



Poyiadzi, R., Santos-Rodriguez, R., & Twomey, N. (2018). Label Propagation for Learning with Label Proportions. In *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP 2018): Proceedings of a meeting held 17-20 September 2018, Aalborg, Denmark* (pp. 264-270). Institute of Electrical and Electronics Engineers (IEEE). <https://doi.org/10.1109/MLSP.2018.8517083>

Publisher's PDF, also known as Version of record

License (if available):  
CC BY-NC-SA

Link to published version (if available):  
[10.1109/MLSP.2018.8517083](https://doi.org/10.1109/MLSP.2018.8517083)

[Link to publication record in Explore Bristol Research](#)  
PDF-document

This is the final published version of the article (version of record). It first appeared online via IEEE at <https://ieeexplore.ieee.org/document/8517083> . Please refer to any applicable terms of use of the publisher.

## University of Bristol - Explore Bristol Research

### General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:  
<http://www.bristol.ac.uk/pure/about/ebr-terms>

# LABEL PROPAGATION FOR LEARNING WITH LABEL PROPORTIONS

*Rafael Poyiadzi, Raul Santos-Rodriguez and Niall Twomey*

Intelligent Systems Laboratory, University of Bristol, BS8 1UB, United Kingdom.

## ABSTRACT

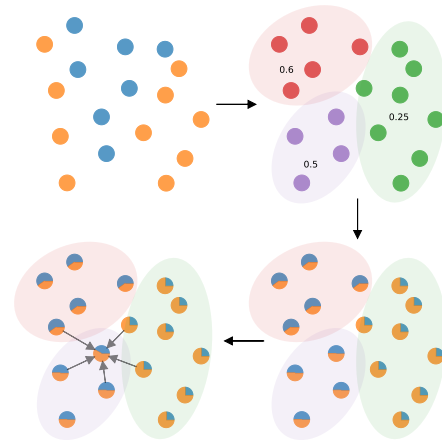
Learning with Label Proportions (LLP) is the problem of recovering the underlying true labels given a dataset when the data is presented in the form of bags. This paradigm is particularly suitable in contexts where providing individual labels is expensive and label aggregates are more easily obtained. In the healthcare domain, it is a burden for a patient to keep a detailed diary of their daily routines, but often they will be amenable to provide higher level summaries of daily behavior. We present a novel and efficient graph-based algorithm that encourages local smoothness and exploits the global structure of the data, while preserving the ‘mass’ of each bag.

## 1. INTRODUCTION

The whole spectrum of learning paradigms ranging from supervised to unsupervised learning is densely packed with different settings that have limited (or no) access to the true labels. For instance, in semi-supervised learning we have access to labels only for a subset of the dataset (usually small). Additionally, we can characterize the uncertainty in the labeling process when learning from noisy and partial labels [1, 2], using noise as a proxy in the former and a subset of labels per example in the later. Differently, in this paper we focus on learning from aggregated labels, also commonly referred to as Learning with Label Proportions (LLP) [3, 4]. In this setting, we assume the data comes in the form of bags of examples and for each of which we are given proportion of labels corresponding to each class. Hence, the supervision ranges from the fully supervised case (as many bags as examples) to the class priors of the whole dataset (one bag).

Applications for LLP range from elections (labels are votes and bags are specified demographic areas) and healthcare (diagnosed diseases are released in proportions according to ZIP code area) [4]. Our motivation stems from the healthcare domain, where the Automatic Activity Recognition in Smart Home environments is key for the monitoring of health conditions such as dementia or diabetes [5]. The task

This work is supported by EurValve (Personalised Decision Support for Heart Valve Disease), Project Number: H2020 PHC-30-201 and the Continuous Behavioural Biomarkers of Cognitive Impairment (CUBOld) project funded by the UK Medical Research Council Momentum Awards under Grant MC-PC-16029.



**Fig. 1.** Illustration of the Label Propagation for Learning with Label Proportions (LP-LLP) procedure. The first step shows the bag generation and in the second step we use each bag’s label proportion as a soft label for each of its points. In the last step we see that a point is influenced by its neighbors.

involves the classification of sensor data as belonging to a predefined set of Activities of Daily Living (ADL). The problem is usually addressed in the supervised setting, assuming that a human annotator (patient) has manually labeled enough examples of each of the classes of interest. However, this approach is not realistic when deploying this systems in the wild, as patients are reluctant to provide detailed labels. LLP offers a more amenable annotation strategy since annotations are aggregates of labels over time e.g. ‘today I slept 8 hours and watched TV for 2 hours’.

Our proposed method is based on a graph-based formulation of the problem where data-points are represented as nodes and their relationships are encoded via edges with associated weights. Each node is initially assigned a partial label derived from its bag’s bag proportion. Then each node’s label is updated based on its neighbors. The problem of preserving estimated bag proportions is relaxed to the one of preserving the total ‘mass’ of each bag. Our approach aims at recovering the true labels based on the assumption of local smoothness and by exploiting the global structure in the data.

Our contributions are twofold. Firstly, we introduce an elegant and novel graph-based approach to learning from label proportions building upon Label Propagation [6]. The main intuition is illustrated in Fig. 1. Secondly, we use label proportions as the only supervision when predicting activeness in Smart Home environments for healthcare.

*Problem Formulation* We assume that we have a set of observations  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  where  $\mathbf{x}_i \in \mathbb{R}^d$ . The true labels,  $\mathbf{y} = \{y_1, \dots, y_n\}$  with  $y_i$  being the label of observation  $\mathbf{x}_i$ , also exist but are hidden. This set is separated into distinct bags  $\mathbf{X} = \bigcup_{k=1}^K \mathbf{B}_k$ , where each  $\mathbf{B}_k$  corresponds to the subset of points assigned to the  $k$ -th bag, and  $\mathbf{B}_k \cap \mathbf{B}_j = \emptyset, \forall k, j \in [K]$ . Moreover, for each bag  $\mathbf{B}_k$  we have access to its class proportions,  $\boldsymbol{\pi}_k = \{\pi_{k,1}, \dots, \pi_{k,c}\}$ , where  $\sum_{h=1}^c \pi_{k,h} = 1$ ,  $\pi_{k,h} \geq 0$ , with  $\pi_{k,h}$  corresponding to the proportion of class  $h$  in bag  $\mathbf{B}_k$  and  $c$  being the number of classes.

The paper is structured as follows. In Sec. 2 we review the related work. In Sec. 3 we introduce the basic formulation of Label Propagation and derive the algorithm that incorporates label proportions. In Sec. 4 we respectively present the empirical analysis in both real and synthetic datasets. Sec. 5 is devoted to the conclusions.

## 2. RELATED WORK

Learning from bags of examples can be tracked back to tasks such as Multiple-Instance Learning [7], where the learner is provided with logical statements indicating the presence of a class in a bag. For example, in binary classification, a bag would have a positive label if it had at least one positive point in it, while it would be labeled as negative otherwise.

Recently, more general methods to deal with label proportions have attracted a wide interest and some of these are discussed in subsequent subsections. Existing algorithms can be loosely placed in three categories. Bayesian approaches such as [8] approach the problem by generating labels consistent with bag proportions. In [3] the authors propose an algorithm that relies on the properties of exponential families and the convergence of the class mean operator, computed from the means and label proportions of each bag. Lastly, maximum-margin approaches [9, 10] pose the problem as either an extension of maximum-margin clustering [11] or Support Vector Regression. We focus on the latter two categories as they outperform the first one experimentally [9].

### 2.1. Conditional Exponential Families

Let  $\mathcal{X}$  and  $\mathcal{Y}$  denote the space of the observations and the (discrete) label space respectively, and let  $\phi(\mathbf{x}, y) : \mathcal{X} \times \mathcal{Y} \rightarrow \mathcal{H}$  be a feature map into a Reproducing Kernel Hilbert Space  $\mathcal{H}$  with kernel  $k((\mathbf{x}, y), (\mathbf{x}', y'))$ . A conditional exponential

family is stated as follows:

$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \exp(\phi(\mathbf{x}, y)^T \boldsymbol{\theta} - g(\boldsymbol{\theta}|\mathbf{x})) \quad \text{with}$$

$$g(\boldsymbol{\theta}|\mathbf{x}) = \log \sum_{y \in \mathcal{Y}} \exp(\phi(\mathbf{x}, y)^T \boldsymbol{\theta})$$

where  $g(\boldsymbol{\theta}|\mathbf{x})$  is a log-partition function and  $\boldsymbol{\theta} \in \mathbb{R}^d$  is the parameter of the distribution. Under the assumption that  $\{(\mathbf{x}_i, y_i)_{i=1}^n\}$  are drawn independently and identically distributed by the distribution  $p(\mathbf{x}, y)$ , one usually optimizes for  $\boldsymbol{\theta}$  by minimizing the regularized negative conditional log-likelihood:

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} \left\{ \sum_{i=1}^n [g(\boldsymbol{\theta}|\mathbf{x}_i)] - n \boldsymbol{\mu}_{XY}^T \boldsymbol{\theta} + \lambda \|\boldsymbol{\theta}\|^2 \right\},$$

where  $\boldsymbol{\mu}_{XY} := \frac{1}{n} \sum_{i=1}^n \phi(\mathbf{x}_i, y_i)$ . Unfortunately, we cannot compute this quantity directly, as the labels are unknown, however in [3] the authors present *MeanMap*, which makes use of the empirical means of the bags to approximate the expectations with respect to the bag distribution. This approach enjoys uniform convergence under the following (strong) assumption: conditioned on its label, a point is independent of its bag assignment  $i$ , namely,  $p(\mathbf{x}|y, i) = p(\mathbf{x}|y)$ . Extensions of MeanMap can be found in [12, 13].

### 2.2. Maximum Margin Approaches

The maximum margin principle has been widely used in both supervised and semi-supervised learning [14]. In [11] it was also introduced to the unsupervised setting under the name Maximum Margin Clustering (MMC).

**$\alpha$ SVM** In [9] the authors present  $\alpha$ SVM, based on MMC with an extra loss function depending on the provided and estimated bag proportions. In MMC, one jointly optimizes over the labels and the separating hyperplane, with the objective of maximizing the margin between the classes, while in  $\alpha$ SVM, an additional constraint on the bag proportions is optimized over. The authors propose two approaches to solve the problem. The first one is of alternating nature, repeating the steps of solving the usual SVM problem and then arranging the points so that a loss on the bag proportions is minimized. Informally, the labels are arranged in such a way, such that, had an SVM been trained on the (labeled) data, it would achieve a maximum margin solution. The second approach is more convoluted and its presentation is omitted due to space.

**Inverse Calibration** In [10] the authors follow the maximum margin principle by developing a model based on the Support Vector Regression. Inverse Calibration (InvCal) replaces the actual dataset with *super-instances* [4], one for each bag, with soft-labels corresponding to their bag-proportions. The proportion of each bag is modeled as  $q_k =$

$(1 + \exp(-\mathbf{w}^T \mathbf{m}_k + b))^{-1}$ , where  $\mathbf{m}_k = \frac{1}{|\mathcal{B}_k|} \sum_{\mathbf{x}_i \in \mathcal{B}_k} \phi(\mathbf{x}_i)$ . The constraints of the objective try to enforce  $q_k \approx \pi_k$  for all bags [4]. As noted by [4]  $q_k$  fails to be a good measure of bag proportions when the data has high variance, or when the distribution of the data depends on the bags, and the mean is not an adequate statistic.

### 3. LABEL PROPAGATION FOR LEARNING WITH LABEL PROPORTIONS (LP-LLP)

Label Propagation is a graph-based approach to semi-supervised learning that represents the data samples as nodes on a graph and models the relationships through connecting edges with associated weights [6]. In this section we will first present LP in the semisupervised setting following [6, 15] and then adopt the procedure to LLP. We then provide a comparison with the  $k$ -Nearest Neighbours approach.

#### 3.1. Label Propagation

Let  $\mathbf{F}, \mathbf{Y} \in \mathbb{R}^{n \times c}$ , where  $c$  is the number of classes.  $\mathbf{F}$  corresponds to a ‘soft’ label matrix with each row corresponding to one point, and each element of that row corresponds to the probability of assigning the particular data sample to its class.  $\mathbf{Y}$  encodes the true labels with  $Y_{ij} = 1$  if  $y_i = j$  and 0 otherwise. Our method is presented in Algorithm 1.

---

#### Algorithm 1: Label Propagation

---

- 1 Compute a similarity matrix  $\mathbf{W}$  for the whole dataset. Restrict the diagonal to  $W_{ii} = 0$ .
  - 2 Let  $\mathbf{D}$  be a diagonal matrix with  $D_{ii}$  being the sum of the  $i$ -th row of  $\mathbf{W}$ . Compute  $\mathbf{S} = \mathbf{D}^{-1} \mathbf{W}$ .
  - 3 Let  $0 < \alpha < 1$  and iterate  
 $F(t+1) = \alpha \mathbf{S} F(t) + (1 - \alpha) \mathbf{Y}$ . ( $F(0) = \mathbf{Y}$ )
  - 4 Let  $F^*$  denote the limit of  $F(t)$ . Assign labels based on  
 $y_i = \arg \max_j F_{ij}^*$ .
- 

The first step assigns weights to the edges based on a similarity function. A popular choice is  $W_{ij} = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$ , with  $\gamma > 0$ . The second step normalizes the similarity matrix such that each row sums to 1. In the third step  $\alpha$  can be thought of as controlling how the information coming from your neighbors and the information coming from the labeled examples are weighted. In the final step labels are assigned.

It can be shown [15] that the sequence converges to  $F^* = (1 - \alpha)(\mathbf{I} - \alpha \mathbf{S})^{-1} \mathbf{Y}$ .

In the following, without loss of generality, we restrict the problem to the binary case for simplicity. We will now use  $\mathbf{f} \in [-1, 1]^n$  to denote the predictions. An individual update is as follows:

$$f_i^{t+1} = \alpha \sum_j \bar{w}_{ij} f_j^t + (1 - \alpha) y_i, \quad (1)$$

where  $\bar{w}_{ij} = w_{ij} / \sum_j w_{ij}$ .

We now provide a different interpretation through the analogy of a random walk. We have  $\mathbf{f}^* = (\mathbf{I} - \alpha \mathbf{S})^{-1} \mathbf{y}$ , which as we have already seen, can be formulated as  $\mathbf{f}^* = \sum_{i=1}^{\infty} (\alpha \mathbf{S})^i \mathbf{y}$ . Think of  $\mathbf{S}$  as a stochastic matrix and consider a walker sitting on the  $i$ -th node about to decide where to move next based on the  $i$ -th row of  $\mathbf{S}$ . What is the probability that on the *next step* the walker lands on a positive node? These (unnormalized) probabilities can be computed by raising  $\mathbf{S}$  to the corresponding power. Imagine keeping track of the count of positive and negative visits, and using them to provide an estimate for the label of the  $i$ -th node. But of course, earlier visits are more important as they have a higher probability of being close to  $i$ . This weighting is given by the value of  $\alpha$ .

#### 3.2. Label Propagation with Label Proportions

In the label proportions setting *no label* is provided for any of the points. Using the walker analogy, after the first step, the walker has landed on node  $j$ , where  $\mathbf{x}_j \in \mathcal{B}_k$  (to avoid confusion, we imply that sample  $\mathbf{x}_j$  is assigned to the  $k$ -th bag. However, in this case, we do not have access to the true label of  $\mathbf{x}_j$ ). What we exploit in this paper is the idea that even if we do not actually know the true label of  $j$ , we have access to its bag’s label proportions, which we cast as prior probability of being assigned to a class. The intuition of a random walk in semisupervised learning carries on, only now, we have replaced the ‘hard’ labels with ‘soft’ labels, that loosely represent probabilities. In the binary classification case, our  $\hat{\mathbf{y}}$  can now be defined as  $\hat{y}_i = \pi_{k,1}$  where  $x_i \in \mathcal{B}_k$  and  $\pi_{k,1}$  represents the proportion of positive labels (1) in bag  $k$  ( $\pi_{k,0} = 1 - \pi_{k,1}$  is equivalently the proportion of negative labels (0) in the same bag). One could now compute  $\mathbf{f}^* = (\mathbf{I} - \alpha \mathbf{S})^{-1} \hat{\mathbf{y}}$ , but trivial decision making on  $\hat{\mathbf{y}}$  does not guarantee preservation of the class proportions.

Let us now have a look at the problem from a regularization perspective [6] as this will allow us to introduce the bag proportions as constraints in a principled manner. Consider the following loss function that we want to minimize.

$$Q(\mathbf{f}) = \sum_{i=1}^n \sum_{j=1}^n s_{ij} (f_i - f_j)^2 + \gamma \sum_{i=1}^n (f_i - y_i)^2$$

where the first term encourages local smoothness while the second penalizes deviation from  $y$ . The balance between the two is controlled through  $\gamma$ . The solution of  $\arg \min_{\mathbf{f}} Q(\mathbf{f})$  gives the same solution as before [6]. In its original form, the problem would be an Integer Program,  $\arg \min_{\mathbf{f} \in \{0,1\}^n} Q(\mathbf{f})$ , which is in general intractable. Building on this, one could enforce bag constraints through a system of linear equations  $\mathbf{A} \mathbf{f} = \mathbf{b}$ , where  $\mathbf{A} \in \mathbb{R}^{K \times n}$ , with  $K$  being the number of bags and  $\mathbf{b} \in \mathbb{R}^K$ .  $\mathbf{A}$  is defined as  $A_{ji} = 1$ , if  $x_i \in \mathcal{B}_j$  and 0 otherwise, and  $\mathbf{b}_j = n_{j,1}$ , where  $n_{k,c}$  corresponds to the number of instances of class  $c$  in  $\mathcal{B}_k$ .

We now have a constrained Integer Program and proceed by relaxing it to a constrained Linear Program as follows. Instead of controlling the exact number of points for each class for each bag, we are instead controlling the total ‘mass’ of each class in each bag. With  $\mathbf{f}$  unconstrained, any  $f_i$  can dominate over the others, rendering the ‘mass’ conservation principle useless. A more suitable constraint would then be  $\mathbf{f} \in [0, 1]^n$  instead, giving our final problem formulation:

$$\begin{aligned} \mathbf{f}^* &= \arg \min_{\mathbf{f} \in [0, 1]^n} Q(\mathbf{f}) \\ \text{s.t. } \mathbf{A}\mathbf{f} &= \mathbf{b} \end{aligned} \quad (2)$$

### Label Propagation for Learning with Label Proportions

The proposed algorithm solves Eq. 2 in two steps, by first solving the unconstrained problem  $\mathbf{f} = (\mathbf{I} - \alpha\mathbf{S})^{-1}\hat{\mathbf{y}}$  (with  $\hat{y}_i = \pi_{k,1}$  where  $x_i \in \mathbf{B}_k$ ), and then applying *Alternating Projections* [16] to determine  $\mathbf{f}^*$  such that  $\mathbf{A}\mathbf{f}^* = \mathbf{b}$ . Alternating Projections is a simple approach for finding a point in the intersection of convex sets. In our case the sets correspond to row and column sums, as each row of  $\mathbf{F}$  should live on a probability simplex and column comes from the bag proportions, respectively. We refer the reader to [16] and the references therein for convergence guarantees.

These two steps are repeated until convergence, with  $\mathbf{f}^{(t+1)} = (\mathbf{I} - \alpha\mathbf{S})^{-1}\mathbf{f}^{(t)}$ , and  $\mathbf{f}^{(0)} = \hat{\mathbf{y}}$ .

The procedure for solving the optimization problem in Eq. 2 is depicted in Alg. 2.

---

**Algorithm 2:** Label Propagation for Learning with Label Proportions (binary case)

---

**Input :** Bag assignment matrix  $\mathbf{A}$  and vector  $\mathbf{b}$  with  $\mathbf{b}_j = \mathbf{n}_{j,1}$

**Output:** Estimated labels  $\hat{\mathbf{f}}$

- 1 Compute similarity matrix  $\mathbf{W}$  and then  $\mathbf{S}$
  - 2 Compute  $\mathbf{f}^{(t+1)} = (\mathbf{I} - \alpha\mathbf{S})^{-1}\mathbf{f}^{(t)}$ , where  $\mathbf{f}^{(0)}$  is defined as:  $f_i^{(0)} = \pi_{k,1}$  for  $x_i \in \mathbf{B}_k, \forall i$ .
  - 3 Solve for  $\mathbf{f}^{(t+1)*}$  using Alternating Projections.
  - 4 Repeat steps (2) and (3) until convergence.
  - 5 Estimate labels based on  $\hat{f}_i = \text{sgn}(f_i^* - 0.5)$
- 

**Weighted k-Nearest Neighbors (k-NN)** We now analyze the connection between LP-LLP and the weighted versions of k-NN. Referring back to the LP formulation we showed that

$$\mathbf{f}^* = (\mathbf{I} - \alpha\mathbf{S})^{-1}\mathbf{y} = \sum_{k=0}^{\infty} (\alpha\mathbf{S})^k \mathbf{y} = \mathbf{y} + (\alpha\mathbf{S})\mathbf{y} + (\alpha\mathbf{S})^2\mathbf{y} + \dots$$

For one sample, and considering already normalized weights to ease the notation, we have:

$$f_i = y_i + \alpha \sum_{j \in \mathcal{N}(i)} \bar{w}_{ij} y_j + \alpha^2 \sum_{k \in \mathcal{N}(i) \cap \mathcal{N}(j)} \bar{w}_{ik} \bar{w}_{kj} y_j + \dots$$

$\mathcal{N}(i)$  refers to the neighborhood of  $i$ , that is  $w_{ij} \neq 0$ .

$$f_i = \sum_{j \in \mathcal{N}(i)} y_j \left[ \alpha \bar{w}_{ij} + \alpha^2 \sum_{h \in \mathcal{N}(i) \cap \mathcal{N}(j)} \bar{w}_{ih} \bar{w}_{hj} + \dots \right]$$

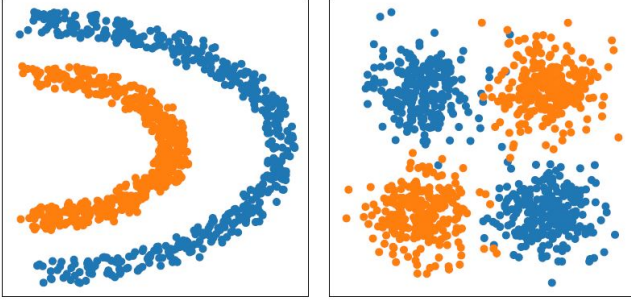
In a weighted k-NN setting, the updates would be of the form  $\hat{f}_i = \sum_{j \in \mathcal{N}(i)} \bar{w}_{ij} y_j$ . Ignoring  $\alpha$  for now, we see that the first term of the LP update is equivalent to the k-NN update. This term encourages nearby points to have the same label (*local smoothness*). Looking at the second term of the LP updates we see that it is an iteration over common neighbors of nodes  $i$  and  $j$  (node  $i$  is the node we wish to update, while node  $j$  is one of the neighbors we will be ‘influenced’ by). If nodes  $i$  and  $j$  have close common neighbors, then this will enhance the ‘influence’ of  $j$  on  $i$ . We can understand this as an exploitation of the *global structure* of our data, points lying on the same structure, should have the same label. These two attributes, local smoothness and global structure, are the underlying principles of many semisupervised learning algorithms, and LP-LLP brings them to the Learning with Label Proportions setting.

**Time Complexity** The algorithm requires the computation of a similarity matrix which would require  $\mathcal{O}(N^2)$ , where  $N$  is the number of data points, and then compute the generalized Laplacian. The bottleneck is computing its inverse which has complexity  $\mathcal{O}(N^3)$ . Similarly to other well-established machine learning algorithms which share this bottleneck, one could make use of approximations that would trade off accuracy for computational expenses [17]. We also note that the per iteration complexity scales linearly in  $N$ , due to the normalization step.

### 3.3. Comparison of Methods

A critical assumption of MeanMap is the conditional independence of a data sample and its bag assignment, given its label. Moreover, the practitioner has no control over the estimated bag proportions. Some of the merits of this approach include probabilistic labels, principled handling of multiclass classification problems and a straightforward implementation. Moreover, the techniques that rely on the max-margin principle lose probabilistic labels but these are not limited to the dependence of data generation to bags. Unlike,  $\alpha$ -SVM, InvCal cannot handle multiclass problems naturally and does not offer a direct control over bag proportions. It is, however, easy and cheap to implement, whereas  $\alpha$ -SVM is more convoluted.

Our proposed graph-based method does not make assumptions on the bag generation process and can seamlessly handle multiclass classification tasks. It does not offer probabilistic labels, but instead it provides constrained values that can be interpreted as ranks. These are then used to provide a degree of control over the estimated bag-proportions.



**Fig. 2.** The two synthetic datasets considered in this paper. From left to right, Half Kernel and XOR.

#### 4. EXPERIMENTS

In our experiments we consider synthetic and real-world datasets. We first describe both and then present the results.

**Synthetic Datasets** We have two different synthetic datasets shown in Fig. 2. The XOR dataset was created by generating four Gaussian distributions with means lying on the corners of a square with size size of 10 and identity covariance matrix. The Half-Kernel dataset was generated by a weighted combination of trigonometric functions,  $\sin$  and  $\cos$ , and added noise.

**Real-World Dataset** In the Sphere Challenge dataset [18], sensor data (including accelerometer, video, and environmental) was collected from 10 people on two separate occasions. There were 8 males and 2 females, with 8 between the ages of 18 to 29 and 2 within the ages of 30 to 39. Each participant was wearing a wrist-worn accelerometer and was asked to complete a series of scripted activities, taking around 25 to 30 minutes in total. The participants performed several activities included that are categorized into ambulation activities (e.g. walking), posture activities (e.g. standing) and transitional activities (e.g. sit to stand). This script was carried out twice in full by each participant on different days, and each was annotated at least twice, producing labels which are themselves proportions. Our experiments consider the task of classifying between ambulatory and sedentary activities. We consider two settings of this dataset. Setting 1 involves predicting ADLs from the raw sensor data, while Setting 2 involves predicting ADLs from the simple baseline features of the dataset, as described in [18].

**Model Setup** We focus only on binary classification problems in this work since this is the setting of our baseline methods. The data samples into three bags. In the case of synthetic data, the data is first generated according to the desired total size and then separated into the bags, respecting the desired bag proportions. We consider that test data is provided in the

format of a bag with known bag proportion. The approach we take is to train our models *with* the test data included, ignoring the true label of each instance. For the experiments we fix parameter  $\alpha$  to 0.50 and choose  $\gamma$  by running the algorithm on a grid of values and then choosing based on the heuristic of highest score in terms of  $\bar{f}^T S \bar{f}$ . We compare LP-LLP against  $\alpha$ SVM and InvCal. MeanMap is not included experiments mainly because its application is not universal, due to its assumptions, and secondly because it has been shown to perform slightly worse than other approaches[9, 10]. Each data-point corresponds to features extracted (such as, mean, min, max etc) from accelerometer data from a small time-window. The data-points were annotated so true labels exist. We ignore the time aspect of the data and separate the data-points in bags at random such that each bag has a pre-specified proportion of each class.

**Discussion** Tables 1, 2, 3 and 4 present our experimental results. The number of experiments performed per dataset was 25. These should be read as follows. The first column describes the format of the dataset, for example, '100A' corresponds to a dataset consisting of a training set of size 100 and a test set of size 20% of that. The letter denotes the bag proportion configuration (recall we have separated our data into 3 bags),  $A - (0.60, 0.40, 0.50)$  and  $B - (0.85, 0.25, 0.40)$ . The rest of the values should be read as *mean(one standard deviation)*. Finally, the model that performs best is emboldened in the results tables. (We would like to point out that for Table 3,  $\alpha$ -SVM did not converge.) As seen empirically our proposed method, in general, compares favorably with alternative approaches over the broad set of experimental configurations considered. With regards to synthetic data, we see that even if  $\alpha$ SVM has a low predictive mean accuracy, it has a high standard deviation. This is due to the method confusing the two classes. In general the performance of all three methods improves with increasingly large datasets, as expected. We note that the Sphere2 configuration of the dataset has the most varied performance (Table 4). We believe that this is a more challenging representation for all models (due the its higher dimensionality) and this is illustrated by higher predictive variance on all models.

#### 5. CONCLUSION

We have introduced a new approach to the problem of learning with label proportions based on a graph-based representation of the data. Our approach encourages local smoothness and exploits global structure to assign labels. This is justified both qualitatively and empirically. Experiments carried out demonstrate the performance of our approach as compared to existing algorithms. Our method is not constrained by the generation of the bags, as opposed to MeanMap. By taking a transductive approach to training we tend to overfit less, which is not always the case for  $\alpha$ SVM and InvCal.

**Table 1. XOR**

	$\infty$ SVM	InvCal	LP-LLP
120A	0.54(0.19)	0.81(0.08)	<b>0.91(0.14)</b>
120B	0.66(0.46)	0.73(0.05)	<b>0.97(0.04)</b>
180A	0.50(0.04)	0.83(0.08)	<b>0.88(0.18)</b>
180B	0.42(0.09)	0.73(0.06)	<b>0.97(0.03)</b>
300A	0.49(0.06)	0.88(0.13)	<b>0.92(0.12)</b>
300B	0.56(0.07)	0.73(0.04)	<b>1.00(0.01)</b>
600A	0.45(0.43)	0.94(0.04)	<b>0.99(0.01)</b>
600B	0.67(0.46)	0.69(0.04)	<b>0.99(0.01)</b>

**Table 2. Half-Kernel**

	$\infty$ SVM	InvCal	LP-LLP
120A	<b>0.81(0.08)</b>	0.46(0.26)	0.58(0.16)
120B	0.73(0.05)	0.56(0.31)	<b>0.74(0.18)</b>
180A	<b>0.83(0.08)</b>	0.63(0.06)	0.59(0.13)
180B	0.73(0.06)	0.50(0.50)	<b>0.93(0.13)</b>
300A	<b>0.88(0.08)</b>	0.67(0.08)	0.84(0.19)
300B	0.73(0.04)	0.50(0.50)	<b>0.96(0.08)</b>
600A	0.94(0.04)	0.70(0.08)	<b>0.96(0.05)</b>
600B	0.69(0.04)	0.74(0.05)	<b>1.00(0.00)</b>

**Table 3. Sphere 1**

	$\infty$ SVM	InvCal	LP-LLP
60A	0.50(0.00)	0.58(0.29)	<b>0.76(0.39)</b>
60B	0.50(0.00)	0.50(0.00)	<b>0.88(0.29)</b>
120A	0.50(0.00)	0.54(0.28)	<b>0.59(0.45)</b>
120B	0.50(0.00)	0.50(0.00)	<b>1.00(0.00)</b>
180A	0.52(0.05)	0.56(0.26)	<b>0.80(0.26)</b>
180B	0.50(0.00)	0.50(0.00)	<b>1.00(0.01)</b>
240A	0.50(0.00)	0.62(0.19)	<b>0.78(0.16)</b>
240B	0.50(0.00)	0.50(0.00)	<b>0.98(0.01)</b>

**Table 4. Sphere 2**

	$\infty$ SVM	InvCal	LP-LLP
60A	0.48(0.03)	<b>0.54(0.09)</b>	0.53(0.09)
60B	0.47(0.04)	<b>0.54(0.07)</b>	0.50(0.09)
120A	0.47(0.11)	0.47(0.12)	<b>0.50(0.13)</b>
120B	0.44(0.09)	0.57(0.10)	<b>0.58(0.12)</b>
180A	0.60(0.11)	<b>0.65(0.14)</b>	0.60(0.16)
180B	0.51(0.10)	0.57(0.13)	<b>0.65(0.16)</b>
240A	<b>0.68(0.16)</b>	0.58(0.19)	0.56(0.20)
240B	0.34(0.19)	0.54(0.10)	<b>0.74(0.12)</b>

## 6. REFERENCES

- [1] Nagarajan Natarajan et al., “Learning with noisy labels,” in *NIPS* 26, 2013, pp. 1196–1204.
- [2] Timothée Cour et al., “Learning from partial labels,” *JMLR*, vol. 12, pp. 1501–1536, 2011.
- [3] Novi Quadrianto et al., “Estimating labels from label proportions,” *JMLR*, vol. 10, pp. 2349–2374, 2009.
- [4] Felix X Yu et al., “On learning from label proportions,” *arXiv preprint arXiv:1402.5902*, 2014.
- [5] Przemyslaw R. Woznowski et al., “SPHERE: A sensor platform for healthcare in a residential environment,” in *Designing, Developing, and Facilitating Smart Cities*, pp. 315–333. Springer, 2017.
- [6] Fei Wang and Changshui Zhang, “Label propagation through linear neighborhoods,” *TKDE*, vol. 20, no. 1, pp. 55–67, 2008.
- [7] Thomas G Dietterich et al., “Solving the multiple instance problem with axis-parallel rectangles,” *Artificial intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.
- [8] Hendrik Kuck and Nando de Freitas, “Learning about individuals from group statistics,” *arXiv preprint arXiv:1207.1393*, 2012.
- [9] Felix X Yu et al., “ $\infty$  svm for learning with label proportions,” *arXiv preprint arXiv:1306.0886*, 2013.
- [10] Stefan Rueding, “SVM classifier estimation from group probabilities,” in *ICML*, 2010, pp. 911–918.
- [11] Linli Xu et al., “Maximum margin clustering,” in *NIPS*, 2005, pp. 1537–1544.
- [12] Giorgio Patrini et al., “Loss factorization, weakly supervised learning and label noise robustness,” in *ICML*, 2016.
- [13] Rafael Poyiadzi et al., “Ordinal label proportions,” in *ECML-PKDD*, 2018.
- [14] Nello Cristianini and John Shawe-Taylor, *An introduction to support vector machines*, CUP, 2000.
- [15] Denny Zhou et al., “Learning with local and global consistency,” in *NIPS*, 2004, pp. 321–328.
- [16] Stephen Boyd and Jon Dattorro, “Alternating projections,” 2003.
- [17] N Kishore Kumar and Jan Schneider, “Literature survey on low rank approximation of matrices,” *Linear and Multilinear Algebra*, vol. 65, no. 11, pp. 2212–2244, 2017.
- [18] Niall Twomey et al., “The SPHERE challenge: Activity recognition with multimodal sensor data,” *arXiv preprint arXiv:1603.00797*, 2016.