Using Machine Learning to Understand and Influence Human Categorization Behavior

by

Bryan R. Gibson

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN–MADISON

2015

Date of final oral examination: 7/31/2015

The dissertation is approved by the following members of the Final Oral Committee:
Xiaojin Zhu, Associate Professor, Computer Sciences
Mark Craven, Professor, Biostatistics and Medical Informatics & C.S.
Timothy R. Rogers, Professor, Psychology
Bilge Mutlu, Associate Professor, Computer Sciences & Industrial Engin.
Martina Rau, Assistant Professor, Educational Psychology & C.S.

To all those that have come before and all those that will follow.

ACKNOWLEDGMENTS

I would like to first acknowledge my supervisor, Professor Jerry Zhu. Without his tireless help, endless patience and boundless insight, I would never have completed this work. It is to him that I owe my introduction to the collaboration between Cognitive Psychology and Machine Learning that has formed the basis of this thesis. Thank you to him for giving me this incredible opportunity.

Thank you to the group of cross-discipline collaborators I have had the amazing opportunity to work with. Firstly, Tim R. Rogers and Chuck W. Kalish, without whose persistence and patience translating between fields this work would not be possible. Additionally I've had the greatest pleasure of working with Joseph Harrison, Dave Andrzejewski, Andrew Goldberg, Kwang-Sung Jun, Junming Sui, Ming Li, Shike Mei, Arthur Glenberg, Jonathan Willford, Mark Joseph, Pradeep Muthukrishnan, Mark Hodges, Anthony Fader, Mark Joseph, Joshua Gerrish, Mark Schaller, Jonathan dePeri.

Many thinks to Prof. Drago Radev for giving me work to fill the time, introducing me to ML and then asking "Why aren't you in grad school?" and many thanks to Jessica Hullman, another amazing collaborator, for introducing me to him. Thank you to Prof. Lada Ademic and Prof. Steven Abney for allowing me to sit in on their classes before I was a proper student and be a complete, if interested, nuisance.

Thank you to Carly Mertez for helping me get started and being an incredible inspiration on how to persevere, produce and succeed. Thank you to Katie O'Donnell for providing endless emotional and gastronomic support through the writing of this thesis, as well

as creating the only environment in NYC in which I could write. Thank you to Lizzy Magarian for being a constant cheerleader throughout and pushing me on. Thank you to Bret and Scott Paysuer for providing fantastic role models for how to be a successful dissertator. Thank you to Erin Woodward for her help with editing, wading through almost unintelligible technical language. And thank you to the many, many other friends I've made along the way.

A giant thank you to all of the faculty and staff at University of Wisconsin who have made the process such a joy to be a part of.

Thank you to my grandfather, Prof. Walter Waring, for providing the inspiration to teach and for all of my incredible students for pushing me to be a better teacher. And thank you to all of my family, especially my aunt and uncle Cathy and Rudiger Grimm for their Olympic level of support when I needed it.

Lastly thank you to my mom, Mary Gibson, and my dad and step-mom, Rich and Deb Gibson, for all of the love, support and for checking in and providing that extra little kick just when I needed it. And thank you to my wonderful, multi-talented, amazing sister Laura. Thank you for being there, always.

— Bryan R. Gibson (2015)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

In both machine learning (ML) and cognitive psychology (CP), categorization is considered a basic task commonly encountered by learning agents and studied in both fields. While a great deal of work in CP has been applied to understanding human learning in *supervised* categorization, little work has been done previously to investigate the effects of both labeled and unlabeled data as in the *semi-supervised* setting. I have had the opportunity to contribute to a number of studies investigating just this situation: human learners tasked with learning a categorization task from some combination of labeled and unlabeled data. This work has involved the use of ML to both (1) better understand how labeled and unlabeled data affect human learners in categorization tasks as well as (2) attempt to influence the resulting behavior using ideas and techniques derived from ML.

The results of this work have shown that (1) in addition to humans being affected by the distribution of unlabeled data, they can also be affected by ordering of the unlabeled items (2) that humans are not constrained in their search of a parameter space when attempting to integrate new unlabeled items with previously labeled experience (3) that humans can learn using underlying manifold structure (4) that the speed of human learning on a supervised task can be affected by prior unlabeled experience and (5) that, using Co-Training constraints, human collaborators can be induced to learn a boundary neither would have likely learned on their own.

1 CLASSIFICATION AS MODEL OF HUMAN CATEGORIZATION

Classification is one of the principal tasks investigated in the field of Machine Learning (ML). The same can be said of the field of Cognitive Psychology (CP), where the same task is known as categorization, the only difference being that the learners under investigation are human beings instead of computer programs. CP has long had an interest in understanding human categorization (also known as human category learning): How we come to conceive of objects in the world as belonging to different categories, and how we use categories to draw inferences about the unobserved properties of objects. To this end a suite of computational models have been devised to model human behavior.

These cognitive models bear a striking resemblance to models designed in the study of ML, even though the goals of the two fields are strikingly different and evolved separately for the most part. In ML, the primary goal is to develop models which generalize from a set of training data to data on which they are tested. In human category learning, however, the primary goal is to create models which produce behavior which matches that of humans in the same task. The similarity of these models begs the question: Can we use statistically-motivated ML models to better understand human behavior?

For the most part, the experimental setting investigated in human category learning has been a Supervised Learning (SL) setting, where the learner is presented with a set of $\langle \mathtt{item}, \mathtt{label} \rangle$ pairs and is asked to learn some mapping from item to label. This use of SL as an experimental procedure has proven exceedingly fertile – countless experiments in this setting have been conducted and a vast array of interesting regularities in human behavior have been documented (Pothos and Wills, 2011).

Only relatively recently have experiments been performed on humans investigating the effect of *labeled* plus *unlabeled* data on human category learning, known as Semi-Supervised Learning (SSL); i.e. the learner is presented with some combination of $\langle \mathtt{item}, \mathtt{label} \rangle$ pairs (labeled data) along with a set of items without labels (unlabeled data). The setting is particularly interesting given that this is the natural combination of experience in real life, a combination of labeled teaching moments and unlabeled everyday experience.

My work has focused on investigating how humans are affected by these combinations of labeled and unlabeled data while performing categorization tasks. Through this work I have had the opportunity to be part of several studies which produced significant findings, all of which fall under a general thesis:

> *Human category learning, shown to be sensitive to both labeled and unlabeled data, can be both better understood and influenced using semi-supervised machine learning models.*

Before exploring these experimental results, we will review and formalize the categorization/classification task under investigation.

1.1 Review of Classification in Machine Learning and Cognitive Psychology

As mentioned previously, the primary experimental focus in the investigation of human category learning has been in the SL setting. The models developed to investigate the observed behavior have been supervised as well, in that they do not contain explicit methods for dealing with unlabeled data. Three dominant models have been developed to describe human behavior in a categorization tasks. These are: *exemplar*, *prototype* and *Rational* models. Equivalences can be shown between these models and the following ML models: *Kernel Density Estimation*, *Gaussian Mixture Models* and *Dirichlet*

Process Mixture Models, respectively. Some of the important relationships between the psychological and machine learning models have been discussed in detail by other researchers (Fried and Holyoak, 1984; Nosofsky, 1991; Ashby and Alfonso-Reese, 1995; Sanborn et al., 2006; Griffiths et al., 2007) while the SSL models discussed here were developed in Zhu, Gibson, Jun, Rogers, Harrison, and Kalish (2010).

Before proceeding, it will be useful to define the categorization task itself, to indicate very generally how mathematical models in CP and ML have been brought to bear on the task, and to introduce some notation.

The standard categorization task asks a learner to label a previously unseen item x_t after viewing a set of labeled examples $\{(x_i, y_i)\}_{i=1}^{t-1}$. In this notation, x_t indicates a multidimensional feature vector that describes a single stimulus item (with t indexing the order in which items are seen over time), and y_t indicates the category label associated with each item. In both CP and ML, the probabilistic way of modeling human category decisions for x_t at time t is to calculate $P(y_t = k \mid x_t, \{(x_i, y_i)\}_{i=1}^{t-1})$, that is, the probability that a person will choose label $y_t = k$ for each of $k \in K$ categories given the current item x_t and the preceding labeled evidence $\{(x, y)\}_{i=1}^{t-1}$ i.e., the training examples viewed prior to the query item x_t.

A common way to compute the probability $P(y_t = k \mid x_t, \{(x_i, y_i)\}_{i=1}^{t-1})$ is via the Bayes rule. Formally the Bayes rule states

$$P(y_t = k \mid x_t, \{(x_i, y_i)\}_{i=1}^{t-1}) = \frac{P(x_t \mid y_t = k, \{(x_i, y_i)\}_{i=1}^{t-1}) P(y_t = k \mid \{(x_i, y_i)\}_{i=1}^{t-1})}{\sum_{k'} P(x_t \mid y_t = k', \{(x_i, y_i)\}_{i=1}^{t-1}) P(y_t = k' \mid \{(x_i, y_i)\}_{i=1}^{t-1})}. \tag{1.1}$$

The first term in the numerator, $P(x_t \mid y_t = k, \{(x_i, y_i)\}_{i=1}^{t-1})$, is the *likelihood*, which specifies the probability of observing item x_t assuming it has the label $y_t = k$. The second term in the numerator, $P(y_t = k \mid \{(x_i, y_i)\}_{i=1}^{t-1})$, is the *prior*, which specifies the probability, prior to observing x_t, that x_t will have label $y_t = k$. The left-hand side, $P(y_t = k \mid x_t, \{(x_i, y_i)\}_{i=1}^{t-1})$, is the *posterior*, which indicates the probability that k is the correct label after seeing x_t. The denominator is a normalization factor so that the posterior probability sums to 1. Once the posterior probability is computed, one can classify x_t by the most likely label:

$$\begin{aligned}\hat{y}_t &= \arg\max_{k \in K} P(y_t = k \mid x_t, \{(x_i, y_i)\}_{i=1}^{t-1}) \qquad (1.2)\\ &= \arg\max_{k \in K} P(x_t \mid y_t = k, \{(x_i, y_i)\}_{i=1}^{t-1}) P(y_t = k \mid \{(x_i, y_i)\}_{i=1}^{t-1}).\end{aligned}$$

The above classification rule minimizes expected error. Alternatively, one can *sample* the class label in a practice known as Gibbs classifier in ML:

$$\hat{y}_t \sim P(y_t = k \mid x_t, \{(x_i, y_i)\}_{i=1}^{t-1}) \qquad (1.3)$$

which corresponds to probability matching in psychology (Myers, 1976; Vulkan, 2000).

In ML there exist a variety of models for computing the posterior via Bayes rule. In all of these models, the prior is typically a multinomial distribution over the values y_t may take (i.e., the different category labels). Thus the primary difference between probabilistic ML models is in how the likelihood term is calculated. Interestingly, three common ML models of this computation bear a striking resemblance to the exemplar, prototype, and Rational models of human categorization. Indeed, certain parametrization of the CP models are formally identical to the ML models. This identity is, perhaps, surprising since the primary goal of the CP work has been to fit observed human behavior in artificial category learning experiments. Many early theorists, with the notable exceptions of Shepard (1991) and Anderson (1991), did not explicitly consider whether the probabilities computed by a given model were correct in any formal sense (see e.g. Medin and Schaffer, 1978; Hintzman, 1986; Rosch et al., 1976). The fact that the CP and ML probabilistic models are formally equivalent thus suggests that human categorization decisions are optimal in some respects – that is, the decisions people make are shaped by estimates of the true posterior probability distribution, and so represent the best decisions that can be made given prior beliefs and learning episodes (Anderson, 1991; Sanborn et al., 2006; Tenenbaum et al., 2006; Griffiths et al., 2011).

The equivalence of CP and probabilistic ML models is also useful for another reason: it allows us to leverage insights from machine learning to develop explicit hypotheses about human SSL. A considerable amount of work in machine learning has focused on how best to exploit both labeled data, consisting of (x, y) pairs, and *unlabeled* data, consisting of only the x observations without y labels. The modification of a supervised model to make use of unlabeled data is sometimes called *lifting* the model. In machine learning the primary motivation for lifting supervised models has been that labeled data are often expensive – that is, data labeling can be time-consuming and often requires an expert in the field. In contrast, unlabeled data are usually plentiful and inexpensive to acquire in large quantities. A key discovery has been that, under certain well-specified assumptions, semi-supervised models can use the potentially inexpensive unlabeled data to greatly improve classifier performance compared to supervised models alone (Balcan and Blum, 2010).

1.2 Semi-Supervised Learning Assumptions

By definition, unlabeled data do not come with labels and so cannot be used directly for supervised learning. Instead, these data provide information about the marginal $P(x)$, that is, the distribution of items in the feature space. To use this information for category learning, assumptions must be made about the nature of the unlabeled item distribution and the relationship between $P(x)$ and $P(y \mid x)$. These assumptions then "steer" how category learning proceeds. SSL is the learning paradigm that adopts such assumptions to make use of both labeled and unlabeled data when learning to categorize.

There are many types of SSL assumptions that can be used to support different kinds of learning models (Chapelle et al., 2006; Zhu and Goldberg, 2009). The assumption most germane to existing psychological models of categorization is the *mixture model assumption*, which states that all items are drawn independently from a probability distribution composed of a mixture of underlying components. The observed distribution of unlabeled examples can thus be used to infer the underlying mixture components, while the comparatively infrequent labeled examples can be used to label each component. We will use the mixture model assumption to create lifted variants of the prototype and Rational models of human semi-supervised learning. The exemplar model is a non-parametric model that requires a slightly different assumption.

Another SSL assumption we will encounter is the *manifold assumption*, also known as graph-based SSL. Here it is assumed that the labels vary slowly along an underlying manifold i.e. the discrete graph formed by connecting nearby items. In other words, if we form a graph among the data-points in dataset D, we can propagate labels along these edges.

Finally, when we investigate the use of the *Co-Training algorithm* to influence human behavior, we will make several assumptions very specific to the task setting, the most important being that learners using two separate views of the data can cooperate to learn a classification.

1.3 Translating Between ML and CP

While the tasks of classification and categorization are identical in ML and CP respectively, the literature describing the two perspectives have developed independently. This has led to a need to translate between the terms used in ML classification and CP categorization. The models I describe here will derive directly from this shared focus. In most cases, like in the case of classification and categorization, terms will be used interchangeably and can be considered equivalent unless specified otherwise.

Though there are many equivalences, it is important to note the differences between the two fields. One constraint of human category learning to consider that the average human can only attend to a single visual stimulus at a time. The implication here is that, in general, humans are performing *online* learning: the learner is constrained by the need to attend to and process one stimuli at a time, resulting in a temporal ordering of the data. In other words, the data points $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$ must be considered in sequence and not all at once, as they would in a *batch* setting. Experimentally, this ordering can either be explicitly enforced, where human learners are exposed to stimuli one at a time, or it may occur implicitly, such as when the learner is given a batch of data to process and can consider each item in the order that they choose. In order to provide an accurate analogue to human behavior, this restriction may need to be considered during model design.

Another constraint to consider is the fact that humans are not tireless. Unlike a computer, a human asked to do a repetitive task will over time grow tired. In order to maintain human attention, the learner must also be motivated. Very often this motivation is external to the task, such as class credit for participating. It is an important thing to consider when designing human experiments, especially when coming from the endlessly patient and cooperative world of ML.

In the next chapter, I will continue to translate between ML and CP, looking at the set of models with roots in both ML and CP. Following that I will show the combination of labeled and unlabeled data can affect the human learner, first motivated in an attempt to understand human categorization behavior and then in an attempt to manipulate this learning.

With regard to understanding human behavior, we will first review previous work proving that humans are sensitive to, and can learn from, the distribution of the unlabeled data. From there I will describe new work which deals with two motivations: understanding human behavior and influencing human behavior using SSL methods.

To better understand human behavior in a SSL setting, I discuss a study investigating how ordering effects can affect human learning. This is followed by an investigation how best to explain category shifts seen in human SSL under a particular family of models. A third experiment is described applying a fundamentally different SSL assumption: the network assumption where underlying manifolds may be perceived. In this experiment we will also see our first instance of attempting to influence human behavior, to attempt and drive human learning to a particular solution.

I then shift two experiments describing explicit attempts to influence human behavior. The first is an attempt to speed human learning using prior unlabeled experience. The second makes use of yet another set of SSL assumptions where Co-Training constraints are applied in an attempt to see two human learners can learn a classification boundary unlikely to be achieved by either learner alone.

This is followed by a final chapter providing a summary of the work, a discussion of future work, and a few of the lessons learned working with humans from the point of view of ML.

2 SEMI-SUPERVISED MODELS OF HUMAN CATEGORIZATION BEHAVIOR

In this chapter we will describe translations between a set of CP models and equivalent ML models, providing additional motivation for the application of ML techniques to the study of human learning.

2.1 Exemplar Model as Kernel Density Estimation

One common model of human categorization is the *exemplar* model, which stores all previously-viewed examples and uses these to estimate the most likely category label for novel query items. The Generalized Context Model (GCM) proposed by Nosofsky (1986, 2011) is probably the best known of this class in human category learning. To facilitate comparison with ML models we consider a specific parametrization of the full GCM model, in which two free parameters, memory strength and dimensional scaling weights (see Nosofsky, 2011), are fixed to one. With this simplification, the GCM model can be described as

$$P(y_t = k \mid \mathbf{x}_t, \{(\mathbf{x}_i, y_i)\}_{i=1}^{t-1}) = \frac{b^{(k)}\left(\sum_{j:y_j=k} s(\mathbf{x}_t, \mathbf{x}_j)\right)}{\sum_{k':k'\in K} b^{(k')}\left(\sum_{j':y_{j'}=k'} s(\mathbf{x}_t, \mathbf{x}_{j'})\right)} \tag{2.1}$$

where $b^{(k)}$ is the bias on category k and $s(\mathbf{x}_i, \mathbf{x}_j)$ is a scaled similarity measure between item $\mathbf{x}_i$ and $\mathbf{x}_j$. The bias term b serves the same role as the prior in the Bayes rule: it indicates the probability of encountering a label with value k prior to observing the query item. Intuitively it is easy to see that the probability of the query item $\mathbf{x}_i$ sharing the same label as a stored item $\mathbf{x}_j$ grows with the similarity s between the queried and stored items. Consequently the probability that the query item receives label k depends on its similarity to all items in category k and its similarity to all other items in the contrasting categories.

This formulation does not specify how the similarity between the queried and stored examples is to be computed. In ML, a common choice for s is the Gaussian kernel, which, in 1D is defined by

$$s(x_i, x_j) = \exp\left[-\frac{1}{2\sigma^2}(x_i - x_j)^2\right] \tag{2.2}$$

where σ^2 is the variance. In psychological models it is more common to employ an exponential similarity gradient, following Shepard (1986). Shepard's 1986 arguments, however, were premised on the assumption that the item distribution $P(\mathbf{x})$ was uniform over discrete dimensions (see Anderson, 1991); in the studies we consider below, the items are sampled from a mixture of Gaussian distributions in a fully continuous space. Empirically, at least one study has found that Gaussian similarity functions can provide a better fit to human behavior for such stimuli (Nosofsky, 1985). Moreover, an interesting property of this class of model is that, in the limit, the estimate of $P(y_t = k \mid \mathbf{x}_t, \{(\mathbf{x}_i, y_i)\}_{i=1}^{t-1})$ is not affected by the shape of the similarity gradient (or the *kernel* in ML). For these reasons, a Gaussian similarity function is used in what follows.

Kernel Density Estimation

A clear analog to exemplar models is Kernel Density Estimation (KDE). Like exemplar models, each labeled example $(\mathbf{x}_i, y_i)$ is retained in KDE and is used to compare against the current query item. One model that makes use of the likelihood estimate provided by KDE is the Nadaraya-Watson kernel estimator (Nadaraya, 1964; Wasserman, 2006; Shi et al., 2008), a regression function that returns a real value. When this estimator is adapted to categorization, the real value provides a direct estimate of the conditional probability $P(y_t = k \mid \mathbf{x}_t, \{(\mathbf{x}_i, y_i)\}_{i=1}^{t-1})$. Given training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{t-1}$, the categorization function is

$$P(y_t = k \mid \mathbf{x}_t, \{(x_i, y_i)\}_{i=1}^{t-1}) = \frac{\sum_{j=1}^{t-1} \mathcal{K}\left(\frac{x_t - x_j}{h}\right) \delta(y_j, k)}{\sum_{j'=1}^{t-1} \mathcal{K}\left(\frac{x_n - x_{j'}}{h}\right)} \tag{2.3}$$

where the kernel function $\mathcal{K}$ determines the weight between the query item x_t and each of the $1, \ldots, t-1$ exemplars x_j, and where $\delta(u, v) = 1$ when $u = v$ and 0 otherwise.

From this description, the equivalence between Equation (2.3) and Equation (2.1) may not be immediately obvious. Under certain parameter settings however, the equivalence becomes clear. The kernel function $\mathcal{K}$ acts like the similarity function $s(\mathbf{x}_i, \mathbf{x}_j)$, returning a value that gives a sense of the "similarity" between the query $\mathbf{x}_n$ and each exemplar $\mathbf{x}_j$. The hyperparameter h, the *bandwidth* parameter, controls how the effect of each exemplar diminishes with distance. Using a Gaussian function for s (in the exemplar model) and a 1-dimensional Gaussian kernel for $\mathcal{K}$ (in the ML model), and setting the bandwidth h to one standard deviation of this Gaussian, the functions become identical:

$$s(x_i, x_j) = \exp\left[\frac{1}{2h^2}(x_i - x_j)^2\right] = \exp\left[\frac{1}{2}\left(\frac{x_i - x_j}{h}\right)^2\right] = \mathcal{K}\left(\frac{x_i - x_j}{h}\right). \tag{2.4}$$

Setting $b^{(k)} = 1$ for all k completes the equivalence. This parametrization of the Nadaraya-Watson KDE is therefore formally identical to the parametrization of the GCM described in (2.1) with the additional constraint that all categories are assumed to be equally likely a-priori.

The Semi-Supervised Exemplar Model

To derive our semi-supervised exemplar model, we describe an SSL version of KDE and make use of the equivalence between the Nadaraya-Watson KDE and the GCM model. The standard model is lifted as follows: When an item $\mathbf{x}_i$ is queried for a label, the supervised model returns $P(y_i = k \mid \mathbf{x}_i)$ for all $k = 1, \ldots, K$ categories. Normally, in supervised learning, the true label y_i will then be received and the labeled $(\mathbf{x}_i, y_i)$ pair added to the training set in preparation for the next query item $\mathbf{x}_{i+1}$. In the semi-supervised setting, $\mathbf{x}_i$ may remain unlabeled, so that no ground truth y_i label is received. Instead of tossing out this unlabeled $\mathbf{x}_i$, as would happen in the supervised case, the real value $P(y_i = k \mid \mathbf{x}_i)$ is calculated for all $k = 1, \ldots, K$ and these values are considered *soft labels* on $\mathbf{x}_i$. The $\mathbf{x}_i$, together with the soft labels, is then added to the training set as a pseudo-labeled exemplar. Thus we now maintain $(\mathbf{x}_i, \mathbf{y}_i)$ pairs where $\mathbf{y}_i$ is a vector with $\mathbf{y}_{ik} = P(y_i = k \mid \mathbf{x}_i),\ k = 1, \ldots, K$. If $\mathbf{x}_i$ is labeled with $y_i = k^*$, the corresponding $\mathbf{y}_{ik^*} = 1$ while $\mathbf{y}_{ik} = 0$ for all other values of k. Algorithm 1 describes the model in detail.

2.2 Prototype Model as Mixture of Gaussians

Unlike the exemplar model, where learning is accomplished by storing all individual training items, learning in the *prototype* model consists of summarizing each category and discarding the training items themselves. The summary is achieved by assuming that each category can be represented with a parametric distribution $P(\mathbf{x} \mid y = k)$, so that only the distribution parameters for each category need be retained. The parameters associated with a given category constitute the category *prototype*. Prototypes do not necessarily correspond to any particular labeled item, but are abstract repre-

Algorithm 1: Semi-Supervised Exemplar Model

```
Given: kernel bandwidth $h$
for $n = 1, 2, \ldots$ do
    Receive $\mathbf{x}_t$ and predict its label using
        $$\arg\max_k P(y_t = k \mid \mathbf{x}_t, \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{t-1}) = \frac{\sum_{j=1}^{t-1} \mathcal{K}\left(\frac{\mathbf{x}_t - \mathbf{x}_j}{h}\right) \mathbf{y}_{jk}}{\sum_{j'=1}^{t-1} \mathcal{K}\left(\frac{\mathbf{x}_t - \mathbf{x}_j'}{h}\right)}. \quad (2.5)$$
    if $\mathbf{x}_t$ is labeled with $y_t = k^*$ then
        Set $\mathbf{y}_{tk} = \begin{cases} 1, & \text{if } k = k^* \\ 0, & \text{o.w.} \end{cases}$, for $k = 1, \ldots, K$.
    else
        Set $\mathbf{y}_{tk} = P(y_t = k \mid \mathbf{x}_t, \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{t-1})$ for $k = 1, \ldots, K$.
    end
    Add $(\mathbf{x}_t, \mathbf{y}_t)$ as an exemplar.
end
```

sentations of all labeled items in the category they represent. For example, if we assume that each category $P(\mathbf{x} \mid y = k)$ has a Gaussian distribution, then the corresponding prototype can be represented by the parameters $\mu^{(k)}$ (mean or "component center") and $\sigma^{2(k)}$ (variance or "spread"). Typically the number of categories K in the model is fixed in advance, before any labeled examples are seen, so that the number of stored prototypes does not grow with the number of examples. A new item is labeled by comparing it to each stored prototype.

A variety of different prototype models have been proposed in the psychological literature. To illustrate the link to ML, we consider the model proposed by Minda and Smith (2011), in which the prototype is simply the sample mean of labeled training examples in a given category. Query items are labeled using the same method as in the exemplar model, by comparison to a set of stored representations. The difference is that the stored representations are category prototypes, and not the labeled training items themselves. Thus it is not surprising that the formal description of the model is very similar:

$$P(y_t = k \mid \mathbf{x}_t, \{(\mathbf{x}_i, y_i)\}_{i=1}^{t-1}) = \frac{b^{(k)}\, s(\mathbf{x}_t, \bar{\mathbf{x}}^{(k)})}{\sum_{k':k' \in K} b^{(k')}\, s(\mathbf{x}_t, \bar{\mathbf{x}}^{(k')})} \tag{2.6}$$

where $\bar{\mathbf{x}}^{(k)}$ is the prototype for category k and $s(\mathbf{x}_i, \bar{\mathbf{x}}^{(k)})$ is a similarity function as in Equation (2.1), except that now $\mathbf{x}_i$ is compared to a single summary representation $\bar{\mathbf{x}}^{(k)}$ of each category k. Just as in the exemplar model, the bias term $b^{(k)}$ encodes the prior belief on label k.

Gaussian Mixture Models

An ML analog to prototype models is the mixture model, in which items are assumed to be generated from some mixture of underlying components. Each component is represented by a set of parameters that are learned from the data, with the number of components fixed before learning. We use the Gaussian Mixture Model (GMM), where each category is represented by a single component corresponding to a Gaussian distribution. The GMM is defined by parameters $\theta = \{\alpha, \mu, \Sigma\}$, where α is the set of non-negative mixing parameters $\{\alpha^{(1)}, \ldots, \alpha^{(K)}\}$, $\sum_{k=1:K} \alpha^{(k)} = 1$, μ a vector of the corresponding K means $(\mu^{(1)}, \ldots, \mu^{(K)})$, and Σ a set of covariance matrices $(\Sigma^{(1)}, \ldots, \Sigma^{(K)})$. When $\mathbf{x}$ is one-dimensional, the covariance matrices are replaced by variances $\sigma^{2(1)}, \ldots, \sigma^{2(K)}$. The model is defined by the joint probability $P(\mathbf{x}_i, y_i \mid \theta) = P(y_i \mid \theta)P(\mathbf{x}_i \mid y_i, \theta)$ where

$$P(y_i = k \mid \theta) = \alpha^{(k)}, \tag{2.7}$$

$$P(\mathbf{x}_i \mid y_i = k, \theta) = \mathcal{N}(\mathbf{x}_i;\ \mu^{(k)}, \Sigma^{(k)}). \tag{2.8}$$

Note that the n training examples seen prior to the query $\mathbf{x}_n$ are not used directly to label new items, but instead are used to estimate the parameters

θ, typically via the maximum likelihood estimate (MLE). The parameter estimates after seeing the $n-1$ examples are denoted as $\hat{\theta}^{(n-1)}$. The probability distribution over category labels for the query item $\mathbf{x}_n$ is then computed as the posterior

$$P(y_t = k \mid \mathbf{x}_t, \hat{\theta}^{(t-1)}) = \frac{P(\mathbf{x}_t \mid y_t = k, \hat{\theta}^{(t-1)})P(y_t = k \mid \hat{\theta}^{(t-1)})}{\sum_{k' \in K} P(\mathbf{x}_t \mid y_t = k', \hat{\theta}^{(t-1)})P(y_t = k' \mid \hat{\theta}^{(t-1)})} \tag{2.9}$$

with the most likely label found by taking $\arg\max_k P(y_t = k \mid \mathbf{x}_t, \hat{\theta}^{(t-1)})$.

As was the case when comparing KDE and exemplar models, GMMs are identical to prototype models under a certain parametrization. As in the exemplar model, we define the similarity function s to be a Gaussian (2.2). Unlike the exemplar model, where we compare the query $\mathbf{x}_t$ to each labeled example, here we only compare it to the set of K prototypes $\{\bar{\mathbf{x}}^{(k)} : k \in K\}$ corresponding to the K categories. For each category, the point $\bar{\mathbf{x}}^{(k)}$ is equal to the sample mean $\hat{\mu}^{(k)}$ for that category in the GMM formulation, while the covariance $\hat{\sigma}^{2(k)}$ enters s implicitly via the definition of multivariate Gaussian probability density function. The set of $\hat{\alpha}$ corresponds to the set of $b^{(k)}$. Thus under these settings the prototype model is equivalent to the GMM.

The Semi-Supervised Prototype Model

Recall that, in the prototype and GMM frameworks, the number of prototypes is fixed, usually equal to the number of categories, and each prototype is encoded by parameters learned from the training set. In the supervised setting these parameters can be computed in closed form by taking the MLE. In the semi-supervised setting, the closed-form computation is no longer possible because it is not clear to which category each unlabeled item belongs, and consequently it is not clear to which parameter estimates the item should contribute. To make use of unlabeled data, the MLE is instead computed using an approximation method, typically the *expectation maximization* (EM) algorithm (Dempster et al., 1977).

In the case of a 2-category Gaussian mixture model with $x \in \mathcal{R}$ and labels $y \in \{0, 1\}$ the sufficient statistics vector is

$$\hat{\phi}(x, y) = (1 - y, (1 - y)x, (1 - y)x^2, y, yx, yx^2). \tag{2.10}$$

Algorithm 2 formulates our procedure for using both labeled and unlabeled data to find a prototype model solution.

2.3 Rational Model as Dirichlet Process Mixture Model

The exemplar model is nonparametric in that the number of representational elements grows directly with the number of training examples and no assumptions are made about the number or distribution of categories. The prototype model is parametric in that there are a fixed number of components (category prototypes) which are defined by a fixed number of parameters. While several psychological models have been proposed that exist between these extremes (e.g., the Varying Abstraction model (Vanpaemel et al., 2005)), perhaps the most influential is Anderson's Rational model (Anderson, 1990, 1991). A version of the Rational algorithm, slightly modified from the presentation in Anderson (1991), is presented in Algorithm 3.

The term $\mathsf{P}(z_i = \ell' \mid x_i)$ controls the probability that a given item will be assigned to a new cluster, with the effect that the number of representational elements in a trained model will vary with this term. This probability in turn depends on a "coupling parameter" that specifies the prior probability of any two items being drawn from the same cluster. When the coupling parameter is low, $\mathsf{P}(z_i = \ell' \mid x_i)$ is high, so each labeled item will likely be

Algorithm 2: Semi-Supervised Prototype Model

```
Given: Prior encoded in $\phi$
Initialize $\theta^{(0)}$ from $\phi$ (see M-step below)
for $t = 1, 2, \ldots$ do
    Receive $x_t$ and calculate $q(y_t) = P(y_t \mid x_t, \theta^{(t-1)})$
    Receive $y_t$ (may be unlabeled), update model

    E-step:
    if $x_t$ is unlabeled then
        $\phi = \phi + \mathbb{E}_q[\tilde{\phi}(x_t, y_t)]$
    else
        $\phi = \phi + \tilde{\phi}(x_t, y_t)$
    end

    M-step: Let $\phi = (n_0, s_0, ss_0, n_1, s_1, ss_1)$.
    Compute $\theta^{(t)}$ as follows:
        $\alpha = \frac{n_1}{n_0 + n_1}$    (2.11)
        $\mu_0 = \frac{s_0}{n_0}$    (2.12)
        $\sigma_0^2 = \frac{ss_0}{n_0} - \left(\frac{s_0}{n_0}\right)^2$    (2.13)
        $\mu_1 = \frac{s_1}{n_1}$    (2.14)
        $\sigma_1^2 = \frac{ss_1}{n_1} - \left(\frac{s_1}{n_1}\right)^2$    (2.15)
    with $n_0$, $n_1$ the weighted sum of items assigned to category 0, 1.
end
```

placed in its own cluster, similar to the exemplar model. When the coupling parameter is high, $P(z_i = \ell' \mid x_i)$ is low and relatively few clusters will be learned, similar to the prototype model. In Anderson (1991), the coupling

Algorithm 3: Rational Model of Categorization

```
Given: cluster assignments $\{z_i\}_{i=1}^{t-1}$ assigning $\{\mathbf{x}_i\}_{i=1}^{t-1}$ to clusters in $L$
for each cluster $l \in L$ do
    calculate $P(z_t = l \mid \mathbf{x}_t, \{\mathbf{x}_i, z_i\}_{i=1}^{t-1})$, the probability that $\mathbf{x}_t$ comes
    from cluster $l$.
end
Also, let $P(z_n = l' \mid \mathbf{x}_t)$ be the probability that $\mathbf{x}_t$ comes from a new
cluster $l'$.

Assign $\mathbf{x}_t$ to the cluster with maximum probability:
    $z_t = \arg\max_{l \in \{L, l'\}} \begin{cases} P(z_t = l \mid \mathbf{x}_t, \{\mathbf{x}_i, z_i\}_{i=1}^{t-1}) \\ P(z_t = l' \mid \mathbf{x}_t) \end{cases}$    (2.16)
If the assigned cluster is the new $l'$, add $l'$ to $L$.
```

parameter is assumed to be fixed in advance of training.

Dirichlet Process Mixture Models

Dirichlet Process Mixture Models (DPMMs) are to KDEs and GMMs as the Rational model is to exemplar and prototype models: DPMMs allow the number of components of the mixture model to grow dynamically with the number of data points observed. Anderson's Rational model was in fact shown to be equivalent to the DPMM (Neal, 1998; Sanborn et al., 2006; Griffiths et al., 2011).

The model presented here is similar to the AClass model of (Mansinghka et al., 2007), which was used for supervised learning. But unlike AClass where each category has its own private DPMM, we stack $(x, y) : x \in \mathcal{R},\ y \in \{0, 1\}$, into an extended feature vector and use one global DPMM: $G \sim DP(G_0, \alpha_2)$, $\theta_1 \ldots \theta_t \sim G$, $(x_i, y_i) \sim F(x, y|\theta_i)$, where G_0 is a base distribution which we take to be the product of Normal-Gamma and Beta, conjugate priors for Normal and binomial: $G_0 = \text{NG}(\mu_0, \kappa_0, \alpha_0, \beta_0)\text{Beta}(\alpha_1, \beta_1)$. $\theta = (\mu, \lambda, p)$ is a parameter vector with the mean and precision of a Gaussian for the x

component, and the "head" probability for the y component. Due to the property of the Dirichlet process (Teh, 2010), many θ's will be identical, creating an implicit clustering of items. F is a product of Gaussian and Bernoulli: $F = \mathrm{Norm}(x; \mu, \lambda)p^y(1-p)^{1-y}$. As is common with DPMM, we introduce cluster membership indices $z_1 \dots z_t$, and integrate out θ and G via particle filtering (Fearnhead, 2004). That is, at iteration $t-1$ we assume the distribution $P(\{z_i\}_{i=1}^{t-1} \mid \{x_i\}_{i=1}^{t-1}, \{y_i\}_{i=1}^{t-2})$ is well-approximated by the empirical distribution on m particles $\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}$, each particle $\mathbf{z}$ a vector of indices $z_1, \dots, z_{t-1}$:

$$P(\{z_i\}_{i=1}^{t-1} \mid \{x_i\}_{i=1}^{t-1}, \{y_i\}_{i=1}^{t-2}) \approx \frac{1}{m} \sum_{l=1}^{m} \delta(\{z_i\}_{i=1}^{t-1}, \mathbf{z}^{(l)}),$$

where $\delta(u, v) = 1$ if $u = v$, and 0 otherwise. Then, at iteration t, after we observe the input item x_t but before seeing its label y_t, the distribution $P(\{z_i\}_{i=1}^{t} \mid \{x_i\}_{i=1}^{t}, \{y_i\}_{i=1}^{t-1})$ can be shown to be proportional to

$$\sum_{l=1}^{m} \delta\left(\{z_i\}_{i=1}^{t-1}, \mathbf{z}^{(l)}\right) P\left(y_{t-1} \mid \mathbf{z}^{(l)}, \{y_i\}_{i=1}^{t-2}\right) P\left(z_t \mid \mathbf{z}^{(l)}\right) P\left(x_t \mid z_t, \mathbf{z}^{(l)}, \{x_i\}_{i=1}^{t-1}\right). \tag{2.17}$$

One would further sample from (2.17) m new particles $\mathbf{z}^{(1)}, \dots \mathbf{z}^{(m)}$. The empirical distribution on these new particles will approximate $P(\{z_i\}_{i=1}^{t} \mid \{x_i\}_{i=1}^{t}, \{y_i\}_{i=1}^{t-1})$. This update is the key to particle filtering, which uses a fixed number of particles to approximate an increasingly complex distribution.

In (2.17), one needs to compute three conditional probabilities. The conditional probability of z_t is computed from the Chinese Restaurant Process prior. Let there be K unique index values $1 \dots K$ in $\{z_i\}_{i=1}^{t-1}$, then

$$P(z_t = k \mid \{z_i\}_{i=1}^{t-1}) = \begin{cases} t_k/(\alpha_2 + t - 1), & k \leqslant K \\ \alpha_2/(\alpha_2 + t - 1), & k = K + 1 \end{cases}$$

where t_k is the number of indices with value k in $\{z_i\}_{i=1}^{t-1}$. The conditional probability of x_t is computed from a student-t distribution,

$$P(x_t \mid \{z_i\}_{i=1}^{t}, \{x_i\}_{i=1}^{t-1}) = t_{2\alpha}\left(x_t \mid \mu, \beta(\kappa+1)/(\alpha\kappa)\right),$$

with

$$\mu = \frac{\kappa_0\mu_0 + N\bar{x}}{\kappa_0 + N} \tag{2.18}$$

$$\kappa = \kappa_0 + N \tag{2.19}$$

$$\alpha = \alpha_0 + \frac{N}{2} \tag{2.20}$$

$$\beta = \beta_0 + \frac{1}{2}\sum_{i=1}^{N} \delta(z_i, z_t)(x_i - \bar{x})^2 + \frac{\kappa_0 N(\bar{x} - \mu_0)^2}{2(\kappa_0 + N)} \tag{2.21}$$

$$N = \sum_{i=1}^{t} \delta(z_i, z_t) \tag{2.22}$$

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{t-1} \delta(z_i, z_t)x_i \tag{2.23}$$

The Semi-Supervised Rational Model

Just as in the semi-supervised exemplar and prototype models, lifting the DPMM requires modifications to accommodate both labeled (x, y) pairs and unlabeled x items with no corresponding ground truth labels y. The key point is that the probability distribution over partition assignments, which is central to the Rational/DPMM approach, is influenced here by the distribution of both labeled and unlabeled examples in the feature space, as well as by the labels given to the labeled items. Unlabeled data thus influence category learning by influencing which partitions of the feature space are most probable.

Importantly, for our semi-supervised variant of DPMM, the conditional

Algorithm 4: Semi-Supervised Rational Model of Categorization

```
Parameters: $\alpha_2, \mu_0, \kappa_0, \alpha_0, \beta_0, \alpha_1, \beta_1$
Initialize $m$ empty particles; $y_0$ =unlabeled
for $t = 1, 2, \ldots$ do
    Receive $y_{t-1}$ (may be unlabeled) and $x_t$
    Re-sample $m$ particles from (2.17)
    Predict $y_t$ with new particles from (2.27)
end
```

probability of y_{t-1} is computed from a beta-binomial distribution

$$P(y_{t-1} \mid \{z_i\}_{i=1}^{t-1}, \{y_i\}_{i=1}^{t-2}) = \frac{c_1 + \alpha_1}{c_0 + c_1 + \alpha_1 + \beta_1}. \tag{2.24}$$

Note some of the y's might be unlabeled. If y_{t-1} is unlabeled, the probability is simply 1 since it must take either one of the labels. If some y's in $\{y_i\}_{i=1}^{t-2}$ are unlabeled, one can show that those are marginalized over, resulting in the following counts:

$$c_1 = \sum_{i=1}^{t-2} \delta(z_i, z_{t-1})\delta(y_i, 1) \tag{2.25}$$

$$c_0 = \sum_{i=1}^{t-2} \delta(z_i, z_{t-1})\delta(y_i, 0) \tag{2.26}$$

Here, we define $\delta(y_i, 1) = \delta(y_i, 0) = 0$ if y_i is unlabeled. Once the particles are updated with (2.17), predicting y_t is straightforward:

$$p(y_t \mid \{x_i\}_{i=1}^{t}, \{y_i\}_{i=1}^{t-1}) \approx \frac{1}{m} \sum_{l=1}^{m} p(y_t \mid \mathbf{z}^{(l)}, \{y_i\}_{i=1}^{t-1}) \tag{2.27}$$

where $p(y_t \mid \mathbf{z}^{(l)}, \{y_i\}_{i=1}^{t-1})$ is computed with (2.24). The complete algorithm is given in Algorithm 4.

3 SEMI-SUPERVISED EFFECTS DUE TO DISTRIBUTION OF UNLABELED DATA: PREVIOUS EVIDENCE

While the models discussed in the previous chapter provide a theoretical basis for considering models of human semi-supervised learning, this does not show that humans actually *are* affected by unlabeled data in a categorization task. The following experiment, conducted by members of my research group prior to my joining, was among the first demonstration of the sensitivity of human learners to unlabeled data in a categorization task.

3.1 Experiment 1: SSL Distribution Effects

The experiment was designed to assess whether human categorization decisions are influenced by the distribution of unlabeled examples (Zhu et al., 2007). 22 students at the University of Wisconsin completed a binary categorization task with complex novel shapes varying in a single continuous parameter $x \in [-2, 2]$ as seen in the examples in Figure 3.1. The two categories were denoted by $y = 0$ or $y = 1$. Participants first received 2 labeled items: $(x, y) = (-1, 0)$ and $(1, 1)$, repeated 10 times each in random order. These items were "labeled" in that feedback indicating the correct response was provided after each trial. Participants next classified 795 unlabeled test examples in one of two experimental conditions, differing only in how the majority of the unlabeled items were generated. In the L-shift condition, 690 of the unlabeled test items were drawn from a mixture of two Gaussians with a trough shifted to the left of the boundary implied by the labeled training items (see Figure 3.2). The other condition, R-shift, varied only in that the trough between the Gaussians was now shifted to the right of the implied labeled boundary. In both conditions, the remaining

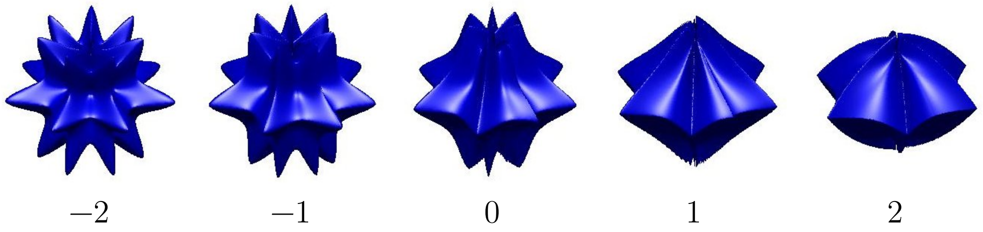


Figure 3.1: Example stimuli used in Zhu et al. (2007), with corresponding x values.

unlabeled test items were items drawn from a grid across the entire range of x, ensuring that both unlabeled distributions spanned the same range. The grid items appeared in random order at the beginning and end of the unsupervised phase, allowing for the measurement of the category boundary participants learned immediately following the supervised experience and following exposure to the unlabeled bimodal distribution.

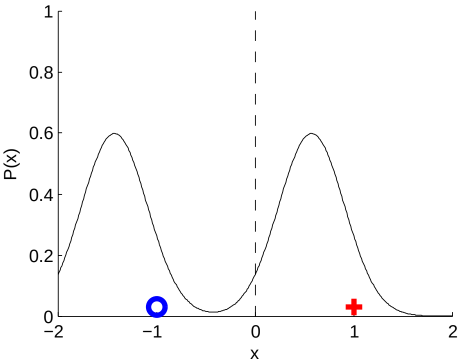


Figure 3.2: Example of the dataset used in the L-shift condition of Zhu et al. (2007). Labeled points are represented as negative (○) and positive (+). The black curve is the bimodal distribution $P(x)$ from which unlabeled items were drawn. The dashed vertical line represents the boundary implied by the labeled points alone. Note that the trough in the unlabeled distribution is shifted to the left with respect to the supervised learning boundary.

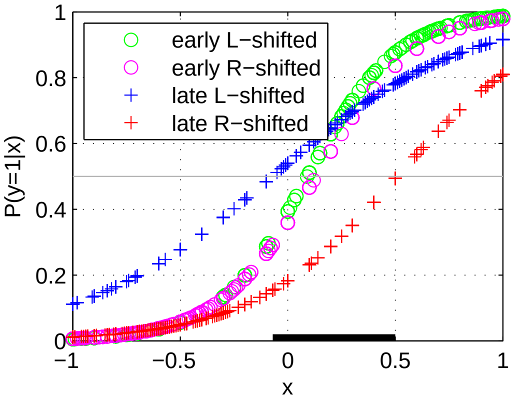


Figure 3.3: Results from shift in unlabeled distribution in Zhu et al. (2007). The thick black line marks items on which the majority human categorization differs in the two conditions.

Figure 3.3 shows a summary of the results by pooling human behavior by condition and fitting logistic regression curves to show the conditional probability $\mathsf{P}(y = 1 \mid x)$. Two subsets of the data are examined. The early subset shows behavior on the first 50 unlabeled test items (drawn right after the labeled training phase), while the late subset shows behavior on the final 50 unlabeled test items (drawn at the end of exposure to unlabeled data).

Comparing the early items, the two groups look essentially the same and the curves overlap. On the late items the curves are substantially different. The decision threshold, i.e., x producing the value $\mathsf{P}(y = 1 \mid x) = 0.5$, shifted in opposite directions in the two conditions, moving to the left in the L-shift condition and to the right in the R-shift condition. In the late subset, the majority of participants classified the items $x \in [-0.07, 0.50]$ differently in the two conditions. If participants were unaffected by unlabeled data, the late test curves should be identical to the early curves and overlap. The fact that they do not indicates that participants *are* affected by the unlabeled data for this categorization task. To statistically test these observations, decision boundaries for the early and late grid-test items were computed separately for each participant using logistic regression on the participant's categorization decisions. A repeated measures analysis of variance assessing the influence of early vs. late and L-shift vs. R-shift on the location of the decision showed a significant interaction between the two factors ($\mathsf{F}(1, 18) = 7.82, \mathsf{p} < 0.02$), indicating that after exposure to the unlabeled data, the decision boundary shifted in significantly different directions for the two groups. Thus exposure to the unlabeled bimodal distribution appears to alter participant's beliefs about the location of the category boundary.

3.2 Experiment 2: Social Categories

The second study had a somewhat different goal – namely to investigate whether semi-supervised learning might provide part of an explanation as to why people are often prone to form incorrect beliefs about social categories (Kalish et al., 2011). The experiment is useful for current purposes, however, because it revealed similar effects to those reported by Zhu et al. (2007) even though it used quite different stimuli and a different method for measuring the effect of unlabeled items. In this experiment the unlabeled distribution was held constant while the location of the original labeled examples varied across experimental groups.

Forty-three undergraduates viewed schematic images of women varying along the single dimension of width. The women were described as coming from one of two islands. As in Experiment 1, each participant first completed a supervised phase where a labeled example from each category (i.e. "Island") was presented five times in random order for a total of 10 labeled examples. In the L-labeled condition participants viewed two relatively thin stimuli (pixel-widths of 80 and 115) while those in the R-labeled condition viewed two somewhat wider stimuli (pixel-widths of 135 and 165). All participants then classified a set of unlabeled items without feedback. In the experimental conditions, both L-labeled and R-labeled groups viewed the same set of unlabeled items, including 37 initial test items sampled from a uniform grid along the full stimulus range, 300 items sampled from a mixture of two Gaussian distributions, and a final set of 37 test items sampled from the grid. The mixture of Gaussians was constructed so that the modes of the distribution lay midway between the labeled points in the L-Labeled and R-labeled conditions (see Figure 3.4). In a control condition, participants received the same L-labeled or R-labeled experience, but only viewed items lying on a grid between the two labeled items in the unsupervised phase.

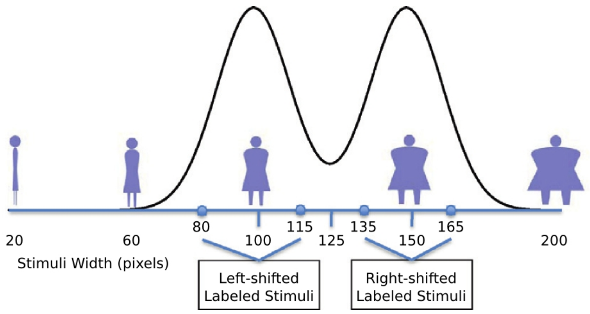


Figure 3.4: Examples of the Island Women stimuli, the labeled points, and the bimodal distribution from which unlabeled items are sampled.

In Zhu et al. (2007) the trough of the unlabeled distribution fell between the labeled points. In contrast, in this study the two labeled points both fell to one side of the trough in the unlabeled distribution, resulting in an even stronger conflict between the boundaries suggested by supervised and unsupervised experience. Given this mismatch, would learners still be affected by the unlabeled distributions? To answer this question, the authors considered three different measures. First, like Zhu et al. (2007), they considered how participants categorized unlabeled items along the grid prior to and following exposure to the bimodal unlabeled distribution. Second, following the unsupervised phase of the experiment, they asked participants to explicitly indicate where the boundary was located by adjusting a slider that controlled the width of a pictured stimulus. Finally, using the same slider, they asked participants to indicate the "most typical" example of each category.

All three measures showed beliefs about category structure to be strongly shaped by the distribution of the unlabeled examples. In the control condition, participant behavior strongly reflected their supervised learning experience: the estimate of the implicit category boundary and the participants' explicit reports of the boundary were closely aligned with and not significantly different from the midpoint between the labeled examples, while their judgments of the most typical example of each class aligned closely with and did not differ significantly from the labeled examples they had received. In comparison, implicit boundary estimates in the experimental groups were significantly shifted toward the trough in the unlabeled distributions – that is, toward the right in the L-labeled condition, and toward the left in the R-labeled condition. This shift was reflected even more strongly in the explicit boundary judgments. Moreover, choices about the most typical examples of each category aligned closely with the modes of the unlabeled distribution, shifting very dramatically away from the labeled items observed in the beginning of the experiment. Perhaps most interestingly, the majority of participants in each condition actually altered their beliefs about one of the two labeled examples, coming to classify it with the opposite label than that viewed during the supervised phase.

Given these substantial effects of unlabeled data, one might inquire whether participants accurately remember the labeled examples and simply change their beliefs about the accuracy of the earlier supervisory feedback, or whether their memory for the labeled items itself changes. Kalish et al. (2011) addressed this question in a follow up experiment where, following exposure to the unlabeled items, participants used the slider in an attempt to reproduce the two labeled items that had appeared at the beginning of the study. Strikingly, their reproduction were also strongly influenced by the unlabeled data, lining up closely with the two modes of the unlabeled distribution even though, in actuality, the two labeled points lay on either side of one of the modes. Thus memory for the original labeled examples appeared to be distorted by exposure to the unlabeled items.

One might further wonder whether the labeled experience has any impact at all in these studies beyond providing basic information about which "cluster" in the unlabeled distribution should get which label. Kalish et al. (2011) were able to show that the labeled information does, in fact, have a persisting influence even after extensive unlabeled experience: despite being exposed to exactly the same unlabeled items, participants in the L-labeled and R-labeled conditions of these studies did not end up with exactly the same beliefs about the location of the boundary. Instead, the L-labeled group's final boundary was displaced significantly to the left of the R-labeled group's final boundary, indicating some lasting effect of the original supervisory experience.

Finally, this study rules out an alternative explanation of the effects of unlabeled data in these experiments. In the Zhu et al. (2007) study, because participants in the different experimental groups viewed different sets of unlabeled items, it was possible that the observed differences in categorization boundaries might arise from perceptual contrast effects. For instance, a given stimulus in that study might look "more pointy" or "less pointy" depending upon how pointy the preceding stimulus was. It is conceivable that these local perceptual contrast effects might lead to consistent differences in the estimated category boundary depending upon the location of the trough in the unlabeled distribution. In the study of Kalish et al. (2011), both experimental groups viewed exactly the same set of unlabeled items, in the same fixed order, but nevertheless showed systematic differences in their estimate of the category boundary depending upon their supervised experience. Thus the learning in this study appears to be truly semi-supervised, reflecting contributions from both labeled and unlabeled experience.

4 SEMI-SUPERVISED EFFECTS DUE TO ORDER OF UNLABELED DATA (ZHU, GIBSON ET AL., 2010)

In this study we used ML models and techniques to investigate the behaviors exhibited by humans in categorization tasks and how that behavior is affected by a mixture of labeled and unlabeled data. We introduced the term *Test-Item Effect* to denote the possibility that unlabeled test items can induce changes to the classifier f in human category learning. Specifically, the Test-Item Effect predicts that two otherwise identical people A, B receiving exactly the same training data can be made to disagree on certain test items x^*, i.e., $f_A(x^*) \neq f_B(x^*)$, simply by manipulating what other test data $x^A_{n+1} \ldots$ and $x^B_{n+1} \ldots$ they are asked to classify, respectively.

My contribution *to this work was in performing the modeling analysis showing that existing SSL models can be modified to reproduce the Test-Item Effect observed in humans.*

While some past research did show that classification behavior can be influenced by the construction of the test set (Zaki and Nosofsky, 2007; Palmeri and Flanery, 1999; Fried and Holyoak, 1984), the Test-Item Effect was not well-understood. The goal of this project was (i) to report Test-Item Effects observed in a human category-learning task, and (ii) to assess whether the different SSL models described in Section 4.2 vary in how well they match the reported human behavior. If some models fit the human data better than others, this suggests that the Test-Item Effects might importantly constrain computational accounts of human category learning. The main contribution of this work was thus to CP and the understanding of human category learning.

4.1 Human Experiment

This experiment consisted of two identical conditions except for one aspect: They shared the same set of test items, but differed in the order the test items were presented to the subjects. As shown below, subjects in these two conditions consistently disagreed on the label of certain test items.

Participants and Materials

40 undergraduate students participated for partial course credit. The stimuli were the novel shapes seen before (Figure 3.1), varying according to a single continuous parameter $x \in [-2, 2]$. There were two classes, denoted as $y = 0$ or $y = 1$.

Procedure

In trial n, a stimulus x_n appeared on a computer screen, and stayed on until the subject pressed one of two keys to label it. All subjects initially had the same 10 labeled trials, where two items occurred alternatively: $(x_n, y_n) = (-2, 0), (2, 1), (-2, 0), (2, 1) \ldots$ For these 10 trials, after the subject pressed her key, a label feedback appeared on screen indicating whether her classification was correct (same as y_n). The computer screen was then cleared, and the stimulus for the next trial appeared. After these labeled trials, subjects were presented with a series of 81 unlabeled test items evenly spaced in feature space: $x_n = -2, -1.95, -1.9, \ldots, 2$. The test items appeared one at a time on the screen, and the subjects had to classify them using the same procedure. However, there was no longer labeled feedback after each classification. Importantly, the subjects were randomly divided into two conditions of equal size. In the "L to R" condition, the order of the test items was as above. In the "R to L" condition, the order was the opposite (i.e., $2, 1.95, 1.9, \ldots, -2$).

Result

Figure 4.1(Left) shows a plot of $P(y = 1|x)$, estimated by the fraction of subjects in each condition who classified x with label $y = 1$. The difference is striking.[1] Subjects in the "L to R" condition tended to classify more test items as $y = 0$, while those in the "R to L" condition tended to classify more as $y = 1$. For instance, for the same test item $x = -0.5$, only 4 out of 20 subjects in the "L to R" condition classified it as $y = 1$, while 15 out of 20 subjects in the "R to L" condition did so. This is significantly different using log odds ratio at $p < 0.0004$. It is clear evidence of the Test-Item Effect, where the effect is produced by the order of test items. In fact, for test items $x \in [-1.2, 0.1]$ a majority-vote among subjects would classify them in opposite ways in those two conditions.

We postulated that the subjects might perform self-reinforcement: that once a person classifies a test item x as in class y, the predicted label y (perhaps weighted by its uncertainty) becomes a training label for the person. For example, for a subject in the "L to R" condition, the first few test items are all near $x = -2$. The subject can easily classify them as $y = 0$ from the training she just received. If self-reinforcement is in effect, these test items will act as additional training data for the $y = 0$ class. This would tend to favor classifying more test items as $y = 0$. The opposite can be said for the "R to L" condition. Such self-reinforcement corresponds to the self-training algorithm in semi-supervised learning (Zhu and Goldberg, 2009, §2.5). Under certain probabilistic models, it can also be interpreted as an Expectation-Maximization (EM) procedure.

[1] We point out that the two curves in Figure 4.1 are not symmetric about $x = 0$, as one would expect. We speculate that this is due to the stimulus space in Figure 3.1 not being perceptually uniform. Our feature x is a parameter used to generate the geometry of the shapes, and does not necessarily match the human perceived similarity between stimuli. Nonetheless, this does not affect the validity of the observed Test-Item Effect, which only depends on the two curves separating from each other.

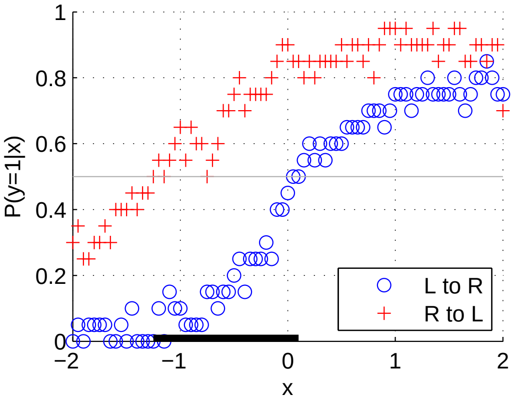


Figure 4.1: The Test-Item Effect due to order. The thick black lines mark items on which the majority human classification differs in the two conditions.

4.2 Model Comparison

In addition to the Text-Item Effect due to order, we can consider the behavior seen in Zhu et al. (2007) as evidence of a different kind of Test-Item Effect, in that case due to distribution (see Section 3.1). We used both experimental datasets to examine how well our SSL models, described in Section 4.2, fit human data.

Parameter tuning

Let $(x_n^{[s]}, y_n^{[s]})$, $n = 1, 2, \ldots$ be the sequence of training and test data that the s-th subject saw during human experiments, where some y's may be unlabeled. Furthermore, let $h_n^{[s]} \in \{0, 1\}$ be the binary classification response the s-th subject made at trial n. Each of our models predicts the label probability $P(y_n|x_{1:n}, y_{1:n-1}, \theta)$ at trial n, given parameter $\theta = h, n_0$, or α_2. We define training set log likelihood as

$$\ell_{tr}(\theta) \equiv \sum_{s \in tr} \sum_{n} \log P(h_n^{[s]} \mid x_{1:n}^{[s]}, y_{1:n-1}^{[s]}, \theta).$$

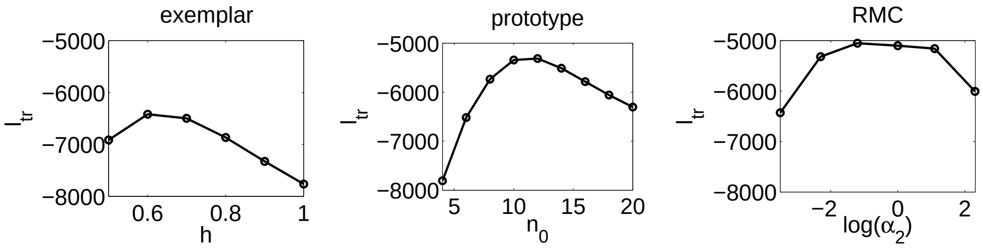


Figure 4.2: Order training set log likelihood $\ell_{\mathtt{tr}}(\theta)$ for a set of θ

	exemplar	prototype	RMC
$\hat{\theta}$	$h = 0.6$	$n_0 = 12$	$\alpha_2 = 0.3$
$\ell_{\mathtt{te}}(\hat{\theta})$	−3727	−2460	−**2169**

Table 4.1: Order test set log likelihood $\ell_{\mathtt{te}}(\hat{\theta})$

Because the order and distribution tasks used the same stimuli, we merge their subjects and fit a single parameter for both tasks.[2] Specifically, we take 32 subjects, eight each from the "order task L to R", "order task R to L", "distribution task L shifted", and "distribution task R shifted" conditions to form the training set $\mathtt{tr}$. The remaining 4, 2, 12, 12 subjects in those conditions form the test set $\mathtt{te}$, and define test set log likelihood $\ell_{\mathtt{te}}(\theta)$ accordingly. These sets are shared by the three models. For each model, we find the maximum likelihood estimate parameter $\hat{\theta} = \arg\max_{\theta} \ell_{\mathtt{tr}}(\theta)$ on the training set using a coarse parameter grid as shown in Figure 4.2.

Observations

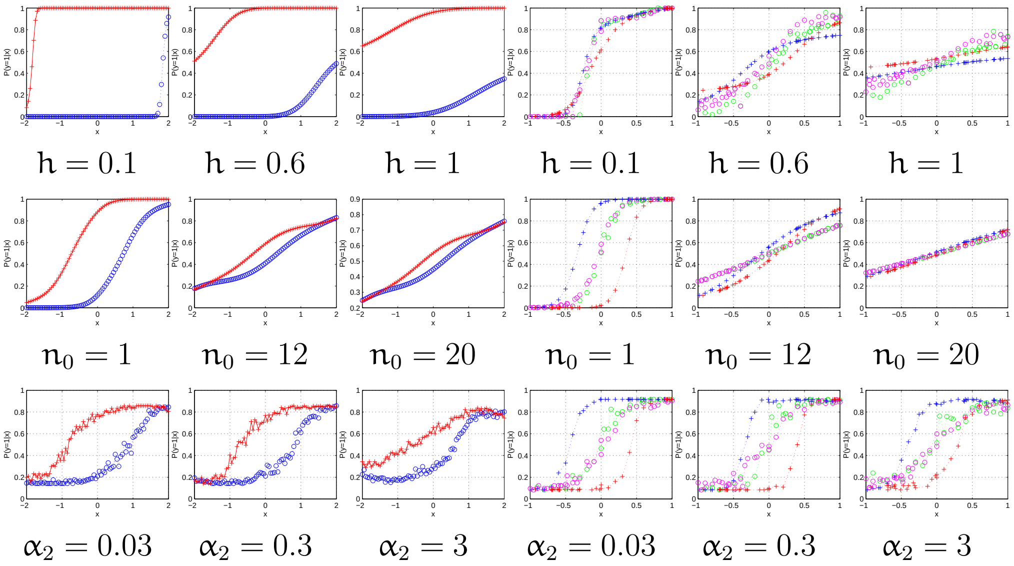
Table 4.1 shows the log likelihood $\ell_{\mathtt{te}}(\hat{\theta})$ on the *test set*, which was not involved in parameter tuning. In addition, Figure 4.3 shows the behavior of

[2] This reduces data sparsity. We assume that because the stimulus space is the same, and the learners have no prior knowledge that the tasks are different, they will use the same parameter setting in both tasks.

the three models over a wide range of parameters (including $\hat{\theta}$). We make a few observations: (i) All three models predict test-items effects. All models show different classification behavior following the same supervised training depending upon the order and distribution of the test items. (ii) Some models are more consistent with the empirical data than others. Specifically, the semi-supervised RMC model showed a qualitatively similar pattern (and the best log-likelihoods) to both datasets under a range of parameter values. The prototype model fared well under some parameter choices but not others; and the exemplar model failed to qualitatively match the empirical data under any of the studied parametrization. The test-item effect thus provides evidence useful for constraining theories of human categorization. In this case, it suggests that the RMC provides a better approximation of human category learning than either prototype or exemplar theories, though to more firmly assess this hypothesis it will be necessary to consider other parametrization of the later kinds of models.

Down-weight unlabeled exemplars

Our semi-supervised exemplar model has the lowest likelihood. On the "order" task, the two curves are too wide apart; on the "distribution" task, they overlap, cross, or even flip. A natural idea for improvement is to afford a weight parameter $w < 1$ to unlabeled exemplars: perhaps a self-assigned label is worth less than a true label. Specifically, one can adapt the Nadaraya-Watson kernel estimator into $r(x) = \sum_{i=1}^{n} \frac{w_i K(\frac{x-x_i}{h})}{\sum_{j=1}^{n} w_i K(\frac{x-x_j}{h})} y_i$, with $w_i = w$ if x_i is unlabeled, and $w_i = 1$ otherwise. Figure 4.4(left) shows $\ell_{tr}(w, h = 0.6)$ for the exemplar model with w ranging from 0 (supervised learning) to 2 (overweight). Clearly, semi-supervised learning ($w > 0$) is much better than supervised learning at explaining the human data. Training likelihood peaks at $w = 0.2$ and decreases thereafter. The *test set* log likelihood with $w = 0.2, h = 0.6$ is -2934, still worse than the other two models (which have only one parameter). The other two panels in Figure 4.4 show



$h = 0.1$ $h = 0.6$ $h = 1$ $h = 0.1$ $h = 0.6$ $h = 1$

$n_0 = 1$ $n_0 = 12$ $n_0 = 20$ $n_0 = 1$ $n_0 = 12$ $n_0 = 20$

$\alpha_2 = 0.03$ $\alpha_2 = 0.3$ $\alpha_2 = 3$ $\alpha_2 = 0.03$ $\alpha_2 = 0.3$ $\alpha_2 = 3$

Figure 4.3: (Top) Semi-supervised exemplar model, (middle) Semi-supervised prototype model, (bottom) Semi-supervised rational model of categorization. Columns 1–3 show model predictions $P(y_n = 1|x_{1:n}, y_{1:n-1})$ on the "order" task (Section 4.1), and columns 4–6 the "distribution" task (Section 3.1). The legend is the same as in Figure 4.1.

exemplar model predictions similar to the top row of Figure 4.3, but with $w = 0.2, h = 0.6$. Overall, down-weight unlabeled exemplar helps, but not overwhelmingly.

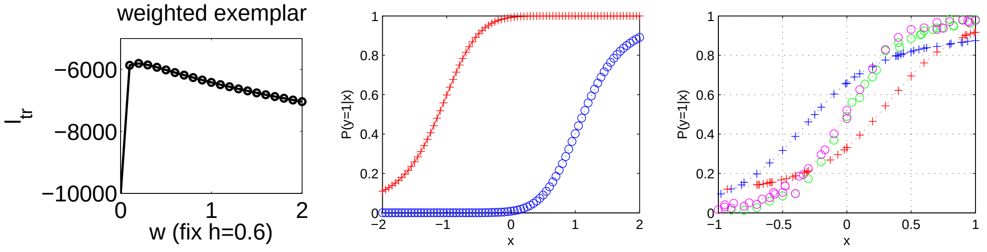


Figure 4.4: Down-weight unlabeled exemplars

While this an interesting adaptation to the model, the primary result of the study not change, the presentation of a novel Test-Item Effect in human categorization, induced by test item order. Together with the previously known distribution-induced effect, to describe this effect called for new online semi-supervised learning models, for which the models described in Section 4.2 were developed. The simulations discussed here show that all of our models exhibited the Test-Item Effect, with semi-supervised RMC giving the best fit.

5 WHAT PARAMETERS ARE AFFECTED IN SEMI-SUPERVISED EFFECTS?(GIBSON, ROGERS, KALISH, ZHU)

In the experiment described in Section 3.1, Zhu et al. (2007) showed that learners presented with unlabeled data drawn from a bimodal distribution whose trough does not align with a previously learned categorization boundary will shift the boundary towards this trough. Subsequent work has shown that such *category shifts* – changes to beliefs about category structure arising from unlabeled learning experiences – can be quite dramatic (Zhu et al., 2010; Gibson et al., 2010; Kalish et al., 2011; Lake and McClelland, 2011; Kalish et al., 2012, 2014).

In this experiment we considered the causes behind these observed category-shifts. If we model human category learning using the prototype or GMM model, we can view learning as a search through the parameter space defining the model. Competing hypotheses suggest different constraints on how this search is performed when unlabeled information is encountered.

My contribution *to this work involved creating a set of formalized models, finding an optimal training set and constructing and performing a human experiment showing that humans are sensitive to all parameters and do not constrain their search of the parameter space.*

5.1 Competing Hypotheses

We considered two general hypotheses: Under the first, the shifts happen because, during the initial supervised phase, participants notice and track one or more parameters of the distribution from which the labeled items are sampled, then seek to maintain a category structure that preserves the noticed parameter. For instance, in Zhu et al.'s (2007) study, the supervised phase involved learning about just two examples (one from each category), each presented 10 times with the order randomized. This experience potentially provides the learner with important information about the two classes that she may then seek to preserve when exposed to the unlabeled distribution. For example, the learner may notice that members of each category occur about equally frequently during the supervised phase. In the unsupervised phase, she may then select a category boundary that divides the unlabeled items approximately in half, preserving this frequency information. Alternatively, the learner might notice that the two categories both have approximately equal variance, and so might learn category structures that preserve roughly equal variation between members of the category.

Since the unlabeled distribution in the original study was bimodal, symmetrical about the trough with peaks of equal width, either of these strategies would lead the learner to shift the boundary to this trough. Indeed, there are many elements of the unsupervised and supervised distributions that differed in this study, any one of which might account for the observed changes in categorization behavior.

The first hypothesis, then, was that learners are trying to preserve specific parameters of the item and label distribution learned during the initial supervised phase. We referred to this as the *heuristic hypothesis*, since there is no principled reason for choosing to preserve a particular parameter from the labeled distribution. Moreover, note that there are several possible variants of the heuristic hypothesis: participants may try to preserve the relative frequencies of the two categories, their variances, their distance from the boundary, and so on.

The second hypothesis was that human beings are true semi-supervised learners – that is, they learn the category structures most likely to have generated all of the observations, labeled and unlabeled, subject to particular implicit assumptions about the relation between labeled and unlabeled examples. In the semi-supervised mixture model described by Zhu et al.

(2007), the assumptions were that (i) items are sampled from a distribution in the feature space that is a mixture of Gaussian components and (ii) items sampled from the same component of the mixture receive the same category label. With these assumptions, it is possible to estimate, from all labeled and unlabeled items, the most likely components of the mixture (and their parameters) and the most likely labels associated with each component. We referred to this as the *SSL hypothesis*.

This experiment attempted to adjudicate which of these hypotheses best explains category-shifts that occur following exposure to unlabeled examples, as documented in prior work.

5.2 Constrained Expectation Maximization Models

To address the question we first formulated a set of models and then attempted to determine which model or models best fit human behavior on a classification task.

Task Definition

The study was performed as a 1D binary classification task (feature values $x \in [0, 1]$ with labels $y \in \{0, 1\}$). We made the strong, yet common, assumption that humans are making use of a Gaussian Mixture Model (GMM). Formally, we defined the parameters of a two-component GMM as $\theta = \{w_0, \mu_0, \sigma_0^2, \mu_1, \sigma_1^2\}$, and let $\Theta = \{\theta\}$, the set of all parametrization of this model. The learner was presented first with a set of labeled items: $L = \{(x_i, y_i)\}$, $i = 1 \ldots n_L$, drawn from a 2-component GMM defined by θ_L, followed by a set of unlabeled items $U = \{(x_j)\}$, $j = n_L + 1 \ldots n_L + n_U$ drawn from another GMM with different parameters θ_U.

We assumed that, when training on L, humans find the maximum likelihood estimate (MLE) denoted $\hat{\theta}_{\mathsf{SL}} \in \Theta$. The learner was then presented with a new set of unlabeled data U which may be drawn from a different distribution than L. Learning from U amounts to performing a search in Θ for a set of parameters that best fit the observed stimuli. Under the heuristic hypotheses, humans search some subspace of Θ for the new optimum, while under the SSL hypothesis, humans search in the whole of Θ.

We also assumed the learner uses some form of expectation-maximization (EM) as the search procedure to find this optimum, the MLE on U, with $\hat{\theta}_{\mathsf{SL}}$ as the starting point for the search Dempster et al. (1977); Bishop (2007). Note that, as an optimization procedure, EM can be applied even when labeled and unlabeled items come from different distributions. Although unusual in ML, EM used on non-iid data is plausible as a mechanism for how humans adapt. Under this assumption, participants are not focused on matching or maintaining particular aspects of the labeled distribution, but are trying to find a parametric model that jointly “explains” the labeled and unlabeled distributions.

For example, humans might be only willing to change the proportion of one class to another ($\hat{w}_0$) leaving the rest of the learned parameters ($\hat{\mu}_0, \hat{\mu}_1, \hat{\sigma}_0^2, \hat{\sigma}_1^2$) fixed as they were in $\hat{\theta}_{\mathsf{SL}}$. Or, they may update both $\hat{w}_0$ and the peaks of the learned distribution ($\hat{\mu}_0,\ \hat{\mu}_1$), but remain insensitive to changes in spread, or variance ($\hat{\sigma}_0^2, \hat{\sigma}_1^2$), of the data. This behavior might be interpreted as the human learner “hanging on” to some beliefs learned on L.

Formalized Cognitive Models

With this task in mind we describe the cognitive models which were under consideration as models of human behavior.

unconstrained SL ($\hat{\theta}_{\mathsf{SL}}$) : This model is a purely supervised learner defined by the parameters $\hat{\theta}_{\mathsf{SL}}$. This model estimates the GMM parameters using the MLE over the labeled set L alone and holds them fixed over the unlabeled test data, in effect ignoring the unlabeled data. It is included as comparison, as we know humans are affected by U. Updates are made using

$$\hat{\mu}_0 = \frac{1}{n_0}\sum_{i=1}^{n_L} \mathbb{1}\{y_i = 0\} x_i \tag{5.1}$$

$$\hat{\sigma}_0^2 = \frac{1}{n_0}\sum_{i=1}^{n_L} \mathbb{1}\{y_i = 0\} (x_i - \hat{\mu}_0)^2 \tag{5.2}$$

$$\hat{w}_0 = \frac{n_0}{n_L} \tag{5.3}$$

with $n_0 = \sum_{i=1}^{n_L} \mathbb{1}\{y_i = 0\}$ ($\hat{\mu}_1$, $\hat{\sigma}_1$ are defined similarly).

unconstrained SSL ($\hat{\theta}_{SSL}$) : We specify the SSL model, defined by the parameters $\hat{\theta}_{SSL}$, before the heuristic models as all other models are derived from this unconstrained version. Consideration must be given as to whether to perform EM on the full data set ($\mathsf{L} + \mathsf{U}$) or to use $\hat{\theta}_{SL}$, the MLE on L, as initialization and perform EM on U alone. We choose the latter as it more closely approximates the situation faced by human learners in the task: initially exposed to L but with no additional feedback as they classify U. For each M-step of EM, the MLE estimates become

$$\hat{\mu_0} = \frac{\sum_{i=n_L+1}^{n} \gamma_i x_i}{\sum_{i=n_L+1}^{n} \gamma_i} \tag{5.4}$$

$$\hat{\sigma_0}^2 = \frac{\sum_{i=n_L+1}^{n} \gamma_i (x_i - \hat{\mu_0})^2}{\sum_{i=n_L+1}^{n} \gamma_i} \tag{5.5}$$

$$\hat{w_0} = \frac{1}{n_U}\sum_{i=n_L+1}^{n} \gamma_i \tag{5.6}$$

$n = n_L + n_U$, responsibilities γ_i calculated at each E-step as

$$\gamma_i = \frac{\hat{w}_0 \mathcal{N}(x_i; \hat{\mu}_0, \hat{\sigma}_0^2)}{\hat{w}_0 \mathcal{N}(x_i; \hat{\mu}_0, \hat{\sigma}_0^2) + (1 - \hat{w}_0)\mathcal{N}(x_i; \hat{\mu}_1, \hat{\sigma}_1^2)} \tag{5.7}$$

and $\hat{\mu}_1$ and $\hat{\sigma}_1$ calculated similarly using $(1 - \gamma_i)$.

All remaining models correspond to our heuristic models. They are all similar to $\hat{\theta}_{SSL}$, but assume that learning is being done by fixing *one* of the GMM parameters to the values learned on L while allowing all others to vary:

constrained means ($\hat{\theta}_\mu$) : Means $\hat{\mu_0}$ and $\hat{\mu_1}$ are fixed at the initialization values learned on L using (5.1). It is as though two prototypes are formed at the modes of the labeled distribution and retained when exposed to U. At each EM iteration t, the values of $\hat{\mu}$ at $t-1$ are simply copied forward. The variances $\hat{\sigma_0}^2, \hat{\sigma_1}^2$, weight $\hat{w_0}$ and responsibilities γ_i are updated using (5.5), (5.6) and (5.7) respectively.

fixed standard deviations ($\hat{\theta}_\sigma$) : The standard deviations $\hat{\sigma_0}$ and $\hat{\sigma_1}$ are fixed at the initialization values learned on L using (5.2). Here, it is the spread of the labeled data that is considered important, and is maintained. Again at each EM iteration the values of $\hat{\sigma_0}$ and $\hat{\sigma_1}$ are simply copied forward. Updates for means, weight and responsibilities are the same as in (5.4), (5.6) and (5.7).

fixed ratio of standard deviations ($\hat{\theta}_r$) : At initialization, the ratio of standard deviations learned on L using (5.2) is calculated as $r = \hat{\sigma}_0/\hat{\sigma}_1$. Again, the spread is considered most important, but now the spread of each class is allowed to vary only so long as the ratio between the two is maintained. As the parameters $\hat{\sigma}_0$ and $\hat{\sigma}_1$ are now tied, the parameter set becomes $\hat{\theta}_r = \{w_0, \mu_0, \mu_1, \sigma\}$. Reformulating the optimization function and solving for σ we find the new update equations

$$\hat{\sigma}^2 = \frac{1}{n_U}\left[\sum_{i=n_L+1}^{n}\left(\gamma_i(x_i-\mu_0)^2 + r^2(1-\gamma_i)(x_i-\mu_1)^2\right)\right] \quad (5.8)$$

$$\gamma_i = \frac{w_0\mathcal{N}(x_i;\hat{\mu}_0,\hat{\sigma}^2)}{w_0\mathcal{N}(x_i;\hat{\mu}_0,\hat{\sigma}^2)+(1-w_0)\mathcal{N}(x_i;\hat{\mu}_1,(\hat{\sigma}/r)^2)}. \quad (5.9)$$

Updates for means and weight are the same as in (5.4) and (5.6).

constrained weight $(\hat{\theta}_w)$: The weight parameter $\hat{w}_0$ is fixed at the initialization value learned on L. In this case it is the frequency of each class which is considered most important to retain from the labeled data. All other updates remain unchanged.

The above models each fix one property. We also consider cognitive models which constrain multiple parameters. For example, the model $\hat{\theta}_{\sigma,w}$ has only two parameters which are free to vary: $\{\hat{\mu}_0, \hat{\mu}_1\}$, with $\hat{w}_0$, $\hat{\sigma}_0^2$ and $\hat{\sigma}_1^2$ fixed. This results in 5 additional models: $\{\hat{\theta}_{\sigma,w}, \hat{\theta}_{r,w}, \hat{\theta}_{\mu,w/\sigma}, \hat{\theta}_{\mu,w/r}, \hat{\theta}_{\mu,\sigma}, \hat{\theta}_{\mu,r}\}$. The model $\hat{\theta}_{\mu,w/\sigma}$ is constrained in means and weight while standard deviations are allowed to vary. The model $\hat{\theta}_{\mu,w/r}$ is constrained in means and weight while ratio of standard deviations is allowed to vary.

The final cognitive model we examined (**propL**) is not probabilistic. In this model, the learner simply calculates the proportion of negative to positive items seen in L. When the learner is then presented with U, they attempt to place a boundary in feature space such that this proportion of negative to positive items is preserved. If the distribution generating unlabeled items is different from that generating the labeled items, the boundary learned on the labeled data will not necessarily be the same one applied to the unlabeled data. This model, $\hat{\theta}_{\mathtt{propL}}$ has only a single parameter n_0/n_L, with the boundary $\hat{b}$ induced from this ratio:

$$\hat{b} = x_{(j)} : \frac{j}{n_U} = \frac{n_0}{n_L}, \quad j \in [1, n_U] \quad (5.10)$$

where $b \in [0, 1]$ and $\{x_{(1)}, x_{(2)}, ..., x_{(n_U)}\}$ are the items in U, sorted by feature value. Note that this model is related to the cognitive models which preserve the GMM weight w_0. However, since this is not a GMM and classification is simply performed by a step function placed at the learned boundary b, the resulting behavior may be different.

With these cognitive models in hand we now discuss how we compared their performance to human behavioral data in order to assess which was the best match.

5.3 Human Experiment and Choosing a Diagnostic Dataset

We designed an experiment which attempted to discriminate which of our proposed models was a best fit to human behavior in the 1D classification task. An important aspect of this design was the construction of the dataset.

A dataset had to be found which would maximally discriminate predictions made by our various models, so that it was as clear as possible which model most strongly matched human behavior. This step was similar in flavor to the *machine teaching* task proposed in Zhu (2013). In that setting, a teacher attempts to design an optimal dataset to teach a (potentially unknown) learner a target hypothesis. The difference here was that we did not have a target we wished our learners to learn, but instead simply wanted our proposed learners to differ as much as possible in their resulting predictions. The similarity was in the search over potential datasets.

To find a good dataset, first a labeled set L of $n_L = 50$ labeled pairs were drawn from $\theta_L = \{w_0 = 0.75, \mu_0 = 0.4, \sigma_0 = 0.12, \mu_1 = 0.8, \sigma_1 = 0.06\}$. A heuristic search was then made over a sparse grid of parameter settings θ_U, varying in all parameters. At each setting a potential unlabeled set $\tilde{U}$ of $n_U = 300$ was drawn. All cognitive models were then trained on L and predictions made on that $\tilde{U}$. We heuristically selected the dataset

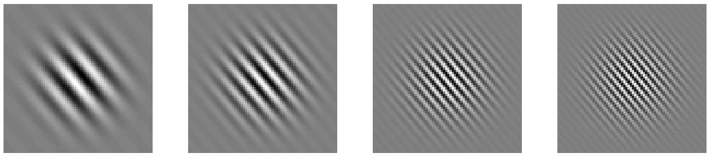


Figure 5.1: Stimuli at $x = 0,\ 0.25,\ 0.75$ and 1 respectively.

$L + \tilde{U}$ with the aim to produce the largest combined pairwise difference between predictions, and therefore largest discriminative power. Additionally, parameters which produced more than one decision boundary in the target range $x \in [0, 1]$ were avoided.

In the end the parameters selected from which U was drawn were $\theta_U = \{w_0 = 0.25, \mu_0 = 0.3, \sigma_0 = 0.05, \mu_1 = 0.6, \sigma_1 = 0.1\}$. Plots of the chosen underlying distributions are shown in Figure 5.2. Importantly note that the labeled and unlabeled distributions varied in all parameters. Figure 5.2 also shows the estimated distributions and boundaries resulting from training each of the cognitive models on the selected dataset.

Procedure

Using this chosen dataset, we performed a human experiment where 49 undergraduate students, participating for partial course credit, were asked to learn a timed classification task. The 1D stimuli used were Gabor patch images varying in only the frequency dimension, with fixed rotation (Figure 5.1). Each participant was asked to classify the $n_L = 50$ labeled images, each classification followed by feedback indicating whether they were correct or incorrect. The participant was then asked to classify the $n_U = 300$ unlabeled stimuli, with no feedback given. All participants classified the same set of stimuli, each a randomized ordering.

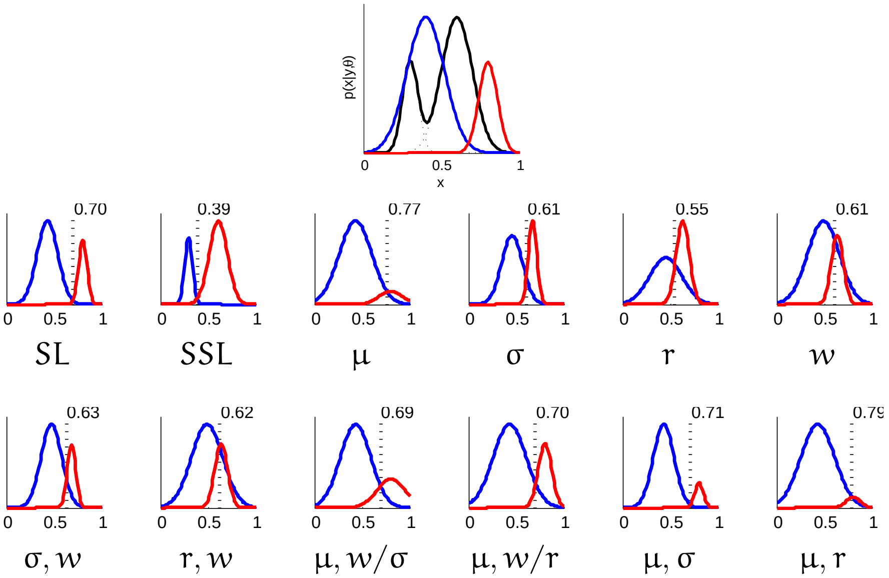


Figure 5.2: Above, the ground truth labeled distributions (in blue and red) and unlabeled distribution (in black). Below, the trained models and most central prediction boundary indicated by a dotted line. The boundary for propL falls at 0.65.

Evaluation Criteria

We call the measurement we used to evaluate our models "agreement". This refers to how well a cognitive model's classification predictions *agree* with observed human behavior. Each participant $k \in \{1, \ldots, K\}$ was asked to classify the set of labeled and unlabeled items in a randomized ordering $(L, U)^{(k)}$. For each participant k we considered the first 50 + 200 items as a training set $(L, U)^{(k)}_{train}$ and the remaining 100 items as a test set $U^{(k)}_{test}$. Though there is certainly no reason to assume that humans will not continue learning on the test set, we did make the assumption that after 200 unlabeled examples, the learned boundary will have stabilized.

Each of our proposed models m was then trained on $(L, U)^{(k)}_{train}$ producing $\hat{\theta}^{(m,k)}$. For the GMM models we used the constrained versions of EM described above while *propL* was calculated directly. We can then determine the predicted boundary $\hat{b}^{(m,k)}$ for each trained model on each dataset. For each of these model m and dataset k pairs we could then make predictions $\hat{y}^{(m,k)} = \mathbb{1}\left\{x_i^{(k)} \leqslant \hat{b}^{(m,k)}\right\}, i = 201, \ldots, 300$ and calculate:

$$\mathsf{agreement}(m,k) = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \mathbb{1}\left\{\hat{y}_i^{(m,k)} = y_i^{(k)}\right\} \tag{5.11}$$

and total mean-agreement for each model over all K participants:

$$\textit{mean-agreement}(m) = \frac{1}{K} \sum_{k=1}^{K} \mathsf{agreement}\,(m,k) \tag{5.12}$$

The mean agreement scores were then used to determine which model was the best fit over all.

Results

Using the method described above, we found that the maximum mean-agreement score is 0.7 for the completely unconstrained model $\hat{\theta}_{SSL}$, simply standard SSL (Figure 5.3, top). A repeated measures one-way ANOVA showed significant difference between model agreements per subject, $F(12, 624) = 26.68, p = 2 \times 10^{-16}$. Additionally, the unconstrained SSL model, $\hat{\theta}_{SSL}$, was a significantly better fit to human behavior than all other models (post-hoc multiple comparison test with Holm correction, $p \leqslant 0.05$), save one, SSL constrained by ratio of standard deviations ($\hat{\theta}_r$, $p = 0.11$).

If we look at which model had the best agreement per participant, unconstrained SSL $\hat{\theta}_{SSL}$ was the clear winner, having the highest agreement on 71% of participants (Figure 5.3, bottom).

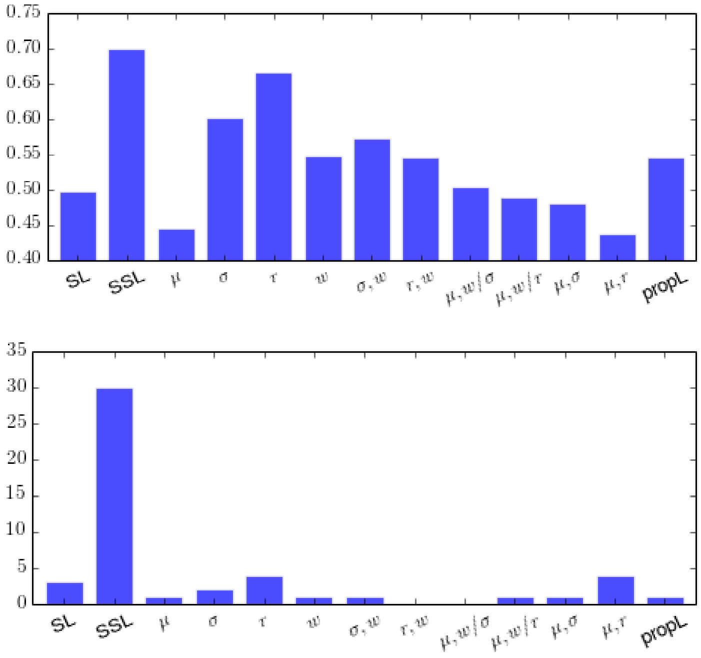


Figure 5.3: Top: mean agreement scores calculated for each model. Bottom: number of participants for which each model is the best match (highest agreement).

5.4 Discussion

The question we set out to answer was what causes the category shifts seen in many semi-supervised learning studies? The two hypotheses were 1) *heuristic*: that humans notice and track some properties or set of parameters of the distribution from which labeled items are sampled, and then seek to preserve these properties when integrating information derived from unlabeled items and 2) *SSL*: that humans are true semi-supervised learners, sensitive to all properties.

In this particular categorization task, our results supported the latter hypothesis: **humans are sensitive to all parameters and do not constrain their search of the parameter space.** They are sensitive to all changes in the unlabeled data distribution as they try to find the category structure most likely to have generated all observations, labeled and unlabeled.

This result should be of interest to both the CP and ML communities. From the CP perspective we can compare these results to those regarding the distinction between generative and discriminative learning Hsu and Griffiths (2010). Recall that to perform categorization, a generative learner attempts to model the full generating distribution $p(x, y)$ while the discriminative learner only attempts to learn a discriminating function $p(y \mid x)$. Several studies have shown that humans are capable of both types of learning Rips (1989); Smith and Sloman (1994); Hsu and Griffiths (2010). In our task where the underlying generating distribution is important due to its non-*iid* nature, the generative learning model is preferred. Our results argue that humans do in fact use a generative model for this particular task, as the SSL model is a better fit than the propL model, a discriminative model. It may be that in other tasks, where discrimination between hypothesized models, or models not in the GMM family, is still possible, this result may not be the case. Additional investigation is required to confirm that our conclusion generalizes to other situations.

From the ML perspective this result matches the intuition that, for best performance on transfer learning, the learner should not be constrained *a priori* without specific knowledge of the relation between the source domain and the target domain. The learner should be allowed to explore the full parameters space when attempting to find the best fit approximation.

Finally, though the evidence points to the unconstrained hypothesis dominating over all, no significant difference was found between it and the model constrained by ratio of standard deviations. The difference here is subtle and additional work is necessary to distinguish whether this model is in fact a good approximation of human behavior or just an artifact of the current study.

6 SEMI-SUPERVISED EFFECTS DUE TO NETWORK STRUCTURE OF UNLABELED DATA: MANIFOLD LEARNING (Gibson et al., 2010)

In the preceding studies we have seen that unlabeled data can affect human learning, and that models making use of SSL assumptions can be used to designed which reproduce the behavior Another SSL assumption that yet been fully discussed is the *manifold* assumption: that data is generated from a distribution which lies along a lower dimensional manifold in the higher dimensional feature space. In this study we examined whether humans are capable of perceiving such manifolds and making use of them in a classification task. We found that, given enough labeled data as well as hints regarding the manifold, humans are capable of propagating labels along a manifold.

My contribution *to this work involved creating a novel experimental interface and stimuli set and performing the human experiment showing that humans can learn using manifolds, given sufficient labeled data and hints regarding the manifold structure.*

6.1 Can Humans Learn Using Manifolds?

Consider a classification task where a learner is given a small set of labeled training items $\{(x_i, y_i)\}_{i \in L}$, $L = 1, \ldots, n_L$ with $x \in \mathbb{R}^2$, $y \in \{-1, 1\}$ In addition, the learner is given some unlabeled items $\{x_i\}_{i \in U}$, $U = \{n_L + 1, \ldots, n\}$, without corresponding labels. Importantly, the labeled and unlabeled items are distributed in a peculiar way in the feature space: they lie on smooth, lower dimension *manifolds*, such as those schematically shown in Figure 6.1(a). The question is: given both the labeled and unlabeled data, how will the learner classify the unlabeled data $\{x_i\}_{i \in U}$? Will the learner ignore the distribution information of the unlabeled data, and simply use the labeled data to form a decision boundary as in Figure 6.1(b)? Or will the learner propagate labels along the nonlinear manifolds as in Figure 6.1(c)?

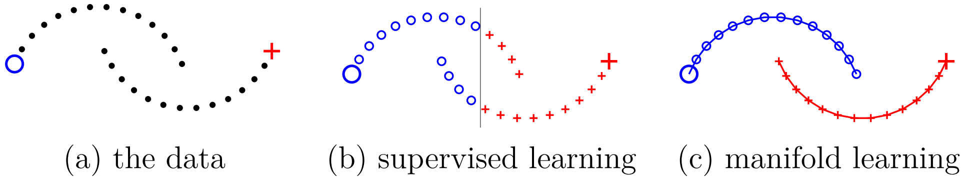

(a) the data (b) supervised learning (c) manifold learning

Figure 6.1: On a dataset with manifold structure, supervised learning and manifold learning make dramatically different predictions. Large symbols represent labeled items, dots unlabeled items.

When the learner is a ML algorithm, this question has been addressed by semi-supervised learning (Chapelle et al., 2006; Zhu and Goldberg, 2009). The designer of the algorithm can *choose* to make the manifold assumption, also known as graph-based semi-supervised learning, which states that the labels vary slowly along the manifolds or the discrete graph formed by connecting nearby items. Consequently, the learning algorithm will predict Figure 6.1(c). The mathematics of manifold learning is well-understood (Belkin et al., 2006; Sindhwani et al., 2005; Zhou et al., 2004; Zhu et al., 2003). Alternatively, the designer can choose to ignore the unlabeled data and perform supervised learning, which results in Figure 6.1(b).

When the learner is a human being, however, the answer is not so clear. Consider that the human learner does not directly see how the items are distributed in the feature space (such as Figure 6.1(a)), but only a set of items (such as those in Figure 6.2(a)). The underlying manifold structure of the data may not be immediately obvious. Thus there are many possibilities for how the human learner will behave: 1) They may completely ignore the manifold structure and perform supervised learning; 2) They may discover the manifold under some learning conditions and not others; or 3) They may always learn using the manifold.

For readers not familiar with manifold learning, the setting might seem artificial. But in fact, many natural stimuli we encounter in everyday life are distributed on manifolds. An important example is face recognition, where different poses (viewing angles) of the same face produce different 2D images. These images can be quite different, as in the frontal and profile views of a person. However, if we continuously change the viewing angle, these 2D images will form a one-dimensional manifold in a very high dimensional image space. This example illustrates the importance of a manifold to facilitate learning: if we can form and maintain such a face manifold, then with a single label (e.g., the name) on one of the face images, we can recognize all other poses of that person by propagating the label along the manifold. The same is true for visual object recognition in general. Other more abstract stimuli form manifolds, or the discrete analogue, graphs. For example, text documents in a corpus occupy a potentially nonlinear manifold in the otherwise very high dimensional space used to represent them, such as the “bag of words” representation.

There exists little empirical evidence addressing the question of whether human beings can learn using manifolds when classifying objects, and the few studies we are aware of come to opposing conclusions. For instance, Wallis and Bülthoff (2001) created artificial image sequences where a frontal face is morphed into the profile face of a different person. When participants were shown such sequences during training, their ability to match frontal and profile faces during testing was impaired. This might be evidence that people depend on manifold structure stemming from temporal and spatial proximity to perform face recognition. On the other hand, Vandist et al. (2009) conducted a categorization experiment where the true decision boundary is at 45 degrees in a 2D stimulus space (i.e., an information integration task). They showed that when the two classes are elongated Gaussian, which are parallel to, and on opposite sides of, the decision boundary, unlabeled data does not help learning. If we view these two elongated Gaussian as linear manifolds, this result suggests that people do not generally learn using manifolds.

This study sought to understand under what conditions, if any, people are capable of manifold learning in a semi-supervised setting. The study has important implications for CP: first, if people are capable of learning manifolds, this suggests that manifold-learning models that have been developed in ML can provide hypotheses about how people categorize objects in natural domains like face recognition, where manifolds appear to capture the true structure of the domain. Second, if there are reliable methods for encouraging manifold learning in people, these methods can be employed to aid learning in other domains that are structured along manifolds. For ML, our study will help in the design of algorithms which can decide when to invoke the manifold learning assumption.

6.2 Human Manifold Learning Experiments

We designed and conducted a set of experiments to study manifold learning in humans, with the following design considerations. First, the task was a "batch learning" paradigm in which participants viewed all labeled and unlabeled items at once (in contrast to "online" or sequential learning paradigm where items appear one at a time). Batch learning allows us to compare human behavior against well-established ML models that typically operate in batch mode. Second, we avoided using faces or familiar 3D objects as stimuli, despite their natural manifold structures as discussed above, because we wished to avoid any bias resulting from strong prior real-world knowledge. Instead, we used unfamiliar stimuli, from which we could add or remove a manifold structure easily. This design should allow our experiments to shed light on people's intrinsic ability to learn using a manifold.

Participants and Materials

In the first set of experiments, 139 university undergraduates participated for partial course credit. A computer interface was created to represent a table with three bins, as shown in Figure 6.2(a). Unlabeled cards were initially placed in a central white bin, with bins to either side colored red and blue to indicate the two classes $y \in \{-1, 1\}$. Each stimulus is a card. Participants sorted cards by clicking and dragging with a mouse. When a card was clicked, other similar cards could be "highlighted" in gray (depending on condition). Labeled cards were pinned down in their respective red or blue bins and could not be moved, indicated by a "pin" in the corner of the card. The layout of the cards was such that all cards remained visible at all times. Unlabeled cards could be re-categorized at any time by dragging from any bin to any other bin. Upon sorting all cards, participants would click a button to indicating completion.

Two sets of stimuli were created. The first, used solely to acquaint the participants with the interface, consisted of a set of 20 cards with animal line drawings on a white background. The images were chosen to approximate a linear continuum between fish and mammal, with shark, dolphin, and whale at the center. The second set of stimuli used for the actual experiment was composed of 82 "crosshair" cards, each with a pair of perpendicular, axis-parallel lines, all of equal length, crossing on a white background. Four examples are shown in Figure 6.2(b). Each card therefore can be encoded as $x \in [0, 1]^2$, whose two features representing the positions of the vertical and horizontal lines, respectively.

Procedure

Each participant was given two tasks to complete:

Task 1 was a practice task to familiarize the participant with the

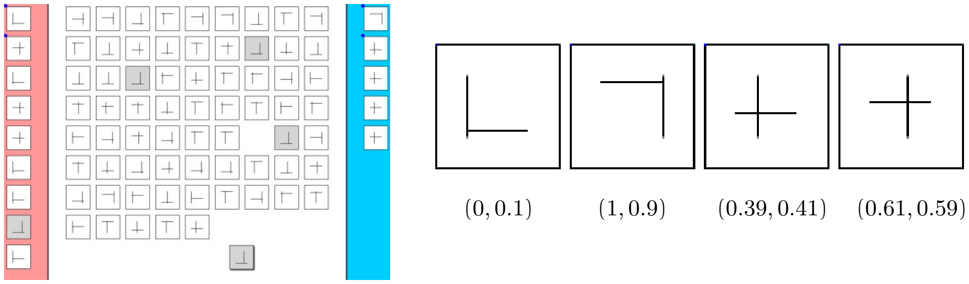


Figure 6.2: Experimental interface (with highlighting shown), and example crosshair stimuli.

interface. The participant was asked to sort the set of 20 animal cards into two categories, with the two ends of the continuum (a clown fish and a dachshund) labeled. Participants were told that when they clicked on a card, highlighting of similar cards might occur. In reality, highlighting was always shown for the two nearest-neighboring cards (on the defined continuum) of a clicked card. Importantly, we designed the dataset so that, near the middle of the continuum, cards from opposite biological classes would be highlighted together. For example, when a dolphin was clicked, both a shark and a whale would be highlighted. The intention was to indicate to the participant that highlighting is not always a clear give-away for class labels. At the end of task 1 their fish vs. mammal classification accuracy was presented. No time limit was enforced.

Task 2 asked the participant to sort a set of 82 crosshair cards into two categories. The set of cards, the number of labeled cards, and the highlighting of cards depended on condition. The participant was again told that some cards might be highlighted, whether the condition actually provided for highlighting or not. The participant was also told that cards that shared highlighting may not all have the same classification. Again, no time limit was enforced. After they completed this task, a follow up questionnaire was administered.

Conditions

Each of the 139 participants was randomly assigned to one of 6 conditions, shown in Figure 6.3, which varied according to three manipulations:

The number of labeled items l can be 2 or 4 (2^l vs. 4^l). For conditions with two labeled items, the labeled items are always $(x_1, y_1 = -1), (x_2, y_2 = 1)$; with four labeled items, they are always $(x_1, y_1 = -1), (x_2, y_2 = 1), (x_3, y_3 = 1), (x_4, y_4 = -1)$. The features of $x_1 \ldots x_4$ are those given in Figure 6.2(b). We chose these four labeled points by maximizing the prediction differences made by seven ML models, as discussed in the next section.

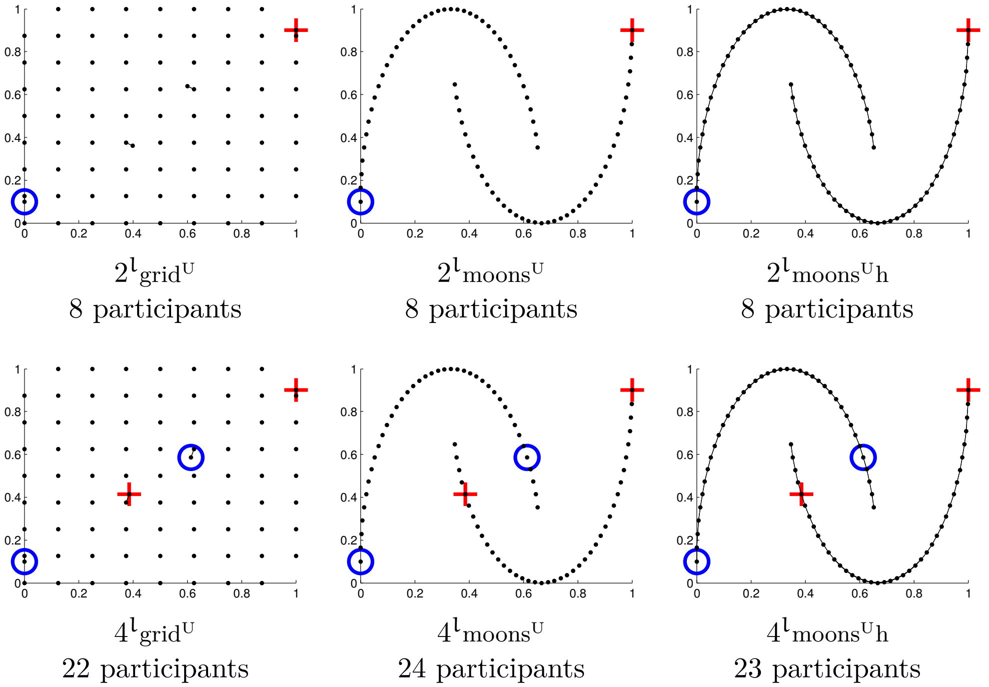
Unlabeled items are distributed on a uniform grid or manifolds (grid^U vs. moons^U). The items $x_5 \ldots x_{82}$ were either on a uniform grid in the 2D feature space, or along two "half-moons", which is a well-studied dataset in the semi-supervised learning community. No linear boundary can separate the two moons in feature space. x_3 and x_4, if unlabeled, are the same as in Figure 6.2(b).

Highlighting similar items or not (the suffix h). For the moons^U conditions, the neighboring cards of any clicked card may be highlighted. The neighborhood is defined as within a radius of $\epsilon = 0.07$ in the Euclidean feature space. This value was chosen as it includes at least two neighbors for each point in the moons^U dataset. To form the unweighted graph shown in Figure 6.3, an edge is placed between all neighboring points.

The rationale for comparing these different conditions will become apparent as we consider how different machine-learning models perform on these datasets.

6.3 Model Predictions

We hypothesized that human participants consider a set of models ranging from simple to sophisticated, and that they would perform model selection



$2^{\mathrm{l}}\mathrm{grid}^{\mathrm{U}}$
8 participants

$2^{\mathrm{l}}\mathrm{moons}^{\mathrm{U}}$
8 participants

$2^{\mathrm{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}$
8 participants

$4^{\mathrm{l}}\mathrm{grid}^{\mathrm{U}}$
22 participants

$4^{\mathrm{l}}\mathrm{moons}^{\mathrm{U}}$
24 participants

$4^{\mathrm{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}$
23 participants

Figure 6.3: The six experimental conditions. Large symbols indicate labeled items, dots unlabeled items. Highlighting is represented as graph edges.

based on the training data given to them. We started by considering seven typical ML models to motivate our choice, and present the models we actually used later on. The seven models are:

(graph) Graph-based semi-supervised learning (Belkin et al., 2006; Zhu et al., 2003), which propagates labels along the graph. It reverts to supervised learning when there is no graph (i.e., no highlighting).

(1NN,ℓ_2) 1-nearest-neighbor classifier with ℓ_2 (Euclidean) distance.

(1NN,ℓ_1) 1-nearest-neighbor classifier with ℓ_1 (Manhattan) distance. These two models are similar to exemplar models in psychology (Nosofsky, 1986).

(multi-v) multiple vertical linear boundaries.

(multi-h) multiple horizontal linear boundaries.

(single-v) a single vertical linear boundary.

(single-h) a single horizontal linear boundary.

Figure 6.4 shows the label predictions made by these 7 models on four of the six conditions. Their predictions on $2^{l}_{\text{moons}^{U}}$ are identical to $2^{l}_{\text{moons}^{U}}\text{h}$, and on $4^{l}_{\text{moons}^{U}}$ are identical to $4^{l}_{\text{moons}^{U}}\text{h}$, except that "(graph)" is not available.

For conceptual simplicity and elegance, instead of using these disparate models we adopted a single model capable of making all these predictions. In particular, we used a Gaussian Process (GP) with different kernels (i.e., covariance functions) k to simulate the seven models.[1] In particular, we found seven different kernels k to match GP classification to each of the seven model predictions on all 6 conditions. This is somewhat unusual in that our GPs were not learned from data, but by matching other model predictions. Nonetheless, it is a valid procedure to create seven different GPs which would later be compared against human data.

For models (1NN,ℓ_2), (multi-v), (multi-h), (single-v), and (single-h), we used diagonal RBF kernels $\mathrm{diag}(\sigma_1^2, \sigma_2^2)$ and tuned σ_1, σ_2 on a coarse parameter grid to minimize classification disagreement w.r.t. the corresponding model prediction on all 6 conditions. For model (1NN,ℓ_1) we used a Laplace kernel and tune its bandwidth. For model (graph), we produced a graph kernel $\tilde{k}$ following the Reproducing Kernel Hilbert Space trick in Sindhwani et al. (2005). That is, we extended a base RBF kernel k with a graph component:

$$\tilde{k}(x, z) = k(x, z) - \mathbf{k}_x^\top (I + cLK)^{-1} cL\mathbf{k}_z \tag{6.1}$$

where x, z are two arbitrary items (not necessarily on the graph), $\mathbf{k}_x = (k(x, x_1), \ldots, k(x, x_{l+u}))^\top$ is the kernel vector between x and all $l+u$ points $x_1 \ldots x_{l+u}$ in the graph, K is the $(l + u) \times (l + u)$ Gram matrix with

[1] For details on GPs see standard textbooks such as Rasmussen and Williams (2006).

$K_{ij} = k(x_i, x_j)$, L is the unnormalized graph Laplacian matrix derived from unweighted edges on the ϵNN graph defined earlier for highlighting, and c is the parameter that we tuned. We took the base RBF kernel k to be the tuned kernel for model (1NN,ℓ_2). It can be shown that $\tilde{k}$ is a valid kernel formed by warping the base kernel k along the graph, see Sindhwani et al. (2005) for technical details. We used the GP classification implementation with Expectation Propagation approximation (Rasmussen and Williams, 2007). In the end, our seven GPs were able to *exactly* match the predictions made by the seven models in Figure 6.4.

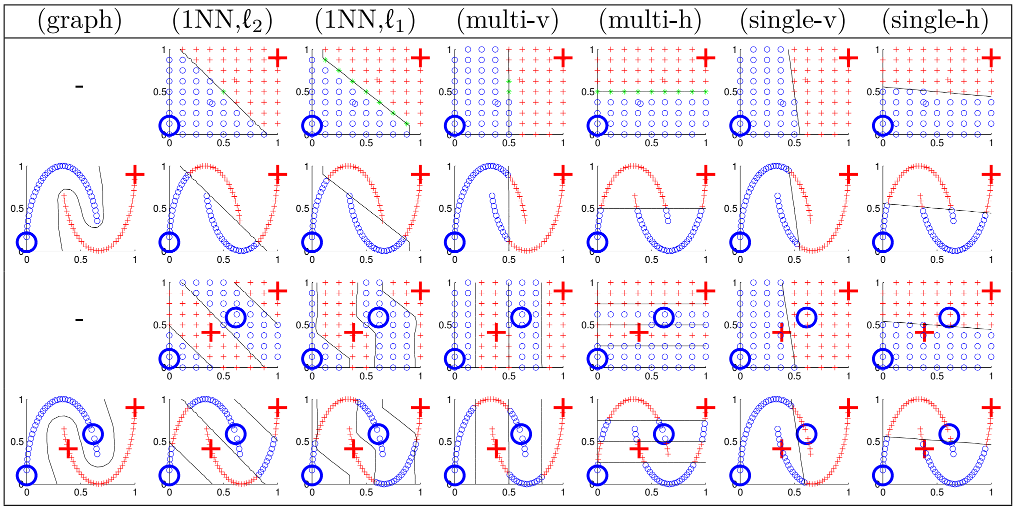


Figure 6.4: Predictions made by the seven models on 4 of the 6 conditions. Rows correspond to $2^{l}_{\text{grid}^{U}}$, $2^{l}_{\text{moons}^{U}h}$, $4^{l}_{\text{grid}^{U}}$ & $4^{l}_{\text{moons}^{U}h}$ respectively

6.4 Behavioral Experiment Results

Using the models described in the previous section, we can compare human categorization behaviors to model predictions. We first consider the aggregate behavior for all participants within each condition. One way to characterize this aggregate behavior is the “majority vote” of the participants on each item. That is, if more than half of the participants classified an item as $y = 1$, the majority vote classification for that item is $y = 1$, and so on. The first row in Figure 6.5 shows the majority vote for each condition. In these and all further plots, blue circles indicate $y = -1$, red pluses $y = 1$, and green stars ambiguous, meaning the classification into positive or negative is half-half. We also compute how well the seven GPs predict human majority votes. The accuracies of these GP models are shown in Table 6.1.[2]

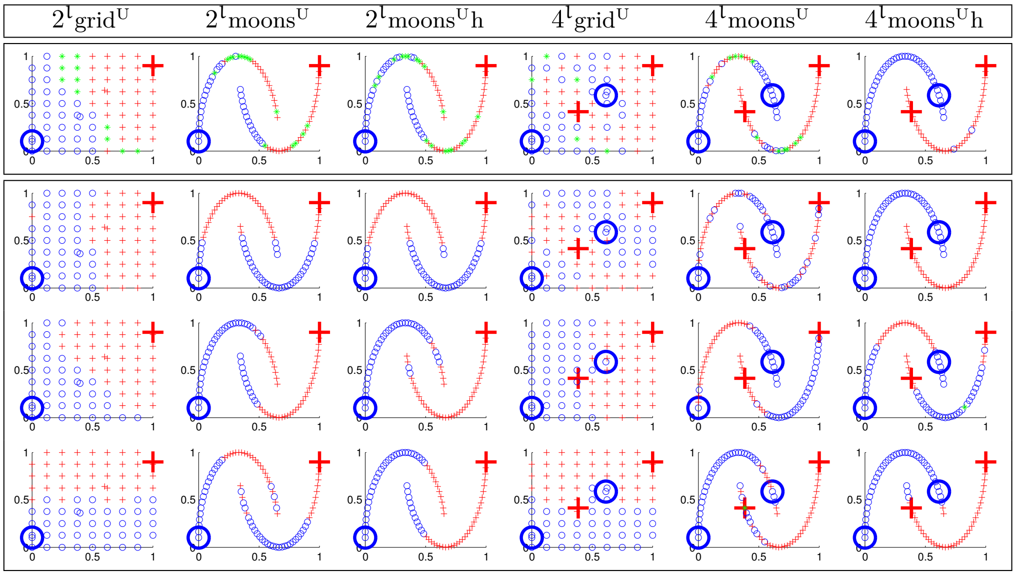

2^{l}grid^{U} 2^{l}moons^{U} $2^{l}\text{moons}^{U}\text{h}$ 4^{l}grid^{U} 4^{l}moons^{U} $4^{l}\text{moons}^{U}\text{h}$

Figure 6.5: Human categorization results. (First row) the majority vote of participants within each condition. (Bottom three rows) a sample of responses from 18 different participants.

Of course, a majority vote only reveals average behavior. We have observed that there are wide participant variabilities. Participants appeared to find the tasks difficult, as their self-reported confidence scores were fairly low in all conditions. It was also noted that strategies for completing the task varied widely, with some participant simply categorizing cards in the

[2] The condition $4^{l}\text{moons}^{U}\text{h}^{R}$ will be explained later in Section 6.5.

	(graph)	(1NN,ℓ_2)	(1NN,ℓ_1)	(multi-v)	(multi-h)	(single-v)	(single-h)
$2^{\mathsf{l}}\mathrm{grid}^{\mathrm{U}}$	0.81	**0.94**	0.84	0.86	0.58	0.85	0.61
$2^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}$	0.47	**0.84**	0.62	0.74	0.42	0.79	0.45
$2^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}$	0.50	**0.78**	0.56	0.76	0.36	0.76	0.39
$4^{\mathsf{l}}\mathrm{grid}^{\mathrm{U}}$	0.54	0.61	**0.64**	**0.64**	0.50	0.60	0.51
$4^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}$	0.64	0.62	0.60	**0.69**	0.47	0.38	0.45
$4^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}$	**0.97**	0.76	0.54	0.64	0.31	0.65	0.26
$4^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}^{\mathrm{R}}$	**0.68**	0.63	0.44	0.56	0.40	0.59	0.42

Table 6.1: GP model accuracy in predicting human majority vote for each condition.

order they appeared on the screen, while others took a much longer, studied approach. Most interestingly, different participants seem to use different models, as the individual participant plots in the bottom three rows of Figure 6.5 suggest. We would like to be able to make a claim about what model, from our set of models, each participant used for classification. In order to do this, we compute *per participant* accuracies of the seven models on that participant's classification. We then find the model M with the highest accuracy for the participant, out of the seven models. If this highest accuracy is above 0.75, we declare that the participant is potentially using model M; otherwise no model is deemed a good fit and we say the participant is using some "other" model. We show the proportion of participants in each condition attributed to each of our seven models, plus "other", in Table 6.2.

	(graph)	(1NN,ℓ_2)	(1NN,ℓ_1)	(multi-v)	(multi-h)	(single-v)	(single-h)	other
$2^{\mathsf{l}}\mathrm{grid}^{\mathrm{U}}$	0.12	0	0.12	**0.25**	**0.25**	0.12	0	0.12
$2^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}$	0	0.12	0	**0.25**	**0.25**	**0.25**	0	0.12
$2^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}$	0.12	0	0	**0.38**	0.25	0	0	0.25
$4^{\mathsf{l}}\mathrm{grid}^{\mathrm{U}}$	0	0.05	0.09	0	0	0.18	0.09	**0.59**
$4^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}$	0.25	0.25	0.12	0.12	0	0.04	0.08	**0.38**
$4^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}$	**0.39**	0.09	0.09	0.04	0.04	0	0.13	0.22
$4^{\mathsf{l}}\mathrm{moons}^{\mathrm{U}}\mathrm{h}^{\mathrm{R}}$	0.13	0.03	0.07	0	0	0.07	0.03	**0.67**

Table 6.2: Percentage of participants potentially using each model

Based on Figure 6.5, Table 6.1, and Table 6.2, we make some observations:

1. When there are only two labeled points, the unlabeled distribution does not encourage humans to perform manifold learning (comparing $2^{l}_{\text{grid}^{U}}$ vs. $2^{l}_{\text{moons}^{U}}$). That is, they do not follow the possible implicit graph structure ($2^{l}_{\text{moons}^{U}}$). Instead, in both conditions they prefer a simple single vertical or horizontal decision boundary, as Table 6.2 shows.[3]

2. With two labeled points, even if they are explicitly given the graph structure in the form of highlighting, participants still do not perform manifold learning (comparing $2^{l}_{\text{moons}^{U}}$ vs. $2^{l}_{\text{moons}^{U}\text{h}}$). It seems they are "blocked" by the simpler vertical or horizontal hypothesis, which perfectly explains the labeled data.

3. When there are four labeled points but no highlighting, the distribution of unlabeled data still does not encourage people to perform manifold learning (comparing $4^{l}_{\text{grid}^{U}}$ vs. $4^{l}_{\text{moons}^{U}}$). This further suggests that people can not easily extract manifold structure from unlabeled data in order to learn, when there is no hint to do so. However, most participants have given up the simple single vertical or horizontal decision boundary, because it contradicts with the four labeled points.

4. Finally, when we provide the graph structure, there is a marked switch to manifold learning (comparing $4^{l}_{\text{moons}^{U}}$ vs. $4^{l}_{\text{moons}^{U}\text{h}}$). This suggests that a combination of the elimination of preferred, simpler hypotheses, together with a stronger graph hint, finally gives the originally less preferred manifold learning model a chance of being used. It is under this condition that we observed human manifold learning behavior.

[3] The two rows in Table 6.1 for these two conditions are therefore misleading, as it averages classification made with vertical and horizontal decision boundaries. Also note that in the 2^{l}conditions (multi-v) and (multi-h) are effectively single linear boundary models (see Figure 6.4) and differ from (single-v) and (single-h) only slightly due to the training method used.

6.5 Humans do not Blindly Follow Suggestions

Do humans really learn using manifolds? Could they have adopted a “follow-the-highlighting” procedure to label the manifolds 100% correctly: in the beginning, click on a labeled card x to highlight its neighboring unlabeled cards; pick one such neighbor x' and classify it with the label of x; now click on (the now labeled) x' to find one of its unlabeled neighbors x'', and repeat? Because our graph has disconnected components with consistently labeled seeds, this procedure will succeed. The procedure is known as propagating-1NN in semi-supervised learning (Algorithm 2.7, Zhu and Goldberg, 2009). In this section we present three arguments that humans are not blindly following the highlighting.

First, participants in $2^{l}\mathrm{moons}^{u}\mathrm{h}$ did not learn the manifold while those in $4^{l}\mathrm{moons}^{u}\mathrm{h}$ did, even though the two conditions have the same ϵNN highlighting.

Second, a necessary condition for follow-the-highlighting is to always classify an unlabeled x' according to a labeled highlighted neighbor x. Conversely, if a participant classifies x' as class y', while all neighbors of x' are either still unlabeled or have labels other than y', she could not have been using follow-the-highlighting on x'. We say she has taken a leap-of-faith on x'. The $4^{l}\mathrm{moons}^{u}\mathrm{h}$ participants had an average of 17 leaps-of-faith among about 78 classifications,[4] while strict follow-the-highlighting procedure would yield zero leaps-of-faith.

Third, the basic challenge of follow-the-highlighting is that the underlying manifold structure of the stimuli may have been irrelevant. Would participants have shown the same behavior, following the highlighting, regardless of the actual stimuli? We therefore designed the following experiment. Take

[4] The individual number of leaps-of-faith were 0, 1, 2, 4, 10, 13, 13, 14, 14, 15, 15, 16, 18, 19, 20, 21, 22, 24, 25, 27, 33, 36, and 36 respectively, for the 23 participants.

the $4^{l}\text{moons}^{U}\text{h}$ graph which has 4 labeled nodes, 78 unlabeled nodes, and an adjacency matrix (i.e., edges) defined by ϵNN, as shown in Figure 6.3. Take a random permutation $\pi = (\pi_1, \ldots, \pi_{78})$. Map the feature vector of the ith unlabeled point to x_{π_i}, while keeping the adjacency matrix the same. This creates the random-looking graph in Figure 6.6(a) which we call $4^{l}\text{moons}^{U}\text{h}^{R}$ condition (the suffix R stands for random), which is equivalent to the $4^{l}\text{moons}^{U}\text{h}$ graph in structure. In particular, there are two connected components with consistent labeled seeds. However, now the highlighted neighbors may look very different than the clicked card.

If we assume humans blindly follow the highlighting (perhaps noisily), then we predict that they are more likely to classify those unlabeled points nearer (in shortest path length on the graph, not Euclidean distance) a labeled point with the latter's label; and that this correlation should be the same under $4^{l}\text{moons}^{U}\text{h}^{R}$ and $4^{l}\text{moons}^{U}\text{h}$. This prediction turns out to be false. 30 additional undergraduates participated in the new $4^{l}\text{moons}^{U}\text{h}^{R}$ condition. Figure 6.6(b) shows the above behavioral evaluation, which does not exhibit the predicted correlation, and is clearly different from the same evaluation for $4^{l}\text{moons}^{U}\text{h}$ in Figure 6.6(c). Again, this is evidence that humans are not just following the highlighting. In fact, human behavior in $4^{l}\text{moons}^{U}\text{h}^{R}$ is similar to 4^{l}moons^{U}. That is, having random highlighting is similar to having no highlighting in how it affects human categorization. This can be seen from the last rows of Tables 6.1 and 6.2, and Figure 6.6(d).[5]

6.6 Discussion

These results suggest that people can perform manifold learning, but only when there is no alternative, simpler explanation of the data, and people need strong hints about the graph structure.

[5] In addition, if we create a GP from the Laplacian of the random highlighting graph, the GP accuracy in predicting $4^{l}\text{moons}^{U}\text{h}^{R}$ human majority vote is 0.46, and the percentage of participants in $4^{l}\text{moons}^{U}\text{h}^{R}$ who can be attributed to this model is 0.

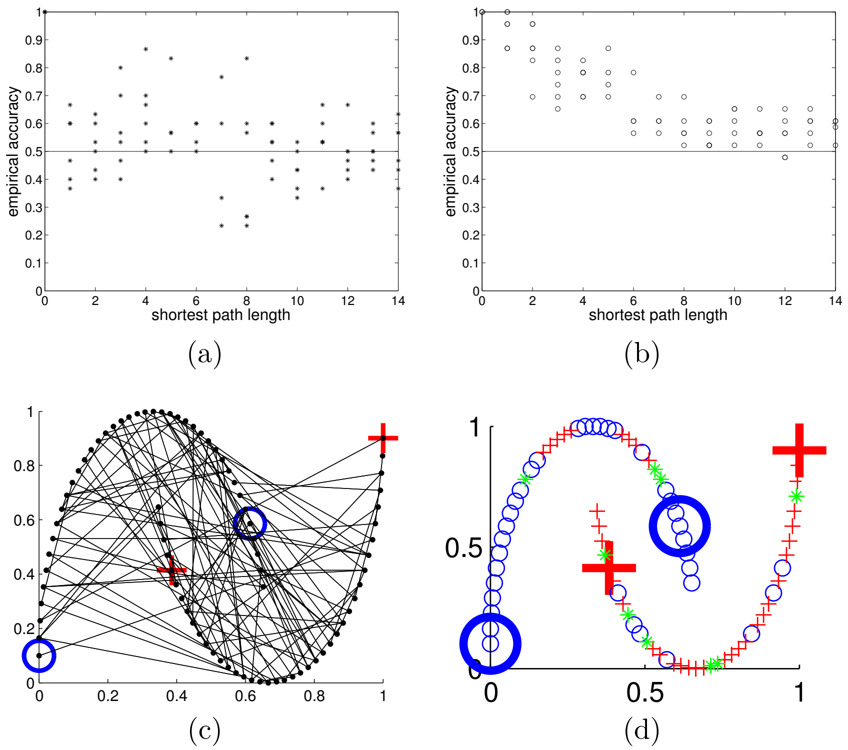


Figure 6.6: The $4^{l}_{\text{moons}}{}^{U}\text{h}^{R}$ experiment with 30 participants. (a) The behavioral evaluation for $4^{l}_{\text{moons}}{}^{U}\text{h}^{R}$, where the x-axis is the shortest path length of an unlabeled point to a labeled point, and the y-axis is the fraction of participants who classified that unlabeled point consistent with the nearest labeled point. (b) The same behavioral evaluation for $4^{l}_{\text{moons}}{}^{U}\text{h}$. (c) The $4^{l}_{\text{moons}}{}^{U}\text{h}^{R}$ condition itself. (d) The majority vote in $4^{l}_{\text{moons}}{}^{U}\text{h}^{R}$.

We propose that Bayesian model selection is one possible way to explain these human behaviors. Recall we defined seven Gaussian Processes, each with a different kernel. For a given GP with kernel k, the evidence $p(y_{1:l} \mid x_{1:l}, k)$ is the marginal likelihood on labeled data, integrating out the hidden discriminant function sampled from the GP. With multiple candidate GP models, one may perform model selection by selecting the one with the largest marginal likelihood. From the absence of manifold learning in conditions without highlighting or with random highlighting, we speculate that the

GP with the graph-based kernel $\tilde{k}$ (6.1) is special: it is accessible in a participant's repertoire only when strong hints (highlighting) exists and agrees with the underlying unlabeled data manifold structure. Under this assumption, we can then explain the contrast between the lack of manifold learning in $2^{l}\mathrm{moons}^{U}\mathrm{h}$, and the presence of manifold learning in $4^{l}\mathrm{moons}^{U}\mathrm{h}$. On one hand, for the $2^{l}\mathrm{moons}^{U}\mathrm{h}$ condition, the evidence for the seven GP models on the two labeled points are: (graph) 0.249, (1NN,ℓ_2) 0.250, (1NN,ℓ_1) 0.250, (multi-v) 0.250, (multi-h) 0.250, (single-v) 0.249, (single-h) 0.249. The graph-based GP has slightly lower evidence than several other GPs, which may be due to our specific choice of kernel parameters in (6.1). In any case, there is no reason to prefer the GP with a graph kernel, and we do not expect humans to learn on manifold in $2^{l}\mathrm{moons}^{U}\mathrm{h}$. On the other hand, for $4^{l}\mathrm{moons}^{U}\mathrm{h}$, the evidence for the seven GP models on those four labeled points are: (graph) 0.0626, (1NN,ℓ_2) 0.0591, (1NN,ℓ_1) 0.0625, (multi-v) 0.0625, (multi-h) 0.0625, (single-v) 0.0341, (single-h) 0.0342. The graph-based GP has a small lead over other GPs. In particular, it is better than the evidence 1/16 for kernels that treat the four labeled points essentially independently. The graph-based GP obtains this lead by warping the space along the two manifolds so that the two positive (resp. negative) labeled points tend to co-vary. Thus, there is a reason to prefer the GP with a graph kernel, and we do expect humans to learn on manifold in $4^{l}\mathrm{moons}^{U}\mathrm{h}$.

We also explored the convex combination of the seven GPs as a richer model for human behavior: $k(\lambda) = \sum_{i=1}^{7} \lambda_i k_i$, where $\lambda_i \geqslant 0, \sum_i \lambda_i = 1$. This allows a weighted combination of kernels to be used, and is more powerful than selecting a single kernel. Again, we optimize the mixing weights λ by maximizing the evidence $p(y_{1:l} \mid x_{1:l}, k(\lambda))$. This is a constrained optimization problem, and can be easily solved up to local optimum (because evidence is in general non-convex) with a projected gradient method, given the gradient of the log evidence. For the $2^{l}\mathrm{moons}^{U}\mathrm{h}$ condition, in 100 trials with random starting λ values, the maximum evidence always converges to 1/4, while the optimum λ is not unique and occupies a subspace $(0, \lambda_2, \lambda_3, \lambda_4, \lambda_5, 0, 0)$ with $\lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 = 1$ and mean $(0, 0.27, 0.25, 0.22, 0.26, 0, 0)$. Note the weight for the graph-based kernel λ_1 is zero. In contrast, for the $4^{l}\mathrm{moons}^{U}\mathrm{h}$ condition, in 100 trials λ overwhelmingly converges to $(1, 0, 0, 0, 0, 0, 0)$ with evidence 0.0626. i.e., it again suggests that people would perform manifold learning in $4^{l}\mathrm{moons}^{U}\mathrm{h}$.

Of course, this Bayesian model selection analysis is over-simplified. For instance, we did not consider people's prior $p(\lambda)$ on GP models, i.e., which model they would prefer before seeing the data. It is possible that humans favor models which produce axis-parallel decision boundaries. Defining and incorporating non-uniform $p(\lambda)$ priors is a topic for future research.

7 INFLUENCING HUMAN BEHAVIOR: VIA PRIOR UNLABELED DATA EXPOSURE (PENDING PUBLICATION)

Imagine a child playing in a classroom. She is about to take a lesson on categorizing things by “sink or float.” On the table are numerous objects such as wood, metal, plastic bottles, rocks, heavy things, light things, and so on. The teacher has not arrived yet; and there is no tub of water to experiment with. She can explore the objects but nothing will tell her whether each object sinks or floats. Can just playing with these objects, experience prior to teaching, speed up her learning of the categorization on sink or float once the lesson starts?

This work focused on such human categorization tasks. A learner, the child in our example, must learn a mapping $\mathsf{f} : \mathsf{X} \mapsto \mathsf{Y}$ from item to category label. In our classroom story, the examples are the objects and the labels are sink or float. Playing with the objects before the lesson can be considered as exposure to *unlabeled data* in that they are presented without category labels. The lesson itself will be *labeled*, where examples are presented along with their corresponding labels according to the underlying concept. The question becomes: what effect does the unlabeled data have on the speed with which the supervised categorization task is learned?

My contribution *to this work involved constructing and performing a human experiment showing that the speed of human learning on a supervised task can be affected by prior unlabeled experience.*

Existing SSL literature in CP assumes that the learner is aware of an upcoming category learning task, and that labeled data always come first to define such a supervised learning task, while unlabeled data is either inter-mixed with labeled data, or comes after labeled data as test items Zhu et al. (2007); Vandist et al. (2009); Gibson et al. (2010); Rogers et al. (2010); Zhu et al. (2010); Kalish et al. (2011); Zhu et al. (2011).

We felt that in many situations, it is far easier (and more natural) to expose a human student to unlabeled experience first rather than later. This exposure can happen even before the student is aware of any future classification task. In ML, it is known that several SSL models can take advantage of prior exposure of unlabeled data sampled iid from the marginal $p(x)$ to facilitate future classification using $p(y \mid x)$. For instance, the unlabeled data can be used to determine the parameters of a Gaussian Mixture Model (GMM), and future labeled data only needs to map each mixture component to a label. However, it was not clear whether human learners benefit from such prior exposure to unlabeled data too, as this "unlabeled data before labeled data" setting is uncommon in the CP literature.

Taking this one step further, we felt that it was also unnecessary to restrict ourselves to conventional SSL assumptions and only expose the student to iid unlabeled data sampled from $p(x)$. Taking cues from recent advances in computational teaching models such as curriculum learning Bengio et al. (2009); Khan et al. (2011), we considered whether we could design a special unlabeled data sequence that is particularly good at guiding future supervised learning? Note that the crucial difference with respect to curriculum learning is that our sequence was *unlabeled* rather than labeled. This was uncharted territory: not only was there no previous cognitive study of such a setting, but also there were no ML SSL models specifically designed for this setting.

We called this setting *Semi-Supervised Teaching* (SST). In SST, the world generates labeled training items and future test items as iid samples from an unobserved joint distribution $p(x, y)$. The learner's goal is to learn a good classifier $f : X \mapsto Y$ to perform well on future test items. This aspect is identical to standard supervised learning. However, there is also a helpful teacher who knows $p(x, y)$, and who wants to help the learner learn faster by exposing the learner to selected unlabeled items before (supervised) learning starts. These unlabeled items need not follow the marginal distribution $p(x)$.

The closest work to SST is perhaps the Test-Item Effect from Zhu et al.

(2010) discussed in Section 3.1. That study involved how predicting the category of test items, *without receiving corrective feedback*, can drastically change a human's category decision boundary. In ML terms, merely applying a classifier f to the test set (without knowing the true label of the test items) changes f itself. The argument there was that Test-Item Effect can be beneficial as a way to correct undesirable biases in previous human category learning, if the teacher can design an appropriate test set. The main difference between Test-Item Effect and SST is that SST presents unlabeled data before supervised learning. This seemingly minor distinction has major ramifications. In Test-Item Effect, the learner needs to apply her current classifier to the test items x and predict a label $\mathsf{f}(\mathsf{x})$. This is equivalent to doing homework on the test items without feedback. In contrast, in SST the learner need not know about future categorization tasks; she does not have a classifier f already, and she does not need to do the homework of categorizing the unlabeled items x to $\mathsf{f}(\mathsf{x})$. Instead, she merely needs to observe the unlabeled items x. This opens up some interesting possibilities. For example, although not studied in this work, it might be possible to present the unlabeled data passively, or subconsciously, to the learner in order to achieve increased speed on subsequent supervised learning.

Having defined SST, the immediate questions were: Does SST have any effect on humans in reality, be it positive or negative? If so, could we explain it with a computational model? This work answered both questions affirmatively. Our contributions were two-fold: 1) we performed a human experiment which shows that unlabeled data *does* have an effect on subsequent categorization learning in humans, but that learning is significantly affected only by the distribution of the unlabeled data but not the order of the unlabeled. and 2) we showed that we can model this behavior using standard SSL models.

7.1 Human Experiment

To study if unlabeled prior experience has any effect on subsequent human category learning, we conducted the following new behavioral experiment.

Participants and Procedure

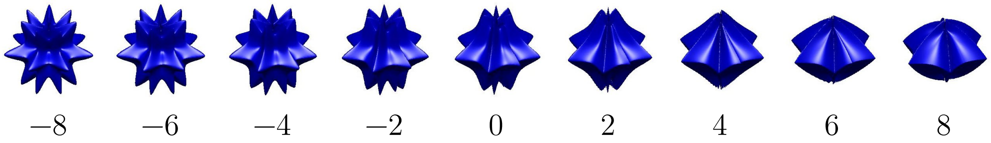


Figure 7.1: Range of example stimuli with corresponding x values.

80 undergraduate university students participated in the study in exchange for partial course credit. They were each presented with a series of 410 stimuli varying in shape according to a single parameter $x \in [-8, 8]$ (Figure 7.1). Each participant performed the following sequence of tasks:

1. Instructional one-back task ($t = 1$–10)
    a) stimulus x_t presented on screen
    b) participant responds *same* / *different* compared to x_{t-1}
    c) correct one-back response displayed
2. unlabeled exposure (one-back task) ($t = 11$–310)
    a) stimulus x_t presented on screen
    b) participant responds *same* / *different* compared to x_{t-1}
3. supervised learning task ($t = 311$–410)
    a) stimulus x_t presented on screen
    b) participant predicts binary class label $\hat{y}_t$

c) *correct* / *incorrect* feedback by comparing $\hat{y_t}$ to y_t

In order for the unlabeled data to have an effect, we need to ensure that the human learner is paying attention to the data (rather than, say, clicking through stimuli without attending). To enforce this attention to the stimuli, a meaningful response regarding the stimuli was asked from the learner. However, being unlabeled, these items must be presented without any information on the subsequent categorization task, so asking the learner to provide a category label $\hat{y}$ was not appropriate. Instead, participants were asked to perform a "one-back" comparison. Participants needed to determine if the current item x_t was identical to the immediately previous item x_{t-1}, responding *same* or *different*. It is important to keep in mind that these one-back responses are completely different from the subsequent categorization labels[1].

In task 1, the 10 unlabeled items shown to participants corresponded to the extremes of the stimuli range ($x = \{-8, 8\}$), accompanied by instructions on how to perform the one-back comparison.

In task 2, each participant was exposed to 300 unlabeled items, corresponding to one of four conditions to be described shortly. Participants performed the one-back comparison to ensure attention.

In task 3, participants categorized 100 items drawn iid uniformly from the stimuli space $x \in [-8, 8]$. Each participant was presented with each item x_t and asked to predict a binary category label $\hat{y}_t$ for that item. The participant was then told whether their $\hat{y}_t$ was correct or incorrect compared to the true labeled y_t, determined by a boundary fixed at $x = -1.6$. With

[1] For the one-back task, unlabeled data needed to be constructed such that identical items appear in sequence with reasonable frequency. To accomplish this, for each dataset, 300 unlabeled items were first created according to condition. From this sequence 120 items (40%) were randomly selected to be identical one-back trials. These selected items where then copied, overwriting the next item in the sequence, resulting in a dataset of 300 items with 40% identical one-back pairs. Note this procedure does not significantly change the distribution or order of unlabeled items with respect to the subsequent supervised task.

this feedback, the participant was expected to gradually learn the true decision boundary.

To summarize, unlabeled data exposure happens in tasks 1 and 2, and supervised category learning happens in task 3.

Conditions

To determine whether this unlabeled exposure had an effect of subsequent supervised learning, participants were randomly split into four conditions. These conditions varied how items were generated for the unlabeled exposure task, specifically in how the unlabeled items were distributed and ordered (see Figure 7.2). In all conditions, the same stimuli were used for the final supervised task.

The 4 conditions were as follows:

- The **trough** condition was motivated by earlier work on human SSL showing that unlabeled data drawn from a mixture model $p(x)$ could reinforce a previously learned boundary, if the boundary coincides with the trough in $p(x)$ Zhu et al. (2007). In this condition unlabeled examples were drawn iid from a 2-component GMM with the same weights and variances $\{w = 0.5,\ \sigma^2 = 0.64\}$ but different means: $\mu_{\mathbf{trough}} = \{-4.8, 1.6\}$. Note that the decision boundary of the subsequent task 3 falls between the modes. The expectation was that this condition would help supervised learning in task 3.

- The **peak** condition was similar except that the GMM was shifted $\mu_{\mathbf{peak}} = \{-1.6, 4.8\}$. The left peak, not the trough, coincided with task 3 decision boundary. We expected that this condition would harm supervised learning in task 3.

- The **uniform** condition was included as a control. In this condition unlabeled examples were drawn iid from uniform$[-8, 8]$. We expect this condition to neither help nor harm learning in task 3.

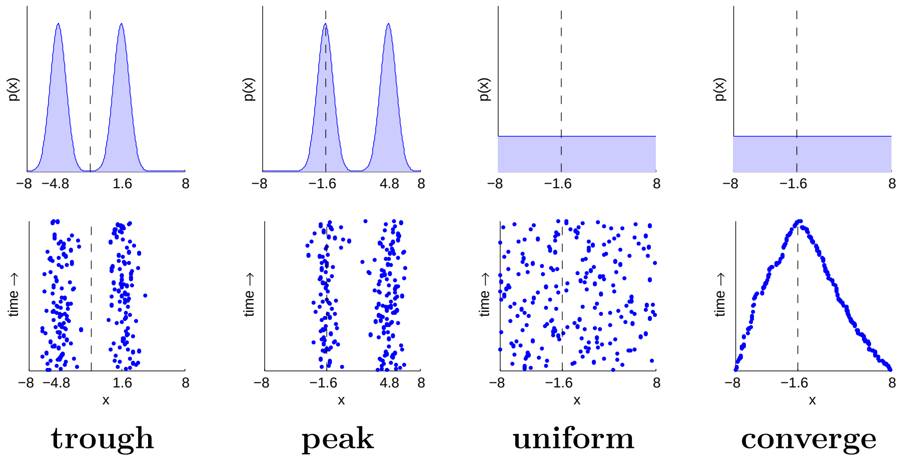


Figure 7.2: Plots describing the four conditions of the human experiment. Each column corresponds to one condition. The top row shows the underlying distributions $p(x)$ from which unlabeled items are drawn in each condition. The bottom row shows the order of unlabeled items as displayed to the learner over time. The dashed line in all plots indicates the true decision boundary in the subsequent categorization task. Note that unlabeled items in the **uniform** and **converge** conditions are both drawn from a uniform distribution over the stimuli space, but that the ordering of the data over time is very different.

- The **converge** condition was inspired by curriculum learning, where sequential ordering of labeled examples from hard to easy is important in guiding a learner toward the decision boundary Bengio et al. (2009); Khan et al. (2011). This condition differed from curriculum learning in that no labels were provided with the examples. To the best of our knowledge no study had looked at the effect of unlabeled data ordering on subsequent category learning. Unlabeled data in this condition was created by first sampling unlabeled items $x \sim \text{uniform}[-8, 8]$ just as in the **uniform** condition. We then ordered the unlabeled items such that they "converged" over time towards the subsequent decision boundary. Standard SSL models that assume unlabeled data are exchangeable would perceive no difference between the **uniform** and **converge** conditions. Since our “curriculum learning” was unlabeled, it was not clear how human learners will perform in this condition.

Results

As we were interested in the effect of exposure to unlabeled data on the *speed* of learning the categorization task, simple accuracy is not appropriate. Instead we used a logistic mixed effects model. With this we could look at both initial accuracy (intercept) and the speed of learning. Using this test we found a significant differences in both initial accuracy and speed of learning between the “trough” and “peak” conditions ($p < 0.001$). The distribution of the unlabeled experience did have an effect on subsequent learning. This followed our expectations from standard SSL models and prior experiments.

While there were some indications that the ordering of the data in the “converge” condition did influence the learner, we did not find a significant difference between the “uniform” and “converge” conditions. The ordering of the unlabeled data did not have a significant effect on the speed of learning.

7.2 Modeling

Having shown that humans are affected by prior unlabeled items, we constructed a computational model which reproduced a difference in behavior between conditions similar to that seen in humans. We chose to model human behavior using a DPMM for two reasons: 1) this was shown to be the best fit to human behavior in Chapter 4 and 2) the learner, prior to the labeled task, had no reason to assume any fixed number of components, making a GMM inappropriate. Additionally, the DPMM is flexible enough to allow each item to be its own component, making modeling using KDE unnecessary.

Being a non-parametric model, the only tuning necessary for the DPMM was to set the mixture hyperparameter which specifies how likely a new cluster or partition will be created for each item observed. Training on the 310 unlabeled items plus the first 50 labeled items, we chose from a set of potential values the mixture hyperparameter setting which provided the largest agreement with human behavior on the last 100 labeled items in each dataset. The best agreement was found at $\alpha = 5$.

Using this hyperparameter setting, we trained four separate DPMM models, one for each condition, producing predicted labels on all test sets. We then compared model predictions between conditions using the same methods used when evaluating human performance.

The results indicated behavior very similar to that seen in humans: 1) a significant difference between trough and peak conditions ($p < 0.00003$) and 2) no statistically significant difference between uniform and converge. This second finding is not surprising as the DPMM treats items as exchangable such that ordering information is discarded. If there had been a difference in human performance between uniform and converge, a standard DPMM would no longer be a viable model.

7.3 Discussion

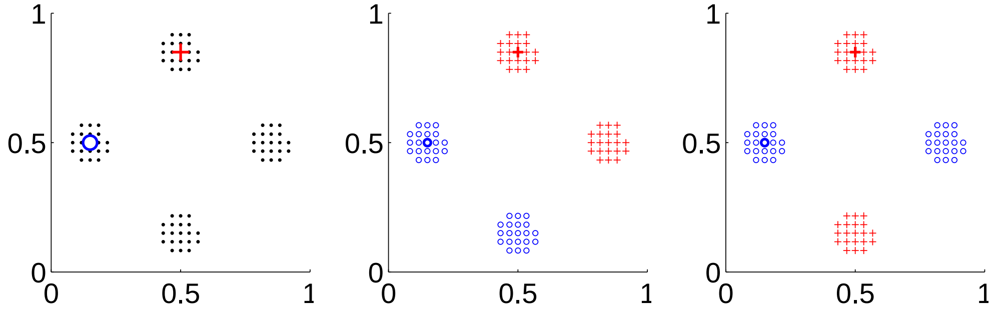
In this work we proposed the concept of Semi-Supervised Teaching: the construction of an unlabeled dataset which could potentially speed learning on a subsequent labeled task. We showed that SST is relevant to human category learning, as the latter is influenced by distribution (and possibly ordering) of prior unlabeled data exposure. We also showed that this difference in behavior can be modelled using a SSL DPMM.

8 INFLUENCING HUMAN BEHAVIOR: CO-TRAINING CONSTRAINTS (ZHU, GIBSON, ROGERS, 2011)

Though human learning abilities are remarkable in many respects, they are also constrained in ways that may seem puzzling to machine learning. As one example, people can have difficulty learning nonlinear decision boundaries without extensive supervision (Love, 2002). As another example, psychologists often distinguish between feature dimensions that are "separable" versus "integral". For separable features (e.g. color and shape), people can selectively attend to one dimension without processing the other. For integral dimensions (e.g. color saturation and brightness) they cannot. In learning problems that are identical from a machine-learning point of view, humans can show quite different patterns of behavior depending on whether the dimensions are integral or separable. For instance, people have difficulty learning non-axis-parallel boundaries for separable but not for integral feature dimensions (Nosofsky and Palmeri, 1996; Ashby and Maddox, 1990).

This work considered whether these characteristics of human learning can be altered by leveraging insights from a machine learning algorithm, namely Co-Training. Co-Training uses unlabeled data to improve learning by encouraging agreement among multiple "base" machine learners, each exposed to a different "view" of the data (see below). The classic Co-Training algorithm (Blum and Mitchell, 1998) and its extensions such as Co-EM (Nigam and Ghani, 2000), Tri-Training (Zhou and Li, 2005), and multiview learning (Brefeld et al., 2006) have enjoyed considerable empirical success and theoretical justification (Johnson and Zhang, 2007; Balcan and Blum, 2010) in machine learning.

One often under-appreciated fact about Co-Training is that it has a different inductive bias, and so can produce quite different classification results from supervised learning. Figure 8.1(a) shows a "diamond" dataset with four clusters, with just one labeled item from each class (blue and red



(a) data (b) supervised (c) Co-Training

Figure 8.1: On this "diamond" dataset, supervised learning and Co-Training, both with 1NN classifiers, produce drastically different outcomes.

points). The task is to classify the unlabeled items (black dots). Supervised learning with the 1-nearest-neighbor (1NN) algorithm[1] learns a diagonal decision boundary in Figure 8.1(b). In contrast, with the same 1NN as base learners the Co-Training algorithm learns a very different solution (Figure 8.1(c)), grouping the top and bottom clusters together in the red class, and the left and right clusters in the blue class.

The linearly non-separable classification achieved by Co-Training is just the kind of solution that human beings have difficulty learning without extensive supervision (Love, 2002). In this work we considered whether the Co-Training algorithm can be used to design a *collaboration policy* for human participants that will promote learning of such "difficult" classifications over the linearly separable outcomes that individuals are prone to acquire on their own. Under this policy, each individual in the collaboration is treated as a "base" learner; each is exposed to a different "view" of the data; and the learning set-up is designed to promote agreement among the collaborators. We empirically assessed behavior in such teams for learning problems with both psychologically-separable and integral stimulus dimensions, and compared performance to individual learners and to teams

[1] 1NN classifiers are closely related to the Generalized Context Model (Nosofsky, 1986) in CP which we discussed in Chapter 2.

collaborating under an alternative policy. In simple learning problems like that shown in Figure 8.1 we will see that our Co-Training collaboration policy leads participants to learn classifications typically thought to be very difficult for humans, and also to show more homogeneous behavior for stimuli defined along separable versus integral dimensions. Though we do not extend the approach to a real-world learning problem here, we will consider how the approach might be used to design collaboration policies for such problems in cases where individuals have difficulty learning the appropriate classifications.

My contribution *to this work consisted of implementation of a novel experimental interface, the design and norming of two stimuli datasets, overseeing the experiment itself and finally performing the analysis showing that, using a variation of the classic Co-Training constraints, we can elicit behavior from human collaborators that is not observed without these constraints.*

8.1 Review of the Co-Training Algorithm

We first review the classic Co-Training algorithm of Blum and Mitchell (1998) as it is closely related to our policy. Assume that each item is parametrized by a feature vector $\mathbf{x}$ and has a corresponding class label y. The input consists of labeled items $\{(\mathbf{x}_i, y_i)\}_{i \in L}$, $L = \{1, \ldots, n_L\}$ and unlabeled items $\{\mathbf{x}_i\}_{i \in U}$, $U = \{n_L + 1 \ldots n\}$. The goal is to learn a classifier $f : \mathbf{x} \mapsto y$ using both the labeled and unlabeled data.

Further assume that the feature vector can be split into two parts (called "views"): $\mathbf{x} = \binom{\mathbf{x}^{(1)}}{\mathbf{x}^{(2)}}$. The Co-Training algorithm trains two base learners $f^{(1)} : \mathbf{x}^{(1)} \mapsto y$ and $f^{(2)} : \mathbf{x}^{(2)} \mapsto y$, each working exclusively on one view. In the beginning, these two base learners are trained on the labeled data. More specifically, $f^{(1)}$ is trained with the first view of the labeled data $(\mathbf{x}_1^{(1)}, y_1) \ldots (\mathbf{x}_{n_L}^{(1)}, y_{n_L})$. Subsequently, whenever $f^{(1)}$ encounters an item $\mathbf{x}$

Algorithm 5: The Co-Training algorithm

```
Input: labeled and unlabeled data where each item has two views;
       learning speed $\mathsf{s}$.
Initialize $\mathsf{L}_1 = \mathsf{L}_2 =$labeled data
repeat
    Train $\mathsf{f}^{(1)}$ from $\mathsf{L}_1$, $\mathsf{f}^{(2)}$ from $\mathsf{L}_2$.
    Classify unlabeled items with $\mathsf{f}^{(1)}$, $\mathsf{f}^{(2)}$ separately.
    Add $\mathsf{f}^{(1)}$'s top $\mathsf{s}$ most confident predictions
      $(\mathbf{x}, \mathsf{f}^{(1)}(\mathbf{x}))$ to $\mathsf{L}_2$, and vice versa.
    Remove these items from the unlabeled data.
until unlabeled data is exhausted;
```

during training or prediction, it always works with the first view $\mathbf{x}^{(1)}$ of the item only and disregards the second view $\mathbf{x}^{(2)}$. $\mathsf{f}^{(2)}$ operates similarly, working only with the second view. The ingenuity is in how the unlabeled data is utilized in an iterative fashion: At each iteration, $\mathsf{f}^{(1)}$ classifies a few unlabeled items that it is most confident about and passes these and their predicted labels as *additional training data* to $\mathsf{f}^{(2)}$. Simultaneously, $\mathsf{f}^{(2)}$ reciprocates. Co-Training then updates both base learners with this additional "pseudo-labeled" data. This repeats until the unlabeled data is exhausted. A slightly simplified version of Blum and Mitchell's Co-Training algorithm is given in Algorithm 5. To classify a new test item $\widetilde{\mathbf{x}}$, one can compare the predictions $\mathsf{f}^{(1)}(\widetilde{\mathbf{x}}^{(1)})$ and $\mathsf{f}^{(2)}(\widetilde{\mathbf{x}}^{(2)})$ and pick the one with higher confidence.

Co-Training is a "wrapper" method in that the two base learners $\mathsf{f}^{(1)}$ and $\mathsf{f}^{(2)}$ can be any learning systems. The only requirement is that each base learner has a notion of confidence, which is used to select which unlabeled items to turn into pseudo labeled data for the other view. Importantly for this work, being a wrapper method enables Co-Training to treat two human collaborators as the base learners.

It is important to understand the conditions under which Co-Training will succeed. We present the sufficient conditions in the original analysis (Blum and Mitchell, 1998), but with new interpretations geared toward our collaboration policy for human learning.

The conditions are:

1. The unlabeled data distribution and the target concept f are *compatible* under the two views. In particular, let $\mathsf{p}(\mathbf{x})$ be the marginal distribution of items. We require that with probability one, $\mathbf{x} \sim \mathsf{p}(\mathbf{x})$ satisfies $\mathsf{f}^{(1)}(\mathbf{x}^{(1)}) = \mathsf{f}^{(2)}(\mathbf{x}^{(2)})$. That is, no item shall have conflicting labels between the two views.

2. Each base learner is able to learn the target concept under its view, given enough labeled data. This refers to standard supervised learning, where the amount of labeled data required may be much larger than in Co-Training.

3. The two views are conditionally independent given the class label: $\mathsf{p}(\mathbf{x}^{(2)} \mid \mathbf{x}^{(1)}, \mathsf{y}) = \mathsf{p}(\mathbf{x}^{(2)} \mid \mathsf{y})$. If one knows the class y, then knowing the features in one view $\mathbf{x}^{(1)}$ does not help one guess the other view $\mathbf{x}^{(2)}$. This condition ensures that the most confident items from $\mathsf{f}^{(1)}$'s perspective do not "pile up on top of each other" from $\mathsf{f}^{(2)}$'s perspective. Rather, they spread out and provide representative (pseudo) training data for the second view.

In subsequent sections, we will see how consideration of these conditions shape our collaboration policy.

The reader might wonder why Co-Training keeps the two views separate. Why not stack the two views back into $\mathbf{x} = \left(\begin{smallmatrix}\mathbf{x}^{(1)}\\ \mathbf{x}^{(2)}\end{smallmatrix}\right)$, and train a supervised learner on $\mathbf{x}$? One reason is their inductive biases leading to *different* classifiers for the same data, as shown in Figure 8.1. To see why this happens, consider how the base learners respond to the bottom and right clusters. For the bottom cluster, the x-axis view will be highly confident that the items belong to the red class because from this view they are nearly identical to the labeled red item. In contrast, the class of the right cluster will be uncertain from this view, since these items are not particularly similar to either labeled item. So, the x-view learner may choose to label some bottom cluster items and pass these to the y-view learner. For the y-view learner, the reverse pattern occurs: the right cluster items very likely belong to the blue class, whereas the class of the bottom cluster items is uncertain. Each view is confident about the items for which the opposing view is uncertain. Thus the two views, working together, converge on the solution shown in Figure 8.1(c). Such difference between supervised learning and Co-Training is general and can be observed with other datasets and choices of base learners. Another example is given in the last section.

8.2 Human Collaboration Policies

We now consider how these ideas from Co-Training can be used to shape a policy for human collaboration. The task we consider is category learning: Two human collaborators are given a number of labeled training items $\{(\mathbf{x}_i, y_i)\}_{i \in L}$ and together must label the unlabeled items $\{\mathbf{x}_i\}_{i \in U}$. One may view the labeled training items as *teaching experiences* given to the collaborators, e.g., by a teacher or a senior worker. It is reasonable to assume that in many cases the availability of teaching is limited. Therefore, the goal is for the dyad to grasp the target concept using as little teaching experience as possible. We assume that the collaborators can see all of the unlabeled items upfront, which is known as transduction in machine learning.[2]

Our main interest is in exploring different *collaboration policies* between the two learners and how these policies affect the learning outcomes. One obvious policy is to allow the two collaborators full access to the data

[2] However, the dyad is also capable of making *inductive* inferences when faced with new test items later on.

Algorithm 6: The Co-Training collaboration policy

```
Input: labeled and unlabeled data, learning speed $s$.
Present the first-view data to Alice, second-view to Bob.
repeat
    Let Alice label her $s$ most confident unlabeled
      items; same for Bob.
    Show Bob's labelings $(\mathbf{x}_{B1}, y_{B1}) \ldots (\mathbf{x}_{Bs}, y_{Bs})$ to
      Alice, and vice versa.
    Remove $\{\mathbf{x}_{A1} \ldots \mathbf{x}_{As}\} \cup \{\mathbf{x}_{B1} \ldots \mathbf{x}_{Bs}\}$ from the
      unlabeled data.
until unlabeled data is exhausted;
```

$\{(\mathbf{x}_i, y_i)\}_{i \in L}, \{\mathbf{x}_i\}_{i \in U}$, and to allow them to fully interact with each other (in terms of discussions, gesturing, etc.). We call this the "full-collaboration" policy. Another policy might be to isolate the learners so that they each have full independent access to the data but cannot communicate or interact. We call this the "no-collaboration" policy.

We introduce a third policy, described in Algorithm 6 and explained below, that is inspired by and closely follows the Co-Training machine learning algorithm. This policy splits each item's feature vector into two views: $\mathbf{x} = \begin{pmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \end{pmatrix}$. The intention is to allow each collaborator only one of the views. In contrast to machine learning, however, it is sometimes impossible to create artificial stimuli with a single view. For instance, the often used Gabor patches (Vandist et al., 2009) vary in frequency and orientation, and it is impossible to depict an orientation without any information about frequency or vice versa. In this case, our policy constructs artificial stimuli that vary along the "viewed" dimension while holding a constant value on the "hidden" dimension (specifically the mean μ of the values on the missing view). So if Alice and Bob are the two collaborators, Alice might see the stimuli as $\mathbf{x}^{(1)}$ or $\begin{pmatrix} \mathbf{x}^{(1)} \\ \mu^{(2)} \end{pmatrix}$, while Bob sees them as $\mathbf{x}^{(2)}$ or $\begin{pmatrix} \mu^{(1)} \\ \mathbf{x}^{(2)} \end{pmatrix}$. Both Alice and Bob also see the labels for the labeled data.

Alice and Bob cannot directly communicate. Instead, at each iteration the policy requires both Alice and Bob to label the s unlabeled items that each is most confident about. After they have both finished, the policy shows Bob's chosen items and labelings $(\mathbf{x}_{B1}, y_{B1}) \ldots (\mathbf{x}_{Bs}, y_{Bs})$ to Alice. Note that, although Bob labeled these item from his view, Alice sees them from her own view. Alice understands that the labels come from Bob, but – in contrast to machine learning – it is up to her whether to believe Bob's labelings (i.e., whether to use them as pseudo labeled data). At the same time, Alice's labelings are shown to Bob. The policy then removes any unlabeled item that has been labeled by *either* Alice or Bob, and proceeds to the next iteration. This repeats until the unlabeled data is exhausted. In the end, each unlabeled item has received a label from Alice or Bob. In the rare cases when both Alice and Bob label the same item differently, the policy breaks the tie arbitrarily.

The only communication that is allowed in the Co-Training policy is label exchange.[3] In this sense, Co-Training falls between the no-collaboration and full-collaboration policies. Our main question is whether the Co-Training policy leads learners toward different classification outcomes than the no-collaboration and full-collaboration policies. We hypothesize that human behavior in the Co-Training policy will be well-predicted by the behavior of the Co-Training algorithm in machine learning, whereas participants will primarily learn linear category boundaries in the other two collaboration conditions. This is not a trivial hypothesis given the differences between human and machine learning discussed above, and the general difficulty human beings have in learning nonlinear decision boundaries without extensive supervision.

[3] In theory, Alice and Bob could agree on a *coding scheme* a priori and encode further information with their choices of items and labelings. We do not consider that possibility here.

8.3 Human Experiments

We designed and conducted a series of experiments to compare human category learning behaviors under the three collaboration policies introduced in the previous section.

Participants and Materials

Across three separate experiments a total of 324 undergraduate students participated for course credit under IRB approval. We programmed networked software to run on a pair of computers so that two participants in separate rooms could collaborate according to the Co-Training policy, preventing any communication between them except that explicitly allowed by the software. The software also runs on a single computer for the full-collaboration and no-collaboration policies. The software was implemented in the ActionScript programming language and runs in Flash Player.

The category learning task was implemented as a card sorting game, see Figure 8.2. Each item $\mathbf{x}$ is represented as a card. The user interface contains a central bin holding the unlabeled cards as well as a bin to the left and to the right into which labeled cards are placed. In the beginning, only the initially-labeled cards are shown in the left or right bins. The participants' task is to sort all cards in the central bin into the left or right bins. Before starting the experiments, participants were told whether or not they would be working with a partner, and were instructed to begin with the card they were most confident about.

We assessed learning behavior in all collaboration conditions with two stimulus sets. Both included items defined over two continuous perceptual features, but differed in the psychological separability of the dimensions. The "separable" set contained Gabor patches varying in spatial frequency and orientation of the grating (Vandist et al., 2009; Ashby and Maddox, 1990). These dimensions are considered separable because it is possible for

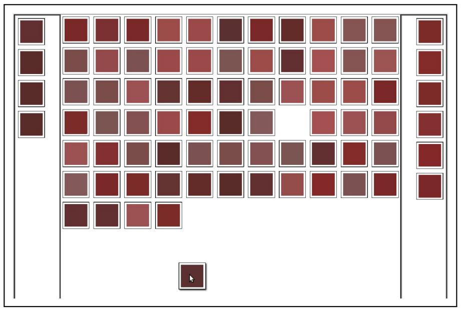


Figure 8.2: Experimental interface

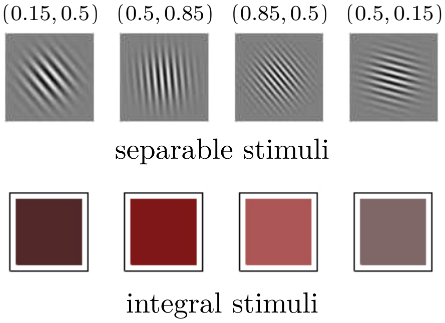


Figure 8.3: Sample stimuli

people to attend to one dimension to the exclusion of the other (Shepard, 1964). The "integral" set contained colored squares of a fixed hue but varying in saturation and brightness. These dimensions are considered to be integral because it is difficult for people to attend to one dimension without also processing the other (Lockhead, 1966). Extensive research has shown that people respond differently to stimuli defined on separable versus integral dimensions in supervised and unsupervised learning tasks (Ashby and Maddox, 1990; Love, 2002; Nosofsky and Palmeri, 1996).

In both cases a stimulus is parametrized by $\mathbf{x} = \left(\begin{smallmatrix} x_1 \\ x_2 \end{smallmatrix}\right) \in [0, 1]^2$. The range of values on each was determined in extensive pilot testing to ensure that participants could discriminate important distances along all dimensions. For Gabor patches, the frequencies were calculated using $\lambda = (x_1 * 5/34) + 2/17$, and the orientations were calculated using $\theta = x_2 * 100$, varying from 0 to 100 degrees clockwise from horizontal. For colored squares, the brightness was calculated using $b = x_1 * 0.5 + 0.25$ and the saturation was calculated using $c = x_2 * 0.9 + 0.5$. Figure 8.3 shows four stimuli corresponding to the cluster centers in Figure 8.1(a), in both the separable and integral stimulus spaces.

The Diamond Dataset and Co-Training Conditions

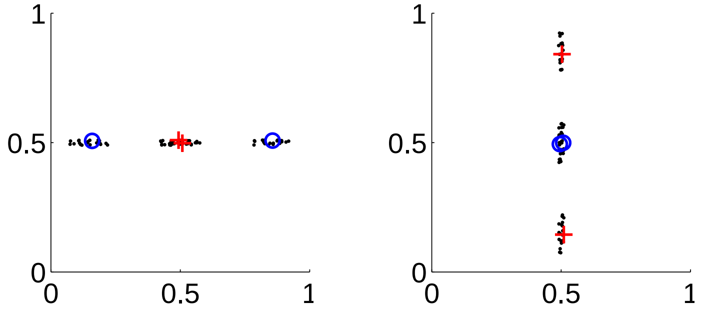
Most of our experiments employ the diamond dataset shown in Figure 8.1(a). It consists of $n = 80$ items evenly divided into 4 clusters. All clusters have radius 0.1. Items within a cluster lie on a regular grid. The two views are the x-axis and y-axis coordinates paired with the mean value of 0.5 on the hidden dimension as previously discussed.

We constructed this dataset with the aim of satisfying the three technical conditions for the Co-Training algorithm. Condition 1 is easy to verify: there exists at least one target concept f, shown in Figure 8.1(c), that is consistent with the marginal $p(\mathbf{x})$. In other words, no item receives contradictory labels across the two views (note this is not true for the concept in Figure 8.1(b)). From the Figure we can also verify that Condition 3 is approximately true:[4] For both classes, knowing an item's x-axis position tells us little about its y-axis position and vice versa.

Condition 2 cannot be verified by consideration of the stimulus set alone. It stipulates that each base learner in Co-Training must, with full supervision and sufficient labeled data, be capable of learning the target concept from only one view. Because the base learners in our study are human beings, we need to determine empirically whether this condition holds. Our first experiment addresses this question.

[**Experiment 1**] 13 participants were divided into two groups: 7 in the first-view group and 6 in the second-view group. Each worked alone as a base learner, and viewed stimuli from the "integral" stimulus set. Participants in the first-view condition saw items varying in the x dimension but fixed at 0.5 on the y dimension, whereas those in the second-view condition saw items varying along the y dimension and fixed at 0.5 in the x dimension, effectively collapsing the dataset into one dimension as shown in Figure 8.4. Participants viewed four labeled items corresponding to the four cluster centers in Figure 8.1(a), and were asked to classify the remaining 76 items.

[4] It would be exactly true if the clusters were squares, not circles.



(a) first-view (b) second-view

Figure 8.4: In Experiment 1 each participant worked with only one view of the dataset. There were four labeled items. Points dithered to show overlap.

Note that this labeled data is twice what is provided in Co-Training. The purpose of the study is to verify that, when provided with this supervised experience, human learners are capable of learning the target concept as it is projected in one view.

Result: The average classification accuracy on the unlabeled items was quite high: 98.9% in the first-view group and 94.7% in the second-view group. These results suggest that people were able to learn the target concept f using only one view in a supervised learning setting given *four* labeled training items, thus verifying the final technical condition of Co-Training. Another pilot study also showed that in Experiment 1, humans cannot learn the concept in Figure 8.1(c) if they saw only the *two* labeled items in Figure 8.1(a) instead of the four. However, as we show next, they will be able to learn it from two labeled items if they perform Co-Training label exchange.

8.4 Results under Different Policies

[**Experiment 2**] Our second experiment compares human learning on the diamond dataset under the Co-Training, full-collaboration, and no-collaboration policies, now using just two labeled items as in Figure 8.1(a).

These three policies were implemented as follows:

Co-Training (`C`): Two partners sit in separate rooms working on a shared categorization task. Each partner sees one of the views and no communication is permitted except through the labeling of cards. Each partner labels one card ($\mathtt{s} = 1$) and is then asked to wait for the other partner. The card labeled by the other partner is highlighted and automatically moves from the unlabeled bin to the appropriate labeled bin. If the partners have by chance labeled the same card, that card is automatically moved from the labeled bin, across the unlabeled bin, into the other labeled bin. This process of labeling followed by viewing is repeated until all cards are labeled.

Full-collaboration (`F`): Two partners sit side-by-side before a single computer working on the same categorization task. They are able to view both features on each card simultaneously. No restriction is made on their communication.

No-collaboration (`N`): A single participant categorizes all cards while viewing both features simultaneously.

Each collaboration policy was paired with the separable (`S`) or integral (`I`) stimuli, resulting in 6 conditions. Participants were assigned randomly to conditions as follows: 21 dyads for `CS`, 25 dyads for `CI`; 20 dyads for `FS`, 26 dyads for `FI`; 45 singles for `NS`, and 34 singles for `NI`.

To summarize the results of a given dyad or individual, we classified each cluster in the diamond dataset as either "red" or "blue" based on a simple majority vote (i.e. the cluster was designated red if more than 50% of the items in it were classified as red, and blue otherwise). Thus there were $2^4 = 16$ different possible patterns for the four clusters. Table 8.1 shows the proportion of participants whose behavior matched each of these patterns across the different conditions. For example, in the `CS` condition, $17/21 \approx 0.8$ fraction of dyads produced the "cross" pattern.

Several observations can be made from Table 8.1. First, the Co-Training policy robustly produces the nonlinear "cross" pattern in about three quarters

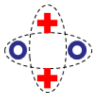
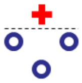
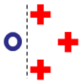
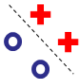
			pattern		
	cross	horz	vert	diag	other
condition					
CS	**0.80**	0.10	0	0.10	0
CI	**0.68**	0.04	0.04	0.20	0.04
FS	0.05	0.25	**0.35**	0.30	0.05
FI	0	0.08	0	**0.92**	0
NS	0.07	**0.42**	0.18	0.31	0.02
NI	0	0	0	**1.00**	0

Table 8.1: The fraction of patterns in cluster-level majority classification. "Other" includes the remaining $16 - 4 = 12$ possible patterns. Boldface indicates the largest fraction within a condition.

of the dyads. This pattern was rarely observed in the full-collaboration and the no-collaboration policies (χ^2 test, $p \ll 0.01$), which both mainly produce linear decision boundaries. This is the main finding of our work: the Co-Training human collaboration policy leads to outcomes dramatically different from no-collaboration and full-collaboration policies, and consistent with that predicted by the machine learning algorithm.

Second, in the full-collaboration and no-collaboration policies, participants showed quite different behaviors for stimuli defined over separable versus integral dimensions, producing axis-parallel boundaries with separable dimensions and "integrated" oblique boundaries with integral dimensions. This pattern has been previously documented in a variety of work in cognition. In Co-Training, however, the separability of the stimulus dimensions does not affect behavior (CS vs. CI, χ^2 test, $p = 0.5$). This is not surprising given that each person sees only one view, but it suggests an interesting application of the policy: Co-Training can enforce consistent classification regardless of the separability of the stimulus dimensions.

Additionally, there was no significant difference between full- and no-collaboration (χ^2 test, $p = 0.7$). Thus the differences observed under the

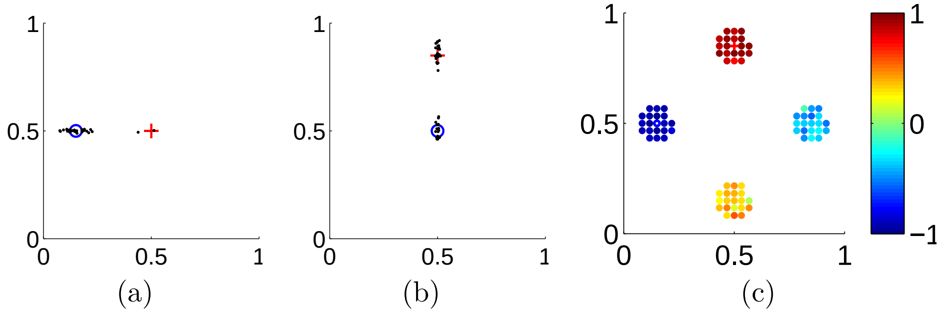
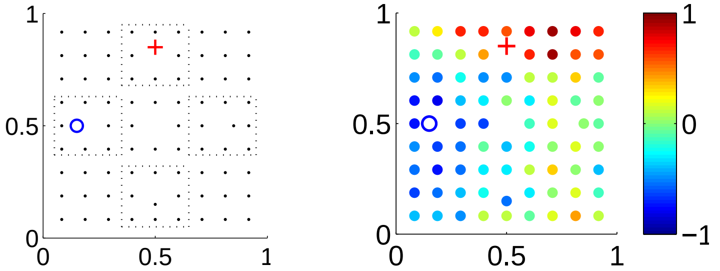


Figure 8.5: Differences between humans and machines (aggregated over `CS` and `CI`). (a) The first unlabeled items (black dots) chosen by the first-view partners. (b) Same, but for the second-view. (c) Per-item average labels.

Co-Training policy were not simply the result of having two individuals working together. Although the Co-Training human collaboration outcome fits machine learning model predictions at the cluster level, we observed some subtle differences suggesting that machine learning algorithms like 1NN may not be the ideal models for human base learners. One difference concerns the unlabeled items that humans label first. Machine base learners would label the items they are most confident about, which will likely be an item that *overlaps* with a labeled item under that view. Participants in our experiments did not always pick such overlapping items, but seemed to settle for items loosely similar to labeled ones, see Figure 8.5(a) and (b).

Another difference is in how sure the humans are. For each unlabeled item, we may average its classification across all dyads in the Co-Training conditions where, if the average is close to -1 (blue) or 1 (red), all dyads label it consistently; 0 if they are quite unsure. Figure 8.5(c) shows this per-item average using a color coding. Items in the top and left clusters (with labeled items) are very certain, while those in the bottom and right clusters are relatively uncertain (though they do have the correct per-item majority vote label). Typical machine Co-Training learners will have higher certainty on these clusters.



(a) The dataset (b) Per-item average labels

Figure 8.6: The counter-example

8.5 A Counter-Example

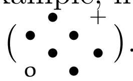
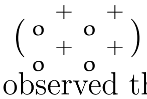
[t] Finally, we investigated human behavior under the Co-Training policy in a learning problem that *violates* Co-Training's technical conditions. The new dataset was identical to the diamond dataset except that the unlabeled items were distributed on a grid, see Figure 8.6(a). The dataset therefore violates Condition 1: items near the four corners receive conflicting labels between the two views.

[**Experiment 3**] 24 dyads worked on this counter-example under the Co-Training policy with the separable stimuli. Apart from the distribution of the unlabeled items, all aspects of the study were identical to Experiment 2. Figure 8.6(b) shows the per-item average labels in Experiment 3. Classification decisions in this study were clearly less certain than those observed in Experiment 2 (see corresponding items in Figure 8.5(c)). To compare with the `CS` row in Table 8.1, we also computed the majority vote pattern for every dyad on each of the four rectangular "clusters" in Figure 8.6(a). The proportion of dyads showing each pattern were: cross 0.00, horz 0.21, vert 0.17, diag 0.33, other 0.29. No dyad produced the cross pattern on this dataset. Thus human Co-Training outcomes depends critically upon the distribution of the unlabeled items.

8.6 Discussion

We showed that, when collaborating according to a novel policy inspired by Co-Training, two human learners behave differently than individual learners or learning pairs collaborating in an unconstrained manner. Specifically, they jointly acquire a nonlinear labeling on the diamond dataset that is highly consistent with the behavior of the machine learning algorithm, yet unusual for human category learning generally. We have also shown that the behaviors elicited by the policy depend upon the distribution of the unlabeled data.

This work employed very simple stimuli constructed to highlight the differences between Co-Training and other learning models. The question thus arises, what relevance do these results have for real-world learning tasks? We believe there are several potentially important implications. First, under the Co-Training policy each participant need view only a subset of an item's features. For problems where the number of relevant features are overwhelming, the policy may provide an efficient way of dividing the problem up so as to make best use of costly human effort. Second, in Co-Training each learner is satisfied with the final result (meaning there is little conflict between the labels given by one partner and the other), even though jointly the team arrives at a solution that would seem unlikely had they both viewed the full features. Co-Training thus provides a means of promoting agreement among team members for classification solutions that otherwise might cause disagreement. Third, the only communication required is label-exchange, which might be useful in situations where communication is costly. Fourth, each learner is "blind" to some of the feature dimensions. The policy might therefore prove useful in sensitive classification tasks where data security is an issue.

Of course, all of these applications depend upon there being real-world tasks of interest that meet the technical conditions that allow Co-Training to work. In this vein, it is worth noting that Co-Training does apply to other datasets beyond the “diamond” set used here. For example, here is a 2D dataset with 8 clusters, two of them initially labeled: . The outcome  is predicted by the Co-Training machine algorithm, and we have observed this behavior in preliminary human studies. To determine whether Co-Training has application for a real dataset, the task organizer must be able to assess whether the problem meets Co-Training’s technical conditions, and must also be able to find views of the data that exploit Co-Training’s properties. These constitute interesting problems for machine learning in their own right, and are a focus for future research.

9 DISCUSSION, FUTURE WORK AND SUMMARY

Here I collect potential real world applications, a motivating goal that remains to be addressed, some of the limitations of the work as it stands, and a few of the lessons learned regarding humans as learners viewed from a machine learning point of view. I then go on to discuss potential future work and finally summarize my contributions as presented in the preceding chapters.

Real World Applications

Over the course of these studies the "How might we apply this work to a real world setting?" has come up repeatedly. This is a common question asked of basic research. The research presented has had two motivations: 1) to better understand human learning behavior and 2) to attempt to influence this behavior. The work can be applied to real world settings in both ways as well, with the clearest applications in education.

Chapter 4 indicates that one should take care when creating a test set to evaluate a learner on a learned concept. The ordering of the test items (as well as the distribution, as shown by Zhu et al. (2007)) can change the learned concept, in an unintentional way. If, however, the evaluator wanted to purposefully change the learned concept without introducing new labels, the distribution and order could be manipulated intentionally. Though there must certainly be non-malicious motivations for such an intervention, it is useful to be able to recognize instances where evaluation data could be manipulated intentionally to confuse the learner in a predefined way.

If there is a learning task which is dependent on discriminating classes of objects, the results of Chapter 7 make it clear that care should be taken not to expose the learner previously to items whose apparent distributions might contradict useful assumptions in the supervised task. The results presented there assume that boundaries fall in low density regions, but it may be that there are other properties of the data that humans will be sensitive to, judging by the results in Chapter 5. Certainly distributions which disagree in some way with the underlying concept (e.g. trough shifted away from the boundary) should be avoided as these may interfere with the speed of learning. Additionally, as was mentioned, exposure to unlabeled items could be designed to deliberately speed later learning, a clear application in education.

Chapter 6 suggests that if there was a learning task where following an underlying manifold would be useful, such as tracking changes of an observed object over time, it is important to stress this information to the learner, and not assume that they will pick up on it on their own. This is certainly true of novel or synthetic stimuli. Another application might be in a task which contains a difficult to perceive manifold structure. Mapping this task to one with a very apparent natural manifold may make the task easier and the manifold more apparent, e.g. following the rotation of an object in 3 dimensions, This leads into future work and the question of feature selection discussed in the next subsection.

The Co-Training constraints discussed in Chapter 8 are somewhat different, and not as directly applicable to education. Here, the effects are 1) a separation of features between learners and 2) a constrained message passing scheme. Since each learner need only view a subset of the features, any task where the features are overwhelming, such as air-traffic control for instance, might be split between learners while still maintaining a complicated learned concept. Another potential application would be in areas where there is some sensitivity to the data such that no single learner should be allowed to view the entire feature set for any data-point. Features could be split between collaborators while, again, maintaining some learned concept.

This is by no means a complete list. It is the author's hope that the ideas given here are simply a springboard for educators and researchers to a larger set of potential uses.

The Two-Way Street

The motivation for this work has always been to better understand and influence human behavior using Isl models. The work presented here is done with respect to these goals. There was an additional motivation to use observations of human behavior in SSL settings to suggest areas of improvement which could be made to ML models.

This two-way street of improvement has not yet materialized, but there is no reason to believe that it cannot be done in the future. As discussed below, there are several limitations to the research that has been presented here, allowing for much continued work in the area. It is still feasible with continued investigation and, importantly, cooperation and collaboration between the Cognitive Psychology and Machine Learning communities, we will see insights from human learning which will inform SSL.

Limitations

While there are many results presented in the preceding chapters, there are limitations that should be mentioned. An important one is that in the majority of the studies mentioned, only a single synthetic stimulus type was tested per study, with very low dimensionality (1 to 2 dimensions), presented in a single modality (visual), to a very specific group of participants (undergraduates living in the Midwestern United States). While the results may be significant under these particular settings, it is important to investigate how they generalize to other settings, such as auditory stimuli, high dimensional stimuli, real world stimuli, other demographic groups, etc.

Another large limitation is the task chosen to investigate, namely binary classification. While this is a valid learning task, it is of such a basic nature that it is difficult if not impossible to directly translate the results given here to more complex tasks e.g. learning to solve algebraic equations or learning to play a musical instrument.

Investigating the same research questions under very different settings is a potentially fruitful avenue left for future work.

Human Learning Considerations

Humans are interesting learners to work with and study. Machine Learning agents are infinitely patient, do exactly as told and make no alterations or interpretations of the input data or output labels unless instructed to do so. The same is not necessarily true of humans.

We saw in Chapter 4 and Zhu et al. (2007) that humans do not necessarily perceive distances in a perceptual space as they may be intended. Depending on the stimuli used, the stimuli space may be warped such that equal distances defined in two regions of space may not be perceived as being equal. Some regions of space may be stretched while others are shrunk, leading to surprising asymmetries like those seen in 4.1.

Humans may also induce features not intended by the researcher. As an example, in the study of manifolds in Chapter 6, participants reported seeing rotated letter “T”s and “L”s in the stimuli consisting of a single vertical and single horizontal line. These representations were not intended and could have a potential effect on the results of the study. It’s for this reason that Gabor patches, like those seen in Chapter 5 are commonly used, as they are believed to correspond to visual feature detectors in a fairly straightforward way. Similarly, humans may infer changes in one feature based on changes seen in another, as in the color swatch stimuli used in Chapter 8. A tremendous amount care must be made when selecting stimuli for human experiments to avoid any unintentional complications.

Another consideration with regard to stimuli is in feature selection. It is often assumed that the learner is given a set of features, all of which are important. This of course is not an assumption a human learner should feel safe making in the real world. It is still not entirely clear how humans do feature selection; similarly how they perform feature integration, creating new features through combinations of existing features. In fact, some of the SSL assumptions discussed, such as the smoothness assumption made use of in the manifold learning case, can simply be seen as an additional feature, picked out from the many available. In this case the additional feature is the identities of the neighbors of any particular item.

A large consideration when choosing candidates to model human performance is that humans primarily appear to learn online, that is they consider stimuli sequentially, rather than in batch, considering all items at once. Even when presented with a batch of data, humans by necessity must attend to and perceive a single stimuli at a time. Machine learners are capable of considering items either online or batch. Models which only learn in batch mode may not be likely candidates for human learning unless there is some way of modifying them to do or approximate online learning.

Finally, machine learners are infinitely patient in that they will consider any number of stimuli, so long as memory and computational constraints are not exceeded. The machine learner will also perceive all available features for each stimuli and perfectly remember them indefinitely unless asked to do otherwise. Human learners will not necessarily cooperate quite so willingly or exactly. A human learner has a limited attention span, usually require some motivation to attend to a task (beyond being told they should), and will, in the vast majority of individuals, not be able to either perceive or remember all available features without extensive training. For instance, when presenting the learner with prior unlabeled items in Chapter 7, a separate task had to be designed to improve confidence that the human learners were actually attending to, and encoding, the unlabeled examples.

While none of these issues are impossible to overcome, special consideration needs to be made when designing human experiments to account for them. Many of the issues, like the stretching of perceptual space or feature selection, could benefit from additional research and may be opportunities for the influence of CP on ML.

9.1 Future Work

As has been mentioned before, there is a tremendous amount work left to be done in the investigation of how humans make use of combinations of labeled and unlabeled data when learning. Some of this work regards larger, overall questions, such as how humans do feature selection, how human perceptual space can be warped for different stimuli, and what the differences truly are between human perception of separable vs. integral feature dimensions. Other work involves addressing the limitations of our experiments discussed above, in particular the "single dataset" issue, where it is necessary to see if the effects seen generalize to other stimuli, other modalities, etc. Still other future work is specific to each of the experiments described.

Chapter 4: Order Effects

In this study only the three models discussed in Chapter 2 were compared to human behavior. It remains to be seen if other models making use of more advanced SSL assumptions, such as manifolds or large margin separation can be formulated to be susceptible to the same sort of effects and then compared with human behavior.

Chapter 5: What Parameters are Learned?

It may be that in other tasks, where discrimination between hypothesized models, or models not in the GMM family, is still possible, the result found (that all parameters were learned from unlabeled data) may not be the case. Additional investigation is required to confirm that our conclusion generalizes to other situations.

Chapter 6: Manifold Learning

This experiment suffered the most from the issues related to human perception of stimuli. It would be particularly useful to confirm the results seen with a more accepted set of stimuli, such as Gabor patches. Additionally, in this experiment, as in others, there was no consideration given to which model participants would prefer before seeing the data. It is possible that humans favor models which produce axis-parallel decision boundaries. Defining and incorporating non-uniform priors over the models is a topic for future research.

Chapter 7: Prior Unlabeled Data

This experiment was one for which the real-world application was the most apparent: exposing a learner to unlabeled data prior to the supervised task. What is not clear is what sort of exposure would be adequate to influence human learning. Would simply being exposed visually to unlabeled data (e.g. representative items displayed on wallpaper in a room a student is playing) be enough to illicit improved learning performance? Or would the learner need more engaged interaction (e.g. playing with material representations of the stimuli, like plastic toys) to see these effects? Additional studies looking at how these ideas might actually be used in teaching students is an obvious and enticing line of investigation.

Chapter 8: Co-Training Constraints

To determine whether Co-Training has application for a real task, the supervisor of the task must be able to assess whether the problem meets Co-Training's technical conditions, and must also be able to find views of the data that exploit Co-Training's properties. These constitute interesting problems for machine learning in their own right, and are a focus for future research.

9.2 Key Contributions

- Showed that existing SSL models can be modified to reproduce the Test-Item Effect observed in humans, where the learned boundary can be affected by the order of test items presented to the learner.
- Showed that humans, when performing a 1D 2-class categorization task, are sensitive to all parameters of the underlying distributions and do not constrain their search of the parameter space.
- Showed that humans can learn using manifolds, given sufficient labeled data and hints regarding the manifold structure.
- Showed that the speed of human learning on a supervised task can be affected by prior unlabeled experience.
- Showed that, using a variation of the classic Co-Training constraints, we can elicit behavior from human collaborators that is not observed without these constraints.

9.3 Conclusion

It is clear that human learners are sensitive to both labeled and unlabeled data when performing a classification task. The work presented here was an effort to both better understand this behavior and to attempt to influence learning. The models described in Chapter 2 provide a strong theoretical link between ML and CP. The results of the studies in the following chapters go on to show empirically that ML models and their associated SSL assumptions can be applied to human learners. While humans remain a black box, and none of the studies described here prove definitively that the models applied in fact match the mechanics of human learning, they do give a strong indication of how humans learn in the semi-supervised classification setting.

REFERENCES

Anderson, John R. 1990. *The adaptive character of thought.* Hillsdale, NJ: Erlbaum.

———. 1991. The adaptive nature of human categorization. *Psychological Review* 98(3):409–429.

Ashby, F. G., and W. T. Maddox. 1990. Integrating information from separable psychological dimensions. *Journal of Experimental Psychology: Human Perception and Performance* 16(3):598–612.

Ashby, F. Gregory, and Leola A. Alfonso-Reese. 1995. Categorization as probability density estimation. *Journal of Mathematical Psychology* 39: 216–233.

Balcan, Maria-Florina, and Avrim Blum. 2010. A discriminative model for semi-supervised learning. *Journal of the ACM* 57(3).

Belkin, Mikhail, Partha Niyogi, and Vikas Sindhwani. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research* 7:2399–2434.

Bengio, Y., J. Louradour, R. Collobert, and J. Weston. 2009. Curriculum learning. In *The 26th international conference on machine learning*, ed. L. Bottou and M. Littman, 41–48. Omnipress.

Bishop, Christopher M. 2007. *Pattern recognition and machine learning.* Springer.

Blum, Avrim, and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *COLT*.

Brefeld, Ulf, Thomas Gaertner, Tobias Scheffer, and Stefan Wrobel. 2006. Efficient co-regularized least squares regression. In *ICML*. Pittsburgh, USA.

Chapelle, Olivier, Alexander Zien, and Bernhard Schölkopf, eds. 2006. *Semi-supervised learning.* MIT Press.

Dempster, A. P., N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 39(1):1–38.

Fearnhead, P. 2004. Particle filters for mixture models with an unknown number of components. *Statistics and Computing* 14:11–21.

Fried, L. S, and K. J Holyoak. 1984. Induction of category distributions: A framework for classification learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 10(2):234–257.

Gibson, Bryan R., Timothy T. Rogers, Charles W. Kalish, and Xiaojin Zhu. 2015. What causes category-shifting in human semi-supervised learning? In *Proceedings of the 37th annual conference of the cognitive science society (CogSci).*

Gibson, Bryan R., Timothy T. Rogers, and Xiaojin Zhu. 2013. Human semi-supervised learning. *Topics in Cognitive Science* 5:132–172.

Gibson, Bryan R., Xiaojin Zhu, Timothy T. Rogers, Charles W. Kalish, and Joseph Harrison. 2010. Humans learn using manifolds, reluctantly. In *Advances in neural information processing systems (NIPS)*, vol. 24.

Griffiths, Thomas L., Kevin R. Canini, Adam N. Sanborn, and Daniel J. Navarro. 2007. Unifying rational models of categorization via the hierarchical dirichlet process. In *Proceedings of the 29th annual conference of the cognitive science society*, 323–328.

Griffiths, Thomas L., Adam N. Sanborn, Kevin R. Canini, Daniel J. Navarro, and Joshua B. Tenenbaum. 2011. Nonparametric bayesian models of categorization. In *Formal approaches in categorization*, ed. Emmanuel M. Pothos and Andy J. Wills, 173–198. Oxford University Press.

Hintzman, D. L. 1986. "schema abstraction" in a multiple-trace memory model. *Psychological Review* 93(4):411–428.

Hsu, Anne Showen, and Thomas E Griffiths. 2010. Effects of generative and discriminative learning on use of category variability. In *32nd annual conference of the cognitive science society.*

Johnson, Rie, and Tong Zhang. 2007. Two-view feature generation model for semi-supervised learning. In *ICML*.

Kalish, Charles W., Timothy T. Rogers, Jonathan Lang, and Xiaojin Zhu. 2011. Can semi-supervised learning explain incorrect beliefs about categories? *Cognition* 120(1):106–118.

Kalish, Charles W., XiaoJin Zhu, and Timothy T. Rogers. 2014. Drift in children's categories: when experienced distributions conflict with prior learning. *Developmental Science.*

Kalish, C.W., S. Kim, and A.M. Young. 2012. How young children learn from examples: Descriptive and inferential problems. *Cognitive Science* 36: 1427–1448.

Khan, Faisal, Xiaojin Zhu, and Bilge Mutlu. 2011. How do humans teach: On curriculum learning and teaching dimension. In *Advances in neural information processing systems (nips)*, vol. 25.

Lake, Brenden M., and James L. McClelland. 2011. Estimating the strength of unlabeled information during semi-supervised learning. In *Proceedings of the 33rd annual conference of the cognitive science society.*

Lockhead, G. R. 1966. Effects of dimensional redundancy on visual discrimination. *Journal of Experimental Psychology: Human Perception and Performance* 3:436–443.

Love, B. C. 2002. Comparing supervised and unsupervised category learning. *Psychonomic Bulletin and Review* 9(4):829–835.

Mansinghka, Vikash K., Daniel M. Roy, Ryan Rifkin, and Josh Tenenbaum. 2007. AClass: an online algorithm for generative classification. In *Proceedings of the 11th international conference on artificial intelligence and statistics (AISTATS)*.

Medin, D.L., and M.M. Schaffer. 1978. Context theory of classification learning. *Psychological Review; Psychological Review* 85(3):207.

Minda, J. P., and J. D. Smith. 2011. Prototype models of categorization: Basic formulation, predictions, and limitations. In *Formal approaches in categorization*, ed. E. M. Pothos and A. J. Wills, 40–64. Cambridge, UK: Cambridge University Press.

Myers, J.L. 1976. Probability learning and sequence learning. In *Handbook of learning and cognitive processes: Approaches to human learning and motivation*, ed. W.K. Estes, 171–205. Hillsdale, NJ: Erlbaum.

Nadaraya, E. A. 1964. On estimating regression. *Theory of Probability and Its Application* 9:141–142.

Neal, Radford M. 1998. Markov chain sampling methods for dirichlet process mixture models. Tech. Rep. 9815, Department of Statistics, University of Toronto.

Nigam, Kamal, and Rayid Ghani. 2000. Analyzing the effectiveness and applicability of co-training. In *CIKM*.

Nosofsky, R. M., and T. J. Palmeri. 1996. Learning to classify integral-dimension stimuli. *Psychonomic Bulletin and Review* 3(2):222–226.

Nosofsky, Robert M. 1985. Overall similarity and the identification of separable-dimension stimuli: A choice model analysis. *Perception and Psychophysics* 38(5):415–432.

———. 1986. Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General* 115(1):39–57.

———. 1991. The relation between the rational model and the context model of categorization. *Psychological Science* 2(6):416–421.

———. 2011. The generalized context model: an exemplar model of classification. In *Formal approaches in categorization*, ed. E. M. Pothos and A. J. Wills, 18–39. Cambridge, UK: Cambridge University Press.

Palmeri, T. J., and M. A. Flanery. 1999. Learning about categories in the absence of training: Profound amnesia and the relationship between perceptual categorization and recognition memory. *Psychological Science* 10:526–530.

Pothos, Emmanuel M., and Andy J. Wills, eds. 2011. *Formal approaches in categorization*. Oxford University Press.

Rasmussen, Carl E., and Christopher K. I. Williams. 2006. *Gaussian processes for machine learning*. MIT Press.

———. 2007. GPML matlab code. `http://www.gaussianprocess.org/gpml/code/matlab/doc/`, accessed May, 2010.

Rips, L. J. 1989. Similarity, typicality, and categorization. In *Similarity and analogical reasoning*, ed. S. Vosniadou and A. Ortony, 21–59. New York, NY: Cambridge University Press.

Rogers, Timothy T., Charles W. Kalish, Bryan R. Gibson, Joseph Harrison, and Xiaojin Zhu. 2010. Semi-supervised learning is observed in a speeded but not an unspeeded 2D categorization task. In *Proceedings of the 32nd annual conference of the cognitive science society (CogSci).*

Rosch, E., C. B. Mervis, Gray, W. D., D. M. Johnson, and P. Boyes-Braem. 1976. Basic objects in natural categories. *Cognitive psychology* 8(3):382–439.

Sanborn, Adam N., Thomas L. Griffiths, and Daniel J. Navarro. 2006. A more rational model of categorization. In *Proceedings of the 28th annual conference of the cognitive science society (CogSci)*, 726–731.

Shepard, R. N. 1964. Attention and the metric structure of the stimulus space. *Journal of Mathematical Psychology* 1:54–87.

———. 1986. Discrimination and generalization in identification and classification: Comment on nosofsky. *Journal of Experimental Psychology: General* 115:58–61.

———. 1991. Integrality versus separability of stimulus dimensions: From an early convergence of evidence to a proposed theoretical basis. In *The perception of structure: Essays in honor of Wendell R. Garner*, ed. G. R. Lockhead and J. R. Pomerantz, 53–71. American Psychological Association.

Shi, L., N. H. Feldman, and T. L. Griffiths. 2008. Performing bayesian inference with exemplar models. In *Proceedings of the 30th annual conference of the cognitive science society (CogSci)*, 745–750.

Sindhwani, Vikas, Partha Niyogi, and Mikhail Belkin. 2005. Beyond the point cloud: from transductive to semi-supervised learning. In *ICML05, 22nd international conference on machine learning.*

Smith, E.E., and S.A. Sloman. 1994. Similarity-versus rule-based categorization. *Memory & Cognition* 22(4):377–86.

Teh, Yee W. 2010. Dirichlet processes. In *Encyclopedia of machine learning.* Springer.

Tenenbaum, J. B., T. L. Griffiths, and Kemp. C. 2006. Theory-based Bayesian models of inductive learning and reasoning. *Trends in Cognitive Science* 10(7):309–318.

Vandist, K., M. De Schryver, and Y. Rosseel. 2009. Semisupervised category learning: The impact of feedback in learning the information-integration task. *Attention, Perception, & Psychophysics* 71(2):328–341.

Vanpaemel, W., G. Storms, and B. Ons. 2005. A varying abstraction model for categorization. In *Proceedings of the 27th annual conference of the cognitive science society (CogSci).*

Vulkan, N. 2000. An economist's perspective on probability matching. *Journal of Economic Surveys* 14:101–118.

Wallis, Guy, and Heinrich H. Bülthoff. 2001. Effects of temporal association on recognition memory. *Proceedings of the National Academy of Sciences* 98(8):4800–4804.

Wasserman, Larry. 2006. *All of nonparametric statistics.* New York, NY, USA: Springer.

Zaki, S. R., and Robert. M. Nosofsky. 2007. A high-distortion enhancement effect in the prototype-learning paradigm: Dramatic effects of category learning during test. *Memory & Cognition* 35:2088–2096.

Zhou, Dengyong, Olivier Bousquet, Thomas Lal, Jason Weston, and Bernhard Schölkopf. 2004. Learning with local and global consistency. In *Advances in neural information processing system 16.*

Zhou, Zhi-Hua, and Ming Li. 2005. Tri-training: exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering* 17(11):1529–1541.

Zhu, X., T. Rogers, R. Qian, and C. Kalish. 2007. Humans perform semi-supervised classification too. In *Proceedings of the 21st conference on artificial intelligence (AAAI)*.

Zhu, Xiaojin. 2013. Machine teaching for bayesian learners in the exponential family. In *Advances in neural information processing systems (NIPS)*.

Zhu, Xiaojin, Zoubin Ghahramani, and John Lafferty. 2003. Semi-supervised learning using Gaussian fields and harmonic functions. In *The 20th international conference on machine learning (icml)*.

Zhu, Xiaojin, Bryan R. Gibson, Kwang-Sung Jun, Timothy T. Rogers, Joseph Harrison, and Chuck Kalish. 2010. Cognitive models of test-item effects in human category learning. In *The 27th international conference on machine learning (ICML)*.

Zhu, Xiaojin, Bryan R. Gibson, and Timothy T. Rogers. 2011. Co-Training as a human collaboration policy. In *The 25th conference on artificial intelligence (AAAI)*.

Zhu, Xiaojin, and Andrew B Goldberg. 2009. *Introduction to semi-supervised learning*. Morgan & Claypool.