COGNITIVE SCIENCE
A Multidisciplinary Journal

Cognitive Science 47 (2023) e13262

ISSN: 1551-6709 online
DOI: 10.1111/cogs.13262

Learning to Learn Functions

Michael Y. Li,[a] Fred Callaway,[b] William D. Thompson,[c] Ryan P. Adams,[c] Thomas L. Griffiths[b,c]

[a]*Department of Computer Science, Stanford University*
[b]*Department of Psychology, Princeton University*
[c]*Department of Computer Science, Princeton University*

Received 29 November 2021; received in revised form 12 January 2023; accepted 17 January 2023

Abstract

Humans can learn complex functional relationships between variables from small amounts of data. In doing so, they draw on prior expectations about the form of these relationships. In three experiments, we show that people learn to adjust these expectations through experience, learning about the likely forms of the functions they will encounter. Previous work has used Gaussian processes—a statistical framework that extends Bayesian nonparametric approaches to regression—to model human function learning. We build on this work, modeling the process of learning to learn functions as a form of hierarchical Bayesian inference about the Gaussian process hyperparameters.

Keywords: Function learning; Gaussian process; Hierarchical Bayesian models; Learning-to-learn; Bayesian nonparametrics

1. Introduction

Many problems humans solve, from learning causal relationships to reasoning about physics, are examples of *function learning*, or learning continuous relationships between inputs and outputs. Human function learning has been extensively studied, resulting in a number of classic models based on the idea that people learn simple rules that characterize functional relationships (Carroll, 1963; Kohn & Meyer, 1991) or form generalizations

Correspondence should be sent to Michael Y. Li, Department of Computer Science, 353 Jane Stanford Way, Stanford University, Stanford, CA 94305-9025. E-mail: michaelyli@stanford.edu

based on similarity (DeLosh et al., 1997). Recently, rational models of function learning based on Gaussian processes (GPs)—an extension of Bayesian nonparametric approaches to regression—have been used to unify these two perspectives (Lucas, Griffiths, Williams, & Kalish, 2015). GPs make explicit the expectations that learners have about the forms of the functions they might encounter, defining prior distributions over functions that can incorporate assumptions about properties, such as linearity, periodicity, symmetry, and smoothness (Brehmer, 1974; DeLosh et al., 1997). These expectations are typically encoded through a *kernel function* that specifies the covariance between two function values, as determined by their inputs. For example, a kernel that assigns high covariance to inputs that are close implies a prior that favors smooth functions. GPs have been shown to capture a range of behavioral phenomena in human function learning experiments (Wilson, Dann, Lucas, & Xing, 2015; Lucas, Griffiths, Williams, & Kalish, 2015; Wu, Schulz, & Gershman, 2019). Since the kernel naturally encodes learners' expectations, previous work has primarily focused on understanding which kernels best capture human behavior (Wilson, Dann, Lucas, & Xing, 2015; Schulz, Tenenbaum, Duvenaud, Speekenbrink, & Gershman, 2017; Wu, Schulz, Speekenbrink, Nelson, & Meder, 2018; Wu, Schulz, & Gershman, 2019). However, this previous work characterizes humans' expectations for functions prior to learning; it does not study how humans adapt their expectations based on previous data.

While choosing an appropriate family of kernels is important when applying GPs to regression problems, it is also crucial to choose *hyperparameters* appropriate for data. Kernel hyperparameters significantly impact the structure the GP can capture, including the average function value, the existence of global trends (e.g., linearly increasing), and the extent to which one can extrapolate from the data (Dowling, Zhao, & Park, 2020; Murray & Adams, 2010; Rasmussen & Williams, 2006; Shahriari, Swersky, Wang, Adams, & de Freitas, 2016; Snoek, Larochelle & Adams, 2012). To learn different kinds of structure, humans must adapt their expectations in ways consistent with learning different hyperparameters—effectively learning to learn functions; this type of learning has not been previously explored in the function learning literature.

For example, consider Fig. 1, which illustrates our experimental task. Participants observe a sequence of scatter plots of functions and are asked to predict the function value at a location in the input space for which the function value has not been observed. After experience with this task (the Training Task, shown in Fig. 1A), they are asked to make a prediction about the function value at the input location indicated by the vertical blue line given the information provided by the single data points (the Test Task, Fig. 1B). Intuitively, their predictions in this task will depend strongly on their expectations. If they experience functions with a linearly increasing trend during the Training Task (top row, Fig. 1A), they may learn to expect an increasing global trend (purple). In contrast, if they experience functions with a linearly decreasing trend (bottom row, Fig. 1A), they may expect a decreasing global trend (orange). The data they have previously seen affect how they predict in the Test Task.

In this paper, we capture this process of learning to learn through *hierarchical Bayesian inference* (Austerweil, Sanborn, & Griffiths, 2019; Kemp, Perfors, & Tenenbaum, 2007; Lucas & Griffiths, 2010). In this model, kernel hyperparameters are inferred through experience with one function and used to inform learning about the next. This model offers a

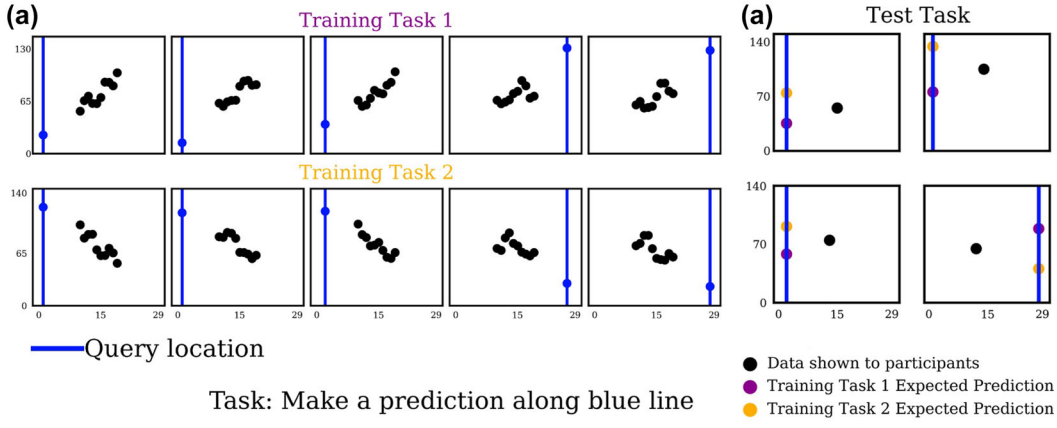

Fig. 1. Task demonstration. (A) Participants are shown the black dots and are asked to predict the function value at the query point. After predicting, their prediction is plotted along with the true function value. (B) We predict that training data will shape participant expectations in a way that will manifest on the test task.

computational-level account of learning to learn functions: in the spirit of Marr (1982) and Anderson (1990), we formulate the problem of learning to learn functions abstractly, explore solutions to this problem from statistics and machine learning, and compare these solutions to human behavior to gain insight. In this case, this hierarchical Bayesian approach reveals whether people are appropriately sensitive to the statistics of previous experience when forming expectations about functions. To evaluate the predictions of this model, we conduct three experiments using the task illustrated in Fig. 1. In each experiment, we present two groups of participants with functions generated from GPs with different hyperparameters. We then compare their predictions on test trials where both groups see the same data. Across all three experiments, we find that participants in different groups make systematically different predictions, providing experimental evidence that people adapt their expectations in ways consistent with hierarchical Bayesian inference.

2. Learning to learn for GPs

Our key theoretical proposal is that the behavioral patterns demonstrated in Fig. 1 can be captured by hierarchical Bayesian inference over GP hyperparameters. In this section, we introduce these ideas, starting with a formal definition of GPs and then demonstrating how hierarchical Bayesian inference can be used to learn functions.

2.1. Gaussian processes

GPs allow us to define distributions over functions; the GP is defined by the property that any finite set of N observations $\{\mathbf{x}_n\}_{n=1}^{N}$ induces a multivariate Gaussian distribution on $\mathbb{R}^N$, where the nth of these points is the function value, $f(\mathbf{x}_n)$, at the input point $\mathbf{x}_n$ (Rasmussen & Williams, 2006). GPs are fully characterized by a mean function $m(\mathbf{x})$ and positive definite

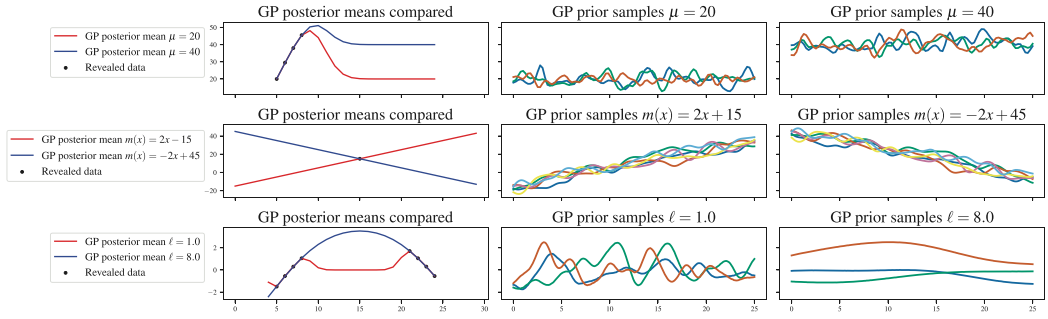

Fig. 2. Effect of GP hyperparameters. Column 1 illustrates how the posterior mean of a GP depend on different hyperparameters. Columns 2 and 3 compare functions sampled from the GP prior. The first row varies a constant prior mean, the second contrasts positive and negative linear functions as the prior mean, and the third varies the smoothing parameter ℓ.

kernel function $k(\mathbf{x}, \mathbf{x}')$ giving the covariance between $f(\mathbf{x})$ and $f(\mathbf{x}')$ as a function of $\mathbf{x}$ and $\mathbf{x}'$. Intuitively, the kernel function can be thought of as encoding expectations about what functions might be represented in observed data. Let $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^{N}$ be a dataset of N pairs of training inputs and corresponding output. The posterior predictive distribution at $f(\mathbf{x}_*)$, conditioned on dataset $\mathcal{D}$, for a new input $\mathbf{x}_*$, is Gaussian with mean and variance given by:

$$\mathbb{E}[f(\mathbf{x}_*)|\mathcal{D}] = m(\mathbf{x}_*) + \mathbf{k}_*{}^T(\mathbf{K} + \sigma^2\mathbf{I})^{-1}(\mathbf{y} - m(\mathbf{x}_*)), \tag{1}$$

$$\mathbb{V}[f(\mathbf{x}_*)|\mathcal{D}] = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*{}^T(\mathbf{K} + \sigma^2\mathbf{I})^{-1}\mathbf{k}_*. \tag{2}$$

Here, $\mathbf{K}$ is an $N \times N$ matrix of covariances evaluated at all pairs of training points with $[\mathbf{K}_{i,j}] = k(\mathbf{x_i}, \mathbf{x_j})$, $\mathbf{I}$ is the identity matrix, and $\mathbf{k}_* = [k(\mathbf{x}_1, \mathbf{x}_*), \ldots, k(\mathbf{x}_N, \mathbf{x}_*)]^T$ is a vector of covariances between the training outputs and $f(\mathbf{x}_*)$ (the superscript ^{T} indicates that we take the transpose of the vector). The σ^2 terms reflect the assumption that the underlying function is corrupted with additive Gaussian noise with variance σ^2.

The mean function $m(\mathbf{x})$ can be set to zero or modeled as a parametric function, whose parameters are learned from data. Using an explicitly defined prior mean function can capture expectations about the absolute or relative value of the functions from the prior (Rasmussen & Williams, 2006). The first row of Fig. 2 shows that the prior mean can control the average function value and the second row shows that it can control globally linear trends.

There are many possible kernel functions but we focus on the radial basis function (RBF) kernel which has been used in previous studies of human function learning (Lucas & Griffiths, 2010). The hierarchical Bayesian inference approach we present in this paper can be naturally extended to other kernels. For the RBF kernel, we have

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left\{-\frac{|\mathbf{x} - \mathbf{x}'|^2}{2\ell^2}\right\}, \tag{3}$$

where σ_f^2, the signal variance, determines the scale of function values and ℓ, the lengthscale parameter, controls how quickly the covariance between two outputs decays. Larger values of ℓ correspond to smoother functions.

The lengthscale ℓ is another hyperparameter that significantly impacts GP behavior, as Fig. 2 illustrates. In column 1, row 3, we compare two zero mean GP models with the same kernel (RBF), fit to the same data, but with different lengthscales (1.0 and 8.0). The predictions are nearly identical around the data but, away from the data, the predictions are drastically different. The short lengthscale model immediately reverts to the prior mean because the kernel function approaches zero in the interpolation region, whereas the long lengthscale model smoothly interpolates.

2.2. Learning to learn and hierarchical Bayes

The change in expectations through experience in Fig. 1 is a form of learning to learn. Previous research in machine learning (Baxter, 1998; Grant, Finn, Levine, Darrell, & Griffiths, 2018) and cognitive science (Austerweil et al., 2019; Kemp et al., 2007; Lucas & Griffiths, 2010) has described this process of learning to learn using hierarchical Bayesian inference. In this approach, rather than having a fixed prior distribution over hypotheses (in the case of function learning, a prior over functions), the learner *infers the prior distribution from data*. In our case, this prior distribution takes the form of a distribution over the GP hyperparameters. Thus, we model the process of learning to learn functions as inference of a distribution over GP hyperparameters from data.

We describe our modeling approach formally. For tractability, we represent the prior on the hyperparameters using a discrete set of means and variances, each of which parameterizes a Gaussian distribution, with fixed mean and variance, that represents a distribution over hyperparameters (we describe these choices in detail in the Supplementary Materials). Let random variable ϕ index a particular Gaussian distribution. Let $\phi \sim \text{Categorical}(\alpha_1 \dots \alpha_k)$ and take support in $\{1, 2, \dots, k\}$; each element in its support will correspond to a Gaussian distribution with fixed mean and variance. The model is represented as:

$$\phi \sim \text{Categorical}(\alpha_1 \dots \alpha_k) \qquad \theta|\phi \sim \text{N}(\mu_\phi, \sigma_\phi^2),$$

where θ denotes the GP hyperparameter, μ_ϕ and σ_ϕ are the mean and standard deviation of a Gaussian distribution and are indexed by ϕ.

Given this model, we now describe how to infer a distribution over ϕ (i.e., learning weights $\alpha_1 \dots \alpha_k$) from data (initially, we assume $\alpha_1 = \alpha_2 = \dots = \alpha_k$). Let $(\mathbf{y}_1, \mathbf{X}_1), (\mathbf{y}_2, \mathbf{X}_2), \dots, (\mathbf{y}_N, \mathbf{X}_N)$ be data observed on N training trials. Then, we have:

$$P(\phi|(\mathbf{y}_1, \mathbf{X}_1), (\mathbf{y}_2, \mathbf{X}_2), \dots, (\mathbf{y}_N, \mathbf{X}_N)) \propto \int_\theta (\prod_{n=1}^{N} P(\mathbf{y}_n|\mathbf{X}_n, \theta)) P(\theta|\phi) P(\phi) d\theta,$$

where $P(\mathbf{y}_n|\mathbf{X}_n, \theta)$ is the GP marginal likelihood of the data for a particular hyperparameter and has a closed-form expression. Given the prior $P(\phi)$, we can obtain $P(\theta) = \sum_k P(\theta|\phi = k)P(\phi = k)$.[1]The discretized prior makes this expression computationally tractable. $P(\theta)$

is a mixture of Gaussians with weights inferred from training data. In models considered hereafter, we utilize this learned prior distribution over hyperparameters, $P(\theta)$. Note, we have described our model assuming a distribution over a single hyperparameter θ is being inferred, but this straightforwardly extends to multiple hyperparameters by letting ϕ index a multivariate Gaussian. In some of our analyses, we will also infer joint distributions over hyperparameters.

2.3. Generating model predictions from learned prior distribution

Given a learned prior distribution $P(\theta)$ from the training trials, how do we generate predictions at a test input $\mathbf{x}^{\text{test}}$ given test dataset $\mathcal{D}^{\text{test}}$? Let $\mu_{(\mathbf{x}^{\text{test}},\mathcal{D}^{\text{test}},\theta)}$ denote the posterior mean of a GP at $\mathbf{x}^{\text{test}}$, fitted with data $\mathcal{D}^{\text{test}}$, and with hyperparameter θ. One approach is to "average" the posterior mean of the GP over $P(\theta)$, the distribution learned from training trials. Another approach is to average the posterior mean of the GP over an updated learned distribution $P(\theta|\mathcal{D}^{\text{test}})$, which is proportional to $P(\mathcal{D}^{\text{test}}|\theta)P(\theta)$ where $P(\theta)$ is the learned prior distribution. We will refer to the former approach as the hyperparameter prior model:

$$\frac{1}{M}\sum_{j=1}^{M}\mu_{(\mathbf{x}^{\text{test}}\mathcal{D}^{\text{test}},\theta_j)},\ \theta_j \sim P(\theta)$$

and the latter approach as the hyperparameter posterior model:

$$\frac{1}{M}\sum_{j=1}^{M}\mu_{(\mathbf{x}^{\text{test}},\mathcal{D}^{\text{test}},\theta_j)},\ \theta_j \sim P(\theta|\mathcal{D}^{\text{test}}).$$

The hyperparameter prior model accounts for data only on training trials, whereas the hyperparameter posterior model also accounts for data on a test trial $\mathcal{D}^{\text{test}}$. In both cases, we use a Monte Carlo approximation to a "fully Bayesian" treatment of GP hyperparameters in which they are integrated out (Filippone & Girolami, 2014; Lalchand & Rasmussen, 2020; Murray & Adams, 2010).

This formulation of hierarchical Bayesian inference over functions makes it possible to generate precise predictions about the consequences of receiving experience with particular classes of functions for the predictions people are likely to generate. In the remainder of the paper, we test these predictions, examining whether people are able to update their expectations about the form of functions and then use those new expectations to generate meaningful predictions about the properties of new functions.

3. Experiment 1: Learning a constant prior mean

One simple GP hyperparameter is the prior mean. In Experiment 1, we assess participants' ability to learn the prior mean of a GP using a task similar to that shown in Fig. 1.

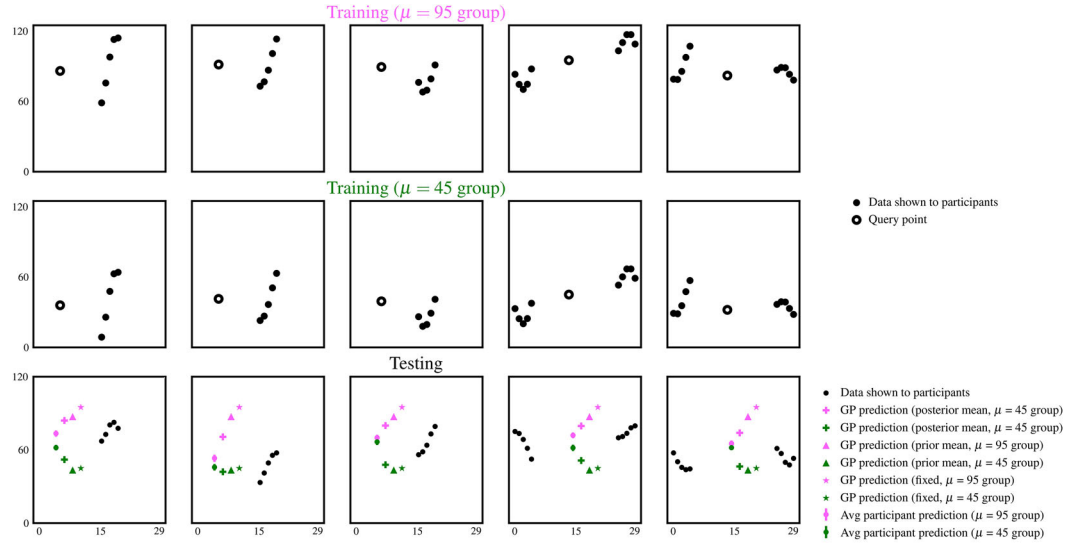

Fig. 3. Experiment 1 results. Rows 1 and 2 correspond to data shown during five training trials, which are vertical translations of each other. Row 3 plots average participant predictions and standard error of mean on test trials against various model predictions (GP predictions with true means, hyperparameter posterior model, and hyperparameter prior model). Participants are influenced by training and are better captured by a model that incorporates the influence of both training and test data. Note that the horizontal positions of the model predictions are jittered to improve legibility; all predictions were made for the x value indicated by the human data.

3.1. Methods

3.1.1. Participants

Eighty-one participants were recruited on Prolific (42 in high mean group, 39 in low mean group). They received $0.45 along with a performance-dependent bonus of up to $0.20. The average pay rate was $10.27 per hour.

3.1.2. Stimuli

We constructed two sequences of 10 functions to produce two different sets of training trials, one for each for participant group. To do this, 10 samples were drawn from a zero mean GP with an RBF kernel ($\ell = 2.0$, $\sigma_f^2 = 200$). To manipulate the mean function value between groups, a constant value of 95 is added to the samples in one group and a constant value of 45 is added to the other group. The test set consists of five functions corresponding to samples from a GP with varying means. The practice set consists of four functions. Two functions are produced by adding 95 to samples from a GP and the other two are produced by adding 45. The top and middle panels of Fig. 3 present example functions from the training trials. The blue dot indicates the true function value where participants were queried and the black dots indicate the data they were shown. The data in different groups are vertical translations of each other.

3.1.3. Task, design, and procedure

Participants completed 19 trials (first 4 "practice," next 10 "training," and last 5 "test") in which they were shown a scatter plot of noiseless (one-dimensional) function values and were asked to predict the function value at a point in the input space for which the function value had not been observed. Participants were randomly split into two groups whose training trials consisted of functions generated by samples from GPs with different prior means. The test trials consisted of a set of functions common across all participants. These test trials control for the test data and thus can highlight different inductive biases across groups of participants.

The experiment was presented in a web browser using PsiTurk. See online Supplementary Materials for screenshots. The instructions read:

> In this experiment, you will be shown data from functions. Some of the data will be drawn as black dots on a graph. Other data is hidden. Your job is to make predictions about the hidden data. We will ask for one prediction per graph. The more accurate you are, the higher bonus you will receive.

Participants were shown four functions during practice trials. Participant responses on practice trials are excluded from all analysis. This text appeared throughout the practice trials:

> Below, you see a graph of black dots. The black dots represent data from a function. Move your cursor along the blue line and click your mouse to place your prediction. After you place your prediction, your prediction will be drawn as a blue dot. For the first 10 functions, the hidden data will also be revealed and drawn as a black dot. Remember, you will get a higher bonus if your predictions are more accurate.

On both training and test trials, participants were shown function values plotted on a graph. Participants were then asked to make a prediction at an unseen location. Participants made a prediction by navigating their cursor and clicking. On training trials, after making a prediction, the true function value at the unseen location was plotted as a black dot. Participants were awarded a larger bonus for more accurate predictions. On test functions, no feedback was given. There were 10 training trials, corresponding to 10 different functions, and five test trials, corresponding to five different functions. Training functions appeared before test functions. However, the presentation order of functions within training and test trials was randomized.

Importantly, the same y-axis and x-axis are used to display scatterplots for all participants across both groups. The distance between the maximum y-value and the average of the function values in the first condition is the same as the distance between the minimum y-value and the average of function values in the second condition.

This task is different from those in early studies of function learning, where data or functions were not explicitly visualized. However, this style is common in recent papers (Schulz et al., 2017; Wilson et al., 2015) and allows us to characterize human behavior separately from limitations of human memory.

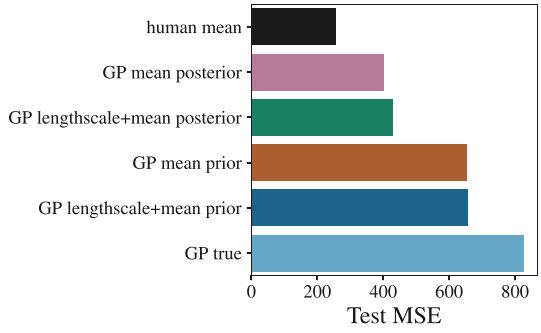

Fig. 4. Comparing GP models on Experiment 1. MSE (mean squared error) of various GP models aggregated across groups and test trials. The GP posterior model that learns a mean hyperparameter and adapts to each test trial performs the best.

3.1.4. Modeling approach

We consider the following GP models (see Section 2.3 for details): a hyperparameter prior model that infers just the prior mean (e.g., GP mean prior), a hyperparameter prior model that infers both the lengthscale and the mean (e.g., GP lengthscale+mean prior), a hyperparameter posterior model that infers just the prior mean (e.g., GP mean posterior), a hyperparameter posterior model that infers both the lengthscale and the mean (e.g., GP lengthscale+mean posterior), and a GP model whose lengthscales and means are set to the true mean and lengthscales (e.g., GP true).[2] The hyperparameter prior model accounts for data only on training trials, whereas the hyperparameter posterior model also accounts for data on each test trial. The contrast between our models will allow us to investigate how training and test data collectively influence participant predictions.

3.2. Results

We hypothesized that participants in the high mean group (offset of 95) would predict higher function values than participants in the low mean group (offset of 45) even when shown exactly the same data. Indeed, in all five test functions, the average prediction in the high mean group is higher (bottom panel, Fig. 3). A mixed effects regression with participant predictions as the outcome variable and random intercepts for each test trial and subject and a fixed effect for the group (high vs. low mean) was performed. The effect of group was statistically significant ($\beta = 7.165, t(79) = 3.67, p < .001$; tests of significance were performed using Satterthwaite's approximation).

Fig. 3 plots the predictions of the hyperparameter posterior and hyperparameter prior models that infer just the prior mean. The hyperparameter posterior model predictions appear as pink and green crosses (corresponding to models fit with high and low mean group training data, respectively) and the hyperparameter prior model predictions appear as pink and green triangles. The hyperparameter posterior model matches mean human predictions more closely than the hyperparameter prior model, a result that is confirmed in Fig. 4 which quantitatively compares the mean squared error (MSE), aggregated across groups

and test trials, of several different models. Both hyperparameter posterior models drastically outperform a simple baseline GP model (i.e., GP true) that uses a fixed lengthscale and mean set to the values that generated the data for each group.

The gap between hyperparameter posterior model predictions, for each group, is significant but smaller, a pattern that is qualitatively similar to human predictions. The hyperparameter posterior model provides better fits because its predictions are influenced by both training and test data. Since it is influenced by training data, it gives a qualitative account of the differences in predictions between participant groups. Since it is also influenced by test data, it accounts for the smaller differences observed between groups compared to the hyperparameter prior model. Interestingly, the hyperparameter posterior model still underpredicts how much participants are influenced by test stimuli. In Section 3 of the Supplementary Materials, we show that the hyperparameter posterior model performs competitively against GP models with fixed parameters optimized for the participant data.

The results indicate that learning has occurred on the training trials that influences how participants predict on test trials. Exposure to a particular prior mean influences participant predictions.

4. Experiment 2: Learning a linear prior mean

Experiment 1 illustrated participants' ability to learn a constant prior mean. A natural followup is investigating whether participants can learn a more complicated parametric prior mean. In Experiment 2, we assessed if participants' can learn the slope of a GP with a linear prior mean.

4.1. Methods

4.1.1. Participants

A total of 105 participants (54 in negative slope group, 51 in positive slope group) were recruited on Prolific. For participation, they received \$0.55 and a performance-dependent bonus up to \$0.30. The average pay rate was \$11 per hour.

4.1.2. Stimuli

We draw 10 samples from a zero mean GP ($\ell = 1.0$, $\sigma_f^2 = 75$). For one group, we add a linear function with a positive slope to each of the 10 samples. For the other group, we take the mirror reflection. The top two rows of Fig. 5 give some examples of the data shown to participants in the training trials. The data can be described as having a global trend with GP residuals. Functions in the training set are presented in randomized order. Participants are randomly assigned to either to a positive slope group or negative slope group.

The test set consists of five single data points (Fig. 5, bottom panel, black points). The motivation behind using a single point is that there is not enough information in the data to infer the slope and so any differences in predictions can be attributed to an inductive bias (for either a positive or negative slope) developed through the training trials. No feedback is

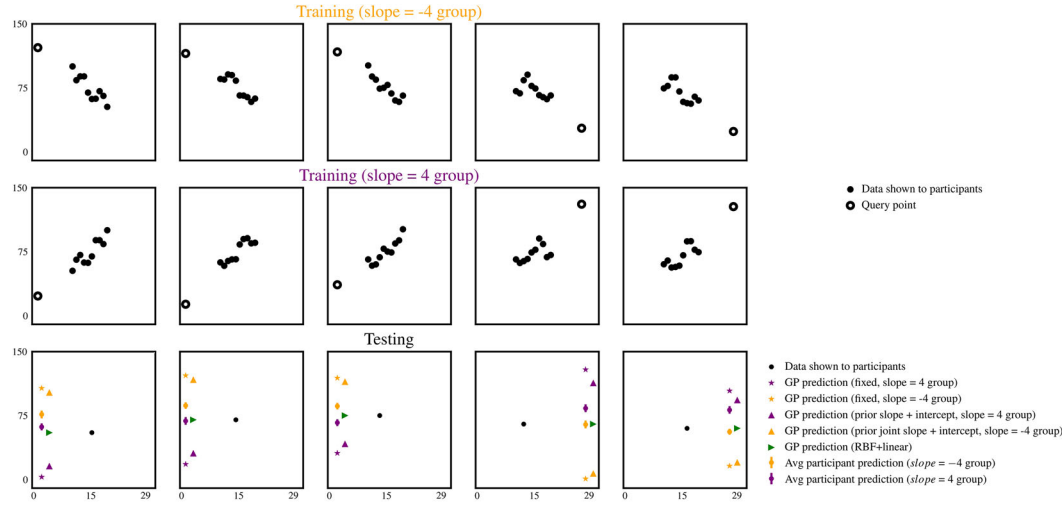

Fig. 5. Experiment 2 results. Rows 1 and 2 present training data, which exhibits either an increasing or decreasing trend. Row 3 compares average participant prediction and standard error of mean to various models. Participants in each group exhibit a slope bias consistent with training. GP with a linear prior mean and slope set to the true slopes or GP with linear prior mean with sampled slopes can capture this bias.

shown during test trials and test trials appear in randomized order. The practice set consists of four functions, two with a positive slope and two with a negative slope.

4.1.3. Task, design, and procedure

These are all identical to Experiment 1.

4.1.4. Modeling approach

We consider three models for Experiment 2. For the first model, we adapt the hyperparameter prior model introduced earlier to infer a distribution over the slope and intercept of a GP with RBF kernel and linear prior mean. For the second model, we fix the slopes of a GP with RBF kernel and linear prior mean to the ground truth values used to generate training trials data. The third model is a zero-mean GP whose kernel is the sum of an RBF kernel and linear kernel (which we call the RBF+linear kernel).

4.2. Results

We predict that participants will learn to expect either a positive or negative slope depending on their condition. The difference in expectations should be reflected on test trials as follows. To the left of the single data point, linear functions with positive slopes passing through the single data point have lower function values than linear functions with negative slope. The opposite relationship will hold to the right of the data point.

We observe the predicted trend in participants' predictions in Fig. 5 (mean predictions appear as purple and orange circles for the positive and negative slope groups, respectively).

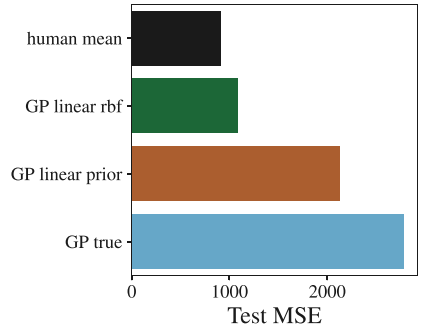

Fig. 6. Comparing GP models for Experiment 2 MSE (mean squared error) of various GP models aggregated across groups and test trials. The linear RBF model attains the lowest MSE but fails to capture the group differences.

On test trials where participants are asked to make predictions to the left of the single data point, the average participant prediction in the positive slope group is lower. On test trials where participants are asked to make predictions to the right of the single data point, the average participant prediction in the positive slope group is higher. We performed a mixed effects regression with participant predictions as the outcome variable and for independent variables we use an interaction term between the group and a binary indicator variable indicating whether the trial asked for a query to the left or right (trial type) and a random effect for each test trial. Our estimate for the coefficient on the interaction term between condition and trial type indicator is statistically significant ($\beta = -39.590, t(518) = -7.310, p < .001$) and our estimate of the coefficient on condition is also statistically significant ($\beta = 22.299, t(518) = 5.315, p < .001$).

The bottom panel of Fig. 5 plots the model predictions described in the Methods section. The orange and purple triangles correspond to predictions produced by the hyperparameter prior model, in which GP predictions are averaged over the learned distribution over slopes. Participants in different groups have a bias toward either a positive or negative slope. If we force the GP with RBF+linear kernel (green sideways triangle) to pass through the revealed data point,[3] it cannot produce a bias toward positive slopes. The GP prediction will be a flat line, as illustrated by green sideways triangles in Fig. 5. On the other hand, we can capture this slope bias by inferring a distribution over the slope and intercept of the linear prior mean.

In Fig. 6, we compare the aggregated MSEs of the GP linear RBF model, the hyperparameter prior model (i.e., GP linear prior), and a GP model whose hyperparameters are set to the parameters that generated the experiment data (i.e., GP true). The hyperparameter prior model significantly outperforms the GP true model. However, the GP linear RBF model outperforms the hyperparameter prior model. Since the mean participant predictions are close to the revealed function values, are participants all just predicting the revealed point? In the Supplementary Materials, we examine the distribution of participant predictions and show that this is not the case. Collectively, Experiments 1 and 2 demonstrate that humans' sensitivity to the mean value and ability to learn global trends, can be captured through the prior mean.

5. Experiment 3: Learning the lengthscale

The previous two experiments show that people learn a prior from experience that changes their predictions on new data. However, in both cases, learning an appropriate prior involves learning a simple statistical property (i.e., the prior mean) of the data but does not involve learning how specific data tends to influence a prediction. Here, we consider a hyperparameter that affects how data should be utilized to make predictions. Specifically, we consider the lengthscale hyperparameter, ℓ, which controls the smoothness of functions sampled from the GP. From a psychological perspective, the lengthscale is interesting because it determines how much participants should generalize from observed data (Wu et al., 2018). The lengthscale hyperparameter is also ubiquitous. Other kernels used to model human function learning (Schulz et al., 2017; Wilson et al., 2015) have analogous lengthscale hyperparameters that also influence smoothness of functions from the prior.

5.1. Methods

5.1.1. Participants

Hundred participants (51 in short lengthscale group, 49 in long lengthscale group) were recruited on Prolific. For participation, they received $0.55 and performance-dependent bonus up to $0.30. The average pay rate was $11 per hour.

5.1.2. Stimuli

We draw two sets of 10 samples from a GP with RBF kernel ($\mu = 60$, $\sigma_f^2 = 200$), one set for each group of participants. The first set of functions is generated by an RBF kernel with lengthscale of $\ell = 2$. The second set of functions is generated by an RBF kernel with lengthscale of $\ell = 11$. Participants are randomly assigned to the short-lengthscale group or the long-lengthscale group. The top two rows of Fig. 7 give example functions from the training trials. The test set consists of four functions, corresponding to four test trials, that are diagnostic for capturing differences between models with different lengthscales.

5.1.3. Task, design, and procedure

These are identical to Experiment 1.

5.1.4. Modeling approach

We consider the following GP models: a hyperparameter prior model that infers just the lengthscale (e.g., GP lengthscale prior), a hyperparameter prior model that infers both the lengthscale and the mean (e.g., GP lengthscale+mean prior), a hyperparameter posterior model that infers just the lengthscale (e.g., GP lengthscale posterior), a hyperparameter posterior model that infers both the lengthscale and the mean (e.g., GP lengthscale+mean posterior), and a GP model whose lengthscales and means are set to the true mean and lengthscales (e.g., GP true).

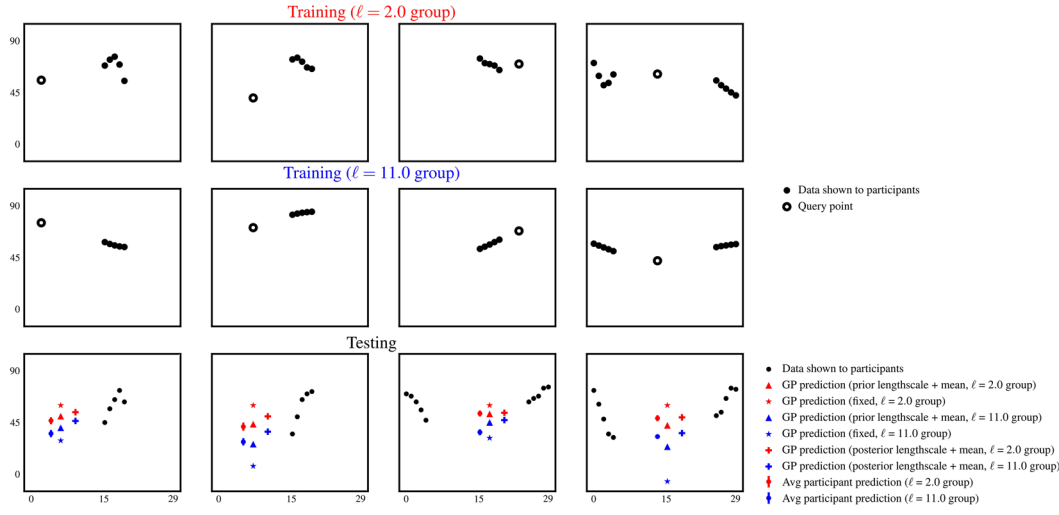

Fig. 7. Experiment 3 results. Rows 1 and 2 present data shown during training. Row 3 presents average participant predictions and standard error of mean on test trials versus GP model predictions. Participants in short-lengthscale group (red circle) revert to the mean, whereas participants in long-lengthscale group (blue circle) smoothly interpolate or extrapolate, consistent with learning the correct lengthscales.

5.2. Results

The differences between human predictions in each group are consistent with the differences between GP predictions with different lengthscales (Fig. 7, bottom panel). The main difference between the predictions under different lengthscales is the range over which one generalizes from the data. The short-lengthscale prediction quickly reverts to the prior mean, while the long-lengthscale prediction will smoothly interpolate or extrapolate. Indeed, participants in the short-lengthscale group revert to the prior mean, whereas participants in the long-lengthscale group extrapolate or interpolate from the revealed data (Fig. 7). We performed a mixed effects regression with participant predictions as the dependent variable and the group (short or long-lengthscale) as a fixed effect and random effects for each subject and each test trial. The coefficient on group was statistically significant ($\beta = 14.206, t(98) = 4.665, p < .001$). These differences suggest that participants make predictions in a way consistent with learning lengthscales.

In Fig. 8, we compare the MSEs of different GP models aggregated across groups and test trials. The models that learn distributions over hyperparameters significantly outperform the model that uses the true lengthscales (GP true). We also find that this model outperforms GP models with fixed parameters optimized for the human data (see Section 3 of Supplementary Materials). Interestingly, the hyperparameter prior models perform slightly better than the hyperparameter posterior models, which adapts to the properties of the test trials, in contrast to Experiment 1. This could suggest that participants are less sensitive to the lengthscale than they are to the mean.

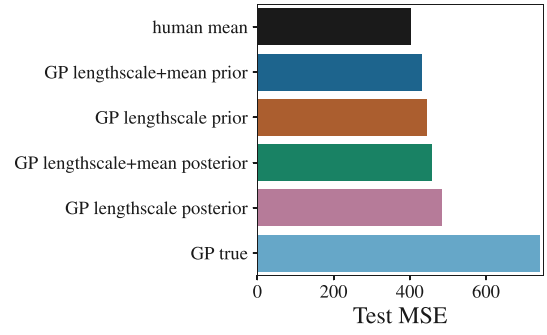

Fig. 8. Comparing GP models for Experiment 3 MSE (mean squared error) of various GP models aggregated across groups and test trials. The GP prior model that learns both a lengthscale and mean hyperparameter performs the best. Notably, all GP models that learn hyperparameters significantly outperform the model that uses the true lengthscale. Interestingly, in contrast to Experiment 1, the GP models that adapt to the test data perform slightly worse, suggesting that people are less sensitive to changes in smoothness on the test data than they are to the mean.

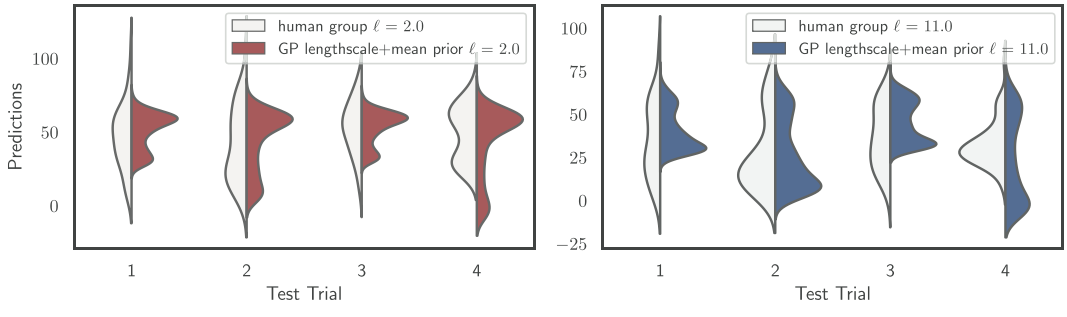

Fig. 9. Comparing GP model prediction versus human prediction distributions for Experiment 3. For each test trial, comparison of distribution of human predictions against GP model predictions for **(left)** $\ell = 2.0$ group and **(right)** $\ell = 11.0$ group. In some cases (e.g., Test Trial 4 on left, Test Trials 1, 2 on right), human predictions are multimodal. This is also reflected in the GP model since its predictions are produced from a learned distribution over lengthscales and means.

Fig. 9 compares the distributions of the predictions of the hyperparameter prior model that infers both the lengthscale and the mean with the human distribution. In some cases (e.g., Test Trial 4 on left, Test Trials 1, 2 on right), human predictions exhibit multimodality. The modes of this distribution essentially correspond to two different "strategies" on the test trials: reverting to the prior mean or smoothly extrapolating or interpolating. This multimodality cannot be captured by a GP model with a fixed lengthscale but can be captured by our model which learns distributions over hyperparameters. One explanation for the multimodality in human data is the existence of two distinct types of participants who predict consistently with either a short or long lengthscale through the entirety of the test trials. Another explanation is that some learners employ a "mixed strategy" and switch between strategies during the test trials. We show evidence for the latter type of learner in Section 2.2 of the Supplementary Materials.

6. Discussion

We studied humans' ability to adapt expectations about functions, modeling this behavior via hierarchical Bayesian inference of GP hyperparameters. We showed that people can learn higher-order structure underlying functions drawn from distributions defined by hyperparameters that control the mean, global trends, and smoothness of functions. People's predictions were broadly consistent with simple hierarchical Bayesian models that learn distributions over those hyperparameters given previously seen functions, and use them to inform predictions about new functions. That is, participants learned to learn new functions through experience.

6.1. Heuristics for function learning

In this work, our main contribution was a computational-level account (Marr, 1982) of learning to learn functions. However, we do not claim that Bayesian inference is the cognitive mechanism by which humans solve this task. Instead, participants might employ simple heuristics (Gigerenzer & Todd, 1999). In this section, we evaluate simple heuristics for our tasks against GP models; we discuss these heuristics in more detail in the Supplementary Materials (Section 4). There are two aims to this comparison. First, we want to take preliminary steps toward bridging the algorithmic-level and computational levels of analysis for human function learning (Griffiths, Lieder, & Goodman, 2015). We view these heuristics as a first step toward developing the algorithmic-level solutions to the computational problem we presented; we do not claim that these heuristics are approximations of GP models, and we leave it to future work to explore any potential connections. Second, since the heuristics are fairly interpretable, we think this comparison gives insight into the relative performance of the GP models across different settings, which could inform future work. Importantly, in making this comparison, we are not primarily interested in assessing which model is preferable. These models are at different levels of analysis and, therefore, offer complementary perspectives on human function learning.

In Fig. 10, we compare the best-performing (as measured by MSE on human data) GP model against several heuristic strategies. The best GP model appears in red, and the heuristics appear in gray. Across all three experiments, the GP model performs comparably or exceeds the performance of heuristics. In Experiment 1, the GP model outperforms a "predict train mean" heuristic that assumes participants predict the mean function value on the training trials and several variations of a heuristic that assumes participants are biased toward more recently seen data. We implement the latter as an exponential-decay Gaussian model where lower values of γ induce higher weights on more recently seen data, which happen to be the test trials in our experiments. In Experiment 2, the GP linear RBF model performs marginally worse than a "truncated Gaussian" heuristic that assumes participants in the negative slope group learn that predictions on the left-hand side of the plot are higher than the revealed data point and that those on the right-hand side are smaller; we consider an analogous heuristic for the positive slope group. In Experiment 3, the GP model significantly outperforms several variations of spline regression and the predict train mean heuristic.

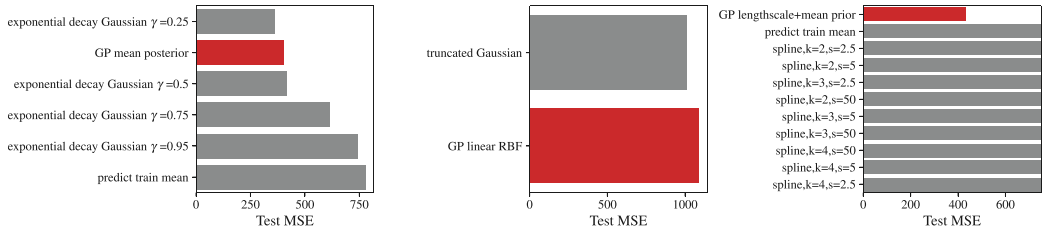

Fig. 10. Comparing best GP model against heuristics. (**left**) For the mean experiment, the best GP model significantly outperforms heuristics that predicts the mean of the training data and the exponentially weighted Gaussian model for two values of γ. The $\gamma = 0.25$ outperforms the GP model, suggesting that participants are more sensitive to the statistics of the test data than our Bayesian model predicts. (**middle**) The truncated Gaussian heuristic performs marginally better, likely because its predictions are guaranteed to reflect a slope bias. (**right**) For the lengthscale experiment, the best GP model outperforms heuristics that predict the mean of training data and several variations of spline regression.

The primary purpose of this comparison was not to distinguish between heuristics and adaptive GP models since these models are at different levels of analysis. However, their relative performance gives insight into when the adaptive GP models are especially well-suited for modeling how humans learn functional relationships. For the simpler settings (Experiments 1 and 2), the adaptive GP model and best heuristic perform similarly. For the most complex setting (Experiment 3), the adaptive GP model outperforms all heuristics. This suggests that the adaptive GP model is better suited for modeling how humans learn more complicated aspects of functional relationships. We leave a thorough exploration of this for future work.

6.2. Limitations and future directions

One limitation of our study is that we presented participants with scatter plots of functions rather than showing the input–output pairs one at a time, as is more common in naturalistic function learning. Although this paradigm has advantages, motivating its use in recent work on function learning (e.g., Schulz et al., 2017; Wu et al., 2019; Schulz, Tenenbaum, Reshef, Speekenbrink, & Gershman, 2015), it differs from earlier work on trial-by-trial learning of functions (e.g., Carroll, 1963; Kohn & Meyer, 1991; DeLosh et al., 1997). Future work could explore if similar results hold in more naturalistic settings. Another limitation of our results is that the GP models tend to predict larger group differences than actually observed in Experiments 1 and 2. For Experiment 1, one explanation is that some participants occasionally make predictions consistent with learning a longer lengthscale than the true lengthscale. This would yield predictions that are more sensitive to the test data than our hierarchical Bayesian model (which learns a shorter lengthscale) predicts. For Experiment 2, the differences may result from a discrepancy between how we formulate the learning task and the actual learning task participants solve. In Experiment 2, the test trial data consist of single points, while the training trial data consist of several points. Our hierarchical Bayesian model assumes that the prior learned during training trials is relevant for the predictions on the test trials. However, the learning problem humans face is potentially more complicated. In addition to learning

a prior from training trials, they also have to recognize that this prior is relevant to the test trials. Future work may consider how to model this aspect of the task by drawing on ideas from meta-learning (Zhou et al., 2021). Participants must also determine which kernel(s) to use and another interesting direction is to model how participants learn this drawing on work from machine learning (Duvenaud, Lloyd, Grosse, Tenenbaum, & Gharamani, 2013). More broadly, our emphasis on the computational level limits the extent to which we can capture human behavior. Another future research direction is to explore how certain forms of approximate GP inference or approximate forms of hyperparameter inference can be realized as simple heuristics. This line of work could help bridge the gap between computational-level models of function learning and psychologically plausible algorithmic-level models.

Although our approach has limitations, it also has strengths that could motivate further investigation. Our hierarchical Bayesian formulation gives rise to heterogeneity in predictions, a phenomenon that is also reflected in human data. Our models that jointly infer the lengthscale and mean can explain why people primarily learn a lengthscale in Experiment 3 and a mean in Experiment 1: the learned marginal distributions of the varying hyperparameter differ across groups, while the marginal distributions of the fixed hyperparameter agree.[4]

6.3. Conclusion

Better understanding how humans learn requires understanding how humans acquire the expectations that inform their learning. In this paper, we showed that hierarchical Bayesian inference provides a way to do this in the novel setting of function learning, building on previous accounts of learning to learn in other settings (Austerweil et al., 2019; Kemp et al., 2007; Lucas & Griffiths, 2010). These results may also be relevant in machine learning, where machines can face the same problem as people: reaching strong conclusions from limited data.

Notes

1 We omit dependence on training trials for brevity.
2 Since there are two sequences of training data, we infer a separate distribution over hyperparameters for each participant group.
3 We accomplish this by setting the offset parameter in the linear kernel to the input location of the revealed data point and set the prior mean to the observed value.
4 See Section 2 in the Supplementary Materials for details.

References

Anderson, J. R. (1990). *The adaptive character of thought*. Erlbaum.

Austerweil, J. L., Sanborn, S., & Griffiths, T. L. (2019). Learning how to generalize. *Cognitive Science*, *43*(8), e12777.

Baxter, J. (1998). Theoretical models of learning to learn. In S. Thrun & L. Pratt (Eds.), *Learning to learn* (pp. 71–94). Springer.

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

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

DeLosh, E. et al. (1997). Extrapolation: The sine qua non of abstraction in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *23*, 968–986.

Dowling, M., Zhao, Y., & Park, I. M. (2020). Non-parametric generalized linear model. *arXiv:2009.01362*.

Duvenaud, D., Lloyd, J., Grosse, R., Tenenbaum, J., & Gharamani, Z. (2013). Structure discovery in nonparametric regression through compositional kernel search. In *Proceedings of the 30th International Conference on Machine Learning* (pp. 1166–1174).

Filippone, M., & Girolami, M. (2014). Pseudo-marginal Bayesian inference for Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(11), 2214–2226.

Gigerenzer, G., & Todd, P. M. (1999). *Simple heuristics that make us smart*. Oxford University Press.

Grant, E., Finn, C., Levine, S., Darrell, T., & Griffiths, T. (2018). Recasting gradient-based meta-learning as hierarchical Bayes. In *International Conference on Learning Representations*.

Griffiths, T. L., Lieder, F., & Goodman, N. D. (2015). Rational use of cognitive resources: Levels of analysis between the computational and the algorithmic. *Topics in Cognitive Science*, *72*, 217–229.

Kemp, C., Perfors, A., & Tenenbaum, J. B. (2007). Learning overhypotheses with hierarchical Bayesian models. *Developmental Science*, *10*(3), 307–321.

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

Lalchand, V., & Rasmussen, C. E. (2020). Approximate inference for fully Bayesian Gaussian process regression. In *Symposium on Advances in Approximate Bayesian Inference* (pp. 1–12). PMLR.

Lucas, C. G., & Griffiths, T. L. (2010). Learning the form of causal relationships using hierarchical Bayesian models. *Cognitive Science*, *34*(1), 113–147.

Lucas, C. G., Griffiths, T. L., Williams, J. J., & Kalish, M. L. (2015). A rational model of function learning. *Psychonomic Bulletin & Review*, *22*(5), 1193–1215.

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information*. San Francisco: W.H. Freeman.

Murray, I., & Adams, R. P. (2010). Slice sampling covariance hyperparameters in latent Gaussian models. In *Advances in Neural Information Processing Systems (NIPS).*

Rasmussen, C. E., & Williams, C. K. (2006). *Gaussian processes for machine learning*. Adaptive Computation and Machine Learning. MIT Press.

Schulz, E., Tenenbaum, J., Reshef, D. N., Speekenbrink, M., & Gershman, S. (2015). Assessing the perceived predictability of functions. In *Proceedings of the 37th Annual Conference of the Cognitive Science Society* (pp. 2116–2121).

Schulz, E., Tenenbaum, J. B., Duvenaud, D., Speekenbrink, M., & Gershman, S. J. (2017). Compositional inductive biases in function learning. *Cognitive Psychology*, *99*, 44–79.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., & de Freitas, N. (2016). Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, *104*(1), 148–175.

Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems (NeurIPS).*

Wilson, A. G., Dann, C., Lucas, C., & Xing, E. P. (2015). The human kernel. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, & R. Garnett, (Eds.), *Advances in neural information processing systems* (pp. 2854–2862).

Wu, C. M., Schulz, E., & Gershman, S. J. (2019). Generalization as diffusion: Human function learning on graphs. In A. Goel, C. Seifert, & C. Freksa, (Eds.), *Proceedings of the 41st Annual Conference of the Cognitive Science Society* (pp. 3122–3128). Montreal, QB: Cognitive Science Society

Wu, C. M., Schulz, E., Speekenbrink, M., Nelson, J. D., & Meder, B. (2018). Generalization guides human exploration in vast decision spaces. *Nature Human Behaviour*, *2*, 915–924.

Zhou, P., Zou, Y., Yuan, X.-T., Feng, J., Xiong, C., & Hoi, S. (2021). Task similarity aware meta learning: Theory-inspired improvement on MAML. In C. de Campos, & Maathuis, M. H. (Eds.), *Proceedings of the 37th Conference on Uncertainty in Artificial Intelligence* (pp. 23–33).

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Data S1
Figure 1.