The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Conditional Linear Regression

Diego Calderon
University of Arkansas
dacalder@uark.edu

Brendan Juba, Zongyi Li
Washington University in St. Louis
{bjuba, zli}@wustl.edu

Lisa Ruan
M.I.T.
llruan@mit.edu

Abstract

Previous work in machine learning and statistics commonly focuses on building models that capture the vast majority of data, possibly ignoring a segment of the population as outliers. By contrast, we may be interested in finding a segment of the population for which we can find a linear rule capable of achieving more accurate predictions. We give an efficient algorithm for the conditional linear regression task, which is the joint task of identifying a significant segment of the population, described by a k-DNF, along with its linear regression fit.

Introduction

Linear regression, a model for understanding the relationship among variables in a data set, is a standard tool widely used in biological and social sciences to predict events and to describe possible relationships between variables. When addressing the task of prediction, machine learning and statistics commonly focus on capturing the vast majority of data, occasionally ignoring a segment of the population as ”outliers” or ”noise,” which could be helpful to better understand the data. Previous work by Juba (2016) gave an algorithm to identify a significant segment of the population for which there exists a highly sparse linear fit, along with a simple rule that describes the subset. Even though sparsity is a desirable feature to have in a linear regression, we might encounter cases with solutions that are not sparse. In these cases, the previous state of the art suffers a running time blow-up that depends on the number of factors considered for the linear prediction rule. To address this problem, Juba (2016) also introduces an algorithm for the dense linear rule case, which chooses the single best term, thus picking only a fraction of the condition. We give an algorithm for conditional linear regression that does not require constant sparsity and recovers a condition of nearly optimal probability. Our algorithm extends an approach introduced by Charikar et al. (2017), which obtains a list of candidate parameter vectors that is guaranteed to have a good set of parameters for any small subset of the data.

Problem Definition

Our input is defined in terms of accessing examples from a joint distribution D over $\{0,1\}^n \times \mathbb{R}^d \times \mathbb{R}$, where a single example is denoted as $(x^{(i)}, y^{(i)}, z^{(i)}) = (x_1, \ldots, x_n, y_1, \ldots, y_d, z)_i$. In this notation, $x^{(i)}$ represents the Boolean attributes (which are described by a binary string of length n whose j-th bit is the value of $x_j^{(i)}$), $y^{(i)}$ represents the input vectors (attributes), and $z^{(i)}$ is the variable we are interested in (label). For example, imagine we want to estimate the price of a car. The label $z \in \mathbb{R}$ is the price, the attributes $y \in \mathbb{R}^d$ are the Mileage, Year, and Number of accidents. The Boolean attributes $x \in \{0,1\}^3$ can be “is American-made,” “is 4x4,” and “is Electric.” In this case, the example (1,0,1,56000,2015,0) describes an American-made, 4x2 electric car, with 56000 miles, from 2015, and with no recorded accidents. In this case, we would be interested in identifying a segment of the population for which a linear rule is highly predictive of the price of certain cars, whereas this linear rule may not provide a good prediction overall in the larger population. Let us imagine that for this data set, and for a target fraction of the population, we found a simple rule that describes the sub-population, along with its linear fit. Recall that our Boolean attributes have the form $x \in \{0,1\}^3$ and suppose that the example must satisfy the condition “is American-made,” and “is Electric.” Now, if we look at an example from the data set, we can verify if it satisfies the condition, and if it does, we have an improvement over a general prediction rule. To be specific, if we look at an arbitrary example, say, (1,0,1,45568, 2016,2), we can make a more approximate prediction of the price of the car. Formally, Juba proposed the following task:

Definition 1 (Conditional l_2-linear regression) *Suppose that D is a joint distribution over $x \in \{0,1\}^n, y \in H \subset \mathbb{R}^d$ and $z \in \mathbb{R}$, where H has l_2 radius r. If there exists an optimal k-DNF condition c^*, and $a^* \in H \subset \mathbb{R}^d$:*

$$\mathbb{E}_D[(\langle a^*, y\rangle - z)^2 | c^*(x) = 1] \le \epsilon$$

$$Pr[c^*(x) = 1] \ge \mu$$

And the error $(\langle a^*, y\rangle - z)$ follows σ-sub gaussian distribution,
Then for $\delta, \gamma \in (0,1)$, we want to find a k-DNF $\hat{c}$, and

$\hat{a} \in \mathbb{R}^d$ *in polynoimal time, such that with probability* $(1-\delta)$*:*

$$\mathbb{E}_D[(\langle \hat{a}, y\rangle - z)^2 | \hat{c}(x) = 1] \leq poly(r, d, n^k)\epsilon$$

$$Pr[\hat{c}(x) = 1] \geq \frac{\mu}{poly(r, d, n^k)}$$

Approach and Preliminary Results

Previous work by Charikar et al. gave an algorithm that outputs a list of parameter vectors by finding the best parameters for each individual example, and clustering the data accordingly. They guarantee that for any subset of the data, one of the candidate outputs will fit the subset well. Specifically, when we set the parameter to be the fit of linear regression problem, given any subset, ideally it should be able to find a regression fit with low regression loss on that subset, compared to the optimal. However, since individual points provide trivial estimates of the regression parameters, this algorithm is not appropriate for linear regression. Our contribution is to modify the algorithm to consider the collection of points that satisfy a term instead of individual points, obtaining an algorithm that finds a list of candidate regression fits. From this list of candidate regression fits, we can a find a linear fit that is near the best possible, and which obtains, at most, polynomially larger loss.

Theorem 2 *By applying the modified algorithm of Charikar et al. on the conditional* l_2*-linear regression problem, we can find a solution* $(\hat{c}, \hat{a})$ *that covers almost as much as the optimal* c^**, with* $\tilde{O}(n^{\frac{3}{2}k})$ *much more expected loss than the error of the optimal.*

The main difficulty with the approach of Charikar et al. is that they are iteratively improving their estimates by reclustering their data using their current estimates of the parameters; they then can obtain a better estimate of the parameters using such an improved clustering. The process cannot continue as long as they wish, because there is no guarantee on an iteration that all of the points that should be included in an ideal clustering will get good estimates of their parameters on every iteration. They can only guarantee that most of the points receive good estimates, and thus stay together during reclustering. Thus, on each iteration, some points from the ideal clustering are lost due to inaccurate estimates of the parameters at those points, and this rate of loss bounds the number of iterations they can safely execute. Finally, the basic per-point estimates of the regression parameters are not adequately tight to obtain a nontrivial (informative) estimate of the regression parameters at the end of this iterative process.

By contrast, we compute estimates of the regression parameters per term rather than for individual points. Since, without much loss of the overall size of the segment, we can assume that each term picks up a significant number of points. We observe that we can always get a sufficiently good estimate of the regression parameters for every term on every iteration. Therefore, we can iteratively re-cluster and re-estimate the parameters as many times as we wish, to obtain an arbitrarily precise estimate of the regression parameters.

Finally, as we mentioned earlier, the technique of Charikar et al. only produces a list of candidate regression parameters: each cluster of the data receives a different set of parameters. But, given such a polynomially large list of parameters, we can use the previous algorithm of Zhang et al.(2017) to recover a condition describing a segment of the population in which the regression parameters give a pretty good fit. We simply modify their algorithm to add weights, to solve a weighted version of their task—an analogous modification to obtain a weighted red-blue set cover algorithm was given by Peleg(2007). We can use the error incurred by a linear rule over a term as a weight for that term, so that the algorithm indeed returns a k-DNF that approximately minimizes the error, as needed.

Future Work

When we apply Charikar et al by considering each term as a point in their setting, we face a double counting problem that each point is contained in multiple terms. Currently we just treat a point as different points in different terms. The duplication results in an $\tilde{O}(n^k)$ blow-up of the error. We believe we can address the double-counting problem inside the algorithm rather than using simple duplication, which will achieve an error of $\tilde{O}(n^{\frac{k}{2}})$ instead of $\tilde{O}(n^{\frac{3}{2}k})$. Any further improvement will entail a better guarantee for the abduction problem for Zhang et al.

Acknowledgements

We thank Jacob Steinhardt for helpful discussions.

References

Charikar, M.; Steinhardt, J.; and Valiant, G. 2017. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, 47–60. New York, NY, USA: ACM.

Juba, B. 2016. Conditional sparse linear regression. *CoRR* abs/1608.05152. Presented in ITCS2017.

Peleg, D. 2007. Approximation algorithms for the label-cover max and red-blue set cover problems. *Journal of Discrete Algorithms* 5(1):55–64.

Zhang, M.; Mathew, T.; and Juba, B. 2017. An improved algorithm for learning to perform abduction. In *Proc. 31st AAAI*, 1257–1265.