

Learning from Satisfying Assignments Using Risk Minimization

Manjish Pal
IIT-Kharagpur
India
manjishster@gmail.com

Subham Pokhriyal
Chang Gung University
Taiwan
subham0100@gmail.com

Abstract—In this paper we consider the problem of Learning from Satisfying Assignments introduced by [1] of finding a distribution that is a close approximation to the uniform distribution over the satisfying assignments of a low complexity Boolean function f . In a later work [2] consider the same problem but with the knowledge of some continuous distribution D and the objective being to estimate D_f , which is D restricted to the satisfying assignments of an unknown Boolean function f . We consider these problems from the point of view of parameter estimation techniques in statistical machine learning and prove similar results that are based on standard optimization algorithms for Risk Minimization.

I. INTRODUCTION

Computational Learning Theory is a field of theoretical computer science that mostly deals with computational complexity theoretic aspects of machine learning with its origins from works of Valiant [5] that introduced the notion of Probably Approximately Correct (PAC) learnability. Learning of Boolean Functions has become an important part of this topic with a variety of results obtained over the past three decades or so. We consider the problem mentioned in the previous sections of estimating the uniform distribution over the satisfying assignments of an unknown Boolean function f from the point of view of standard parameter estimation techniques in Statistical Machine Learning which are highly well established but not much used in the context of Boolean function analysis. An extensive motivation and background for the problem is mentioned in [1], [2]. The problem has been considered in another set up in which we assume that D is a known continuous distribution like Gaussian, log-concave etc. and we are given i.i.d. samples from D_f , which is defined as D restricted to satisfying assignments of f where f is again an unknown Boolean function, and we intend to estimate D_f . In these works the authors prove both algorithmic and impossibility results depending on the nature of f (for eg. f being a threshold function w.r.t. a low degree polynomial). In this paper we restrict our attention to the original version of the problem and hope that our ideas would translate to other case as well.

Parameter estimation is a widely explored topic in machine learning with a variety of techniques which have been developed over several years. The main goal in this exploration is to approximate certain unknown distribution (from which the data is assumed to be coming) by a distribution that

depend on certain parameters θ ; and use certain optimization procedures to find the values of the parameters θ . In this paper we consider the same problem of learning the distribution of satisfying assignments but we use standard parameter estimation techniques for solving this problem. More formally, for an unknown distribution $P(x)$ chosen from an unknown distribution D we approximate it with $P_\theta(x)$ and compute the parameters θ by maximizing the log likelihood function. It turns out that this optimization is equivalent to minimizing the expected log loss, which in turn is Risk Minimization under the distribution D . We use these relations along with bounds that relate KL-divergence and l_1 distance to get results similar and comparable with De et al.

II. RISK MINIMIZATION

Risk Minimization [4] is a statistical machine learning framework in which we formally define the notion of error of a classifier. The risk minimization can be defined by either w.r.t the given data i.e. Empirical Risk Minimization (ERM) or it can be defined w.r.t the distribution from which the data is assumed to be coming from. More formally, ERM for a hypothesis h and a dataset S with $|S| = m$ is defined as

$$L_S(h) = |\{i \in [m] : h(x_i) \neq y_i\}|/m$$

whereas the risk w.r.t. the distribution D the risk is given as

$$L_S(f, h) = \mathbb{P}_{x \sim D}[f(x) \neq h(x)]$$

We will use these notions of risk minimization to revisit some already established relation between Maximum Likelihood Estimator (MLE) and Empirical Risk Minimization when we are dealing with a particular loss function. In general when we have the loss function $l(\mathbf{w}, x)$ as

$$l(\mathbf{w}, x) = -\log(P_{\mathbf{w}}[x])$$

where $P_{\mathbf{w}}[x]$ is the parametric estimate of the true and unknown $P[x]$ (the probability of choosing x from the unknown distribution D) and \mathbf{w} is the unknown parameter to be estimated. It is known that according to this choice of loss

function, the MLE is same as the Risk Minimization i.e.

$$\begin{aligned} E[l(\mathbf{w}, x)] &= -\sum_x P[x] \log P_{\mathbf{w}}[x] \\ &= \underbrace{\sum_x P[x] \log \frac{P[x]}{P_{\mathbf{w}}[x]}} + \sum_x P[x] \log \frac{1}{P[x]} \end{aligned}$$

We notice that the term in the above equation that is in braces is the *Kullback Leibler Divergence* of the two distributions $P[x]$ and $P_{\mathbf{w}}[x]$ which we will denote by $KL(P, P_{\mathbf{w}})$. Intuitively, minimizing the term $E[l(\mathbf{w}, x)]$ w.r.t. \mathbf{w} leads to the KL divergence being very small because the other term is independent of \mathbf{w} . There are two approaches that can be followed to minimize $E[l(\mathbf{w}, x)]$: first is to use the celebrated *Expectation Maximization Algorithm* which is not of much use for us directly because the algorithm is well known to get stuck at local optima. Instead we prefer the second approach that is based on using the Stochastic Gradient Descent algorithm to serve our purpose that is guaranteed to return a value of \mathbf{w} that is close to the global optimum. In what follows we describe the SGD algorithm and later describe how these results can be used to get bounds similar to prior works in the context of learning from the satisfying assignments of a Boolean function.

III. STOCHASTIC GRADIENT DESCENT

Over the past few years, many learning algorithms for minimizing risk functions have been proposed in the field of convex optimization. One of the important algorithms for convex optimization that is widely used in machine learning is called Stochastic Gradient Descent (SGD). In SGD we try to minimize risk function $L_D(\mathbf{w})$ in which we are not aware of the unknown distribution D and hence we can't directly compute gradient for $L_D(w)$ that is needed for standard gradient descent. The approach of SGD is to initialize the gradient in random direction and use update rule of gradient descent algorithm for optimization. A major component in the SGD algorithm is the notion of the set of *subgradients* that is denoted by $\partial l(\mathbf{w}, z)$ that is mentioned in the following vanilla version of SGD. For a detailed analysis of SGD algorithm in the context of machine learning can be obtained from [4].

Algorithm 1: SGD(X, η, T); // Minimize $L_D(\mathbf{w})$

Input: Scalar $\eta > 0$, integer $T > 0$;
Initialization: $\mathbf{w}^{(1)} = 0$;
for $t = 1$; $t \leq T$; $t = t + 1$ **do**
 sample $x \sim D$; // this step picks an element from given data
 pick $v_t \in \partial l(\mathbf{w}^t, x)$; // this step uses randomization.
 update $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta v_t$;
end
Output $\bar{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^T \mathbf{w}^{(t)}$;

Theorem 1: Consider a convex-Lipschitz-bounded learning problem with parameters ρ, B . Then, for every $\epsilon > 0$, if we run the SGD method for minimizing $L_D(\mathbf{w})$ with a number of iterations (i.e., number of examples)

$$T \geq \frac{B^2 \rho^2}{\epsilon^2}$$

and with $\eta = \sqrt{\frac{B^2 \rho^2}{\epsilon^2}}$, then the output of SGD satisfies

$$E[L_D(\bar{\mathbf{w}})] \leq \min_{\mathbf{w} \in H} L_D(\mathbf{w}) + \epsilon.$$

IV. BOOLEAN FUNCTIONS AND MAIN RESULT

In this section we describe how our results relate with the results in the context of the problem of learning from satisfying assignments of a Boolean Function. In the Boolean Function setting we are given satisfying assignments of an unknown Boolean function $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$ and our objective is to output a highly accurate estimate of $U_{f^{-1}(1)}$. In a different version of the problem one is given a continuous distribution D and i.i.d. samples of D_f where D_f is obtained by restricting D to satisfying assignments of an unknown Boolean function f . The objective again is to get an accurate estimate of D_f . In those papers the authors measure the distance of the estimated distribution and the actual distribution by *variation distance* that is also the l_1 -norm of the two probability distributions. In this paper we first estimate there distance by KL divergence and then use the distortions bounds to translate the error bounds to the l_1 -norm.

Lemma 1: If $\bar{\mathbf{w}}$ is the value returned by the algorithm then

$$KL(P, P_{\bar{\mathbf{w}}}) \leq \epsilon \text{ and } l_1(P, P_{\bar{\mathbf{w}}}) \leq \sqrt{2\epsilon}$$

a) Proof: Let $\min_{\mathbf{w} \in H} L_D(\mathbf{w}) = A(\mathbf{w}')$ and $\Delta = \sum_x P[x] \log \frac{1}{P[x]}$. Clearly $KL(P, P_{\mathbf{w}'}) = 0$. Thus $L_D(\bar{\mathbf{w}}) = KL(P, P_{\bar{\mathbf{w}}}) + \Delta$. Therefore,

$$|A(\mathbf{w}') - L_D(\bar{\mathbf{w}})| = |KL(P, P_{\mathbf{w}'}) - KL(P, P_{\bar{\mathbf{w}}})|$$

From Theorem 1 we know that $|A(\mathbf{w}') - L_D(\bar{\mathbf{w}})| \leq \epsilon$. Hence, $KL(P, P_{\bar{\mathbf{w}}}) \leq \epsilon$. From Pinsker Inequality we have $l_1^2(P, P_{\bar{\mathbf{w}}}) \leq 2KL(P, P_{\bar{\mathbf{w}}})$.

$$l_1(P, P_{\bar{\mathbf{w}}}) \leq \sqrt{2\epsilon}$$

□

Thus we have the following result, st $1 - \delta$ a probability distribution $P_{\bar{\mathbf{w}}}$ that is close to uniform distribution over $f^{-1}(1)$ by a variation distance of ϵ .

In our framework, we notice that for SGD to be executed we need to know an exact functional form for $L_D(\mathbf{w}) = E[l(\mathbf{w}, x)] = g(\langle \mathbf{w}, x \rangle)$ in order to compute $\nabla(g)$ w.r.t. \mathbf{w} . From the well known *No Free Lunch Theorem* [4] we are aware that in order to achieve learnability of any target function or distribution, we need to have some prior information about the class of hypothesis we are restricting ourselves to. In the current set up, we incorporate this information by assuming that $E[l(\mathbf{w}, x)]$ can be expressed as $g(\langle \mathbf{w}, x \rangle)$ where g is convex ρ -Lipschitz function. For eg. we can choose g to be any of sin, sigmoid, $\sqrt{x^2 + 5}$ which are 1-Lipschitz. Our assumption about $L_D(\mathbf{w})$ as $g(\langle \mathbf{w}, x \rangle)$ is similar to the assumptions made by [1], [2] in which authors put a

restrictions on f as a low degree threshold function. Although we have not been able to show a mathematical equivalence between the assumptions; both the assumptions are restricting the nature of optimization problem to certain extent.

Also w.l.o.g. we will assume that the value of B used in the SGD is 1 i.e. we are optimizing inside a unit sphere, $\|\mathbf{w}\| = 1$.

Retrieving x given $\bar{\mathbf{w}}$: Using the “reformulation” of $L_D(\mathbf{w})$ as $g(\langle \mathbf{w}, x \rangle)$, we are able retrieve $\bar{\mathbf{w}}$ and $b = L_D(\bar{\mathbf{w}})$ but our aim was to sample assignments x that are from the desired distribution. For that we simply pick an $\bar{x} \in \{-1, 1\}^n$ and compute $a = g(\langle \bar{\mathbf{w}}, \bar{x} \rangle)$ and compare with b . If $|a - b| \leq \epsilon_1$ then we declare that $\bar{x} \in f^{-1}(1)$ otherwise not.

Algorithm 2: Algorithm A

Input: $X = x_1, x_2, \dots, x_k$ be the satisfying assignment of f ;
 $\bar{w} = \text{SGD}(X, B, \rho, \epsilon)$; // This step uses g , a 1-Lipschitz function
Output: $P_{\bar{\mathbf{w}}}$; // x can be obtained from $P_{\bar{\mathbf{w}}}$

Theorem 2: The Algorithm A returns a probability distribution $P_{\bar{\mathbf{w}}}$ that is ϵ close to the unknown distribution $U_{f^{-1}(1)}$ w.r.t. variation distance with probability atleast $1 - \delta$ and runs in time in $\Theta(\frac{n}{\epsilon^2} \log(\frac{1}{\delta}))$ where n is the number of variables of the Boolean function f .

b) Proof: From Theorem 1 we know that one iteration of the SGD requires $\Theta(\frac{B^2 \rho^2}{\epsilon^2})$ and from the previous discussion

we know that $B = 1$ and we can choose an $g(\langle \mathbf{w}, x \rangle)$ such that it is ρ -lipschitz with $\rho = 1$ for example l can be chosen as sigmoid, sine etc. Using Chernoff-Hoeffding [4] bounds we can infer that by repeating the SGD algorithm independently $\log(\frac{1}{\delta})$ iterations, we can success probability to $1 - \delta$.

V. CONCLUSION

We have shown that the Empirical Risk Minimization approach can lead us to results that are comparable [2] for the problem of learning from satisfying assignments [2]. The technique of machine learning models for parameter estimation helps us to estimate the uniform distribution over $f^{-1}(1)$. Since we know that stochastic gradient descent algorithm converges in $\Theta(\frac{n}{\epsilon^2})$ iterations, We have been able to conclude that our algorithm for estimating $U_{f^{-1}(1)}$ runs in $\Theta(\frac{n}{\epsilon^2} \log(\frac{1}{\delta}))$ time and w.h.p. returns a probability distribution that has low variation distance with $U_{f^{-1}(1)}$.

REFERENCES

- [1] A. Bhattacharya, P. Kar, M. Pal, “On Low Distortion Embeddings of Statistical Distance Measures into Low Dimensional Spaces” 20th Database and Expert Systems Applications (DEXA), 164-172 (2009).
- [2] A. De, I. Diakonikolas, Rocco A. Servedio, “Learning from satisfying assignments”, 26th Symposium on Discrete Algorithms (SODA), 478-497 (2015).
- [3] L. Canonne, A. De, Rocco A. Servedio, “Learning from satisfying assignments under continuous distributions”, 31st Symposium on Discrete Algorithms (SODA), 82-101 (2020).
- [4] S. Ben-David, S. Shalev-Shwartz, “Understanding Machine Learning: From Theory to Algorithms”, Cambridge University Press, (2014).
- [5] L. Valiant, “A Theory of the Learnable”, Communications of the ACM (11), 1134-1142 (1984).