Better safe than sorry: Risky function exploitation through safe optimization

Eric Schulz[1], **Quentin J.M. Huys**[2], **Dominik R. Bach**[3], **Maarten Speekenbrink**[1] **& Andreas Krause**[4]

[1]Department of Experimental Psychology, University College London, London, WC1H0AP
[2]Translational Neuromodeling Unit, ETH and University of Zürich, Wilfriedstrasse 6, 8032 Zürich
[3]Psychiatric Hospital, University of Zürich, Lenggstrasse 31, 8032 Zürich
[4]Institute for Machine Learning, ETH Zürich, Universitaetstrasse 6, 8092 Zürich

Abstract

Exploration-exploitation of functions, that is learning and optimizing a mapping between inputs and expected outputs, is ubiquitous to many real world situations. These situations sometimes require us to avoid certain outcomes at all cost, for example because they are poisonous, harmful, or otherwise dangerous. We test participants' behavior in scenarios in which they have to find the optimum of a function while at the same time avoid outputs below a certain threshold. In two experiments, we find that Safe-Optimization, a Gaussian Process-based exploration-exploitation algorithm, describes participants' behavior well and that participants seem to care first about whether a point is safe and then try to pick the optimal point from all such safe points. This means that their trade-off between exploration and exploitation indicates intelligent, approximate, and homeostasis-driven behavior.

Keywords: Safe Optimization, Function Learning, Approximate Learning, Gaussian Process, Homeostasis

Introduction

Imagine you are hosting a dinner party. In the afternoon, you open up your fridge and kitchen cupboards to find a plethora of ingredients at your disposal. Aiming to amaze your friends with an unique culinary experience, you decide to prepare something extraordinary not found in recipe books. Considering your options, you generate expectations of how the tastes of different ingredients combine and interact to produce a – hopefully memorable – culinary experience. You have time to try out some options and experience their overall taste, learning about the effects of unusual combinations and methods of preparation. At the same time, however, you need to avoid certain combinations at all costs, for example those that are inedible, poisonous, or otherwise bad.

This scenario is an example of a multi-armed bandit task (Srinivas et al., 2009), where there are a number of actions or 'arms' of the bandit (e.g., the possible dishes) which lead to initially unknown and stochastic outcomes or rewards (e.g., the taste of the dish), which are related to a set of features (e.g., the ingredients, the method of preparation, etc.). Through experience, one can learn the function which maps the features to the rewards and maximize the overall rewards gained over repeated plays of the bandit. A key issue in optimal behavior in such tasks is known as the exploration-exploitation dilemma: should I take an action which I know will lead to a high reward, or try an unknown action to experience its outcome and thereby learn more about the function mapping features to rewards, increasing my chances of gaining higher rewards in the future? In order to avoid certain bad outcomes (e.g., poisonous dishes), one should only explore uncertain options which are likely to be 'safe'. Such restricted exploration-exploitation problems are ubiquitous in daily life, from choosing which restaurant to visit, which car to buy, all the way to whom to befriend. In our previous research on human behavior in contextual multi-armed bandits (Schulz et al., 2015a,b), we found that participants' behavior is well-described by Gaussian Process regression, a non-parametric regression tool that adapts its complexity to the data at hand by the means of Bayesian posterior computation.

The aim of the present study is to assess how people behave when they have to maximize their rewards whilst avoiding outcomes below a given threshold. The task is couched as a function learning task, where participants choose an input and observe and accrue the output of the function. In two experiments with a uni- and bivariate function, we find that participants efficiently adapt their exploration-exploitation behavior to risk-inducing situations. Overall, they are well-described by a Gaussian Process-based safe optimization algorithm that tries to safely expand a set of 'explorers' while simultaneously maximizing outputs within a set of possible 'maximizers; (Sui et al., 2015). Such behavior might be based on the principle of homeostasis maintenance (Korn & Bach, 2015), where organisms need to forage for food while avoiding the probability of starvation. Additionally, we find evidence that participants first assess whether points are safe and then attempt to maximize within this safe subset. This simplification of the task in terms of subgoals resonates well with recent results on approximate planning strategies in complex dynamic tasks (Huys et al., 2015).

Modeling learning and optimization

If the task is to learn and maximize an unknown function, then two ingredients are needed: (a) a model to represent an unknown function, for which we will use Gaussian process regression, and (b) a method to safely

[1]Corresponding author: eric.schulz@cs.ucl.ac.uk

choose the next inputs, for which we will use a safe optimization algorithm.

Learning a function

We assume people represent and learn a function through Gaussian process regression, a universal function learning algorithm which has been supported in previous research (Griffiths et al., 2009; Schulz et al., 2015b).
A Gaussian Process ($\mathcal{GP}$) is a stochastic process of which the marginal distribution of any finite collection of observations is multivariate Gaussian (Rasmussen, 2006). It is a non-parametric Bayesian approach towards regression problems and can be seen as a rational model of function learning as it adapts its complexity to the data encountered. Let $f(\boldsymbol{x})$ be a function mapping an input $\boldsymbol{x} = (x_1, \ldots, x_d)^\top$ to an output y. A $\mathcal{GP}$ defines a distribution $p(f)$ over such functions. A $\mathcal{GP}$ is parametrized by a mean function $m(\boldsymbol{x})$ and a covariance (or kernel) function, $k(\boldsymbol{x}, \boldsymbol{x}')$:

$$m(\boldsymbol{x}) = \mathbb{E}\left[f(\boldsymbol{x})\right] \tag{1}$$
$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}\left[(f(\boldsymbol{x}) - m(\boldsymbol{x}))(f(\boldsymbol{x}') - m(\boldsymbol{x}'))\right] \tag{2}$$

At time t, we have collected observations $\mathbf{y}_{1:t} = [y_1, y_2, \ldots, y_t]^\top$ at inputs $\boldsymbol{x}_{1:t} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t)$. For each outcome y_t, we assume

$$y_t = f(\boldsymbol{x}_t) + \epsilon_t \qquad \epsilon_t \sim \mathcal{N}(0, \sigma^2) \tag{3}$$

Given a $\mathcal{GP}$ prior on the functions

$$f(\boldsymbol{x}) \sim \mathcal{GP}\left(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')\right), \tag{4}$$

the posterior over f is also a $\mathcal{GP}$ with

$$m_t(\boldsymbol{x}) = k_{1:t}(\boldsymbol{x})^\top (\boldsymbol{K}_{1:t} + \sigma^2 \boldsymbol{I}_t) \boldsymbol{y}_{1:t} \tag{5}$$
$$k_t(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}_{1:t}(\boldsymbol{x})^\top (\boldsymbol{K}_{1:t} + \sigma^2 \boldsymbol{I}_t)^{-1} \boldsymbol{k}_{1:t}(\boldsymbol{x}') \tag{6}$$

where $\boldsymbol{k}_{1:t}(\boldsymbol{x}) = [k(\boldsymbol{x}_1, \boldsymbol{x}), \ldots, k(\boldsymbol{x}_t, \boldsymbol{x})]^\top$, $\boldsymbol{K}_{1:t}$ is the positive definite kernel matrix $[k(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j=1,\ldots,t}$, and $\boldsymbol{I}_t$ is a t by t identity matrix. This posterior distribution can be used to derive predictions for each possible input $\boldsymbol{x}$ on the next time point, which again follow a Gaussian distribution. A key aspect of a $\mathcal{GP}$ is the covariance or kernel function k. The choice of a kernel function corresponds to assumptions about the kind of functions a learner expects. Here, we will use a squared exponential kernel:

$$k_{\text{sqe}}(\boldsymbol{x}, \boldsymbol{x}') = \theta_1^2 \exp\left(-\frac{(\boldsymbol{x} - \boldsymbol{x}')^2}{2\theta_2^2}\right) \tag{7}$$

This kernel induces a universal function learning engine and has been found to describe human function learning well (Griffiths et al., 2009).

Optimizing a function

Given a learned representation of a function at time t, this knowledge needs to be used to choose a next input at time $t+1$. This is done through an acquisition function that takes the expected output for each input and the associated uncertainty to balance exploration and exploitation (Brochu et al., 2010).
An algorithm that is well-poised to cope with the additional requirement to avoid outcomes below a threshold first separates possible inputs into those that are likely to provide outputs above the threshold (the safe set) and those that are not, and then separates this safe set further into a set of maximizers (inputs that are likely to provide the maximum output) and expanders (inputs that are likely to expand the safe set). Following Berkenkamp et al. (2015), we define upper and a lower bounds of a confidence interval as sum of the current expectation m_{t-1} and its attached uncertainty σ_{t-1}.

$$u_t(\boldsymbol{x}) = m_{t-1}(\boldsymbol{x}) + \beta_t \sigma_{t-1}(\boldsymbol{x}) \tag{8}$$
$$l_t(\boldsymbol{x}) = m_{t-1}(\boldsymbol{x}) - \beta_t \sigma_{t-1}(\boldsymbol{x}). \tag{9}$$

the parameter β_t determines the width of the confidence bound, and we set it to $\beta_t = 3$ to assure high safety a priori (i.e. 99.9%). Using these bounds, we can define the safe set as all the input points in the set $\mathcal{X}$ of available inputs that are likely to lead to output values above the safe threshold, $J_{\min}$

$$\mathcal{S}_t = \{\boldsymbol{x} \in \mathcal{X} | l_t(\boldsymbol{x}) \geq J_{\min}\} \tag{10}$$

The set of potential maximizers contains all safe inputs that are likely to obtain the maximum output value; these are the safe inputs for which the upper confidence bound u_t is above the best lower bound:

$$\mathcal{M}_t = \{\boldsymbol{x} \in \mathcal{S}_t | u_t(\boldsymbol{x}) \geq \max_{\boldsymbol{x}' \in \mathcal{X}} l_t(\boldsymbol{x}')\} \tag{11}$$

To find a set of expanders, we define

$$g_t(\boldsymbol{x}) = |\{\boldsymbol{x}' \in \mathcal{X} \setminus \mathcal{S}_t | l_{t,(\boldsymbol{x}, u_t(\boldsymbol{x}))}(\boldsymbol{x}') \geq J_{\min}\}| \tag{12}$$

where $l_{t,(\boldsymbol{x}, u_t(\boldsymbol{x}))}(\boldsymbol{x}')$ is the lower bound of $\boldsymbol{x}'$ based on past data and a predicted outcome for $\boldsymbol{x}$ which provides a new upper bound $u_t(\boldsymbol{x})$. The function is used to determine how many inputs are added to the safe set after choosing input $\boldsymbol{x}$ and observing the output it provides. This function is positive only if the new data point has a non-negligible chance to expand the safe set. The set of possible expanders is then defined as

$$\mathcal{G}_t = \{\boldsymbol{x} \in \mathcal{S}_t | g_t(\boldsymbol{x}) \geq 0\} \tag{13}$$

Normally, the safe optimization routine picks as the next point a safe point that is within the intersection of expanding and maximizing points, but currently shows the highest uncertainty measured by the difference between

the upper and the lower bound. However, for this first investigation of human behavior within safe exploration scenarios we will focus on how participants choices of input points are influenced by simple membership of the 3 sets and if their behavior can be described by more heuristic, stepwise decision behavior.

Experiment 1: Univariate functions

The first experiment required participants to maximize unknown univariate functions $f : x \to y$. On each trial $t = 1, \ldots, 10$ in a block, they could choose an input value $x \in \{0, 0.5, 1, \ldots, 10\}$ to observe (and accrue) an output $y = f(x) + \epsilon$ with noise term $\epsilon \sim \mathcal{N}(0, 1)$. The underlying functions were sampled from a $\mathcal{GP}$ with a squared exponential kernel (l=1, θ=1). The objective was to maximize the sum of the obtained outputs over all trials in a block. A threshold $J_{\min}$ was introduced and a block was ended abruptly if an output below this threshold was obtained. On average, that threshold was fixed to separate 50% of the points into safe and unsafe points. Before the first trial, an initial safe point above the threshold was provided. A screenshot is shown in Figure .

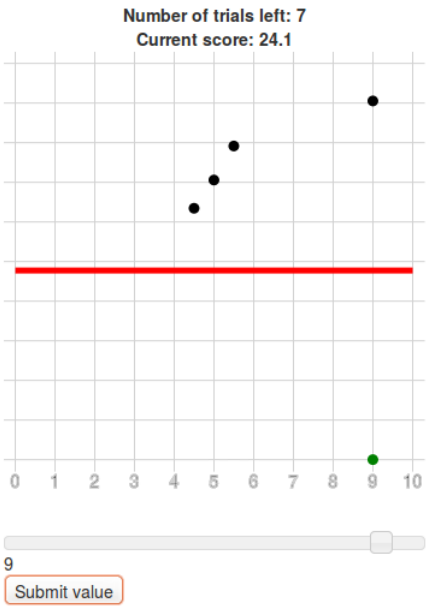

Figure 1: Screenshot of first experiment.

Participants

61 participants (36 female) with an average age of 32.95 (SD = 8.02) were recruited via Amazon Mechanical Turk and received $1 for their participation and a bonus of up to $1, in proportion to their overall score.

Procedure

Participants were told that they had to maximize an unknown function while at the same time trying to avoid sampling below the red line as this would end the current block. After reading the instructions and performing an example task, they had to correctly answer 4 questions to check their understanding, then performed the task, and at the end saw their total score.

Results

As shown in Figure 2, participants obtained outputs higher than expected by chance on the large majority of trials and indeed the average score per participant was significantly higher than chance, $t(60) = 13.311$, $p < 0.01$. In addition, the average number of trials per block

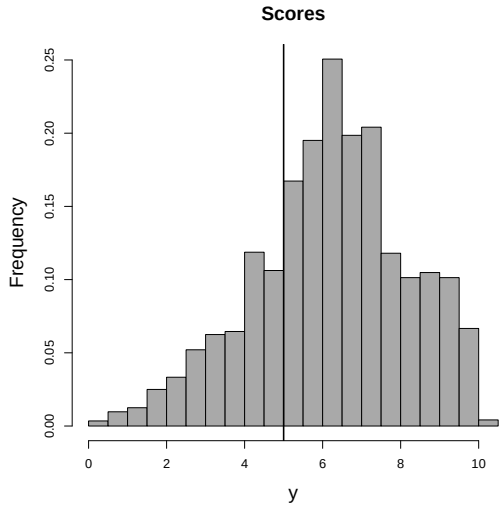

Figure 2: Scores per trial. Black line: chance level.

statistically exceeded what would be expected if participants chose completely at random, $t(548) = 5.1201$, $p < 0.01$ and participants' scores were positively correlated with trials ($r = 0.25$, $p < 0.01$). Taken together, these results indicate that participants learned the task and tended to chose safe inputs.

We used mixed-effects logistic regression analysis to asses which factors influenced participants' choices. The dependent variable was whether each input was chosen or not on each trial for each participant. As predictors, we used indicator variables for membership of an input of the safe, maximization, and expander set. Results indicated that the most plausible model was a model that contains all variables as fixed effects and a participant-specific random intercept, indicating that participants were influenced by set membership in an overall similar fashion. The coefficients of the fixed effects are presented in Table 1 below. Comparing the magnitude of the slopes of the predictors, we can conclude that participants cared about all of the sets, but mostly about

Table 1: Fixed effects estimate. Significant estimates are flagged.

Variable	b	$SE(b)$
Intercept	-4.26^*	0.04
Safe set	1.57^*	0.06
Maximizer set	1.72^*	0.05
Expander set	0.12	0.05

whether or not a point was safe and/or a maximizer. Next, we used a random intercept decision tree analysis (Sela & Simonoff, 2011) to assess whether participants might utilize a simple but effective heuristic strategy that can be implemented as a decision tree, as suggested by Huys et al. (2012). For this, we replaced the indicators of set membership with probability assessments, substituting membership of the maximizer set with the probability of improvement, the safe set with the probability of being above the threshold, and the expander set with the probability of safely expanding the set (assessed through one step ahead forward simulation). The probability of improvement is defined as the probability that an input $\boldsymbol{x}$ produces a higher output than the input $\boldsymbol{x}^+$ that is currently thought to provide the maximum, and can be calculated as

$$PI_t(\boldsymbol{x}) = P\left(f(\boldsymbol{x}) \geq f(\boldsymbol{x}^+)\right) \tag{14}$$

$$= \Phi\left(\frac{m_t(\boldsymbol{x}) - f(\boldsymbol{x}^+)}{\sigma_t(\boldsymbol{x})}\right) \tag{15}$$

where Φ is the cumulative Normal distribution function. Figure 3 depicts the decision tree which best fitted participants' choices. This analysis shows that participants seem to partition the problem into two sub-goals: first they conservatively assess whether or not a point is safe, then they maximize within that safe set. That expanders are not considered within this decision process could be due to the brevity of the task (10 trials) or to risk aversion (i.e., the fear of sampling below the threshold). The non-inclusion of possibly expanding points also

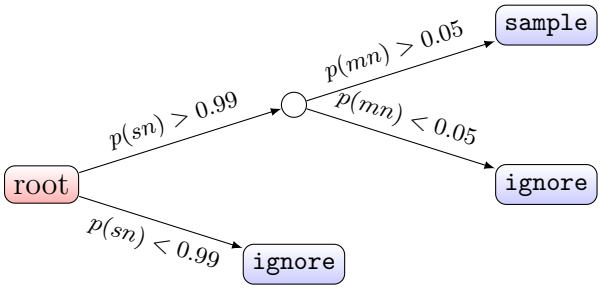

Figure 3: Multi-level decision tree minimizing log-loss.

means that participant only tried to maximize very locally, something that can also be seen when the distance of chosen points to the initially provided input point, $x_{\text{start}} - x_t$ is calculated as shown in Figure 4.

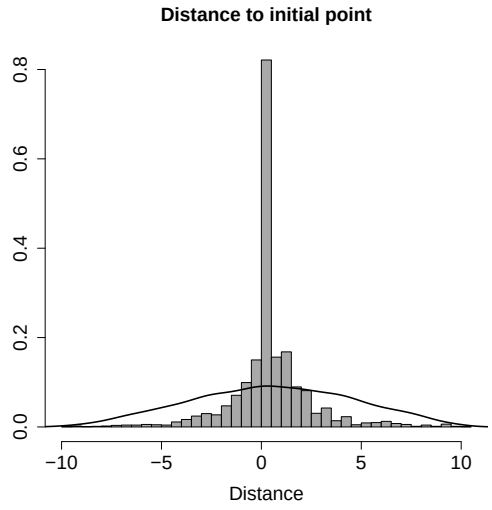

Figure 4: Distance of chosen input points to initially provided point. Black line indicates expected density for sampling at random.

This means that people's behavior in this uni-variate function optimization experiment was based on the attempt of locally maximizing points that they strongly perceived as safe.

Experiment 2: Bivariate functions

In the second experiment, participants were asked to maximize an unknown bivariate function $f : \boldsymbol{x} \rightarrow y$ with $\boldsymbol{x} = (x_1, x_2)^\top$, defined over the grid $x_1, x_2 \in [0, 0.05, 0.1, \ldots, 1]$, with $y = f(\boldsymbol{x}) + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 1)$. As in Experiment 1, the function f was sampled on each block from a $\mathcal{GP}$ with a squared exponential kernel ($l = 2$,$\theta = 1$). The output values y varied between 0 and 100 and one initial point above 50 was provided. We varied the level of risk within-participants: there were 10 blocks in total out of which 5 were "normal", that is unconstrained maximization tasks without a threshold and 5 were "safe" blocks in which obtaining an output below 50 caused the current block to end abruptly. The blocks were presented in randomly permuted order. A screenshot is shown in Figure 5.

Participants

62 participants (37 male), with an average age of 31.77 years (SD = 8.97) were recruited via Amazon Mechanical Turk and received $1 for their participation and a performance-dependent bonus of up to $1. The average completion time of the whole experiment was 11 minutes.

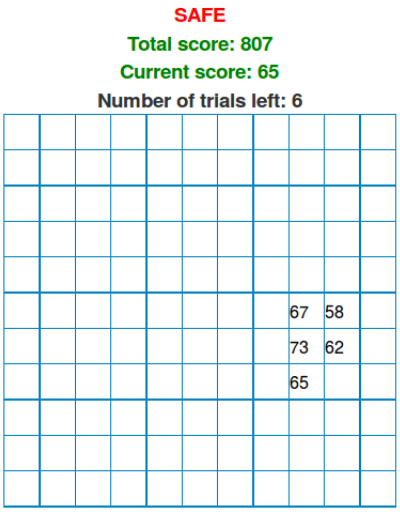

Figure 5: Screenshot of second experiment.

Procedure

The procedure was essentially the same as for Experiment 1, apart from additional detailed instructions regarding the difference between normal unconstrained (without a threshold) and safe (with a threshold) trials.

Results

As shown in Figure 6, participants scored better than expected by chance in both the safe and the normal conditions ($t(\text{normal} = 50) = 24.9$ with $p < 0.01$; $t(\text{safe} = 59) = 9.3$ with $p < 0.01$). The reason why chance level performance is higher in the safe condition is that scores below 50 were not allowed and therefore the output was truncated to be above 50. This time, participants within the safe condition did not complete more trials in a block than expected by randomly choosing inputs on the grid ($t(\text{length} = 5) = -0.32$ with $p = 0.72$). A similar mixed-effects logistic regression analysis as used for Experiment 1 (Table 2) showed that participants seemed to care most about scoring above the threshold in both conditions. As expected, this effect was more pronounced in the safe conditions than in the normal conditions. Still, the presence of this effect in the normal condition is interesting as it did not matter whether or not participants scored below the threshold. If scoring below 50 did not matter, participants should have not cared as much about sampling above this point in the normal condition as they actually did. One explanation for this might be a transfer effect by which participants assume that sampling below 50 is generally bad. The maximizer set only had a small influence on participants' choices that was slightly bigger for the safe condition. Participants showed no tendency to expand the safe set in either of the conditions. This indicates that most chosen inputs were close to the initial safe input and previously chosen inputs. This relatively high risk aversion is understandable, as the bivariate task is more difficult than the univariate one of Experiment 1. Lastly, a random intercept decision tree analysis (Figure 7) showed that in the best fitting model, only the probability of being above the threshold mattered. This indicates that participants only seemed to care about whether or not an input was safe, simplifying the task to a great extent with a strong focus on the probability of losing. Such simplification makes sense in light of the relative complexity of the bivariate task.

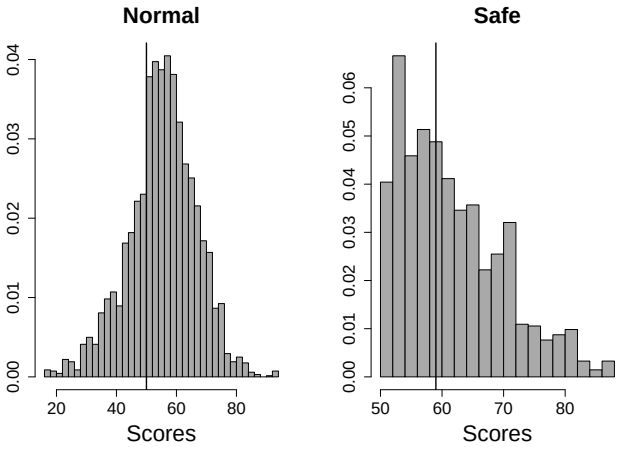

Figure 6: Scores per trial.

Table 2: Fixed effects estimates. Significant estimates are flagged.

Condition	Variable	b	SE(b)
Normal	Intercept	-5.17^*	0.04
	Safe Set	1.35^*	0.04
	Maximizer	0.13^*	0.04
	Expander	0.04	0.05
Safe	Intercept	-5.92^*	0.09
	Safe Set	2.11^*	0.07
	Maximizer	0.23^*	0.09
	Expander	0.03	0.08

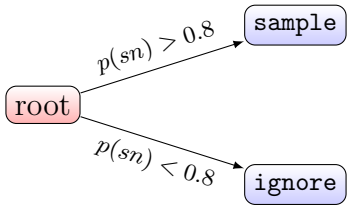

Figure 7: Decision tree minimizing log-loss.

This means that participants again only sampled locally, staying close to the initial point (Figure 8).

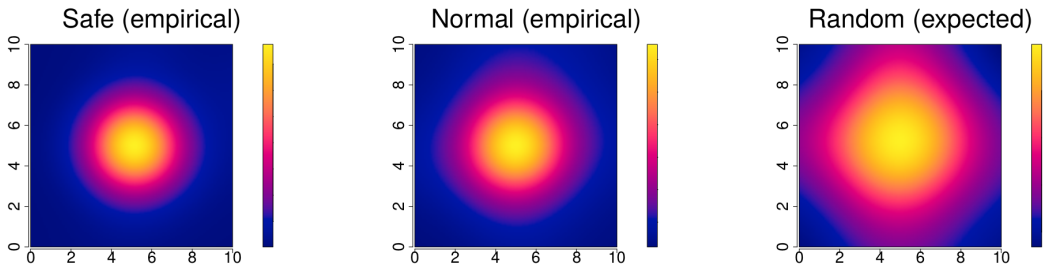

Figure 8: Distance of participants' sample points to initial point for safe condition, normal condition, and the theoretically expected distance for random sampling. Participants stay a little closer to the initial point in the safe condition than in the normal condition. Randomly sampling would theoretically cause much higher distances.

If participants sampled locations are treated as a Poisson process and centred around the initial point, then the posterior density of sampled points shows that participants stayed very close to the initial point in the safe condition, sampled a little further away from the initial point in the normal condition, but never sampled as dispersively as a completely random sampler.

Discussion and Conclusion

Learning unknown functions and exploiting this knowledge to maximise rewards are essential cognitive skills. Such tasks can be formalized as bandit tasks and here we focused on a restricted version thereof where outcomes below a given threshold need to be avoided. We found that participants' behavior was described well by a Gaussian Process safe optimization routine that establishes safe sets and then tries to maximize outputs within these sets. Participants mostly ignored input points that could expand the safe set, shunning risks and maximizing outputs locally, thereby preferring to rather be "safe than sorry".

Participants' behavior was consistent with a sequential heuristic in which they first determined whether inputs were safe and then maximized within this safe set. While this strategy involves only local searches, it can result in truly auspicious behavior, especially when the choices are limited. Participants' focus on avoiding unsafe inputs is consistent with a biological homeostasis maintenance principle that prioritizes not loosing everything over gaining as much as possible. The continued influence of the save threshold on participants' choices in the normal condition, where it had no effect on their potential earnings, might be due to participants generalizing their evaluation of outputs below the threshold as "bad" from the safe conditions.

In future work, we want to focus on what factors drive participants to switch from explorative to safe behavior and in which situations switching constitutes as a normative strategy, for example because it is minimizing costs (Bach, 2015). As we have only focused on functions sampled from a squared exponential kernel here, both for the description of participants' intuitive function learning process and for the actual functions sampled within the task, another direction is to assume different kernel parametrizations of these functions as those lead to diverse theoretical predictions about how fast participants are able to learn (Schulz et al., 2015c). Future work could also extend our approach to active versions of more traditional models such as heuristics and weight-based strategies (Parpart et al., 2015).

Unlike previous work on human behavior in the bandit setting, which has focused on pure optimization primarily, our work explored a relatively novel facet–optimizing risky functions. We expect that this new approach will provide further insights into how people resourcefully optimize outcomes in the real world.

References

Bach, D. R. (2015). Anxiety-like behavioural inhibition is normative under environmental threat-reward correlations. *PLOS Computational Biology*, *11*, e1004646.

Berkenkamp, F., Schoellig, A. P., & Krause, A. (2015). Safe controller optimization for quadrotors with Gaussian Processes. *arXiv preprint arXiv:1509.01066*.

Brochu, E., Cora, V. M., & De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*.

Griffiths, T. L., Lucas, C., Williams, J., & Kalish, M. L. (2009). Modeling human function learning with Gaussian Processes. In *Advances in Neural Information Processing Systems*, (pp. 553–560).

Huys, Q., Eshel, N., ONions, E., Sheridan, L., Dayan, P., & Roiser, J. P. (2012). Bonsai trees in your head: How the Pavlovian system sculpts goal-directed choices by pruning decision trees. *PLoS Computational Biology*, *8*, e1002410.

Huys, Q. J., Lally, N., Faulkner, P., Eshel, N., Seifritz, E., Gershman, S. J., Dayan, P., & Roiser, J. P. (2015). Interplay of approximate planning strategies. *Proceedings of the National Academy of Sciences*, *112*, 3098–3103.

Korn, C. W., & Bach, D. R. (2015). Maintaining homeostasis by decision-making. *PLOS Computational Biology*, *11(5):e1004301*.

Parpart, P., Schulz, E., Speekenbrink, M., & Love, B. C. (2015). Active learning as a means to distinguish among prominent decision strategies. In *Proceedings of the Thirty-Seventh Annual Conference of the Cognitive Science Society*, (pp. 1829–1834).

Rasmussen, C. E. (2006). Gaussian Processes for machine learning.

Schulz, E., Konstantinidis, E., & Speekenbrink, M. (2015a). Exploration-exploitation in a contextual multi-armed Bandit Task. *Proceedings of the 13th International Conference on Cognitive Modeling. Groningen, NL*.

Schulz, E., Konstantinidis, E., & Speekenbrink, M. (2015b). Learning and decisions in contextual multi-armed bandit tasks. *Proceedings of the 37th annual conference of the cognitive science society*, (pp. 2122–2127).

Schulz, E., Tenenbaum, J. B., Reshef, D. N., Speekenbrink, M., & Gershman, S. J. (2015c). Assessing the perceived predictability of functions. *Proceedings of the 37th annual conference of the cognitive science society*, (pp. 2116–2121).

Sela, R., & Simonoff, J. (2011). Reemtree: Regression trees with random effects. *R package version 0.90*, *3*, 741–749.

Srinivas, N., Krause, A., Kakade, S. M., & Seeger, M. (2009). Gaussian Process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*.

Sui, Y., Gotovos, A., Burdick, J. W., & Krause, A. (2015). Safe exploration for optimization with Gaussian Processes. In *International Conference on Machine Learning (ICML)*.