On the Computable Learning of Continuous Features

Nathanael Ackerman[1], Julian Asilis[1,2], Jieqi Di[2], Cameron Freer[3], and Jean-Baptiste Tristan[2]
[1]Harvard University, [2]Boston College, [3]Massachusetts Institute of Technology

In computational learning theory, the task of a learner is to predict labels from features. Such (feature, label) pairs are drawn from a universe $\mathcal{X} \times \mathcal{Y}$ according to a probability distribution $\mathcal{D}$ that is unknown to the learner. A *learner* takes a sequence of examples $S = \big((x_1, y_1), \dots, (x_n, y_n)\big)$ and outputs a *hypothesis* $h\colon \mathcal{X} \to \mathcal{Y}$. The *true error* under $\mathcal{D}$ is defined to be $L_{\mathcal{D}}(h) = \mathcal{D}\big(\{(x, y) \mid y \neq h(x)\}\big)$ and the *empirical error* is defined to be $L_S(h) = \frac{\sum |h(x_i) - y_i|}{n}$; notably, the empirical error of h can be known by a learner while the true error cannot.

Definition 1. Let $\mathbb{D}$ be a collection of distributions on $\mathcal{X} \times \mathcal{Y}$ and let $\mathcal{H}$ be a class of hypotheses. A learner A is said to *learn $\mathcal{H}$ with respect to* $\mathbb{D}$ if there exists a function $m\colon (0,1)^2 \to \mathbb{N}$ with the following property: for every $\epsilon, \delta \in (0,1)$ and every distribution $\mathcal{D} \in \mathbb{D}$, a finite $\mathcal{D}$-i.i.d. sample S with $|S| \geq m(\epsilon, \delta)$ is such that, with probability at least $(1-\delta)$ over the choice of S, the learner A outputs a hypothesis $A(S)$ with

$$L_{\mathcal{D}}(A(S)) \leq \inf_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon.$$

When such an A exists, we say that $\mathcal{H}$ is *PAC learnable with respect to* $\mathbb{D}$. In the case where $\mathbb{D}$ consists of all distributions on $\mathcal{X} \times \mathcal{Y}$, we say that A is an *agnostic PAC learner for* $\mathcal{H}$, and that $\mathcal{H}$ is *agnostically PAC learnable*. When $\mathbb{D}$ consists of those distributions $\mathcal{D}$ for which $L_{\mathcal{D}}(h) = 0$ for some $h \in \mathcal{H}$, then we say that A is a *PAC learner for $\mathcal{H}$ in the realizable case*, and that $\mathcal{H}$ is *PAC learnable in the realizable case*.

In classical PAC learning theory, the learnability of a hypothesis class is characterized by its *VC-dimension*, a measure of its complexity which considers the class of restrictions of hypotheses in $\mathcal{H}$ to a finite set U, $\mathcal{H}|_U$. In particular, the VC-dimension of $\mathcal{H}$ equals the cardinality of the largest finite set $U \subseteq \mathcal{X}$ for which $\mathcal{H}|_U = 2^U$. If no such U exists, then $\mathrm{VC}(\mathcal{H}) = \infty$.

Theorem 2 is the main classical result relating VC-dimension to PAC learning. It holds for hypothesis classes satisfying certain mild technical assumptions (see [BEHW89]), which in particular are satisfied for countable hypothesis classes, as we consider here.

Theorem 2 (Fundamental Theorem of Statistical Learning (see, e.g., [SB14, Theorem 6.7]))**.** *Let $\mathcal{H}$ be a hypothesis class of functions from a domain $\mathcal{X}$ to $\{0,1\}$. Then the following are equivalent:*

1. *$\mathcal{H}$ has finite VC-dimension.*
2. *$\mathcal{H}$ is PAC learnable in the realizable case.*
3. *$\mathcal{H}$ is agnostically PAC learnable.*
4. *Any ERM learner is a successful PAC learner for $\mathcal{H}$, over any family of measures.*

Because of this result, we will refer to a hypothesis class satisfying any of these conditions as simply *PAC learnable.*

In the basic PAC learning theory, learners are permitted to be arbitrary functions from the set $(\mathcal{X} \times \mathcal{Y})^{<\omega}$ of finite sequences to the collection of functions from $\mathcal{X}$ to $\mathcal{Y}$. (For example, one might define A to be any learner which selects a hypothesis in $\mathcal{H}$ attaining minimal empirical error, known as an *empirical risk minimization (ERM) learner.*). Under *efficient PAC learning*, however, learners are obligated to be computable and to furthermore have a running time which is polynomial in $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$.

We study the intermediate setting where learners are required to be computable but not resource-bounded. The recent paper [AAB+20] has studied this setting in the case of binary classification ($\mathcal{Y} = \{0, 1\}$) with discrete features ($\mathcal{X} \subseteq \mathbb{N}$). We generalize this setting by allowing $\mathcal{X}$ to be an arbitrary *computable Polish space*; that is, a triple $(X, d, (s_i)_{i\in\mathbb{N}})$ where (X, d) is a Polish space, $(s_i)_{i\in\mathbb{N}}$ an enumeration of a dense subset of ideal points, and d a distance function that is uniformly computable on the ideal points. A similar definition of a computable learner is developed in [CMPR21], based on slightly different collections of spaces and maps.

A key notion, defined in [AAB+20], is that of a *computably enumerable representable* (CER) hypothesis class, namely a class that admits a computable enumeration of codes for its elements.

Definition 3. A hypothesis class $\mathcal{H}$ is *computably agnostically PAC learnable* if $\mathcal{H}$ is CER and there is an agnostic PAC learner A for $\mathcal{H}$ which is computable as a function from $(\mathcal{X} \times \mathcal{Y})^{<\omega}$ to $\mathcal{H}$, considered as metric spaces. The class $\mathcal{H}$ is *computably PAC learnable in the realizable case* if some PAC learner A for $\mathcal{H}$ in the realizable case is computable on the set of finite sequences $\big((x_1, y_1), \ldots, (x_n, y_n)\big)$ for which $\big\{(x_1, y_1), \ldots, (x_n, y_n)\big\}$ is a subset of the graph of some $h \in \mathcal{H}$.

We characterize the computability of ERM learners for CER classes, providing a positive result for computable learning in the realizable case. Write $\mathbf{0}'$ for the halting set, i.e., the set of $n \in \mathbb{N}$ for which the nth Turing machine halts on empty input.

Theorem 4. *For every CER class $\mathcal{H}$ that is PAC learnable, some ERM learner of it is $\mathbf{0}'$-computable.*

Theorem 5. *There is a PAC learnable CER hypothesis class $\mathcal{H}$ such that every ERM learner for it computes $\mathbf{0}'$.*

However, if we are willing to restrict the domain on which learners must succeed, we can find an ERM that is computable on this domain.

Theorem 6. *For every CER class $\mathcal{H}$ that is PAC learnable, some ERM learner of it is computable in the realizable case, and hence $\mathcal{H}$ is computably PAC learnable in the realizable case.*

References

[AAB+20] S. Agarwal, N. Ananthakrishnan, S. Ben-David, T. Lechner, and R. Urner, *On learnability with computable learners*, Proceedings of the 31st International Conference on Algorithmic Learning Theory (ALT) (San Diego, California, USA), Proceedings of Machine Learning Research, vol. 117, 2020, pp. 48–60.

[BEHW89] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, *Learnability and the Vapnik–Chervonenkis dimension*, J. ACM **36** (1989), no. 4, 929–965.

[CMPR21] T. Crook, J. Morgan, A. Pauly, and M. Roggenbach, *A computability perspective on (verified) machine learning*, arXiv e-print 2102.06585 (2021).

[SB14] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*, Cambridge University Press, 2014.