The Galileo Company

23 A Durham Drive
Amherst, NY 14228

(716) 636 7865
(607) 272 8057 FAX

NEURAL NETWORKS:

Applications of the Cognitive Revolution To
Advertising, Marketing and Political Research

Joseph Woelfel, Ph.D.

Rah Press

The Cognitive Revolution

The last three decades have seen change in our understanding of mental processes so fundamental and sweeping it is called "The Cognitive Revolution." The academic community refers to all "mental" processes as "cognitive processes", from the Latin word *cogere*, to think. (Rene DesCartes made this word famous with his dictum "Cogito ergo Sum", which means "I think, therefore I am.")

Cognitive processes include such processes as thinking, feeling, attitude change and the like, but computers and modern communication systems have broadened our understanding to include not only collective cognitive processes, like changes in the beliefs and attitudes of groups and cultures, markets and market segments, but even symbolic processes in computers. Some scientists and philosophers consider cognitive processes to be any manipulation of symbolic information whatever.

To gauge the depth of the cognitive revolution, it helps to understand that, only a few years ago, many if not most scientists believed that cognitive processes where inherently unobservable, and either couldn't or shouldn't be a studied at all. Behaviorism was the predominant philosophy, and its adherents treated humans as "black boxes", observing inputs to the box and the outputs that resulted without speculating about what was going on inside. Now, however, cognitive science is one of the fastest growing areas in the academic community, and includes workers from anthropology, communication, computer science, geography, linguistics, philosophy, physics, psychology and other disciplines. These thousands of scientists and scholars focus explicitly on what was thought to be unobservable only a few years ago -- cognitive processes.

Although the cognitive revolution has been wide ranging, there are three areas of development which seem particularly interesting. The first of these is a change in the way thinking and reasoning is described, from a categorical to a "fuzzy" model. The second is a deepening understanding of the fundamental processes by which the brain operates -- an understanding of the physical principles underlying *neural networks.* Third, explosive developments in computing have made it possible to simulate cognitive processes in computer software. This has made it possible to test theories that would otherwise have been purely speculation. And it has also given rise to whole new technologies which are now revolutionizing many aspects of human endeavor.

Fuzzy Logic

Since the time of Aristotle, scholars have thought of reasoning as a categorical process. As Bruner said, “The most self evident aspect of our experience is that we place things into categories. That is a *man* and he is *boarding a bus* with the intention of *getting some relaxation.*”

Each and every man is a member of the category *man*, and no individual man is more or less a member of the category than any other. The boundaries of the category are crisp and distinct, and each member of the category is assigned into that category because he, she or it possess the defining characteristics or essential features of that category. (For the category *man*, Aristotle required two such characteristics, *rational animal*.)

Reasoning or thinking within the Aristotelian categorical model was by syllogism, a method of nesting or including categories within categories, as in the familiar classic syllogism

All men are mortal
Socrates is a man
Socrates is mortal.

This is a very powerful model, and it has lived a useful life for over two thousand years. But there are real problems with the categorical model, all flowing from the assumption that categories have sharp, distinct boundaries, and that all members of any category are to be considered identical as far as their membership in the category is concerned.

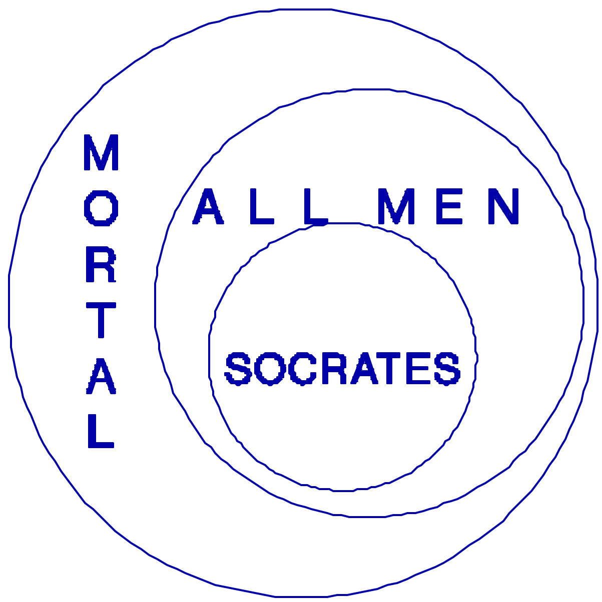


FIGURE 1. -- ARISTOTELIAN SYLLOGISM

In fact, category boundaries are seldom very sharp, and honest observers regularly disagree about whether objects belong to one category or another. (Aristotle did not believe women, slaves, and most non-Greeks belonged to the category *rational*, and thus were not actually "men").

One major step in the Cognitive Revolution was to reconsider categories not as sharply bounded "bins" into which objects could be classified as belonging or not belonging, but rather as "fuzzy sets" with no real boundaries, which faded continuously into neighboring categories.

One well known fuzzy set model, instead of classifying objects as members or nonmembers of any given category, instead assigns them a membership score ranging from zero to one, with 1.0 being a complete, perfect member, and 0.0 not being a member at all. The "best" members of a category -- that is, those that best typify or exemplify the category -- are called "prototype members." Collies and German Shepherds are prototypical dogs, while Schnauzers and Pekingese are less "doglike." Members are assigned values based on their similarity to the prototype members; dogs that are very similar to Shepherds and Collies get a high number, while those that are more dissimilar get lower numbers.

Reasoning within this kind of fuzzy logic consists not of classical syllogisms like the Aristotelian model, but rather with a calculus of probabilities. Fuzzy logics have had wide application in traditional expert systems, and have been very well received particularly in Asia, where industrial designers have incorporated fuzzy logic reasoning even into household appliances like vacuum cleaners.

An even fuzzier model familiar to advertising, marketing and political researchers is the Galileo™ model. Galileo does away with categories completely, and simply assigns scores to pairs of "objects" based on their similarity or dissimilarity. Objects that are very similar are placed close to each other, while objects that are different are placed far apart.

Figure 2 shows the way Galileo would represent the "Dog" category. Collies and Shepherds are close to one another, since they are seen as similar. Pomeranians and Chihuahua's are far from Collies and Shepherds, but close to each other. Terriers and Spaniels lie between these extremes, since they are more similar to Collies and Shepherds than are Pomeranians and Chihuahuas, and they are more similar to Pomeranians and Chihuahua's than are Collies and Shepherds. *In a Galileo, objects are not defined by being placed in categories, but rather by their pattern of similarities and dissimilarities with other objects.* In spite of their name, fuzzy logics are actually more precise than category logics, because they can recognize *degrees* of similarity rather than lumping all similar objects together as if they were identical.

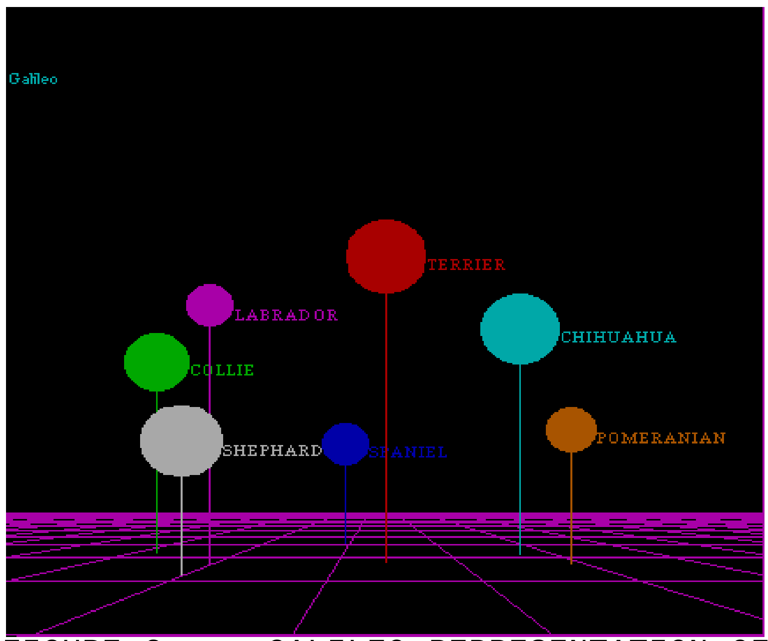


FIGURE 2. -- GALILEO REPRESENTATION OF DOGS

Neural Networks

The human brain is perhaps the most complicated device we know, and it is folly to believe we understand it fully. Deep questions of consciousness, coordination and control remain unsolved. But it is fair to say that fundamental understandings of how the networks of interconnected neurons in the brain store and retrieve patterns of information in principle are beginning to emerge. A natural neural network (like the brain) consists of neurons, each of which may be connected to many other neurons. (In a human brain, there are about 100 billion neurons, each of which is connected, on the average, to about a thousand other neurons.) When a neuron is stimulated, it becomes "active", and sends signals to all the other neurons to which it is connected.

Neural networks store information as patterns in the same way that a TV screen or theater marquee or electronic scoreboard does: By activating some of the dots or light bulbs and leaving others off, any pattern can be displayed. (Researchers have actually identified more than a dozen maps of the visual field in the human brain.) But because the neurons in a neural network are connected to each other, the neural network can do more than simply display patterns of information: it can store and retrieve those patterns, and recognize patterns it has stored even if they are distorted or incomplete.

Although the actual functioning of a neural network like the human brain can be so complicated as to be beyond comprehension, in principle the way a neural network works is very simple and easy to understand. A neural network learns by connecting together the neurons which represent any particular pattern. Since they are connected together, *when some of them are activated, they spread their activation to the others connected to them, which turns on the rest of the pattern.* Thus, when a network sees part of a pattern, it can recall the rest of the pattern, even in spite of incomplete or erroneous information, as long as enough of the pattern is there to activate the rest.

Figure 3 shows a hypothetical network consisting of six nodes representing the words "Cat", "Dog", "Barks", "Howls", "Meows", and "Purrs". Each of these nodes may take on the value "0" (off), or "1" (on). The nodes are connected to each other by weights which represent their relative "closeness" in the network.[1]
They communicate with each other by a simple threshold rule: the signal sent from any node i to any node j equals the product of the activation value of i and strength of the connection between i and j. Thus the total signal received by any node j will be the sum of the signals received from all the other nodes, or

[1] In the present example, the weights are essentially the correlations between frequencies of occurrence of the various words. Thus "Meows" and "Cat" tend to "go together", with a weight of .8, while "Meow" and "Dog" have a negative coefficient of -.8.

$S_j = \Sigma a_i w_{ij}$; I=1,n,

where a_j= the activation of the j_{th} node

n= the number of nodes in the system, and

w_{ij} = the strength (weight) of the connection between the i_{th} and the j_{th} node.

The way a node responds to the set of signals it receives is determined by its activation function; in this case we adopt the rule that the node will be activated if the sum of its input signals is positive; otherwise it will be turned off, or

$$a_i = \begin{cases} +1 \text{ if } x > 0 \\ \text{unchanged if } x = 0 \\ -1 \text{ if } x < 0 \end{cases}$$

Input = "Meows"

	Cat	Dog	Barks	Howls	Meows	Purrs
Cat		-.8	-.9	.2	.8	.9
Dog	-.8		.9	.3	-.8	-.7
Barks	-.9	.9		.5	-.3	-.9
Howls	.2	.3	.5		-.2	-.1
Meows	.8	-.8	-.3	-.2	+1	.8
Purrs	.9	-.7	-.9	-.1	.8	
	on	off	off	off	on	on

FIGURE 3. -- HYPOTHETICAL NETWORK OF DOGS AND CATS

Following this rule, we assume the network receives the input "Meows" from its environment (i.e., the node which represents "Meows" has been activated.) This sets the activation value of "Meows" at +1, and the activation values of the other nodes at 0. Multiplying the weights in each column by the activation values of the corresponding rows, then summing within each column shows that the activation of the node "Meows" will "spread" to the nodes "Cat" and "Purrs", setting their activations to 1, but will leave the nodes "Dog", "Barks" and "Howls" off.

Input = "Howls"

		Cat	Dog	Barks	Howls	Meows	Purrs
	Cat		-.8	-.9	.2	.8	.9
	Dog	-.8		.9	.3	-.8	-.7
	Barks	-.9	.9		.5	-.3	-.9
+1	Howls	.2	.3	.5		-.2	-.1
	Meows	.8	-.8	-.3	-.2		.8
	Purrs	.9	-.7	-.9	-.1	.8	
		on	on	on	on	off	off

FIGURE 4

Figure 4 shows that activating the node "Howls", will also activate the nodes "Cat", "Dog" and "Barks". Figure 5 shows that activating both the nodes "Barks" and "Howls" will also activate "Dog", but will leave "Cat", "Meows" and "Purrs" off.

Input = "Howls" and "Barks"

		Cat	Dog	Barks	Howls	Meows	Purrs
	Cat		-.8	-.9	.2	.8	.9
	Dog	-.8		.9	.3	-.8	-.7
+1	Barks	-.9	.9		.5	-.3	-.9
+1	Howls	.2	.3	.5		-.2	-.1
	Meows	.8	-.8	-.3	-.2		.8
	Purrs	.9	-.7	-.9	-.1	.8	
		off	on	on	on	off	off

FIGURE 5

These examples clearly show that communication among the nodes of the network produces an apparently qualitative change in the pattern recognition and storage capabilities of the network. When the nodes do not communicate, the network can represent a pattern of virtually any complexity when activated directly by the environment, but the complete input is required to produce the complete pattern.

When the nodes communicate, however, the complete pattern can be produced with only a partial input. When a sufficient subset of the nodes in a stored pattern is activated, the activation of those nodes will "spread" through the links and in turn activate the rest of the nodes in the pattern.

It is worth emphasizing the fundamental role communication as it has been defined here plays in this process. A pattern is stored by "connecting" its elements together. Things that "go together" are "close". Nodes (or elements) in turn *communicate* their activation values to other nodes in proportion to their closeness in the communication network. If a node is "on", it will tend to transmit that "on-ness" to other nodes through the links between them, so that the "on-ness" will spread to other nodes which represent the other elements in the pattern. Similarly, if a node is "off", it will tend to communicate its "off-ness" to other nodes through the links between them. *The entire pattern is encoded in the pattern of communication among the nodes as connections or weights, and can be recovered by the activation of any suitable subset of nodes.*

Artificial Neural Networks

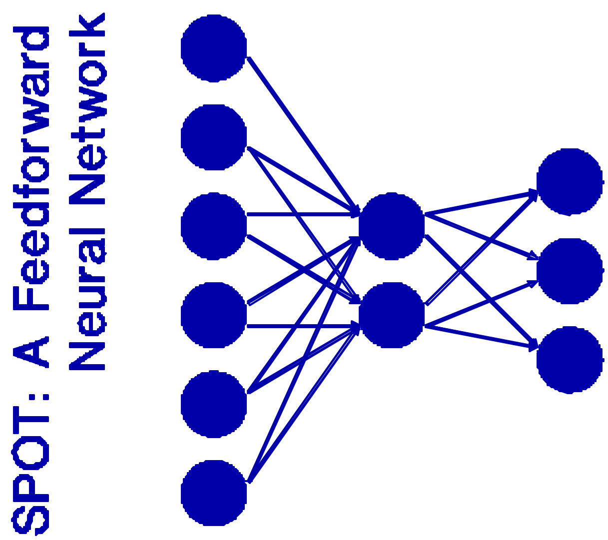
The explosive development of computer hardware and software technology, along with rapidly increasing interest in cognitive processes on the part of Computer Scientists has provided a powerful stimulus to the development of neural network technology. While modern silicon hardware is no match for the technology of the human brain, it is sufficiently potent to provide convincing simulations of natural neural networks. Moreover, these artificial neural networks (ANN's) have developed new and original network designs which are not simulations of naturally occurring networks. Several of these artificial neural networks are already well-developed practical technologies which can provide effective and ingenious solutions to real world problems.

Types of Artificial Neural Networks

Neural networks, real or artificial, have three essential variables: how many neurons they contain, how each neuron responds to inputs (the *activation function*), and how the neurons are connected to each other. As neural networks learn new patterns, the connections among neurons change, since a network's "memory" consists entirely in the pattern of connections or "weights" among the neurons. Networks may be classified according to the way in which their weights change during learning. Although dozens and perhaps hundreds of different kinds of artificial neural networks have already been developed and run, all are variations of two basic types, self-organizing networks and supervised neural networks.

Self-Organizing Neural Networks

Self-organizing neural networks (often called "unsupervised" networks) learn patterns by a simple Pavlovian conditioning rule: When two or more neurons are simultaneously active, the connection among them is strengthened. This means, quite simply, that neurons that have behaved similarly in the past are likely to behave similarly in the future. Self-organizing networks receive information in the form of patterns, which they learn to recognize, and which they can recall later. Self-organizing networks develop an internal representation of the information to which they have been exposed. They are useful because one can enter fragments of a pattern the network has learned, even in somewhat distorted form, and the network can recover the original pattern.



Supervised Neural Networks

Supervised neural networks are usually designed in layers typically including one input layer, one output layer, and one middle or "hidden" layer. Initially these layers are randomly connected from input to hidden and from hidden to output. When a pattern of information is input to the input layer, the activation pattern of the input layer is fed forward to the hidden layer, which in turn feeds its activation pattern forward to the output layer. (These networks are frequently called *feedforward networks.*) Because the layers are randomly connected, the output will be a random pattern.

In a supervised network, however, a trainer or supervisor presents the network with a "correct" output pattern. By comparing the actual output to the correct output, the network calculates a set of errors, and adjusts its connection weights bit by bit until the network produces the correct output to within a specified tolerance. This kind of network can be trained to produce the correct outputs for a series of inputs, and is very useful in situations where a number of previous cases or examples are available to train the network.

Although there are many variations, the most common method supervised networks use to adjust their weights is a mathematical procedure which expresses the errors as a function of the weights, calculates the derivatives of this function and adjusts these weights by a steepest-descent algorithm. Because this method involves tracing the errors as functions of the connections of hidden to output layers, then backwards to the activations of the hidden neurons, then backward again to the connections from input to hidden layers, this method is called *backpropagation*.

Applications of Artificial Neural Networks

The unique characteristics of artificial neural networks have led to a wide range of applications. Self organizing networks, like human beings, have the capacity to observe their environment and learn about it without supervision. Supervised neural networks, like human beings, can be trained to associate input information with appropriate outputs. Neural networks are well suited for tasks a human being might do, but, for one reason or another, can't or won't do. And, because they have some distinctly human characteristics, neural networks are particularly useful for simulating human cognitive processes such as attitude formation and change.

SWITCH

SWITCH is just about the simplest neural network there is: each neuron simply takes in signals from all the other neurons to which it is connected, sums them up and passes them along to all the others to which it is connected. Even such a simple network can have real uses, though.

This table shows the results of entering loyalty data from the 1989 new car buyer study into SWITCH. SWITCH is easily able to calculate how many cars of each type will remain in service for each of the following ten years, and to calculate what percent of the market each will sell for each of these years. assuming, of course, that nothing changes during that period. Of course, many things *will* change, but nonetheless, SWITCH can provide a fixed baseline for planning.

CATPAC

CATPAC is a self-organizing neural network that is optimized for reading text. It can read any ASCII text and learn the underlying concepts conveyed by the text. CATPAC provides both a complete neural network of the interrelationships among the chief words in the text, along with a diameter-method cluster analysis of the main meanings.

CATPAC can be useful for reading textual information from any source, including books, newspapers, magazines, electronic full-text data bases, transcripts of focus groups, in-depth interviews and open-ended questions. Among the important applications of CATPAC are the identification of important product attributes and market segmentation studies. This figure shows such an application. Here, you can see the attributes associated with a select set of fast food restaurants.

CATPAC can also provide outputs which can serve as input to other neural networks for further analysis.

Product	% in Service 1	2	3	4	5	6	7	8	9	10
OTHER	.734	.711	.686	.662	.638	.615	.592	.571	.550	.531
ESCORT	.030	.026	.023	.020	.018	.017	.015	.014	.013	.013
CIVIC	.023	.026	.030	.033	.036	.039	.042	.044	.047	.049
COROLLA	.024	.027	.030	.032	.034	.036	.038	.039	.041	.042
TERCEL	.010	.012	.013	.014	.016	.017	.018	.019	.021	.022
MUSTANG	.013	.013	.013	.012	.012	.012	.012	.011	.011	.011
CELICA	.009	.011	.012	.014	.015	.017	.018	.020	.021	.022
TEMPO	.025	.022	.019	.017	.015	.014	.013	.012	.011	.011
ACCORD	.045	.054	.063	.071	.080	.088	.097	.104	.112	.119
CAMRY	.030	.036	.043	.049	.056	.062	.069	.075	.081	.087
TAURUS	.034	.039	.044	.048	.052	.055	.058	.060	.062	.064
TBIRD	.011	.011	.011	.011	.011	.011	.011	.011	.011	.011
CROWN	.012	.013	.014	.015	.016	.017	.018	.019	.019	.020

Product	Market Share 1	2	3	4	5	6	7	8	9	10
OTHER		.688	.678	.669	.660	.650	.641	.632	.623	.615
ESCORT		.032	.030	.029	.028	.027	.026	.026	.025	.024
CIVIC		.029	.031	.032	.034	.036	.038	.040	.042	.043
COROLLA		.027	.027	.028	.029	.030	.031	.031	.032	.033
TERCEL		.013	.014	.014	.015	.016	.017	.017	.018	.019
MUSTANG		.014	.014	.014	.014	.014	.014	.014	.014	.014
CELICA		.013	.014	.015	.016	.017	.018	.019	.020	.021
TEMPO		.027	.026	.025	.024	.023	.022	.022	.021	.021
ACCORD		.053	.056	.059	.062	.065	.068	.071	.073	.076
CAMRY		.038	.041	.043	.046	.048	.051	.053	.056	.058
TAURUS		.043	.044	.045	.046	.047	.047	.048	.048	.048
TBIRD		.012	.012	.013	.013	.013	.013	.014	.014	.014
CROWN		.012	.013	.013	.014	.014	.014	.014	.015	.015

FIGURE 7. -- Percent of Fleet in Service and Percent Market Share based on loyalty data from a new car buyer study.

GALILEO

Galileo is an artificial neural network in which products, attributes and people are represented as neurons. Each of these products, attributes and people may be more or less tightly connected to each other. Products that are similar may be tightly connected, so that activating "Coke" in the network will probably activate "Pepsi" as well. Products will be tightly connected to their attributes as well,

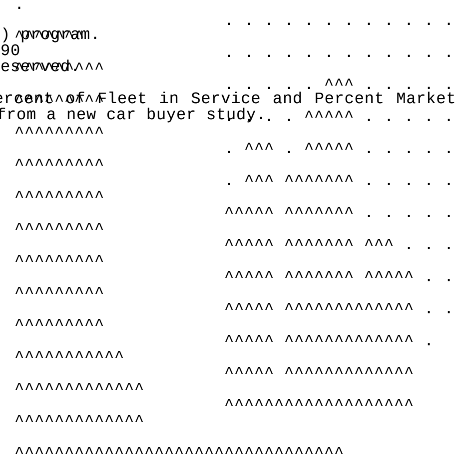


FIGURE 8. -- CATPAC of Pizza Interviews

so that activating "sweet", "brown", "carbonated" will probably activate Pepsi and Coke.

Attributes can also be connected to each other, so that activating attributes like "sweet", "satisfying", "filling" and the like may well also activate other attributes like "delicious" or "fattening."

Galileo also represents people as neurons. People can be connected to both attributes and products; they are tightly connected to attributes that make a difference to them, and they are more tightly connected to products and services that they buy and use than to those that they don't buy or use. All product development, advertising and marketing strategies can be seen as efforts to connect a product or service more tightly to people.

Galileo doesn't represent neurons as simply "connected" or "not connected" to each other, but instead measures the precise *degree* of each connection. This means that Galileo includes not only information about whether a car, for example, is smooth riding, but also represents precisely how smooth riding it is. Galileo does not simply say a product, service or object belongs to a category, but instead says *to what degree* it belongs to that category. A system which quantifies the degree to which objects belong to categories is called a "fuzzy logic."

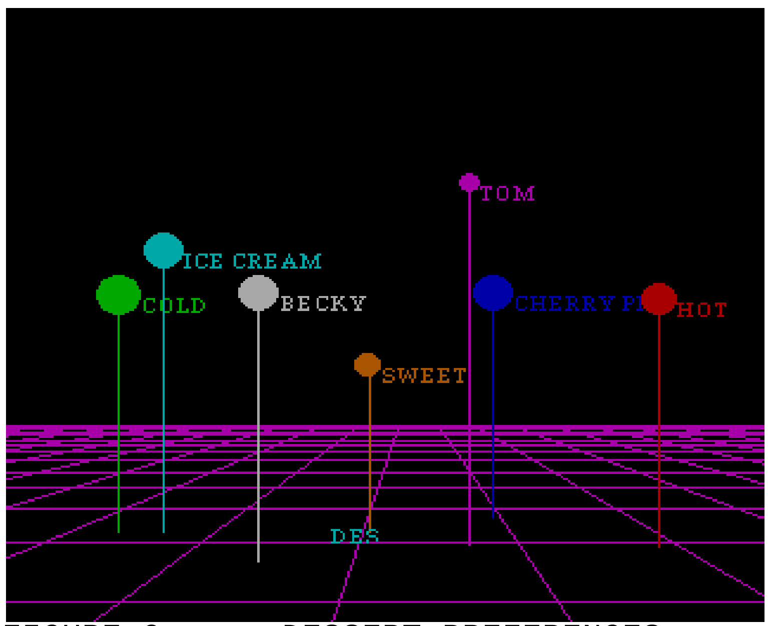


FIGURE 9. -- DESSERT PREFERENCES

In a natural neural network, neurons that are tightly connected are typically located close to one another. Galileo provides diagrams based on this principle in the form of "maps," which can help give an intuitive picture of the structure of the network.

This figure shows a map of Dessert Preferences for Tom and Becky. It shows that "Ice Cream" is closer to cold than "Cherry Pie", which is closer to "Hot." Both are about the same distance from "Sweet." Tom prefers a hot dessert, and the map shows him closer to "Hot" than "Cold." Becky prefers a cold dessert, and is closer to "Cold" in the map. She's also closer to "Ice Cream" than Tom, who is closer to "Cherry Pie" than Becky. We should expect Tom to choose Cherry Pie more often than Becky, while Becky would be expected to choose Ice Cream more often than Tom. We'd also expect Tom to choose Cherry Pie more often than Ice Cream, and Becky to choose Ice Cream more often than Cherry Pie.

While the map is useful for getting an intuitive feel for the structure of the network, more precise information is always available. Galileo can write out any distances desired in a simple format, as this table shows.

One of the major reasons Galileo has been so widely used in advertising and market research is its ability to calculate optimum strategies for strengthening and weakening connections between the neurons.

MEAN GALILEO DISTANCES

Tom and Becky's Desert Preferences

CHERRY PIE

Attribute	Distance	N
HOT	16.50	2.
COLD	76.50	2.
SWEET	32.00	2.
TOM	24.00	2.
BECKY	56.00	2.

ICE CREAM

Attribute	Distance	N
HOT	98.50	2.
COLD	8.50	2.
SWEET	41.00	2.
TOM	55.00	2.
BECKY	19.00	2.

FIGURE 10. -- TOM AND BECKY'S DESSERT PREFERENCES

Using Galileo's strategic planning abilities, it's possible to find strategies which will strengthen the connection of a product or service with its potential customers. In order to design an effective strategy for repositioning a product or service in the customers' minds, it is only necessary to specify what position in the market the product or service is meant to fill. Galileo software will automatically calculate what connections need to be strengthened and which weakened to achieve the desired positioning.

This table shows several examples of strategies developed by Galileo to reposition Ice Cream closer to Tom.

```
ICE CREAM
ASSOCIATED  WITH
------------------------------
  13.667 PERCENT  = HOT
 -15.667 PERCENT  = SWEET
  70.667 PERCENT  = CHERRY PIE

VALUE OF MINIMUM =      1.060

ICE CREAM
ASSOCIATED  WITH
------------------------------
  32.333 PERCENT  = HOT
  31.583 PERCENT  = CHERRY PIE
  36.083 PERCENT  = BECKY

VALUE OF MINIMUM =     14.395

ICE CREAM
ASSOCIATED  WITH
------------------------------
 -21.167 PERCENT  = SWEET
  73.917 PERCENT  = CHERRY PIE
   4.917 PERCENT  = BECKY

VALUE OF MINIMUM =     13.664
```

FIGURE 11. -- OPTIMUM STRATEGIES TO REPOSITION ICE CREAM CLOSER TO TOM

The first strategy Galileo suggests tightens the connection between Ice Cream and both "Hot" and "Cherry Pie", while weakening the connection between Ice Cream and "Sweet."

The second strategy suggests tightening the connections between Ice Cream and "Hot," "Cherry Pie", and Becky. The third suggests weakening the connection between Ice Cream and "Sweet", while tightening the connections to "Cherry Pie" and Becky.

There are as many ways of strengthening and weakening connections as human imagination can devise, but the most common are advertising and actually changing the product or service. However one proceeds to implement the strategies, Galileo provides a convenient way to track the progress of the repositioning. Galileo can take data from directly from text using CATPAC, or from industry standard quantitative measurements, including complete paired comparison ratio scales for extremely precise results.

ORESME

ORESME is a self-organizing neural network that simulates the cognitive processes of individuals or groups of people, such as markets or market segments. ORESME represents objects, products, attributes, people, or any other concept as neurons in a network. Mentioning one or more of these objects (as one would in an advertisement) activates the neurons which represent those objects. These activated neurons in turn activate those other neurons to which they are closely connected, while turning off those neurons to which they are negatively connected. This *interactive activation and competition network* thus simulates the process by which one or more ideas stimulates still other ideas.

Threshold = .000 Restoring Force = .250 Damping Factor = .000

Concept	Cycles X 1									
	1	2	3	4	5	6	7	8	9	10
SPORTY LOOKING	.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
FUN TO DRIVE	1.0	.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
FAMILY CAR	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
GOOD VALUE	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
PRACTICAL	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
AFFORDABLE	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
EXCITING	.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
APPEALS TO OLDER PEO	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
LUXURIOUS	.0	.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
RELIABLE	1.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
HONDA ACCORD	.0	1.0	.0	.0	.0	.0	.0	.0	.0	.0
SUBARU LEGACY	1.0	1.0	1.0	1.0	.0	.0	.0	.0	.0	.0
FORD TEMPO	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
TOYOTA CAMRY	.0	1.0	.0	.0	.0	.0	.0	.0	.0	.0
NISSAN STANZA	.0	1.0	.0	.0	.0	.0	.0	.0	.0	.0
CHRYSLER LEBARON GTS	.0	.0	.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
PONTIAC GRAND AM	.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
YOURSELF	.0	1.0	.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

This hypothetical analysis indicates that advertising SUBARU LEGACY as "fun to drive" and "reliable" might increase its appeal in the short run, but might eventually result in the decision to buy a PONTIAC GRAND AM or a CHRYSLER LEBARON GTS in the longer term.

FIGURE 12. -- Example of ORESEME output file.

ORESME can be helpful in alerting advertisers to the potential problems which might arise from unexpected connotations of otherwise useful message strategies. ORESME can accept inputs from CATPAC or GALILEO, or can develop its own network interactively.

SUPERVISED NETWORKS

By far the most commonly used neural networks are supervised networks, usually back propagation models. These models are of great generality, but for the most part are most useful when your goal is to make predictions, when you have a poor or inaccurate theory on which to base your guesses, when the information you do have is in the form of *cases*, and that information is incomplete or poorly measured.

A prototypical case is bank loan applications. Here the bank can provide records of many previous cases from files of applicants and their attributes, along with an indication of whether they defaulted their loan or not. The supervised network then "learns" to predict the outcome of these cases, and can be tested for its predictive ability by being asked to predict the outcome of other cases from the bank's files. Once trained, the network becomes an expert loan risk appraiser. Other common tasks for supervised networks include the prediction of the outcome of football games or horse races, stock market fluctuations, classification of disease from symptoms, equipment failure diagnoses and similar problems.