OPT2023: 15th Annual Workshop on Optimization for Machine Learning

(Un)certainty selection methods for Active Learning on Label Distributions

James Spann JSPANN2@CS.ROCHESTER.EDU
University of Rochester

Pratik Bongale PSB4346@G.RIT.EDU
Rochester Institute of Technology

Christopher Homan CMH@CS.RIT.EDU
Rochester Institute of Technology

Abstract

Some supervised learning problems can require predicting a probability distribution over more than one possible (set of) answer(s). In such cases, a major scaling issue is the amount of labels needed since, compared to their single- or multi-label counterparts, distributional labels are typically (1) harder to learn and (2) more expensive to obtain for training and testing. In this paper, we explore the use of active learning to alleviate this bottleneck. We progressively train a label distribution learning model by selectively labeling data and, achieving the minimum error rate with fifty percent fewer data items than non-active learning strategies. Our experiments show that certainty-based query strategies outperform uncertainty-based ones on the label distribution learning problems we study.

1. Introduction

Label Distribution learning (LDL) [5] is a process that seeks to train a model to learn a series of probability distributions. For integers k, n, and c and set of data items $X = \{x_1, ..., x_n\} \in \mathbb{R}^{n \times k}$, the goal of LDL is to solve

$$\arg\min_{\theta}\{\mathcal{L}(y_1, \dots, y_n, p(\cdot|x_1;\theta), \dots, p(\cdot|x_n;\theta))\}, \tag{1}$$

where $\mathcal{L}$ is a loss function and, for each $i \in \{1, \dots, n\}$, $p(\cdot|x_i;\theta)$ is a conditional probability function from a family parameterized by $\theta \in \Theta$ over a set of *labels* $\{1, \dots, c\}$, and $y_i = p(\cdot|x_i)$ is the *ground truth label distribution* over $\{1, \dots, c\}$.

Unlike traditional machine learning problems, which use probability distributions to model labels, these distributions typically represent uncertainty about what the true label(s) may be. In LDL, the distributions themselves *are* the ground truth (about which the model may be uncertain). However, obtaining enough labeled data can be expensive, even when conventional (i.e., nondistributional) label prediction is the goal. And compared to their single- or multi-label counterparts, distributional labels are typically harder to learn and thus require even more labels.

In instances where data is abundant but labels are scarce or expensive to obtain, the semi-supervised optimization process of *Active Learning* has been applied to sample and label training examples as the model trains for image data[6–8], natural language problems [10, 11], and as an optimization technique for LDL [2, 3]. It is based on the idea that data items have different training utilities at different points in the learning process. For example, early in the training cycle some items may be more informative to the learning algorithm than others, resulting in faster convergence if these items are in the training set. Active learning posits that these more informative items can be discovered via a *query strategy*, which works in conjunction with a learning models, or *kernels*.

This paper addresses the following research questions:

RQ1. Which active learning query strategies are best suited for label distribution learning?

RQ2. How does the use of different active learning kernels affect the learning strategies?

To answer these questions, we consider 6 different (un)certainty-based query strategies, compare and contrast the performance of each in an active learning loop that uses one of two algorithms as a learning kernel: one previously designed Geng [5] for LDL for non-active learning and the other a purpose-built multi-layer perceptron. We test each query strategy/learning kernel on 9 label distribution benchmark data sets, and 2 distribution based datasets.

2. Methods

2.1. Data

We obtained from Geng's website[1] datasets with probability distributions as labels. Each of the nine datasets represents a particular experiment at with the yeast *Saccharomyces cerevisiae*. Each set contains a total of 2465 items $x_1, \ldots, x_n$, where each item is a single gene, represented as a 24-dimension feature vector, and a corresponding collection of ground truth label distributions $y_1 \ldots, y_n$, where varies by experiment. Each label dimension represents a point in time and the label distribution represents the relative degree to which a gene is expressed in each time period [4]. Thus, for this datasets, the interpretation of the label distributions is strictly frequentist, as the distributions in no way represent belief. We also evaluate our data against a dataset of from natural scenes [13]. Each sample is a histogram of features with 294 features to predict a distribution of 9 labels. We use 1250 items for the pool of unlabeled data $\mathcal{P}_1$ (of course, we do have their ground truth labels, but in order to simulate active learning we pretend that they are known only to the labeling oracle $\mathcal{O}$), 138 items for the seed set $\mathcal{T}_1$, 345 items for testing, and 740 for development.

2.2. Evaluation strategies

To address RQ1, we consider a variety of query strategies $\mathbf{q}$. As our baseline, we consider a ***random*** strategy, which simply samples p random data items from $\mathcal{P}_t$.

The simplest variant of uncertainty sampling is ***least-confidence sampling***:

$$x = \underset{x}{argmax}\ 1 \ - \ P_{\theta_t}(\hat{y}|x) \tag{2}$$

where $\hat{y} = p(\cdot|x, \theta_t)$ is the probability distribution predicted by model $\mathcal{M}_{\theta_t}$, and $P_{\theta_t}(\hat{y}|x)$ is a measure of the confidence in this choice. This strategy considers only the most probable label from every predicted distribution $\hat{y}_i$ and queries the data item with least value for its most probable label.

[1] http://cse.seu.edu.cn/PersonalPage/xgeng/LDL/index.htm

Min-margin sampling considers the first and second most probable labels for each data item, computes the margin (difference) between them, and picks the item with the smallest margin:

$$x = \underset{x}{argmin}\, P_{\theta_t}(\hat{y_1}|x) \;-\; P_{\theta_t}(\hat{y_2}|x), \tag{3}$$

where $\hat{y_1}$ and $\hat{y_2}$ are the first and second most probable labels, respectively.

A more general uncertainty sampling strategy which considers all class labels in the output probability distribution is *entropy sampling*, more specifically ***maximum entropy*** sampling:

$$x = \underset{x}{argmax} \; -\sum_{j=1}^{c} P_{\theta_t}(y_j|x) \log P_{\theta_t}(y_j|x). \tag{4}$$

In contrast to uncertainty sampling, ***certainty sampling*** selects the data items about which the current trained model $\mathcal{M}_{\theta_i}$ is most certain. Strategies ***most-confidence***, ***max-margin***, and ***min-entropy*** are derived from equations 2–4, respectively, by swapping $argmax$ and $argmin$.

2.3. Learning kernels

We consider two learning kernels. The first (BP) is the multilayer perceptron with backpropagation described in Section 3. The second (BFGS) is based on Geng [5]. It uses a maximum entropy model and Broyden–Fletcher–Goldfarb–Shanno Wright and Nocedal [12] optimization to solve the following likelihood problem.

$$\mathcal{L}(X, \theta) = \sum_i \frac{exp(\sum_k \theta_{y,k}\; g_k(x_i))\; g_k(x_i)}{\sum_j exp(\sum_k \theta_{y,k}\; g_k(x_i))} - \sum_i y_{ij}\; g_k(x_i) \tag{5}$$

Here, $\theta_{j,k}$ is an element in θ, which is a weight matrix of dimensions $c \times k$ and $g_k(x_i)$ returns the k^{th} feature of x_i. The trained model is thus

$$p(y|x;\theta) = \frac{1}{Z} exp(\sum_k \theta_{y,k}\; g_k(x)), \tag{6}$$

where Z is the normalization term $\sum_y exp(\sum_k \theta_{y,k}\; g_k(x))$.

2.4. Procedure

In separate trials, we perform active learning using each of the seven query strategies $\mathbf{q}$ described in Section 2.2, i.e, random selection, plus the certainty and uncertainty measures, to each of these datasets. For each round t, we select $p = 3$ items $x_{t_1}, x_{t_2}, x_{t_3}$ from $\mathcal{P}_t$ according to the predictive model $\mathcal{M}_{\theta_t}$ trained on the current set of labeled data $\mathcal{T}_t$ and $\mathbf{q}$ (and ignoring the ground truth labels y_i that came with the dataset). We simulate oracle queries by replacing the label estimates from $\mathcal{M}_{\theta_t}$ with the ground truth label distributions $y_{t_1}, y_{t_2}, y_{t_3}$.

To address RQ1, we compare the performances of each of the query strategies by comparing the ground truth label distributions with those predicted by each BFGS model $\mathcal{M}(\theta_t)$. Following Geng [5] and Cha [1] we used Kullback-Leibler (KL) divergence [9], and Chebyshev distance.

KL Divergence	cdc	cold	diau	dtt	elu	heat	spo	spo5	spoem
Random	7.74 ± 0.4153	16.9651 ± 1.2944	19.2977 ± 1.5442	7.416 ± 0.5843	6.7995 ± 0.3458	15.9199 ± 1.005	34.3738 ± 2.7863	50.4735 ± 5.6015	39.1161 ± 4.2505
Max Entropy	8.5894 ± 0.4661	18.752 ± 1.5519	23.9989 ± 2.2937	9.7604 ± 0.9095	7.5148 ± 0.3801	20.174 ± 1.3142	54.5264 ± 7.4844	68.6491 ± 8.0387	58.0959 ± 8.0252
Min Margin	8.3429 ± 0.4168	16.6227 ± 1.3898	18.9023 ± 1.4547	8.8441 ± 0.7318	6.8131 ± 0.3476	17.7868 ± 1.1109	46.6425 ± 4.045	59.3369 ± 6.781	75.527 ± 12.0989
Least Confident	8.5396 ± 0.4368	21.63 ± 1.8809	25.9292 ± 2.314	8.3941 ± 0.6139	7.8678 ± 0.3861	21.6111 ± 1.7544	60.3035 ± 6.6798	78.0245 ± 13.0412	63.3094 ± 10.9939
Min Entropy	7.4194 ± 0.4045	17.6975 ± 1.5788	**16.7083 ± 1.1846**	7.2283 ± 0.5996	**6.6228 ± 0.3394**	14.7676 ± 0.8449	**32.8759 ± 2.2884**	44.6456 ± 4.9838	38.6297 ± 4.5318
Max Margin	7.9085 ± 0.4072	15.4389 ± 1.2317	17.3011 ± 1.2825	7.0337 ± 0.5405	6.6423 ± 0.3374	15.7999 ± 0.9404	36.457 ± 4.7362	38.4463 ± 3.2941	**32.8868 ± 3.9536**
Most Confident	**7.3523 ± 0.4216**	**15.3118 ± 1.1745**	17.0514 ± 1.14	**6.7255 ± 0.5315**	6.7289 ± 0.3322	**14.2484 ± 0.8064**	35.7975 ± 2.6431	**37.1758 ± 3.2184**	34.8506 ± 4.4048

Table 1: KL Divergence ($X \cdot 10^3$) for each query strategy with Multilayer Perceptron as the learning kernel on different datasets after an additional 625 data items have been labeled (52 training iterations with 24 items learned per iteration).

KL Divergence	cdc	cold	diau	dtt	elu	heat	spo	spo5	spoem
Random	7.1845 ± 0.4069	13.7043 ± 1.1953	14.464 ± 0.9623	5.9848 ± 0.4968	6.2958 ± 0.3207	13.2299 ± 0.7607	25.5691 ± 1.6216	30.0843 ± 2.4449	25.2061 ± 2.5839
Max Entropy	7.8507 ± 0.4303	15.0741 ± 1.3206	15.0347 ± 1.0066	6.8314 ± 0.5413	6.9403 ± 0.3436	14.9128 ± 0.934	26.9083 ± 1.7541	35.7583 ± 2.8943	27.6666 ± 2.806
Min Margin	7.1892 ± 0.4042	**13.5698 ± 1.1803**	14.7971 ± 1.0016	5.9913 ± 0.4924	6.3859 ± 0.3223	13.37 ± 0.7528	26.6803 ± 1.7192	31.1693 ± 2.4599	27.6666 ± 2.806
Least Confident	7.7155 ± 0.4257	13.7095 ± 1.1783	14.9207 ± 0.9703	6.6705 ± 0.5267	6.801 ± 0.3426	14.7509 ± 0.9102	27.1759 ± 1.7709	33.7974 ± 2.7318	27.6666 ± 2.806
Min Entropy	**7.1265 ± 0.4049**	13.8187 ± 1.2006	14.1918 ± 0.9293	6.0146 ± 0.512	**6.212 ± 0.3169**	13.0141 ± 0.7429	**25.482 ± 1.6521**	**29.8538 ± 2.4533**	**25.0856 ± 2.542**
Max Margin	7.1955 ± 0.4048	13.9401 ± 1.1996	14.3974 ± 0.9313	**5.976 ± 0.5048**	6.2504 ± 0.3186	13.0708 ± 0.7552	25.7807 ± 1.6722	30.1339 ± 2.4864	25.0856 ± 2.542
Most Confident	7.1416 ± 0.4057	13.8523 ± 1.1872	**14.1649 ± 0.9219**	6.0104 ± 0.5105	6.2158 ± 0.3162	**12.9624 ± 0.7459**	25.6142 ± 1.6515	30.1364 ± 2.4661	25.0856 ± 2.542

Table 2: KL Divergence ($X \cdot 10^3$) for each query strategy with BFGS as the learning kernel on different datasets after an additional 625 data items have been labeled (52 training iterations with 24 items learned per iteration).

3. Experiments

We introduce a multi-layer perceptron (MLP), with a (softmax) activation function on the output layer and two hidden layers (24 nodes in the first layer, 60 nodes in the second layer) with hyperbolic tangent activation functions as the model we seek to train. We used mean squared error as the loss function $\mathcal{L}$ and backpropagation with stochastic gradient descent optimization to minimize $\mathcal{L}$.

We implemented our active learning pipeline in Python 3.6 with Tensorflow 1.8.0 (see Figure 2 for pseudocode). It learns the LDL model $\mathcal{M}_{\theta_t}$ with a learning kernel $\mathcal{K}$ using a small seed set $\mathcal{T}_1$ of training data that is labeled by an *oracle* that provides ground truth label distributions. For each round $t \in \{1, \ldots\}$, it: (1) selects p items $x_{t_1}, \ldots, x_{t_p}$ from a large pool $\mathcal{P}_t$ of unlabeled data items according to a *query strategy* $\mathbf{q}$ and label distribution estimates provided by $\mathcal{M}_{\theta_t}$, (2) queries the oracle about each of the p selected items and (3) adds the items, along with the labels the oracle provides to $\mathcal{T}_t$, yielding a new training set $\mathcal{T}_{t+1}$, (4) runs the learning kernel again on $\mathcal{T}_{t+1}$, yielding a new model $\mathcal{M}_{\theta_{t+1}}$, and the next round begins.

4. Results

Regarding RQ1, Table 2 shows the KL divergence of each query strategy on each of the datasets for $\mathcal{M}_{\theta_{165}}$ ($||\mathcal{T}|| = 625$), obtaining via LDL active learning with BFGS as the learning kernel. It demonstrates that most-confidence and min-entropy strategies are consistently among the best performers, and outperform the baseline random strategy; the max-margin and min-margin strategy remain close to random sampling. However, least-confidence and max-entropy sampling strategies do not show any reasonable improvement for LDL, and often underperforms random selection.

A representative sampling of the learning curves (Figure 1) show, first, that Chebyshev distance and KL divergence are closely related throughout the learning process. Second, although it is closer

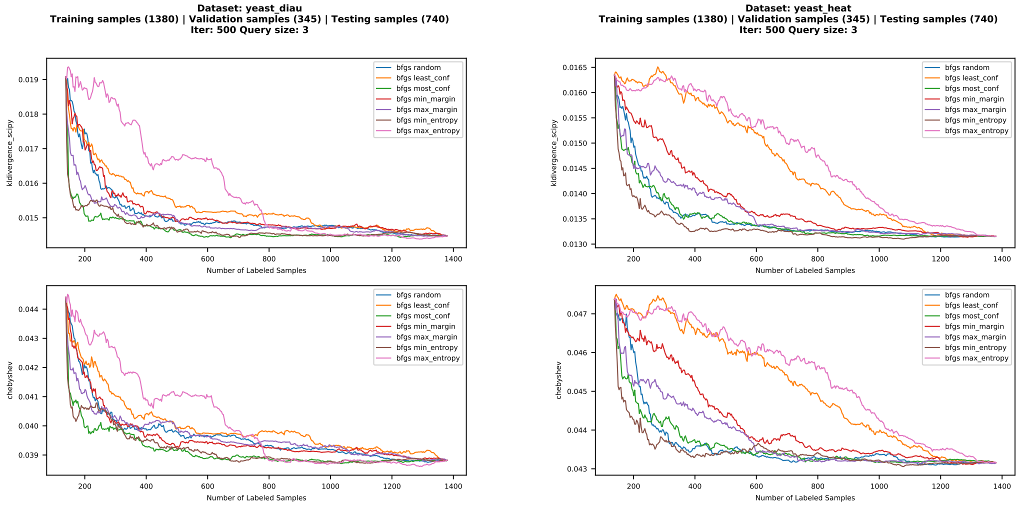

Figure 1: Performance of active learning with BFGS as the learning kernel on the Yeast-diau (left) and Yeast-heat (right) datasets using the sampling strategies we consider.

some cases than others, uncertainty sampling consistently underperforms random sampling, and certainty-based strategies work best.

Regarding RQ1, although direct numerical comparisons between LDL and single label learning are problematic, as they are distinct learning goals, one can at least compare the ranked differences in the performances of the query strategies. Here, Figure 1 paints a much fuzzier picture, with all of the learning curves much more tangled, compared to the LDL curves. It does appear that most confidence sampling again does very well, however, so do min-margin and max-entropy, two strategies that do not perform well on LDL.

Although uncertainty-based sampling strategies are popular for single- and multi-label learning, our experiments showed they are not applicable to some LDL problems. In spite of BFGS being designed for LDL, there may be inductive bias in the maximum entropy model against label distributions with higher levels of entropy. Indeed, given the quasi-linear nature of maximum entropy models, this seems plausible. Label distributions with higher entropy or lower confidence or margins exhibit less variance than those of lower entropy or higher confidence or margins, and thus provide a weaker signal for the learning algorithm to process. This suggests that models that are less sensitive to "vanishing gradients," such as hierarchical Bayesian networks, might be a good choice for purpose-built active LDL. Certainly, such models can represent distributions of distributions, and can thus properly decouple label distributions from any notion of degree of belief.

Another possibility is simply that there is so much entropy in the ground truth label distribution of each item that the items whose predicted label distributions have the lowest entropy are the most likely to be incorrect, and so obtaining their true labels and adding them to the training set results in the greatest improvement in prediction performance. However, if this were the case, one would expect that the difference in performance between certainly and uncertainty models would be greater in the datasets where the ground truth label entropy is greater. However, Table 2 shows the mean entropy of each set, and there does not appear to be such a correlation.

5. Conclusion & Future Work

We investigated six active learning strategies applicable to the LDL problem domain. While we were limited by the number of kernel tested in this study, experimental results indicate that certainty based active learning strategies can reduce the number of labels required for label distribution learning, and suggests that an LDL model can be quickly trained by strategically selecting examples with a certain probability distribution of class labels.

References

[1] Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *City*, 1(2):1, 2007.

[2] Xinyue Dong, Shilin Gu, Wenzhang Zhuge, Tingjin Luo, and Chenping Hou. Active label distribution learning. *Neurocomputing*, 436:12–21, 2021.

[3] Xinyue Dong, Tingjin Luo, Ruidong Fan, Wenzhang Zhuge, and Chenping Hou. Active label distribution learning via kernel maximum mean discrepancy. *Frontiers of Computer Science*, 17(4):174327, 2023.

[4] Michael B Eisen, Paul T Spellman, Patrick O Brown, and David Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.

[5] Xin Geng. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748, 2016.

[6] Xin Geng, Chao Yin, and Zhi-Hua Zhou. Facial age estimation by learning from label distributions. *IEEE transactions on pattern analysis and machine intelligence*, 35(10):2401–2412, 2013.

[7] Xin Geng, Qin Wang, and Yu Xia. Facial age estimation by adaptive label distribution learning. In *2014 22nd International Conference on Pattern Recognition*, pages 4465–4470. IEEE, 2014.

[8] Peng Hou, Xin Geng, Zeng-Wei Huo, and Jia-Qi Lv. Semi-supervised adaptive label distribution learning for facial age estimation. In *Thirty-First AAAI Conference on Artificial Intelligence*, pages 2015–2021, 2017.

[9] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[10] Tong Liu, Akash Venkatachalam, Pratik Sanjay Bongale, and Christopher Homan. Learning to predict population-level label distributions. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 1111–1120. ACM, 2019.

[11] Amirreza Shirani, Franck Dernoncourt, Paul Asente, Nedim Lipka, Seokhwan Kim, Jose Echevarria, and Thamar Solorio. Learning emphasis selection for written text in visual media from crowd-sourced label distributions. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 1167–1172, 2019.

[12] Stephen Wright and Jorge Nocedal. Numerical optimization. *Springer Science*, 35(67-68):7, 1999.

[13] Hongming Zhang, Wen Gao, Xilin Chen, and Debin Zhao. Object detection using spatial histogram features. *Image and Vision Computing*, 24(4):327–341, 2006.

6. Appendix A

```
ACTIVELEARNLDL$_{q,\mathcal{O},\mathcal{L}}$ (pool of unlabeled data $\mathcal{P}_1$,
    labeled seed data $\mathcal{T}_1$)

1. $t \leftarrow 1$.
2. Do until convergence:
    (a) $\theta_t \leftarrow \mathcal{K}(\mathcal{T}_t)$, where $\mathcal{K}$ is a learning algorithm
    (b) $x_{t_1}, \ldots, x_{t_p} \leftarrow q_{\theta_t}(\mathcal{P}_t)$, where $q$ is a query strategy
    (c) $y_{t_1}, \ldots, y_{t_p} \leftarrow \mathcal{O}(x_{t_1}), \ldots, \mathcal{O}(x_{t_p})$, where $\mathcal{O}$ is a ground truth oracle and $y_{t_1}, \ldots, y_{t_p}$ are label distributions
    (d) $\mathcal{T}_{t+1} \leftarrow \mathcal{T}_t \cup \{(x_{t_1}, y_{t_1}), \ldots, (x_{t_p}, y_{t_p})\}$
    (e) $\mathcal{P}_{t+1} \leftarrow \mathcal{P}_t \setminus \{x_{t_1}, \ldots, x_{t_p}\}$
3. return $\theta_t$
```

Figure 2: Pseudocode for the active learning algorithm studied here. It uses various *query strategies* to select, usually over a series of *Active Learning training iterations* a set training data that maximizes the expected learning outcomes. Unlike other active learning problems, for LDL, for each data item, the oracle provides a probability distribution over the labels, rather than a single (set of) labels.