SUSTAIN: A Model of Human Category Learning

Bradley C. Love and Douglas L. Medin
Northwestern University

SUSTAIN (Supervised and Unsupervised STratified Adaptive Incremental Network) is a network model of human category learning. SUSTAIN is a three layer model where learning between the first two layers is unsupervised, while learning between the top two layers is supervised. SUSTAIN clusters inputs in an unsupervised fashion until it groups input patterns inappropriately (as signaled by the supervised portion of the network). When such an error occurs, SUSTAIN alters its architecture, recruiting a new unit that is tuned to correctly classify the exception. Units recruited to capture exceptions can evolve into prototypes/attractors/rules in their own right. SUSTAIN's adaptive architecture allows it to master simple classification problems quickly, while still retaining the capacity to learn difficult mappings. SUSTAIN also adjusts its sensitivity to input dimensions during the course of learning, paying more attention to dimensions relevant to the classification task. Shepard, Hovland, and Jenkins's (1961) challenging category learning data is fit successfully by SUSTAIN. Other applications of SUSTAIN are discussed. SUSTAIN is compared to other classification models.

Introduction

Some categories have a very simple structure, while others can be complex. Accordingly, learning how to properly classify items as members of category "A" or "B" can be almost trivial (e.g., the value of a single input dimension determines membership) or can be so difficult that no regularity is discovered (e.g., rote memorization of every category member is required to determine membership).

Classifications are harder to master when the decision boundary (in a multi-dimensional space of possible inputs) is highly irregular and when there are multiple boundaries (e.g., all the members of category "A" do not fall inside one contiguous region of the input space). Difficult classification problems (problems with complex decision boundaries) typically involve categories that have a complex internal structure, perhaps consisting of multiple prototypes (i.e., category subtypes) and a number of exceptions. Linguistic analyses have demonstrated that many categories have a rich internal structure (Lakoff, 1987). Very simple learning models will fail to master difficult categorizations with complex boundaries (i.e., categories with rich internal structure). For instance, a purely linear model, like the perceptron (Rosenblatt, 1958), will be unable to master a classification when the mapping from input features to category labels is nonlinear.

Interestingly, a complex nonlinear model, such as a back-propagation model (Rumelhart, Hinton, & Williams, 1986) with many hidden units, can learn complex decision boundaries but will perform poorly on a simple problem (e.g., a problem where the decision boundary is linear). In such cases, the more complex model will generalize poorly by over-fitting the training data. Thus, making a model too powerful or too weak is undesirable. Geman, Bienenstock, and Doursat (1992) termed this tradeoff between data fitting and generalization as the bias/variance dilemma. In brief, when a network is too simple it is overly biased and cannot learn the correct boundaries. Conversely, when a network is too powerful, it masters the training set, but the boundaries it learns are somewhat arbitrary and are highly influenced by the training sample, leading to poor generalization.

Unfortunately, the complexity of learning models is usually fixed prior to learning. For instance, in network models, the number of intermediate level processing units (which governs model complexity) must usually be chosen in advance. The problem may not be avoidable by treating the number of intermediate units as an additional parameter, because certain architectures may be preferable at certain stages of the learning process. For example, Elman (1994) provides computational evidence (which seems in accord with findings from developmental psychology) that beginning with a simple network and adding complexity as learning progresses improves overall performance.

Models with an adaptive architecture (like SUSTAIN), do not need to specify the number of intermediate units prior to learning. Some models (including SUSTAIN) begin with a small network and expand the network when necessary. Most methods expand the network when overall error (the difference between desired and observed output) is high. For example, the cascade-correlation model (Fahlman & Lebiere, 1990) expands the network vertically with additional intermediate layers, creating higher-order feature detectors. Other

 B. C. Love was supported by the Office of Naval Research under the National Defense Science and Engineering Graduate Fellowship Program. We would like to thank the reviewers for their excellent comments. Correspondence concerning this research should be addressed to loveb@nwu.edu or Bradley C. Love, Department of Psychology, Northwestern University, Evanston, IL 60208

models expand horizontally when error is high (Ash, 1989; Azimi-Sadjadi, Sheedvash, & Trujillo, 1993).

Unlike the aforementioned models, SUSTAIN does not accrue units based on overall error. Instead, SUSTAIN adds a new intermediate level unit when the unsupervised part of the network clusters input patterns in a manner deemed inappropriate by the supervised part of the network. This happens when two input patterns (that differ) belong to the same cluster and the differences between the two input patterns proves critical for successfully mastering the classification. When such an error occurs, SUSTAIN splits the cluster into two clusters by adding an intermediate unit. Thus, intermediate level units in SUSTAIN encode the prototypes (a category can have multiple prototypes) and exceptions of the categories being learned. The method for adding units in SUSTAIN is psychologically motivated by the intuition that people ignore differences when they can (a bias towards simple solutions), but will note differences when forced to by environmental feedback. Additionally, intermediate level units in SUSTAIN are intended to be psychologically real. We claim that SUSTAIN acquires and modifies its prototypes and exceptions in a manner analogous to how people infer a category's internal structure.

Another aspect of networks that is usually fixed, but should vary depending upon the nature of the learning problem, is the activation function of an intermediate level unit. In backpropagation networks, the steepness of a hidden unit's sigmoidal shaped activation function is set as a parameter. In models where an intermediate level unit's activation function is viewed as a receptive field (e.g., Poggio & Girosi, 1990; Kruschke, 1992), the shape of a unit's receptive field is set as a parameter.

An intermediate level unit in SUSTAIN integrates the responses from multiple receptive fields (each subcateogry unit has a receptive field for each input dimension). SUSTAIN treats the shape of a receptive field as something to be learned, rather than as a parameter. SUSTAIN assumes that receptive fields are initially broadly tuned and are adjusted during the course of learning to maximize the receptive field's response to inputs. Intermediate units with peaked (narrow) receptive fields can be described as highly focused. Receptive fields that develop tighter tunings are capable of stronger responses to stimuli (see Figure 1). As an outcome of learning how to perform a classification, SUSTAIN learns which dimensions of the stimuli are relevant and should be attended to. Conceiving of attention as enhancing the tuning of cells is consistent with current work on the neural basis of attention (Treue & Maunsell, 1996).

An Overview of SUSTAIN

SUSTAIN consists of three layers: input, subcategory, and category. Input layer units take on real values to encode information about the environment (e.g., the encoding of a stimulus item that needs to be classified as belonging to category

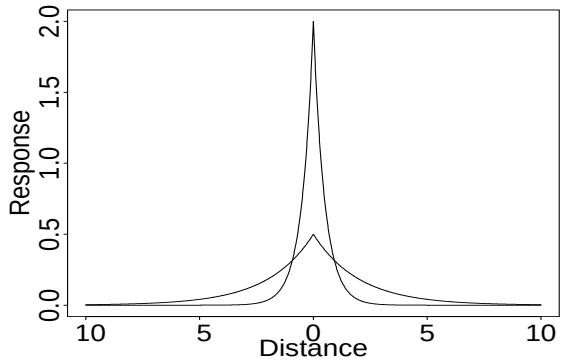

Figure 1. Both units respond maximally when a stimulus appears in the center of their receptive field (a .5 response for the broadly tuned unit; a 2.0 response for the tightly tuned unit). Compared to the broadly tuned unit, the tightly tuned unit's response is stronger to stimuli close to the center and is weaker for stimuli farther from the center (the crossover point occurs at a distance from center of .9 (approximately).

"A" or "B"). Units in the subcategory layer (the intermediate layer) encode the prototypes and exceptions of the category units. Subcategory units compete with one another to respond to patterns at the input layer with the winner (the subcategory unit that is most active) being reinforced. Weights are adjusted according to the Kohonen unsupervised learning rule for developing self-organizing maps (Kohonen, 1984). When a subcategory unit "wins" the centers of its receptive fields (there is a receptive field for each input dimension) move in the direction of the input pattern, minimizing the distance between the centers and the input pattern. This method is similar to a number of clustering techniques used for classification and pattern recognition, such as maximum-distance, K-means, and isodata (Tou & Gonzalez, 1974; Duda & Hart, 1972).

One novel aspect of our implementation is that this unsupervised learning procedure is combined with a supervised procedure. When a subcategory unit responds strongly to an input pattern (i.e., it is the winner) and has an excitatory connection to the inappropriate category unit (i.e., the subcategory unit predicts "A" and the correct answer is "B"), the network shuts off the subcategory unit and recruits a new subcategory unit that responds maximally to the misclassified input pattern (i.e., the new unit's receptive fields are centered upon the input pattern).[1]

The process continues with the new unit competing with the other subcategory units to respond to input patterns with the position of the winner's receptive fields being updated, as well as its connection to the category units by the one layer delta learning rule (Rumelhart et al., 1986). At a minimum,

[1]Initially the network only has one subcategory unit that is centered upon the first input pattern.

there must be as many subcategory units as category units when category responses are mutually exclusive.

Previous proposals that bear some resemblance to SUSTAIN include counterpropagation networks (Hecht-Nielsen, 1988) which are multilayer networks where the Kohonen learning rule is used for the bottom two layers. Simpson has explored a supervised version of the Kohonen network where the model does not determine which cluster is the winner, but is told (Simpson, 1989). This change greatly speeds up learning. Interestingly, our approach to clustering is not properly characterized as being either supervised or unsupervised. Clustering is unsupervised unless the network makes a serious clustering error (i.e., an incorrect prediction). A serious error leads to the creation of a new cluster; otherwise learning at the subcategory layer is completely unsupervised.

Another interesting aspect of SUSTAIN's subcategory units is that in addition to adjusting the centers (i.e., the position) of their receptive fields, the sensitivities (i.e., the shape) of their receptive fields also are adjusted in response to input patterns. Input units (i.e., dimensions of the input pattern) that provide consistent evidence (i.e., the position of the subcategory units' receptive fields for that dimension does not have to be adjusted often), develop tighter tunings (see Figure 1). These more reliable input dimensions receive more attention. SUSTAIN uncovers (and explicitly represents) which dimensions are relevant for classification.

Mathematical Formulation

Receptive fields have an exponential shape with a receptive field's response decreasing exponentially as distance from its center increases:

$$\alpha(\mu) = \lambda e^{-\lambda\mu} \tag{1}$$

where λ is the tuning of the receptive field, and μ is the distance of the stimulus from the center of the field. Arguments for activation dropping off exponentially can be found in (Shepard, 1987).

While receptive fields with different λ have different shapes, for any λ, the area "underneath" a receptive field is constant:

$$\int_0^\infty \alpha(\mu)d\mu = \int_0^\infty \lambda e^{-\lambda\mu}d\mu = 1. \tag{2}$$

For a given μ, the λ that maximizes $\alpha(\mu)$ can be computed by differentiating:

$$\frac{\partial\alpha}{\partial\lambda} = e^{-\lambda\mu}(1-\lambda\mu). \tag{3}$$

These properties of exponentials prove useful in formulating SUSTAIN.

The activation of a subcategory unit is given by:

$$A_{H_j} = \frac{\sum_{i=1}^{n}(\lambda_i)^r e^{-\lambda_i\mu_{ij}}}{\sum_{i=1}^{n}(\lambda_i)^r} \tag{4}$$

where n is the number of input units, λ_i is the tuning of each subcategory unit's receptive field for the ith input dimension, μ_{ij} is the distance between the center of subcategory unit j's receptive field for the ith input unit and the output of the ith input unit (distance is simply the absolute value of the difference of these two terms), and r is an attentional parameter (always nonnegative). When r is high, input units with tighter tunings (units that seem relevant) dominate the activation function. Equation 4 sums the responses of the receptive fields for each input dimension and normalizes the sum. The activation of a subcategory unit is bound between 0 (exclusive) and 1 (inclusive).

Subcategory units compete to respond to input patterns and in turn inhibit one another. When many subcategory units are strongly activated, the output of the winning unit is less. Units inhibit each other according to:

$$O_{H_j} = \frac{(A_{H_j})^\beta}{\sum_{i=1}^{m}(A_{H_i})^\beta}A_{H_j} \tag{5}$$

where β is the lateral inhibition parameter (always nonnegative) and m is the number of subcategory units. When β is small, competing units strongly inhibit the winner. When β is high the winner is weakly inhibited. Units other than the winner have their output set to zero.[2]

After feedback is provided by the "experimenter", if the winner predicts the wrong category, its output is set to zero and a new unit is recruited:

$$\text{for all } j \text{ and } k, \text{ if } (t_k O_{H_j} w_{jk} < 0), \text{ then recruit a new unit} \tag{6}$$

where t_k is the target value for category unit k and w_{jk} is the weight from subcategory unit j to category unit k. When a new unit is recruited its receptive fields are centered on the misclassified input pattern and the subcategory units' activations and outputs are recalculated.

If a new subcategory unit is not created, the centers of the winner's receptive fields are adjusted:

$$\Delta w_{ij} = \eta(O_{I_i} - w_{ij}) \tag{7}$$

where η is the learning rate, O_{I_i} is the output of input unit i. The centers of the winner's receptive fields move towards the input pattern according to the Kohonen learning rule. This learning rule centers the prototype (i.e., the cluster's center) amidst the members of the prototype.

Using our result from Equation 3, receptive field tunings are updated according to:

$$\Delta\lambda_i = \eta e^{-\lambda_i\mu_{ij}}\left(1-\lambda_i\mu_{ij}\right). \tag{8}$$

Only the winning subcategory unit updates the value of λ_i. Equation 8 adjusts the shape of the receptive field for each

[2]The model (as specified) can have multiple winners. For instance, there could always be two winners. More complex schemes could also be considered for determining the number of winners. We do not explore any of these possibilities because they are less conceptually clear and the data does not demand it.

input so that each input can maximize its influence on subcategory units. Initially, λ_i is set to be broadly tuned. For example, if input unit i takes on values between -1 and 1, the maximum distance between the ith input unit's output and the position of a subcategory unit's receptive field (for the ith dimension) is 2, so λ_i is set to .5 because that is the optimal setting of λ_i for μ equal to 2 (i.e., Equation 8 equals zero).

Activation is spread from the winning subcategory unit to the category units:

$$A_{C_k} = O_{H_j} w_{jk} \tag{9}$$

where A_{C_k} is the activation of the kth category unit and O_{H_j} is the output of the winning subcategory unit.

The output of a category unit is given by:

$$\begin{aligned} &\text{if } (C_k \text{ is nominal and } |A_{C_k}| > 1), \text{ then } O_{C_k} = \tfrac{A_{C_k}}{|A_{C_k}|} \\ &\text{else } O_{C_k} = A_{C_k} \end{aligned} \tag{10}$$

where O_{C_k} is the output of the kth category unit. If the feedback given to subjects concerning C_k is nominal (e.g., the item is in category "A" not "B"), then C_k is nominal. Kruschke (1992) refers to this kind of teaching signal as a "humble teacher" and explains when its use is appropriate.

When a subcategory unit is recruited, weights from the unit to the category units are set to zero. The one layer delta learning rule (Rumelhart et al., 1986) is used to adjust weights these weights:

$$\Delta w_{jk} = \eta(t_k - O_{C_k}) O_{H_j}. \tag{11}$$

Note that only the winner will have its weights adjusted since it is the only subcategory unit with a nonzero output.

The following equation determines the response probabilities (for nominal classifications):

$$Pr(k) = \frac{(O_{C_k} + 1)^d}{\sum_{i=1}^{p} (O_{C_i} + 1)^d} \tag{12}$$

where d is a response parameter (always nonnegative) and p is the number of category units. The category unit with the largest output is almost always chosen when d is large. In Equation 12, one is added to each category unit's output to avoid performing calculations over negative numbers. The Luce choice rule is a special case ($d = 1$) of this decision rule (Luce, 1963).

Empirically Testing SUSTAIN

Critiques of computational models have historically focused on what functions are learnable (e.g., Minsky & Papert, 1969). This trend continues with Hornik, Stinchcombe, and White (1990) proving that backpropagation networks are universal approximators (given enough hidden units) and Poggio and Girosi (1990) demonstrating similar results for their model. Unfortunately, researchers have not focused on the *time course* of learning. A more informative test of a model's performance requires examining which functions (i.e., classifications) a model can easily learn and which functions are difficult to master. For a model of human category learning, it is not sufficient to show that a model can learn some function (e.g., logical XOR), but one must show that a model can match the learning curves of human subjects over a variety of functions, using the same parameter values. As models become more sophisticated, fitting a diverse set of studies with the same parameter values may prove to be a useful test of models.

In accord with this stance, SUSTAIN is fit to a variety of human learning data (here we focus on Shepard et al. (1961)) using the same parameter values: $\eta = .1$, $\beta = 1.0$, $r = 3.5$, and $d = 8.0$ (Love & Medin, 1998). The parameters were determined by beginning with $\eta = .1$, $\beta = 1.0$, $r = 1.0$, and $d = 1.0$ and adjusting them by hand until a good qualitative fit of the data was achieved. Because each of SUSTAIN's parameters has an intuitive meaning, it was easy to fit the data. For example, when we noticed SUSTAIN was not sufficiently biased towards solutions focusing on a small number of input dimensions, the value of the attentional parameter r was increased. When we noticed overall accuracy was too low, the value of then decision parameter d was increased until SUSTAIN sufficiently stressed accuracy.

Modeling Shepard et al. (1961)

Shepard et al.'s (1961) classic experiments on human category learning provided challenging data to fit. Subjects learned to classify 8 objects that varied on three binary dimensions (shape, size, and color) into two categories (four items per category). On every trial, subjects assigned the stimulus to a category and feedback was provided. Subjects trained for 16 blocks (each object was shown twice per block in a random order) or until they completed two consecutive blocks without an error. Six different assignments of objects to categories were tested with the six problems varying in difficulty (Type I was the easiest to master, Type VI the hardest). The logical structure of the six problems is shown in Table 1. The Type I problem only requires attention along one input dimension, while the Type II problem requires attention to two dimensions (Type II is XOR with an irrelevant dimension). Types III-V require attention along all three dimensions but some regularities exist (Types III-V can be classified as rule plus exception problems). Type VI requires attention to all three dimensions and has no regularities across any pair of dimensions.

Nosofsky et al. (1994) replicated Shepard et al. (1961) with more subjects and traced out learning curves. Figure 2 shows the learning curves for the six problem types. The basic finding is that Type I is learned faster than Type II which is learned faster than Types III-V which are learned faster than VI. This data is particularly challenging for learning models as most models fail to predict Type II easier than Types III-V.

Table 1
The logical structure of the six classification problems tested in Shepard et al. (1961) is shown. The physical attributes (e.g., large, dark, triangle, etc.) were randomly assigned to an input dimension for each subject.

Input	I	II	III	IV	V	VI
111	A	A	B	B	B	B
112	A	A	B	B	B	A
121	A	B	B	B	B	A
122	A	B	A	A	A	B
211	B	B	A	B	A	A
212	B	B	B	A	A	B
221	B	A	A	A	A	B
222	B	A	A	A	B	A

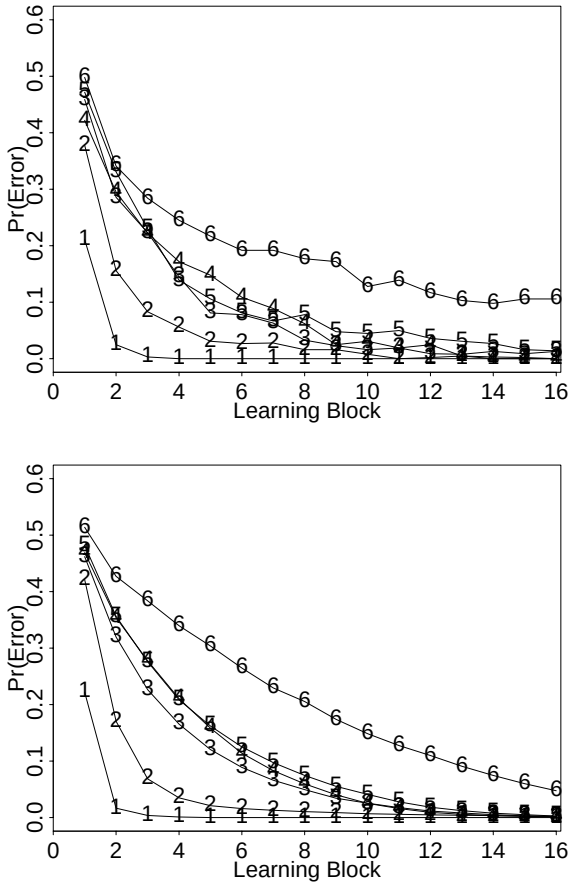

Figure 2. Each learning block consisted of two presentations of each stimulus (in a random order). Nosofsky et al.'s (1994) replication of Shepard et al. (1961) is shown on top. Below, SUSTAIN's fit of Nosofsky et al.'s (1994) data is shown (averaged over 10,000 runs on each problem).

The only models known to reasonably fit these data are ALCOVE (Kruschke, 1992) and RULEX (Nosofsky, Palmeri, & McKinley, 1994b). RULEX is designed to classify stimuli that can be represented by binary features, while ALCOVE is an exemplar based model.

SUSTAIN's fit of Nosofsky et al.'s data is also shown in Figure 2. While fitting the data (see Section Empirically Testing SUSTAIN), the same difficulty ordering of the six problem types was observed for all combinations of parameter values with the exception that for high values of d all six problem types were of equal difficulty (in this case SUSTAIN masters each problem within the first learning block).

How SUSTAIN Solves the Six Problems

SUSTAIN is not a black box and it is possible to understand how SUSTAIN solves a classification problem (perhaps gaining insight into the problem itself). Table 2 shows the number of subcategory units SUSTAIN recruited by problem type. The most common solution for the Type I problem was to create one unit for each category. Type I has a simple category structure (the value of first dimension determines membership). Accordingly, SUSTAIN solves the problem with only two subcategory units. Type II requires attention to two dimensions. SUSTAIN solved the Type II problem by allocating two units to each category. Each subcategory unit responded to two input patterns, largely ignoring the irrelevant dimension. Because category members are highly dissimilar (e.g., 121 and 212 are in the same category), SUSTAIN formed two clusters for each category (ignoring differences on the irrelevant dimension). Types III-V can be roughly characterized as imperfect rule plus exception categories. SUSTAIN solved these problems by uncovering regularities and memorizing exceptions (devoting a unit for one pattern). Type VI has no regularities that can be exploited, forcing SUSTAIN to "memorize" each pattern (i.e., SUSTAIN devoted a subcategory unit to each input pattern).[3]

The right column of Table 2 shows the mean λ value (averaged over the three input dimensions) at the end of the second block for each problem.[4] The sharpness of the mean tuning was positively correlated with the number of subcategory units recruited. Only one dimension develops a sharp tuning

[3]Occasionally, SUSTAIN recruited nine subcategory units (one more than the number of input patterns). This occurred when a subcategory unit responding to one input pattern was "stolen" by another input pattern belonging to the same category (i.e., the subcategory unit temporarily responded to two input patterns). Because no regularities exist in the Type VI problem, each subcategory unit can only encode one input pattern. The input pattern whose subcategory unit was "stolen" is forced to recruit a new subcategory unit.

[4]SUSTAIN's mean tunings are reported after two learning blocks because some runs reached criterion at that point. Differences in tunings between the six conditions are magnified when later blocks are examined.

Table 2
SUSTAIN's Final Architecture and mean λ (2nd block).

Problem Type	Mean Subcategory Units	Mean λ
I	2.2	2.0
II	4.3	2.8
III	5.9	3.0
IV	6.3	3.1
V	6.5	3.2
VI	8.2	3.5

in the Type I problem (the network learns the other two dimensions are irrelevant), while all three dimensions develop a sharp tuning in the Type VI problem (the network learns all three dimensions are highly relevant).

Discussion

SUSTAIN is an adaptive architecture, tailoring its architecture to the problem at hand. It is motivated by the basic psychological notion that people prefer general solutions and ignore distinctions when possible. SUSTAIN's potential is highlighted by its successfully fit of Shepard et al.'s (1961) six problem types. While this task uses binary input dimensions, SUSTAIN is not restricted to this input format. SUSTAIN has been successfully applied to Billman & Knutson's (1996) unsupervised learning data and Medin, Gerald, and Murphy's (1983) data on item and category learning where input patterns consist of attributes that are mulitvalued (Love & Medin, 1998).

References

Ash, T. (1989). Dynamic node creation in backpropagation networks. *Connection Science*, *1*(4), 365–375.

Azimi-Sadjadi, M. R., Sheedvash, S., & Trujillo, F. O. (1993). Recursive dynamic node creation in multilayer neural networks. *IEEE Transactions on Neural Networks*, *4*(2), 242–256.

Billman, D. & Knutson, J. (1996). Unsupervised concept learning and value systematicity: A complex whole aids learning the parts. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, *22*(2), 458–475.

Duda, R. O. & Hart, P. E. (1972). *Pattern Classification and Scene Analysis*. New York: Wiley.

Elman, J. L. (1994). Implicit learning in neural networks: The importance of starting small. In C. Umilta & M. Moscovitch (Eds.), *Attention and performance XV: Conscious and nonconscious information processing.*, pp. 861–888. Cambridge, MA: MIT Press.

Fahlman, S. E. & Lebiere, C. (1990). The cascade-correlation learning architecture. In D. S. Touretzky (Ed.), *Advances in Neural Information Processing Systems 2. Proceedings of the 1989 Conference*, pp. 524–532. San Mateo, CA: Morgan Kaufmann.

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, *4*(1), 1–58.

Hecht-Nielsen, R. (1988). Applications of counterpropagation networks. *Neural Networks*, *1*(2), 131–139.

Hornik, K., Stinchcombe, M., & White, H. (1990). Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural Networks*, *3*(5), 551–560.

Kohonen, T. (1984). *Self-Organization and Associative Memory*. Berlin, Heidelberg: Springer. 3rd ed. 1989.

Kruschke, J. K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review*, *99*(1), 22–44.

Lakoff, G. (1987). *Women, fire, and dangerous things: What categories reveal about the mind*. Chicago: University of Chicago Press.

Love, B. C. & Medin, D. L. (1998). A model of human category learning. In Preparation.

Luce, R. D. (1963). Detection and recognion. In R. D. Luce, R. R. Busg, & E. Galanter (Eds.), *Handbook of Mathematical Psychology*, pp. 103–189. New York: Wiley.

Medin, D. L., Gerald, G. I., & Murphy, T. D. (1983). Relationships between item and category learning: Evidence that abstraction is not automatic. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, *9*, 607–625.

Minsky, M. L. & Papert, S. (1969). *Perceptrons: An introduction to computational geometry*. Cambridge, MA: MIT Press.

Nosofsky, R. M., Gluck, M. A., Palmeri, T. J., McKinley, S. C., & Glauthier, P. (1994a). Comparing models of rule based classification learning: A replication and extension of Shepard, Hovland, and Jenkins (1961). *Memory & Cognition*, *22*, 352–369.

Nosofsky, R. M., Palmeri, T. J., & McKinley, S. C. (1994b). Rule-plus-exception model of classification learning. *Psychological Review*, *101*(1), 53–79.

Poggio, T. & Girosi, F. (1990). Regularization algorithms for learning that are equivalent to multilayer networks. *Science*, *247*, 978–982.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage in the brain. *Psychological Review*, *65*, 386–408.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*, 533–536.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science*, *237*, 1317–1323.

Shepard, R. N., Hovland, C. L., & Jenkins, H. M. (1961). Learning and memorization of classifications. *Psychological Monographs*, *75*(13, Whole No. 517).

Simpson, P. K. (1989). *Artificial Neural Systems*. Elmsford, NY: Pergamon Press.

Tou, J. T. & Gonzalez, R. C. (1974). *Pattern Recognition Principles*. Reading: Addison-Wesley.

Treue, S. & Maunsell, J. H. R. (1996). Attentional modulation of visual motion processing in cortical areas mt and mst. *Nature*, *382*(6591), 539–541.