Published in final edited form as:
Proc AAAI Conf Hum Comput Crowdsourc. 2019 ; 7(1): 59–67.

Human Evaluation of Models Built for Interpretability

Isaac Lage[*,1], **Emily Chen**[*,1], **Jeffrey He**[*,1], **Menaka Narayanan**[*,1], **Been Kim**[2], **Samuel J. Gershman**[1], **Finale Doshi-Velez**[1]

[1]Harvard University,

[2]Google

Abstract

Recent years have seen a boom in interest in interpretable machine learning systems built on models that can be understood, at least to some degree, by domain experts. However, exactly what kinds of models are truly human-interpretable remains poorly understood. This work advances our understanding of precisely which factors make models interpretable in the context of decision sets, a specific class of logic-based model. We conduct carefully controlled human-subject experiments in two domains across three tasks based on human-simulatability through which we identify specific types of complexity that affect performance more heavily than others–trends that are consistent across tasks and domains. These results can inform the choice of regularizers during optimization to learn more interpretable models, and their consistency suggests that there may exist common design principles for interpretable machine learning systems.

Introduction

The relatively recent widespread adoption of machine learning systems in real, complex environments has lead to increased attention to anticipating system behavior during deployment. An oft-cited example of unanticipated behavior is described in Caruana et al. (2015), where a classifier trained to predict pneumonia-related complications learned a negative association between asthma and mortality. By careful inspection of the model, the authors determined this serious error occurred because asthmatic patients with pneumonia were treated more aggressively, reducing their mortality rate in both the training and test sets.

Examples such as these have led to a call for the development of models that are not only accurate but also *interpretable*. In particular, human-simulatable models (Lipton 2016) are meant to be simple enough for a consumer of the model to step through the calculations in a reasonable amount of time. A particularly appealing property of human-simulatable models is that they enable an expert to combine their human computation and reasoning with the model's logic; for example, they can catch and adjust for situations in which the model's decision process relies on irrelevant features or discriminates against protected groups. Examples of human-simulatable models include human-simulatable regression functions

isaaclage@g.harvard.edu .

[*] Equal contribution

(e.g. Caruana et al., 2015; Ustun and Rudin, 2016), and various logic-based methods (e.g. Wang and Rudin, 2015; Lakkaraju, Bach, and Leskovec, 2016; Singh, Ribeiro, and Guestrin, 2016).

But even types of models designed to be human-simulatable may cease to be so when they are too complex. For example, a decision set—a model mapping inputs to outputs through a set of independent, logical rules—with 5,000 rules will be much harder for a human to simulate than a decision set with 5 rules. Such simplicity can be achieved by adding a term to the objective function designed to limit the model's complexity—we call these *regularizers*.

The challenge is determining which types of complexity affect human-simulatability and by how much, since this will guide the choice of regularizers for interpretability. For example, decision sets—the model we focus on in this work—can be regularized in many ways, including limiting the number of rules, limiting the number of terms in a rule, and limiting the overlap between different rules (Lakkaraju, Bach, and Leskovec 2016). While each of these likely increases the human-simulatability of the model to some degree, over-regularization potentially comes at the cost of predictive performance: a very simple model may not be able to model the data accurately. Because of this, we want to employ only enough regularization during optimization to help us reach our goal. The first question we address in this work is therefore understanding which types of complexity have the greatest effect on human-simulatability.

A natural follow-on question is whether the types of complexity that matter depend on the context, or whether there exist general design principles for human-simulatability. In the broader field of interpretable machine learning, studies to date have shown conflicting results. For example, Poursabzi-Sangdeh et al. (2017) find that longer models are more difficult to simulate, while Allahyari and Lavesson (2011); Elomaa (2017); Kulesza et al. (2013) find that larger models can be more interpretable. It is unclear whether these differences in conclusions are due to different notions of interpretability, different choices of models, differences in tasks, or differences in domains. The second question we address in this work is whether there are consistent underlying principles that can be used to develop interpretable models.

Since the question of how different types of complexity affect human-simulatability ultimately depends on people's ability to replicate the model's computations, we address it through a series of careful human-subjects experiments on Amazon Mechanical Turk. We test the effects of three different types of decision-set-complexity across two domains and three tasks. Our results show a clear and consistent ranking of the types of complexity across contexts. Introducing intermediate concepts, or cognitive chunks, has a larger impact on human-simulatability than increasing model size, which in turn has a larger impact than repeating terms. This gives us insight into how to employ regularizers when learning decision sets to achieve human-simulatability, and suggests that there may exist common design principles for how to create human-simulatable machine learning systems.

Related Work

Interpretable Machine Learning.

Interpretable machine learning methods aim to optimize models to be predictive and understandable to humans. Lipton (2016) classifies properties of interpretable models into those measuring the transparency of a model, and those corresponding to how well a black-box model can be interpreted after learning (post-hoc). He then argues that the strictest notion of transparency is simulatability, which corresponds to whether a human can walk through the model's computations to arrive at the output from the input in a reasonable amount of time. This goal of human-simulatability can be achieved through choosing an interpretable model class (e.g. Caruana et al., 2015; Ustun and Rudin, 2016; Wang and Rudin, 2015; Lakkaraju, Bach, and Leskovec, 2016), or by adding regularizers to the optimization objective designed to facilitate interpretability (e.g. Wu et al., 2018; Plumb et al., 2019; Ross and Doshi-Velez, 2018; Lakkaraju, Bach, and Leskovec, 2016). Many solutions draw from both approaches since Lipton (2016) argues that even inherently interpretable models can be difficult to simulate without some regularization. In this work, we focus on characterizing the effect of different types of complexity on human-simulatability for an interpretable class of logic-based models: decision sets (Lakkaraju, Bach, and Leskovec 2016). This is in contrast to many of the works above that either do not empirically justify the interpretability of their approach through human-evaluation (e.g. Ustun and Rudin, 2016; Plumb et al., 2019) or that run user studies comparing their approach to baselines, but do not attempt to yield generalizable insights (e.g. Lakkaraju, Bach, and Leskovec, 2016).

Domain Specific Human Factors for Interpretable ML.

Specific application areas have also evaluated the desired properties of interpretable machine learning systems within the context of the application. For example, Tintarev and Masthoff (2015) provide a survey in the context of recommendation systems, noting differences between the kind of system explanations that manipulate trust and the kind that increase the odds of a good decision. These studies often examine whether the system explanation(s) have an effect on performance on a downstream task. Horsky et al. (2012) describe how presenting the right clinical data alongside a decision support recommendation can help with adoption and trust. Bussone, Stumpf, and O'Sullivan (2015) found that overly detailed explanations from clinical decision support systems enhance trust but also create over-reliance; short or absent explanations prevent over-reliance but decrease trust. These studies span a variety of extrinsic tasks, and given the specificity of each explanation type, domain and task, identifying generalizable properties is challenging.

General Human Factors for Interpretable ML.

Closer to the objectives of this work, Kulesza et al. (2013) performed a qualitative study in which they varied the soundness and the completeness of an explanation of a recommendation system. They found completeness was important for participants to build accurate mental models of the system. Allahyari and Lavesson (2011); Elomaa (2017) also find that larger models can sometimes be more interpretable. Schmid et al. (2016) find that human-recognizable intermediate predicates in inductive knowledge programs can sometimes improve simulation time. Poursabzi-Sangdeh et al. (2017) manipulate the size and transparency of a model and find that longer models and black-box models are harder to simulate accurately on a real-world application predicting housing prices. Our work fits into this category of empirical study of interpretable model evaluation and extends it in two key ways. We consider logic-based models—a major category of interpretable models where this question has not been well explored, and we consider several different tasks and domains, allowing us to demonstrate that our results are generalizable to new machine learning problems.

Research Questions

Our central research question is to empirically determine which types of complexity most affect human-simulatability, and whether this depends on context, to inform the choice of regularizers for learning interpretable models. In this section, we first describe the class of models and the specific types of complexity we study, then the set of tasks we ask participants to complete, and the domains in which we run our experiments. See Table 1 for an overview.

Model

In this work, we study *decision sets* (also known as rule sets). These are logic-based models–a frequently studied category in the interpretability literature (e.g. Subramanian et al., 1992; Huysmans et al., 2011), that have been demonstrated to be more interpretable than other forms of logic-based models. Specifically, Lakkaraju, Bach, and Leskovec (2016) found that subjects are faster and more accurate at describing local decision boundaries of decision sets than rule lists. There exist a variety of techniques for learning decision sets from data (e.g. Frank and Witten, 1998; Cohen, 1995; Clark and Boswell, 1991; Lakkaraju, Bach, and Leskovec, 2016). Additionally, they can be used as post-hoc explanations of black-box models like neural-networks (Ribeiro, Singh, and Guestrin, 2016; Lakkaraju et al., 2017).

Formally, decision sets consist of a collection of logical rules, each mapping some function of the inputs to a collection of outputs. For example, the box titled: “The Alien’s Preferences” in Figure 1 shows a decision set where each line contains a clause in disjunctive normal form (an or-ofands) of the inputs (blue words), which, if true, maps to the output (orange words–also in disjunctive normal form).

Types of Complexity

In order to learn interpretable decision sets, Lakkaraju, Bach, and Leskovec (2016) added several regularization terms to explicitly encourage interpretability. The existence of multiple types of complexity that can be used as interpretability-regularizers for this model class motivates the work we do to determine which ones most affect human-simulatability. In our experiments, we considered three types of complexity—model size, the introduction of new cognitive chunks, and variable repetitions. These are supported by insights from the psychology and interpretability literature (Freitas, 2014; Feldman, 2000; Gottwald and Garner, 1972; Miller, 1956; Treisman and Gelade, 1980), and were found in pilot studies to have larger effects than reasonable alternatives. See our tech report for details (Lage et al. 2019).

V1: Model Size.—Model size, measured in a variety of ways, is often considered as a factor that affects interpretability. Freitas (2014) reviews the large amount of work that has defined interpretability as a function of model size and argues that other interpretability metrics should be considered, but this one is clearly important. Results from psychology support this idea. Feldman (2000) found that human concept learning correlates with the boolean complexity of the concept, suggesting that it will take people longer to process decision sets with more complex logical structures.

To capture the notion of model size, we varied the size of the model across two dimensions: the *total number of lines* in the decision set, and the *number of terms within the output clause*. The first corresponds to increasing the vertical size of the model—the number of rules—while the second corresponds to increasing the horizontal size of the model—the number of terms in each rule. We focused on output clauses because they were harder to parse: input clauses could be quickly scanned for relevant terms, but output clauses had to be processed completely to determine the correct answer.

V2: Cognitive Chunks.—Adding intermediate terms to the model may facilitate the task by allowing people to remember 1 concept instead of a conjunction or disjunction of concepts. Gottwald and Garner (1972) found that people classify objects that require processing 2 dimensions more slowly than those that require processing a single dimension. Miller (1956) argued that humans can hold about seven items simultaneously in working memory, suggesting that human-interpretable explanations should obey some kind of capacity limit, but that these items can correspond to complex *cognitive chunks*—for example, 'CIAFBINSA' is easier to remember when it is recoded as the more meaningful units 'CIA', 'FBI', 'NSA.' On the machine learning side, a large amount of work has been done on representation learning under the assumption that learning intermediate representations can simplify prediction tasks e.g. Chen et al. (2016).

To explore this type of complexity, we varied the *number of cognitive chunks*, and whether they were *implicitly or explicitly* defined. We define a cognitive chunk as a clause in disjunctive normal form of the inputs that may recur throughout the decision set. Explicit cognitive chunks are mapped to a name that is then used to reference the chunk throughout the decision set, while implicit cognitive chunks recur throughout the decision set without ever being explicitly named. See Figure 2 for an example.

V3: Repeated Terms.—Increasing the number of times that relevant features (i.e. those appearing in the input example) appear in the model may increase the difficulty of the tasks since people will likely need to scan through each of the lines where a relevant feature appears to answer the question. Treisman and Gelade (1980) discusses the feature-integration theory of attention in psychology that says that people can find items that are different in exactly 1 dimensions in parallel time, but that they must linearly scan all relevant items when the one in question is different only in a conjunction of features. Repeated terms was also measured indirectly by Lakkaraju, Bach, and Leskovec (2016) through the amount of overlap between different decision rules.

To capture this notion, we varied the *number of variable repetitions*—the number of times that features in the input example were repeated in the decision set.

Tasks

An interpretable model can facilitate many downstream tasks including improving safety and increasing trust. We address our second research question: whether there are consistent underlying principles that can be used to develop interpretable models, by considering 3 distinct tasks in order to investigate whether our results are consistent across them. All 3 tasks are simulation-based, and are designed to test how well humans can use models to accomplish goals suggested by the interpretability literature. Since they measure how well people can answer questions by simulating the model's predictions, they do not require people to bring in outside knowledge, allowing us to rigorously control for participants' preexisting knowledge and assumptions.

Simulation.—In this task, we asked people to predict the system's recommendation given a model and an input example. The task requires participants to step through the model's computations with the example in order to produce the same prediction as the model. This task directly measures the notion of human-simulatability described by Lipton (2016). Additionally, this was used as a measure of interpretability in Poursabzi-Sangdeh et al. (2017). See Figure 1 for a screenshot of the simulation task.

Verification.—In this task, we asked participants to verify whether the system's prediction is consistent with a recommended prediction, given a model and an input example. This task measures how well people can verify whether the prediction of a machine learning system corresponds to a suggested prediction. This corresponds to a scenario where the user may want to verify that the prediction given by the machine learning system matches their expectation, and when it does not, to understand why. See Lage et al. (2019) for a screenshot of the verification task.

Counterfactual.—In this task, we asked participants to determine whether the correctness of a recommended prediction would change if one of the features in the input example were replaced with a different feature. This task measures how well people can detect a change in the prediction based on a small change in one of the features of the input example. This task addresses the question of using explanations of machine learning systems to understand why an outcome happened for a specific example in question and not other examples. Miller (2019) argues, based on insights from the social sciences, that effective explanations involve this kind of contrastive reasoning. More concretely, this task corresponds to a setting where a user wants to verify whether a model is making predictions fairly by checking whether the same prediction would have been made for a person of a different race or gender. See Lage et al. (2019) for a screenshot of the counterfactual task.

Domains—A second source of context that may influence what makes a model interpretable is the domain. We further investigate our second research question of whether there are consistent underlying principles that can be used to develop interpretable models by running the same experiments in two different domains: a low-risk recipe domain, and a high-risk clinical domain. These two different settings were designed to elicit very different impressions of the stakes of the machine learning model's prediction; a mistake in the recipe domain is only unpleasant, while a mistake in the clinical domain could be fatal. We used aliens as the object of study in both domains, which allowed us to control for peoples' priors about what makes a good recipe or prescription.

Recipe.—Study participants were told that the machine learning system had studied a group of aliens and determined each of their individual food preferences in various settings (e.g., weekend or laughing). This resembles a product recommendation setting where the participants are customers wishing to inspect a recommendation given by the system. Here, the inputs are settings, the outputs are groups of food, and the recommendations are specific foods.

Clinical.—Study participants were told that the machine learning system had studied a group of aliens and determined personalized treatment strategies for various symptoms (e.g., sore throat). This resembles a clinical decision support setting where the participants are doctors inspecting the treatment suggested by the system. Here, the inputs are symptoms, the outputs are classes of drugs, and the recommendations are specific drugs. We chose drug names that start with the first letter of the drug class (e.g., antibiotics were Aerove, Adenon and Athoxin) to replicate the level of ease and familiarity of food names.

Methods

Conditions

For each type of complexity, we created a set of conditions based on the experimental levels defined as follows:

V1: Model Size.—We manipulated the length of the decision set (varying between 2, 5, and 10 lines) and the length of the output clause (varying between 2 and 5 terms).

V2: Cognitive Chunks.—We manipulated the number of cognitive chunks introduced (varying between 1, 3 and 5), and whether they were embedded into the decision set (implicit) or abstracted out into new cognitive chunks and later referenced by name (explicit).

V3: Repeated Terms.—We manipulated the number of times the input conditions appeared in the decision set (varying between 2, 3, 4 and 5).

For each type of complexity, we hand-generated one decision set for each combination of the levels. For example, we created a decision set with 5 lines and 2 output terms for V1, and a decision set with 3 implicit cognitive chunks for V2. We used each decision set 3 times—once with each task—with the terms randomly permuted to prevent memorization. The logical structure of all the conditions were identical across the 2 domains. See Lage et al. (2019) for additional details about the conditions.

Metrics

We considered 3 metrics. *Response time* was measured as the number of seconds from when the task was displayed until the subject hit the submit button on the interface. *Accuracy* was measured as whether the subject correctly identified output consistency for verification questions, the presence or absence of a change in recommendation correctness under the perturbation for counterfactual questions, and any combination of correct categories for simulation questions. *Subjective Difficulty of Use* was measured on a 5-point Likert scale. After submitting their answer for each question, but before being told if their answer was correct, the participant was asked to subjectively rate how difficult it was to use the model to answer the question on a scale from 1 to 5 where 1 was very easy to use, 3 was neutral and 5 was very hard to use.

Procedure

We ran an experiment for each of the 3 types of complexity in each of the 2 domains for a total of 6 experiments. Each experiment consisted of the conditions defined in the Conditions subsection tested once each for 18 trials in V1 and V2 and 15 trials in V3 (the different types of complexity have different numbers of levels associated with them).

Each participant took exactly one experiment. Thus, the tasks and levels of the type of complexity were within-subjects variables. Keeping the latter within-subjects helped us reduce variance due to the abilities of each subject for the key comparisons in our results. The domain and type of complexity were kept as between-subjects variables because their inclusion in a single experiment would have resulted in a prohibitively long study for Amazon Mechanical Turk.

The question order was block-randomized for every participant: participants were always shown a verification, then a simulation, then a counterfactual question, but which levels they came from was randomly determined. This allowed us to reduce variance due to learning effects, had our analysis been dominated by subjects who randomly ended up with all of the simulation questions at the beginning, for example.

Participants

We recruited 150 subjects for each of our 6 experiments through Amazon Mechanical Turk (900 subjects all together) with each experiment posted successively between July and December 2018.[1] Participants were paid $3 for completing the study and experiments took 19.42 minutes on average. This study was approved by our institution's IRB.

Participants were told that their primary goal was accuracy, and their secondary goal was speed. This emulates a scenario where the user must make correct decisions about whether to trust the system and speed is only secondarily important to increase efficiency. Additionally, our pilot studies (see Lage et al. (2019)) found that when participants were instructed to answer quickly or under time pressure, they preferred to guess rather than

[1] We argue that recruiting subjects in this manner is valid since we do not make direct comparisons between experiments.

risk not answering. They were instructed that each question involved a different alien to discourage them from generalizing between questions.

A major challenge with recruiting participants from Amazon Mechanical Turk is that it is typically assumed that a user would first have a reasonable amount of training to use the system (see e.g. Lakkaraju, Bach, and Leskovec, 2016; Wu et al., 2018 who assume the users of interpretable systems are domain experts). Doing such an extensive training was not realistic in this setting, so we used the following strategy: First, participants were given a tutorial on each task and the interface. Next, participants were given a set of three practice questions, one drawn from each task. If they answered these correctly, they could move directly to the experiment, and otherwise they were given an additional set of three practice questions. We then excluded participants from the analysis who did not get all of one of the two sets of practice questions correct. This is similar to Hutton, Liu, and Martin (2012), who filtered participants in a crowd-sourced evaluation of classifier interpretability who got many questions wrong. While it is possible that those we excluded could have learned the task with more training, we used this process to filter for participants who already had a basic competency in manipulating logic structures. (Not doing such an exclusion would have added noise from participants for whom decision sets were already extremely foreign.)

Finally, we excluded an additional 6 participants who took more than 5 minutes to answer a single question under the assumption that they got distracted during the experiment. Table 2 describes the total number of participants that remained in each experiment out of the original 150 participants. A statistical analysis determined that the populations of participants included in the final results were not significantly different between the domains. See Lage et al. (2019).

Most participants were from the US or Canada, were less than 50 years old and held a Bachelor’s degree. This may bias our results to people of this specific background. See Lage et al. (2019) for a more detailed breakdown.

Experimental Interface

Figure 1 shows our interface for the simulation task in V3 in the clinical domain. The *observations* section refers to the input example. The *preferences* section contains a description of the model’s internal prediction logic. Finally, the *disease medications* section contains a dictionary defining *concepts* relevant to the experiment (for example, which medications are antibiotics). The interface is identical in the recipe domain with the medications replaced with food and the medical observations replaced with non-medical settings (e.g. weekend, laughing). The verification and counterfactual questions contain an additional *recommendation* section below the *observations* section that contains a suggested prediction. The verification question asks the user whether the suggested prescription treats the alien symptoms in the medical domain, or whether the alien is satisfied with the suggested meal in the recipe domain. The counterfactual question additionally includes a replacement for one of the features in the *observations* section (for example, what if ‘hoarse’ were replaced with ‘anxious’), and asks whether the effectiveness of the treatment or the alien’s satisfaction with its meal would change. The choice of location for all elements was chosen based on pilot studies. See Lage et al. (2019) for a description of these and additional details about the interfaces for the verification and counterfactual tasks.

Analysis

We computed linear regressions for the continuous outputs (response time, subjective difficulty of use) and logistic regressions for binary output (accuracy) to estimate the effect of each type of complexity and task on the outcome variable. We report p-values as significant that are less than $\alpha = 0.05$ after a Bonferroni multiple comparisons correction across all tests of all experiments. Response time for each condition was computed only for subjects who correctly answered the question. Response time and subjective difficulty of use were normalized across participants by subtracting the participant-specific mean. If an experiment had more than one independent variable—e.g., number of lines and terms in output—we performed one regression with both variables. We included whether the task was a verification or a counterfactual question as binary variables that should be interpreted with respect to the simulation task. Regressions were performed with the statsmodels library (Seabold and Perktold 2010) and included an intercept term.

Results

We report the results of the statistical analysis in Table 3 with significant p-values highlighted in bold, and we visualize the results from the recipe domain in Figure 3. See Lage et al. (2019) for a visualization of the results from the clinical domain. Our main findings are: greater complexity generally results in longer response times; adding cognitive chunks had the clearest impact, while the effect of model size was less clear, and repeated terms had no obvious effect; the results are consistent across domains and tasks.

Greater complexity results in longer response times, with the most marked effects for cognitive chunks, followed by model size, then number of variable repetitions.

Unsurprisingly, adding complexity generally increases response times, suggesting that regularizing these types of complexity can be effective for increasing interpretability. In Figure 3, we see that increasing the number of lines, the number of terms within a line, adding new cognitive chunks, and repeating variables all show trends towards increasing response time. Table 3 reports which of these trends are statistically significant: the number of cognitive chunks and whether these are implicitly embedded in the decision set or explicitly defined had a statistically significant effect on response time in both domains, the number of lines and the number of output terms had a statistically significant effect on response time only in the recipe domain, and the number of repeated variables did not have a statistically significant effect in either domain.

The magnitude of the increase in response time also varies across these factors. (Note that the y-axes in Figure 3 all have the same scale for easy comparison.) Introducing new cognitive chunks can result in overall increases in response time on the order of 20 seconds, whereas increases in length have effects on the order of 10 seconds. Increases in variable repetition has an effect of at most a few second.

Finally, we found that participants had significantly longer response times when new cognitive chunks were made explicit rather than implicitly embedded in a line. This ran counter to our expectations since users had to process fewer but longer lines with implicit cognitive chunks compared to the same chunks defined explicitly.

Consistency across domains: Magnitudes of effects change, but trends stay the same.

In all experiments, the general trends are consistent across both the recipe and clinical domains. Sometimes an effect is weaker or unclear, but never is an effect clearly reversed. There were 21 cases of factors that had a statistically significant effect on a dependent variable in at least 1 of the 2 domains. For 19 of those, the 95% confidence interval of both domains had the same sign (i.e., the entire 95% confidence interval was positive for both domains or negative for both domains). For the other 2 (the effect of verification questions on accuracy and response time for experiment V1), one of the domains (clinical) was inconclusive (interval overlaps zero).

Consistency across tasks: Relative trends stay the same, different absolute values.

The effects of different types of complexity on response time were also consistent across tasks. That said, actual response times varied significantly between tasks. In Figure 3, we see that the response times for simulation questions are consistently low, and the response times for counterfactual questions are consistently high (statistically significant across all experiments except V2 in the Recipe domain). Response times for verification questions are generally in between, and often statistically significantly higher than the easiest setting of simulation.

Consistency across metrics: subjective difficulty of use follows response time, less clear trends in accuracy.

So far, we have focused our discussion on response time. In Table 3, we see that subjective difficulty of use largely replicates the findings of response time. We see that simulation questions are significantly easier for participants to answer than counterfactuals. We also see a statistically significant effect of the number of cognitive chunks, model length, and number of output terms. The finding that implicit cognitive chunks are less difficult to use than explicit cognitive chunks appeared only in the recipe domain.

Unlike response time and subjective difficulty of use, where the trends were significant and consistent, the effect of different types of complexity on accuracy was less clear. None of the effects were statistically significant, and there are no clear trends. We believe that this was because we asked participants to be accurate primarily and fast secondarily, effectively pushing any effects into response time. But even when participants were coached to be accurate, some tasks proved harder than others: counterfactual tasks had significantly lower accuracies than simulation tasks.

Discussion

Observation: Consistent patterns provide guidance for the design of interpretable machine learning systems.

In this work, we found consistent patterns across tasks, and domains for the effect of different types of decision set complexity on human-simulatability. These patterns suggest that, for decision sets, the introduction of new cognitive chunks or abstractions had the greatest effect on response time, then model size (overall length or length of line), and finally there was relatively little effect due to variable repetition. These patterns are interesting because machine learning researchers have focused on making decision set lines orthogonal (e.g. Lakkaraju, Bach, and Leskovec, 2016), which is equivalent to regularizing the number of variable repetitions, but perhaps, based on these results, efforts should be more focused on length and if and how new concepts are introduced. This knowledge can help us expand the faithfulness of the model to what it is describing with minimal sacrifices in human ability to process it.

Observation: Consistency of the results across tasks and metrics suggests studies of interpretability can be conducted with simplified tasks at a lower cost.

While the relative ordering of the types of complexity was the same for both the simulation and counterfactual tasks, the counterfactual tasks were more difficult for people to answer correctly. This suggests that we may be able to simplify human-subject studies by using simpler tasks that capture the relevant notions of interpretability. To measure human-simulatability, for example, it seems that the simulation task can be used in future experiments to obtain the same results at a lower cost. A second possibility for simplifying tasks is to rely on people's subjective evaluations of interpretability. In our results, the correlation between peoples' subjective evaluation of difficulty and the more objective measure of response time suggests that this can be done. However this should be followed up since it may be an artifact of running this study on Amazon Mechanical Turk where faster response times correspond to higher pay rates.

Future Work: Using Amazon Mechanical Turk to evaluate interpretability requires additional study.

While simplifying tasks is one possible way to make interpretability studies on Amazon Mechanical Turk more effective, finding other ways to do so remains an open question. In our experiments, the criteria we used to exclude participants who were not able to complete the tasks effectively at the beginning of the experiment excluded over half of the participants. This is likely because the models and tasks were challenging for laypeople since they were designed with expert users in mind. Whether and how Amazon Mechanical Turk studies should be used to evaluate notions of interpretability associated with domain experts warrants future study.

Future Work: The unexpected preference for implicit cognitive chunks should be unpacked in future experiments.

An unexpected finding of our study that warrants further investigation is that implicit cognitive chunks were easier for people to process than explicit ones. This could be because explicitly instantiating new concepts made the decision set harder to scan, or because people prefer to resolve the answer in one long line, rather than two short ones. Follow-up studies should investigate whether this finding persists when the familiarity of the concept or the number of times it is used within the decision set increases. These insights could guide representation learning efforts.

Limitations.

There are several areas of our experiment design that future studies could explore further. Whether interfaces for interpretable machine learning models can be optimized to present the same information more effectively is an open question. In this project, we fixed ours to something reasonable based on pilot studies, but this warrants further study. Future studies measuring interpretability with humans should also explore whether the results of this study generalize to other classes of models–logic-based or otherwise, as well as whether certain model classes are inherently more interpretable or are preferred in certain cases. Finally, while this work relied on a set of synthetic tasks building on the notion of human-simulatability, future work will need to connect performance on these basic tasks to real-world tasks, like finding errors or deciding whether to trust a model, that are more difficult to run in controlled settings.

Conclusion

In this work, we investigated the effect of different types of complexity for decisions sets on the interpretability of the models, measured through three different tasks based on human-simulatability. Our results suggest that the number of cognitive chunks, then the model size are the most important types of complexity to regularize in order to learn interpretable models, while the number of variable repetitions has relatively little effect. These results are consistent across tasks and domains, which suggests that there are general design principles that can be used to guide the development of interpretable machine learning systems.

Acknowledgments

The authors acknowledge PAIR at Google, a Google Faculty Research Award, and the Harvard Berkman Klein Center for their support. IL is supported by NIH 5T32LM012411-02.

References

Allahyari H, and Lavesson N 2011. User-oriented assessment of classification model understandability. In 11th scandinavian conference on Artificial intelligence. IOS Press.

Bussone A; Stumpf S; and O'Sullivan D 2015. The role of explanations on trust and reliance in clinical decision support systems. In Healthcare Informatics (ICHI), 2015 International Conference on, 160–169. IEEE.

Caruana R; Lou Y; Gehrke J; Koch P; Sturm M; and Elhadad N 2015. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In Proceedings of the 21th

ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1721–1730. ACM.

Chen X; Duan Y; Houthooft R; Schulman J; Sutskever I; and Abbeel P 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In Advances in Neural Information Processing Systems 29. Curran Associates, Inc. 2172–2180.

Clark P, and Boswell R 1991. Rule induction with cn2: Some recent improvements. In European Working Session on Learning, 151–163. Springer.

Cohen WW 1995. Fast effective rule induction. In Proceedings of the twelfth international conference on machine learning, 115–123.

Elomaa T 2017. In defense of c4. 5: Notes on learning one-level decision trees. ML-94 254:62.

Feldman J 2000. Minimization of boolean complexity in human concept learning. Nature 407(6804):630–633. [PubMed: 11034211]

Frank E, and Witten IH 1998. Generating accurate rule sets without global optimization. In Proceedings of the Fifteenth International Conference on Machine Learning, ICML ‘98, 144–151. Morgan Kaufmann Publishers Inc.

Freitas AA 2014. Comprehensible classification models: a position paper. ACM SIGKDD explorations newsletter 15(1):1–10.

Gottwald RL, and Garner WR 1972. Effects of focusing strategy on speeded classification with grouping, filtering, and condensation tasks. Perception and Psychophysics 179–182.

Horsky J; Schiff GD; Johnston D; Mercincavage L; Bell D; and Middleton B 2012. Interface design principles for usable decision support: a targeted review of best practices for clinical prescribing interventions. Journal of biomedical informatics 45(6):1202–1216. [PubMed: 22995208]

Hutton A; Liu A; and Martin C 2012. Crowdsourcing evaluations of classifier interpretability. In AAAI Spring Symposium Series. AAAI Press.

Huysmans J; Dejaeger K; Mues C; Vanthienen J; and Baesens B 2011. An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. Decision Support Systems.

Kulesza T; Stumpf S; Burnett M; Yang S; Kwan I; and Wong W-K 2013. Too much, too little, or just right? ways explanations impact end users’ mental models. In Visual Languages and Human-Centric Computing (VL/HCC), 2013 IEEE Symposium on, 3–10. IEEE.

Lage I; Chen E; He J; Narayanan M; Kim B; Gershman S; and Doshi-Velez F 2019. An evaluation of the human-interpretability of explanation. arxiv abs/1902.00006.

Lakkaraju H; Bach SH; and Leskovec J 2016. Interpretable decision sets: A joint framework for description and prediction. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1675–1684. ACM.

Lakkaraju H; Kamar E; Caruana R; and Leskovec J 2017. Interpretable & explorable approximations of black box models. arxiv abs/1707.01154.

Lipton ZC 2016. The mythos of model interpretability. In ICML 2016 Workshop on Human Interpretability in Machine Learning.

Miller GA 1956. The magical number seven, plus or minus two: some limits on our capacity for processing information. Psychological review 63(2):81. [PubMed: 13310704]

Miller T 2019. Explanation in artificial intelligence: Insights from the social sciences. Artificial Intelligence 267:1–38.

Plumb G; Al-Shedivat M; Xing E; and Talwalkar A 2019. Regularizing black-box models for improved interpretability. arxiv abs/1902.06787.

Poursabzi-Sangdeh F; Goldstein DG; Hofman JM; Vaughan JW; and Wallach H 2017. Manipulating and measuring model interpretability. In NIPS Workshop on Transparent and Interpretable Machine Learning in Safety Critical Environments.

Ribeiro MT; Singh S; and Guestrin C 2016. Why should i trust you?: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135–1144. ACM.

Ross A, and Doshi-Velez F 2018. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In AAAI Conference on Artificial Intelligence. AAAI Press.

Schmid U; Zeller C; Besold T; Tamaddoni-Nezhad A; and Muggleton S 2016. How does predicate invention affect human comprehensibility? In International Conference on Inductive Logic Programming, 52–67. Springer.

Seabold S, and Perktold J 2010. Statsmodels: Econometric and statistical modeling with python. In Proceedings of the 9th Python in Science Conference, volume 57, 61.

Singh S; Ribeiro MT; and Guestrin C 2016. Programs as black-box explanations. arXiv preprint arXiv:1611.07579.

Subramanian GH; Nosek J; Raghunathan SP; and Kanitkar SS 1992. A comparison of the decision table and tree. Communications of the ACM 35(1):89–94.

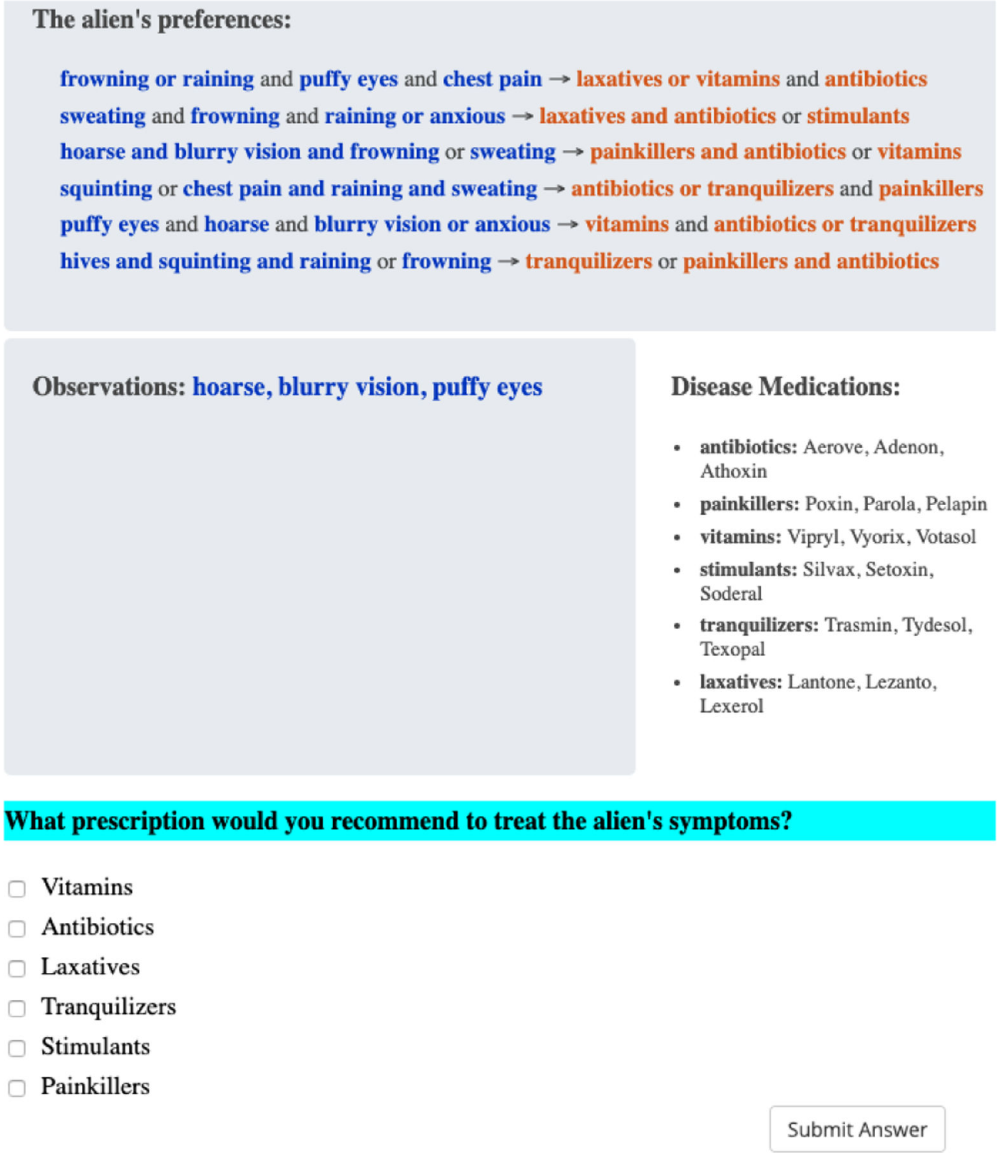
Tintarev N, and Masthoff J 2015. Explaining recommendations: Design and evaluation. In Recommender Systems Handbook. Springer. 353–382.

Treisman AM, and Gelade G 1980. A feature-integration theory of attention. Cognitive Psychology 97–136. [PubMed: 7351125]

Ustun B, and Rudin C 2016. Supersparse linear integer models for optimized medical scoring systems. Machine Learning 102(3):349–391.

Wang F, and Rudin C 2015. Falling rule lists. In Artificial Intelligence and Statistics, 1013–1022.

Wu M; Hughes M; Parbhoo S; Zazzi M; Roth V; and Doshi-Velez F 2018. Beyond sparsity: Tree regularization of deep models for interpretability. In AAAI Conference on Artificial Intelligence. AAAI Press.



The alien's preferences:

frowning or raining and **puffy eyes** and **chest pain** → **laxatives or vitamins** and **antibiotics**
sweating and **frowning** and **raining or anxious** → **laxatives and antibiotics** or **stimulants**
hoarse and blurry vision and frowning or **sweating** → **painkillers and antibiotics** or **vitamins**
squinting or **chest pain and raining and sweating** → **antibiotics or tranquilizers** and **painkillers**
puffy eyes and **hoarse** and **blurry vision or anxious** → **vitamins** and **antibiotics or tranquilizers**
hives and squinting and raining or **frowning** → **tranquilizers** or **painkillers and antibiotics**

Observations: hoarse, blurry vision, puffy eyes

Disease Medications:

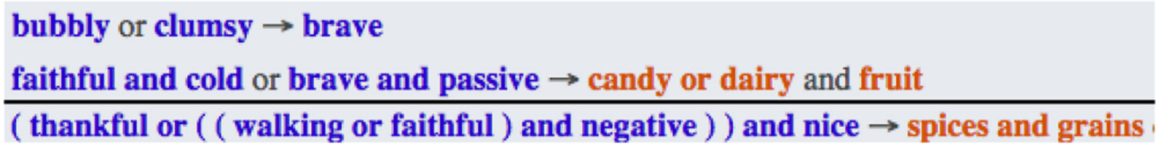
- **antibiotics:** Aerove, Adenon, Athoxin
- **painkillers:** Poxin, Parola, Pelapin
- **vitamins:** Vipryl, Vyorix, Votasol
- **stimulants:** Silvax, Setoxin, Soderal
- **tranquilizers:** Trasmin, Tydesol, Texopal
- **laxatives:** Lantone, Lezanto, Lexerol

What prescription would you recommend to treat the alien's symptoms?

□ Vitamins
□ Antibiotics
□ Laxatives
□ Tranquilizers
□ Stimulants
□ Painkillers

Figure 1:

Screenshot of our interface for the simulation task in the V3–Variable Repetition experiment in the Clinical domain. The middle left box shows the observations we give participants about the alien–i.e. the features of the input example. Participants must give a valid recommendation that will satisfy the alien given the observations and preferences. The “Disease medications” box defines the necessary concepts for the experiment. Each task is coded in a different color (here, blue) to visually distinguish them.



bubbly or clumsy → brave

faithful and cold or brave and passive → candy or dairy and fruit

(thankful or ((walking or faithful) and negative)) and nice → spices and grains

Figure 2:
An explicit (top) and an implicit (bottom) cognitive chunk used inside a decision rule.

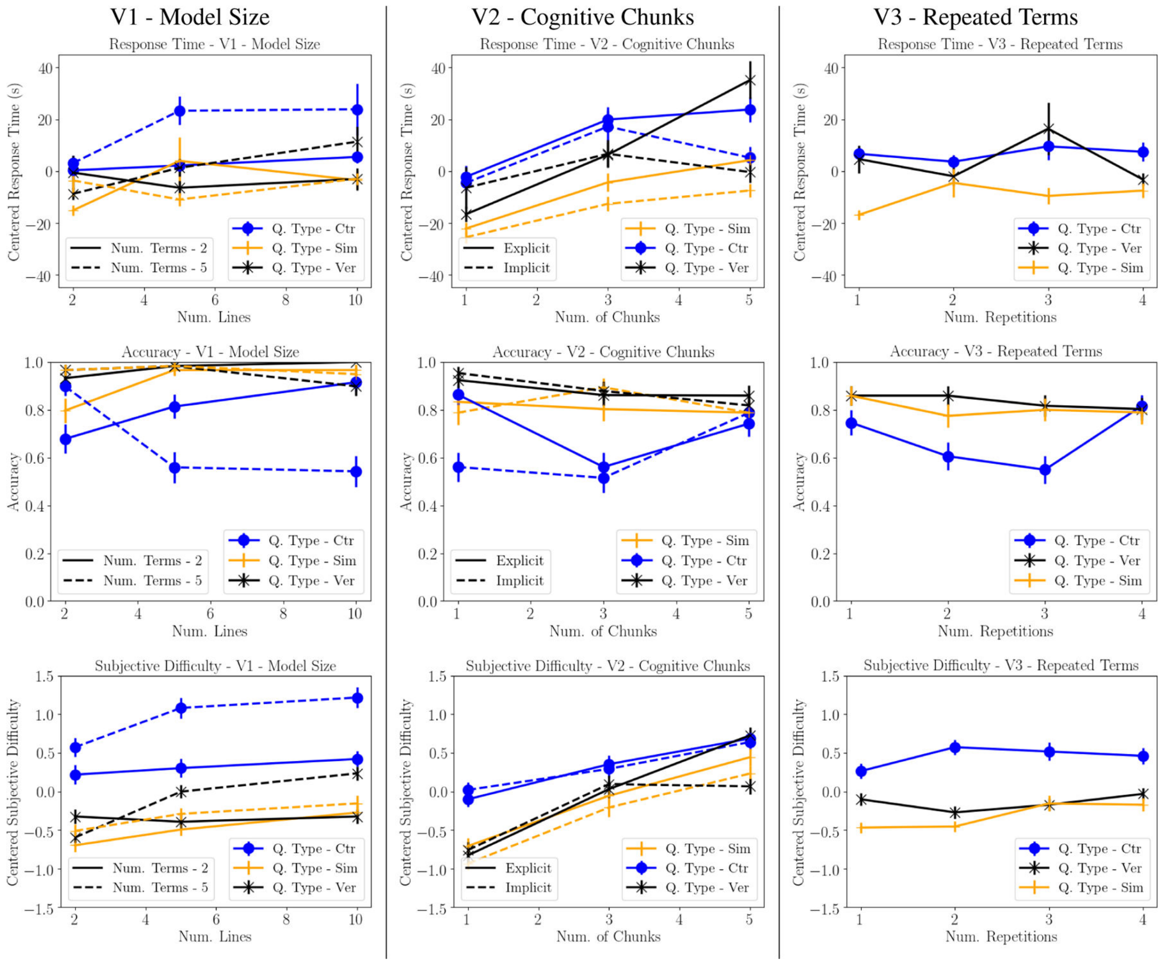


Figure 3:

Accuracy, response time and subjective evaluation for all experiments in the recipe domain. Vertical lines signify standard errors. See Lage et al. (2019) for the corresponding set of figures in the clinical domain.

Table 1:

We conduct 3 experiments testing different types of complexity in 2 parallel domains. We ask participants to complete 3 different tasks based on human-simulatability.

Types of Complexity	**Tasks**	**Domains**
V1: Model Size	Verification	Recipe
V2: Cognitive Chunks	Simulation	Clinical
V3: Repeated Terms	Counterfactual	

Table 2:

Number of participants who met our inclusion criteria for each experiment.

Experiment	Clinical	Recipe
Model Size (V1)	69	59
Cognitive Chunks (V2)	55	62
Repeated Terms (V3)	52	70

Table 3:

Significance tests for each factor for normalized response time, accuracy and normalized subjective difficulty of use. A single linear regression was computed for each independent variable for each of V1, V2, and V3 in each domain. Coefficients for verification and counterfactual tasks should be interpreted with respect to the simulation task. Highlighted p-values are significant at $\alpha = 0.05$ with a Bonferroni multiple comparisons correction across all tests of all experiments.

	Response Time				**Accuracy**				**Subjective Difficulty**			
	Clinical		**Recipe**		**Clinical**		**Recipe**		**Clinical**		**Recipe**	
	Weight	**P-Value**	**Weight**	**P-Value**	**Weight**	**P-Value**	**Weight**	**P-Value**	**Weight**	**P-Value**	**Weight**	**P-Value**
# Lines (V1)	01.17	**<.0001**	01.01	.0032	0.03	.2360	0.006	.8420	0.05	**<.0001**	0.05	**<.0001**
# Terms (V1)	02.35	**<.0001**	01.57	.0378	−0.14	.0110	−0.12	.0771	0.12	**<.0001**	0.12	**<.0001**
Ver. (V1)	10.50	**<.0001**	04.11	.1210	0.93	**0.0007**	0.48	.1740	0.13	.0187	0.17	.0048
Counter. (V1)	21.00	**<.0001**	13.70	**<.0001**	−1.41	**<.0001**	−1.70	**<.0001**	1.01	**<.0001**	1.04	**<.0001**
# Chunks (V2)	06.04	**<.0001**	05.88	**<.0001**	−0.04	.4200	−0.04	.4160	0.18	**<.0001**	0.25	**<.0001**
Implicit (V2)	−13.00	**<.0001**	−07.93	**0.0005**	−0.25	.0930	−0.18	.2220	−0.23	**<.0001**	−0.12	.0171
Ver. (V2)	16.30	**<.0001**	15.40	**<.0001**	0.65	.0008	0.53	.0090	0.07	.3050	0.09	.1400
Counter. (V2)	08.56	.0265	19.90	**<.0001**	−0.37	.0294	−0.77	**<.0001**	0.29	**<.0001**	0.52	**<.0001**
# Rep. (V3)	01.90	.2470	00.88	.4630	0.02	.8040	−0.05	.5240	0.06	.0373	0.07	.0041
Ver. (V3)	13.00	.0035	13.70	**<.0001**	0.15	.5960	0.20	.3710	0.03	.6640	0.17	.0090
Counter. (V3)	20.30	**<.0001**	16.60	**<.0001**	−1.07	**<.0001**	−0.67	**0.0007**	0.89	**<.0001**	0.77	**<.0001**