Artificial Neural Networks for Cluster Analysis[1]

Joseph Woelfel

University at Buffalo

January 24, 1993

[1] Paper presented at the American Marketing Association Attitude Research Conference, Phoenix, AZ, Jan. 1993. I am grateful for the assistance of Nick Stoyanoff and Scott Danielsen of Terra Research and Computing Co.

Rah Press

Every aspect of our experience is like this. Clouds form and dissolve as they move across the sky, the land gradually lifts until it forms hills and mountains, seas lap unendingly against ever-changing shores, trees shake in the wind, grow, change their leaves, just as people are born, grow up, get fatter or thinner, happier or sadder, healthy and sick.

Yet when we speak of our experiences, we speak of them as if they were sharply bounded: that is a tree, this is a hill, there is George, he is a conservative, a good provider and decent husband. This ability to transform an essentially boundryless experience into sharply defined mental categories is the basis of human perception and communication. The process of drawing categories from continuous experience is called *induction*, and induction has defied philosophical analysis since the ancients.

Just how hard a problem this has turned out to be can be shown by the quality of the minds who have dealt with the question and by the completeness of their disagreement: Plato (5th century B.C.) thought that concepts or categories were "remembered" (albeit dimly) from an earlier, more perfect, existence. Aristotle (Plato's student) believed the categories of thought could be discovered by examining experience to find the essential aspects which persisted across changes and individual variations.

But by the time of David Hume (Hume died in 1776) no one had yet discovered how to do this, and so many Philosophers, Hume included, came to believe that concepts could not be derived from experience. Immanuel Kant (died 1804), caught between the belief that some concepts seemed absolutely necessary and the belief that they could not be derived from experience, decided that concepts must be built into the mind at birth -- he called these concepts "a priori." Thomas Jefferson (died 1826) made no effort to uncover the origins of his own most important concepts, and simply held "...these truths to be self-evident." Much later, Martin Luther King (died 1968) explained the origin of his most basic concepts simply by saying "I had a dream."

By the time of Einstein, however, (died 1955), many scientists, including Einstein, had come to believe that concepts are simply "a creation of man," "logically entirely arbitrary,", and "from the point of view of logic freely chosen conventions" (Einstein, 1991).

Although a complete analysis of all that has been written about the formation of categories would make a fair history of philosophy, in the past two and a half millennia, our best minds have believed that categories (1) came from heaven; (2) were essential features discovered by observation; (3) came from mystical sources; (4) were inborn in the mind from birth, and, most recently, (5) are completely arbitrary, made up by humans for their own convenience. Is this confusing enough? If you're not confused, perhaps you don't understand the seriousness of the problem!

Artificial Neural Networks and Cluster Analysis

Social research methodologists have dealt with this same question under a variety of names, most commonly "cluster analysis." Following on original work by Spearman (1903), most work that followed has attempted to find some set of criteria which will file a set of stimuli into a (usually smaller) set of "piles" or categories based on their underlying similarity relations. Essentially, cluster analysis represents an attempt to find an algorithm that can solve the philosophical problem of induction automatically.

Most methods of cluster analysis begin with a concept about what kinds of clusters are desirable. This concept is invariably expressed in the form of a criterion. One such criterion -- perhaps the most common -- is that the "best" clusters are those in which the ratio of between-cluster variance to within-cluster variance is a maximum. This, however, is by no means the only kind of criterion that could be chosen. Perhaps even more common is the criterion that 26 categories ought to be formed, and that each stimuli should be sorted into one and only one of those categories based on the initial letter of its name.

In addition to this criterion, whatever it might be, each technique requires some algorithm which tries alternative ways of filing the stimuli into different bins, checking the value of the criterion for each alternative way.

Clustering techniques may also be subjected to arbitrary constraints. Typical constraints might require the solution to fit into a fixed number of categories, or that each stimulus must fall into one and only one category, for example. Different kinds of criteria, along with different kinds of constraints (and, of course, different kinds of input data!) will yield different outcomes; that is, different classifications or categories or clusters of the input elements.

What kind of clustering criterion is best?

The most important realization the cluster analyst can make is to understand that *there is no "correct" way to cluster*. As Einstein said, the concepts used to classify experience can not be found in data, but are "a creation of man." A cluster analysis is necessarily a combination of human interest and the data, and never of the data alone. Cluster analysis algorithms, therefore, always rely on human judgments, and will produce different clustering results based on differing judgments. Perhaps more importantly, a traditional cluster analysis algorithm will always (obviously) produce the same clustering outcome everytime it is fed the same data, as long as some human being doesn't change it's parameters. *But a human being will not classify or cluster the same set of objects in the same way each time. How a person breaks up the continuous, ever changing world of experiences into categories depends not only on the experiences, but on the interests of the person.* Thus, there is no "correct" clustering solution; only those that match human interests more or less closely.[2]

---

[2] This should not be taken to mean that all clustering algorithms are created equal. Some algorithms are known to be so inefficient they are seldom if ever feasible. Others produce clusters that are seldom of interest to any purpose people are likely to have. Some others (such as algorithms that maximize the

The fact that there is no single "correct" way to cluster any given set of stimuli complicates the process of selecting a useful clustering algorithm. In principle, a two step process is required: first, the human analyst must decide what kind of clusters he or she needs or wants in the particular situation in question, and second, must identify a criterion for clustering that will yield clusters as much like those desired as possible. In practice, however, this is almost never done: analysts usually make use of an algorithm which is available to them because it is included in a statistics package to which they have access (usually set to the default options), or they rely on the expert judgments of a methodological consultant who is almost certainly less qualified than the client to determine what kind of clustering is needed in that particular situation.

Artificial Neural Networks

Recently, however, a new kind of analysis system, the Artificial Neural Network (ANN) has appeared on the scene, and may well shed a new kind of light on this well-aged problem.

In one sense, the Artificial Neural Network can -- and ought to be -- thought of as simply another tool in the researcher's toolkit, to be used where it works better than other tools, and left at home when inappropriate. But there is another sense in which ANN's as clustering devices provide a fundamentally new approach as well as a powerful new way to think about categorization and clustering. This new and fundamental aspect of ANN's follows from the fact that they represent a

---

between/within cluster variances) produce clusters that are always of interest for some purposes, e.g., search and retrieval systems.

It does mean, however, that no single criterion for clustering can match the wide array of ways in which people actually cluster their own experiences at different times for different purposes, and, hence, there is no such thing as a "best" algorithm.

synthesis of fundamental new discoveries about how clustering occurs in nature -- that is, in real neural networks, and in real brains.

Most important is the fact that Artificial Neural Networks do not work by maximizing or minimizing some criterion as to how clusters should be optimized, as do conventional algorithms. Rather, *neural networks work by examining examples of existing clusters, and they then learn to produce clusters like those they studied.* This means that the analyst need not understand what criterion is being maximized or minimized to produce clusters of the sort he or she wishes -- it is only necessary to produce some examples of clusters that already exist. These "cases" are then studied by the neural network, which learns how to produce others like them.

A Backpropagation Example

There are two major kinds of neural networks: *supervised* and *self-organizing*. By far the best known are back-propagation supervised models, which have been well defined elsewhere (Rummelhart, et.al, 1988, Woelfel, 1993). To show how a back-propagation supervised neural network might be used for cluster analysis, a simple problem was constructed using the concepts P51, P38, B17, B29, BOMBERS, FIGHTERS, ALLIES, and AXIS. Of these, the first 7 were used as input characteristics (the equivalent of independent variables in regression analysis), while the last four were used as output characteristics.

Data consisted of "cases", analogous to the regression model case, where each case consisted of a specific set of values of both input and output characteristics. The first case, for example, gave these values (dependent or output characteristics below the line):

P51	1
P38	0

ZERO 0

ME1090

B17 0

B29 0

HEINKEL 0

FIGHTER 1

BOMBER 0

ALLIES 1

AXIS 0

This case indicates that the P51 is an Allied Fighter.

Note that it is not necessary to restrict ourselves to a single "active" input characteristic per case, as the following case shows:

P51 1

P38 1

ZERO 0

ME1090

B17 1

B29 1

HEINKEL 0

FIGHTER 1

BOMBER 1

ALLIES	1
AXIS	0 .

This case says that the P51, P38, B17 and B29 are associated with fighters, bombers and the Allies.

The network developed to deal with these cases consisted of seven input nodes, 3 hidden nodes and four output nodes. It trained in 4600 "training events", and produced a solution which allows convenient classification of any input. For example, if one were to input the following values of the input characteristics, the trained network would estimate the output characteristic values as follows:

P51	1
P38	0
ZERO	0
ME109	0
B17	0
B29	0
HEINKEL	0

FIGHTER	.89
BOMBER	.11
ALLIES	.90
AXIS	.10 .

This output means that, when faced with a P51, the network classifies it as an allied fighter. But consider the following input:

P51	1
P38	1
ZERO	1
ME109	1
B17	0
B29	0
HEINKEL	0

FIGHTER	.90
BOMBER	.10
ALLIES	.09
AXIS	.11

This means that, when faced with a two allied fighters and two axis fighters, the network decides they are members of the category "fighter" and declines to say whether they are allies or axis. Although this is a very simple example (deliberately designed to be so) it shows that there is a sense in which the neural network is non-hierarchical. The neural network does not assign each stimulus into its one best category, but assigns each input stimulus into one or another (or several) categories *depending on the context in which it is seen*. Just how important this might be is obvious if we consider the difference between a tiger in a cage and a tiger in your living room.

Unsupervised Neural Networks as Clustering algorithms.

There are still problems with the supervised approach to classification and clustering, however. In one important sense, these networks are still hierarchical, since they treat one set of stimuli (usually elements or members of categories) as inputs and others (usually the category names) as outputs. While it is possible to define a problem within a supervised network so that this is not so, there is an easier and more direct way to deal with the problem using unsupervised networks.

In an unsupervised network, however there need be no distinction between input and output nodes. (Some unsupervised plans do make such a distinction, but this is not necessary.) What follows is the description of a simple *interactive activation and competition neural network (IAC)*, which is particularly well suited for clustering applications.

Structure of a simple unsupervised network

Consider a network consisting of 11 nodes, none of which are connected to any of the others. Let each node represent one of the stimuli in the preceding example, i.e., P51, P38, B17, B29, BOMBERS, FIGHTERS, ALLIES, and AXIS. Now we can expose this network to the data by the following rule. When it reads a "case," each stimuli that occurs in the case will activate its corresponding neuron. If the network "reads" a case that says "P51, P38, FIGHTER, ALLIES," for example, the nodes that correspond to those four stimuli will become active.

A simple learning rule

Now we adopt a second rule (called a Hebbian learning rule, which is mathematically equivalent to Pavlov's law of association) which says that *the connection among any nodes that are simultaneously active will be strengthened, while all others are weakened.* Clearly, after reading several cases, those stimuli which co-occur in the cases will tend to become positively interconnected in the network, while those that seldom or never co-occur will become negatively interconnected. The net result will be a square similarities-dissimilarities matrix which expresses the interrelations among the stimuli in a concise fashion.

Operation of the network

A very wide variety of operational rules exist, but the following is one of the most powerful. Whenever a node is active, it transmits its activation to all the other nodes to which it is connected with a force that is proportional to the activation value multiplied by the connection strength. Thus, if we consider the activation value of a given node to be 1, and it is connected to a second node with a connection strength of .5, it will transmit an activation force to that node of .5. If it is connected to a third node with an activation value of -.7, it will communicate to that third node an activation force of -.7. Since this force is negative, we can consider it a force which attempts to turn that third node off rather than on.

Now, what happens to the nodes that are not active? Each of these receives activation forces from all the other nodes to which it is connected. Some of these forces are positive, while others may be negative. In a typical network, each node sums up the incoming forces (usually in a nonlinear summation function, typically a logistic) and, if the resulting sum is greater than some present threshold value, the node itself becomes active. In some networks, the activation values of the nodes may be binary, i.e., on or off, zero or one, or plus or minus one, while in the network considered here, the activation value is continuous, allowing any positive or negative number.

What results is an interaction among the nodes, each competing to turn on or off others in the network (hence the name interactive activation and competition.) In practice, in terms of our simple network, we might activate the node called "P38," which node will in turn attempt to turn on some other nodes and turn off some others, depending on the connection strengths among them. As some others turn on, they will in turn attempt to turn on and off still others, and so on. (Each of these stages is called a "cycle.")

A simple example

To show how such a simple IAC network can be used as a clustering machine, the cases in appendix 1 were read into CLUSTER, a modified prototype of the commercial program ORESME (Terra, 1990).

Typical cases are shown in figure 1. As the network reads these cases, it adjusts its connection strengths according to the frequency of co-occupance of stimuli in the cases. Once it has read all the cases, the final connection strengths or weights are arrayed in a square matrix that has the formal properties of a similarities matrix. This matrix can be treated statistically as if it were generated by any of a variety of conventional statistical methods.

p51
p38
zero
me109
fighter
-1
b17
b29
heinkel
bomber
-1
zero
heinkel
me109
axis
-1

Figure 1: Typical cases for an IAC network; (-1's indicate end of case)

Input	Output
Bomber	Bom
Fighter	Fighter, P38,
Axis	Axis,
P38	Fight
Axis, Fighter	Axis, Fighter
Axis, Bomber	Axis, Bombe
B17	Bom
P51, B17	P38,

Table 1: Inputs to the IAC network (

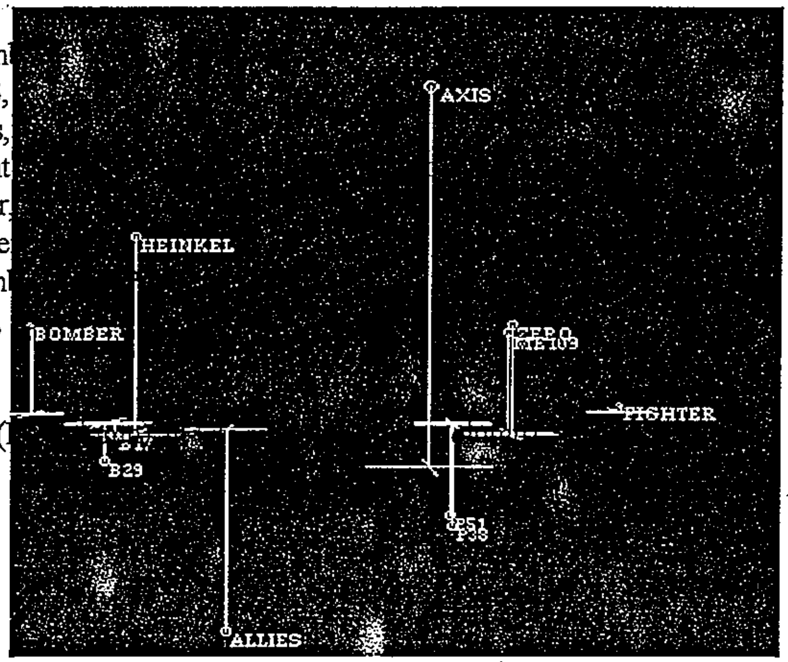

Figure 2 shows a perceptual map made by the Galileo program (Terra, 1990) based on the similarities matrix generated by the network. In this map, we can see that planes toward the top of the space tend to be Axis planes, while those at the bottom tend to be Allied planes; those to the right of the screen tend to be fighters, while those to the left are bombers.

In a typical conventional cluster analysis, the analyst would attempt to derive some criterion which would draw circles or spheroids or perhaps nonregular geometric surfaces which would separate the several stimuli into clusters. But the unsupervised neural net works quite differently. Instead of developing bins filled with elements, the neural net may be queried. One can enter a stimuli or set of stimuli into the network, and it will respond with the most closely related stimuli. Table 1 shows the results of entering various stimuli into the IAC network trained using the cases in appendix 1:

Woelfel - Cluster Analysis

As Table 1 makes clear, the common practice of deriving an exhaustive set of clusters, with each element occurring in one and only one cluster (as we do in a typical dendogram or Venn Diagram) is not sufficiently flexible to represent the depth of information available from the neural network. The IAC network, for example, is able to show that the stimulus "B17", taken alone, is part of the category of all the aircraft, and indeed elicits the names of all the other aircraft and the category "Bomber" when input into CLUSTER. But when the same term, "B17," is input into the network along with "P51," only the Allied aircraft are elicited.

Moreover, Table 1 also shows that the IAC network is completely non-hierarchical; one can enter an element and retrieve its category name (and the other members of the category), or one can enter the category name and retrieve its elements. Indeed, the network does not treat category names differently than it does element names; all are simply "objects" or "stimuli" which are more or less similar to others.

Mistakes

Some analysts might consider the clusters developed by the network to be imperfect according to some ideal classification scheme. For example, when one names a single fighter plane, the network might respond with the names of some bombers as well as fighters. This kind of thinking, however, is symptomatic of the old belief that there exist a Platonic "correct" set of categories and a correct rule for assignment of members. This is not the case. In fact, the network is meant to apply a process which is meant, insofar as is reasonably possible, to be similar to the process human beings would use in making the same kinds of judgments. If the network worked "perfectly", it would make the same classification decisions that a human being would make if he or she read the same input data. Thus, if the cases provided show strong relationships between planes of various categories, we would expect a human reader to be reminded of planes in the various categories when stimulated to think of one of them.

Conclusion

Neural networks provide a new and different way to think about cluster analysis. They ought not be thought of as replacements for conventional techniques, but rather new methods that make new kinds of clustering and new applications possible.

Among the advantages they provide are the following:

o There is no need to define a criterion which can produce the kinds of clusters needed. Neural networks can develop their own criteria from example, given a set of existing cases.

o Neural networks are non-hierarchical, and need not divide experience into arbitrary levels of generality

o Items need not be assigned to one and only one category, but may be found in several at once: thus a P38 can be an allied plane and a fighter plane.

o Which category or categories an item is assigned can vary depending on the context in which it is mentioned

o Neural networks may well provide a simulation of actual human judgments of the same data.

This list by no means exhausts the list of advantages a neural approach to clustering and classification can provide to the analyst. But, most important, neural networks provide a range of new possibilities and new applications which have only begun to be understood. It would be a mistake, however, to believe that ANN's provide simply another computational mechanism to achieve the same result aimed for by conventional clustering techniques. Using these new methods to produce old results is a waste of time; there already exist a set of procedures for accomplishing those aims. The real test of neural networks will be finding new applications and new uses that are beyond the range of conventional procedures.