following procedure.

1. Generate  $p$  random vectors of the segmentation criteria having the same averages as the empirical data set.
2. Determine a two segment solution by means of a cluster analysis algorithm and compute the corresponding  $E$ .
3. Repeat steps 1 and 2  $m$  times (with  $m=100$ ).

The null hypothesis of randomness can be rejected with significance  $r/m$ , if the  $E$  of the two cluster solution for the empirical data obtained by the same cluster analysis algorithm is lower equal than the  $r$  smallest  $E$  values of the  $m$  simulated data sets. If rejection of the null hypotheses occurs at a low significance value (say  $\leq 0.01$ ), this means strong evidence for a two-segment structure.

#### 4. Empirical study

##### 4.1. Data

Our data set consists of usages of brands (product category: household cleaners) in different usage situations, demographic variables and attitudes (see Table 1). The respondents constitute a representative random sample of 1007 housewives.

Seven different brands A,B,C,D,E,F,G of cleaners and five different usage situations 1, . . . , 5 (Table 1) are distinguished. This leads to 35 different usages A1, A2, A3, A4, A5, B1, . . . , G1, G2, G3, G4, G5. A1 up to G5 are all binary variables, where, e.g.  $A1=1$  means that the respondent uses cleaner A in situation 1,  $A1=0$  that the housewife does not use cleaner A in situation 1 etc.

We only consider as segmentation criteria 20 of these 35 usages having a minimum frequency of 50 (see Table 2). After deletion of incorrect data 831 respondents remain for analysis.

Table 1  
Variables considered

<i>Usage situations</i>	
Synthetic surfaces	
Lacquered surfaces	
Tiles	
Ceramics, enamel	
Floors, stairs	
<i>Demographic variables</i>	
Age	
Household size	
Number of children	
Housewife's education	
Housewife's occupation	
Second residence	
Population size of household residence	
Household members with income	
Household income	
<i>Attitude variables</i>	
Cleaning the household is cumbersome	
It is better to buy products that save work even if they are a bit more expensive	
I appreciate it if my family helps with the housework	
If you do not see to it that the household is absolutely clean infections are probable	
Most of the cleaners are too sharp	
For specific chores in the household you need special cleaners	
I like to try new cleaners	

##### 4.2. Results

Both the K-means and the backpropagation algorithms start with 100 different initial random values for cluster memberships and parameter values, respectively. Table 3 contains results for the best (i.e. minimum square-error) solution of each algorithm among the 100 solutions for a varying number of segments.

Table 2  
Segmentation criteria used

Brand	Usage situation				
	1	2	3	4	5
A	A1	A2	A3		A5
B	B1	B2	B3		B5
C	C1		C3		C5
D	D1	D2			
E			E3	E4	
F				F4	
G	G1	G2	G3	G4	

Table 3  
Square-error and Davies–Bouldin index

<i>H</i>	K-means		Neural network	
	<i>E</i>	DB	<i>E</i>	DB
2	1687.27	2.66	1581.06	0.51
3	1557.46	2.65	1347.72	1.02
4	1466.77	2.37	1069.65	1.22
5	1383.12	2.23	839.40	1.22
6	1320.05	2.21	615.20	1.14
7	1276.08	2.10	380.62	1.34
8	1226.85	1.99	283.48	1.38
9	1165.25	2.04	211.01	1.60
10	1144.66	2.25	132.98	1.66
11	1134.54	1.95	96.20	1.72
12	1100.99	1.92	49.80	2.07
13	1086.27	2.02	47.91	2.01
14	1060.24	1.97	38.99	2.37
15	1030.44	1.92	33.16	2.16
16	1010.45	1.89	26.24	2.31
17	998.98	2.03	30.44	2.05
18	989.55	1.94	35.86	1.99
19	962.57	1.97	25.74	2.08
20	951.37	1.95	29.09	1.85

For the K-means algorithm the Davies–Bouldin index attains its minimum value for a number of 16 segments. But it must be emphasized that for this solution within-segment variation is high relative to between-segment variation. Overall behavior of the index is typical for weak cluster structure or random data.

For the feedforward neural network all square-error values are much lower than those for K-means for any number of segments between 2 and 11. Similar to K-means *E* decreases when the number of segments increases, making decision on the number of segments difficult. The Davies–Bouldin index becomes minimal for a two segment solution.

Therefore it is not clear if there is any cluster structure in the data analyzed. To answer this question we use the test against randomness introduced in Section 3. The computations show that square error-values for all 100 randomly generated data sets are higher than the *E* for the two segment solution obtained by the neural network. This result strongly confirms existence of two segments among the respondents with regard to the segmentation criteria considered.

The best two segment solutions obtained by both K-means and the feedforward network are compared using demographic and attitude variables as external criteria. To this end we estimate logistic regression models with membership in the first segment as dependent variable and external criteria as independent variables. Table 4 shows the logistic regression model for the segmentation determined by the feedforward network. Probability of membership in the first segment increases if population size of the residence is greater than 50 000, the housewife is between 20 and 29 years old and has vocational schooling.

For each respondent values of external criteria are inserted into the relevant logistic regression equations. A respondent is assigned to the first (second) segment if the membership probability computed this way is higher (lower) than 0.5. This procedure leads to hit rates of 65.5% and 50.1% for the feedforward neural net and K-means, respectively. Therefore we conclude that clustering by means of the feedforward net is superior.

We now present some of the results obtained by the two-segment solution of the feedforward network. Average memberships amount to 0.663 and 0.337 in the first and second segment, respectively. The standard deviation of membership values is 0.243. If each person is assigned to exactly one cluster on the basis of her maximum membership value, cluster sizes are 541 and 290 persons in the first and second segment, respectively.

Weights of connections between input variables and hidden units  $\alpha_{oj}$  may be used to interpret the clusters for managerial purposes (see Table 5). The higher the absolute value of such a weight is,

Table 4  
Logistic regression model for the neural network segmentation

Independent variable	Coefficient	t-value
Population size		
2001–5000	–1.51	–9.25
5000–50000	–1.38	–8.13
Age 20–29 yr	1.43	12.31
Primary education	–0.73	–6.44
Vocational school	2.71	25.97
Constant	3.85	40.22

Contains variables significant with  $\alpha = 0.01$ .

Table 5  
Weights of the neural network

Input variable	First hidden unit		Second hidden unit	
	$\alpha_{o1}$	$\beta_{1o}$	$\alpha_{o2}$	$\beta_{2o}$
A1	-0.279	-2.316	-0.074	-1.195
A2	-0.189	-2.553	-0.181	-2.040
A3	-0.225	-2.620	-0.083	-1.458
A5	-0.128	-2.887	-0.163	-2.295
B1	-0.406	-7.879	0.292	3.203
B2	-0.446	-8.265	0.432	2.187
B3	-0.402	-5.594	0.292	1.483
B5	-0.235	-2.950	0.036	-0.197
C1	-0.227	-2.903	-0.072	-1.193
C3	-0.213	-2.738	-0.130	-1.244
C5	-0.195	-2.558	-0.157	-1.757
D1	-0.190	-1.772	-0.169	-2.213
D2	-0.185	-2.167	-0.203	-2.570
E3	-0.198	-2.719	-0.194	-2.014
E4	-0.223	-2.059	-0.108	-1.150
F4	-0.240	-2.557	-0.045	-0.835
G1	-0.073	0.321	-0.607	-4.331
G2	-0.178	-1.036	-0.311	-5.010
G3	0.298	3.039	-1.303	-14.468
G4	0.198	4.029	-1.008	-6.877

the more characteristic the input variable is for the segment regarded. Positive weights indicate that usage of a brand in the respective situation is associated with membership in the segment. On the other hand, negative weights show that non-usage of a brand in a certain situation is associated with membership in the segment.

According to Table 5 using brand G for cleaning tiles or ceramics and enamel as well as not using brand B for cleaning synthetic or lacquered surfaces or tiles is seen to be important for membership in the first segment. Using brand B for cleaning synthetic or lacquered surfaces or tiles as well as not using brand G for cleaning synthetic surfaces, tiles or ceramics and enamel is characteristic for membership in the second segment.

## 5. Conclusions

For a real life data set the proposed feedforward neural network model resulted in a two segment solution that was confirmed by appropriate tests. On the other hand, the K-means

algorithm failed in discovering any somewhat stronger cluster structure. Moreover, classification of respondents on the basis of external criteria not used to form clusters was better for the neural network solution.

This is in contrast to the studies mentioned in the introductory section in which artificial neural networks (self-organizing maps, competitive learning) did not succeed in excelling K-means.

An obvious reason for this result could be the fact that the specified feedforward neural network model is more flexible than the methods considered in these studies with regard to the form of association between segment memberships and segmentation criteria. Feedforward networks with one layer of hidden units with sigmoidal (e.g. multinomial logistic) functions are guaranteed to approximate any continuous multivariate function with any desired precision given a sufficient number of hidden units (Ripley, 1993). Such properties are not known to exist for neural networks of the unsupervised learning type. On the whole it therefore seems to be worthwhile to consider feedforward nets to solve cluster analysis problems if they possess an appropriate architecture.

## References

- Balakrishnan, P.V., Cooper, M.C., Jacob, V.S., Lewis, P.A., 1994. A study of the classification capabilities of neural networks using unsupervised learning: A comparison with k-means clustering. *Psychometrika* 59, 509–525.
- Balakrishnan, P.V., Cooper, M.C., Jacob, V.S., Lewis, P.A., 1996. Comparative performance of the FSCL neural net and K-means algorithm for market segmentation. *European Journal of Operational Research* 93, 346–357.
- Bishop, C.M., 1995. *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford.
- Bridle, J.S., 1990. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation parameters. In: Touretzky, D.S. (Ed.), *Advances in Neural Information Processing Systems 2*. Morgan Kaufmann, San Mateo, CA, pp. 211–217.
- Cheng, B., Titterton, D.M., 1994. Neural networks: A review from a statistical perspective. *Statistical Science* 9, 2–54.
- Davies, D.L., Bouldin, D.W., 1979. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1, 224–227.

- Green, P.E., 1971. A new approach to market segmentation. *Business Horizons* 20, 61–73.
- Green, P.E., Krieger, A.M., 1995. Alternative approaches to cluster-based market segmentation. *Journal of the Market Research Society* 3, 221–239.
- Haykin, S., 1994. *Neural Networks. A Comprehensive Foundation*. MacMillan, New York.
- Hertz, J., Krogh, A., Palmer, R.G., 1991. *Introduction to the Theory of Neural Computation*. Addison-Wesley, Redwood City, CA.
- Jain, A.K., Dubes, R.C., 1988. *Algorithms for Clustering Data*. Prentice-Hall, Englewood Cliffs, NJ.
- Kohonen, T., 1984. *Self-Organization and Associative Memory*. Springer, Berlin.
- Krishnamurthi, A.K., Ahalt, S.C., Melton, D.E., Chen, P., 1990. Neural networks for vector quantization of speech and images. *IEEE Journal on Selected Areas in Communication* 8, 1449–1457.
- Ripley, B.D., 1993. Statistical aspects of neural networks. In: Barndorff-Nielsen, O.E., Jensen, J.L., Kendall, W.S. (Eds.), *Networks and Chaos – Statistical and Probabilistic Aspects*. Chapman & Hall, London, pp. 40–123.
- Ripley, B.D., 1996. *Pattern Recognition and Neural Networks*. Cambridge University Press, New York.
- Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representations by error propagation. In: Rumelhart, D.E., McClelland, J.L. (Eds.), *Parallel Distributed Processing. Explorations in the Microstructure of Cognition 1*. MIT Press, Cambridge, MA, pp. 318–362.
- Wind, Y., 1978. Issues and advances in segmentation research. *Journal of Marketing Research* 15, 317–337.