Posterior Distribution Learning (PDL): A Novel Supervised Learning Framework

Enmei Tu[1], Jie Yang[1,*], Zhenghong Jia[2], and Nicola Kasabov[3]

[1] Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, China
hellotem@hotmail.com, jieyang@sjtu.edu.cn
[2] School of Information Science and Engineering, Xinjiang University, Urumqi, 830046, China
jzhh@xju.edu.cn
[3] The Knowledge Engineering and Discovery Research Institute,
Auckland University of Technology, Auckland, New Zealand
nkasabov@aut.ac.nz

Abstract. In order to obtain a robust supervised model with good generalization ability, traditional supervised learning method has to be trained with sufficient well labeled and uniformly distributed samples. However, in many real applications, the cost of labeled samples is generally very expensive. How to make use of ample easily available unlabeled samples to remedy the insufficiency of labeled samples to train a supervised model is of great interest and practical significance. In this paper we propose a new supervised learning framework, Posterior Distribution Learning (PDL), which could train a robust supervised model with very a few labeled samples by including those unlabeled samples into training stage. Experimental results on both synthetic and real world data sets are presented to demonstrate the effectiveness of the proposed framework.

Keywords: distribution learning, nonlinear regression, manifold classification.

1 Introduction

Supervised learning method is widely used in various areas, because once it is well trained, it builds a model in the whole input space and thus can predict any unseen sample with high speed and good accuracy. However, in order to obtain such a model with good generalization ability, one needs to train a supervised classifier with sufficient well labeled and uniformly distributed samples. But in many real applications, the cost of labeled samples is generally very expensive, as the labeling process usually takes much time and resource. This poses an obstacle to applying supervised learning method to those applications in which one needs to classify large amount of unlabeled samples with a very few labeled samples. How to make use of the large quantity of easily available unlabeled samples to train a supervised learning classifier and, meanwhile, to reduce the demand and requirement upon the labeled samples is still an interesting problem and of great practical significance.

* Corresponding author.

C.K. Loo et al. (Eds.): ICONIP 2014, Part I, LNCS 8834, pp. 86–94, 2014.

The problem of incorporating unlabeled samples to remedy the insufficiency of labeled samples to improve the performance of supervised learning method has been studied for many years and previous works generally can be casted into two categories. One is the co-training strategy [1-4]. These algorithms either require the data set has two or more distinct views, each of which is sufficient to make good classification alone, or require different classifiers can discover the diversity in the data set. But most of real world data sets do not meet these requirements. The other category is self-training [5-8]. These algorithms are restricted to binary classification tasks, in which only labeled samples of one class (called positive samples) are known and labeled samples of the other class (called negative samples, possibly a mixture of several classes) are unknown. A theoretical study of this strategy can be found in [9].

In this paper we propose a novel supervised learning framework which contains two steps to incorporate the distribution of unlabeled samples to train a robust supervised model with a few labeled samples: the posterior probability estimation and the posterior distributions regression. The first step estimates the posterior probabilities of each sample in a part of (or the whole) the data set and the second step fit a single multivariate posterior distribution function for all classes in the data space. When a new sample comes, the posterior distribution function can give directly its posterior probability to each class and thus the class label can be obtained using minimum Bayes error rule. Unlike previous works, the new supervised learning framework has the following characteristics: (1) it does not put constraints on the data set, such as multi-view property; (2) it is multi-class and more time efficient because it does not require training several classifiers iteratively. Instead, it fits a single model in the input space; (3) most importantly, it can greatly improve classification results by incorporating the distribution of unlabeled sample. Experimental results on both synthetic and real world data sets are presented to demonstrate validity and effectiveness of the proposed framework and algorithm.

2 Posterior Distribution Learning (PDL)

Let us consider a data set $X = \{x_1, x_2, ..., x_{t-1}, x_t, x_{t+1}, ..., x_n\}$. The first t samples are labeled by $\omega_i \in \{1, 2, ...C\}, i = 1...t$ and the rest samples are unlabeled. We use $X_T = \{x_1, x_2, ..., x_{t-1}, x_t\}$ and $X_U = \{x_{t+1}, x_{t+2}, ..., x_n\}$ to denote the labeled sample set and unlabeled sample set, respectively. With a little abuse of notation, we also use X (similarly, X_T and X_U) to denote the data matrix $(x_1, x_2, ..., x_n) \in R^{d \times n}$. We define a learning set $X_L = \{x_1, x_2, ..., x_{t-1}, x_t, x_{t+1}, ..., x_l\}$, which contains all the labeled samples and $l - t$ ($l \le n$) unlabeled samples in X.

2.1 A New Supervised Learning Framework

The diagram of the proposed supervised learning framework is shown in Figure 1.

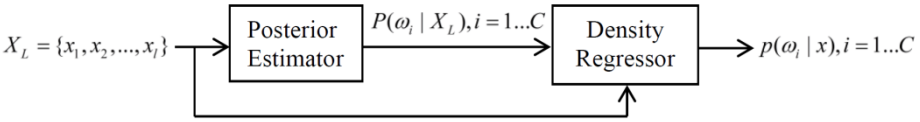

Fig. 1. Diagram of the proposed supervised learning framework that includes distribution of unlabeled samples into training stage

The posterior estimator computes $P(\omega_i \mid x_j)$, the posterior probability of $x_j, j = 1..l$ coming from class $\omega_i, i = 1..C$ .We require the output of the Estimator to have the following properties:

- $\forall x_j \in X_L, P(\omega_j = i \mid x_j) \geq 0; i = 1..C$.
- $\forall x_j \in X_L, \sum_{i=1}^{C} P(\omega_j = i \mid x_j) = 1$.
- $x_j \in X_L$, $P(\omega_j = i \mid x_j) = 1$ if x_j is labeled to class i .
- $x_j \in X_L$, $P(\omega_j = i \mid x_j) > P(\omega_j = k \mid x_j)$, $k \neq i$ and $k = 1..C$, if x_j is unlabeled and more similar to labeled samples in class i than other classes.

Let $F(x_j) = \left(P(\omega_j = 1 \mid x_j), P(\omega_j = 2 \mid x_j), ..., P(\omega_j = C \mid x_j)\right) \in R^C$, then $F(x_j)$ can be deemed as a sample point of a continuous function $F(x)$ and it can be regressed by the Density Regressor. After this, for a new sample x , its posterior probability can be obtained directly with $F(x)$ and the class label is determined by the minimum Bayes error rule.

2.2 Posterior Estimator: Propagate Posterior Probability from Labeled Samples to Unlabeled Samples

The posterior probabilities of a labeled sample are known, i.e. $P(\omega_j = i \mid x_j) = 1$ if x_j is from class *i* and $P(\omega_j = k \mid x_j) = 0, k \neq i$. We set $P(\omega_j = i \mid x_j) = 0, \ i = 1..C$ if x_j is unlabeled. Define an initial posterior probability matrix $F_T \in R^{l \times C}$ over X_L as $(F_T)_{ij} = P(\omega_j = i \mid x_j)$. Then we construct a full connected graph $G(X_L, A)$ on learning set X_L , where the adjacency matrix $A = \{A_{ij} \mid i, j = 1..l\}$ is computed by Gaussian kernel. Then the posterior probability matrix F_L is computed as follows

1. Compute $S = D^{-1/2} A D^{-1/2}$, where *D* is a diagonal matrix with $D_{ii} = \sum_{j=1}^{l} A_{ij}$.
2. Evaluate equation $\tilde{F} = I_{rate} S \tilde{F} + (I - I_{rate}) F_T$ repeatedly until convergence.
3. Normalize $F_L = G^{-1} \tilde{F}$, where G is a diagonal matrix with $G_{ii} = \sum_{j=1}^{C} \tilde{F}_{ij}$.

where I is the identity matrix and I_{rate} is a diagonal matrix defining the propagation rates on different directions. The method to determine I_{rate} will be described in Section 2.4. We treat the posterior probability of the labeled samples as information sources (or activation sources) and propagate this information iteratively over the graph [10, 11]. In the first iteration $\tilde{F} = F_T$.

2.3 Density Regressor: A Robust Multivariate Nonlinear Model to Regress the Posterior Distribution

Denote $F(x_j) = \left(P(\omega_j = 1 \mid x_j), P(\omega_j = 2 \mid x_j), ..., P(\omega_j = C \mid x_j)\right)$, i.e. $F(x_j)$ is the j^{th} row of F_L. We build a model $F(x) = \left(p(\omega = 1 \mid x), p(\omega = 2 \mid x), ..., p(\omega = C \mid x)\right) \in R^C$, where the posterior density function of class i is $p(\omega = i \mid x) = w_i^T \varphi_i(x) + b_i$ and $\varphi_i(\cdot)$ is a unknown nonlinear mapping function which maps the input space to the so-called feature space, whose dimensionality are unknown and can be very large (possibly infinite). Without loss of generality, we assume all mapping functions are same, written as $\varphi(x)$. So $F(x) = W^T \varphi(x) + b$, where $W = (w_1, w_2, ..., w_C)^T$, whose row dimension is same as that of the feature space. Note that $F(x)$ is a vector-valued function.

We compute $F(x)$ by solving the following optimization problem

$$\min J(W, b, E) = \frac{1}{2}\|W\|^2 + \frac{1}{2}\gamma \sum_{j=1}^{l} v_j \left\| r_j \right\|^2, \ s.t. \ F(x_j) = W^T \varphi(x_j) + b + r_j, \ j = 1...l \qquad (1)$$

where $E = (r_1, r_2, ..., r_l) \in R^{C \times l}$ is the error matrix and γ is a regularization parameter. It is worth noting that because the mapping function is unknown and the dimensionality of the feature space can be arbitrary large, problem (1) is quite different from ridge regression (or linear regression or Tikhonov regularization) and cannot be solved directly in primal form. $v = (v_1, v_2, ..., v_l)$ is a weight vector to reduce the sensitivity of the sum of squared error (SSE) to noise and outliers. Method to obtain v will be given in Section 2.4. One can also simply set v_k to 1. In this case, problem (1) becomes a regular least squares support vector machine [12].

It can be shown that problem (1) is a convex optimization problem. According to KKT conditions, the optimal solution meets the following linear equations

$$\begin{bmatrix} 0 & e^T \\ e & K + \eta V \end{bmatrix} \begin{bmatrix} b^T \\ \Lambda^T \end{bmatrix} = \begin{bmatrix} 0 \\ F_L^T \end{bmatrix} \qquad (2)$$

where K is a kernel matrix $K_{ij} = \varphi(x_i)^T \varphi(x_j)$ and $\Lambda = (\lambda_1, \lambda_2, ..., \lambda_l) \in R^{c \times l}$ is the Lagrange multiplier matrix. V is a diagonal matrix with $V_{ii} = v_i$ and $e = (1, 1, ..., 1)^T \in R^l$. After obtaining b and Λ, we can evaluate $F(x)$ at any point x by

$$F(x) = \Lambda \hat{K} + b \tag{3}$$

where $\hat{K} = (\varphi(x_1), \varphi(x_2), ..., \varphi(x_l))^T \varphi(x)$. Note that to compute $F(x)$ one only needs to know $\varphi(x)^T \varphi(x)$ even the concrete form of $\varphi(x)$ is unknown. This is the so-called kernel trick. We define $K_{ij} = \varphi(x_i)^T \varphi(x_j) = \exp(-\| x_i - x_j \|^2 / 2\sigma^2)$, the Gaussian kernel which performs well for most situations.

2.4 The Propagation Rate Matrix and the Weight Vector

The propagation rate matrix I_{rate} has to be designed so that the propagation speed alone high sample density direction is large while speed along sparse sample density direction is small, because sparse region usually means boundaries or overlapping region of classes and low speed can effectively suppress incorrect propagation. Here I_{rate} is computed by $(I_{rate})_{ii} = \exp\left(-\overline{d}_i^2 / 2\sigma^2\right)$, where $\overline{d}_i$ is the mean distances between x_i and its N nearest neighbors (N is empirically set to 20). Following a similar approach as in [11], it is easy to demonstrate the convergence of the algorithm.

The weight vector of $x_j, j = 1..l$ is set to $v_j = \max_{i=1..C} F(x_j) - \max_{k=1..C, k \neq i} F(x_j)$, the difference between its largest and the second largest posterior probability. Because equal posterior probabilities indicate that the sample is ambiguous to all classes and thus the posterior probability is less informative and unreliable, so the weight should be small.

3 Experimental Results

3.1 Synthetic Data Sets

We conduct experiments on two synthetic data sets: two-moon data set and two-circle data set. The data sets are shown in Fig 2, in which the black dots are unlabeled samples and the color shapes are labeled samples, one labeled sample for each class.

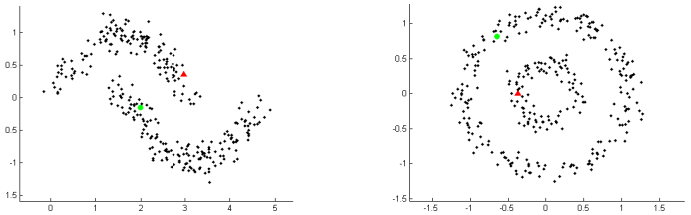

Fig. 2. Two synthetic data sets. The black dots are unlabeled samples. The green rounds and the red triangles are labeled samples, one for each class.

The experimental results of PDL are shown in Fig 3 and results of SVM in Fig 4. For SVM, we use radial basis kernel with $\sigma = 0.1$. From these results we can see that given a very few labeled samples and many unlabeled samples, PDL can train a

supervised model with a very good generalization ability. In contrast, traditional supervised learning method fails to obtain a reliable supervised model because the labeled samples contain very limited information to train the model with good generalization.

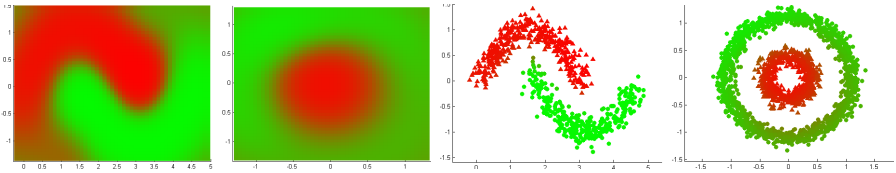

Fig. 3. Experimental results of PDL. Top: posterior probability distribution $F(x)$ learnt in the input space; bottom: classification result of out-of-sample data generated independently.

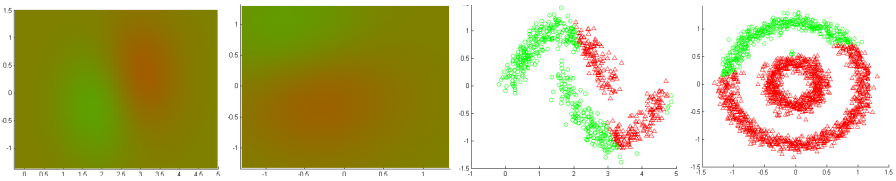

Fig. 4. Experimental results of SVM. Top: posterior probability distribution $F(x)$ learnt in the input space; bottom: classification result of out-of-sample data generated independently.

3.2 Comparison with Traditional Supervised Learning Method

We conduct experiments on 6 real world data sets: USPS hand-writing digit image and UCI repository data sets. The information of the data sets is listed in Table 1.

Table 1. Inforamtion of the experimental data sets

	USPS	segmentation	banknote	pendigits	skin	miniBoo
n	9298	2086	1348	10992	245057	130064
d	256	19	4	16	3	50
C	10	6	2	10	2	2

The split of X_L and X_U is random and the size of them are $0.7n$ and $0.3n$, respectively. The baseline algorithms are: support vector machine (SVM), k nearest neighbors (kNN), artificial neural networks (ANN), Naïve Bayes (NB) and decision tree (DT). We use radial basis kernel for SVM and three-layer back-propagation networks with $\lceil d/2 \rceil_{10}$ neurons in the hidden layer for ANN, where $\lceil x \rceil_{10}$ is the smallest integer greater than or equal to the $\max(x,10)$. Baseline algorithms are trained with X_T. Each algorithm runs 10 times and the final result is the average error rate of classifying X_U over 10 runs. We adopt the grid-search strategy to tune the parameter and the one producing lowest error rate is selected. Experimental results are shown in Table 2.

We can see that by taking the unlabeled samples into training stage, PDL outperforms traditional supervised learning algorithms. It is worth noting that for traditional supervised learning algorithms trained with a very few labeled samples, the error rate does not always decrease as the number of labeled samples increase. This indicates

Table 2. Classification error rate (%) of real world data set

	1 labeled sample / class						3 labeled samples / class						5 labeled samples / class					
	SVM	kNN	ANN	NB	DT	PDL	SVM	kNN	ANN	NB	DT	PDL	SVM	kNN	ANN	NB	DT	PDL
usps	59.77	63.66	65.61	45.97	84.73	**45.41**	49.79	41.95	42.20	97.85	74.77	**23.53**	43.16	31.08	37.87	85.47	69.14	**18.76**
segmentation	38.21	56.76	55.35	82.97	71.58	**30.27**	29.60	32.88	42.73	50.00	63.58	**20.10**	25.88	28.43	25.05	30.53	29.82	**18.21**
banknote	38.69	45.11	38.27	38.69	45.11	**8.72**	27.75	32.81	17.01	41.19	44.52	**4.94**	16.10	20.52	12.20	29.33	16.79	**2.89**
pendigits	89.63	67.19	47.21	66.59	89.08	**18.47**	89.57	28.17	21.68	44.18	80.56	**7.34**	90.39	21.16	19.77	35.99	61.04	**8.82**
skin	71.63	71.61	44.07	83.13	71.61	**15.95**	28.62	28.43	16.17	18.62	71.40	**12.69**	71.47	25.37	20.63	23.28	27.37	**7.96**
miniBoo	71.80	71.80	29.97	95.11	71.80	**22.27**	71.85	29.93	31.96	71.92	71.85	**19.11**	71.87	18.56	35.21	27.68	26.83	**18.31**

that given a very small number of labeled samples, some of the labeled samples may bring negative influence to the supervised model and the impact can be very large because of insufficiency of training samples. But for PDL, this negative effect is overcome because the posterior propagation algorithm performs a kind of graph diffusion, so even a few and not well positioned labeled samples can be utilized correctly. This is the key difference between PDL and traditional supervised learning method for training a supervised model.

3.3 Comparison with the State-of-Art Algorithms

We also compare the proposed PDL with two recently reported algorithms, the Tri-training (TriT) [13] and the virtual label regression (VLR) [14], which also train supervised learning classifiers using both labeled and unlabeled samples. For the two algorithms, we use the parameters provided in the paper. These algorithms and PDL are all trained with X_L and then used to classify X_U. The average results over 10 runs are reported in Table 3.

Table 3. Classification error rate (%) of real world data set

	1 labeled sample / class			3 labeled samples / class			5 labeled samples / class		
	TriT	VLR	PDL	TriT	VLR	PDL	TriT	VLR	PDL
usps	68.12	55.22	**45.41**	50.22	41.45	**23.53**	33.44	36.16	**18.76**
segmentation	74.49	31.08	**30.27**	40.35	28.43	**20.10**	25.58	24.46	**18.21**
banknote	41.88	22.81	**8.72**	22.07	**4.91**	4.94	6.47	**1.38**	2.89
pendigits	84.57	40.76	**18.47**	41.28	29.20	**7.43**	23.49	28.91	**8.82**
skin	54.33	32.96	**15.95**	34.21	17.46	**12.69**	21.42	17.92	**7.96**
miniBoo	39.00	32.82	**22.27**	20.72	**18.11**	19.11	**17.22**	27.65	18.31

We can see that PDL can achieve better results for most of the cases. For tri-training or co-training algorithms, it is easy to introduce noise labels as the training set grows and the impact of these noise labels can be very large. For VLR, it regresses a linear model using the discrete class-indicator vector of each sample, so it can hardly capture the nonlinearity and continuousness of the posterior density functions. In contrast, PDL first propagates posterior information from labeled samples to

unlabeled samples on graph and then regresses a nonlinear model in the input space. But graph has a tight relationship with the manifold structure, thus the underlying manifold information is thus also encoded in the distribution function $F(x)$. Therefore PDL shows a promising potential for classifying manifold-distributed data set.

4 Discussions and Conclusions

In this paper we developed a novel two-step framework to learn a robust supervised model using a very few training samples and plenty of unlabeled samples. The framework first propagates posterior information from labeled samples to unlabeled samples and then regresses a multivariate posterior function in input space. As the experiments demonstrated, the proposed method can greatly reduce the number of training samples and thus reduce human burden to obtain labeled samples. This has not only theoretical interest but also great practical value. In the future we will focus on theoretical analysis and extending the framework to other existing supervised learning algorithms.

Acknowledgements: This research is partly supported by NSFC, China (No: 61273258，31100672, 61375048), Ph.D. Programs Foundation of Ministry of Education of China (No.20120073110018).

References

1. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: COLT, pp. 92–100. ACM (1998)
2. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM. Machine Learning 39, 103–134 (2000)
3. Amini, M.-R., Gallinari, P.: The use of unlabeled data to improve supervised learning for text summarization. In: SIGIR, pp. 105–112. ACM (2002)
4. Goldman, S., Zhou, Y.: Enhancing supervised learning with unlabeled data. In: ICML, pp. 327–334. Citeseer (2000)
5. Denis, F.: PAC learning from positive statistical queries. In: Richter, M.M., Smith, C.H., Wiehagen, R., Zeugmann, T. (eds.) ALT 1998. LNCS (LNAI), vol. 1501, pp. 112–126. Springer, Heidelberg (1998)
6. Li, X., Liu, B.: Learning to classify texts using positive and unlabeled data. In: IJCAI, vol. 3, pp. 587–592 (2003)
7. Liu, B., Lee, W.S., Yu, P.S., Li, X.: Partially supervised classification of text documents. In: ICML, vol. 2, pp. 387–394. Citeseer (2002)
8. Chi, M., Bruzzone, L.: A semilabeled-sample-driven bagging technique for ill-posed classification problems. IEEE Geoscience and Remote Sensing Letters 2, 69–73 (2005)
9. Lee, W.S., Liu, B.: Learning with positive and unlabeled examples using weighted logistic regression. In: ICML, vol. 3, pp. 448–455 (2003)

10. Shrager, J., Hogg, T., Huberman, B.A.: Observation of phase transitions in spreading activation networks. Science 236, 1092–1094 (1987)
11. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: NIPS, pp. 595–602 (2004)
12. Suykens, J.A., Vandewalle, J.: Least squares support vector machine classifiers. Neural Processing Letters 9, 293–300 (1999)
13. Zhou, Z.-H., Li, M.: Tri-training: Exploiting unlabeled data using three classifiers. IEEE Transactions on Knowledge and Data Engineering 17, 1529–1541 (2005)
14. Nie, F., Xu, D., Li, X., Xiang, S.: Semisupervised dimensionality reduction and classification through virtual label regression. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 41, 675–685 (2011)