Testing for Overfitting

James Schmidt*
Johns Hopkins University Applied Physics Laboratory
Johns Hopkins University

April 5, 2023

Abstract

High complexity models are notorious in machine learning for overfitting, a phenomenon in which models well represents data but fail to generalize to an underlying data generating process. A typical procedure for heuristically circumventing overfitting is to compute empirical risk on a holdout set and halt once (or flag that/when) it begins to increase. While such practice often helps to output a well-generalizing model, the impetus is widely understood as a heuristic salve. We introduce and argue for a statistically grounded hypothesis test by means of which overfitting may be quantitatively defined and detected. In line with concentration inequalities, we rely on the prominent result that empirical means with high probability should be close to their true mean to conclude that they should be close to each other. We stipulate conditions under which this test is valid, explain why training data alone is insufficient for generalization determination, how the test may be used for identifying overfitting, and articulate a further nuance with respect to which this test may flag distribution shift.

1 Introduction

Supervised machine learning is severely underdetermined: a finite labeled data set is used to search a function space for an appropriate model fitting both the data and "from where the data comes." While the full function space is often at least two infinite orders of magnitude greater than the data, practitioners usually restrict search to a hypothesis class that is parametrized as a finite dimensional space. If this hypothesis class is too restricted, the search may output a model which fails to represent or approximate the data well enough; if, on the other hand, the class is too rich, the output model may represent the data *too* well, in that the model fails to represent the underlying distribution from which data is drawn. Generally, this tradeoff between *underfitting* and *overfitting*, respectively, is asymmetric: a model which fits data may still generalize to the underlying distribution, while a model which underfits data usually does not the distribution. Said differently, underfitting is *detectable* in the course of performance evaluation while overfitting cannot be identified by performance on the training data alone ([4]).

To check for and guard against overfitting, standard practice sets aside a holdout set disjoint from training to compute performance separately. Isolating this computation to another dataset mitigates the aspect blindness of training data performance to overfitting by decoupling the optimization process, which we will see compromises iid assumptions, from evaluation. Thus, an ordinary training procedure incorporates two distinct steps: 1. optimization with training data to fit data and 2. verify generalization by evaluating performance on holdout data. While vague heuristics motivating this two-step procedure abound in the literature and research community, rigorous statistical rationale less ubiquitously accompany justification of its use.

*email: aschmi40@jhu.edu

Moreover, this two step process treats training data and holdout data as altogether different kinds of things, with different tasks and different intended uses. As such, separating the conclusions we draw from training data and holdout data threatens to undermine the original impetus according to which training data is used for training in the first place, namely to optimize some function for the express purpose of optimizing its *expectation*.

We explain the reasons for this paradox, and propose a solution which translates into a statistical test which may be deployed for both defining and identifying overfitting, using modified Law of Large Numbers intuition that empirical means should approximate their expectation. In section 2, we review requisite background for the supervised learning problem, discuss the problem with training data, how it relates to overfitting, and why we would still like to use model performance with respect to training data to asses generalization. In section 3, we detail the statistical test for achieving this end, and give commentary on how this test elucidates the *meaning* of overfitting. We end with some plots in section 4 illustrating the use of the test in simulation.

2 Technical Background

2.1 Supervised Machine Learning

The setting for a supervised machine learning problem starts with the following data:

1. a joint probability space $(\mathcal{X} \times \mathcal{Y}, \mathbb{P}_{\mathcal{X}\times\mathcal{Y}})$,
2. labeled data $S = \big((x_1, y_1), \dots, (x_m, y_m)\big) \in (\mathcal{X} \times \mathcal{Y})^\omega := \bigcup_{m\in\mathbb{N}} (\mathcal{X} \times \mathcal{Y})^m$,
3. a hypothesis class $\mathcal{H} \subset \mathcal{Y}^{\mathcal{X}}$, usually finite dimensional, elements $\tilde{y} \in \mathcal{H}$ of which are called *models*, and
4. a cost function generator $c : \mathcal{H} \to \mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$ mapping a model $\tilde{y}$ to random variable $c_{\tilde{y}}$, whose output $c_{\tilde{y}}(x, y)$ on input (x, y) is a measure of fit between prediction $\tilde{y}(x)$ and label y.

The goal is to concoct an *algorithm* $\hat{y}_{(\cdot)} : (\mathcal{X} \times \mathcal{Y})^\omega \to \mathcal{H}$ which outputs a model $\hat{y}_S$ with small expected cost

$$\mathbb{E}(c_{\hat{y}_S}) \approx \inf_{\tilde{y}\in\mathcal{H}} \mathbb{E}(c_{\tilde{y}}).$$

The measure $\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ generating data (x_i, y_i) is usually unknown, and data S is used to proxy approximate expectation and to optimize with respect to the expected risk function

$$\begin{array}{rccc} \mathbb{E}(c_{(\cdot)}) : & \mathcal{H} & \to & \mathbb{R} \\ & \tilde{y} & \mapsto & \mathbb{E}(c_{\tilde{y}}). \end{array} \tag{1}$$

The standard algorithm for optimizing expectation is empirical risk minimization, namely

$$\hat{y}_S \in \arg\min_{\tilde{y}\in\mathcal{H}} e_S(\tilde{y}), \tag{2}$$

where

$$e_S(\tilde{y}) := \frac{1}{|S|} \sum_{(x,y)\in S} c_{\tilde{y}}(x, y). \tag{3}$$

Law of Large Numbers intuition suggests that

$$e_S(\tilde{y}) \approx \mathbb{E}(c_{\tilde{y}}) \tag{4}$$

when $|S|$ is large, so supposing as much, an output $\hat{y}_S$ of eq. (2) ought to be a close approximation of the true goal, in the sense that

$$\mathbb{E}(c_{\hat{y}_S}) \approx \inf_{\tilde{y}\in\mathcal{H}} \mathbb{E}(c_{\tilde{y}}). \tag{5}$$

To the extent that a model $\tilde{y} \in \mathcal{H}$ (approximately) satisfies eq. (4), we say that the model *generalizes* (ε-generalizes if the error in approximation is bounded by ε), and to the extent that models in

$\mathcal{H}$ can be guaranteed to generalize optimality $\inf_{\tilde{y}} \mathbb{E}(c_{\tilde{y}})$, we say that $\mathcal{H}$ is some kind of *learnable*. The familiar and formal notion of *probably approximately correct* (PAC) learnability, for example, extends guarantees of concentration bounds to an optimization (over $\mathcal{H}$) context, and defines $\mathcal{H}$ as PAC learnable if there is a sample complexity $\mu : (0,1)^2 \to \mathbb{N}$ for which $\hat{y}_S$ may be guaranteed to ε-generalize with at least $1-\delta$ probability as long as $|S| > \mu(\varepsilon, \delta)$ ([10] [5]). Properly quantifying the character and richness of $\mathcal{H}$ (as captured, e.g., by VC dimension) demarcates learnability conditions, and various theoretical results exist providing such guarantees.

2.2 Overfitting and Genearlization

Absent formal learnability guarantees, it turns out that LLN reasoning is not sufficient for ensuring generalization. The reasons are multifarious but substantively turn around currying ([7, §2.3]) of the cost function generator $c : \mathcal{H} \to \mathbb{R}^{(\mathcal{X}\times\mathcal{Y})}$. For a *fixed* $\tilde{y} \in \mathcal{H}$, the map $c_{\tilde{y}} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is a random variable, and therefore defines a measure $\mathbb{P}_{\mathbb{R}}$ on $\mathbb{R}$ by $\mathbb{P}_{\mathbb{R}}([a,b]) := \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}\big(c_{\tilde{y}}^{-1}([a,b])\big)$. This means, among other things, given data $S = \big((x_1,y_1),\ldots,(x_m,y_m)\big) \sim_{\text{iid}} \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$, that $c_{\tilde{y}}(S) := \big(c_{\tilde{y}}(x_1,y_1),\ldots,c_{\tilde{y}}(x_m,y_m)\big) \sim_{\text{iid}} \mathbb{P}_{\mathbb{R}}$. And independence invites valid conclusions of various concentration results.

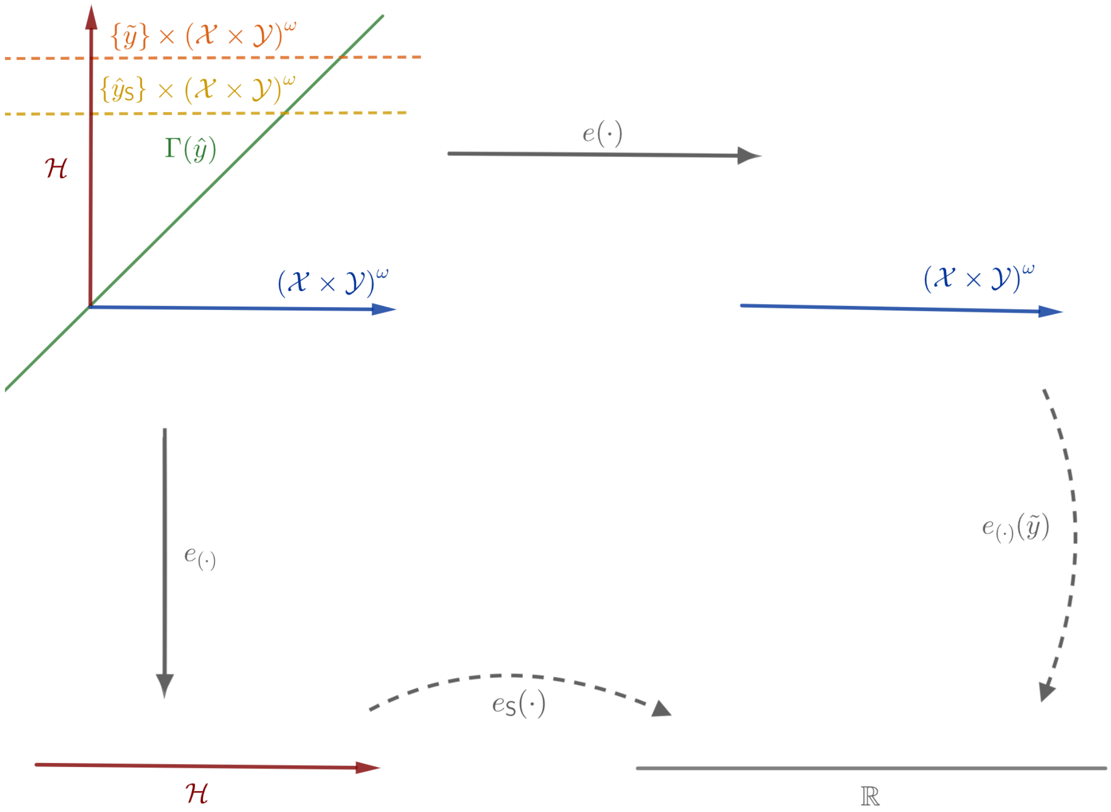
$\{\tilde{y}\} \times (\mathcal{X}\times\mathcal{Y})^\omega$
$\{\hat{y}_S\} \times (\mathcal{X}\times\mathcal{Y})^\omega$
$\Gamma(\hat{y})$
$\mathcal{H}$
$(\mathcal{X}\times\mathcal{Y})^\omega$
$e(\cdot)$
$(\mathcal{X}\times\mathcal{Y})^\omega$
$e_{(\cdot)}$
$e_{(\cdot)}(\tilde{y})$
$e_S(\cdot)$
$\mathcal{H}$
$\mathbb{R}$

Figure 1: Slicing the Cost Function $c : \mathcal{H} \times (\mathcal{X}\times\mathcal{Y})^\omega \to \mathbb{R}$

Searching over a function space, in the supervised machine learning setting, adds complications to otherwise innocuous independence conclusions. For the learning algorithm $\hat{y} : (\mathcal{X} \times \mathcal{Y})^\omega \to \mathcal{H}$ first fixes *data* $S \in (\mathcal{X}\times\mathcal{Y})^\omega$ in search of a certain minimum with respect to *this* data. Given different data, the algorithm outputs a different model. Formally, the *curried* cost generator $c_{(\cdot)} : \mathcal{H} \to \mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$ defines an empirical risk generator $e(\cdot) : \mathcal{H} \to \mathbb{R}^{(\mathcal{X}\times\mathcal{Y})^\omega}$ defined by sending $\tilde{y} \mapsto e_{(\cdot)}(\tilde{y})$, the latter of which is defined by mapping data $S \in (\mathcal{X}\times\mathcal{Y})^\omega$ to $e_S(\tilde{y})$ (eq. (3)), and with respect to which LLN reasoning and the like may properly apply. The optimization procedure, however, flips the currying around: fixing data $(x,y) \in \mathcal{X}\times\mathcal{Y}$, we have a cost on models

$c_{(\cdot)}(x,y) : \mathcal{H} \to \mathbb{R}$ defined by $\tilde{y} \mapsto c_{\tilde{y}}(x,y)$, which extends to empirical risk $e_S(\cdot) : \mathcal{H} \to \mathbb{R}$ mapping model $\tilde{y} \mapsto e_{\tilde{y}}(S)$, instantiating the curried function $e_{(\cdot)} : \left(\mathcal{X} \times \mathcal{Y}\right)^{\omega} \to \mathbb{R}^{\mathcal{H}}$.[1]

It turns out that order of operations matter. To distill the overfitting problem, we consider uncurried versions $\mathcal{H} \times (\mathcal{X} \times \mathcal{Y})^{\omega} \overset{c_{(\cdot)}(\cdot)}{\underset{e_{(\cdot)}(\cdot)}{\rightrightarrows}} \mathbb{R}$ of the cost and empirical risk functions (see fig. 1), thereby deprioritizing either data $S \in (\mathcal{X} \times \mathcal{Y})^{\omega}$ or model $\tilde{y} \in \mathcal{H}$, and define a diagonal of sorts,

$$\Gamma(\hat{y}) := \left\{(\tilde{y}, S) \in \mathcal{H} \times (\mathcal{X} \times \mathcal{Y})^{\omega} : \tilde{y} = \hat{y}_S\right\}.$$

Explicitly, $\Gamma(\hat{y})$ is the pullback of diagram $\mathcal{H} \times (\mathcal{X} \times \mathcal{Y})^{\omega} \overset{\hat{y}_{(\cdot)}\circ\pi_2}{\underset{\mathrm{id}_{\mathcal{H}}\circ\pi_1}{\rightrightarrows}} \mathcal{H}$. Situating the point that evaluating empirical performance for a model $\hat{y}_S$ with training data S lives in $\Gamma(\hat{y})$ elucidates why performance of training data is "aspect blind" to overfitting: Law of Large Numbers reasoning does not apply in this regime.

Consider that a sequence of datasets

$$\begin{array}{rcl} S_1 & = & (x_1, y_1) \in (\mathcal{X} \times \mathcal{Y})^1, \\ S_2 & = & \big(S_1, (x_2, y_2)\big) \in (\mathcal{X} \times \mathcal{Y})^2, \\ & \vdots & \\ S_m & = & \big(S_{m-1}, (x_m, y_m)\big) \in (\mathcal{X} \times \mathcal{Y})^m, \\ & \vdots & \end{array}$$

with each $S_j \sim_{\mathrm{iid}} \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$, induces a sequence of models

$$\hat{y}_{S_1}, \hat{y}_{S_2}, \ldots, \hat{y}_{S_m}, \ldots \in \mathcal{H}.$$

The sequence of models consequently induces a sequence of finite sequences of costs

$$\begin{array}{rcl} c_{\hat{y}_{S_1}}(S_1) & = & c_{\hat{y}_{S_1}}(x_1, y_1) \in \mathbb{R}, \\ c_{\hat{y}_{S_2}}(S_2) & = & \big(c_{\hat{y}_{S_2}}(x_1, y_1), c_{\hat{y}_{S_2}}(x_2, y_2)\big) \in \mathbb{R}^2, \\ & \vdots & \\ c_{\hat{y}_{S_m}}(S_m) & = & \big(c_{\hat{y}_{S_m}}(x_1, y_1), \ldots, c_{\hat{y}_{S_m}}(x_m, y_m)\big) \in \mathbb{R}^m, \\ & \vdots & \end{array}$$

which clearly is not guaranteed to be iid, unless miraculously the cost functions

$$c_{\hat{y}_{S_1}}, c_{\hat{y}_{S_2}}, \ldots, c_{\hat{y}_{S_m}}, \ldots$$

all induce the same measure $\mathbb{P}_{c_{\hat{y}}(\mathcal{X}\times\mathcal{Y})}$ on $\mathbb{R}$, for which there is no apriori reason to suppose. Thus, one may of course place an appropriate measure $\mathbb{P}_{\Gamma}$ on $\Gamma(\hat{y})$—in fact, this pullback naturally inherits from measures on $(\mathcal{X} \times \mathcal{Y})^{\omega}$—and certainly iid samples $S \in (\mathcal{X} \times \mathcal{Y})^{\omega}$ induce iid samples $\hat{y}_S \sim_{\mathrm{iid}} \mathbb{P}_{\Gamma}$, but statements of events on this set do not extend to iid conditions on sequences of costs. For such, we must isolate our attention to slices $\{\hat{y}_S\} \times (\mathcal{X} \times \mathcal{Y})^{\omega}$ (see fig. 2), for which sample $S' = \big((x'_1, y'_1), \ldots, (x_k, y'_k)\big) \sim_{\mathrm{iid}} \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ induces truly independent and identically distributed sample $\big(c_{\hat{y}_S}(x'_1, y'_1), \ldots, c_{\hat{y}_S}(x'_k, y'_k)\big) \sim_{\mathrm{iid}} \mathbb{P}_{c_{\hat{y}_S}(\mathcal{X}\times\mathcal{Y})}$.

While we cannot rely on empirical risk $e_S(\hat{y}_S)$ by itself to relay generalization performance, we *may* in concert with $e_{S'}(\hat{y}_S)$ for some *other* data $S' \in (\mathcal{X} \times \mathcal{Y})^{\omega}$, usually called a holdout or validation set. Typically performance at each training stage is evaluated on the holdout set, and early stopping conditions verify that validation performance continues to improve [6]. An onset

[1] The reversal or roles in subscripts between c and e is unfortunate, but otherwise reflective of the primary purpose of each function, namely that $c_{\tilde{y}}$ measures performance of model $\tilde{y}$ on a datapoint $(x, y) \in \mathcal{X} \times \mathcal{Y}$ while e_S measures empirical risk of fixed data on a model $\tilde{y} \in \mathcal{H}$.

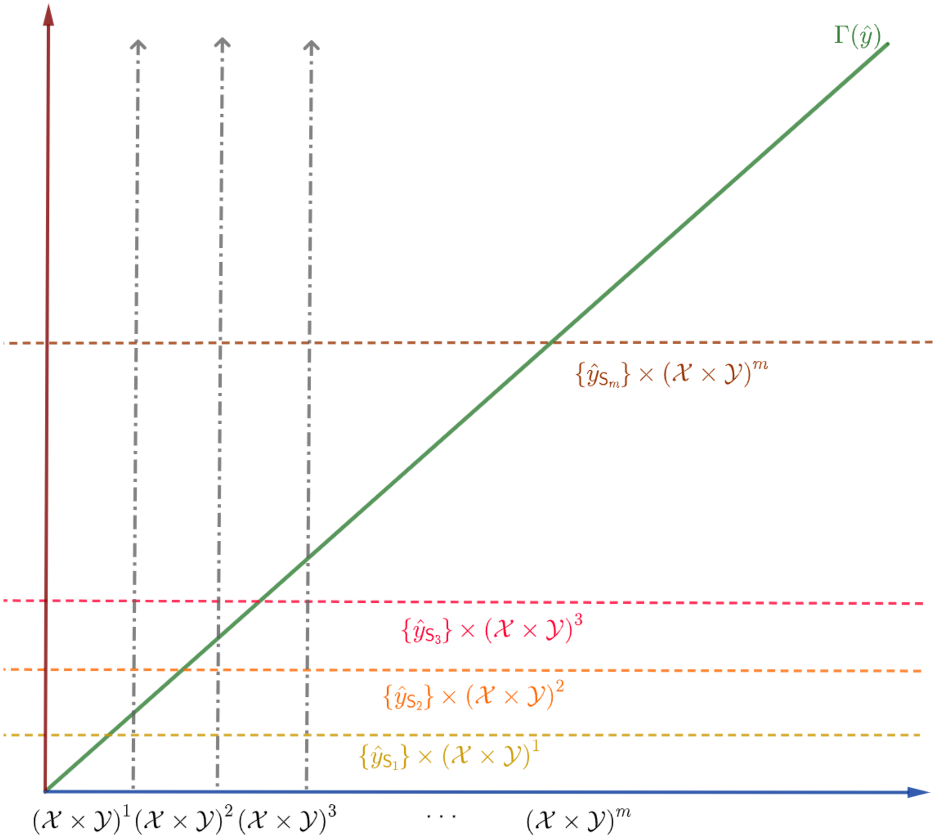
$\Gamma(\hat{y})$
$\{\hat{y}_{S_m}\} \times (\mathcal{X} \times \mathcal{Y})^m$
$\{\hat{y}_{S_3}\} \times (\mathcal{X} \times \mathcal{Y})^3$
$\{\hat{y}_{S_2}\} \times (\mathcal{X} \times \mathcal{Y})^2$
$\{\hat{y}_{S_1}\} \times (\mathcal{X} \times \mathcal{Y})^1$
$(\mathcal{X} \times \mathcal{Y})^1$
$(\mathcal{X} \times \mathcal{Y})^2$
$(\mathcal{X} \times \mathcal{Y})^3$
$\cdots$
$(\mathcal{X} \times \mathcal{Y})^m$

Figure 2: Sequences of models $\hat{y}_{S_1}, \hat{y}_{S_2}, \dots, \hat{y}_{S_m}, \dots$

of validation performance degradation can be interpreted as indication of overfitting. Illustrations of overfitting in the literature (e.g. [1], [4], [11]) display performance on training data compared with performance on holdout data, often parameterized by model complexity or training step. Notable in such graphics is a general profile *shape* with little attention paid to how close one ought to expect both curves to be.

Many considerations of overfitting consider performance across various models ([9], [8]). By contrast, we introduce a statistical test, for a fixed model, based on classic concentration inequalities, with respect to which overfitting may *quantitatively* be defined, relying on comparison of validation performance to training set performance. We argue that because model construction uses and depends on (minimization with respect to $\tilde{y}$ of) $e_S(\tilde{y})$, we should be able to conclude performance with *it*. In fact, comparison against empirical risk $e_S(\hat{y}_S)$ provides an anchor against which we may draw rigorous statistical conclusions. The test we provide amounts to much of the same as common stopping criteria for training, though the reasoning we give is both grounded in the math and provides threads for distinguishing causes of error.

3 Detecting Overfitting

3.1 The Test

We focus on the case where cost $c_{(\cdot)} : \mathcal{H} \times (\mathcal{X} \times \mathcal{Y}) \to \mathbb{R}$ is bounded as $c_{\hat{y}_S} \subset [0, 1]$, such as most classification problems or restricted classes of regression problems. In this case, Hoeffding-like bounds abound and we expect that

$$\mathbb{P}_{(\mathcal{X}\times\mathcal{Y})^k}\left(\left|\mathbb{E}(c_{\hat{y}_S}) - e_{(\cdot)}(\hat{y}_S)\right| > \varepsilon\right) < 2e^{-2\varepsilon^2 k}. \qquad (6)$$

In other words, for independently and identically distributed sampled data $S' \in (\mathcal{X} \times \mathcal{Y})^k$, $e_{S'}(\hat{y}_S) \approx \mathbb{E}(c_{\hat{y}_S}) \pm \varepsilon$ with probability at least $1 - e^{-2\varepsilon^2/k}$.[2] While $S \in (\mathcal{X} \times \mathcal{Y})^m$ is also drawn independently, by assumption, we cannot quite conclude the same of $e_S(c_{\hat{y}_S})$ because (as discussed above) with respect to the $c_{\hat{y}_S}$-induced measure on $\mathbb{R}$, the sequence $(c_{\hat{y}_S}(x_1, y_1), \dots, c_{\hat{y}_S}(x_m, y_m))$ is not. We may, however, suppose that a *consequence* of independence holds, namely that

$$|\mathbb{E}(c_{\hat{y}_S}) - e_S(\hat{y}_S)| < \varepsilon/2, \tag{7}$$

and use this (possibly counterfactual) supposition to test its truth. While possibly counterintuitive, a bound of the form in (7) is exactly what we desire from a generalizing model $\hat{y}_S$.

Proposition 3.1 (Test for Overfitting)**.** Suppose that bound (7) holds. Then

$$\mathbb{P}_{(\mathcal{X}\times\mathcal{Y})^k}\left(\left|e_S(\hat{y}_S) - e_{S'}(\hat{y}_S)\right| > \varepsilon\right) \le 2e^{-\frac{\varepsilon^2 k}{2}}. \tag{8}$$

Therefore, the null hypothesis that trained model $\hat{y}_S$ $\frac{\varepsilon}{2}$-generalizes may be tested using probability bound eq. (8).

Proof. Since

$$\begin{aligned} \left|e_S(\hat{y}_S) - e_{S'}(\hat{y}_S)\right| &= \left|e_S(\hat{y}_S) - \mathbb{E}(c_{\hat{y}_S}) + \mathbb{E}(c_{\hat{y}_S}) - e_{S'}(\hat{y}_S)\right| \\ &\le \left|e_S(\hat{y}_S) - \mathbb{E}(c_{\hat{y}_S})\right| + \left|\mathbb{E}(c_{\hat{y}_S}) - e_{S'}(\hat{y}_S)\right| \\ &< \tfrac{\varepsilon}{2} + \left|\mathbb{E}(c_{\hat{y}_S}) - e_{S'}(\hat{y}_S)\right| \end{aligned}$$

we conclude

$$\left\{\left|e_S(\hat{y}_S) - e_{(\cdot)}(\hat{y}_S)\right| > \varepsilon\right\} \subseteq \left\{\left|\mathbb{E}(c_{\hat{y}_S}) - e_{(\cdot)}(\hat{y}_S)\right| > \frac{\varepsilon}{2}\right\}.$$

Inclusion of events implies inequality of measures, and we apply Hoeffding (inequality (9)) to bound the right hand side probability $\mathbb{P}\left(\left|\mathbb{E}(c_{\hat{y}_S}) - e_{(\cdot)}(\hat{y}_S)\right| > \frac{\varepsilon}{2}\right)$. □

The test allows us to quantitatively define overfitting.

Definition 3.1. Let $S \sim_{\text{iid}} \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ and $\hat{y}_S \in \mathcal{H}$. We say that $\hat{y}_S$ *ε-overfits* S if

$$e_S(\hat{y}_S) < \mathbb{E}(c_{\hat{y}_S}) - \varepsilon.$$

Notice that use of holdout data for evaluation by itself provides an absolute approximation of performance, while in tandem with training data, we gain quantified (un)certainty about generalization. Finally, the probability in (8) depends on the size of data of validation data, but not on the size of training data. This conclusion is correct: while we would like more training data to correlate with higher likelihood of performance, the problem in section 2.2 indicates that such intuition may not find a straightforward grounding in probability. Presumably, one may be less inclined to hypothesize satisfactory model performance when training with little data, but we cannot speak confidently beyond personal proclivities.

3.2 Interpreting the Output

Overfitting is a heuristic notion which suggests a model has fit the data and not the distribution which generated it. On closer inspection, however, the test we propose does not provide indication of *only* overfitting. In fact, the supposition of generalization is one with respect to a certain (fixed) distribution; this test thus additionally assumes that the test data $S' \sim_{\text{iid}} \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ as well. It may not. For there may be some form of distributional shift according to which $S' \sim_{\text{iid}} \mathbb{P}'_{\mathcal{X}\times\mathcal{Y}}$, in which case we cannot guarantee inquality (8), at least not if the expectation $\mathbb{E}(c_{\hat{y}_S})$ is computed with respect to the original measure $d\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$. In other words, instantiation of event $\left\{\left|e_S(\hat{y}_S) - e_{(\cdot)}(\hat{y}_S)\right| > \varepsilon\right\}$ by inequality $|e_S(\hat{y}_S) - e_{S'}(\hat{y}_S)| > \varepsilon$ may suggest:

1. an unlikely sample S' was received (all the hypotheses hold),

[2] The fact that $\mathbb{E}(c_{(\cdot)})$ and $e(\cdot)$ both take $\hat{y}_S$ as argument is irrelevant: the bound holds for any $\tilde{y} \in \mathcal{H}$.

2. $\hat{y}_S$ does not generalize $\mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ with respect to $c_{(\cdot)}$ (overfitting), or
3. $S' \not\sim_{\mathrm{iid}} \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$ (possible distributional shift).

It is important when running a statistical test to respect the scope of what it purports to evaluate: namely, *if* a set of assumptions hold—in this case 1. that $\hat{y}_S$ $\frac{\varepsilon}{2}$-generalizes (eq. (7)) and 2. $S' \sim_{\mathrm{iid}} \mathbb{P}_{\mathcal{X}\times\mathcal{Y}}$—then the probability that a certain kind of event occurs is bounded by some value which is explicitly calculable. Realization of the unlikely and unlucky event by S' can either mean S' really is unlucky or that one of the assumptions fails.

While this test is expressed with respect to the cost function $c_{\hat{y}}$ or $c_{\hat{y}_S}$, it need not be so limited. In fact, any map $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ could be used to probe the distribution, substituting the appropriate concentration inequality depending on the range of f. When $f(\mathcal{X} \times \mathcal{Y})$ is bounded, we may rely on a version of Hoeffding, which converges exponentially. Subsequent work will investigate the use of *random projections* to examine distribution shift and uncertainty quantification, as a means of testing to eliminate or isolate the above obfuscating condition #3.

4 In Simulation

Code implementing this test can be found at https://github.com/schmidttgenstein/qudos.git. In each of the following illustrations, we plot empirical performance (accuracy for a binary classification problem) with respect to both training and holdout data on the left, and the absolute difference on the right. These curves are plotted against training epoch, and each pair uses a different size for S' with respect to which either probability or precision depend (appendix A). Fixing the confidence at $p = 0.0075$ (so that we sample a set satisfying error less than precision with probability at least $1 - 0.0075$), we denote the precision with the dashed red line in the right figures. Per Hoeffding's inequality (9), this precision may be made finer with more data. Worth noting that the test in proposition 3.1 does not intrinsically relate to early stopping: a model may overfit and cease to overfit at various epochs in training (see, e.g., fig. 3).

Results in fig. 3 and fig. 4 use generated data and a multilayer perceptron for binary classification. Results in fig. 5 and fig. 6 use a ResNet18 architecture network for subsetted functional map of the world data, filtered for binary classification ([3]).

4.1 Simulated Data

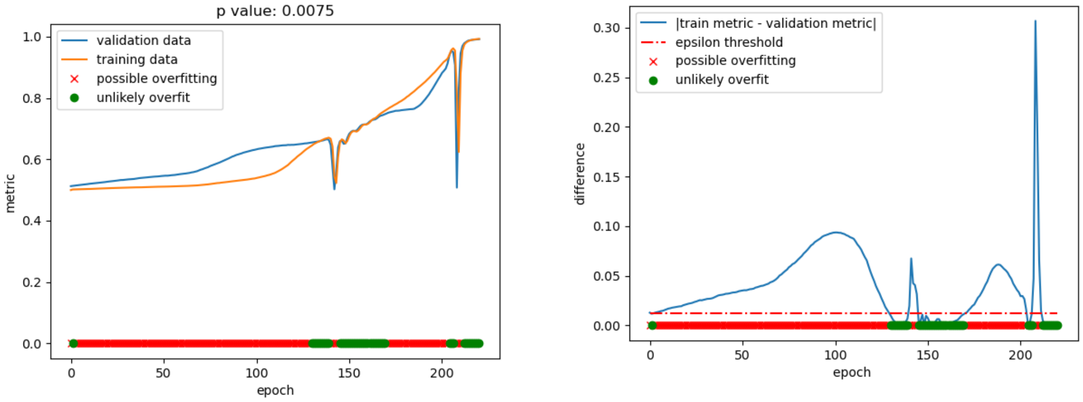

Figure 3: $k = 20,000$

4.2 Functional Map of the World

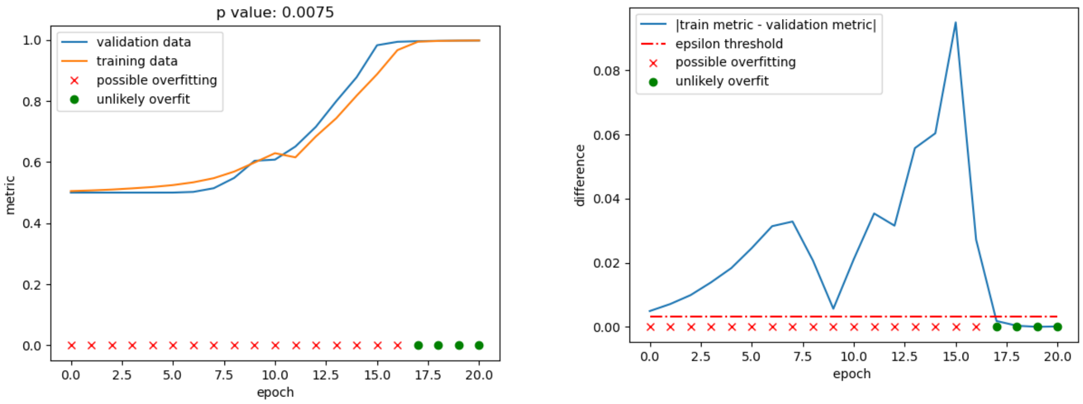

Figure 4: $k = 500,000$

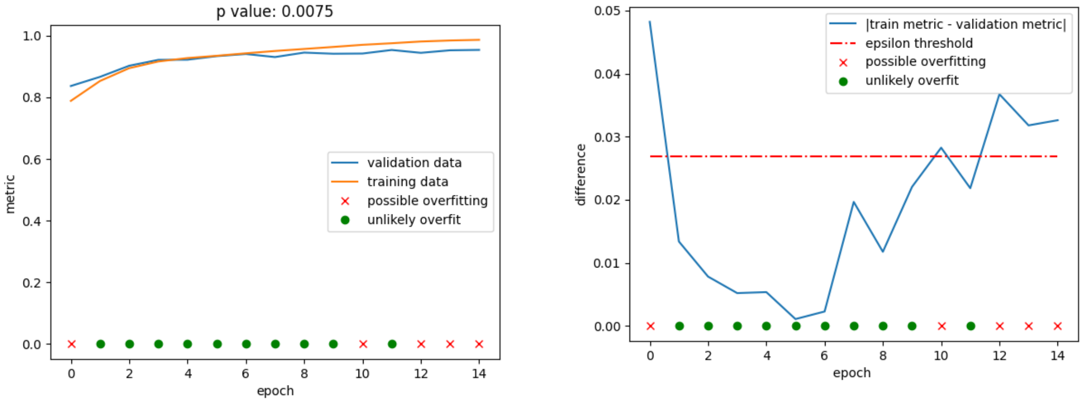

Figure 5: FMoW $k = 3850$

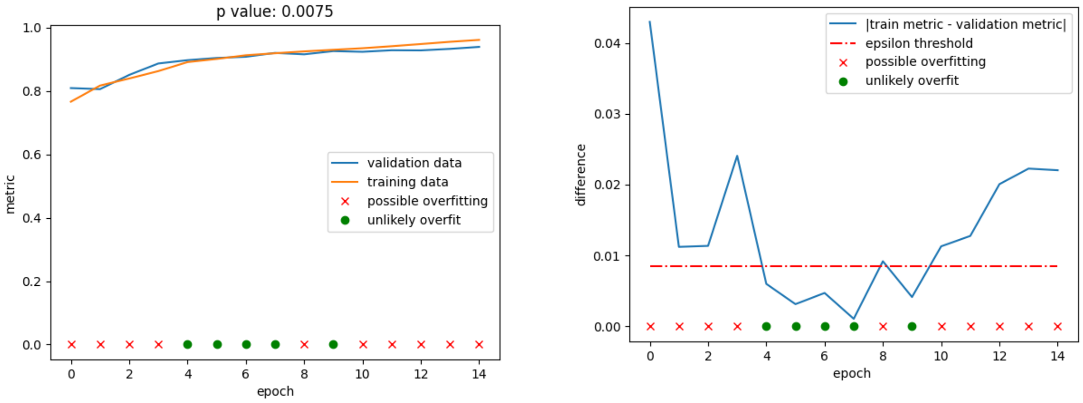

Figure 6: FMoW $k = 38,540$

A Hoeffding's Inequality for Statistical Hypothesis Testing

Hoeffding's inequality gives a probability bound for independent sample $S = (x_1, \dots, x_m) \sim_{iid} \mathbb{P}_{\mathcal{X}}$ when $\mathcal{X} = [0, 1]$, namely:

$$\mathbb{P}_{\mathcal{X}}\left(\left|\int_{\mathcal{X}} x\,d\mathbb{P}_{\mathcal{X}}(x) - \frac{1}{|S|}\sum_{x\in S}\right|\right) < 2e^{-2\varepsilon^2|S|}. \tag{9}$$

Therefore, given any two of confidence specification $\delta \in (0, 1)$, data set sized $|S| = m$, and precision bound $\varepsilon \in (0, 1)$, one may readily solve for the third.

Proof of its verity and other applications may be found in various probability texts ([2], [10], [5]).

References

[1] J. Bilmes. Underfitting and overfitting in machine learning. *UW ECE Course Notes*, 2020. 5

[2] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 02 2013. 9

[3] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. pages 6172–6180, 2018. 7

[4] B. Ghojogh and M. Crowley. The theory behind overfitting, cross validation, regularization, bagging, and boosting: Tutorial. *arXiv [stat.ML]*, 2019. 1, 5

[5] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning*. MIT Press, 2018. 3, 9

[6] Lutz Prechelt. *Early Stopping — But When?*, pages 53–67. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. 4

[7] E. Riehl. *Category Theory in Context*. Dover, 2016. 3

[8] R. Roelofs. Measuring generalization and overfitting in machine learning. 2019. 5

[9] Rebecca Roelofs, Vaishaal Shankar, Benjamin Recht, Sara Fridovich-Keil, Moritz Hardt, John Miller, and Ludwig Schmidt. A meta-analysis of overfitting in machine learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 5

[10] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning*. Cambridge University Press, 2014. 3, 9

[11] M. Valdenegro-Toro and M. Sabatelli. Machine learning students overfit to overfitting. *arXiv [cs.LG]*. 5