On *Mechanistic Stability*, Generalization, and Robustness

Alan Sun, Ethan Sun, Warren Shepard

awsun@cmu.edu, {ethan.k.sun.26, warren.a.shepard}@dartmouth.edu

January 21, 2025

Abstract

Herein, we introduce a notion which we call *mechanistic stability* which characterizes the sensitivity of a models' prediction criteria with respect to perturbations of the task specification.

Contents

1 Defining Mechanistic Staiblity

At the highest level, *mechanistic stability* is the stability of a model's decision criteria with respect to changes in the input. In many cases, we expect a strong, generalizable learner to also be one that is mechanistically stable. For example, given two-operand addition problems, we would expect a strong learner to add four-digit numbers in the same way it adds eight-digit numbers. On the other hand, a mechanistically stable learner may be necessary for fairness. Consider a model that evaluates job applicants. A legal and ethical model must not change its evaluation criteria based on an applicant's gender. In this paper, we formalize using tools from category theory, this notion of mechanistic stability. Then, we show that mechanistic stability is a sufficient condition for both generalization and robustness. Finally, we provide a host of empirical results to showcase when mechanistic stability can be induced or inhibited. Throughout the paper, we use the task of two-operand addition as a running example.

This remainder of this section is organized as follows:

1. First, in Section 1.1, we formalize what we mean by "changes in the input" through the concept of a *permissible partition* over a data distribution.
2. Then, in Section 1.2, we define "stability a model's decision criteria" through a category-theoretic equivalence between a model's mechanism on a specific *permisssible partition*.
3. FInally, in Section 1.3, by combining these components together, we arrive at our notion of *mechanistic stability*.

Our contribution focuses on the supervised learning regime. Under this setup, any model's input is chosen from a set $\mathcal{X}$ and its outputs lie in a set $\mathcal{Y}$. Let $(\mathcal{X}, \mathcal{F}_\mathcal{X}, \mathbb{P}_\mathcal{X})$ and $(\mathcal{Y}, \mathcal{F}_\mathcal{Y}, \mathbb{P}_\mathcal{Y})$ be probability spaces over $\mathcal{X}, \mathcal{Y}$, respectively. Denote by $(\mathcal{X} \times \mathcal{Y}, \mathcal{F}_\mathcal{X} \otimes \mathcal{F}_\mathcal{Y}, \mathbb{P}_\mathcal{X} \times \mathbb{P}_\mathcal{Y})$ the product probability space over all input-output pairs. We call any probability measure $\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ that is absolutely continuous[1] to $\mathbb{P}_\mathcal{X} \times \mathcal{P}_\mathcal{Y}$ a *data distribution*. A data distribution also determines the conditional distribution $\mathcal{P}_{\mathcal{X}\times\mathcal{Y}}[y|x]$ which intuitively captures the elements of $\mathcal{Y}$ that are an appropriate response given $\mathcal{X}$. Therefore, any supervised learning task can be seen as learning the distribution $\mathcal{P}_{\mathcal{X}\times\mathcal{Y}}$.

Definition 1.1 (Task) *A **task**, T, is a data distribution which we also denote as $\mathcal{D}$.*

For the remainder of the paper, our discussions resolve around a *single, fixed, but arbitrary* task. A brief discussion on the generalizations of this concept to the multitask setting can be found in ... Moreover, we use $T, \mathcal{D}, \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ interchangably.

1.1 Partitioning Subtasks

Many tasks contain inherent structure. For two-operand addition, we can divide this tasks into *subtasks*, where each subtask is the set of all m-by-n digit addition problems. Herein, we formalize this notion of task substructure through *subtasks*. To do this, we partition the universe of all possible input-outputs through a *permissble partitioning*.

Definition 1.2 (Permisslbe Partition) *For a given task T with distribution $\mathcal{D}$, a **permissible partition**, denoted by $\mathcal{S}$, is a countable collection of measurable subsets of $\mathcal{F}_\mathcal{X} \otimes \mathcal{F}_\mathcal{Y}$ that satisfies*

$$s' \cap s = \emptyset \qquad \textit{for all } s, s' \in \mathcal{S}, \tag{1}$$

$$\bigcup_{s\in\mathcal{S}} s = \mathcal{X} \times \mathcal{Y}, \tag{2}$$

[1] For any $E \in \mathcal{F}_\mathcal{X} \otimes \mathcal{F}_\mathcal{Y}$, if $(\mathcal{P}_\mathcal{X} \times \mathcal{P}_\mathcal{Y})(E) = 0$, then $\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}(E) = 0$.

$$\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}[s] > 0 \qquad \textit{for all } s \in \mathcal{S}. \tag{3}$$

Henceforth, unless otherwise specified, we assume that all partitions are permissible. A partitions not only carves up $\mathcal{X} \times \mathcal{Y}$ but also induces two new types of distributions. These are shown and expanded upon in Fig. 1.

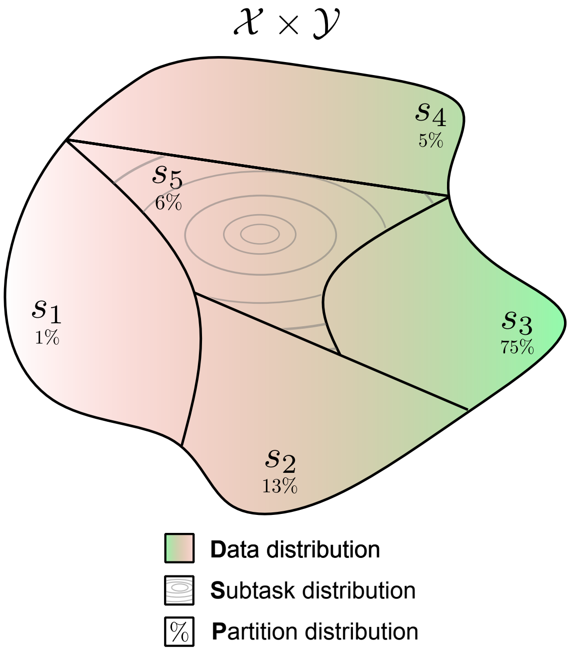

Figure 1: Starting with a data distribution $\mathcal{D}$, a partition induces two additional distributions of interest: subtask distributions (**S**), and the partition distribution (**P**). The underlying data distribution of the task (**D**) is shown as a color gradient and behaves independent of any partition. On the other hand, the subtask distribution (**S**) is the data distribution conditional on a particular element of the partition. Lastly, the partition distribution represents the probability that an input-output pair drawn from **D** will fall in any element of the partition.

By Def. 1.2, since any element of a partition has non-zero probability mass, the data distribution conditioned on this element is well-defined and is itself also a data distribution (see Def 1.1). Thus, we can now formally define the notion of a *subtask*.

Definition 1.3 (Subtask) *For a given task, T, and its corresponding data distribution $\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$, consider any partition $\mathcal{S}$ of T. For any $s \in \mathcal{S}$, a* ***subtask****, T_s, is the conditional distribution*

$$\mathbb{P}_{\mathcal{X}\times\mathcal{Y}|s}[A] = \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}[A|s] = \frac{\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}[A \cap s]}{\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}[s]}.$$

This last equation is the definition of conditional probability. When it is clear from context what the partition is we denote this distribution as $\mathbb{P}_s$.

For any partition, we also define a discrete *partition distribution* that will come in use technically in a bit.

Definition 1.4 *For a given task T, parition $\mathcal{S}$, the* ***partition distribution*** *is a probability space $(\mathcal{S}, 2^{\mathcal{S}}, \mathbb{P}_{\mathcal{S}})$,* *where for all $s \in \mathcal{S}$, $\mathbb{P}_{\mathcal{S}}[s] = \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}[s]$.*

Remark 1.1 *The conditions for a partition to be permissible is quite weak. Are there perhaps more desirable conditiosn that we should require out permissble partitions to have? Fundamentally, as the partitions get larger and larger, it should be easier to achieve mechanistic stability. Some things to think about:*

- *Changing "countable collection" to a "finite collection"?*
- *Instead of only requiring that our partitions have positive probability mass, maybe we need that the subsets be ε-representative (Shalev-Shwartz and Ben-David, 2014).*

1.2 Causal Equivalence: A Category-Theoretic Perspective

Herein, we introduce some basic category-theoretic concepts and use them to formally define causal graphs and their equivalence,. These constructions originate from Jacobs et al. (2021) and have since been expanded upon through works such as Beckers and Halpern (2019); Otsuka and Saigo (2022); **?**); **?** have extended this framework for their own specific applications. We provide a brief introduction here and refer the reader to both Jacobs et al. (2021) for the detailed constructions.

Definition 1.5 *Let Stoch be the category of all probability spaces. Its objects are probability spaces and its morphisms are Markov transition kernels[a].*

[a] One can find a detailed construction here. This is an extension of the framework presented in Jacobs et al. (2021) as they only consider the construction of Stoch with finite sets and transition matrices.

A causal graph not only contains the probability dynamics between its variables (the semantics of the causal graph), but it also contains syntactical information that encode real-world knowledge a priori restricting the set of permissible probability transition dynamics. We can also define a category that captures these structures.

Definition 1.6 *Let $G = (V_G, E_G)$ be a directed acyclic graph (DAG) with vertices V_G and edges E_G. Denote by Syn_G be the free CDU category[a] generated by G.*

[a] for a detailed construction of this see Jacobs et al. (2021).

Both Stoch and Syn_G are *symmetric monoidal categories*. We are now ready to formally define a causal graph.

Definition 1.7 *A **causal graph** defined by the DAG G, is a functor $F : \mathsf{Syn}_G \to \mathsf{Stoch}$.*

Next is a result from Jacobs et al. (2021) which shows that we haev in some sense sufficiently captured the set of all causal graphs.

Proposition 1.1 *There is a 1-1 correspondance between the Bayesian networks on a DAG G and the functors of type $\mathsf{Syn}_G \to \mathsf{Stoch}$.*

We then directly use this category-theoretic notion of causality to define equivalence between two causal models on the same causal graph which is a special case of causal-equivalence defined in Otsuka and Saigo (2022).

Definition 1.8 *Let $F, H : \mathsf{Syn}_G \rightrightarrows \mathsf{Stoch}$ be two causal graphs on G. Then, F, H are **causally-equivalent** if there exists a natural isomorphsim from F to H.*

For any model M, we may consider its mechanisms to be a causal graph, a functor from $\mathsf{Syn}_M \to \mathsf{Stoch}$. Moreover, for two distinct subcircuits/submechanisms of the same model, we can model them as parallel functors $\mathsf{Syn}_M \rightrightarrows \mathsf{Stoch}$.

1.3 Mechanistic Stability

With these notions of causal equivalence defined in the previous section, we are now ready to define mechanisitic stability.

Definition 1.9 (ε-Mechanistically Stable) *A model M is ε-mechanistically stable for any $\varepsilon > 0$ with respect to some task T, if*

$$\sup_{\mathcal{S}:\textit{partition}} \left[\mathbb{P}_{s,s' \sim \mathcal{S} \otimes \mathcal{S}}[F_s \ncong F_{s'}]\right] < \varepsilon, \tag{4}$$

where $\mathbb{P}_{s,s'}$ is the measure on the product partition space[a], F_s is the causal graph induced by the subtask distribution s and $F_s \ncong \mathbb{F}_{s'}$ denotes that the causal graphs $F_s, F_{s'}$ are not causally equivalent.

[a] sampling two partitions at randomly according to the partition distribution defined in Def. 1.4.

2 Main results

Herein, is a wish list of the main results that we wish to achieve.

Theorem 2.1 *Mechanistic stability $\Rightarrow$ in-distribution robustness, assuming that both $\mathcal{X}, \mathcal{Y}$ are metric spaces. We also need that the task distribution $\mathcal{D}_{\mathcal{X}\times\mathcal{Y}}$ is smooth.*

Theorem 2.2 *Mechanistic stability $\Rightarrow$ in-distribution generalization.*

Maybe for proving this latter theorem, we can look at some proofs relating regularization. What is a notion of in-distribution generalization that makes sense to look at here. Can we say something stronger maybe? What about out-of-distribution? like distribution shift?

3 Related Literature

What is the relationship between our notion of mechanistic stability and the more general notion of algorithmic stability? Is it potentially the same as algorithmic stability applied to in-context learning?

Algorithmic stability. What is the relationship between algorithmic stability and the notion of mechanistic stability that we are defining, both in terms of the generalization bounds that are guaranteed with algorithmic stability and in terms of the concept itself (how are we measuring the distance between two learned hypotheses?)

Mechanistic interpretability. Recently, there has a been a push to interpret neural networks by uncovering their *mechanisms*. A mechanism of a neural network with respect to some task is a minimal subgraph of its computational graph that wholly characterizes the network's behavior on this task (Wang et al., 2022). After exposing the driving mechanisms

A brief introduction to the field and how our work provides rigorous guarantees for a lot of the work being done in MI.

Graphical models. Probably need to rethink the title here, we want to explain the relationship between directed acyclic graphs, causal graphs, and the mechanisms that we are extracting from the neural network. Our underlying assumption is that all of these things are the same.

4 Experimental Methods

Can we show mechanistic stability and instability on a host of tasks?

1. IOI
2. Colored Objects
3. Arithmetic
4. General algorithmic tasks
5. General reasoning tasks (this and the previous task would benefit from increased test time compute and enhanced supervision; so we can we somehow increase stability by increasing one of these factors?)
6. General knowledge tasks (would not benefit from chain-of-thought or more test time compute)

What is the effect of scale on mechanistic stability? Or conversely, the benefits that we see in terms of performance as a result of increased scale, does this come from increased mechanistic stability?

References

Sander Beckers and Joseph Y Halpern. Abstracting causal models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2678–2685, 2019. Issue: 01.

Bart Jacobs, Aleks Kissinger, and Fabio Zanasi. Causal inference via string diagram surgery: A diagrammatic approach to interventions and counterfactuals. *Mathematical Structures in Computer Science*, 31(5):553–574, 2021. Publisher: Cambridge University Press.

Jun Otsuka and Hayato Saigo. On the Equivalence of Causal Models: A Category-Theoretic Approach. In Bernhard Schölkopf, Caroline Uhler, and Kun Zhang, editors, *Proceedings of the First Conference on Causal Learning and Reasoning*, volume 177 of *Proceedings of Machine Learning Research*, pages 634–646. PMLR, April 2022. URL https://proceedings.mlr.press/v177/otsuka22a.html.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the Wild: a Circuit for Indirect Object Identification in GPT-2 Small. In *The Eleventh International Conference on Learning Representations*, September 2022. URL https://openreview.net/forum?id=NpsVSN6o4ul.

5 Category-Theoretic Causality