ABSTRACT
In the past few years co-clustering has emerged as an important data mining tool for two way data analysis. Co-clustering is more advantageous over traditional one dimensional clustering in many ways such as, ability to find highly correlated sub-groups of rows and columns. However, one of the overlooked benefits of co-clustering is that, it can be used to extract meaningful knowledge for various other knowledge extraction purposes. For example, building predictive models with high dimensional data and heterogeneous population is a non-trivial task. Co-clusters extracted from such data, which shows similar pattern in both the dimension, can be used for a more accurate predictive model building. Several applications such as finding patient-disease cohorts in health care analysis, finding user-genre groups in recommendation systems and community detection problems can benefit from co-clustering technique that utilizes the predictive power of the data to generate co-clusters for improved data analysis.

In this paper, we present the novel idea of Predictive Overlapping Co-Clustering (POCC) as an optimization problem for a more effective and improved predictive analysis. Our algorithm generates optimal co-clusters by maximizing predictive power of the co-clusters subject to the constraints on the number of row and column clusters. In this paper precision, recall and f-measure have been used as evaluation measures of the resulting co-clusters. Results of our algorithm has been compared with two other well-known techniques - K-means and Spectral co-clustering, over four real data set namely, Leukemia, Internet-Ads, Ovarian cancer and MovieLens data set. The results demonstrate the effectiveness and utility of our algorithm POCC in practice.

Keywords
Co-clustering, Predictive power, Simulated annealing

1. INTRODUCTION
The real life data in general can be considered dyadic in nature, i.e. the data can be represented as a two dimensional matrix with rows and column being two separate entities of interest. Some common examples include co-occurrence matrix, rating matrix, and proximity matrix. An important problem in dyadic data analysis is finding block structures hidden in the data matrix. Finding hidden blocks of data can be beneficial in several applications. For example, we may be interested in finding groups of patients that show similar activity pattern under a specific subset of health care conditions [29], simultaneously clustering movies and user ratings in collaborative filtering [13], finding document and word clusters in text clustering [12], grouping genes with similar properties based on their expression patterns under various conditions or across different tissue samples in bioinformatics [6, 8]. Co-clustering is an important and efficient solution for this purpose that exploits the duality between data point and features by grouping them based on their distribution over the other (data points or features) [11, 16].

Most of the co-clustering algorithm focuses on finding co-clusters with single membership of a data point in the data matrix [12, 2]. Although these techniques generate efficient results over real data set, these algorithms are based on the assumption that, a single data point can belong to only one cluster. This assumption is often not completely valid since, in real life there is a high probability that a single data point can belong to multiple clusters with varying degree of its membership with the clusters. For example, in recommendation system a group of user may prefer pop music as well as country music. In fact, several real life situations that deal with high dimensional data with heterogeneous population can benefit more from finding co-clusters that overlap each other. One important example can be finding co-cluster from Electronic Health Records or EHR (hospital data) for predictive analysis in health care. EHR data in health care is often high dimensional with heterogeneous population that makes co-clustering a suitable approach for finding groups of patients and disease conditions. However, each of these co-clusters of patient-disease condition should reflect patient sub-populations that potentially share co-morbid diagnoses as shown in Figure 1. Hence, in this scenario detecting overlapping co-clusters would help capture the most utilisable pattern that exist in the data.

There are past researches that developed different approaches of generating co-clusters such as bi-partite graphs [11] or model based [3] co-clustering techniques. However, develop-
ing a co-clustering approach based on classification principles can be advantageous in generating co-clusters that are high in their predictive capability. In this paper we present a novel co-clustering algorithm called Predictive Overlapping Co-Clustering (POCC). The key idea is that, our algorithm generates optimal co-clusters by maximizing predictive power of the co-clusters subject to the constraints on the number of row and column clusters. Co-clustering can be defined as the simultaneous clustering of rows and columns of a data matrix subjected to optimization of some given criteria. We define POCC as an optimal co-clustering that minimizes the “loss” in predictive power between original data matrix and the co-clustered data matrices. Equivalently, we define it as the one that maximizes the predictive power between original data matrix and the co-clustered data matrices. In this paper, we present a novel algorithm that maximizes a given gain function. Since, we aim concomitantly to develop an overlapping clustering, the goal of this algorithm is to seek a “soft” clustering of both dimensions such that the “gain” in “Predictive Power” of the co-clusters is maximized given a fixed number of row and column clusters. We assume that, class information is available for calculating the predictive power during cluster formation. The resulting algorithm is interesting since at each step, the algorithm encompasses row clusters as well as column clusters for noise removal. The row clusters are generated by identification of a distinguished soft partition of the data matrix in the row dimension such that data point belonging to a partition has strong intra-object resemblance. The column clusters are generated in a similar way. The optimal clustering criteria for a soft partition can be obtained using generalized least squared error functions [4]. Our algorithm is suitable for high dimensional data because it reduces dimensions at each iteration by removing noisy rows and columns, thus estimating parameters fewer than the traditional one dimensional clustering. Our algorithm is essentially a co-clustering algorithm that uses classification technique. The result is a set of overlapping co-clusters with reduced row and column noise and a higher predictive power than the original data.

For evaluation of our algorithm we compare POCC with two other traditional co-clustering algorithms. Evaluation measures that we use are cluster precision, recall and f-measure calculated over pairs of points, as defined in [3]. The main contributions of this paper are -

- We propose a novel co-clustering algorithm which generates overlapping co-clusters that are high in their predictive capability.
- We demonstrate using empirical results that, our approach yield better quality clusters as compared to two other traditional clustering algorithms in most of the data sets.
- co-clusters generated have higher predictive power than the original data and are suitable for complex predictive model building in applications such as health care and recommendation systems.

2. RELATED WORK

Co-clustering has become a topic of interest in the past few years due to the numerous important applications in which it has achieved success such as for finding gene expression patterns [28], document and word clustering [11, 12], clustering tags and social data sources [14], recommendation systems [13]. Co-clustering has been applied successfully in other areas such as Biological networks [17], co-clustering medical images [18], co-clustering of denatured hemoglobin [31]. Popular techniques of co-clustering are bipartite spectral graph partitioning [11], information-theoretic co-clustering [12] and collaborative filtering framework based co-clustering [13]. The earliest works in co-clustering was done in 1972 using hierarchical row and column clustering in matrices by a local greedy splitting procedure [19]. In this paper, the author proposed a hierarchical partition based two way clustering algorithm that splits the original data matrix into set of sub-matrices and used variance for evaluating the quality of each sub matrix. In this paper a criteria for partitioning other than a constant value was also proposed, for example a two way analysis of variance model and a mean squared residue scoring approach. Later this method was improved by [35] that introduced a backward pruning method for generating an optimal number of two way clusters. [6] proposed a co-clustering algorithm that uses a mean squared residue as the measure of the coherence of the genes and conditions for analysis of gene expression data.

In Information theory domain [36] proposed an approach called “Information bottleneck theory” that was developed for one dimensional clustering. Later [12] extended their work and proposed a co-clustering technique using the concepts of information theory where the authors posed the co-clustering problem as an optimization problem where a non-negative matrix was viewed as an empirical joint probability distribution of two discrete random variables. Another important paper [11] proposed a co-clustering technique that was modeled based on bi-partite graphs and their minimal cuts. In this paper, the authors used a graph formulation with a spectral heuristic that uses eigenvectors to co-cluster documents and words in text mining. In our paper we compare our technique with this spectral co-clustering [11] technique for evaluation purpose in the later sections.

Most of the works in the past have focused on “crisp” or partition based co-clustering and very few recent research can handle overlapping co-clusters. Even for one-way clustering, there are few algorithms known as “soft” clustering algorithms which can identify overlapping clusters. One of the earliest example is fuzzy c-means clustering [4]. One of the notable works in overlapping co-clustering was [32] where the authors have proposed an overlapping co-clustering model
that can be applied with a variety of clustering distance functions. Other important works in overlapping co-clustering that has been shown to be of immense utility in various fields includes [21, 33, 24, 57] and [38].

There are very few works in the past, to the best of our knowledge that utilizes predictive power of a data matrix to generate co-clusters that are of improved quality and predictive power. A related semi-supervised research has been done by [10] where the authors have developed a classification algorithm using the concept of clustering as an extension to information theoretic co-clustering [12].

Ours technique is a semi-supervised approach towards developing an improved co-clustering technique that can find utilization in fields which deals with data with heterogeneous distribution. In this paper, we propose a novel co-clustering model that utilizes predictive power of the co-clusters to optimize the final co-clusters formed by removing noisy rows and noisy columns. The noisy rows and noisy columns are removed using a simulated annealing approach for optimizing the noise removal [5].

The result is co-clusters that have greater predictive power than the original data matrix. In this paper, we aim to develop a novel co-clustering technique for generating high quality co-clusters that would aid in an improved predictive analysis.

3. PROBLEM STATEMENT

Common data mining applications such as predictive modeling often handles data that might have been drawn from heterogeneous populations with different distributions. In the past, many co-clustering techniques have been developed that considered the importance of heterogeneity in population [22, 28, 7]. In real life, often data is the result of concocting of multiple sub-populations and hence, it is unlikely that individual in the population will have similar set of parameter values. For example, in health care, electronic health records (EHRs) data consists of valuable information about diverse patient sub-populations combined together. EHR hospital data is often very high dimensional and has heterogeneous population. In other words, EHR consists of a large number of hidden naturally occurring homogeneous groups of patients. Statistical insights describing an overall population are beneficial, but they are often not specific enough to be used as the basis for a patient centered decisions making. Standard practices rely on extracting useful cumulative summary statistics from data, with an underlying assumption that patients across a population can be analyzed uniformly without paying attention to the heterogeneity in their treatment response pattern and historical data. However, while most patients respond to similar treatments and follow predictable trajectories, there are segments of patients with distinct complicated trajectories. Thus, in order to improve the efficiency and effectiveness of predictive models, it is necessary to segment the original data space contemporaneously in to co-clusters of patients that show similar activity pattern under a specific subset of health care conditions. Predictive analysis using co-clusters instead of the whole data matrix, which contains thousands of dimensions and millions of records, would result in an augmented success in the process of decision making in health care research. The statistical insights taken from these co-clusters would help effective predictive model building and an improved health outcome. Similarly, in other domains, such as recommendation systems, finding customer-genre co-clusters from a heterogeneous population would help provide guidance for complex predictive model building and improved prediction of an outcome of interest for example, predicting movies for customers with similar interest.

Keeping in mind the above two requirements - 1) overlapping co-clusters and 2) improved predictive power of the co-clusters, we seek to build a overlapping co-clustering algorithm that takes into account the predictive power of the co-clusters while its formation. The result will be a set of co-clusters with higher predictive power than the original data matrix. These co-clusters would assist in complex predictive model building and achieving an improved outcome of interest. POCC algorithm is interesting as it uses predictive power of the co-clusters to detect and eliminate noisy rows and noisy columns while cluster formation and results in homogeneous blocks of data that are high in their predictive capability. Most importantly, this algorithm seeks to find overlapping or "soft" co-clusters that qualifies it as an algorithm that is closely capable of capturing the structure of real world data.

3.1 Problem Formulation

Let $C$ represents the data matrix denoting a $m \times n$ matrix. Each object in $C$ belongs to objects from the rows $X$ represented as $x_1, x_2, ..., x_m$ and columns $Y$ represented as $y_1, y_2, ..., y_n$.

Our goal is generating co-clusters $(\hat{X}, \hat{Y})$ from $C$ by iterative clustering of $X$ and $Y$ into into $k$ and $l$ "soft" clusters respectively until it reaches a given objective function. Let $k$ clusters of $X$ be denoted as $x_1, x_2, ..., x_k$ and $l$ clusters of $Y$ be denoted as $y_1, y_2, ..., y_l$. Let the co-clusters for row and columns be formuated as $M_X$ and $M_Y$ defined as

$$\hat{X} = M_X(X) = x_1, x_2, ..., x_m \rightarrow \hat{x_1}, \hat{x_2}, ..., \hat{x_k}$$ and

$$\hat{Y} = M_Y(Y) = y_1, y_2, ..., y_n \rightarrow \hat{y_1}, \hat{y_2}, ..., \hat{y_l}.$$  

We denote the tuple $(M_X, M_Y)$ as a co-cluster mapping function. Given a tuple $(M_X, M_Y)$ we generate co-clusters as follows - we remove noise while row wise soft clustering followed by noise removal while column wise soft clustering. We re-order the X such that all rows in X belonging to $\hat{x_1}$ cluster are arranged first, followed by all rows in X belonging to $\hat{x_2}$ and so on. Similarly, we re-order the Y such that all rows in Y belonging to $\hat{y_1}$ cluster are arranged first, followed by all rows in Y belonging to $\hat{y_2}$ cluster are arranged and so on. The result is that the data matrix is divided into small two dimensional blocks which we denote as co-clusters.

In general, cluster evaluation is highly subjective and this task relies upon the judgment of the domain expert as validation of the clusters can be expensive and time consuming. Clustering results can be evaluated using application specific criteria such as predictive power [54]. Predictive power of a co-cluster may be referred as the ability of a co-cluster to generate testable predictions. Following similar logic we improve the quality of the co-clusters using the criteria - gain in predictive power as defined in equation [5].
The gain in predictive power \( F(\hat{X}; \hat{Y}) - F(X; Y) \) can be explained as the quantity that facilitates the search for an optimal co-clustering. \( F(X; Y) \) is the function that generates the predictive power in \((X; Y)\) data matrix. \( F(\hat{X}; \hat{Y}) \) is the function that generates the predictive power of \((X; Y)\) co-clusters. We assume that we have the class labels for our predictive power analysis task.

The main component of our objective function is the predictive power of data matrix as well as the co-clusters. In this paper, we consider classification accuracy of a data matrix as the predictive power. Therefore, \( F(X; Y) \) can be expressed as defined in equation 2

\[
F(X; Y) = \frac{t_p + t_n}{t_p + t_n + f_p + f_n}
\]  

where \( t_p \) is the number of true positives, \( t_n \) is the number of true negatives, \( f_p \) is the number of false positives and \( f_n \) is the number of false negatives in the given classification task. Similarly, in this paper, we consider \( F(\hat{X}; \hat{Y}) \) as the mean predictive power of the generated co-clusters. This can be defined as given in equation 3

\[
F(\hat{X}; \hat{Y}) = \frac{1}{j} \sum_{i=1}^{j} \frac{t_{pi} + t_{ni}}{t_{pi} + t_{ni} + f_{pi} + f_{ni}}
\]  

where \( j \) is the number of co-clusters generated. \( F(\hat{X}; \hat{Y}) \) equals to the mean predictive power of the co-clusters generated. \( t_{pi} \), \( t_{ni} \), \( f_{pi} \) and \( f_{ni} \) refers to the number of true positives, true negatives, false positives and false negatives of the \( i^{th} \) co-clusters respectively.

Therefore, the objective function can be rewritten as given in equation

\[
\text{Maximize} \left( \frac{1}{j} \sum_{i=1}^{j} \frac{t_{pi} + t_{ni}}{t_{pi} + t_{ni} + f_{pi} + f_{ni}} - \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \right)
\]  

s.t. the constraints on number of row cluster \( k \) and number of column cluster \( l \), where \( 1 < j < k \times l \) in a given iteration depending on the number of noisy rows and columns detected. In this equation we assume that \( F(\hat{X}; \hat{Y}) \) the mean predictive power of the co-clusters generated in any given iterative stage. One can consider any empirical scores such as the highest or lowest predictive power of a cluster in a given set of detected co-clusters e.t.c.

Definition 2. The co-clustering algorithm generates a non-decreasing objective function in each iteration, iff there is noise removal i.e.

\[
F(\hat{X}; \hat{Y})^{(t')} - F(X; Y)^{(t')} < F(\hat{X}; \hat{Y})^{(t'+1)} - F(X; Y)^{(t'+1)}
\]  

\[
F(\hat{X}; \hat{Y})^{(t')} - F(X; Y)^{(t')} < F(\hat{X}; \hat{Y})^{(t'+1)} - F(X; Y)^{(t'+1)}
\]  

Proof Given the predictive power of the actual data matrix \( F(X; Y)^{(t')} \) is constant in each iteration \( F(X; Y)^{(t')} = F(X; Y)^{(t'+1)} \) for all \( t' \), let us omit this term for a while from equation 5 for ease of understanding. Let us denote the predictive power of the co-clusters as

\[
\tilde{F}(\hat{X}; \hat{Y})^{(t')} = \frac{\tilde{f}_a^{(t')}}{\tilde{t}_a^{(t')} + \tilde{f}_a^{(t')}}
\]  

where \( \tilde{f}_a^{(t')} \) is mean number of false positives and false negatives of a set of co-clusters in the \((t')^{th} \) iteration. Similarly \( \tilde{t}_a^{(t')} \) is the mean number of true positives and true negatives of the co-clusters in the \((t')^{th} \) iterations.

Now, assuming there is noise removal with an increase in predictive power at a given iteration, two conditions will be met - 1) decrease in the number of instances in the \((t'+1)^{th} \) iteration, i.e.

\[
\tilde{f}_a^{(t')} + \tilde{t}_a^{(t')} > \tilde{f}_a^{(t'+1)} + \tilde{t}_a^{(t'+1)}
\]  

2) Since, the predictive power increases i.e. error decreases, therefore either of the following is true \( \frac{\tilde{f}_a^{(t')}}{\tilde{t}_a^{(t')} + \tilde{f}_a^{(t')}} = \frac{\tilde{f}_a^{(t'+1)}}{\tilde{t}_a^{(t'+1)} + \tilde{f}_a^{(t'+1)}} \)

OR \( \frac{\tilde{f}_a^{(t')}}{\tilde{t}_a^{(t')} + \tilde{f}_a^{(t')}} = \frac{\tilde{f}_a^{(t'+1)} + y}{\tilde{t}_a^{(t'+1)} + \tilde{f}_a^{(t'+1)} + y} \) where \( y \) is the decrease in the number of false positives and false negatives and \( y \) is the increase in the number of true positives and true negatives.

Hence, in either of the case \( \frac{\tilde{f}_a^{(t'+1)} + y}{\tilde{t}_a^{(t'+1)} + \tilde{f}_a^{(t'+1)}} > \frac{\tilde{f}_a^{(t')}}{\tilde{t}_a^{(t')} + \tilde{f}_a^{(t')}} \). As

\[
\frac{\tilde{f}_a^{(t'+1)} + y}{\tilde{t}_a^{(t'+1)} + \tilde{f}_a^{(t'+1)}} - F(X; Y)^{(t'+1)} > \frac{\tilde{t}_a^{(t')}}{\tilde{t}_a^{(t')} + \tilde{f}_a^{(t')}} - F(X; Y)^{(t')}
\]  

it proves that the objective function is non-decreasing in each iteration iff there is a noise removal.

4. METHODOLOGY
4.1 Co-clustering Algorithm
We describe a novel co-clustering algorithm POCC based on maximization of the objective function defined in equation subject to the constraints on the number of row and column clusters. The entire algorithm has been shown diagrammatically in figure 6. We also tabulated all the notations in table II for the ease of understanding and reference. We propose to develop POCC as an iterative process, with co-clusters getting refined in each iteration aided by the removal of noisy row and noisy columns while maximization of the objective function. As the cluster mapping function, we plan to use fuzzy C-means clustering [23], which is an overlapping one dimensional clustering algorithm. For model convergence we plan to use two mechanisms namely, 1) model optimization by satisfying the objective function and 2) by using the concept of simulated annealing [20]. Simulated annealing is an interesting and useful optimization tool firstly because, it would assist in removing noisy rows (patients) and noisy
column (variables) from the data matrix and secondly, it would enable the model to converge to a global optima with co-clusters with the maximum predictive power. The expected outcome is a group of co-clusters with higher predictive power than the original data set.

The co-clustering algorithm works as follows. At first an initial "soft" co-clustering \((M^1_X, M^1_Y)\) is generated with successive removal of noisy row and noisy columns. The algorithm then generates co-clustering \((M^{2}_X, M^{2}_Y)\), \((M^{3}_X, M^{3}_Y)\) ... in successive iteration until the objective function is reached. The full algorithm is detailed in Algorithm 1.

In step 1, a "soft" row clustering \(M_X\) is obtained from \(C\). The predictive power of each row cluster \(x_1, x_2, ..., x_k\) (let it be \(\rho_{row1}; \rho_{row2}; ..., \rho_{rowk}\)) is compared with the predictive power of \((X; Y)\), (lets call it \(\rho\)) assuming we have the class label information. We assign a probability threshold \(\tau_{row}\) for comparison and determining if a row cluster should be removed as a noisy row cluster. Using simulated annealing [20] the algorithm determines \(P_{row}\), the probability of a row cluster to be considered as noise. The equation (9) shows the computation of \(P_{row}\) as a general calculation. \(\Delta \rho\) represents the gain in predictive power of the a cluster from last iteration. \(T\) is the cooling schedule parameter that controls the objective function from reaching a bad local maxima. If \(P_{row}\) for a given iteration meets the constraint \(\tau_{row}\), it denotes the probability of removing noisy row is optimum for the current iterative stage. If any of \(\rho_{row1}; \rho_{row2}; ..., \rho_{rowk}\) does not meet \(\tau_{row}\), those rows belonging to the row cluster is retained and then it proceeds to the next step with the new data \(C'\).

\[
P = e^{-\Delta \rho / T} \tag{9}
\]

In step 2, a "soft" column clustering \(M_Y\) is obtained from \(C'\). The predictive power of each column cluster \(y_1, y_2, ..., y_l\) (let it be \(\rho_{col1}; \rho_{col2}; ..., \rho_{coll}\)) is compared with the predictive power of \((X; Y)\) i.e. \(\rho\) assuming we have the class label information. We assign a probability threshold \(\tau_{col}\) for comparison and determining if a column cluster should be removed as a noisy column cluster. Using simulated annealing [20] the algorithm determines \(P_{col}\), the probability of a column cluster to be considered as noise. If \(P_{col}\) as calculated from equation 9 for a given iteration meets the constraint \(\tau_{col}\), it denotes the probability of removing noisy column is optimum for the current iterative stage. If any of \(\rho_{col1}; \rho_{col2}; ..., \rho_{coll}\) does not meet \(\tau_{col}\), it is retained and the process proceeds to the next step.

In step 3, co-clusters are generated from the new data matrix \(C''\) after removing noisy row and noisy columns (if any). The algorithm now checks if the newly formed co-clusters meets the objective function. This check is done after each iteration. If the objective function is not met, the algorithm proceeds with step one again with \(C''\). The objective function is coupled with the convergence criteria \(\tau_{con}\) for each iteration. If both is met in any iteration, the algorithm stops and outputs the co-clusters as the final result. \(\tau_{con}\) is controlled using the concept of simulated annealing [20] following equation (9) that helps avoid the objective function from reaching a bad local maxima.

### 5. EXPERIMENTAL RESULTS

In the following sections we show early results of POCC algorithm over four real data and compare them with two popular techniques known as K-means and Spectral co-clustering [11]. We compare the results of cluster precision, recall and J-measure as described in the following section.

#### 5.1 Data set used

We use four real world data set namely MovieLens data set, Internet-Ads, Leukemia and Ovarian, as given in table 2. We used Internet-ads data set [25], which represents a set of possible advertisements on Internet pages. The features of this data set consists of the image geometry, URL and text of the image, phrases occurring in the URL and words occurring near the anchor text and the anchor text itself. Leukemia data set [15] is a micro array data set of molecular classification of cancer. We also use Ovarian data set [27] obtained from National Cancer Institute. We used MovieLens data set, which is a publicly available data set used for movie recommendation systems developed by grouplens.org in University of Minnesota. The MovieLens data set consisted of 100,000 movie ratings by 940 users for 1673 movies. Each user rated the movies at a scale of 1-5, where 1 denotes extreme dislike and 5 denotes strong approval. We choose three genres - Action, Adventure and Animation movies rated by 940 users for 1673 movies. Detailed description of the data sets are given in table 2. In all data set we consider binary class membership except the MovieLens data where we consider three categories or class labels. The class label information was made available as the true group information, for the predictive analysis during cluster for-

| Notations | Descriptions |
|-----------|--------------|
| \(C\)    | Original data matrix |
| \(X\)    | rows of \(C\) |
| \(Y\)    | columns of \(C\) |
| \(x_1, x_2, ..., x_m\) | objects in \(X\) taking value from \(X\) |
| \(y_1, y_2, ..., y_n\) | objects in \(Y\) taking value from \(Y\) |
| \(M_X\)  | Co-cluster functions for row |
| \(M_Y\)  | Co-cluster functions for column |
| \(x_1, x_2, ..., x_k\) | \(k\) clusters of \(X\) |
| \(y_1, y_2, ..., y_l\) | \(l\) clusters of \(Y\) |
| \(F()\)  | Predictive power function |
| \(\rho\) | Number of false positives |
| \(\rho_n\) | Number of false negatives |
| \(\rho_p\) | Number of true positives |
| \(\rho_t\) | Number of true negatives |
| \(\rho'\) | number of iteration |
| \(\rho\) | Predictive power of \(C\) |
| \(\Delta \rho\) | Gain in \(\rho\) from last iteration |
| \(\rho_{row1}; ..., \rho_{rowk}\) | Predictive power of each row cluster |
| \(\rho_{col1}; ..., \rho_{coll}\) | Predictive power of each column cluster |
| \(\tau_{row}\) | threshold for row noise removal |
| \(\tau_{col}\) | threshold for column noise removal |
| \(P_{row}\) | probability of iteration |
| \(P_{row}\) | probability of row noise removal |
| \(P_{col}\) | probability of column noise removal |
| \(\tau_{con}\) | Convergence criteria |
| \(C'\) | Data matrix after row noise removal |
| \(C''\) | Data matrix after column noise removal |

Table 1: Notations and their meaningful description
Algorithm 1 Predictive Overlapping co-clustering

Input:
- Data Matrix $C$, $k$ row clusters, $l$ column clusters, $T$ cooling schedule parameter, probability threshold $\tau_{row}$, $\tau_{col}$, $\tau_{ccr}$

Output:
- co-clusters $m_x, m_y$

Algorithm:
1. Computer $\rho_x$ - Predictive power of $C$.
2. Initialization, set $t=1$ the number of iteration, start with a "soft" row clustering $M'_X$.
3. Compute $\rho_{row1}, \rho_{row2}, ..., \rho_{rowk}$ the predictive power of the row clusters $x_1, x_2, ..., x_k$
4. Compute $P_{row}$ the probability that a row cluster $x_k$ is a noise using equation 9.
5. Compare $P_{row}$ to $\tau_{row}$. Remove the rows from $C$ that belongs to $x_a$, where $1 < a < k$ for which $P_{row} \geq \tau_{row}$. Update $C$ to $C'$.
6. Compute "soft" column clustering $M'_Y$ using $C'$.
7. Compute $P_{col1}, P_{col2}, ..., P_{colm}$ the predictive power of the col clusters $y_1, y_2, ..., y_l$
8. Compute $P_{col}$ the probability that a col cluster $y_l$ is a noise using equation 9.
9. Compare $P_{col}$ to $\tau_{col}$. Remove the columns from $C'$ that belongs to $y_b$, where $1 < b < k$ for which $P_{col} \geq \tau_{col}$. Update $C'$ to $C''$.
10. Compute $m_x, m_y$ the co-clusters from $C''$ using Algorithm 2.
11. Compute $P_{itr}$ the probability of convergence/ new iteration using equation 9.
12. Output $m_x, m_y$ if the objective function in equation 11 is met and $P_{itr} \geq \tau_{ccr}$. Other wise continue Step 2 with $C''$ and $t=2$.

Algorithm 2 Generate Co-clusters

Input:
- Data Matrix $C''$, row clusters $x_1, x_2, ..., x_k$, column clusters $y_1, y_2, ..., y_l$

Output:
- co-clusters $m_x, m_y$

Algorithm:
1. Re-order the $X$ such that all rows in $X$ belonging to $x_1$ are arranged first, followed by all rows in $X$ belonging to $x_2$ and so on.
2. Re-order the columns in $Y$ such that all columns in $Y$ belonging to $y_1$ are arranged first, followed by all columns in $Y$ belonging to $y_2$ and so on.
3. Output the small two dimensional blocks generated from dividing $C''$ by reordering the rows and columns by row clusters and column clusters.

Table 2: Data Sets with Attributes and Instances

| Data Set   | Attributes | Instances |
|------------|------------|-----------|
| Internet-Ads | 1559       | 3279      |
| Ovarian    | 15155      | 253       |
| Leukemia   | 7130       | 72        |
| MovieLens  | 417        | 940       |

5.2 Evaluation Measures

In this paper we evaluate the quality of the co-clusters using three types of evaluation measures as follows.

- **Precison.** We use precision for evaluating cluster quality as defined in [32]. In the overlapping clustering results, the measure precision tries to estimate whether the prediction of each pair of points that share at least one cluster, are in the same cluster and it is correct with respect to the underlying true groups or class in the data. Precision is calculated as the fraction of pairs of objects correctly put in the same cluster as given in equation 10.

$$
\text{Precision} = \frac{\text{Number of Correctly Identified pairs}}{\text{Number of Identified pairs}}
$$

(10)

- **Recall.** We use Recall for evaluating cluster quality as defined in [32]. In the overlapping clustering results, the measure recall tries to estimate whether the prediction of each pair of points that share at least one cluster, are in the same cluster and it is correct with respect to the true groups or class in the data. Recall is calculated as the fraction of actual pair of objects that were identified as given in equation 11.

$$
\text{Recall} = \frac{\text{Number of Correctly Identified pairs}}{\text{Number of True pairs}}
$$

(11)

- **F-measure.** This is the harmonic mean of precision and recall using equation 10 and 11 [32] above. This can be calculated as given in equation 12.

$$
F\text{- measure} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}
$$

(12)

5.3 Results and Discussion

5.4 Results

In this section, we demonstrate the results of POCC over four data set and compare them with that of two other clustering techniques. In figure 2 we plot the number of iterations vs gain in the predictive power (accuracy in our case), showing a run of our algorithm POCC with movieLens data set. It is worth noticing that the objective function value, which is the gain in predictive power, is consistently non-decreasing with the number of iterations. This shows that POCC improves cluster quality in iterations until the final objective function is met. The risk of reaching bad local
maxima is avoided by using the concept of simulated annealing by generating the probability of noise removal as well as probability of iteration. In figures 3, 4 and 5 we demonstrate the result of cluster precision, recall and f-measure as defined in equations 10, 11 and 12. All the data set contains binary class labels. The number of row cluster as well as column cluster has been primarily set as parameters in the first iteration of the algorithm for all data set. The number of initial row and column clusters including the cooling schedule are parameters that will be provided by the user. In this analysis, we considered “accuracy” of classification as the predictive measure of a cluster. The number of clusters including the co-clusters slowly gets refined as the algorithm proceeds. Evaluating clusters is a non-trivial task. Hence, we aimed at evaluating our co-clustering algorithm by analyzing the cluster quality with the three measures mentioned above. Precision, Recall and F-measure are the standard benchmark evaluation measures that have been used in many co-clustering works such as [3] and [32] with convincing outcomes.

In figure 4 we observe that POCC has a significantly higher precision in Leukemia and Internet-ads data set as compared to K-means and Spectral co-clustering. The precision in the data sets Leukemia and MovieLens is marginally less than K-means and spectral co-clustering. In figure 5 we found that in ovarian and MovieLens data set, the recall is significantly higher than the other methods. In data set Leukemia the recall is higher with POCC than K-means. Recall with spectral clustering in the same data set is 1.0 which is the highest among the three. However, recall of 1.0, indicates that there might be high probability that all data point had been assigned to a single cluster since the precision is not 1.0, with spectral clustering. Thus, we can claim that in Leukemia data set, recall of POCC is better. Figure 6 shows the cluster f-measure of the four data set. It indicates that POCC performs significantly better than the two algorithms in most of the data sets. This is because, though precision in two of the data set is lower compared to K-means and spectral co-clustering, the f-measure is balanced by the higher recall value indicating generation of better quality cluster with POCC. These results show that POCC performs comparatively better in terms of cluster quality analysis. Since, in figure 2 we have shown that gain in predictive power of the co-cluster is non-decreasing in each iteration, we can safely assume that, predictive power of the co-clusters would be higher than the actual data matrix, if there is sufficient noise removal. This shows the success of our primary motive i.e. problem of finding co-clusters for a predictive analysis in heterogeneous data population. POCC will facilitate generation of co-clusters that can be used for an improved predictive analysis. Our experiments and the results provide initial evidence for the success of our co-clustering framework. We believe that this framework has the potential to bring about improvement in the predictive analysis tasks in many different application domains.

6. CONCLUSION AND FUTURE WORK

In this paper, we proposed a co-clustering strategy that uses predictive score of the data set for improving the quality of co-clusters. We present POCC as an optimization problem where the main task is maximizing the gain in predictive power while improving the quality of co-clusters by removing noisy rows and noisy columns. The result is a set of overlapping co-clusters that are high in their predictive power. The results show that POCC brings about notable improvement in co-clustering with real world data set as compared to two other traditional co-clustering techniques. This proves that POCC is well suited for many real life applications where we handle high dimensional data set with heterogeneous population. Prediction analysis with such high dimensional heterogeneous data is complex. Our algorithm POCC is a novel algorithm that generates co-clusters emphasizing on the predictive power i.e. classification accuracy or error rate for noise removal and improving the quality of the co-cluster. POCC can find applications in many different fields of healthcare, recommendation systems and other areas where data with heterogeneous population is a common
Figure 6: Diagrammatic representation of the algorithmic flow

Our future plan is improvement of our algorithm to adapt our technique in different interesting application areas.

7. ACKNOWLEDGMENTS

We would like to thank Atanu Roy for his critical reviews and technical feedback during developing this research. We also thank the GroupLens Research Group at the University of Minnesota for the MovieLens data set.

8. REFERENCES

[1] V. Athitsos and S. Sclaroff. Boosting nearest neighbor classifiers for multiclass recognition. In Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on, pages 45–45. IEEE, 2005.

[2] A. Banerjee, I. Dhillon, J. Ghosh, S. Mallela, and D. S. Modha. A generalized maximum entropy approach to bregman co-clustering and matrix approximation. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 509–514. ACM, 2004.

[3] A. Banerjee, C. Krumpelman, J. Ghosh, S. Basu, and R. J. Mooney. Model-based overlapping clustering. In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pages 532–537. ACM, 2005.

[4] J. C. Bezdek, R. Ehrlich, and W. Full. Fcm: The fuzzy c-means clustering algorithm. Computers & Geosciences, 10(2):191–203, 1984.

[5] S. Brooks and B. Morgan. Optimization using simulated annealing. The Statistician, pages 241–257, 1995.

[6] Y. Cheng and G. M. Church. Biclustering of expression data. In Ismb, volume 8, pages 93–103, 2000.

[7] A. D. Chiaravalloti, G. Greco, A. Guzzo, and L. Pontieri. An information-theoretic framework for high-order co-clustering of heterogeneous objects. In Machine Learning: ECML 2006, pages 598–605. Springer, 2006.

[8] H. Cho, I. S. Dhillon, Y. Guan, and S. Sra. Minimum sum-squared residue co-clustering of gene expression data. In SDM, volume 3, page 3, 2004.

[9] S. Cooley, D. Weis lurf, L. Guethlein, J. Klein, T. Wang, C. Le, S. Marsh, D. Geraghty, S. Spellman, M. Haagenson, et al. Donor selection for natural killer cell receptor genes leads to superior survival after unrelated transplantation for acute myelogenous leukemia. Blood, 116(14):2411–2419, 2010.

[10] W. Dai, G.-R. Xue, Q. Yang, and Y. Yu. Co-clustering based classification for out-of-domain documents. In Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 210–219. ACM, 2007.

[11] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, pages 269–274. ACM, 2001.

[12] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretic co-clustering. In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 89–98. ACM, 2003.

[13] T. George and S. Merugu. A scalable collaborative filtering framework based on co-clustering. In Data Mining, Fifth IEEE International Conference on, pages 4–pp. IEEE, 2005.

[14] E. Giannakidou, V. Koutsonikola, A. Vakali, and Y. Kompatsiaris. Co-clustering tags and social data sources. In Web-Age Information Management, 2008. WAIM’08. The Ninth International Conference on, pages 317–324. IEEE, 2008.

[15] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. science, 286(5439):531–537, 1999.

[16] Q. Gu and J. Zhou. Co-clustering on manifolds. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 359–368. ACM, 2009.

[17] D. Hanisch, A. Zien, R. Zimmer, and T. Lengauer. Co-clustering of biological networks and gene expression data. Bioinformatics, 18(suppl 1):S145–S154, 2002.

[18] M. Hanmandlu, S. Susan, V. Madasu, and B. Lovell. Fuzzy co-clustering of medical images using bacterial foraging. In Image and Vision Computing New Zealand, 2008. IVCNZ 2008. 23rd International Conference, pages 1–6. IEEE, 2008.

[19] J. A. Hartigan. Direct clustering of a data matrix. Journal of the american statistical association, 67(337):123–129, 1972.

[20] S. Kirkpatrick, D. G. Jr., and M. P. Vecchi. Optimization by simmulated annealing. science,
[21] K. Kummamuru, A. Dhawale, and R. Krishnapuram. Fuzzy co-clustering of documents and keywords. In Fuzzy Systems, 2003. FUZZ’03. The 12th IEEE International Conference on, volume 2, pages 772–777. IEEE, 2003.

[22] G. H. Lubke and B. Muthén. Investigating population heterogeneity with factor mixture models. Psychological methods, 10(1):21, 2005.

[23] N. R. Pal and J. C. Bezdek. On cluster validity for the fuzzy c-means model. Fuzzy Systems, IEEE Transactions on, 3(3):370–379, 1995.

[24] S. Papadimitriou and J. Sun. Disco: Distributed co-clustering with map-reduce: A case study towards petabyte-scale end-to-end mining. In Data Mining, 2008. ICDM’08, Eighth IEEE International Conference on, pages 512–521. IEEE, 2008.

[25] S. P. Pathical. Classification in High Dimensional Feature Spaces through Random Subspace Ensembles. PhD thesis, University of Toledo, 2010.

[26] R. G. Pensà and J.-F. Boulicaut. Constrained co-clustering of gene expression data. In SDM, pages 25–36, 2008.

[27] E. F. Petricoin III, A. M. Ardekani, B. A. Hitt, P. J. Levine, V. A. Fusaro, S. M. Steinberg, G. B. Mills, C. Simone, D. A. Fishman, E. C. Kohn, et al. Use of proteomic patterns in serum to identify ovarian cancer. The lancet, 359(9306):572–577, 2002.

[28] A. Ribbens, J. Hermans, F. Maes, D. Vandermeulen, and P. Suetens. Unsupervised segmentation, clustering and groupwise registration of heterogeneous populations of brain mri images. 2013.

[29] B. Saha, D.-S. Pham, D. Phung, and S. Venkatesh. Clustering patient medical records via sparse subspace representation. In Advances in Knowledge Discovery and Data Mining, pages 123–134. Springer, 2013.

[30] C. Sarkar, S. Cooley, and J. Srivastava. Improved feature selection for hematopoietic cell transplantation outcome prediction using rank aggregation. 2012.

[31] K. Schlüter and D. Drenckhahn. Co-clustering of denatured hemoglobin with band 3: its role in binding of autoantibodies against band 3 to abnormal and aged erythrocytes. Proceedings of the National Academy of Sciences, 83(16):6137–6141, 1986.

[32] M. Shafiei and E. Milios. Model-based overlapping co-clustering. In Proceeding of SIAM Conference on Data Mining, 2006.

[33] M. M. Shafiei and E. E. Milios. Latent dirichlet co-clustering. In Data Mining, 2006. ICDM’06. Sixth International Conference on, pages 542–551. IEEE, 2006.

[34] K. Steinhaeuser, N. V. Chawla, and A. R. Ganguly. Comparing predictive power in climate data: Clustering matters. In Advances in Spatial and Temporal Databases, pages 39–55. Springer, 2011.

[35] R. Tibshirani, T. Hastie, M. Eisen, D. Ross, D. Botstein, P. Brown, et al. Clustering methods for the analysis of dna microarray data. Dept. Statist., Stanford Univ., Stanford, CA, Tech. Rep, 1999.

[36] N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. arXiv preprint physics/0004057, 2000.

[37] X. Wang, L. Tang, H. Gao, and H. Liu. Discovering overlapping groups in social media. In Data Mining (ICDM), 2010 IEEE 10th International Conference on, pages 569–578. IEEE, 2010.

[38] L. F. Wu, T. R. Hughes, A. P. Davierwala, M. D. Robinson, R. Stoughton, and S. J. Altschuler. Large-scale prediction of saccharomyces cerevisiae gene function using overlapping transcriptional clusters. Nature genetics, 31(3):255–265, 2002.