Superspace extrapolation reveals inductive biases in function learning

Christopher G. Lucas
cglucas@cmu.edu
Department of Psychology
Carnegie Mellon University

Douglas Sterling
douglas.sterling@bccn-berlin.de
Bernstein Center for Computational
Neuroscience, Berlin

Charles Kemp
ckemp@cmu.edu
Department of Psychology
Carnegie Mellon University

Abstract

We introduce a new approach for exploring how humans learn and represent functional relationships based on limited observations. We focus on a problem called *superspace extrapolation*, where learners observe training examples drawn from an n-dimensional space and must extrapolate to an $n+1$-dimensional superspace of the training examples. Many existing psychological models predict that superspace extrapolation should be fundamentally underdetermined, but we show that humans are able to extrapolate both linear and non-linear functions under these conditions. We also show that a Bayesian model can account for our results given a hypothesis space that includes families of simple functional relationships.

Introduction

People regularly face situations where they must reason about functions defined over continuous variables. For example, consider a truck driver who wants to predict how quickly his truck can accelerate based on the mass of his cargo. If the driver has transported similar masses in the past, he can generate an accurate prediction by recalling the accelerations observed on these previous occasions. The real test of whether and how he has learned the function is how he extrapolates from past examples and makes predictions about loads that are much lighter or heavier than those he has seen in the past. Figure 1a, for example, shows how a learner might use linear extrapolation to generalize on the basis of two examples.

In any function learning setting, extrapolation judgments are shaped by the examples observed and the assumptions or *inductive bias* that the human brings to the problem. Minimizing the information carried by the training examples makes the role of the inductive bias especially apparent. Here we explore how humans learn functions from impoverished training data, and focus in particular on the problem of *superspace extrapolation*. Given training examples that fall within an n-dimensional space, we study how learners extrapolate to an $n+1$-dimensional superspace that encloses the training examples. If the underlying function is one-dimensional, superspace extrapolation requires the learner to generalize on the basis of a single training example (Figure 1b). We focus on the corresponding problem in two dimensions, where the learner observes training examples drawn from a one-dimensional space and must generalize to the full two-dimensional space (Figures 1c-f).

Superspace extrapolation is an interesting problem in its own right, but also provides a way to distinguish between competing accounts of function learning. The psychological literature on function learning includes two prominent approaches that we will call the rule-based approach and the similarity-based approach. The rule-based approach proposes that humans rely on a set of parametric functions that have explicit mental representations, including linear functions, polynomial functions, and others (Carroll, 1963; Brehmer, 1974; Koh & Meyer, 1991; Koh, 1993; Bott & Heit, 2004). The similarity-based approach proposes that humans remember specific examples encountered during training, and make predictions about test points based on similarity to the training points (Busemeyer, Myung, & McDaniel, 1993; Kelley & Busemeyer, 2008). Similarity-based approaches have traditionally struggled to account for extrapolation, and superspace extrapolation is especially challenging for these approaches. We show that humans are able to learn several different functions in a superspace extrapolation paradigm, which supports the idea that people can formulate and use explicit representations of both linear and nonlinear functions.

The hybrid approach to function learning proposes that humans can make both rule-based and similarity-based inferences. We show that this approach can account for our data by evaluating a hybrid model that builds on the Gaussian process account of Griffiths, Lucas, Williams, and Kalish (2009). Other models of function learning are prominent in the literature, and here we mention two representative examples. The Population of Linear Experts (POLE) model (Kalish, Lewandowsky, & Kruschke, 2004) proposes that humans learn functions that are piecewise linear (in the 1D case) or piecewise planar (in the 2D case). Since the training examples in a superspace extrapolation task are collinear, any given piecewise planar function belongs to an infinite family of piecewise planar functions that make very different extrapolation predictions but fit the training examples equally well. For example, Figures 1c and 1d show two different extrapolation functions that account perfectly for the same set of training examples. As a result, models that rely exclusively on linear extrapolation suggest that superspace extrapolation problems are fundamentally underdetermined and are unlikely to lead to consistent patterns of human responses. The Sigma model (Juslin, Karlsson, & Olsson, 2008) is an alternative approach which proposes that humans can acquire explicit representations of linear functions, but that knowledge about non-linear functions is “carried implicitly by memory for exemplars.” We show that people are successfully able to extrapolate non-linear functions in a superspace extrapolation paradigm, which suggests that the rule-based component of a hybrid approach should include room for non-linear functions.

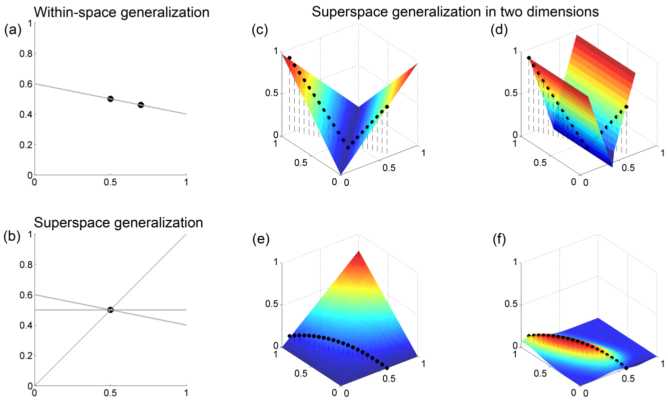

Figure 1: Examples of superspace extrapolation in one and two dimensions. Black dots are training points. (a) Standard function learning with one cue dimension; (b) Extrapolating from cues in a zero-dimensional subspace; (c) Superspace extrapolation in two dimensions, where $f(x,y) = |x-y|$; (d) A second example of superspace extrapolation applied to $|x-y|$, assuming a difference piecewise linear function; (e) Superspace extrapolation where $f(x,y) = xy$; (f) A second example using xy, with similarity-based extrapolation.

Experiment

We developed a behavioral experiment with two goals in mind. The first and most basic goal is to find out whether superspace extrapolation is possible at all. Expecting participants to make generalizations about a function given a single training point seems unreasonable (Figure 1b), and it is possible that participants will find the two dimensional version of superspace extrapolation equally underdetermined. If superspace extrapolation turns out to be possible, our second goal is to understand how this kind of extrapolation is achieved. In particular, we aimed for a task that could address whether participants use explicit rules to make inferences that go beyond linear and similarity-based extrapolation.

We hypothesized that participants could learn a range of two-dimensional functions and chose to focus on five specific functions that are relatively simple and qualitatively different from one another. These functions are shown in Table 1 and plotted in Figure 2. Note that the family of functions includes both linear and non-linear functions. Consider one such function, the absolute difference function plotted in Figure 1c. Suppose that a learner observes the training points shown in black, which happen to fall along a line. There are many possible ways to extrapolate from the training points to the entire space—for example, Figure 1d shows an extrapolation to an axis-aligned function that is especially simple in the sense that it is invariant with respect to one of the dimensions. If people extrapolate by fitting a piecewise linear (i.e. planar) function to the training points, then there seems to be no reason to prefer the extrapolation in Figure 1c to Figure 1d or the infinitely many alternative extrapolations that fit two planes to the training points. On the other hand, if $|x-y|$ is in human learners' representational toolkit, we might predict that their extrapolations would resemble Figure 1c.

If extrapolation in cases like Figure 1c does depend on explicit rules, then extrapolations might vary dramatically if the positions of the training points are rotated. For example, the function $f(x,y) = |x-y|$ turns into the function $|x-1+y|$ when rotated by $\pi/2$ around the line $(0.5, 0.5, t)$ which passes through $(0.5, 0.5)$ in the xy-plane and is perpendicular to the z-axis. It seems plausible that participants rely on a hypothesis space of rules that can accommodate the original but not the rotated function. We therefore compare each extrapolation problem to a rotated variant where the training points are rotated around the line $(0.5, 0.5, t)$, and predict that participants will be able to learn the unrotated but not the rotated version of each function. Linear extrapolation is equally possible in the rotated and unrotated cases, and a linear extrapolation account therefore predicts no qualitative difference between these two versions of the problem. Many similarity-based approaches also predict that the rotated and unrotated versions should lead to similar results, since similarity metrics (e.g. Euclidean distance) are often rotation-invariant.

Methods

Participants. 33 participants were recruited from Carnegie Mellon's participant pool and the local community and received course credit or ten dollars for participating.

Materials. Cues were presented using adjacent horizontal bars and participants made predictions by adjusting a third horizontal bar centered under the midpoint between the cue bars. Each bar had a bounding box, so the range of valid values—which we denote with $[0, 1]$ for simplicity—was evident to participants. No numerical information was provided about any of the variables. Feedback presentations took the form of a green bar overlaid on the prediction bar.

Procedure. Participants were told that they would have to learn several cause-effect relationships through trial-and-error. Each participant was presented with the five distinct functions listed in Table 1 in random order, in either *rotated* or *unrotated* form. For a given unrotated function $f(x,y)$ and a rotation angle θ_r, we define a rotated function $g(x,y) = f(x',y')$ where (x',y') is the result of rotating (x,y) around the point $(0.5, 0.5)$ by θ_r. Table 1 contains explicit definitions of the unrotated and rotated versions of all functions. For each function, participants saw a training phase followed by a test phase. Both phases consisted of a series of trials in which participants were presented with cues (x,y) and asked to predict $f(x,y)$.

The training phase included 40 randomly-ordered examples that fell along a single line. Specifically, training examples fell at equal intervals along a line segment with length 0.9 centered at $(0.5, 0.5)$, making an angle of θ_l (see Table 1) with the x-axis. After each training prediction, participants who gave guesses within 0.04 of the true value moved to the next example point, while inaccurate guesses were followed by feedback in which the correct value of $f(x,y)$ was presented and participants had to adjust their prediction to match

Name	Unrot. $f(.)$	θ_r	Rot. $f(.)$	θ_l
Projection	x	$\frac{1}{2}\pi$	$1-y$	$\frac{1}{8}\pi$
Average	$\frac{1}{2}(x+y)$	$-\frac{1}{2}\pi$	$\frac{1}{2}(x+1-y)$	$\frac{3}{8}\pi$
Product	xy	$\frac{1}{2}\pi$	$x(1-y)$	$\frac{5}{8}\pi$
Difference	$\lvert x-y\rvert$	$\frac{1}{2}\pi$	$\lvert x+y-1\rvert$	$\frac{5}{8}\pi$
Max	$\max(x,y)$	$-\frac{1}{2}\pi$	$\max(x,1-y)$	$\frac{7}{8}\pi$

Table 1: List of functions that participants learned. θ_r refers to the relative angles of the original and rotated functions, and θ_l denotes the angle of the original line.

that value in order to continue.

In the subsequent test phase, participants received no feedback and were presented with 10 equidistant points along the original training line, 10 within-space points that fell beyond the extrema of the original line, and 36 superspace extrapolation points in a uniform 6-by-6 grid over the $[0,1] \times [0,1]$ range. After each test phase, participants were prompted to describe what they thought the function was before moving on to the next function. The bars corresponding to variables x and y were selected randomly.

Experimental Results

We excluded one participant who did not attempt to learn the functions, indicated by a mean absolute error exceeding 0.25. Ten of the 32 participants who remained did not complete all of the functions in the allotted hour, but each version (rotated or unrotated) of each function was completed by at least 11 participants. The side on which x and y were presented had no significant influence on performance, so the two orientations were grouped together.

The five panels labeled (iii) in Figure 2 show average human responses for the five unrotated functions. The black dots show responses for the training points, and the surfaces show responses for the extrapolation points. In all cases, participants were able to learn the function values for the training points, and their extrapolation judgments were qualitatively similar in all cases to the true functions. Note that superspace extrapolation was possible even for the three nonlinear functions in the set. The panels labeled (iv) in Figure 2 show average responses for the rotated functions. Participants appeared to learn the rotated projection function, but extrapolation judgments for the four other rotated functions appear qualitatively different from the true functions. Table 1 shows that the rotated version of the projection function is $f(x,y) = 1-y$. Recall that the cues were presented using sliders on horizontal bars, and that the value of each cue corresponds visually to the proportion of the bar to the left of the slider. The rotated projection function can be learned by paying attention to the complement of the y cue, or the proportion of the y-bar to the right of the slider. Although responses for the rotated projection function suggested that participants are sensitive to complements in some cases, responses for the remaining rotated functions suggest that participants find it difficult to learn simple functions defined in terms of complements.

Function	MAE$_u$	MAE$_r$	p
x	0.039	0.055	0.51
$(x+y)/2$	0.071	0.151	0.00088
xy	0.092	0.140	0.042
$\lvert x-y\rvert$	0.123	0.247	0.0015
$\max(x,y)$	0.046	0.130	0.0077

Table 2: Mean absolute error for test points in learning rotated MAE$_r$ and unrotated functions (MAE$_u$). p-values were obtained using a two-tailed permutation test, using 200,000 samples per test.

The descriptions provided by individual participants indicated that many had acquired explicit representations of the unrotated functions. Five examples of these descriptions are: “effect was identical to cause B” (projection); “roughly the average of the two causes” (average); “fraction multiplication” (product); “difference of the causes” (difference); “the larger of the two values” (max). Responses for the rotated functions sometimes indicated complex hypotheses, but more often indicated confusion or uncertainty about the nature of the function. These descriptions indicate that some individuals had clearly learned the functions. To further explore responses at the individual level, we looked at the extent to which individual participants’ judgments fit the true functions versus several alternatives, for both unrotated and rotated functions, shown in Figure 3. The space of candidate functions included all true unrotated and rotated functions, along with a set of simple alternatives shown in the caption to Figure 3. The alternatives include a function that captures complete uncertainty ($f(x,y) = 0.5$), floor and ceiling responses ($f(x,y) = 1$ and $f(x,y) = 0$), and some simple linear combinations of x and y. For all of the unrotated functions, most participants’ extrapolation judgments were best fit by the true function. For the rotated versions, the modal judgment only matched the true function for rotations of $f(x,y) = x$ and $f(x,y) = \max(x,y)$.

Additional evidence that individual participants often learned the true functions relatively well is provided by examining the mean absolute error with respect to the true function. Performance for the projection function was near-ceiling for both unrotated and rotated versions, but in all other cases participants had lower mean absolute error for the unrotated functions than the rotated functions. Table 2 shows mean absolute error for the extrapolation points, and a similar pattern held for the training points, with significantly better performance in the unrotated cases for $f(x,y) \in \{x, |x-y|, xy, \max(x,y)\}$ at $\alpha = 0.05$.

Previous experiments have explored the relative learnability of one-dimensional functions, and our results provide some initial evidence about a learnability ordering for two dimensional functions. The results in Table 2 suggest that the

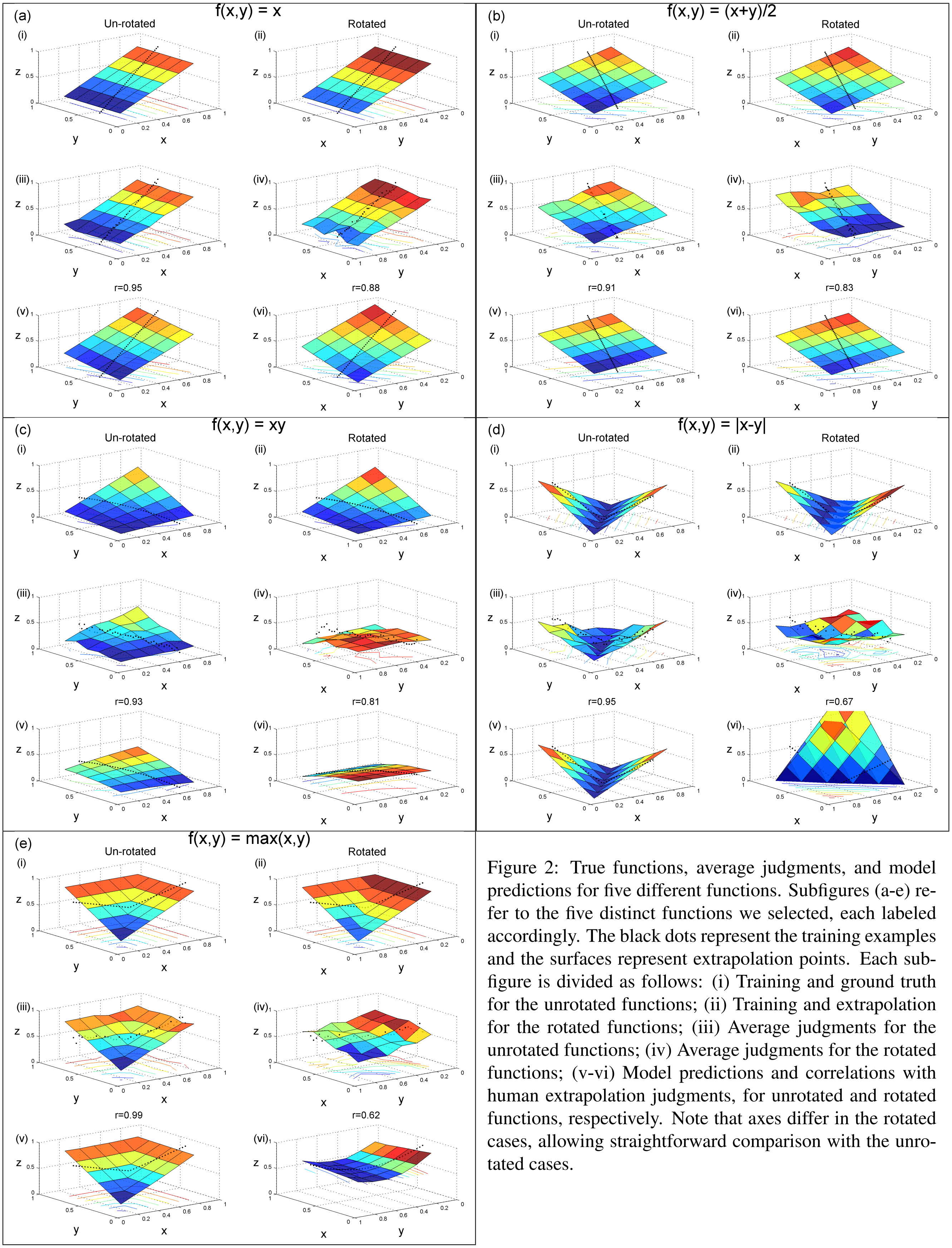

Figure 2: True functions, average judgments, and model predictions for five different functions. Subfigures (a-e) refer to the five distinct functions we selected, each labeled accordingly. The black dots represent the training examples and the surfaces represent extrapolation points. Each subfigure is divided as follows: (i) Training and ground truth for the unrotated functions; (ii) Training and extrapolation for the rotated functions; (iii) Average judgments for the unrotated functions; (iv) Average judgments for the rotated functions; (v-vi) Model predictions and correlations with human extrapolation judgments, for unrotated and rotated functions, respectively. Note that axes differ in the rotated cases, allowing straightforward comparison with the unrotated cases.

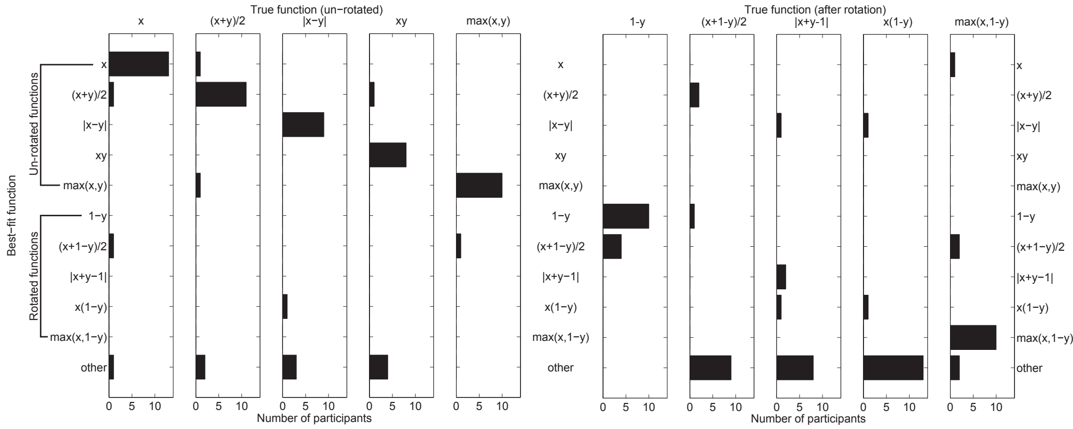

Figure 3: Trained functions versus functions that best fit participants' judgments. The *other* group includes $f(x, y) \in \{1, 0.5, 0, 0.5x, 0.5y, 0.75x + 0.25y, 0.25x + 0.75y\}$. Best-fitting functions were those that minimized mean squared error.

projection and maximum functions are easiest to learn, and that the product and difference functions are most difficult to learn.

Taken together, our data provide strong evidence that humans are capable of superspace extrapolation, and can learn both linear and non-linear functions under this paradigm. The lower performance for the rotated functions suggests that the space of learnable functions is restricted. Both results are compatible with the idea that people can acquire explicit representations of rules, but raise challenges for approaches that focus on similarity-based computations alone. Our data, however, are compatible with a hybrid approach, and the next section describes one such approach that accounts for our data relatively well.

Modeling superspace extrapolation

The hybrid approach to function learning is motivated by the idea that humans readily learn certain rules but fall back on similarity-based computations when no simple rule is consistent with the observed examples. The rule-based component of this approach can potentially explain how humans carry out superspace extrapolation when learning the unrotated functions in our experiment, and the similarity-based component may help to explain responses for the rotated functions.

To demonstrate that the hybrid approach can account for our data, we developed a computational model which assumes that humans make use of a hypothesis space that contains several families of functions. Some of these function families correspond to simple rules, and others are more generic and include all smooth functions. The first column of Table 3 shows one such hypothesis space that includes linear, quadratic, difference, maximum and product functions, along with one generic family of smooth functions. Given this hypothesis space and a set of training examples, extrapolation judgments can be made by using the posterior distribution over the space of functions.

The model we implemented builds on the hybrid approach of Griffiths et al. (2009), which uses Gaussian processes to capture both rule-based and similarity-based function learning. As originally presented, this model takes kernel functions that express linear and quadratic rules as well as a standard similarity-based kernel function for which the covariance between any two points $\mathbf{x}$ and $\mathbf{x}'$ is $K(\mathbf{x}, \mathbf{x}') = \theta_1 \exp(-\frac{1}{\theta_2^2}||\mathbf{x} - \mathbf{x}'||^2)$, where θ_1 and θ_2 determine the smoothness of the function. Intuitively, this last kernel expresses the assumption that functions are locally smooth, and was used to produce the extrapolations in Figure 1f. The model generates predictions by integrating over all possible functions for all function types, integrating out all applicable parameters. For a more detailed description, see Griffiths et al. (2009).

Function family	Prior	Mean function
$\boldsymbol{\beta}[1\ x\ y]^T$	0.5	$\mu_0 = 0, \mu_1 = \mu_2 = 1$
$\boldsymbol{\beta}[1\ x\ y]^T$	0.4	$\mu_0 = 1, \mu_1 = \mu_2 = -1$
$\boldsymbol{\beta}[1\ x\ y\ x^2\ y^2]^T$	0.09	$\mu_0 = \mu_1 = \mu_2 = \mu_3 = 0$
$\beta_1 \lvert x - y \rvert$	0.001	$\mu_1 = 1$
$\beta_1 \max(x, y)$	0.001	$\mu_1 = 1$
$\beta_1 xy$	0.001	$\mu_1 = 1$
Smooth functions	0.01	$f(x, y) = 0$

Table 3: Hypothesis space captured by the Gaussian process model. Un-normalized prior probabilities are given for each function family for readability. For the first five families, coefficients β_i are distributed normally around μ_i with a common variance for each coefficient. The difference, maximum, and product families are not described by Griffiths et al. (2009), but the prior probabilities on all remaining families and the μ_i values for these families are drawn from Griffiths et al. (2009).

Our extension to the original model was to add a kernel capturing each of the three non-linear rules in our experiment, which are equivalent to Bayesian regression models of the form $\beta|x - y|$, $\beta\max(x, y)$, and βxy, where β is a coefficient distributed normally around one. We assigned each new kernel a prior probability of 0.001, or one tenth that of the leastprobable kernel in the original model, before renormalizing kernel probabilities. See Table 3 for a summary of all of the kernels in the model and their corresponding probabilities.

Model predictions are shown in Figures 2e and f, along with correlations with human extrapolation judgments. In most cases the predictions of this model closely matched participants' superspace extrapolations for both unrotated and rotated functions. The latter result is the more striking of the two, as the predictions arise from averaging over several kernels rather than choosing suitable ones in advance.

The one major discrepancy between model predictions and human judgments occurs for the rotated version of $|x - y|$, where the extrapolation judgments predicted by the model are substantially more extreme than the human responses. This result is driven by the fact that the family of difference functions in Table 3 can perfectly account for the rotated training points if β_1 takes a value larger than 1. Unlike the model, humans may be unable to learn weighted versions of the difference function in Table 3, which could be captured by setting the coefficient β_1 for this family to 1. The model represents the simplest possible extension of the Gaussian process account of Griffiths et al. (2009), but adjusting the priors on the coefficients may result in a more accurate model of human learning.

Alternative models

Several recent models that address extrapolation in function learning and multiple cue judgment, including POLE (Kalish et al., 2004), EXAM (DeLosh, Busemeyer, & McDaniel, 1997), and Sigma (Juslin et al., 2008), all suggest that humans extrapolate according to linear functions. In their present forms, none of these models appear to account for our results. We fit the POLE model to our data and found that extrapolations were consistently piecewise linear in one cue dimension, and invariant to the other, taking a form like that in Figure 1c. This approach to superspace extrapolation seems plausible *a priori* but it does not reflect the behavior of our participants. The EXAM model makes extrapolation predictions using the nearest past examples to a new point, implying that peoples' judgments are invariant to the rotation of a given function, which is inconsistent with our data. Finally, the Sigma model (Juslin et al., 2008) proposes that humans can acquire explicit representations of linear functions but that extrapolation of non-linear functions relies on similarity-based generalization. The Sigma model is therefore inconsistent with our finding that people were able to learn several non-linear functions.

Conclusion

We introduced the problem of superspace extrapolation, which provides a new way to explore the inductive biases that people bring to the task of function learning. Our data suggest that these inductive biases include a toolkit of linear and non-linear rules that can be compared against the available data. Our results challenge several popular accounts of function learning, but we showed that they are compatible with a hybrid approach to function learning that accommodates both explicit rules and similarity-based inferences.

Superspace extrapolation requires learners to go beyond the available data in a fundamental way, and other problems where humans make inferences based on limited data have also provided important evidence about human inductive biases (Shepard, 1994; Feldman, 1997). Psychologists sometimes study what can be learned from textual corpora and other massive data sets, but exploring what humans learn from highly constrained data sets can be equally valuable.

Acknowledgments. This work was supported by the James S. McDonnell Foundation Causal Learning Collaborative Initiative and by NSF award CDI-0835797.

References

Bott, L., & Heit, E. (2004). Nonmonotonic extrapolation in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *30*(1).

Brehmer, B. (1974). Hypotheses about relations between scaled variables in the learning of probabilistic inference tasks. *Organizational Behavior and Human Decision Processes*, *11*, 1-27.

Busemeyer, J., Myung, I. J., & McDaniel, M. (1993). Cue competition effects: Theoretical implications for adaptive network learning models. *Psychological Science*, *4*(3), 196.

Carroll, J. (1963). Functional learning: The learning of continuous functional mappings relating stimulus and response continua.

DeLosh, E. L., Busemeyer, J. R., & McDaniel, M. A. (1997). Extrapolation: The sine qua non of abstraction in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *23*, 968-986.

Feldman, J. (1997). The structure of perceptual categories. *Journal of Mathematical Psychology*, *41*, 145-170.

Griffiths, T. L., Lucas, C. G., Williams, J. J., & Kalish, M. L. (2009). Modeling human function learning with Gaussian processes. *Advances in Neural Information Processing Systems*, *21*.

Juslin, P., Karlsson, L., & Olsson, H. (2008). Information integration in multiple cue judgment: A division of labor hypothesis. *Cognition*, *106*(1), 259–298.

Kalish, M., Lewandowsky, S., & Kruschke, J. (2004). Population of linear experts: Knowledge partitioning and function learning. *Psychological Review*, *111*, 1072-1099.

Kelley, H., & Busemeyer, J. (2008). A comparison of models for learning how to dynamically integrate multiple cues in order to forecast continuous criteria. *Journal of Mathematical Psychology*, *52*(4), 218–240.

Koh, K. (1993). Induction of combination rules in two-dimensional function learning. *Memory & cognition*, *21*(5), 573–590.

Koh, K., & Meyer, D. (1991). Function learning: Induction of continuous stimulus-response relations. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *17*(5), 811–836.

Shepard, R. N. (1994). Perceptual-cognitive universals as reflections of the world. *Psychonomic Bulletin and Review*, *1*, 2-29.