Low ℓ_1-Norm and Guarantees on Sparsifiability

Shai Shalev-Shwartz SHAI@TTI-C.ORG
Nathan Srebro NATI@UCHICAGO.EDU
Toyota Technological Institute at Chicago, 1427 East 60th Street, Chicago IL 60637, USA

Abstract

We consider the following problem: given a linear predictor $\mathbf{w}$ with low ℓ_1-norm, is it always possible to obtain a sparse predictor with similar error? It is interesting to understand this question as a further step in understanding the relationship between sparsity and the ℓ_1-norm, which is often used as a surrogate to sparsity. We show that for any $\epsilon > 0$, there exists a predictor with expected loss at most ϵ more than $\mathbf{w}$ that uses only $O\left((\|\mathbf{w}\|_1/\epsilon)^2\right)$ features. Furthermore, such a predictor can be obtained using a simple randomized procedure. We show that this bound is tight, and hence the simple randomized procedure is in a sense optimal.

1. Introduction

Although many features might be available for use in a prediction task, it is often beneficial to use only a small subset of the available features, even at the cost of a small degradation in performance relative to a predictor that uses more features. Focusing on linear prediction, it is generally difficult to find the best predictor subject to a constraint on the number of features used (the *sparsity* of the predictor). A common alternative is to seek a good predictor with small ℓ_1-norm, using this measure as a surrogate for sparsity. However, the resulting predictor need not necessarily be sparse. A common approach is to somehow obtain a sparse predictor from the learned low-ℓ_1-norm predictor. But can this always be done without significantly sacrificing performance?

We study the question of "sparsification" of linear predictors. Can a low-ℓ_1-norm predictor always be sparsified? I.e., does the existence of a good linear predictor with low ℓ_1-norm guarantee the existence of a good linear predictor that uses only a small number of features? If so, what is the relationship between the ℓ_1-norm and the number of features necessary to achieve similar performance? And is there a simple procedure for "sparsifying" a predictor, i.e. obtaining a good sparse predictor from a good predictor with low ℓ_1-norm?

We provide a simple randomized procedure for obtaining a sparse predictor $\tilde{\mathbf{w}}$ from a low-ℓ_1-predictor $\mathbf{w}$. We show that for any allowed degradation $\epsilon > 0$, the sparsification procedure can produce a linear predictor $\tilde{\mathbf{w}}$ that uses only $O\left((\|\mathbf{w}\|_1/\epsilon)^2\right)$ features (independent of the overall number of features used by $\mathbf{w}$), and has expected loss at most ϵ worse than $\mathbf{w}$. Furthermore, we show that this relationship is tight (in the worst case): as many as $\Omega\left((\|\mathbf{w}\|_1/\epsilon)^2\right)$ features might be required in order to get within ϵ of the expected loss of a linear predictor $\mathbf{w}$. We also show that the existence of a predictor with low ℓ_2-norm is *not* enough to guarantee the existence of a sparse predictor. This is perhaps not surprising, and provides further insight as to why ℓ_1-regularization is preferable to ℓ_2-regularization when sparsity is the true objective. Finally, we show that the common sparsification heuristic, in which the smallest elements of $\mathbf{w}$ are zeroed, might produce poor sparse predictors. For constructing our tightness results we derive a generalization of Khintchine inequality that holds for biased random variables. We believe that this inequality can be useful for deriving additional lower bounds in machine learning, involving linear loss functions.

Related work Much work on compressed sensing focuses on conditions, both on the labels and on the training examples (i.e. the design matrix), under which the optimal ℓ_1-norm predictor will be sparse. But these conditions don't generally hold in machine learning applications (e.g. when many features are redundant), while we might still hope to be able to use ℓ_1-norm regularization in order to get a sparse predictor.

Ng (Ng, 2004) considers PAC learning of a sparse predictor, and shows that ℓ_1-norm regularization is competitive with the best sparse predictor, while ℓ_2-regularization does not appear to be. In such a scenario we are not interested in the resulting predictor being sparse (it won't necessarily be sparse), but only in its generalization performance. In contrast, in this paper we *are* interested in the resulting predictor being sparse, but do not study ℓ_1-regularized learning. Rather, we assume we already have a good low-ℓ_1-norm predictor, and ask whether we can obtain from it a good predictor that is sparse.

The converse of our question, focusing on linear classification, was recently resolved by Servedio (Servedio, 2006): given a sparse linear separator, can it always be represented using small weights?

The randomized sparsification procedure we suggest was previously proposed by Schapire et al (Schapire et al., 1997), as a tool for obtaining generalization bounds for boosting. However, Schapire et al's bound depends on $\log(m)$, where m is the number of examples in the input distribution, and is therefore only valid for guaranteeing performance over a finite sample. Our bound does not depend on m and is adequate for guaranteeing performance over an arbitrary source distribution.

Studying neural networks with bounded fan-in, Lee et al (Lee et al., 1996) addressed an equivalent formulation of this question, providing an upper bound similar to ours, for the special case of the squared-error loss. Here we obtain a more general result, that holds for any (Lipschitz-continuous) loss function. Furthermore, we present matching upper and lower bounds, which together tightly characterize the possible sparseness guaranteed by low ℓ_1-norm.

2. Guaranteed Sparsification Procedure

Let $\mathbf{w} \in \mathbb{R}^n$ be an arbitrary (possibly dense) predictor. Without loss of generality, we assume that $w_j \geq 0$ for all j (since otherwise, if $w_j < 0$, we can flip the sign of the j'th feature). Thus, the predictor $\mathbf{w}/\|\mathbf{w}\|_1$ defines a probability measure over the set $[n]$. To motivate our construction we would like to note that the prediction $\langle \mathbf{w}, \mathbf{x} \rangle$ can be viewed as the expected value of the elements in $\mathbf{x}$ according to the distribution $\mathbf{w}/\|\mathbf{w}\|_1$ (scaled by $\|\mathbf{w}\|_1$). We can approximate this expected value by an empirical average of randomly selected elements of $\mathbf{x}$. Since our goal is to find a sparse predictor whose predictions are similar to those of $\mathbf{w}$, we construct the sparse predictor by randomly selecting S elements from $[n]$ based on the probability measure $\mathbf{w}/\|\mathbf{w}\|_1$.

Formally, let r be a sequence of i.i.d. random variables over $[n]$ with $\mathbb{P}(\mathsf{r}_i = j) = \frac{w_j}{\|\mathbf{w}\|_1}$. We set our sparse predictor to be $\tilde{\mathbf{w}} = \frac{\|\mathbf{w}\|_1}{S} \sum_{i=1}^{S} \mathbf{e}^{\mathsf{r}_i}$, where $\mathbf{e}^i$ is the ith standard basis vector.

Theorem 1 *Let $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_\infty \leq 1\}$ be an instance space, $\mathcal{Y}$ be a target space, $\mathcal{D}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$ and $L : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ be a loss function which is λ-Lipschitz with respect to its first argument. For any $\mathbf{w} \in \mathbb{R}^n_+$, and any $\delta > 0$, with probability at least $1 - \delta$ over the choice of r, we have:*

$$\mathbb{E}_{(\mathsf{x},\mathsf{y})\sim\mathcal{D}}[L(\langle \tilde{\mathbf{w}}, \mathsf{x} \rangle, \mathsf{y})] \leq \mathbb{E}_{(\mathsf{x},\mathsf{y})\sim\mathcal{D}}[L(\langle \mathbf{w}, \mathsf{x} \rangle, \mathsf{y})] + \sqrt{2}\frac{\lambda \|\mathbf{w}\|_1}{\sqrt{S}} \left(\sqrt{\log(1/\delta)} + 5 \right)$$

Taking δ close to one, we can conclude that the existence of a predictor $\mathbf{w}$ with expected loss l, guarantees the existence of a sparse predictor $\tilde{\mathbf{w}}$, with $\|\tilde{\mathbf{w}}\|_0 \leq (7.1\,\lambda\,\|\mathbf{w}\|_1/\epsilon)^2$ and expected loss at most $l + \epsilon$. Furthermore, after learning $\mathbf{w}$, we can efficiently construct a sparse predictor $\tilde{\mathbf{w}}$, with sparsity almost as above. Note that to perform the sparsification we do not need access to the source distribution $\mathcal{D}$ nor to any samples—the sparse predictor $\tilde{\mathbf{w}}$ is a (random) function of only the (dense) predictor $\mathbf{w}$. If we do have access to samples, and a low ℓ_1-norm predictor with low training error, we can repeatdly (randomly) construct a sparse predictor until we verify that the sparse predictor indeed has low training error. Theorem 1, with $\mathcal{D}$ set to the empirical distribution, bounds the expected runtime of this procedure.

3. Tightness and Extreme Examples

We now argue that the procedure of the previous section is optimal in the sense that no other procedure can yield a better sparsity guarantee (better by more than a constant factor) in terms of the ℓ_1-norm of the input predictor w.

We will use the following lemma, which generalizes the Khintchine inequality also to biased random variables. We use the lemma in order to obtain lower bounds on the mean-absolute error in terms of the bias and variance of the prediction:

Lemma 1 *Let $\mathsf{x} = (\mathsf{x}_1, \ldots, \mathsf{x}_n)$ be a sequence of independent Bernoulli random variables with $0.05 \leq \mathbb{P}[\mathsf{x}_k = 1] \leq 0.95$. Let Q be an arbitrary polynomial over n variables of degree d. Then,*

$$\mathbb{E}[\,|Q(\mathsf{x})|\,] \;\geq\; (0.2)^d\, \mathbb{E}[\,|Q(\mathsf{x})|^2\,]^{\frac{1}{2}} \quad .$$

Theorem 2 *For any $B > 2$ and $l > 0$, there exists a data distribution, such that a (dense) predictor $\mathbf{w}$ with $\|\mathbf{w}\|_1 = B$ can achieve mean absolute-error ($L(a,b) = |a - b|$) less than l, but for any $\epsilon \leq 0.1$,* *at least $B^2/(45\,\epsilon^2)$ features must be used for achieving mean absolute-error less than ϵ.*

To prove the theorem, we present an input distribution $\mathcal{D}$ for which we have a specific (dense) predictor with $\|\mathbf{w}\|_1 = B$ and mean absolute-error l, and also a lower bound on mean absolute-error of any sparse predictor.

Consider an instance space $\mathcal{X} = \{+1,-1\}^n$, where $n \geq 1/(la)^2$, and a target space $\mathcal{Y} = \{+1,-1\}$. The distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ is as follows. First, the label y is uniformly distributed with $\mathbb{P}(\mathsf{y} = 1) = \frac{1}{2}$. Next, the features $\mathsf{x}_1, \ldots, \mathsf{x}_n$ are identically distributed and are independent conditioned on y, with $\mathbb{P}(\mathsf{x}_i = \mathsf{y} \mid \mathsf{y}) = \frac{1+a}{2}$, where $a = 1/B$. Thus, the correlation between each feature and the label is $1/B$. In such an example, the "information" about the label is spread among all features, and in order to obtain a good predictor, this distributed information needs to be pulled together, e.g. using a dense linear predictor.

Without using Lemma 1 we can obtain a lower bound of $\|\mathbf{u}\|_0 = \Omega(B^2/\epsilon)$ on the sparsity of a predictor achieving squared-error at most ϵ. This lower bound for the special case of the squared error is tight and matches the upper-bound analysis of Lee et al (Lee et al., 1996). Using Lemma 1 allows us to demonstrate a squared dependence on ϵ might be necessary.

3.1. Low ℓ_2-norm does not guarantee sparsifiability

One might ask if the existence of a predictor with low ℓ_2-norm can also guarantee the existence of a sparse predictor. Perhaps even if our proposed sparsification procedure does not work well on predictors with low ℓ_2-norm, a different procedure might be used to sparsify such predictors. We show that this is not the case, by presenting examples where good predictions can be obtained by predictors with arbitrarily low ℓ_2-norm, but for which an arbitrarily high number of features is required in order to achieve a fixed performance.

To do so, we use the same type of data distribution and dense predictor as in the previous section. Setting $w_i = 1/(na)$ yields $\|\mathbf{w}\|_2 = 1/(a\sqrt{n})$. Therefore, we can decrease the correlation a as we increase the dimension n, keeping the ℓ_2-norm of the dense predictor fixed, but requiring an increasing number of features in order to obtain good performance.

3.2. Sparsifying by considering only large weights

The procedure described in Section 2 involves random sampling of the features. An alternative deterministic procedure, commonly used in practice, is to choose only the features with the largest weights, or in other words, to zero small weights of the predictor (and perhaps readjust the remaining weights). We consider applying this deterministic procedure to a low ℓ_1-norm predictor $\mathbf{w}^*$ learned by minimizing the expected loss subject to ℓ_1-norm regularization. Even on such an "optimal" predictor $\mathbf{w}^*$, using only the features with largest coefficients can yield a large degradation in performance. This can happen when many features are highly correlated.

Specifically, for any arbitrarily large S and arbitrarily small l, we show an example in which the optimal predictor $\mathbf{w}^*$ with ℓ_1-norm at most 3 achieves mean absolute-error at most l, but using any re-weighting of the S features with the largest coefficients yields mean absolute-error of at least 0.02. Note that if $S \geq (25/\epsilon)^2$, our randomized procedure would yield a S-sparse predictor with error at most $l + \epsilon$, which we could set arbitrarily close to zero.

We again define a joint distribution over binary targets $\mathsf{y} \in \{+1,-1\}$ and binary feature vectors $\mathbf{\mathsf{x}} \in \{+1,-1\}^{Sn}$, with $n = 7/l^2$ (i.e. the overall dimensionality is $7S/l^2$). For convenience we will label the features with two indices: $\mathsf{x}_{1,1}, \ldots, \mathsf{x}_{S,n}$. To describe the data distribution we use another set of n (latent) binary random variables $\mathsf{z}_1, \ldots, \mathsf{z}_n \in \{+1,-1\}$, i.i.d. given y, with $\mathbb{P}[\mathsf{z}_i = \mathsf{y} \,|\, \mathsf{y}] = (1 + \frac{1}{3} + \frac{i-1}{3(n-1)})/2$ (i.e. the correlation between these variables and the labels are between 1/3 and 2/3). The features $\mathsf{x}_{i,j}$ are independent given $\mathbf{\mathsf{z}}, \mathsf{y}$, and are specified by $\mathbb{P}[\mathsf{x}_{i,j} = \mathsf{z}_i \,|\, \mathsf{z}_i] = 7/8 = (1 + 0.75)/2$. The features are thus grouped into n groups of S highly correlated features, where the correlation between each feature and the label varies between 1/4 and 1/2.

The minimum-mean-absolute-error predictor among those with ℓ_1-norm bounded by three achieves mean absolute-error less than l. However, in this optimal predictor, the weights $w_{n,i}$ corresponding to features in the last group will be larger than any other weights, and so these features will be selected as the maximal weight features. But since these features are all highly correlated, using any combination of them will not yield mean absolute-error better than 0.02.

We also note that the features in the last group would also be the first S features selected by following the ℓ_1-norm regularization path or by related methods such as LARS (Efron et al., 2004).

References

B. Efron, T. Hastie, I. Johnstone, R. Tibshirani. Least Angle Regression. *Annals of Statistics*, 32(2), 2004.

Lee, W. S., Bartlett, P. L., & Williamson, R. C. (1996). Efficient agnostic learning of neural networks with bounded fan-in. *IEEE Transactions on Information Theory*, *42*, 2118–2132.

Ng, A. (2004). Feature selection, l_1 vs. l_2 regularization, and rotational invariance. *Proceedings of the Twenty-First International Conference on Machine Learning.*

Schapire, R., Freund, Y., Bartlett, P., & Lee, W. (1997). Boosting the margin: A new explanation for the effectiveness of voting methods. *Machine Learning: Proceedings of the Fourteenth International Conference* (pp. 322–330). To appear, *The Annals of Statistics.*

Servedio, R. (2006). Every linear threshold function has a low-weight approximator. *Eighteenth Annual Conference on Computational Complexity (CCC).*