HYPERPARAMETER OPTIMIZATION IS DECEIVING US, AND HOW TO STOP IT

A. Feder Cooper,[*] Yucheng Lu, and Chris De Sa

1 INTRODUCTION

Unlike the learned output parameters of a ML model, hyperparameters (HPs) are inputs provided to the learning algorithm that guide the learning process (Feurer & Hutter, 2019). While HPs are known to greatly influence overall algorithm performance (e.g., convergence rate, correctness, generalizability), research papers that do not focus explicitly on mechanisms of HP optimization (HPO) tend to treat HP selection as an empirical afterthought (Sivaprasad et al., 2020; Melis et al., 2018). The learning problem is of primary theoretical interest. HPO particulars are relegated to the realm of empirical curiosity (Wilson et al., 2017; Schneider et al., 2019; Chen et al., 2018) — "just an engineering problem" at the boundary of what ML research treats as "real science" (Gieryn, 1983).

This is evident from common practices in the literature to conduct HPO: It is typical to pick a small set of possible HPs and to perform grid or random search over them, comparing the empirical performance of the resulting trained models and reporting on the one that performs best (Hsu et al., 2003; Larochelle et al., 2007). For grid search, the grid points are often manually set by "folklore" parameters—values put forth in classic papers as rules-of-thumb concerning, e.g., how to set the learning rate (LeCun et al., 1998; Hinton, 2012; Pedregosa et al., 2011). This kind of ad-hoc decision-making in HPO, while generally accepted in the ML community, does not reflect a search for scientific truth: The chosen HPs might appear to yield desirable empirical performance; however, this process provides no actual assurances concerning whether the selected HPs are optimal.

We argue that **the process of drawing conclusions using HPO should itself be an object of study**. Our contribution is to put forward the first theoretically-backed characterization for making trustworthy conclusions about algorithm performance using HPO. We address theoretically the following empirically-observed problem: When comparing algorithms, $\mathcal{J}$ and $\mathcal{K}$, searching one subspace can pick HPs that yield the conclusion that $\mathcal{J}$ outperforms $\mathcal{K}$, whereas searching another can select HPs that entail the opposite. In short, your choice of hyperparameters can deceive you—a problem that we term *hyperparameter deception*. We formalize this problem and prove a defense to counteract it.

2 ILLUSTRATING DECEPTION: INTUITION

Running supervised learning is often thought of as a double-loop optimization problem, H:

$$\arg\min_{\lambda \in \Lambda} \ \mathbb{E}_x[\mathcal{L}_{\text{HPO}}(x; \mathcal{A}_\lambda(\mathcal{M}_\lambda, X_{\text{train}})] = \lambda^* \tag{1}$$

The inner-loop is typically called "training." It learns the parameters θ of some model $\mathcal{M}_\lambda$ by running an algorithm $\mathcal{A}_\lambda$ on a dataset X_{train}. Both the training algorithm and the model are parameterized by a vector of *hyperparameters* (HPs) λ (e.g. the learning rate and network size). The outer-loop optimization finds HPs λ^* from a set of allowable HPs Λ: λ^* results in a trained model that performs the best in expectation on "fresh" examples x drawn from the same source as the training set, as measured by some loss $\mathcal{L}_{\text{HPO}}$. An algorithm H that attempts this task is called a *hyperparameter optimization* (HPO) procedure (Defined more formally in the Appendix).

How do we pick the Λ within which H looks for the best-performing λ^*? Often, Λ is hand-picked using "folklore parameters." H then involves manually testing these popular options, selecting the λ that performs best on the chosen validation metric. More principled methods include grid search, used for decades (John, 1994), and random search, popularized by Bergstra & Bengio (2012). For the former, HPO evaluates $\mathcal{A}_\lambda$ on a grid of HPs λ. For the latter, the values in each tested configuration λ are randomly sampled from chosen distributions. Importantly, both of these algorithms are parameterized themselves: Random search requires distributions from which to sample and grid search requires inputting the spacing between different configuration points in the grid. We call

[*]Author emails: afc78@cornell.edu, yl2967@cornell.edu, cmd353@cornell.edu

these HPO-procedure-input values *hyper-hyperparameters*. We call the output of HPO a *log*, which has all of the information necessary to make running H reproducible.

Running H is a crucial part of model development. Yet, in practice, a researcher runs HPO (perhaps a few times) for the algorithm under evaluation, until they achieve and can report results (and logs) that align with the argument they want to make about the algorithm's performance. This process is rather ad-hoc and does not necessarily yield reliable knowledge about the algorithm's performance more generally. Our goal is to study HPO in this scientific-knowledge sense: We want to develop ways to reason about how we derive knowledge from empirical investigations involving HPO.

Studying grid search shows the need for bringing rigor to this process. How we set the hyper-HPs to determine the grid can directly impact our conclusions, even making it possible to draw contradictory conclusions about algorithm performance—an observation that informs our formalization for reasoning about deception in Section 4. We explain the intuition behind deception via an example: The comparison between SGD and SGD with heavy ball from Wilson et al. (2017) in Figure 1. The original step size α grid from Wilson et al. (2017) is $\{2, 1, 0.5, 0.25, 0.05, 0.01\}$ and the best test accuracy occurs for SGD at $\alpha = 1$. However, when we change the grid range to be $\{1.5, 0.75, 0.375, 0.15, 0.025\}$—notably excluding $\alpha = 1$—we conclude the opposite: SGD with heavy ball performs best in terms of test accuracy, when $\alpha = 0.15$. We also observe this phenomenon in NLP (Merity et al., 2016) (Appendix).

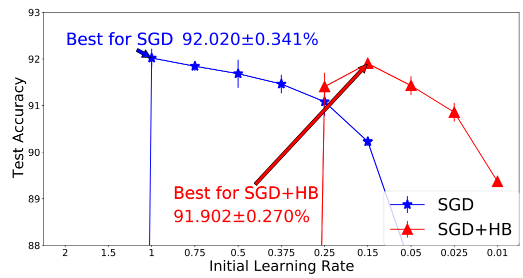


Figure 1: Demonstrating *hyperparameter deception* in Wilson et al. (2017)'s VGG16 experiment. We use 5 different seeds and average. Each point contains an error bar within 0.3%.

These results reveal that, even in highly cited and respected experiments, it is possible to be deceived; we can draw inconsistent conclusions simply by changing the hyper-HP for grid spacing. By varying the HPO procedure, it is possible to develop results that are wrong about performance, or else correct about performance but for the wrong reasons (e.g., by picking "lucky" HPs). Neither of these outcomes constitutes rigorous knowledge (Gettier, 1963; Lehrer, 1979).

3 Epistemic Hyperparameter Optimization

Section 2 shows that applying standard HPO methods can be deceptive: Our beliefs about performance can be controlled by happenstance, wishful thinking, or potentially by an adversary trying to trick us with a tampered set of HPO logs. This leaves us in a position where the "knowledge" we derived may not be knowledge at all—we could easily, had circumstances been different, have concluded the opposite. We therefore propose that **the process of drawing conclusions using HPO should itself be an object of study**. We start by formalizing this sort of reasoning process, which we call Epistemic Hyperparameter Optimization (EHPO).

Definition 1 *An* ***epistemic hyperparameter optimization procedure (EHPO)*** *is a tuple* $(\mathcal{H}, \mathcal{F})$ *where* $\mathcal{H}$ *is a set of HPO procedures* H *(Definition 3) and* $\mathcal{F}$ *is a function that maps a set of HPO logs* $\mathcal{L}$ *to a set of logical sentences* $\mathcal{P}$*. An execution of EHPO involves running each* $H \in \mathcal{H}$ *some number of times (each run produces a log* ℓ*) and then evaluating* $\mathcal{F}$ *on the set of logs produced to output the* conclusions *we draw from all of the HPO runs.*

In practice, it is most common to run EHPO for two algorithms, $\mathcal{J}$ and $\mathcal{K}$, and to compare their performance to conclude which is better-suited for the learning task at hand. H contains at least one HPO procedure that runs $\mathcal{J}$ and one that runs $\mathcal{K}$; possible conclusions in the co-domain of $\mathcal{F}$ include p = "$\mathcal{J}$ performs better than $\mathcal{K}$", and $\neg p$ = "$\mathcal{J}$ does not perform better than $\mathcal{K}$". Intuitively, EHPO is deceptive whenever it could produce both p and also could (if configured differently or due to randomness) produce $\neg p$. We can be deceived if EHPO could entail logically inconsistent results.

We find it useful to frame deception in terms of an adversary, akin to Descartes's Evil Demon. Imagine a demon who is trying to deceive us about the relative performance of different algorithms via running EHPO. The demon has a set $\mathcal{L}$ of HPO logs, which it can modify either by running HPO $H \in \mathcal{H}$ with whatever hyper-HPs $s \in \mathcal{S}$ it wants (producing a new log ℓ, which it adds to $\mathcal{L}$) or by erasing some of the logs in its set. Eventually, it stops and presents us with $\mathcal{L}$, from which we draw some conclusions using $\mathcal{F}$. The demon may be trying to deceive us via the conclusions it is possible to draw. For example, $\mathcal{L}$ may lead us to conclude that one algorithm performs better than another, when picking a different set of hyper-HPs could have generated logs that would lead us to conclude differently. We want to be sure that we will not be deceived by any logs the demon "could" produce.

4 A Logic for Reasoning about Deception

We now develop axioms for EHPO to help reason about hyperparameter deception. Modal logic to be a useful way to express this problem (Chellas, 1980; Emerson, 1991; Garson, 2018); it inherits the tools of propositional logic and adds two operators: $\Diamond$ to represent *possibility* and $\Box$ to represent *necessity*, making it the natural choice to express the "could" intuition from the previous section. The well-formed formulas ϕ are given recursively in Backus-Naur form: [1] $\phi := P \mid \neg\phi \mid \phi \wedge \phi \mid \Diamond\phi$, where P is any atomic proposition. For example, $\Diamond p$ reads, "It is possible that p.". The axioms of modal logic are as follows (appendix),[2] where Q and R are any formula: $\vdash Q \rightarrow \Box Q$ *(necessitation)* and $\Box(Q \rightarrow R) \rightarrow (\Box Q \rightarrow \Box R)$ *(distribution)*. To reason about EHPO requires an extension of standard modal logic, as we need *two* modal operators: one to express the possible results of the demon running EHPO and one to express our belief. The former must also be an *indexed* modal logic, where "how possible" something is is quantified by the compute capabilities of the demon. Combining these logics yields well-formed formulas

$$\psi := P \mid \neg\psi \mid \psi \wedge \psi \mid \Diamond_t \psi \mid \mathcal{B}\psi$$

for any atomic proposition, P, and for any positive real t (which we assign the semantics of "time").

Our semantics are defined using sets of logs $\mathcal{L}$: We use $\mathcal{L} \models p$, read "$\mathcal{L}$ models p" to mean that the sentence p is true for the set of logs $\mathcal{L}$. We suppose that an EHPO user has in mind some atomic propositions (propositions of the background logic unrelated to possibility or belief, such as "$\mathcal{J}$ performs better than $\mathcal{K}$") with semantics that are already defined. We also inherit the usual semantics of $\wedge$ ("and") and $\neg$ ("not") from ordinary propositional logic. In what follows, we show how we can use this to construct semantics for our modal operators of possibility and belief.

Expressing the possible outcomes of EHPO. Our formalization of possibility is based on the demon. Unlike Descartes, we need not concern ourselves with "supremely powerful" demons: our potential deceivers are mere mortal ML researchers (or adversaries) with bounded compute resources:

Definition 2 *Let Σ denote the set of randomized **strategies** for the demon. Each $\sigma \in \Sigma$ is a function that specifies which action the demon will take: Given its current set of logs $\mathcal{L}$, either 1) running a new H with hyper-hyperparameters s (for which the demon gets a new, randomly-generated seed) 2) erasing some logs, or 3) returning. We let $\sigma[\mathcal{L}]$ denote the outputs of σ running, starting from $\mathcal{L}$ (i.e., the demon is given the logs in $\mathcal{L}$ to start, then gets to run a strategy σ). Let $\tau_\sigma(\mathcal{L})$ denote the total time taken to run strategy σ; this is equivalent to the sum of the times T it takes each HPO procedure $H \in \mathcal{H}$ used in the demon's strategy to run. Note that since σ is a randomized strategy (and HPO runs H are randomized as well), both $\sigma[\mathcal{L}]$ and $\tau_\sigma(\mathcal{L})$ are random variables. For any formula p, we say $\mathcal{L} \models \Diamond_t p$ if and only if $\exists \sigma \in \Sigma,\ \mathbb{P}(\sigma[\mathcal{L}] \models p) = 1 \ \wedge\ \mathbb{E}[\tau_\sigma(\mathcal{L})] \leq t$.*

Informally, $\Diamond_t p$ means that an adversary could adopt a strategy σ that is guaranteed to cause the desired outcome p to be the case while taking time at most t in expectation. We will usually choose t to be an upper bound on what is considered a reasonable amount of time to run HPO. In this case, any practical adversary cannot consistently bring about p unless $\Diamond_t p$. Our indexed modal logic inherits many axioms of modal logic, with indexes added (Appendix).

Expressing how we draw conclusions. We use the modal operator $\mathcal{B}$ from belief logic to model our belief in the truth of the conclusions drawn from running EHPO. We model ourselves as a consistent *Type 1* reasoner (Smullyan, 1986) (Appendix): i.e. for any formula p, we require consistency: $\neg(\mathcal{B}p \wedge \mathcal{B}\neg p)$, where $\mathcal{B}p$ reads "It is concluded that p." The semantics for our belief are straightforward: In the context of EHPO, in which our conclusions are based on function $\mathcal{F}$, we say that a set of logs $\mathcal{L}$ models a formula $\mathcal{B}p$ when our set of conclusions $\mathcal{F}(\mathcal{L})$ contains p, i.e., $\mathcal{L} \models \mathcal{B}p \ \equiv\ p \in \mathcal{F}(\mathcal{L})$.

Expressing deception. Now we want to show how $\Diamond_t$ and $\mathcal{B}$ interact with each other to formally express what we informally illustrated in Section 2: hyperparameter deception. We combine modal

[1] Note $\Box$ is syntactic sugar, with $\Box p \equiv \neg\Diamond\neg p$; "or" has $p \vee q \equiv \neg(\neg p \wedge \neg q)$;"implies" has $p \rightarrow q \equiv \neg p \vee q$.
[2] Here, "$\vdash Q$" means that Q is a theorem of propositional logic.

logics (Appendix) to define an axiom for EHPO being deception-free: for any formula p,

$$\neg\left(\Diamond_t \mathcal{B} p \ \wedge \ \Diamond_t \mathcal{B} \neg p\right) \qquad\qquad \textit{(t-non-deceptive).}$$

Informally, if there exists a strategy by which the demon could get us to conclude p in t expected time, then there can exist no t-time strategy by which the demon could have gotten us to believe $\neg p$. We say that EHPO is non-deceptive if it satisfies all of the axioms above. If t-non-deceptiveness does not hold for some p, then even if we conclude p after running EHPO, we cannot claim to *know* p; our belief as to the truth-value of p could be under the complete control of an adversary.

5 Defending against Deception

Our formulation is not trivial: It is sufficiently expressive to show when deception occurs, e.g. for grid search (Appendix), and to reason about mechanisms of defense against it. Intuitively, if there is no adversary that can consistently control whether we believe algorithm $\mathcal{J}$ is better than $\mathcal{K}$ or its negation (and p or $\neg p$ more generally), then we are defended against deception. Suppose we have been drawing conclusions using some "naive" belief operator $\mathcal{B}_{\text{naive}}$ that satisfies Section 4's axioms. We can use $\mathcal{B}_{\text{naive}}$ to construct a new operator $\mathcal{B}_*$ that is guaranteed to be deception-free:

$$\mathcal{B}_* p \ \equiv \ \mathcal{B}_{\text{naive}} p \ \wedge \ \neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p.$$

That is, we conclude p only if both our naive reasoner would have concluded p, and it is impossible for an adversary to get it to conclude $\neg p$ in time t. This enables us to show t-non-deceptiveness:

$$\Diamond_t \mathcal{B}_* p \equiv \Diamond_t\left(\mathcal{B}_{\text{naive}} p \ \wedge \ \neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p\right) \rightarrow \Diamond_t\left(\neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p\right) \rightarrow \neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p \quad \text{(See Appendix)},$$

$$\Diamond_t \mathcal{B}_* \neg p \equiv \Diamond_t\left(\mathcal{B}_{\text{naive}} \neg p \ \wedge \ \neg \Diamond_t \mathcal{B}_{\text{naive}} p\right) \rightarrow \Diamond_t \mathcal{B}_{\text{naive}} \neg p \qquad \text{(See Appendix)},$$

which immediately lets us derive the t-non-deceptive axiom by contradiction. This derivation illustrates the power of our logical formulation: We can validate defenses against deception in only a few lines, without needing to refer to the underlying semantics of EHPO.

We now illustrate that it is possible to implement a defense by constructing a concrete EHPO that satisfies our axioms for a particular HPO setup. We use *random search* as the underlying HPO procedure. Random search takes two hyper-HPs, a distribution μ over the HP space and a number of trials K to run. Running HPO consists of K independent trials of the learning algorithm $\mathcal{A}_{\lambda_1}, \mathcal{A}_{\lambda_2}, \ldots, \mathcal{A}_{\lambda_K}$, where the HPs λ_k are independently drawn from μ, taking expected time proportional to K. We will suppose there is only one algorithm under consideration, $\mathcal{A}$, and that the set of allowable hyper-HPs (and in turn allowable HPs) is constrained, such that any two allowable random-search distributions μ and ν have Renyi-∞-divergence at most a constant $D_\infty(\mu \| \nu) \leq \gamma$.

We consider starting with a naive reasoner $\mathcal{B}_{\text{naive}}$, which draws conclusions from only a single log containing K trials. Our goal is to construct a $\mathcal{B}_*$ that has a "defended reasoner" $\mathcal{F}$. This $\mathcal{F}$ should weaken the conclusions of $\mathcal{F}_{\text{naive}}$ (i.e., $\mathcal{F}(\mathcal{L}) \subseteq \mathcal{F}_{\text{naive}}(\mathcal{L})$ for any $\mathcal{L}$) and be guaranteed to be t-non-deceptive. The $\mathcal{B}_*$ we construct similarly draws conclusions from a single log, but does so from a log containing KR trials for some fixed $R \in \mathbb{N}$. It evaluates a conclusion p by dividing the log's KR trials into R groups of K trials, evaluating $\mathcal{B}_{\text{naive}} p$ on each group, and concluding p only if $\mathcal{B}_{\text{naive}}$ also concluded p on all R groups. In the Appendix, we prove $\mathcal{B}_*$ will be t-non-deceptive if R is set to be $R = \sqrt{t \exp(\gamma K)/K} = O(\sqrt{t})$. This result both validates the defense and does so with a log size—and compute requirement for good-faith EHPO—that is sublinear in t. We empirically demonstrate a defense in the Appendix.

6 Conclusion: Toward More Robust ML

We consider our work on formalizing hyperparameter deception to be just as urgent as advocacy for better reproducibility (Henderson et al., 2018; Raff, 2019; Bouthillier et al., 2019). Reproducibility is only part of the story for ensuring robustness; it is necessary, but not sufficient. While reproducibility guards against brittle findings through the "good science" practice of replicable results, it does not guarantee that those results are actually correct. We address this need, providing a mechanism for reasoning more rigorously about algorithm performance in the context of HPO.

Our work also highlights how there is a human element, not a just statistical one, to bias in ML pipelines: Practitioners make decisions about hyper-HPs that can heavily influence performance. The human element of biasing solution spaces has been discussed in sociotechnical writing (Friedman & Nissenbaum, 1996), in AI (Mitchell, 1980), in the context of "p-hacking" (Gelman & Loken, 2019), and, more recently, was also the focus of Isbell (2020)'s NeurIPS keynote. Our push here to formalize conclusions from HPO has the potential to alleviate the effects of such bias.

ACKNOWLEDGEMENTS

We would like to thank the following individuals for feedback on earlier ideas, drafts, and iterations of this work: Harry Auster, Prof. Solon Barocas, Jerry Chee, Jessica Zosa Forde, Kweku Kwegyir-Aggrey, Prof. Karen Levy, and Prof. Helen Nissenbaum. We would also like to thank the Artificial Intelligance Policy & Practice initiative at Cornell University.

REFERENCES

James Bergstra and Yoshua Bengio. Random Search for Hyper-Parameter Optimization. *J. Mach. Learn. Res.*, 13:281–305, February 2012. ISSN 1532-4435.

James S. Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for Hyper-Parameter Optimization. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 24*, pp. 2546–2554. Curran Associates, Inc., 2011.

Guram Bezhanishvili and Wesley H. Holliday. A semantic hierarchy for intuitionistic logic. *Indagationes Mathematicae*, 30(3):403 – 469, 2019. ISSN 0019-3577.

Xavier Bouthillier, César Laurent, and Pascal Vincent. Unreproducible Research is Reproducible. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 725–734. PMLR, 09–15 Jun 2019.

Brian F. Chellas. *Modal Logic - An Introduction*. Cambridge University Press, 1980.

Irene Chen, Fredrik D Johansson, and David Sontag. Why Is My Classifier Discriminatory? In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31, pp. 3539–3550. Curran Associates, Inc., 2018.

Dami Choi, Christopher J. Shallue, Zachary Nado, Jaehoon Lee, Chris J. Maddison, and George E. Dahl. On Empirical Comparisons of Optimizers for Deep Learning, 2019.

René Descartes. *Discourse on Method and Meditations on First Philosophy*. Hackett Publishing Company, Inc., Translator Donald A. Cress, 4th edition, 1998. Meditation One: Concerning Those Things That Can Be Called into Doubt.

E. Allen Emerson. *Temporal and Modal Logic*, pp. 995–1072. MIT Press, Cambridge, MA, USA, 1991. ISBN 0444880747.

Matthias Feurer and Frank Hutter. Hyperparameter Optimization. In Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren (eds.), *Automated Machine Learning: Methods, Systems, Challenges*, pp. 3–33. Springer International Publishing, 2019.

Batya Friedman and Helen Nissenbaum. Bias in Computer Systems. *ACM Trans. Inf. Syst.*, 14(3): 330–347, July 1996. ISSN 1046-8188.

Sébastien Gadat, Fabien Panloup, Sofiane Saadane, et al. Stochastic heavy ball. *Electronic Journal of Statistics*, 12(1):461–529, 2018.

James Garson. Modal Logic. In *The Stanford Encyclopedia of Philosophy*. Fall 2018 Edition, Edward N. Zalta (ed.), 2018.

A. Gelman and Eric Loken. The garden of forking paths: Why multiple comparisons can be a problem, even when there is no ”fishing expedition” or ”p-hacking” and the research hypothesis was posited ahead of time, 2019.

Edmund L. Gettier. Is Justified True Belief Knowledge? *Analysis*, 23(6):121–123, 06 1963.

Thomas F Gieryn. Boundary-Work and the Demarcation of Science from Non-Science: Strains and Interests in Professional Ideologies of Scientists. *American Sociological Review*, 48(6):781–795, 1983.

Joseph Y. Halpern, Dov Samet, and Ella Segev. Defining Knowledge in Terms of Belief: The Modal Logic Perspective. *Rev. Symb. Log.*, 2(3):469–487, 2009.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep Reinforcement Learning that Matters. In *Thirty-Second AAAI Conference On Artificial Intelligence*, 2018.

Geoffrey E. Hinton. A Practical Guide to Training Restricted Boltzmann Machines. In Grégoire Montavon, Geneviève B. Orr, and Klaus-Robert Müller (eds.), *Neural Networks: Tricks of the Trade: Second Edition*, pp. 599–619. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

Chihwei Hsu, Chihchung Chang, and ChihJen Lin. A Practical Guide to Support Vector Classification, November 2003.

Charles Isbell. You Can't Escape Hyperparameters and Latent Variables: Machine Learning as a Software Engineering Enterprise. NeurIPS Keynote, 2020.

George H. John. Cross-Validated C4.5: Using Error Estimation for Automatic Parameter Selection. Technical report, Stanford University, Stanford, CA, USA, 1994.

Diederik Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*, 12 2014.

Hugo Larochelle, Dumitru Erhan, Aaron Courville, James Bergstra, and Yoshua Bengio. An Empirical Evaluation of Deep Architectures on Problems with Many factors of Variation. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pp. 473–480, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595937933.

Yann LeCun, Léon Bottou, Genevieve B. Orr, and Klaus-Robert Müller. Efficient BackProp. In *Neural Networks: Tricks of the Trade, This Book is an Outgrowth of a 1996 NIPS Workshop*, pp. 9–50, Berlin, Heidelberg, 1998. Springer-Verlag. ISBN 3540653112.

Keith Lehrer. The Gettier Problem and the Analysis of Knowledge. In George Sotiros Pappas (ed.), *Justification and Knowledge: New Studies in Epistemology*, pp. 65–78. Springer Netherlands, Dordrecht, 1979.

Chaoyue Liu and Mikhail Belkin. Accelerating sgd with momentum for over-parameterized learning. *arXiv preprint arXiv:1810.13395*, 2018.

Gábor Melis, Chris Dyer, and Phil Blunsom. On the State of the At of Evaluation in Neural Language models. In *International Conference on Learning Representations*, 2018.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

Tom M. Mitchell. The Need for Biases in Learning Generalizations. Technical report, Rutgers University, New Brunswick, NJ, 1980. `http://www-cgi.cs.cmu.edu/~tom/pubs/NeedForBias_1980.pdf`.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A Metric Learning Reality Check, 2020.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and et al. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, November 2011. ISSN 1532-4435.

Edward Raff. A Step Toward Quantifying Independently Reproducible Machine Learning Research. In *NeurIPS*, 2019.

Rasmus Rendsvig and John Symons. Epistemic Logic. In *The Stanford Encyclopedia of Philosophy*. Summer 2019 Edition, Edward N. Zalta (ed.), 2019.

F. Schneider, L. Balles, and P. Hennig. DeepOBS: A Deep Learning Optimizer Benchmark Suite. In *7th International Conference on Learning Representations (ICLR)*. ICLR, May 2019.

Dana Scott. Advice on modal logic. In Karel Lambert (ed.), *Philosophical Problems in Logic: Some Recent Developments*, pp. 143–173. Springer Netherlands, Dordrecht, 1970.

Prabhu Teja Sivaprasad, Florian Mai, Thijs Vogels, Martin Jaggi, and François Fleuret. Optimizer Benchmarking Needs to Account for Hyperparameter Tuning. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9036–9045. PMLR, 13–18 Jul 2020.

Raymond M. Smullyan. Logicians Who Reason about Themselves. In *Proceedings of the 1986 Conference on Theoretical Aspects of Reasoning about Knowledge*, TARK '86, pp. 341–352, San Francisco, CA, USA, 1986. Morgan Kaufmann Publishers Inc. ISBN 0934613049.

Richmond H. Thomason. Combinations of tense and modality. In D. Gabbay and F. Guenthner (eds.), *Handbook of Philosophical Logic: Volume II: Extensions of Classical Logic*, pp. 135–165. Springer Netherlands, Dordrecht, 1984. ISBN 978-94-009-6259-0.

Johan van Benthem. Epistemic Logic and Epistemology: The State of Their Affairs. *Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition*, 128(1):49–76, 2006.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The Marginal Value of Adaptive Gradient Methods in Machine Learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 4148–4158. Curran Associates, Inc., 2017.

A APPENDIX

A.1 FORMALLY DEFINING HPO

In practice, since we do not have access to the distribution from which the x are sampled, we cannot calculate the expected loss exactly. Instead, it is standard to use a validation dataset distinct from the training dataset to compute an approximation, which is used to determine λ^*.

Based on this and our definition of hyper-hyperparameters in Section 2, we provide the following formal definition for a hyperparameter optimization procedure:

Definition 3 *An HPO procedure H is a tuple $(H_*, \mathcal{S}, \Lambda, \mathcal{A}, \mathcal{M}, X)$ where H_* is a randomized algorithm, $\mathcal{S}$ is a set of allowable hyper-hyperparameters, Λ is a set of allowable hyperparameters, $\mathcal{A}$ is a learning algorithm, $\mathcal{M}$ is a model, and X is some dataset (usually split into train and validation sets). When run, H_* takes as input a hyper-hyperparameter configuration $s \in \mathcal{S}$, then proceeds to run $\mathcal{A}_\lambda$ (on $\mathcal{M}$ using data from X) some number of times for different hyperparameters $\lambda \in \Lambda$. Finally, H_* outputs a tuple $(\lambda^*, \theta^*, T, \ell)$, where λ^* is the optimal hyperparameter choice, θ^* are the corresponding parameters found for $\mathcal{M}_\lambda$, T is the total amount of* time *the HPO algorithm took to run, and ℓ is a **log** that records all the choices and measurements made during HPO. The log has all of the information necessary to make running H reproducible.*

For overall comprehension purposes, it is sufficient to understand the intution for H that we provide in Section 2. However, for our formalization and proofs, Definition 3 is what we use formally when referring to H.

A.2 ADDITIONAL EXPERIMENTAL RESULTS

We provide expanded empirical results for all experiments in Section 2, in which we plot the test accuracy for each hyperparameter configuration.

A.2.1 A TOY EXAMPLE

We first provide more explanation of the intuition behind hyperparameter deception using a toy example / scenario:

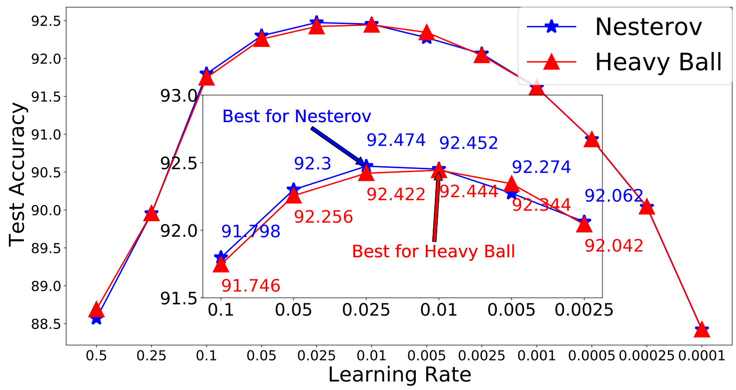
While methods like GD and SGD have been used for decades, and are guaranteed to converge to the global minimum on convex learning problems, developing new methods for solving convex problems at scale is an active area of research. It is common to develop a new algorithm and then compare it against a baseline: The two methods are trained for the same training budget and then compared to see which has superior performance.

We compare two commonly-used momentum-based SGD variants: SGD with Nesterov acceleration (Liu & Belkin, 2018) and SGD with heavy ball momentum (Gadat et al., 2018) for logistic regression on MNIST (Figure 2). These algorithms are of a similar flavor; they both modify the update of SGD with a momentum term, so it is natural to compare their performance. We apply grid search on the learning rate and compare the final test accuracy. How we set the hyper-hyperparameter for grid spacing can lead to logically inconsistent conclusions. Figure 2a, which uses a finer-grained powers-of-2 grid, leads to the conclusion that Nesterov performs best, while the coarser powers-of-10 grid in 2b leads to the contradictory conclusion that heavy ball gives superior performance. In short, we can be deceived into concluding the wrong method performs better based on how we specify the grid.

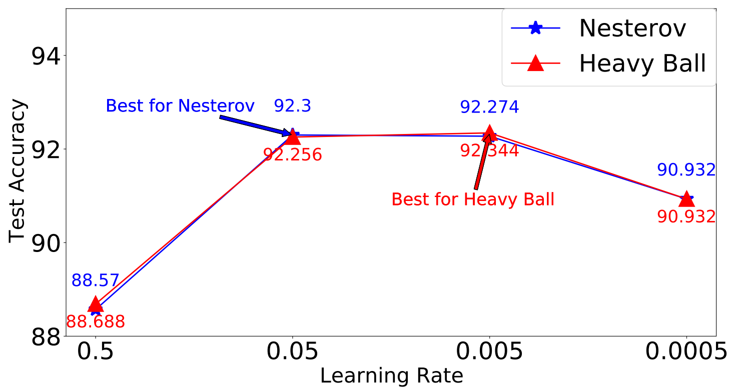
Experimental setup. We run this experiment on a local machine configured with a 2.6GHz Inter (R) Xeon(R) CPU and 4GB memory. We set the mini-batch size to be 64, the momentum constant to be 0.9 for both optimizers, and the total number of epochs to be 20. We adopt a constant learning rate and cross entropy loss.

A.2.2 DECEPTION IN ML RESEARCH

We emphasize that being inadvertently deceived by HPO is a real problem, even in excellent research; it is not limited to our toy example above. We found instances of this phenomenon in



(a) Test accuracy with a fine-grained grid. Adjacent grids points are related by a factor of 2. We conclude Nesterov performs the best.



(b) Test accuracy with a coarse-grained grid. Adjacent grids points are related by a factor of 10. We conclude that heavy ball performs the best.

Figure 2: Comparing the test accuracy of Nesterov acceleration and heavy ball on MNIST for different learning rates, on a fine-grained grid (a) and coarse-grained grid (b). Depending on how we set the hyper-hyperparameter for determining the grid spacing, not only do we conclude a different λ^*, we also draw different conclusions about whether Nesterov or heavy ball performs better.

well-cited papers across multiple domains: Wilson et al. (2017), in which they compare different optimizers training VGG16 on CIFAR10, and Merity et al. (2016)'s experiments with a LSTM on Wikitext-2.

Hyperparameter Deception in Vision – Implementing Wilson et al. (2017)

Experimental setup. We run this experiment on a local machine configured with a 4-core 2.6GHz Inter (R) Xeon(R) CPU, 16GB memory and an NIVIDIA GTX 2080Ti GPU. Following the exact configuration from Wilson et al. (2017), we set the mini-batch size to be 128, the momentum constant to be 0.9 and the weight decay to be $5e^{-4}$ for both optimizers.

The learning rate is scheduled to follow a linear rule: The learning rate is decayed by a factor of 10 every 25 epochs. The total number of epochs is set to be 250. For the dataset, we apply random horizontal flipping and normalization. Note that Wilson et al. (2017) does not apply random cropping on CIAFR10; thus we omit this step to be consistent with their approach. We use cross entropy loss.

Our final accuracy is within 0.3% of Wilson et al. (2017), due to the two experiments using different random seeds.

Hyperparameter Deception in NLP — Implementing Merity et al. (2016)

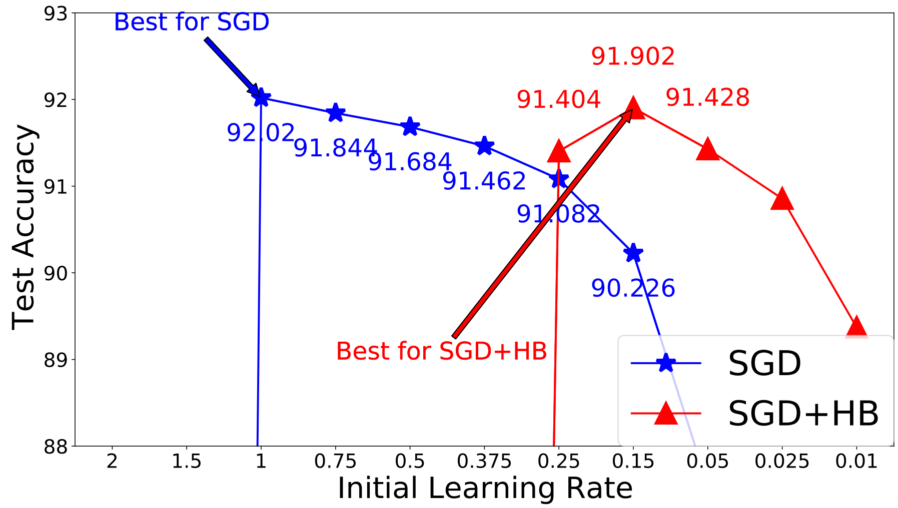


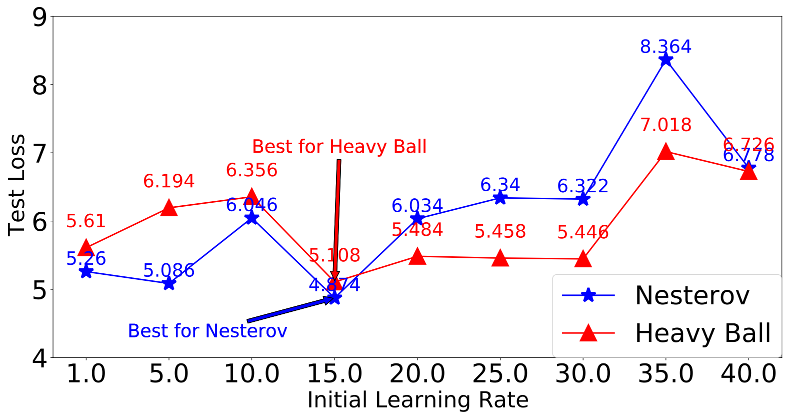
Figure 3: Demonstrating *hyperparameter deception* in Wilson et al. (2017)'s VGG16 experiment. Following their original experiments, we obtain these results by using 5 different seeds and averaging.

Experimental setup. We run this experiment on a local machine configured with a 4-core 2.6GHz Inter (R) Xeon(R) CPU, 16GB memory and an NIVIDIA GTX 2080Ti GPU. Following the exact configuration from Merity et al. (2016) and the open source implementation in the Pytorch example repository[3], we set the hidden layer size of the LSTM to be 200 and the BPTT length to be 35. We adopt 0.25 as the fine-tuned the value of gradient clipping and a dropout rate of 0.2 to the output of LSTM. We train for 50 epochs using a batch size of 20.

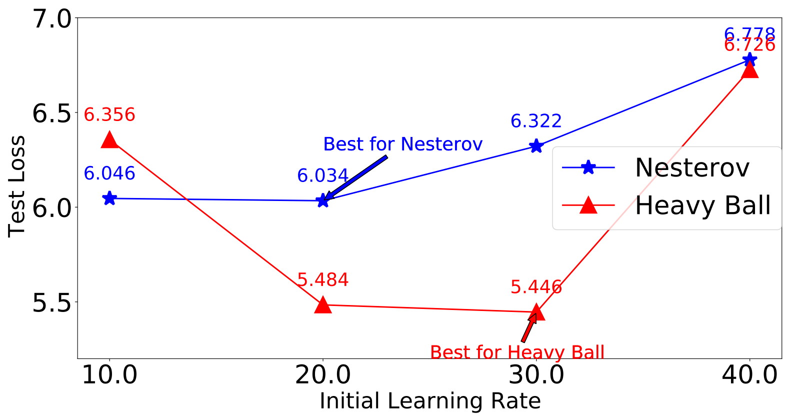
A.2.3 Related Empirical Work

Recent empirical work supports our results. While ad-hoc subsetting of the search space is a common and tacitly accepted practice among ML researchers, it can result in suboptimal performance—results that do not impart knowledge about how algorithms actually perform. Reported results tend to be impressive for some subset of the hyperparameters that the authors chose to test, but modifying HPO can lead to vastly different performance outcomes (Choi et al., 2019; Sivaprasad et al., 2020; Melis et al., 2018; Musgrave et al., 2020; Bouthillier et al., 2019).

[3]https://github.com/pytorch/examples/tree/master/word_language_model



(a) Test loss with a fine-grained grid. We conclude Nesterov performs the best.



(b) Test loss with a coarse-grained grid. We conclude that heavy ball performs the best.

Figure 4: Comparing the test loss of Nesterov acceleration and heavy ball on Wikitext-2 for different learning rates, on a fine-grained grid (a) and coarse-grained grid (b). We repeat each grid with 5 different random seeds.

A.2.4 Defense Experiments

In this section we provide more information about the implementation of a random-search-based defense to hyperparameter deception, which we discuss in Section 5.

Our Implemented Defense Algorithm

The defense we implement in our experiments is a bit different than what we describe in our theoretical results in Section 5. In particular, in practice it is easier to implement subsampling rather than resampling. We outline the algorithm we implemented below in Algorithm 1.

The defense criterion ϵ just sets our sensitivity for how we much agreement we need between the logs we've sampled in order to output a conclusion p.

Empirical Support for a Defense to Hyperparameter Deception

Now we re-run the toy experiment above, using a slightly modified defense (Algorithm 1: Instead of requiring hits on all R independent groups of trials, we use subsamples of all available trials and require hits on at least a $1 - \epsilon$ fraction of the subsamples.[4] We adopt three candidate optimizers: (1) SGD with Nesterov acceleration (Nes), (2) SGD with Heavy Ball method (HB), and (3) Adam (Kingma & Ba, 2014). The defended reasoner runs random search with 125 trials. We then take

[4]Practically speaking, it is easier to subsample than resample.

Algorithm 1 Defense with Random Search

Require: A set of K logs produced by random search $\{\mathcal{L}_i\}_{i=1}^{K}$, defense subsampling budget M, criterion constant ϵ, subsample size B.

```
for m = 1, ..., M do
    Subsample B logs: {L_i}_{i=1}^B ~ {L_i}_{i=1}^K.
    Obtain conclusions {P_i}_{i=1}^B from {L_i}_{i=1}^B.
    Obtain output conclusion for m:
        P^(m) <- Majority({P_i}_{i=1}^B)
end for
if ∃p s.t. ≥ (1 − ε)M of {P^(m)}_{i=1}^M conclude p then
    Conclude p.
else
    Conclude nothing.
end if
```

Table 1: A defense for Logistic Regression (LR) on MNIST. $p_{a,b}$ denotes the proposition $\mathcal{O}_a > \mathcal{O}_b$, i.e. *Training LR with optimizer $\mathcal{O}_a$ generalizes better than with optimizer $\mathcal{O}_b$ on MNIST.* We similarly define $p_{b,a}$. We adopt $\epsilon = 0.1$ for drawing conclusions.

Comparison	$p_{a,b}$	$p_{b,a}$	Conclude
Nes vs. HB	Nes > HB 47.52%	Nes < HB 52.48%	None
Adam vs. HB	Adam > HB 0.83%	Adam < HB 99.17%	$p_{b,a}$
Nes vs. Adam	Nes > Adam 94.91%	Nes < Adam 5.09%	$p_{a,b}$

10000 subsamples of size $K = 11$, requiring at least an at least $1 - \epsilon$ fraction of hits. We pass them to the naive reasoner, which makes conclusions based on which algorithm performed best across K trials.

Experimental setup. We run this experiment on a local machine configured with a 2.6GHz Inter (R) Xeon(R) CPU and 4GB memory. For the batch size and momentum constant, we adopt the same values as used in the Logistic Regression experiment in Appendix A.2.1. For uniform sampling within random search, we set the range to be [0.0001, 0.5].

B Modal Logic

We first provide the necessary background on modal logic, which will inform the proofs in this appendix (Appendix B.1). We then describe our possibility logic—a logic for representing the possible results of the evil demon running EHPO—and prove that it is a valid modal logic (Appendix C.1). We then present a primer on modal belief logic (Appendix C.2), and suggest a proof for the validity of combining our modal possibility logic with modal belief logic (Appendix C.3).

B.1 Axioms from Kripke Semantics

Kripke semantics in modal logic inherits all of the the axioms from propositional logic, which assigns values T and F to each atom p, and adds two operators, one for representing *necessity* ($\Box$) and one for *possibility* ($\Diamond$).

- $\Box p$ reads "It is necessary that p".
- $\Diamond p$ reads "It is possible that p".

The $\Diamond$ operator is just syntactic sugar, as it can be represented in terms of $\neg$ and $\Box$:

$$\Diamond p \equiv \neg\Box\neg p \tag{2}$$

which can be read as:

"It is possible that p" is equivalent to "It is not necessary that not p."

The complete set of rules is as follows:

- Every atom p is a sentence.
- If D is a sentence, then
  - $\neg D$ is a sentence.
  - $\Box D$ is a sentence.
  - $\Diamond D$ is a sentence.
- If D and E are sentences, then
  - $D \wedge E$ is a sentence.
  - $D \vee E$ is a sentence.
  - $D \rightarrow E$ is a sentence.
  - $D \leftrightarrow E$ is a sentence
- $\Box(\mathcal{D} \rightarrow \mathcal{E}) \rightarrow (\Box\mathcal{D} \rightarrow \Box\mathcal{E})$ (Distribution)
- $\mathcal{D} \rightarrow \Box\mathcal{D}$ (Necessitation)

B.2 Possible Worlds Semantics

Modal logic introduces a notion of *possible worlds*. Broadly speaking, a possible world represents the state of how the world *is* or potentially *could be* (Chellas, 1980; Garson, 2018). Informally, $\Box D$ means that D is true at *every* world (Equation 3); $\Diamond D$ means that D is true at *some* world (Equation 4).

Possible worlds give a different semantics from more familiar propositional logic. In the latter, we assign truth values $\{T, F\}$ to propositional variables $p \in \mathcal{P}$, from which we can construct and evaluate sentences $D \in \mathcal{D}$ in a truth table. In the former, we introduce a set of possible worlds, $\mathcal{W}$, for which each $w \in \mathcal{W}$ has own truth value for each p. This means that the value of each p can differ across different worlds w. Modal logic introduces the idea of valuation function,

$$\mathcal{V} : (\mathcal{W} \times \mathcal{D}) \rightarrow \{T, F\}$$

to assign truth values to logical sentences at different worlds. This in turn allows us to express the formulas, axioms, and inference rules of propositional logic in terms of $\mathcal{V}$. For example,

$$\mathcal{V}(w, \neg D) = T \leftrightarrow \mathcal{V}(w, D) = F$$

There are other rules that each correspond to a traditional truth-table sentence evaluation, but conditioned on the world in which the evaluation occurs. We omit these for brevity and refer the reader to Chellas (1980).

We do include the valuation rules for the $\Box$ and $\Diamond$ operators that modal logic introduces (Equations 3 & 4). To do so, we need to introduce one more concept: The accessibility relation, $\mathcal{R}$. $\mathcal{R}$ provides a frame of reference for one particular possible world to access other possible worlds; it is a way from moving from world to world. So, for an informal example, $\mathcal{R}w_1w_2$ means that w_2 is possible relative to w_1, i.e. we can reach w_2 from w_1. Such a relation allows for a world to be possible relative potentially to some worlds but not others. More formally,

$$R \subseteq \mathcal{W} \times \mathcal{W}$$

Overall, the important point is that we have a collection of worlds $\mathcal{W}$, an accessibility relation $\mathcal{R}$, and a valuation function $\mathcal{V}$, which together defines a Kripke model, which captures this system:

$$\mathcal{M} = \langle \mathcal{W}, \mathcal{R}, \mathcal{V} \rangle$$

Finally, we can give the valuation function rules for $\Box$ and $\Diamond$:

$$\mathcal{V}(w, \Box D) = T \leftrightarrow \forall w', (\mathcal{R}ww' \rightarrow \mathcal{V}(w', D) = T) \tag{3}$$

$$\mathcal{V}(w, \Diamond D) = T \leftrightarrow \exists w', (\mathcal{R}ww' \wedge \mathcal{V}(w', D) = T) \tag{4}$$

Informally, for $\Box D$ to be true in a world, it must be true in every possible world that is reachable by that world. For $\Diamond D$ to be true in a world, it must be true in some possible world that is reachable by that world.

C Our Multimodal Logic Formulation

C.1 A Logic for Reasoning about the Conclusion of EHPO

As in Section 4, we can define the well-formed formulas of our indexed modal logic recursively in Backus-Naur form, where t is any real number and P is any atomic proposition

$$\kappa := P \mid \neg\kappa \mid \kappa \wedge \kappa \mid \Diamond_t \kappa \tag{5}$$

where κ is a well-formed formula.

As we note in Section 4, where we first present this form of defining modal-logic, $\Box$ is syntactic sugar, with $\Box p \equiv \neg\Diamond\neg p$ (which remains true for our indexed modal logic). Similarly, "or" has $p \vee q \equiv \neg(\neg p \wedge \neg q)$ and "implies" has $p \rightarrow q \equiv \neg p \vee q$, which is why we do not include them for brevity in this recursive definition.

We explicitly define the relevant semantics for $\Diamond_t$ for reasoning about the demon's behavior in running EHPO. For clarity, we replicate that definition of the semantics of expressing the possible outcomes of EHPO conducted in bounded time (Definition 2, respectively) below:

Definition 4 *Let Σ denote the set of randomized **strategies** for the demon. Each $\sigma \in \Sigma$ is a function that specifies which action the demon will take: Given its current set of logs $\mathcal{L}$, either 1) running a new H with hyper-hyperparameters s (for which the demon gets a new, randomly-generated seed) 2) erasing some logs, or 3) returning.*

We can now define what the demon can reliably bring about, in terms of executing a strategy in bounded time:

Definition 5 *We let $\sigma[\mathcal{L}]$ denote the outputs of σ running, starting from $\mathcal{L}$ (i.e., the demon is given the logs in $\mathcal{L}$ to start, then gets to run a strategy σ). Let $\tau_\sigma(\mathcal{L})$ denote the total time taken to run strategy σ; this is equivalent to the sum of the times T it takes each HPO procedure $H \in \mathcal{H}$ used in the demon's strategy to run. Note that since σ is a randomized strategy (and HPO runs H are randomized as well), both $\sigma[\mathcal{L}]$ and $\tau_\sigma(\mathcal{L})$ are random variables.*

For any formula p, we say $\mathcal{L} \models \Diamond_t p$ if and only if

$$\exists \sigma \in \Sigma,\ \mathbb{P}(\sigma[\mathcal{L}] \models p) = 1 \ \wedge\ \mathbb{E}[\tau_\sigma(\mathcal{L})] \leq t.$$

C.1.1 A Possible Worlds Interpretation

Drawing on the possible worlds semantics that modal logic provides (Section B.2), we can define specific possible worlds semantics for our logic for expressing the actions of the demon in EHPO from above.

Definition 6 *A **possible world** represents the set of logs $\mathcal{L}$ the demon has produced at time $\tau_\sigma(\mathcal{L})$, i.e. after having concluded running EHPO, and the set of formulas $\mathcal{P}$ that modeled from $\mathcal{L}$ via function $\mathcal{F}$.*

Therefore, different possible worlds represent the states that *could have existed* if the evil demon had excecuted different strategies (Definition 2). In other words, if it had performed EHPO with different learning algorithms, different HPO procedures, different hyper-hyperparameter settings, different amounts of time (less than the total upper bound), different learning tasks, different models, etc... to produce a different set of logs $\mathcal{L}$ and corresponding set of conclusions $\mathcal{P}$.

In this formulation, the demon has knowledge of all possible worlds; it is trying to fool us about the relative performance of algorithms by showing as an intentionally deceptive world. Informally, moving from world to world (via an accessibility relation) corresponds to the demon running more passes of HPO to produce more logs to include in $\mathcal{L}$.

C.1.2 PROPERTIES OF THE EHPO REASONING LOGIC / LOGIC 5

We first provide some intuition concerning proving properties of non-indexed modal logic, and then show how we can derive parallel properties for our indexed modal logic, whose syntax and semantics are described above.

Proving properties of un-indexed modal logic:

$\Box$ distributes over $\wedge$

$$
\begin{aligned}
\Box(p \wedge q) \to (\Box p \wedge \Box q) & \qquad (\Box \textit{ distributes over } \wedge)\\
\textbf{Inner proof 1} & \\
p \wedge q & \\
p & \\
(p \wedge q) \to p & \\
\Box((p \wedge q) \to p) & \qquad (\textit{Necessitation})\\
\Box(p \wedge q) \to \Box p & \qquad (\textit{Distribution})\\
\Box p & \qquad (\textit{By assuming the hypothesis})\\
\textbf{Inner proof 2} & \\
p \wedge q & \\
q & \\
(p \wedge q) \to q & \\
\Box((p \wedge q) \to q) & \qquad (\textit{Necessitation})\\
\Box(p \wedge q) \to \Box q & \qquad (\textit{Distribution})\\
\Box q & \qquad (\textit{By assuming the hypothesis})\\
\Box p \wedge \Box q & \qquad (\textit{By inner proof 1, inner proof 2, assuming the hypothesis})
\end{aligned}
$$

We can show a similar result for $\Diamond$ and $\wedge$, omitted for brevity.

$\Diamond$ distributes over $\vee$

$$
\begin{aligned}
\Diamond(p \vee q) \to (\Diamond p \vee \Diamond q) & \qquad (\Diamond \textit{ distributes over } \vee)\\
\neg\Box\neg(p \vee q) \to (\Diamond p \vee \Diamond q) & \qquad (\Diamond a \equiv \neg\Box\neg a)\\
\neg\Box(\neg p \wedge \neg q) \to (\Diamond p \vee \Diamond q) & \qquad (\neg(a \vee b) \equiv (\neg a \wedge \neg b))\\
\neg(\Box\neg p \wedge \Box\neg q) \to (\Diamond p \vee \Diamond q) & \qquad (\Box \textit{ distributes over } \wedge)\\
(\neg\Box\neg p \vee \neg\Box\neg q) \to (\Diamond p \vee \Diamond q) & \qquad (\neg(a \wedge b) \equiv (\neg a \vee \neg b))\\
(\Diamond p \vee \Diamond q) \to (\Diamond p \vee \Diamond q) & \qquad (\Diamond a \equiv \neg\Box\neg a)
\end{aligned}
$$

We can show a similar result for $\Box$ and $\vee$, omitted for brevity.

Proving properties of indexed modal logic:

We remind the reader that the following are the axioms of our indexed modal logic:

$$\begin{aligned} &\vdash (p \to q) \to (\Diamond_t p \to \Diamond_t q) && \textit{(necessitation + distribution)} \\ &p \to \Diamond_t p && \textit{(reflexivity)} \\ &\Diamond_t \Diamond_s p \to \Diamond_{t+s} p && \textit{(transitivity)} \\ &\Diamond_s \Box_t p \to \Box_t p && \textit{(modal axiom 5)}, \end{aligned}$$

In short, to summarize these semantics—the demon has knowledge of all possible hyper-hyperparameters, and it can pick whichever ones it wants to run EHPO within a bounded time budget t to realize the outcomes it wants: $\Diamond_t p$ means it can realize p.

We inherit distribution and necessitation from un-indexed modal logic; they are axiomatic based on Kripke semantics. We provide greater intuition and proofs below.

Notes on necessitation for $\Box_t$:

Necessitation for our indexed necessary operator can be written as follows:

$$\vdash p \to \Box_i p$$

As we note in Section 4, $\vdash$ just means here that p is a theorem of propositional logic. So, if p is a theorem, then so is $\Box_t p$. By theorem we just mean that p is provable by our axioms (these being the only assumptions we can use); so whenever p fits this definition, we can say $\Box_t p$.

For our semantics, this just means that when p is a theorem, it is necessary that p in time t.

Distribution for $\Box_t$:

$$\Box_t (p \to q) \to (\Box_t p \to \Box_t q)$$

We provide three ways to verify distribution over implication for $\Box_t$. From this, we will prove distribution over implication for $\Diamond_t$

A. The first follows from an argument about the semantics of possible worlds from the Kripke model of our system (Sections B.2 & C.1.1).

   i. It is fair to reason that distribution is self-evident given the definitions of implication ($\to$, formed from $\neg$ and $\vee$ in our syntax for well-formed formulas for our EPHO logic, given at (5) and necessity ($\Box_t$, formed from $\neg$ and $\Diamond_t$ in our syntax for well-formed formulas for our EHPO logic, given at (5)).

   ii. Similarly, we can further support this via our semantics of possible worlds.
   We can understand $\Box_t p$ to mean, informally, that it an adversary does adopt a strategy σ that is guaranteed to cause the desired conclusion p to be the case while take at most time t in expectation. Formally, as an "necessary" analog to the semantics of $\Diamond_t$ given in Definition 2:
   For any formula p, we say $\mathcal{L} \models \Box_t p$ if and only if

   $$\forall \sigma \in \Sigma, \; \mathbb{P}(\sigma[\mathcal{L}] \models p) = 1 \; \wedge \; \mathbb{E}[\tau_\sigma(\mathcal{L})] \leq t.$$

   Given $p \to q$ is true in **all accessible worlds** (i.e, the definition of necessary), then we can say that q is true in all accessible worlds whenever p is true in all accessible worlds. As in i. above, this just follows / is axiomatic from the definitions of necessity and implication for Kripke semantics.

B. We can also prove distribution by contradiction.

i. Suppose that the distribution axiom does not hold. That is, the hypothesis

$$\Box_t(p \to q)$$

is true and the conclusion

$$\Box_t p \to \Box_t q$$

is false.

ii. By similar reasoning, from above $\Box_t p \to \Box_t q$ being false, we can say that $\Box_t p$ is true and $\Box_t q$ is false.

iii. We can use Modal Axiom M (reflexivity, proven in the next section) to say $\Box_t p \to p$. Since $\Box_t p$ is true, we can use *modus ponens* to determine that p is true.

iv. We can also say

$$\Box_t(p \to q) \to (p \to q) \qquad \textit{(By Modal Axiom M (reflexivity))}$$

v. Since we $\Box_t(p \to q)$ is true from above, we can conclude via *modus ponens* that $p \to q$ must also be true.

vi. We concluded above that p is true, so we can again use *modus ponens* and the fact that $p \to q$ is true to conclude that q is true.

vii. By necessitation, we can then also say $q \to \Box_t q$, and conclude that $\Box_t q$ is true. This is a contradiction, as above we said that $\Box_t$ is false.

viii. Therefore, by contradiction, $\Box_t(p \to q) \to (\Box_t p \to \Box_t q)$ is proved.

C. We can separately take an intuitionistic approach to verify the distribution axiom (Bezhanishvili & Holliday, 2019):

i. Let b be an **actual proof** of $p \to q$ so that we can say $a.b$ is a proof of $\Box_t(p \to q)$.

ii. Let d be an **actual proof** of p so that we can say $c.d$ is a proof of $\Box_t p$.

iii. From i. and ii., $b(d)$ is an **actual proof** of q, i.e. b (an actual proof of $p \to q$) is supplied d (an actual proof of p), and therefore can conclude q via an actual proof.

iv. From iii., we can say this results in a proof of $\Box_t q$, i.e. $e.[b(d)]$.

v. The above i.-iv. describes a function, $f : a.b \to f_{(a.b)}$. In other words, given **any proof** $a.b$ (i.e., of $\Box_t(p \to q)$) we can return function $f_{(a.b)}$, which turns **any proof** $c.d$ (i.e., of $\Box_t p$) into a proof $e.[b(d)]$ (i.e., of $\Box_t q$).

vi. $f_{(a.b)}$ is thus a proof of $\Box_t p \to \Box_t q$.

vii. From i.-vi., we gone from $a.b$ (a proof of $\Box_t(p \to q)$) to a proof of $\Box_t p \to \Box_t q$, i.e. have intuitionistically shown that $\Box_t(p \to q) \to (\Box_t p \to \Box_t q)$

Distribution and $\Diamond_t$:

We provide the following axiom in our logic:

$$\vdash (p \to q) \to (\Diamond_t p \to \Diamond_t q) \qquad (\textit{necessitation and distribution})$$

and we now demonstrate it to be valid.

$$\begin{aligned}
\vdash (p \to q) \to \Box_t(p \to q) & \qquad (\textit{necessitation}) \\
\to\ \Box_t(\neg q \to \neg p) & \qquad (\textit{modus tollens}) \\
\to\ (\Box_t \neg q \to \Box_t \neg p) & \qquad (\textit{distribution}) \\
\to\ (\neg\Box_t \neg p \to \neg\Box_t \neg q) & \qquad (\textit{modus tollens}) \\
\to\ (\Diamond_t p \to \Diamond_t q) & \qquad (\Diamond_t a \equiv \neg\Box_t \neg a)
\end{aligned}$$

This concludes our proof, for how the axioms are jointly stated.

Further, we could also say

$$(p \to q) \to \Diamond_t(p \to q) \qquad (\textit{Modal axiom M (reflexivity)})$$

And therefore also derive distribution over implication for possibility:

$$\Diamond_t(p \to q) \to (\Diamond_t p \to \Diamond_t q)$$

Modal Axiom M: Reflexivity

$$p \to \Diamond_t p$$

This axiom follows from how we have defined the semantics of our indexed modal logic (Definition 2). It follows from the fact that the demon could choose to do nothing.

We can provide a bit more color to the above as follows:

We can also derive this rule from necessitation, defined above (and from the general intuition / semantics of modal logic that necessity implies possibility). First, we can say that necessity implies possibility. We can see this a) from a possible worlds perspective and b) directly from our axioms. From a possible worlds perspective, this follows from the definition of the operators. Necessity means that there is truth at every accessible possible world, while possibility means there is truth at some accessible possible world, which puts that possible truth in time t as a subset of necessary truth in time t. From the axioms, we verify

$$\begin{aligned}
&\Box_t p \to \Diamond_t p && (\textit{Theorem to verify, which also corresponds to Modal Axiom D (serial)})\\
&\neg\Box_t p \lor \Diamond_t p && (\textit{Applying } p \to q \textit{ is equiv. } \neg p \lor q)\\
&\Diamond_t \neg p \lor \Diamond_t p && (\textit{By modal conversion, } \neg\Box_t p \to \Diamond_t \neg p)\\
& && (\textit{Which for our semantics is tautological})
\end{aligned}$$

That is, in time t it is possible that p or it is possible that p, which allows for us also to not conclude anything (in the case that the demon chooses to do nothing).

We can then say,

$$\begin{aligned}
(\Box_t p \to p) \to \Diamond_t p \quad && (\textit{By necessitation and } \Box_t p \to \Diamond_t p)\\
p \to \Diamond_t p \quad && (\textit{By concluding } p \textit{ from necessitation})
\end{aligned}$$

Another way to understand this axiom is again in terms of possible worlds. We can say in our system that every world is possible in relation to itself. This corresponds to the accessibility relation $\mathcal{R}ww$. As such, an equivalent way to model reflexivity is in terms of the following:

$$\Box_t p \to p$$

That is, if $\Box_t p$ holds in world w, then p also holds in world w, as is the case for $\mathcal{R}ww$. We can see this by proving $\Box_t p \to p$ by contradiction. Assuming this were false, we would need to construct a world w in which $\Box_t p$ is true and p is false. If $\Box_t p$ is true at world w, then by definition p is true at every world that w accesses. For our purposes, this holds, as $\Box_t p$ means that it is necessary for p to be the case in time t; any world that we access from this world w (i.e. by say increasing time, running more HPO) would require p to hold. Since $\mathcal{R}ww$ means that w accesses itself, that means that p must also be true at w, yielding the contradiction.

Modal Axiom 4: Transitivity

$$\Diamond_t \Diamond_s p \to \Diamond_{t+s} p \tag{6}$$

We can similarly understand transitivity to be valid intuitively from the behavior of the demon and in relation to the semantics of our possible worlds. We do an abbreviated treatment (in relation to what we say for reflexivity above) for brevity.

In terms of the demon, we note that in our semantics $\Diamond_t p$ means that it is possible for the demon to bring about conclusion p via its choices in time t. Similarly, we could say the same for $\Diamond_s p$; this means it is possible for the demon to bring about conclusion p in time s. If it is possible in time t that it is possible in time s to bring about p, this is equivalent in our semantics to saying that it is possible in time $t + s$ to bring about conclusion p.

We can understand this rule (perhaps more clearly) in terms of possible worlds and accessibility relations.

For worlds w_n,

$$\forall w_1, \forall w_2, \forall w_3, \mathcal{R}w_1w_2 \wedge \mathcal{R}w_2w_3 \rightarrow \mathcal{R}w_1w_3$$

In other words, this accessibility relation indicates that if w_1 accesses w_2 and if w_2 accesses w_3, then w_1 accesses w_3.

For understanding this in terms of relative possibility, we could frame this as, if w_3 is possible relative to w_2 and if w_2 is possible relative to w_1, then w_3 is possible relative to w_1. For our semantics of the demon, this means that in some time if in some time b we can get to some possible world w_3 from when we're in w_2 and in time a we can get to some possible world w_2 when we're in w_1, then in time $a + b$ we can get to w_3 from w_1

This axiom is akin to us regarding a string of exclusively possible or exclusively necessary modal operators as just one possible or necessary modal operator, respectively; we regard then regard sum of times as the amount of time it takes to bring about p (again, being necessary or possible, respectively).

Modal Axiom 5: Symmetric

$$\Diamond_s \Box_t p \rightarrow \Box_t p \tag{7}$$

We can similarly understand that our modal logic is symmetric; this is valid intuitively from the behavior of the demon. We further abbreviate our treatment for brevity. In terms of the demon, we note that in our semantics $\Diamond_s p$ means that it is possible for the demon to bring about conclusion p via its choices in time s. We can also say $\Box_t p$ means that it is necessary for the demon to bring about p in time t. If it is possible in time s that it is necessary in time s to bring about p, this is equivalent in our semantics to saying that it is necessary in time t to bring about conclusion p. In other words, we can disregard would could have possibly happened in time s from the demon's behavior and only regard what was necessary in time t for the demon to do in order to bring about p.

This axiom is akin to us just regarding the rightmost modal operator when we have a mix of modal operators applied iteratively; we can disregard what was possible or necessary in the time prior to the rightmost operator, and say that what we can say about a sentence p (whether it is possible or necessary) just relates to how much time the last operator required to bring about p.

C.2 Belief Logic

The logic of belief is a type of modal logic, called doxastic logic (Halpern et al., 2009; Rendsvig & Symons, 2019; van Benthem, 2006), where the modal operator $\mathcal{B}$ is used to express belief. Different types of reasoners can be defined using axioms that involve $\mathcal{B}$ (Smullyan, 1986).

We can formulate the doxastic logic of belief in Backus-Naur form:

For any atomic proposition P, we define recursively a well-formed formula ϕ as

$$\phi \coloneqq P \mid \neg\phi \mid \phi \wedge \phi \mid \mathcal{B}\phi \tag{8}$$

where $\mathcal{B}$ means "It is believed that ϕ". We interpret this recursively where p is the base case, meaning that ϕ is p if it is an atom, $\neg\phi$ is well-formed if ϕ is well-formed. We can also define $\vee$, $\rightarrow$, and $\leftrightarrow$ from $\neg$ and $\wedge$, as in propositional logic.

As stated in Section 4, we model a consistent Type 1 reasoner (Smullyan, 1986), which has access to all of propositional logic, has their beliefs logical closed under *modus ponens*, and does not derive contradictions. In other words, we have the following axioms:

$$\neg(\mathcal{B}p \wedge \mathcal{B}\neg p) \equiv \mathcal{B}p \rightarrow \neg\mathcal{B}\neg p$$

which is the consistency axiom,

$$\vdash p \rightarrow \mathcal{B}p$$

which is akin to Necessitation above in Section B.1 and means that we believe all tautologies, and

$$\mathcal{B}(p \rightarrow q) \rightarrow (\mathcal{B}p \rightarrow \mathcal{B}q)$$

which means that belief distributes over implication. This notably does not include

$$\mathcal{B}p \rightarrow p$$

which essentially means that we do not allow for believing p to entail concluding p. This corresponds to us actually wanting to run hyperparameter optimization before we conclude anything to be true. We do not just want to conclude something to be true based only on *a priori* information. This is akin to picking folkore parameters and concluding they are optimal without running hyperparameter optimization.

C.3 Combining Logics

It is a well known result that we can combine modal logics to make a multimodal logic (Scott, 1970). In particular, we refer the reader to results on *fusion* (Thomason, 1984).

For a brief intuition, we are able to combine our EHPO logic with belief logic since we are operating over the same set of possible worlds. The results of running EHPO produce a particular possible world, to which we apply our logic of belief in order to reason about the conclusions drawn in that world.

C.4 Our Multimodal Logic

We develop the following multimodal logic, which we also state in Section 4:

$$\psi := P \mid \neg\psi \mid \psi \wedge \psi \mid \Diamond_t\psi \mid \mathcal{B}\psi$$

C.4.1 Axioms

We give this multimodal logic semantics to express our t-non-deceptiveness axiom, which we repeat below for completeness:

For any formula p,

$$\neg\left(\Diamond_t\mathcal{B}p \wedge \Diamond_t\mathcal{B}\neg p\right)$$

We can similarly express a t-deceptiveness axiom:

For any formula p,

$$\Diamond_t\mathcal{B}p \rightarrow \neg\Diamond_t\mathcal{B}\neg p$$

To reiterate, *multimodal* just means that we have multiple different modes of reasoning, in this case our $\Diamond_t$ semantics for the demon doing EHPO and our consistent Type 1 reasoner operator $\mathcal{B}$.

Given a reasonable maximum time budget t, we say that EHPO is t-non-deceptive if it satisfies all of axioms above. Moreover, based on this notion of t-non-deceptiveness, we can express what it means to have a defense to being deceived.

C.4.2 Proving that $\mathcal{B}_*$ is non-deceptive

We provide more details on our proof summary in Section 5 for why $\mathcal{B}_*$ satisifes t-non-deceptiveness.

To recapitulate what we say in Section 5, we suppose some "naive" belief operator $\mathcal{B}_{\text{naive}}$ (based on a conclusion function $\mathcal{F}_{\text{naive}}$) that satisfies the axioms of Section 4.

We want to use $\mathcal{B}_{\text{naive}}$ to construct a new operator $\mathcal{B}_*$ that is guaranteed to be deception-free.

To do so, we define the belief operator $\mathcal{B}_*$ such that for any statement p,

$$\mathcal{B}_* p \;\equiv\; \mathcal{B}_{\text{naive}} p \;\wedge\; \neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p.$$

That is, we conclude p only if both our naive reasoner would have concluded p, and it is impossible for an adversary to get it to conclude $\neg p$ in time t.

We then say that this this enables us to show t-non-deceptiveness, following directly from the axioms in a proof by contradiction.

We curtailed the steps in Section 5 to show this. We show an unabbreviated version here of the proof by contradiction for completeness. We show by contradiction that $\mathcal{B}_*$ satisfies t-non-deceptiveness,

$$\neg \left(\Diamond_t \mathcal{B}_* p \wedge \Diamond_t \mathcal{B}_* \neg p \right)$$

So, for proof by contradiction we suppose $\Diamond_t \mathcal{B}_* p \wedge \Diamond_t \mathcal{B}_* \neg p$ is true, and then evaluate each component of the conjunction:

$$\begin{aligned}
\Diamond_t \mathcal{B}_* p &\equiv \Diamond_t \left(\mathcal{B}_{\text{naive}} p \;\wedge\; \neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p \right) && \text{(By applying } \Diamond_t \text{ to our definition of } \mathcal{B}_* p) \\
&\to \Diamond_t \left(\neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p \right) && ((a \wedge b) \to a \text{, so by distribution } \Diamond_t (a \wedge b) \to \Diamond_t a) \\
&\to \neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p && \text{(by Modal Axiom 5),} \\
\Diamond_t \mathcal{B}_* \neg p &\equiv \Diamond_t \left(\mathcal{B}_{\text{naive}} \neg p \;\wedge\; \neg \Diamond_t \mathcal{B}_{\text{naive}} p \right) && \text{(By applying } \Diamond_t \text{ to our definition of } \mathcal{B}_* \neg p) \\
&\to \Diamond_t \mathcal{B}_{\text{naive}} \neg p && ((a \wedge b) \to a \text{, so by distribution } \Diamond_t (a \wedge b) \to \Diamond_t a)
\end{aligned}$$

which yields

$$\Diamond_t \mathcal{B}_* p \wedge \Diamond_t \mathcal{B}_* \neg p \equiv \neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p \wedge \Diamond_t \mathcal{B}_{\text{naive}} \neg p$$

and $\neg \Diamond_t \mathcal{B}_{\text{naive}} \neg p \wedge \Diamond_t \mathcal{B}_{\text{naive}} \neg p$ must be false, which yields a contradiction. Therefore by contradiction we derive the t-non-deceptiveness axiom for $\mathcal{B}_*$.

While this analysis shows that some sort of defense is always possible, it may not be practical to compute $\mathcal{B}_*$ as defined here because we cannot easily evaluate whether $\Diamond_t \mathcal{B}_{\text{naive}} \neg p$.

D Showing Grid Search Entails Deceptive EHPO

We return to our empirical demonstration of hyperparameter deception in Section 2, and provide an intuition for characterizing what we observe in terms of the demon using a strategy σ to deceive us about the conclusions of EHPO (See Appendix D for more formal results). We run EHPO twice, using two strategies σ_1 and σ_2. For σ_1, there is one HPO procedure $H \in \mathcal{H}$, which is grid search with a powers-of-two grid (Figure 2a) to produce one log $\ell \in L_1$. The total time to run σ_1, $\tau_{\sigma_1}(\mathcal{L}_1)$, was $\sim$2 hours and 20 minutes. We have a similar setup for σ_2, using a coarser powers-of-ten grid (Figure 2b), where $\tau_{\sigma_2}(\mathcal{L}_2)$ was $\sim$1 hour and 10 minutes, which we deem to be reasonable HPO time budgets. We denote p to be "Training LR with Nesterov performs better than with heavy ball on MNIST." $\mathcal{F}$, which maps from logs to conclusions, can be as naive as checking which algorithm yields the best overall test accuracy. For this example, we additionally test for statistical significance. Based on the results of running σ_1, we conclude p (Figure 2a); for σ_2 we conclude $\neg p$ (Figure 2b). This violates the t-non-deceptive axiom. In other words, when using grid search in EHPO with different grids, we could conclude the inconsistent results p and $\neg p$ within a reasonable time budget.

We note that without additional assumptions (e.g. Lipschitz continuity), it is not possible to build a defense using grid search. This is why we focus our defense strategy on random search.

E Validating Defenses to Hyperparameter Deception

Suppose we are considering HPO via random search (Bergstra et al., 2011), in which the set of allowable hyper-hyperparameters contains tuples (μ, M), where μ is a distribution over all possible hyperparameter sets Λ and M is the number of different hyperparameter configuration trials to run. This set S is the Cartesian product of the set of allowable distributions D ($\mu \in D$) and M.

Suppose that for any two allowable distributions $\mu, \nu \in D$ and any event A (a measurable subset of Λ), $\mu(A) \leq e^{\gamma} \cdot \nu(A)$ (i.e., the Renyi ∞-divergence between any pair of distributions is at most γ). This bounds how much the choice of hyper-hyperparameter can affect the hyperparameters in HPO.

We also suppose we start from a naive reasoner (expressed via the operator $\mathcal{B}_{\text{naive}}$), which draws conclusions based on a log with K trials. For this scenario, we are only concerned with the reasoner's conclusions from K-trial logs. We therefore assume w.l.o.g. that the reasoner draws no conclusions unless presented with exactly one log with exactly K trials.

For some constant $R \in \mathbb{N}$, we construct a new reasoner $\mathcal{B}_*$ that does the following: It draws conclusions only from a single log with exactly KR trials (otherwise it concludes nothing). To evaluate a proposition p, it splits the log into R groups of K trials each, evaluates $\mathcal{B}_{naive}$ on p on each of those R groups, and then concludes p only if $\mathcal{B}_{naive}$ also concluded p on all R groups.

Now consider a particular (arbitrary) proposition p. Since $\mathcal{B}_*$ draws conclusions based on only a single log, any strategy σ for the demon is equivalent to one that maintains at most one log at all times (the "best" log it found so far for its purposes, as it can discard the rest).

Let Q be the supremum, taken over all allowable distributions μ, of the probability that running a group of K random search trials using that distribution will result in a log that would convince the $\mathcal{B}_{naive}$ of p. Similarly, let $Q_{\neg}$ be the supremum, taken over all allowable distributions ν, of the probability that running a group of K trials using that distribution will result in a log that would convince $\mathcal{B}_{naive}$ of $\neg p$.

Observe that Q is the probability of an event in a product distribution of K independent random variables each distributed according to μ, and similarly for $Q_{\neg}$, and the corresponding events are disjoint. By independent additivity of the Renyi divergence, the Renyi ∞-divergence between these corresponding product measures will be γK. It follows that

$$1 - Q \geq \exp(-\gamma K) Q_{\neg}$$

and

$$1 - Q_{\neg} \geq \exp(-\gamma K) Q$$

From here it's fairly easy to conclude that

$$Q + Q_{\neg} \leq \frac{2}{1 + \exp(-\gamma K)}.$$

Now, an EHPO procedure using random search with KR trials will convince $\mathcal{B}_*$ of p with probability Q^R, since all R independently sampled groups of K trials must "hit" and each hit happens with probability Q. Thus, the expected time it will take the fastest strategy to convince us of p is $Q^{-R} \cdot KR$. Similarly, the fastest strategy to convince us of $\neg p$ takes expected time $Q_{\neg}^{-R} \cdot KR$.

Suppose now, by way of contradiction, that the t-non-deceptiveness axiom is violated, and there are strategies that can achieve both of these in time at most t. That is,

$$Q^{-R} \cdot KR \leq t \qquad \text{and} \qquad Q_{\neg}^{-R} \cdot KR \leq t.$$

From here, it's fairly easy to conclude that

$$Q + Q_{\neg} \geq 2 \left(\frac{KR}{t} \right)^{1/R}.$$

Combining this with our conclusion above gives

$$\left(\frac{KR}{t}\right)^{1/R} \leq \frac{1}{1+\exp(-\gamma K)}.$$

It's clear that we can cause this to be violated by setting R to be large enough. Observe that

$$\frac{1}{1+\exp(-\gamma K)} \leq \exp(-\exp(-\gamma K)),$$

so

$$\left(\frac{KR}{t}\right)^{1/R} \leq \exp(-\exp(-\gamma K)).$$

Taking the root of both sides gives

$$\left(\frac{KR}{t}\right)^{\frac{1}{R\exp(-\gamma K)}} \leq \frac{1}{e}.$$

Finally, substitute

$$\beta = R\exp(-\gamma K).$$

Then, taking the root of both sides gives

$$\left(\frac{\beta K}{t\exp(-\gamma K)}\right)^{1/\beta} \leq \frac{1}{e}.$$

Finally, set

$$\beta = \sqrt{\frac{t\exp(-\gamma K)}{K}}.$$

This gives

$$\left(\frac{1}{\beta}\right)^{1/\beta} \leq \frac{1}{e}.$$

But this is impossible, as the minimum of x^x occurs above $1/e$. This gives a concrete value of R as

$$R = \beta\exp(\gamma K) = \sqrt{\frac{t\exp(\gamma K)}{K}} = O(\sqrt{t}).$$

This shows that, for this task, if we run our constructed EHPO with $R = O(\sqrt{t})$ assigned in this way, it will be guaranteed to be t-non-deceptive.