PLDL: A Novel Method for Label Distribution Learning

Venkatanareshbabu Kuppili, Mainak Biswas, and Damodar Edla
Computer Science and Engineering, National Institute of Technology Goa, India

Abstract: *The nature, volume and orientation of data have been changed a lot in the last few years. The changed situation has beckoned data scientists to modify traditional algorithms and innovate new methods for processing new type of high volume, extremely complex data. One of the challenges is label ambiguity in the data, where the distribution of the significance of the labels matters. In this paper, a new method named Probabilistic Label Distribution Learning (PLDL) has been proposed for a computing degree of the belongingness. It is based on a proposed new Label Probability Density Function (LPDF) derived from Parzon estimate. The LPDF has been used in Algorithm Adoption K-Nearest Neighbors (AA-KNN) for Label Distribution Learning (LDL). Probability density estimators are used to estimate this ambiguity for each and every label. The overall degree of the belongingness of unseen instance has been evaluated on various real datasets. Comparative performance evaluation in terms of prediction accuracy of the proposed PLDL has been made with Algorithm adaptation KNN, Multilayer Perceptron, Levenberg-Marquardt neural network and layer recurrent neural for Label Distribution Learning. It has been observed that the increase in prediction accuracy for the proposed PLDL is highly statistically significant for most of the real datasets when compared with the standard algorithms for LDL.*

Keywords: *Multi-label classification, data mining, label distribution learning, probability density function*.

Received September 19, 2016; accepted January 16, 2018

1. Introduction

Multi-Label Learning (MLL) [27] has succeeded Single-Label Learning (SLL) [30] where multi-label classification allows the instances to belong to more than one class, i.e, in music classification a song may belong to more than one genre. Multi-label classification algorithms has existed for a long time. Maximum entropy model [31] have been used for multi-label classification. In Biology, protein function for multi-label classification has been discussed [29]. A multi-label classification for music categorization [14] and multi-categorical algorithm semantic scene classification [2] algorithm has also been developed e.g., in semantic scene classification, a photograph can belong to more than one conceptual class, such as sunsets and beaches at the same time. Several multi-label classification methods are being developed by researchers over multi-labeled datasets to improve efficiency and precision. Random k-label sets were used for random subset of labels called Label Power set (LP) to deal with application domains dealing in large number of labels [23, 24]. The proposed method is called Random k Labelset (RAkEL) which deals with creating strategies for creating LPs, which is, creating subsets of labels, that exists in the dataset as a different class value of a single-label classification task. A method [18] which integrates and develops the concepts of random subspace [12], bagging [3] and random k-labelsets [24] ensemble learning methods to form an approach to classify multi-labeled data. In here multi-label classifiers are trained by using the randomly selected subsets. At the end of iteration, optimized parameters are selected and the ensemble MLC classifiers are constructed. Pruned sets has been used to perform multi-label classifications [20]. Classifier Chains (CC) on Binary Relevance (BR) methods have been used for multi-label classification [21]. In Classifier Chain method, the input domain is defined as $X^{d} \epsilon$ R for all possible attribute values. An instance is defined by a vector of d-attribute values x={x_1,x_2,…,x_d}. The set L={1, 2,…, l} is the output domain of all possible labels. Each instance x is associated with a subset of these labels y={y_1, y_2, …, y_l}, where $\mathrm{y_j}$ =1 if x instance belongs to label j, 0 otherwise. Read inducted a relationship by appending, $\mathrm{y_j}$ labels, with attribute values of x, one by one, to give y_{j+1}←$x_{1,2,\ldots d}$. $y_{1,2,\ldots l}$, which creates a chain. Read [21] used both probabilistic Probabilistic Classifier Chains (PCC) and Ensemble Chain Sequence (ECC), to compute classification. The traditional mining algorithms ought to be seen for label ambiguity in new era of data mining algorithms. Label Distribution Learning (LDL) [7, 8, 9] is a new way to view MLL. Instances are defined through the degree to which it is represented by its labels or classes, form the basis for Label Distribution Learning. LDL can be seen as a new way to view whole data mining. Traditional data mining operations have been always discrete in their operations, that is, classification operation have always been mutually exclusive. LDL has removed this restriction in a way that any instance can be defined through its confidence or degree of belongingness to all the labels.

A more natural way to label an instance x is to assign a real number d_x^y to each possible label y, representing the degree to which y describes x. For example, if x represents a protein, y represents a cancer, then d_x^y should be the expression level of the protein x in the cancer y. Further, suppose that the label set is complete, i.e., using all the labels in the set can always fully describe the instance. Then, $\sum_y d_x^y = 1$ such d_x^y is called the description degree of y to x. For a particular instance, the degrees of belongingness of all the labels can be seen as a probability distribution [25]. Data mining algorithms can be adapted to LDL in various ways. Geng [9] has proposed that there are three ways in which older and newer problems can be resolved through LDL, which are:

a. Problem Transformation: Convert single-label examples into weighted single-label examples, i.e., each of *n* single-label instance is transformed to *c* single-label examples such that it forms a *c* x *n* matrix, where each value represents the degree d_x^y, *x* varies from *i*=1..*n* instances, and *y* varies from *j*=1..*c*, for *c* labels. A machine learning algorithm must be able to predict confidence/probability or degree of belongingness $d_{x_i}^{y_j}$=*P*(y_j,x_i)for each label y_j
b. Algorithm Adaptation: Some of the prevalent algorithms can be naturally extended to deal with label distributions. The k-NN algorithm [13, 27] is one such algorithm that can be adapted for LDL. Given a new instance x, its nearest neighbors are first found in the set. Then, the mean of the label distributions of all the nearest neighbors is calculated as the label distribution of x. This adapted algorithm is denoted by Algorithm Adoption K-Nearest Neighbors (AA-kNN), where 'AA' is the abbreviation of 'Algorithm Adaptation'.
c. Specialized Algorithm: Certain algorithms meet criteria of LDL exquisitely. Geng *et al*. [7, 8] proposed two algorithms CPNN and IIS-LLD for facial age estimation, which had datasets meeting all criteria of LDL dataset.

In view of algorithm adaptation, AA-kNN has been developed by Geng [9]. K-Nearest Neighbors have been used extensively in data mining operation such as classification for both single-label [11] and multi-label [28]. In this paper, new method has been proposed for LDL [9]. Every label or category has PDF which is derived from degree of belongingness of each label of the data samples. A way for computation of PDF is through PDF estimators [4, 17, 18], since computation of PDF is complex task [19]. Cacoullos [4] extended Parzen's [19] estimates and showed the special case that multivariate kernel is a product of univariate kernels. If an unseen instance is compared with instances of a particular label or category, it shows asymptotic approach of mentioned instance towards the particular label. This asymptotic approach can be approximated to form the degree of belongingness in LDL.

The remaining sections of the paper are organized as follows. Section 2 explains the derivation of proposed Label Probabilistic Derivation Function (LPDF) and section 3 discusses proposed algorithm for Probabilistic Label Distribution Learning (PLDL). Section 4 presents comparative performance evaluation of PLDL with other well-known methods for LDL on real datasets and finally concluding remarks are given in section 5.

2. Proposed Method

This section deals about new method named PLDL. Probabilistic density function of instances belong to a particular class can describe about the nature of the group of instances [25]. This motivated us to develop a novel algorithm named Probabilistic Label Distribution Learning. It is based on a novel Label based Probability Density Function (LPDF) derived from Parzen estimate. The PLDL is useful to compute the degree of belongingness of multiple categories of an unseen instance. The unseen instance's attribute values are compared with instances' attribute values for a particular label or category. It reflects test instance's asymptotic approaching towards density of particular label or category. Since these comparison is carried out with every label or category this asymptotic approach can be quantified to form degree of belongingness for each label. The process to develop this model is given in section 2.1.

2.1. Derivation for Label Probabilistic Distribution (LPDF)

Parzen [19] in his classical paper showed various PDF estimators which asymptotically reaches to its parent density provided it is continuous.

Parzen [19] showed how one may construct a family of estimates of *f*(*X*), which is consistent at all points *X* in which the PDF is continuous. It is given by

$$f(X^j) = \frac{1}{m\lambda} \sum_{i=1}^{m} \omega\left(\frac{X_i^j - X_i^{tr}}{\lambda}\right) \quad (1)$$

Where

m denotes number of attributes/features

X^{tr} denotes neighborhood pattern

X^j denotes test pattern

ω(y) denotes weighting function

λ represents the size of Parzen window.

Parzen results can be extended [4] to estimate special case that the multivariate kernel is a product of univariate kernels. The extension principle is given by Equation (2) which is:

$$f^{l}(X^{j}) = \frac{1}{(2\pi)^{\frac{m}{2}}\sigma^{m}} \cdot \frac{1}{m} \sum_{i=1}^{m} exp\left(-\frac{\left(x_i^j - x_i^{tr}\right)^{T}(x_i^j - x_i^{tr})}{2\sigma^2}\right) \qquad (2)$$

Where
m denotes number of attributes
X^{tr} denotes existing pattern
X^{j} denotes test pattern σ denotes smoothing operatorσdenotessmoothingparameter

LPDF is derived from the above estimators and designed in such a way that it can learn label distributions of given data. In LDL, we try to compute degree of belongingness of any incoming instance for any category or label. Given the set of neighborhood instances X^j ϵ (X^1, X^2,…, X^n) and their degrees of belongingness defined, that is , $d_{x^{tr}}^{l} \in D^{tr}$ is known where *l* ϵ (1, 2, …, L) ,and D^{tr} is the set of all degree of belongingness to for pattern X^{tr} with the condition that ,

$$\sum_{l=1}^{L} d_{x^{tr}}^{l} = 1 \qquad (3)$$

Where *tr* ϵ (1, 2,…, n) . Since probability estimators shows closeness of any test instance to its parent category's density level using each existing pattern, we multiply this closeness with degree of belongingness of each existing pattern to get an overall estimate of belongingness of given test instance. From these overall estimates LPDF or $f_{LDL}(x)$ as computed as shown in Equation (4) for category l where *l* ϵ (1, 2, …, L)

$$f_{LDL}^{l}(X_j)^{tr} = d_{X^{tr}}^{l} * \left\{\frac{1}{(2\pi)^{\frac{m}{2}}\sigma^{n}} \cdot \frac{1}{m} \sum_{i=1}^{m} exp\left(-\frac{\left(x_i^j - x_i^{tr}\right)^{T}\left(x_i^j - x_i^{tr}\right)}{2\sigma^2}\right)\right\} \qquad (4)$$

Where $d_{X^{tr}}^{l}$ denotes degree of belongingness of existing pattern *tr* for category *l*

Next, the degree of belongingness/confidence for each particular label are added

$$f_{LDL}^{l}(X_j) = \sum_{tr=1}^{N} f_{LDL}^{l}(X_j)^{tr}\ \mathrm{f}_{\mathrm{LDL}}^{\mathrm{l}}(\mathrm{X_j}) = \sum_{\mathrm{tr}=1}^{\mathrm{kNN}} \mathrm{f}_{\mathrm{LDL}}^{\mathrm{l}}(\mathrm{X_j})^{\mathrm{tr}} \qquad (5)$$

N denotes number of K-nearest neighbors.

After calculating estimate of all categories and adding them all, $Sum(f_{LDL})$ is obtained

$$Sum(f_{LDL}) = \sum_{l=1}^{L} f_{LDL}^{l}(X_j) \qquad (6)$$

The degree of belongingness is obtained by dividing each estimation with the total sum. The degree of belongingness of test instance for category *A* is given by:

$$d_{x_j}^{l} = \frac{f_{LDL}^{l}}{Sum(f_{LDL})} = \frac{f_{LDL}^{l}(x_j)}{\sum_{l=1}^{L} f_{LDL}^{l}(x_j)} \qquad (7)$$

Equation (7) satisfies condition Equation (3), for example, if there are only two labels *A* and *B*, the summation would result in unity

$$d_{test\ prediction}^{A} + d_{test\ prediction}^{B} = \frac{f_{LDL}^{A}}{f_{LDL}^{A}(x) + f_{LDL}^{B}(x)} + \frac{f_{LDL}^{B}}{f_{LDL}^{A}(x) + f_{LDL}^{B}(x)} = 1$$

The Equation (7) is given as a nutshell in Equation (8)

$$d_{test}^{l} = \frac{\sum_{k=1}^{N}\left[d_k^l \cdot \left\{\frac{1}{(2\pi)^{\frac{m}{2}} * \sigma^{m}} \cdot \frac{1}{m} \cdot \left(\sum_{i=1}^{m} \exp\left(\frac{\left(t_i^{test} - t_i^{k}\right) * \left(t_i^{test} - t_i^{k}\right)^{T}}{2\sigma^2}\right)\right)\right\}\right]}{\sum_{l=1}^{L}\left[\sum_{k=1}^{N}\left[d_k^l \cdot \left\{\frac{1}{(2\pi)^{\frac{m}{2}} * \sigma^{m}} \cdot \frac{1}{m} \cdot \left(\sum_{i=1}^{m} \exp\left(\frac{\left(t_i^{test} - t_i^{k}\right) * \left(t_i^{test} - t_i^{k}\right)^{T}}{2\sigma^2}\right)\right)\right\}\right]\right]} \mathrm{d}_{\mathrm{test}}^{\mathrm{l}} = \frac{\sum_{\mathrm{k}=1}^{\mathrm{kNN}}\left[\mathrm{d}_{\mathrm{k}}^{\mathrm{l}} \cdot \left\{\frac{1}{(2\pi)^{\frac{\mathrm{m}}{2}} * \sigma^{\mathrm{m}*\mathrm{f}}} \cdot \frac{1}{\mathrm{m}} \cdot \left(\sum_{\mathrm{i}=1}^{\mathrm{m}} \exp\left(\frac{\left(\mathrm{t_i^{test}}\text{-}\mathrm{t_i^k}\right) * \left(\mathrm{t_i^{test}}\text{-}\mathrm{t_i^k}\right)^{\mathrm{T}}}{2\sigma^2}\right)\right)\right\}\right]}{\sum_{\mathrm{l}=1}^{\mathrm{L}}\left[\sum_{\mathrm{k}=1}^{\mathrm{kNN}}\left[\mathrm{d}_{\mathrm{k}}^{\mathrm{l}} \cdot \left\{\frac{1}{(2\pi)^{\frac{\mathrm{m}}{2}} * \sigma^{\mathrm{m}*\mathrm{f}}} \cdot \frac{1}{\mathrm{m}} \cdot \left(\sum_{\mathrm{i}=1}^{\mathrm{m}} \exp\left(\frac{\left(\mathrm{t_i^{test}}\text{-}\mathrm{t_i^k}\right) * \left(\mathrm{t_i^{test}}\text{-}\mathrm{t_i^k}\right)^{\mathrm{T}}}{2\sigma^2}\right)\right)\right\}\right]\right]} \qquad (8)$$

Where
N denotes the number of K-nearest neighbors
k denotes existing pattern for *k*th nearest neighbor
d_k^l denotes degree of belongingness for *k*th nearest neighbor d_{test}^{l} denotes degree of belongingness for test pattern

The squared difference *diff* is obtained from squaring the difference between label distribution of predicted and actual test instance, which is given by Equation (8).

$$diff = sqrt \sum_{label=1}^{l} \left(d_{test\ prediction}^{label} - d_{test\ actual}^{label}\right)^{2} \qquad (9)$$

This *diff* shows the error accumulated over between actual and predicted distribution of degrees by application of any methodology. The lesser the accumulated error the better the methodology for LDL.

Algorithm 1: PLDL (X^j, DX^j)

```
# Input: the existing dataset
# Compute average diff as MSE between predicted and desired label distribution
# Determine test case instaces X^j within dataset tr using K10 cross-validation.
# DX^j consists of all label distribution of X^j
# Find K-Nearest Neighbors of X^j based on Euclidean Distance
# Initialize Sum, Array, SumArray, D'X^j and diff with zeros with size of array in brackets
Sum = zeros(1,1)
Array = zeros(k-NN,L)
SumArray = zeros(1,L)
D'X^j = zeros(1,L)
diff = zeros(1,1)
for (k = 1 to N)
{
    for (l = 1 to L)
    {
      1. $f_{LDL}^{l}(\boldsymbol{X^j}) = \frac{1}{(2\pi)^{\frac{n}{2}}\sigma^{n}} \cdot \frac{1}{n} \sum_{i=1}^{n} exp\left(-\frac{\left(\boldsymbol{X_i^j} - \boldsymbol{X_i^k}\right)^{T}(\boldsymbol{X_i^j} - \boldsymbol{X_i^k})}{2\sigma^2}\right)$
      2. Array(k,l) = $f_{LDL}^{l}(\boldsymbol{X^j})$
      3. Array(k,l) = Array(k,l) $*\ d_{X^{tr}}^{l}$
    }
}
for (l = 1 to L)
{
    for ( k = 1 to N)
    {
    SumArray(1,l) = SumArray(1,l)+ Array(k,l)
    }
    Sum = Sum + SumArray(1,l)
}
```

```
for ( l = 1 to L){
    $D'X^j (1,l) = SumArray(1,l)/ Sum$
    $diff = diff + (D'X^j - DX^j)* (D'X^j - DX^j)T$
}
# MSE
$diff = sqrt(diff)$
```

3. Algorithm for Probabilistic Label Distribution Learning

LPDF has been applied to AA-kNN [9] in our algorithm shown in Algorithm 1. A random fraction of the dataset is used as test subject for evaluation of our algorithm. All the degree of belongingness/ distribution of the test dataset is stored in an array D_X^j. Each test case is taken and its K-nearest neighbors are found out using Euclidean distance. For each neighbor among k-neighbors Label Probabilistic Distribution Function (LPDF) for each label is found out. LPDF for each label are summed up to give '*l*' values and stored in SumArray. The values in SumArray are added to give Sum. All SumArray values when divided by Sum gives degree of belongingness/label distributions for test instance and stored in $D'_X{}^j$.Finally squared difference is computed between D_X^j and $D'_X{}^j$ to give the difference between predicted and actual output.

4. Results

Each dataset is tested through tenfold cross validation. List of all real world datasets [10] are given in Table 1. The description of the datasets is as follows:

4.1. SJAFFE Dataset

The first dataset is extension of widely used facial expression image databases, i.e., Japanese Female Facial Expression (JAFFE) [16]. The JAFFE database contains 213 grayscale expression images by 10 Japanese female models. A 243-dimension feature vector is extracted from each image by the method of Local Binary Patterns (LBP) [1]. Each of the images is given score by 60 persons on the 6 basic emotions (i.e., happiness, sadness, surprise, fear, anger, and disgust) with a 5-point scale. The average score of each emotion represents the emotion intensity. Instead of only considering the emotion with the highest score as most work on JAFFE does, the dataset Scored Japanese Female Facial Expression (SJAFFE) (Scored JAFFE) keeps all the scores and normalizes them into a label distribution over all the six emotion labels.

4.2. Yeast Datasets and its Variants

Dataset No.2 to Dataset No. 6 (from Yeast-cdc to Yeast-alpha) are real-world datasets [6] collected from biological experiments on the yeast Saccharomyces cerevisiae. For each dataset, the labels correspond to the discrete time points are collected during one biological experiment. The gene expression level at each time point is recorded and normalized. It provides a natural measure of the description degree/degree of belongingness of the corresponding label. The number of labels in the five Yeast Gene datasets along with SJAFFE is summarized in Table 1. The description degrees (normalized gene expression levels) of all the labels (time points) constitute a label distribution for a particular yeast gene. The proposed methodology has been compared with standard methods such as AA-kNN [9] multilayer perceptron [5], Levenberg-Marquardt [22] and layer recurrent neural network [15]. Number of hidden nodes used in multilayer perceptron are ten. The Levenberg-Marquardt neural network is also used for classification. A recurrent neural network is a class of artificial neural network where connections form directed cycles which creates an internal state of the network allowing it to exhibit dynamic temporal behavior. Unlike multilayer perceptron, RNNs can use their internal memory to process inputs of arbitrary sequences. Tables 2, 3, 4, 5, 6, and 7 indicates Mean Squared Error (M.S.E) obtained using PLDL and other standard algorithms for LDL on SJAFFEE and variants of yeast datasets. Table 2 shows M.S.E obtained using PLDL and other standard algorithms for LDL on SJAFFEE. It is observed that PLDL gives least MSE when compared with all other methods. Table 3 indicate simulation results on Yeast-cdc dataset. It has been observed from all simulations that PLDL prevails over other methods in terms of M.S.E.

Table 4 shows M.S.E obtained for PLDL and other standard LDL on Yeast_elu dataset. Ten rows of Table 4 are obtained during k10 fold cross validation. It has also been observed that PLDL outperforms other standard methods in terms of M.S.E. Table 5 shows M.S.E obtained for PLDL and other standard LDL on Yeast_spo5 dataset. Ten rows of Table 5 are obtained during k10 fold cross validation. It has also been observed that PLDL outperforms other standard methods in terms of M.S.E. Table 6 shows M.S.E obtained for PLDL and other standard LDL on Yeast_spoem dataset. Ten rows of Table 6 are obtained during k10 fold cross validation. It has also been observed that PLDL outperforms other standard methods in terms of M.S.E. Table 7 shows M.S.E obtained for PLDL and other standard LDL on Yeast alpha dataset. Ten rows of Table 6 are obtained during k10 fold cross validation. It has also been observed that PLDL outperforms other standard methods in terms of M.S.E. Table 8 shows the average M.S.E computed of all the ten cross validation trials of each method for each Dataset. Graphical representation of the Table 8 is shown in Figure 1. The least difference are shown in bold. It is observed that maximum simulations the proposed methodology gives far better result, than the prevalent ones. In order to test the performance of PLDL with other models in terms of MSE, t-test is applied for comparison of the mean of MSEs of ten cross validations for each of the dataset. The p-value for MSE is given in column 3, 4, 5, 6 in Table 9. All the p-values from Table 9 indicate that the MSE values for PLDL model w.r.t existing methods for LDL is statistically significant.

Table 1. List of Datasets used for LDL.

Dataset	Instances	Features	Labels
SJAFFE	213	243	6
Yeast-cdc	2,465	24	15
Yeast-elu	2,465	24	14
Yeast-spo5	2,465	24	4
Yeast-spo	2,465	24	6
Yeast-alpha	2,465	24	18

Table 2. M.S.E obtained using PLDL and other standard algorithms for LDL on SJAFFEE dataset.

SN.	PLDL	AA-kNN	Multilayer Perceptron	Le-Ma	Layrcenet
1	**0.102**	0.173	0.200	0.160	0.172
2	**0.129**	0.132	0.161	0.147	0.177
3	**0.131**	0.145	0.167	0.149	0.132
4	**0.145**	0.149	0.164	0.161	0.146
5	**0.145**	0.15	0.157	0.178	0.158
6	**0.142**	0.151	0.152	0.144	0.167
7	**0.136**	0.148	0.182	0.18	0.158
8	**0.147**	0.151	0.149	0.15	0.178
9	**0.146**	0.152	0.15	0.162	0.164
10	**0.134**	0.145	0.156	0.169	0.145

Table 3. M.S.E obtained using PLDL and other standard algorithms for LDL on Yeast-cdc dataset.

SN.	PLDL	AA-kNN	Multilayer Perceptron	Le-Ma	Layrcenet
1	**0.023**	0.027	0.029	0.028	0.028
2	**0.016**	0.029	0.029	0.029	0.029
3	**0.014**	0.029	0.029	0.029	0.029
4	**0.025**	0.029	0.028	0.028	0.028
5	**0.014**	0.029	0.029	0.029	0.029
6	**0.016**	0.029	0.029	0.028	0.028
7	**0.023**	0.03	0.029	0.03	0.029
8	**0.022**	0.028	0.028	0.028	0.028
9	**0.017**	0.028	0.028	0.027	0.027
10	**0.015**	0.028	0.028	0.028	0.028

Table 4. M.S.E obtained using PLDL and other standard algorithms for LDL on Yeast-elu dataset.

SN.	PLDL	AA-kNN	Multilayer Perceptron	Le-Ma	Layrcenet
1	**0.014**	0.028	0.027	0.028	0.028
2	**0.014**	0.027	0.027	0.027	0.027
3	**0.016**	0.028	0.028	0.027	0.028
4	**0.014**	0.028	0.027	0.027	0.027
5	**0.022**	0.028	0.028	0.028	0.028
6	**0.018**	0.028	0.028	0.029	0.028
7	**0.023**	0.029	0.028	0.028	0.028
8	**0.018**	0.028	0.027	0.027	0.027
9	**0.021**	0.028	0.028	0.028	0.028
10	**0.014**	0.027	0.027	0.027	0.027

Table 5. M.S.E obtained using PLDL and other standard algorithms for LDL on Yeast_spo5 dataset.

SN.	PLDL	AA-kNN	Multilayer Perceptron	Le-Ma	Layrcenet
1	**0.064**	0.119	0.123	0.121	0.12
2	**0.102**	0.118	0.122	0.119	0.118
3	**0.061**	0.121	0.121	0.124	0.119
4	**0.075**	0.120	0.117	0.121	0.118
5	**0.085**	0.116	0.116	0.115	0.118
6	**0.092**	0.125	0.126	0.128	0.127
7	**0.079**	0.115	0.117	0.120	0.115
8	**0.092**	0.123	0.122	0.123	0.121
9	**0.079**	0.121	0.119	0.118	0.122
10	**0.114**	0.118	0.118	0.118	0.119

Table 6. M.S.E obtained for PLDL and other standard LDL on Yeast_spoem dataset.

SN.	PLDL	AA-kNN	Multilayer Perceptron	Le-Ma	Layrcenet
1	**0.080**	0.082	0.083	0.083	0.082
2	**0.052**	0.083	0.083	0.081	0.082
3	**0.048**	0.084	0.085	0.084	0.084
4	**0.080**	0.085	0.084	0.085	0.085
5	**0.062**	0.082	0.082	0.082	0.081
6	**0.107**	0.145	0.156	0.169	0.145
7	**0.054**	0.083	0.082	0.082	0.083
8	**0.055**	0.082	0.083	0.084	0.082
9	**0.045**	0.085	0.082	0.082	0.084
10	**0.051**	0.086	0.086	0.088	0.087

Table7. M.S.E obtained for PLDL and other standard LDL on Yeast_alpha dataset.

SN.	PLDL	AA-kNN	Multilayer Perceptron	Le-Ma	Layrcenet
1	**0.020**	0.024	0.023	0.023	0.023
2	**0.013**	0.024	0.024	0.024	0.024
3	**0.017**	0.024	0.023	0.023	0.023
4	**0.012**	0.023	0.023	0.023	0.023
5	**0.013**	0.023	0.022	0.023	0.022
6	**0.016**	0.025	0.025	0.025	0.025
7	**0.014**	0.024	0.024	0.024	0.024
8	**0.012**	0.024	0.023	0.024	0.023
9	**0.013**	0.022	0.022	0.022	0.022
10	**0.021**	0.024	0.024	0.024	0.024

Table 8. Comparative performance evaluation of all methodologies for LDL.

SN.	Datasets	PLDL	AA-kNN	Multilayer Perceptron	Le-Ma	Layrcenet
1	**SJAFFE**	**0.136**	0.15	0.164	0.16	0.16
2	**Yeast-cdc**	**0.018**	0.029	0.029	0.028	0.028
3	**Yeast-elu**	**0.017**	0.028	0.028	0.028	0.028
4	**Yeast-spo5**	**0.084**	0.12	0.12	0.121	0.12
5	**Yeast-spoem**	**0.063**	0.09	0.091	0.092	0.09
6	**Yeast-alpha**	**0.015**	0.024	0.023	0.023	0.023

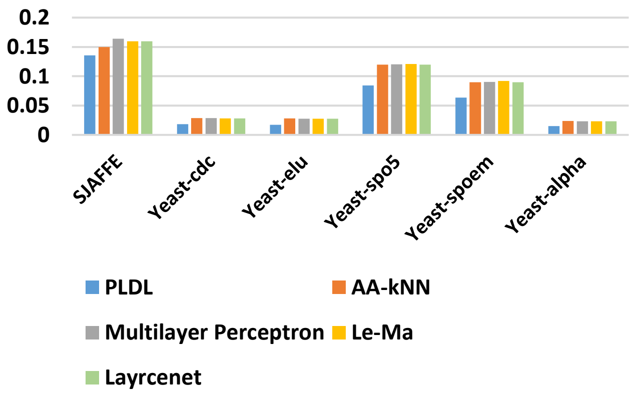

Figure 1. Graphical representation of Average M.S.E obtained using PLDL and exisrting algorithms for LDL on all datasets.

Table 9. Statistical significance of proposed method over other standard methods for LDL.

Datasets	PLDL with AA-kNN	PLDL with MP	PLDL with Le-Ma	PLDL with Layrcenet
SJAFFE	0.0599 (α=0.05992)	0.0124 (α=0.012399)	0.0021 (α=0.002081)	0.0060(α=0.0061)
Yeast-cdc	3.9e-05(α=3.9e-05)	2.8e-05(α=2.8e-05)	3.4e-05(α=3.4e-05)	4.1e-05(α=4.1e-05)
Yeast-elu	2.6e-06(α=2.6e-06)	3.5e-06(α=3.5e-06)	3.6e-06 (α=3.6e-06)	3.9e-06 (α=3.9e-06)
Yeast-spo5	1.0e-04 (α=1.0e-04)	8.4e-05(α=8.4e-05)	1.1e-04(α=1.1e-04)	7.9e-05(α=7.9e-05)
Yeast-spoem	1.6e-04 (α=1.6e-04)	2.4e-04 (α=2.4e-04)	4.9e-04 (α=4.9e-04)	1.4e-04 (α=1.4e-04)
Yeast-alpha	2.1e-05(α=2.1e-05)	2.5e-05(α=2.5e-05)	2.4e-05(α=2.4e-05)	2.7e-05(α=2.7e-05)

5. Conclusions

In this paper a new method PLDL has been proposed for Label Distribution Learning. A new function PDF has been proposed and it's been observed that it is highly capable of predicting label distributions than existing methods such as AA-kNN, Multilayer Perceptron, Levenberg-Marquardt Neural Network and Layer Recurrent Neural Network. Comparative performance of PLDL with the existing methods shows superiority of PLDL in terms of accuracy over other methods. The results of experiments suggests further investigation of application of PLDL in non-textual, non-numeric datasets and usage of the same in Big Data applications. PLDL can also be used for Natural Language Processing (NLP) such as Arab phonemes recognition [26].

Acknowledgement

We would like to thank MediaLab Asia for their support.

References

[1] Ahonen T., Hadi A., and Pietikainen M., "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037-2041, 2006.

[2] Boutell M., Luo J., Shen X., and Brown C., "Learning Multi-Label Scene Classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757-1771, 2004.

[3] Breiman L., "Bagging Predictors," *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.

[4] Cacoullos T., "Estimation of A Multivariate Density," *Annals of the Institute of Statistical Mathematics*, vol. 18, no.1, pp. 179-189, 1966.

[5] Chaudhuri B. and Bhattacharya U., "Efficient Training and Improved Performance of Multilayer Perceptron in Pattern Classification," *Neurocomputing*, vol. 34, no. 1, pp. 11-27, 2000.

[6] Eisen M., Spellman P., Brown P., and Botstein D., "Cluster Analysis and Display of Genome-Wide Expression Patterns," *Proceedings of the National Academy of Sciences*, vol. 95, no. 25, pp. 14863-14868, 1998.

[7] Geng X., Smith-Miles K., and Zhou Z., "Facial Age Estimation By Learning From Label Distributions," *in Proceedings of the 24th AAAI Conference on Artificial Intelligence*, Atlanta, pp. 451-456, 2010.

[8] Geng X., Yin C., and Zhou Z., "Facial Age Estimation By Learning from Label Distributions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2401-2412, 2013.

[9] Geng X., "Label Distribution Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1734-1748, 2016.

[10] Geng X. and Rongzi J., "Label Distribution Learning," *in Proceedings of IEEE 13th International Conference on Data Mining Workshops. IEEE Computer Society*, Washington, 2013.

[11] Han J., Pei J., and Kamber M., *Data Mining: Concepts and Techniques*, Elsevier, 2011.

[12] Ho T., "The Random Subspace Method for Constructing Decision Forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832-844, 1998.

[13] Larose D., *Discovering Knowledge in Data: An Introduction to Data Mining*, John Wiley and Sons, 2014.

[14] Li T., Ogihara M., and Li Q., "A Comparative Study on Content-Based Music Genre Classification," *in Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, Toronto, pp. 282-289, 2003.

[15] Liu Q. and Wang J., "A One-Layer Recurrent Neural Network with A Discontinuous Hard-Limiting Activation Function for Quadratic Programming," *IEEE Transactions on Neural Networks*, vol. 19, no. 4, pp. 558-570, 2008.

[16] Lyons M., Akamatsu S., Kamachi M., and Gyoba J., "Coding Facial Expressions with Gabor Wavelets," *in Proceedings of 3rd IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, pp. 200-205, 1998.

[17] Murthy V., "Estimation of Probability Density," *The Annals of Mathematical Statistics*, vol. 36, no. 3, pp. 1027-1031, 1965.

[18] Nasierding G., Kouzani A., and Tsoumakas G., "A Triple-Random Ensemble Classification Method for Mining Multi-Label Data," *IEEE International Conference on Data Mining Workshops*, Sydney, pp. 49-56, 2010.

[19] Parzen E., "On Estimation of a Probability Density Function and Mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065-1076, 1962.

[20] Read J., Bernhard P., and Holmes G. "Multi-Label Classification Using Ensembles of Pruned Sets," *in Proceedings of 8th IEEE International Conference on Data Mining*, Pisa, pp. 995-1000, 2008.

[21] Read J., Pfahringer B., Holmes G., and Frank E., "Classifier Chains for Multi-Label Classification," *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Berlin, pp. 254-269, 2009.

[22] Sapna S., Tamilarasi A., and Kumar M., "Backpropagation Learning Algorithm Based on Levenberg Marquardt Algorithm," *Computer Science and Information Technology*, vol. 2, no. 4, pp. 393-398, 2012.

[23] Tsoumakas G. and Katakis I., "Multi-label Classification: An Overview," *International Journal Data Warehousing and Mining*, pp. 1-13, 2006.

[24] Tsoumakas G., Katakis I., and Vlahavas I., "Random K-Labelsets for Multilabel Classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 7, pp. 1079-1089, 2011.

[25] Vapnik V., *Statistical Learning Theory*, Wiley, 1998.

[26] Zarrouk E. and Benayed Y., "Hybrid SVM/HMM Model for the Arab Phonemes Recognition," *The International Arab Journal of Information Technology*, vol. 13 no. 5, pp. 574-582, 2016.

[27] Zhang M. and Zhou Z., "ML-KNN: A Lazy Learning Approach to Multi-Label Learning," *Pattern Recognition*, vol. 40, no. 7, pp. 2038-2048, 2007.

[28] Zhang M. and Zhou Z., "A K-Nearest Neighbor Based Algorithm for Multi-Label Classification," *IEEE International Conference on Granular Computing*, Beijing, pp. 718-721, 2005.

[29] Zhang Y., Nur Z., and Milios E., "Narrative Text Classification for Automatic Key Phrase Extraction in Web Document Corpora," *in Proceedings of the 7th Annual ACM International Workshop on Web Information and Data Management*, Bremen, pp. 51-58, 2005.

[30] Zhou Z., Zhang M., Huang S., and Li Y., "Multi-Instance Multi-Label Learning," *Artificial Intelligence*, vol. 176, no. 1, pp. 2291-2320, 2012.

[31] Zhu S., Ji X., Xu W., and Gong Y., "Multi-Labelled Classification Using Maximum Entropy Method," *in Proceedings of The 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Salvador, pp. 274-281, 2005.

Appendix: Symbol Table

Table A1. Symbol table.

SN.	Symbol	Denotation
1	m	number of attributes/features
2	X^{tr}	neighborhood pattern
3	X^{j}	test pattern
4	$\omega(y)$	weighting function
5	λ	size of Parzen window
6	σ	smoothing operator
7	N	number of k-nearest neighbors
8	$d^{l}_{X^{tr}}$	degree of belongingness (*d*) of existing pattern X^{tr} for category *l*
9	d^{l}_{k}	degree of belongingness for kth nearest neighbor
10	d^{l}_{test}	degree of belongingness for test pattern
11	*diff*	error accumulated over between actual and predicted distribution of degrees

Venkatanareshbabu Kuppili PhD, is with the Machine Learning Group, Department of CSE, NIT Goa, India, where he is currently an Assistant Professor. He was with Evalueserve pvt. ltd, as a Senior Research Associate. He is also actively involved in teaching and research development for the Graduate Program in Computer Science and Engineering Department at the NIT Goa. He has authored a number of research papers published in reputed international journals.

Mainak Biswas M.Tech, completed his B.Tech Degree in Information Technology from Govt. College of Engineering and Ceramic Technology, Kolkata in 2007, M.Tech in Distributed and Mobile Computing from Jadavpur University, Kolkata in 2009 and is currently pursuing his PhD in NIT Goa.Previously he has served as Assistant Professor in OPJU, Raigarh in 2009-13 and MITS, Gwalior in 2014-15. He has published and presented papers in various International Journals and Conferences.

Damodar Edla PhD received B.Sc degree from Kakatiya University in 2004, M.Sc degree from University of Hyderabad in 2006. M.Tech. and PhD degree in computer science and engineering from ISM Dhanbad in 2009 and 2013 respectively. His research area is Data mining and Wireless Sensor Networks. He is currently Assistant Professor in CSE Department, NIT Goa. He has more than 20 research publications in reputed international journals and conferences.