2011 11th IEEE International Conference on Data Mining Workshops

Boosting Unsupervised Additive Clustering Using Cluster-Wise Optimization and Multi-Label Learning

Stephen L. France
Lubar School of Business
University of Wisconsin-Milwaukee
Milwaukee, WI
e-mail: france@uwm.edu

Ahmed Abbasi
McIntire School of Commerce
University of Virginia
Charlottesville, VA
e-mail: abbasi@comm.virginia.edu

Abstract— **Additive or overlapping clustering is a technique that is used to analyze overlapping cluster structure in data. In this paper, we motivate the overlapping clustering problem using an example of categorizing movies. We describe the ADCLUS and INDCLUS overlapping clustering models as discrete versions of the CANDECOMP/PARAFAC models. We describe the scalability problems inherent in current overlapping clustering approaches. We give a framework and algorithm for scaling up unsupervised overlapping clustering using a combination of a cluster-wise optimization technique and techniques from multi-label learning. Our framework uses a subset of data to find a training solution and then uses multi-label techniques to find labels for the remaining data.**

Keywords- Additive, clustering, multi-label, unsupervised, supervised

I. INTRODUCTION

In this paper, we describe a methodology for large scale overlapping cluster analysis. In overlapping clustering, each item clustered can be a member of multiple clusters or classes. This is opposed to partitioning clustering, where each item is assigned to exactly one cluster and fuzzy clustering, where each item has a fuzzy membership probability for each cluster. The overlapping clustering techniques described in this paper utilize an additive decomposition of similarity [43]. Thus in the psychology literature, overlapping clustering is often called "additive clustering".

A. Motivation

Consider the problem of classifying N movies into M categories. Each movie can belong to multiple categories. For movie i and category j the category assignment is $p_{ij} = 1$ if the movie i is assigned to category j and $p_{ij} = 0$ if movie i is not assigned to category j. The cluster assignments can be stored in an $N \times M$ matrix $\mathbf{P}$. For an exploratory cluster analysis problem (unsupervised learning), the categories are assigned using some measure of proximity between the movies. The measure of proximity can be derived from behavioral data (e.g., movie attendance), preference data (e.g., movie reviews), or shared features (actors/actresses etc.). The number of categories can be selected arbitrarily or can be decided using some measure of cluster solution fit.

In a supervised problem, the data is split into training and test instances. Each movie has set of features describing the movie. For example, the cinema attendance of a single user could be a feature. The value of this feature for a movie is 1 if the user had seen the movie and 0 if the user has not seen the movie. Given a training set of N_1 movies with M_1 features ($N_1 \times M_1$ matrix $\mathbf{F}_1$) classified into M_2 categories ($N_1 \times M_2$ matrix $\mathbf{P}_1$) and a test set of N_2 movies ($N_2 \times M_1$ matrix $\mathbf{F}_2$), the supervised learning problem is to predict the $N_2 \times M_2$ matrix of categories $\mathbf{P}_2$. The problem is summarized in (1). In data mining terminology this problem is known as the multi-label learning/classification problem [41].

$$\begin{array}{ccc} & M_1 & M_2 \\ N_1 & \mathbf{F}_1 & \mathbf{P}_1 \\ N_2 & \mathbf{F}_2 & \mathbf{P}_2 = ? \end{array} \quad (1)$$

In this paper we concentrate on the unsupervised learning problem, but utilize supervised learning techniques to help scale-up the unsupervised problem. The rationale behind the unsupervised problem is to help discover, explore, and visualize categories using patterns in the underlying perceptual or behavioral data. There is no one correct categorization/clustering of set of data. Categories can be based on subjective knowledge and can be determined by individuals, institutions, or by cultural norms [22]. Returning to the movie example, a user may categorize movies based upon viewing preferences and an online retailer such as Amazon or Netflix may categorize movies to help consumers navigate their website and find movies. An unsupervised learning technique, such as overlapping clustering, can be used to help explore possible categorization schemes and also to test how subjective categorization schemes correspond to variance in the underlying data.

978-0-7695-4409-0/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDMW.2011.40

II. TECHNICAL DETAILS

A. Overlapping Clustering

Overlapping clustering techniques have long been used to analyze similarity of data in psychological experiments, but have been generally implemented on small ($n < 100$) datasets. In this section, we describe the basic overlapping clustering models and methods for fitting these models. A measure of similarity s_{ij} between two items i and j can be modeled as a weighted sum of overlapping cluster assignments. The basic model [36] is referred to as the ADCLUS model and is given in (2).

$$\hat{s}_{ij} = \sum_{m=1}^{M} w_m p_{im} p_{jm} + c \tag{2}$$

, where $\hat{s}_{ij}$ is the reconstructed proximity between item i and item j, M is the number of clusters in the solution, c is a fitting constant, and p_{im} and p_{jm} are binary cluster membership indicators, i.e., if item i is in cluster m then p_{im} is 1, otherwise p_{im} is 0. The variance accounted for (VAF) by the model is given in (3).

$$VAF = \frac{\sum_i \sum_{j>i} \left(s_{ij} - \hat{s}_{ij}\right)^2}{\sum_i \sum_{j>i} \left(s_{ij} - \bar{s}_{ij}\right)^2} \tag{3}$$

, where $\bar{s}$ is the average value of s_{ij}. An individual differences version of the model called INDCLUS [6] is given in (4). INDCLUS can model multiple similarity matrices, each representing an item or group of items. There is a single cluster structure, but each individual or group of individuals has a separate set of cluster weights. Groups of individuals can be defined by splitting the dataset using categorical variables [10]. Multi-way INDCLUS [8] can be thought of a categorical version of the CANDECOMP/PARAFAC model [7,24]. A further generation of INDCLUS that allows for additional weighting schemes is given in [15].

$$\hat{s}_{kij} = \sum_{m=1}^{M} w_{km} p_{im} p_{jm} + c_k \tag{4}$$

, where k denotes the subject, w_{km} denotes the weight for subject k on cluster m, and c_k denotes the fitting constant for subject k. The VAF for the INDCLUS model is the average VAF across all k.

The model can be fit by minimizing a least squares optimization criterion. The optimization model given is NP-Hard [19] and the special case where $k = 1$ (ADCLUS) is also NP-Hard. Heuristic algorithms to fit the INDCLUS model include the original INDCLUS algorithm [6], MAPCLUS [4], SINDCLUS [9], SYMPRES [26], and a tabu search like technique [33]. Maximum-likelihood techniques have been used to fit the $k = 1$ ADCLUS model [31,39]. A maximum likelihood approach with applications to supervised multi-label learning problems is given in [5]. The number of clusters M is usually fixed, but alternatively, a minimum number of clusters and minimum VAF can be specified [29].

The scalability of SINDCLUS, SYMPRES, and heuristic extensions for these algorithms is tested in [19]. The algorithms were run for fixed periods of time on multiple datasets and for $125 < N < 500$. Overall, SYMPRES is the most scalable algorithm for $250 < N < 500$, outperforming both SINDCLUS and tabu search variants of SINDCLUS/SYMPRES. Both SINDCLUS and SYMPRES are cluster-wise (optimize each cluster separately) algorithms. SINDCLUS allows relaxed solutions, with $p_{im}p_{im}$ transformed to $p_{im}q_{im}$, with the possibility that $p_{im} \neq q_{im}$. This relaxes the algorithm solution space and allows SINDCLUS to obtain better solutions for small n, but as n increases the proportion of solutions where $p_{im} \neq q_{im}$ increases. For $N > 2000$, the run time for the SYMPRES algorithm becomes unmanageable and thus the algorithm is not suited to large scale data mining applications. In the next section we detail a methodology for improving the scalability of overlapping clustering. We give a brief description of the SYMPRES algorithm and then describe how supervised multi-label classification techniques can be used to improve scalability.

B. The SYMPRES algorithm

The INDCLUS model (6) can be expressed in matrix form for all i and j.

$$\mathbf{S}_k = \mathbf{P}\mathbf{W}_k\mathbf{P}' + c_k\mathbf{1}\mathbf{1}' + error \quad \forall k = 1..k, \tag{6}$$

, where $\mathbf{S}_k$ contains similarities for subject k, $\mathbf{P}$ is an $N \times M$ matrix of cluster memberships, $\mathbf{W}_k$ is a diagonal cluster weight matrix, and c_k is an additive constant. The INDCLUS equation can be written as (7).

At each iteration of the SYMPRES algorithm, $M - 1$ clusters are taken to be fixed and the values of $\mathbf{p}_m$ and w_{mk} ($\forall k = 1..K$) for a single cluster m are optimized.

$$\begin{aligned} \bar{\mathbf{S}}_k &= \mathbf{p}_m w_{mk} \mathbf{p}_m' + error \\ &= \mathbf{S}_k - \mathbf{P}_{(-m)}\mathbf{W}_{k(-m)}\mathbf{P}_{(-m)}' + c_k\mathbf{1}\mathbf{1}' \quad \forall k = 1..k \end{aligned} \tag{7}$$

$$\min f\left(\mathbf{p}_m\right) = \sum_{k=1}^{K} \left\| \bar{\mathbf{S}}_k - w_{mk}\mathbf{p}_m\mathbf{p}_m' \right\|^2 \tag{8}$$

This function can be optimized as (9).

$$\min f\left(\mathbf{p}_i\right) = \text{constant} + \mathbf{p}_i'\mathbf{A}\mathbf{p}_i,$$

$$\text{, where } \mathbf{A} = \left(\sum_{k=1}^{K} w_{ik}^2 \mathbf{M}_k - 2\sum_{k=1}^{K} w_k \bar{\mathbf{S}}_k \mathbf{M}_k \right) \tag{9}$$

Here $\mathbf{M}_k$ is a weighting matrix for $\bar{\mathbf{S}}_k$. The full update algorithm is given in [26].

C. Boosting Algorithm

The algorithm or framework described in this section can

be used to help scale-up small scale overlapping clustering solutions. The algorithm can be thought of as a "connective algorithm" and it utilizes both existing overlapping clustering optimization algorithms and multi-label learning techniques. The algorithm is independent of the overlapping clustering optimization technique used. In the experimentation section we utilize SYMPRES to find the initial small scale overlapping clustering solution, but any of the algorithms described in section II part C could be used.

The algorithm is designed for situations where the number of items N to be clustered is larger than the number of items that can be handled by existing algorithms. For example, if overlapping clustering technique A can be guaranteed to cluster 1000 items in a reasonable time (say less than 10 minutes) then the boosting algorithm would not be required for a 500 item data set. For a 10,000 item data set, one could run the boosting algorithm with a training size of $N_1 < 1000$.

The algorithm is analogous to several methods for scaling up continuous dimensionality reduction solutions. Nyström methods [32], such as Landmark MDS [16], use a subset of the source data to estimate a transition matrix and then use this matrix to transform the entire dataset. In [1], a variant of large scale MDS is described. A standard MDS algorithm is used to find a lower dimensional solution for a subset of the data and this solution is used to train a neural network. The neural network is used to find a lower dimensional mapping for the remaining points.

The boosting algorithm/framework is given below.

Algorithm 1

Input: T – Maximum training time, K – Number of data matrices, $\mathbf{F}_k$ – Input data matrices (N item × M features) $\forall k = 1..K$, N_1 – Number of training items.

Output: P ($\mathbf{W}_k$ and c_k for all k), VAF

Steps

1. Set time $T = 0$, set VAF_1 to be 0, and start the training clock.
2. Randomly select N_1 training items (select the same items for each k) and denote the $N_1 \times M$ matrices as $\mathbf{F}_k$ ($\forall k = 1..k$).
3. Calculate similarity matrices $\overline{\mathbf{S}}_k$ from the initial data using an appropriate similarity transformation $\overline{\mathbf{S}}_k = f_s(\mathbf{F}_{1k})$ $\forall k = 1..K$. Here f_s can be any valid similarity function (e.g., cosine similarity, correlation, Euclidean similarity etc.).
4. Run the SYMPRES algorithm returning VAF, $\mathbf{P}$, $\mathbf{W}_k$ ($\forall k = 1..K$), and c_k ($\forall k = 1..K$). If ($VAF > VAF_1$) then set $VAF_1 = VAF$, $\mathbf{P}_1 = \mathbf{P}$, $\mathbf{F}_{1k} = \mathbf{F}_k$, $\mathbf{W}_{1k} = \mathbf{W}_k$ ($\forall k = 1..K$) and $c_{1k} = c_k$ ($\forall k = 1..K$) as the best solution.
5. If elapsed training time > T then go to 6 otherwise return to 2.
6. Select the N_2 remaining items not in $\mathbf{F}_{1k}$ as $\mathbf{F}_{2k}$ (for $k = 1\ldots K$).
7. Find $\mathbf{P}_2$ by solving the supervised multi-label learning problem given in (1).
8. Combine $\mathbf{P}_1$ and $\mathbf{P}_2$ to find the overall clustering solution (10).

$$\mathbf{S}_k = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \end{bmatrix} \mathbf{W}_k \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \end{bmatrix}' + c_k \mathbf{1}\mathbf{1}' + error, \forall k = 1..k \qquad (10)$$

9. Run a partial version of SYMPRES (or SINDCLUS). For each cluster m and data matrix k, solve $\min f(\mathbf{p}_m) = \left\| \overline{\mathbf{S}}_k - w_{mk} \mathbf{p}_m \mathbf{p}_m' \right\|^2$, but only alter the weights w_{ik}. Keep $\mathbf{p}_m$ fixed. The additive constant can be modeled as the weight for an additional universal cluster with all cluster memberships equal to 1.
10. Calculate the total VAF from (10) and denote this value as VAF_T. The success of the scaling-up procedure (SS) is calculated as the percentage of VAF retained by the new solution (11).

$$SS = \left[1 - \frac{VAF_1 - VAF_T}{VAF_1} \right] \times 100\% \qquad (11)$$

III. EXPERIMENTATION

A. Experimental Design

We tested the algorithm on a set of real world data sets. The data sets tested are summarized in Table 1.

TABLE I

Name	DESCRIPTION	Rows	COLS	Ref
Adult	Person level data from the 2000 census.	48842	14	[28]
IUsage	Demographics of internet users.	10104	72	[13]
Magic	Data on high energy gamma particles	19020	11	[3]
Mushroom	Physical characteristics of mushrooms.	8124	22	[35]
Parkinson	Parkinson's disease telemonitoring data.	5875	26	[40]
Pen Digits	Pixel information on hand written digits.	10992	16	[2]
Spambase	Email spam characteristics	4601	57	[25]
Wine Quality	A range of quality indicators for both red and white wine.	6497	12	[14]

We tested the algorithm using training random samples of sizes N_1 = 500, 1000, and 2000. For each dataset $\mathbf{F}$, we created 5 samples $\mathbf{F}_1$ for each sample size. Each sample $\mathbf{F}_1$ gave an $N_1 \times M_1$ matrix. The remaining $N - N_1$ items were placed in $\mathbf{F}_2$. We then created a similarity matrix $\mathbf{S}_1$ for each $\mathbf{F}_1$. For discrete data, we calculated distances using the Hamming metric. For continuous data we used the cosine distance metric. The cosine metric is robust and gives good results on a range of data [18,37]. For mixed data, we used a weighted combination of the cosine and Hamming metrics. The weights were assigned to equalize variance between dimensions. The similarity matrix $\mathbf{S}_1$ was calculated using $\mathbf{S}_1 = max\{\mathbf{D}_1\} - \mathbf{D}_1$, where $\mathbf{D}_1$ is the distance matrix. We then ran the basic SYMPRES algorithm to gain a clustering $\mathbf{P}_1$ and weights $\mathbf{W}_1$ for each $\mathbf{S}$ and thus each $\mathbf{F}_1$. We used SYMPRES rather than SINDCLUS to guarantee symmetric solutions, but in real life applications any suitable overlapping clustering algorithm could be used. We then learned the remaining clusters $\mathbf{P}_2$ using the algorithm described in section II part *C* and each of the multi-label learning techniques summarized in Table 2.

TABLE 2
MULTI-LABEL LEARNING TECHNIQUES

Name	DESCRIPTION	Ref
KNN	Utilizes a *k*-nearest neighbors approach and Bayesian prior to estimate the posterior using the maximum a posteriori principle.	[48]
ML-BP	Uses a backpropogation neural network to predict labels. The neural network trained using a set of labeled training data.	[47]
ML-RBF	Uses cluster analysis to find centroids of each class. Then uses these base 'centroid' vectors and a radial basis function methods to train a radial basis function (RBF) neural network.	[45]
NBayes	Uses PCA and feature selection to preprocess the data and then uses Naïve Bayes to predict class labels.	[46]
SVM	Decomposes multi-label learning problem into multiple two-class problems and using an optimization function for minimizing Hamming loss to find an overall labeling.	[17]

As the problem is unsupervised, we report the quality of the solution with respect to the solution criterion by calculating $\mathbf{S}_2$ for $\mathbf{F}_2$ and then calculating the VAF from the ADCLUS model using (2) and (3). We give both the VAF for the training data and the VAF for the overall solution. We also report the difference between these values. For example, consider the Magic data set and a sample size of 1000. Let VAF_1 be the VAF calculated using $\mathbf{P}_1$, $\mathbf{S}_1$, $\mathbf{W}_1$, $\mathbf{F}_1$, and c for the 1000 item SINDCLUS training solution. Let VAF_2 be the overall VAF for the 19020 item overall solution calculated using $\mathbf{P}$, $\mathbf{S}$, $\mathbf{W}$, $\mathbf{F}$, and c. Let $VAF_{diff} = VAF_1 - VAF_2$. A good solution is one in which the boosted solution has a similar VAF to the original training solution, but with many more items. To put the value of VAF into context, for each data set we created 100 random clusterings. For each clustering we performed step 9 of the algorithm to optimize the weights (and thus VAF) for the model with respect to the random clustering. We then calculated the VAF of the model. We denote the average of these values as the dataset's "base" VAF.

We ran five replications for each combination of parameters. The parameters used are summarized in Table 3.

TABLE 3
EXPERIMENTAL PARAMETER SUMMARY

Parameter	DESCRIPTION	No of values
Data set	The data sets described in Table 1.	8
Sample size	The size of the training sample taken from the data: 500, 1000, or 2000.	3
No. Clusters	The number of clusters to be taken: 4, 8, or 16.	3
ML-Method	The multi-label techniques to be tested.	5

In total, there are 8 × 3 × 3 × 5 = 240 experimental conditions, with 5 replications for each experimental condition. We ran the experiments using Matlab on Dell Poweredge T100, with Windows 7 64 bit, 8 GB of memory and a single Zeon 2.0Ghz CPU. We adapted code from [27] for the SYMPRES algorithm and we used code taken from the associated papers referenced in Table 2 for implementing multi-label learning. Initial parameter testing was run for each of the five multi-label learning algorithms.

B. Results

The results are summarized in Fig. 1−3. In Fig. 1, we give the average VAF for the complete solutions split by sample size and the number of clusters. We also give the average value of VAF for the training solution on the sample data and the "base VAF". We split the experimental runs by sample size and number of clusters. One can see that there is very little difference in VAF between the training solutions and the boosted solutions relative to the base VAF. The ML-kNN and ML-RBF techniques seem to give the best values of VAF, though some runs of the ML-RBF technique failed for the Census data set, so this may bias results slightly.

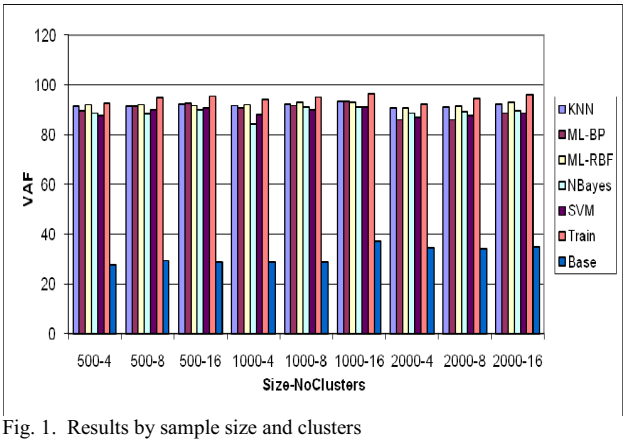

Fig. 1. Results by sample size and clusters

In Fig. 2, the results are summarized by the data set used. All results are complete, except for the Census data set, where some ML-RBF runs failed due to memory constraints. For the Census and IUsage data sets the base VAF is less than 0 as the average sum of squared error (SSE) is greater than the total sum of squares (SST) for **S**. Again, there is good performance from the ML-kNN technique. The technique worked well for all data sets and for one dataset (PenDigits) the ML-kNN technique resulted in an average VAF greater than the training VAF.

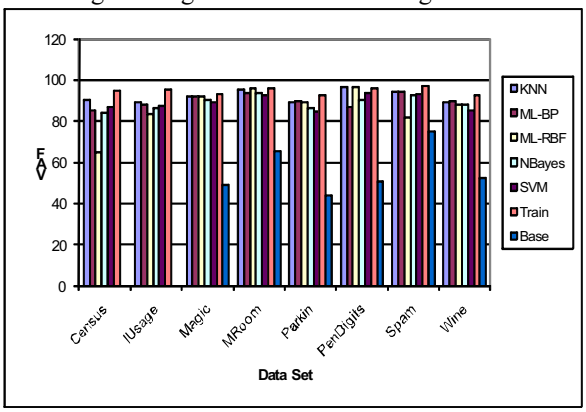

Fig. 2. Results by data set

A box-plot for VAF_{diff} is given in fig. 3. Results are split by the multi-label learning technique and by the size of the training sample. The boxes denote the middle two data quartiles. Observations that lie outside the interquartile range are plotted individually. One can see from the scatterplot that the ML-kNN and ML-RBF techniques give the tightest and most consistent solutions. As the size of the sample training data increases, the number of poor solutions (high value of VAF_{diff}) decreases.

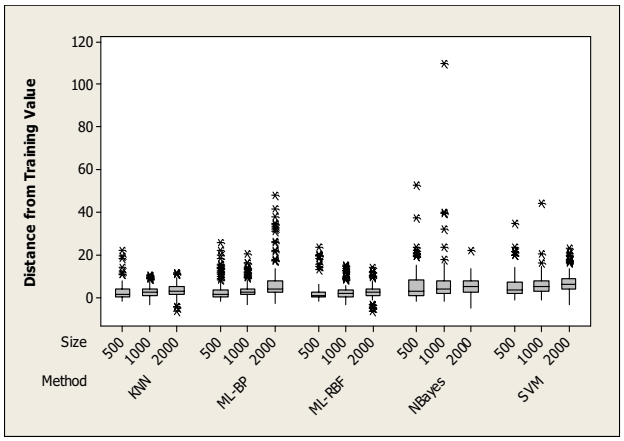

Fig. 3. Box and whisker plot

We analyzed the data using a full-factorial ANOVA model with the sample size, no. clusters, data set (name), and (mult-label learning) method taken as factors. The ANOVA table is given in fig. 4. The size, ML-method, and data set name are significant at the $p < 0.05$ level. The no. clusters factor is not significant. All two factor interactions except for no. clusters × dataset are significant.

Dependent Variable:DistTrain

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	34946.362[a]	352	99.279	4.761	.000
Intercept	28364.050	1	28364.050	1360.280	.000
Size	427.558	2	213.779	10.252	.000
Clusters	102.195	2	51.097	2.451	.087
Method	3065.632	4	766.408	36.755	.000
Name	6278.141	7	896.877	43.012	.000
Size * Clusters	370.234	4	92.558	4.439	.001
Size * Method	1248.233	8	156.029	7.483	.000
Size * Name	2938.531	14	209.895	10.066	.000
Clusters * Method	256.060	8	32.008	1.535	.140
Clusters * Name	2193.755	14	156.697	7.515	.000
Method * Name	3482.044	28	124.359	5.964	.000
Size * Clusters * Method	556.888	16	34.806	1.669	.046
Size * Clusters * Name	1488.876	28	53.174	2.550	.000
Size * Method * Name	5819.694	54	107.772	5.169	.000
Clusters * Method * Name	2216.238	55	40.295	1.932	.000
Size * Clusters * Method * Name	2371.636	108	21.960	1.053	.341
Error	28316.516	1358	20.852		
Total	93774.806	1711			
Corrected Total	63262.878	1710			

a. R Squared = .552 (Adjusted R Squared = .436)

Fig. 4. ANOVA Table: Between subject effects

We performed post-hoc tests using the Scheffé test [34], which was chosen as it is known to give conservative confidence interval bounds. The results are given in Fig. 5. By performance, the multi-label learning techniques can be split into three groups. ML-kNN and ML-RBF have the best performance with an average value of VAF_{diff} approximately 2.5. ML-BP and Naïve Bayes are in a single group and SVM overlaps with Naïve Bayes. Overall ML-RBF and ML-kNN have the best performance, but for ML-RBF, this performance may be biased by scalability problems on the larger census data set.

Scheffe

Method	N	Subset 1	Subset 2	Subset 3
ML-RBF	311	2.4239E0		
KNN	350	2.6512E0		
ML-BP	350		4.5640E0	
NBayes	350		5.6155E0	5.6155E0
SVM	350			5.6592E0
Sig.		.981	.060	1.000

Means for groups in homogeneous subsets are displayed.
Based on observed means.
The error term is Mean Square(Error) = 20.852.

Fig. 5. Results of Scheffe test

IV. EXAMPLE AND VISUALIZATION

Thus far, we have described a framework and an algorithm for scaling up additive clustering and we have run some experimental tests to prove the effectiveness of our approach. But how can additive clustering be used for large scale visualization and exploratory data analysis? For traditional psychological applications on small data sets, for example [36], one can visualize items using principal component analysis (PCA) or multidimensional scaling (MDS) and then hand draw overlapping solutions. Hand drawing solutions is not feasible for larger scale data sets. We present a short, ad-hoc algorithm for drawing overlapping clustering solutions.

Algorithm 2

Input: **F**– Input data matrix ($N \times M_1$), **P** clustering matrix ($N \times M_2$),

Output: ($N \times 3$) vector of RGB values.

Steps

1. Calculate the Hamming difference between each pair of clusters. For N items and clusters $\mathbf{p}_a$ and $\mathbf{p}_b$, the Hamming distance between the clusters is given in (12).

$$d_H(\mathbf{p}_1, \mathbf{p}_2) = \sum_{i=1}^{N} |p_{ia} - p_{ib}| \qquad (12)$$

2. Split the clusters into three groups. The clusters should be assigned to minimize the total within group Hamming distance.
3. Each of the three groups should be assigned to one of the three primary colors, red, green, or blue. If there is only one cluster in the group then it is assigned full saturation (255). If there are $V > 1$ clusters in the group, then cluster j (relative to only the clusters in the group) is given saturation $255 \times (j/V)$. Denote the color for cluster j as an element of the color row vector $\mathbf{RGB}_j = [R \quad G \quad B]$. For example, if the color is red and the saturation is $255/2$ then the cluster vector $\mathbf{RGB}_j = [255/2 \quad 0 \quad 0]$.
4. The items are visualized by using PCA or any other dimensionality reduction technique to transform **F** into 2 or 3 dimensions.
5. Each item has a cluster membership row vector $[\mathbf{p}_{i1} \cdots \mathbf{p}_{ij} \cdots \mathbf{p}_{iM_2}]$ and each cluster has an RGB component $\mathbf{RGB}_j$. Set the initial color vector for each item to be $\mathbf{RGB}_i = [0 \quad 0 \quad 0]$. For each element p_{ij} of $\mathbf{p}_{i\cdot} = [p_{i1} \cdots p_{ij} \cdots p_{iM_2}]$, if $p_{ij} = 1$ then $\mathbf{RGB}_i = \mathbf{RGB}_i + \mathbf{RGB}_j$.

The rationale behind the algorithm is to distribute colors across the RGB spectrum and give similar clusters similar colors. A simple 3 cluster PCA visualization for the wine data is given in fig. 6. Here each cluster is assigned to a color and each color has full saturation. Large pixels are used so that one can fully see the overlap.

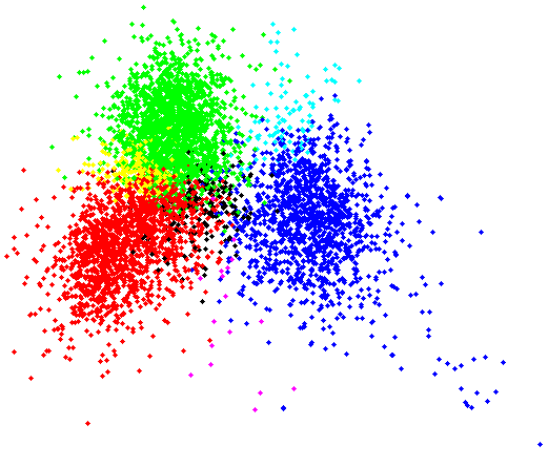

Fig. 6. 3 cluster wine data solution visualization

The overlapping clustering solution corresponds strongly to the PCA solution. There are three distinct clusters. There are small yellow, cyan, and purple regions, which indicate items that belong to 2 clusters. There is a region of black in the center of the visualization that indicates items belonging to all three clusters. Relating the PCA dimensions to the original data, the green cluster and red cluster indicate mostly white wines and the green cluster indicates higher quality wines. There is some overlap for "mid quality" wines. The blue cluster indicates mostly red wines.

V. CONCLUSIONS AND FUTURE WORK

Overlapping or additive clustering is an unsupervised data analysis technique that has been used extensively to examine data structure in the psychometric/psychological fields. However, to fit an additive clustering model requires the solution of an NP Hard optimization problem. Memory and processing requirements for existing algorithms increase rapidly with problem size. Thus, additive

clustering has found limited applications for large scale data analysis. In this paper we have described a framework and algorithm for scaling up additive clustering by using supervised multi-learning techniques to help "boost" solutions gained from unsupervised solutions calculated on subsets of the data. We devise a set of experiments on a range of data sets taken from the data mining literature. We show that our algorithm gives good solutions, with very little loss in VAF. In particular, using the ML-kNN algorithm for multi-label learning [48] gives particularly good and consistent results.

There is much scope for future work. Additional multi-label techniques could be tested, such as those summarized in [41,42]. The ML-kNN algorithm provides a flexible approach to multi-label learning. Work could be done with the ML-kNN algorithm to optimize the algorithm for the task at hand. Variants of weighted *k*-nearest neighbors algorithm could be used.

Further work could be done on the evaluation and analysis of overlapping clustering solutions. A measure of clustering reliability could be used to evaluate the stability of clustering solutions. The Omega index [12] is an overlapping clustering formulation of the Rand index for measuring clustering solution reliability and could be used in this context. The VAF measure does not account for the number of clusters. A pseudo-F test could be used to give a measure of fit adjusted for the number of clusters. Given certain parametric assumptions, the Bayesian information criterion could be used, as described in [30].

At a broader level, the use of additive clustering as an exploratory technique could be investigated with a wide range of data sets. For example, additive clustering could be explored in conjunction with other unsupervised techniques, such as dimensionality reduction and canonical analysis. Additive clustering could be used to help evaluate categorizations/labelings. As discussed in [22], classifications and labeling can be built up over time by a range of actors. For example, a set of multi-label movie categorizations may be based on stereotypes and on ad-hoc decisions. The categorizations may not reflect patterns in movie watching and review data. Additive clustering could be used to help explore labelings and evaluate labelings with respect to the underlying variance in the data.

REFERENCES

[1] D. K. Agrafiotis, D. N. Rassokhin, and V. S. Lobanov, "Multidimensional scaling and visualization of large molecular similarity tables," *Journal of Computational Chemistry*, vol. 22, Apr. 2001, pp. 488–500, doi:10.1002/1096-987X.

[2] F. Alimoglu and E. Alpaydin, "Methods of combining multiple classifiers based on different representations for pen-based handwriting recognition," Proc. Fifth Turkish Artificial Intelligence and Artificial Neural Networks Symp. (TAINN 96), 1996.

[3] R. K. Bock, A. Chilingarian, M. Gaug, F. Hakl, T. Hengstebeck, M. Jirina, J. Klaschka, E. Kotrc, P. Savický, S. Towers, A. Vaiciulis, and W. Wittek, "Methods for multidimensional event classification: a case study using images from a Cherenkov gamma-ray telescope," Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 516, Jan. 2004, pp. 511-528, doi:10.1016/j.nima.2003.08.157.

[4] P. Arabie and J. D. Carroll, "MAPCLUS: A mathematical programming approach to fitting the ADCLUS Model," Psychometrika, vol. 45, Jun. 1980, pp. 211-235, doi: 10.1007/BF02294077.

[5] A. Banerjee, C. Krumpelman, S. Basu, and R. J. Mooney, "Model-Based Overlapping Clustering," Proc. Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-05), ACM Press, Aug. 205, pp. 532-537, 2005.

[6] J. D. Carroll and P. Arabie, "INDCLUS: An individual differences generalization of the ADCLUS Model and the MAPCLUS algorithm," Psychometrika, vol. 48, Jun. 1983, pp. 157-169, doi:10.1007/BF02294012.

[7] J. D. Carroll and J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckert-Young" decomposition," Psychometrika, vol. 35, Sep.1970, pp. 283-319, doi:10.1007/BF02310791.

[8] J. D. Carroll and A. Chaturvedi, "A general approach to clustering and multidimensional scaling of two-way, three-way, and higher-way data," in Geometric Representations of Perceptual Phenomena, R. D. Luce, M. D'Zmura, D. Hoffman, G. J. Iverson, and A. K. Romney, Eds. Mahwah: Lawrence Erlbaum Associates, 1995, pp. 295-318.

[9] A. Chaturvedi and J. D. Carroll, "An alternating combinatorial optimization approach to fitting the INDCLUS and generalized INDCLUS models," Journal of Classification, vol. 11, Sep. 1994, pp. 155-170, doi: 10.1007/BF01195676.

[10] A. Chaturvedi and J. D. Carroll, "Deriving Market Structures Via Additive Decomposition of Market Shares (Application of Three-Way Generalized SINDCLUS)," DIMACS Workshop on Algorithms for Multidimensional Scaling, 2001.

[11] A. Chaturvedi and J. D. Carroll, "CLUSCALE ("CLUstering and Multidimensional SCAL[E]Ing"): A three-way hybrid model incorporating overlapping clustering and multidimensional scaling structure,". Journal of Classification, vol. 23, Sep. 2006, pp. 269-299, doi: 10.1007/s00357-006-0016-0.

[12] L. M. Collins and C. W. Dent, "Omega: A general formulation of the Rand index of cluster recovery suitable for non-disjoint solutions," Multivariate Behavioral Research, vol. 23, Apr. 1988, pp. 231-242, doi:10.1207/s15327906mbr2302_6.

[13] D. Cook, "Internet Usage Data Data Set," [available at http://archive.ics.uci.edu/ml/datasets/Internet+Usage+Data], 1997.

[14] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis, "Modeling wine preferences by data mining from physicochemical properties," Decison Support Systems, vol. 47, Nov. 2009, pp. 547-553, http://dx.doi.org/10.1016/j.dss.2009.05.016

[15] W. S. DeSarbo, "GENNCLUS: New models for general nonhierarchical clustering analysis," Psychometrika, vol. 47, Dec. 1982, pp. 449-475, doi:10.1007/BF02293709.

[16] V. de Silva and J. B. Tenenbaum, "Sparse multidimensional scaling using landmark points," unpublished, 2004.

[17] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in Advances in Neural Information Processing Systems 14, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge: MIT Press, 2002, pp. 681-687.

[18] S. L. France, J. D. Carroll, and H. Xiong, "Distance Metrics for High Dimensional Nearest Neighborhood Recovery: Compression and Normalization," Information Sciences, in press, doi:10.1016/j.ins.2011.07.048.

[19] S. L. France, W. Chen, and Y. Deng. "Additive clustering: Analysis, experimentation, and meta-heuristics," unpublished, 2010.

[20] F. Glover, "Tabu search - part I," *ORSA Journal on Computing*, vol. 1, Sum. 1989, pp. 190-206, 10.1287/ijoc.1.3.190.

[21] F. Glover, "Tabu search - part II," *ORSA Journal on Computing*, vol. 2, Win. 1990, pp. 4-32, doi:10.1287/ijoc.2.1.4.

[22] R. J. Glushko, P. P. Maglio, T. Matlock, and L. W. Barsalou, “Categorization in the wild,” *Trends in Cognitive Sciences*, vol. 12, Apr. 2008, pp. 129-135, doi:10.1016/j.tics.2008.01.007.

[23] M. Harper, "MovieLens Data Set," [available at http://www.grouplens.org/node/73], 2006.

[24] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis," unpublished, 1970.

[25] M. Hopkins, E. Reeber, G. Forman and J. Suermondt, "Spambase Data Set," [available at http://archive.ics.uci.edu/ml/datasets/Spambase], 1999.

[26] H. A. L. Kiers, “A Modification of the SINDCLUS algorithm for fitting the ADCLUS and INDCLUS models,” Journal of Classification, vol. 14, Sep. 1997, pp. 297-310, doi:10.1007/s003579900014.

[27] H. A. L. Kiers, "SINDCLUS and SYMPRES software," [available at http://www.ppsw.rug.nl/~kiers/indclus.zip], 1997.

[28] R. Kohavi and B. Becker, "Adult Data Set," [available at http://archive.ics.uci.edu/ml/datasets/Adult], 1996.

[29] M. D. Lee, “An extraction and regularization approach to additive clustering,” Journal of Classification, vol. 16, Jul. 1999, pp. 255-281, doi:10.1007/s003579900056.

[30] M. D. Lee, “On the Complexity of Additive Clustering Models,” Journal of Mathematical Psychology, vol. 45, Feb. 2001, pp. 131-148, doi:10.1006/jmps.1999.1299.

[31] D. J. Navarro and T. L. Griffiths, "Latent Features in Similarity Judgments: A Nonparametric Bayesian Approach," Neural Computation, vol. 20, Nov. 2008, pp. 2597-2628, doi:10.1162/neco.2008.04-07-504.

[32] J. C. Platt, "FastMap, MetricMap, and Landmark MDS are all Nyström Algorithms," unpublished, 2005.

[33] W. Ruml, Constructing Distributed Representations using Additive Clustering," in Advances in Neural Information Processing Systems 14, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge: MIT Press, 2002, pp. 107-114.

[34] H. Scheffé, The Analysis of Variance. New York, NY: Wiley, 1959.

[35] J. Schlimmer, "Mushroom Data Set," [available at http://archive.ics.uci.edu/ml/datasets/Mushroom], 1987.

[36] R. N. Shepard and P. Arabie. “Additive clustering: representation of similarities as combinations of discrete overlapping properties,” *Psychological Review*, vol. 86, Mar. 1979, pp. 87-123, doi:10.1037/0033-295X.86.2.87.

[37] A. Strehl, J. Ghosh and R. Mooney, "Impact of similarity measures on web-page clustering," Proc. National Conference on Artificial Intelligence: Workshop of Artificial Intelligence for Web Search (AAAI 2000), AAAI Press, Jul. 2000, pp. 58-64.

[38] J. M. F. Ten Berge and H. A. L. Kiers, “A Comparison of two methods for fitting the INDCLUS Model,” *Journal of Classification*, vol. 22, Sep. 2005, pp. 273-286, doi:10.1007/s00357-005-0017-4.

[39] J. B. Tenenbaum, “Learning the structure of similarity,” in Neural Information Processing Systems 8, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds. Cambridge, MA: MIT Press, 1996, pp. 59-65.

[40] A. Tsanas, M. A. Little, P. E. McSharry and L. O. Ramig, "Accurate Telemonitoring of Parkinson's Disease Progression by Noninvasive Speech Tests," IEEE Trans. Biomedical Engineering, vol. 57, Apr. 2010, pp. 884-893, doi:10.1109/TBME.2009.2036000.

[41] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," in Data Mining and Knowledge Discovery Handbook, O. Maimon and L. Rokach, Eds. New York, NY: Springer, 2010, pp. 667-685, doi: 10.1007/978-0-387-09823-4_34.

[42] G. Tsoumakas, I. Katakis, I. Vlahavas, “Random k-labelsets for multi-label classification,” IEEE Trans. Knowledge and Data Engineering, vol. 23, Jul. 2011, pp. 1079-1089, doi:10.1109/TKDE.2010.164.

[43] A. Tversky, "Features of Similarity," *Psychological Review,* vol. 84, Jul. 1977, pp. 327-352, , doi:10.1037/0033-295X.84.4.327.

[44] Voorhees, Ellen M. (2008), "TREC Text REtrieval Conference," [available at http://trec.nist.gov].

[45] M. Zhang, “ML-RBF: RBF neural networks for multi-label learning,” *Neural Processing Letters*, vol. 29, Apr. 2009, pp. 61-74, doi:10.1007/s11063-009-9095-3.

[46] M. Zhang, J. M. Peña, V. Robles, “Feature selection for multi-label naive Bayes classification,” *Information Sciences*, vol. 179, Sep. 2009, pp. 3218-3229, doi:10.1016/j.ins.2009.06.010.

[47] M. Zhang and Z. Zhou, “Multilabel neural networks with applications to functional genomics and text categorization,” *IEEE Trans. Knowledge and Data Engineering*, vol. 18, Apr. 2006, pp. 1338-1351, doi:10.1109/TKDE.2006.162.

[48] M. Zhang, and Z. Zhou, “ML-KNN: A lazy learning approach to multi-label learning,” *Pattern Recognition*, vol. 40, Jul. 2007, pp. 2038-2048, doi:10.1016/j.patcog.2006.12.019.