A Game Theoretic Approach for Feature Clustering and Its Application to Feature Selection

Dinesh Garg[1], S. Sundararajan[1], Shirish Shevade[2]

[1] Yahoo! Labs, Bangalore, India. (dineshg,ssrajan@yahoo-inc.com)
[2] Department of CSA, IISc, Bangalore, India. (shirish@csa.iisc.ernet.in)

Abstract. In this paper, we develop a game theoretic approach for clustering features in a learning problem. Feature clustering can serve as an important preprocessing step in many problems such as feature selection, dimensionality reduction, etc. In this approach, we view features as rational players of a coalitional game where they form coalitions (or clusters) among themselves in order to maximize their individual payoffs. We show how Nash Stable Partition (NSP), a well known concept in coalitional game theory, provides a natural way of clustering features. Through this approach, one can obtain some desirable properties of the clusters by choosing appropriate payoff functions. For a small number of features, the NSP based clustering can be found by solving an integer linear program (ILP). However, for large number of features, the ILP based approach does not scale well. Interestingly, a key result that we prove on the equivalence between a k-size NSP of a coalitional game and minimum k-cut of an appropriately constructed graph comes in handy for large scale problems. We thus propose a hierarchical approach for feature clustering to large scale problems. In this paper, we use feature selection problem (in a classification setting) as a running example to illustrate our approach. We conduct experiments to illustrate the efficacy of our approach.[3]

1 Introduction

In many supervised and unsupervised learning problems, one often needs to conduct a preprocessing step on a feature set representing the input to make the learning task feasible. Having some insights about the usefulness of the features can add value to finding the better solutions. For example, imagine a classification problem where the number of features is so large that it is not possible to train any satisfactory model in the allotted time with available resources. In such a situation, one needs to select a subset of features on which a model can be trained in a reasonable amount of time without significant degradation in generalization performance. This problem is known as the *feature selection* problem [12], [9], [17]. We believe that if we can group the features according

[3] We will keep using the terms coalition and cluster interchangeably.

to the following two criteria then it will not only make the task of feature selection easy but also would greatly improve the quality of the features selected and hence the final generalization performance.
(1) Relevant Features vs Irrelevant Features: For a classification task, a feature f_i is a relevant feature if removal of f_i alone will result in performance deterioration of an optimal classifier. Otherwise, we call it an irrelevant feature. See Kohavi and John [12] for a detailed discussion.
(2) Substitutable vs Complementary Features: For a classification task, we characterize two features f_i and f_j as substitutable features if there is no significant difference in the generalization performance of a classifier that is trained using both the features and the generalization performance of a classifier that is trained using just (any) one of these two features. On the other hand, if the generalization performance of the first classifier is significantly better than that of the latter ones then we attribute these two features as complementary features.

It is easy to see that having the above insights about the features would greatly simplify the task of feature selection - *one just needs to select a set of features of the desired size in such a manner that all the features in that set are* relevant *and* complementary *to each other.* This insight about the features can be obtained by using the feature clustering approach proposed in this paper.

In this paper, we develop a novel game theoretic approach for clustering the features where the features are interpreted as players who are allowed to form coalitions (or clusters)[4] among themselves in order to maximize their individual payoffs (defined later). In this approach, the choice of a payoff function would determine whether all the features within a cluster would be substitutable or complementary to each other. It is important to mention that although we demonstrate the feature clustering approach through the idea of substitutable and complementary features, the approach is quite generic and can be used in other problems like dimensionality reduction and community detection in web graphs [7]. For any other problem, one needs to select a payoff function appropriately so that the desired insights become apparent in clustering of the features. We believe that this game theory based approach for feature clustering is quite novel and unique till date.

The key contributions of this paper are as follows: (1) We draw an analogy between a coalitional game and feature clustering problem and then show that Nash Stable Partition (NSP), a well known solution concept in coalitional game theory [15], provides a natural way of clustering the features; (2) We show how to get an NSP based clustering by solving an integer linear program (ILP). The solution to this ILP gives an NSP based clustering of the features for a given payoff function under the assumption of *pure hedonic* setting (defined later); (3) We illustrate that depending upon how a payoff function is chosen, the NSP based clustering would have a property that either substitutable features are grouped together in one cluster or complementary features are grouped together in one cluster. One can use any standard technique [9], [17] to choose features from these clusters. We also suggest a simple cluster ranking based technique for

[4] We will keep using the terms coalition and cluster interchangeably.

selecting the features from a given set of feature clusters; (4) Finally, we propose a hierarchical scheme for feature clustering for the scenario where the feature set size is very large and solving the ILP is very expensive. Interestingly, a key result that we prove on the equivalence between a k-size NSP of a coalitional game and minimum k-cut of an appropriately constructed graph [13],[4] comes in handy for large scale problems. We thus propose a hierarchical approach for feature clustering to large scale problems; (5) We demonstrate the efficacy of our approach through a set of experiments conducted on real world as well as synthetic datasets.

In what follows, we give a brief review of the related literature in Section 2 and then summarize the coalitional game fundamentals in Section 3. Section 4 illustrates the NSP based approach for feature clustering. Section 5 explains the approaches for selecting a desired number of features after performing feature clustering. In Section 6, we prove a key result on the equivalence between a k-size NSP of a coalitional game and minimum k-cut of an appropriately constructed graph. Based on this result, we develop a hierarchical approach for feature clustering when the feature set size is large. Section 7 constitutes the experimental results. We conclude the paper in Section 8.

2 Related Work

Feature selection can be thought of as a dimensionality reduction technique. The problem of feature selection is to find a feature subset S with m features, which jointly have the largest dependency on the target class. Feature selection methods can be of two types: *filter* methods and *wrapper* methods [12]. Filter methods select the relevant features by ranking all the features using some measure. On the other hand, the wrapper methods treat the induction algorithm as a black box and interact with it to assess the usefulness of a subset of features.

Different measures have been suggested in the literature for ranking the features. See Guyon and Elisseeff [9] and the references therein. Some commonly used measures are the absolute value of the Pearson correlation coefficient and mutual information [17]. Peng et al [17] studied the feature selection problem using maximum statistical dependency criterion based on the mutual information. Particularly relevant to our work is the approach suggested by Cohen et al [3], which poses the problem of feature selection as a cooperative game problem.

Cohen et al [3] proposed to use Shapley value (a solution concept for cooperative games) for ranking the features. In this approach, each feature is treated as a player of a game and the idea is to evaluate the marginal contribution of every feature to the classification accuracy by using Shapley value and then eliminate those features whose marginal contribution is less than a certain threshold. Since the calculation of the Shapley value involves summing over all possible permutations of the features, it becomes impractical to determine the Shapley value if the feature set size is large. It was therefore proposed to use multi perturbation approach of approximating the Shapley value (proposed by Keinan et al [11]).

The main drawback of this approach is that it is computationally expensive as different classifiers need to be trained using different feature subsets.

3 Coalitional Games Preliminaries

We start with definitions of a few basic concepts in coalitional game theory which would serve as building blocks for the rest of the paper. These concepts are fairly standard in the game theory literature and can be found in [8].

Definition 1 (Coalitional Games). *An n-person coalitional game (under pure hedonic setting) is a pair $(N, u(\cdot))$, where $N = \{x_1, x_2, \ldots, x_n\}$ is the set of players and $u(\cdot) = (u_1(\cdot), u_2(\cdot), \ldots, u_n(\cdot))$ is the profile of players' utility functions. Utility function $u_i(\cdot)$ is defined over the set of coalitions $\{C \subset N | i \in C\}$ and $u_i(C)$ signifies the utility of the player x_i if he decides to join the coalition C.*

The pure hedonic setting characterizes a special class of coalitional games where the utility of a player due to joining a coalition depends only on the composition of the coalition. The term *pure hedonic setting* was coined by [6]. The examples of hedonic setting include the formation of social clubs, political parties, and faculties at a university. In the rest of this paper, we will be working with coalitional games under the pure hedonic setting, which we refer to as hedonic games.

The key question addressed by coalitional game theory is that for a given coalitional game, what coalitional structure would emerge if the players play the game. A coalition structure, denoted by $\mathscr{C} = \{C_1, C_2, \ldots, C_k\}$, is a partitioning of the players into k disjoint sets $C_1, \ldots, C_k$. There are several stability concepts proposed in the coalitional game theory [2]. In this paper we focus only on *Nash Stability.*

Definition 2 (Nash Stable Partition (NSP)). *Given a hedonic game $(N, u(\cdot))$, a partition $\mathscr{C} = \{C_1, C_2, \ldots, C_k\}$ is Nash stable if for every player i, we have $u_i(C_{\mathscr{C}}(i)) \geq u_i(C_j \cup \{i\})$ $\forall C_j \in \mathscr{C} \cup \{\varnothing\}$, where $C_{\mathscr{C}}(i)$ denotes the set $C_j \in \mathscr{C}$ such that $i \in C_j$. In simple words, a partition $\mathscr{C}$ is an NSP if no player can benefit from switching his current coalition $C_{\mathscr{C}}(i)$ given that all the other players are sticking to the coalitions suggested by the partition $\mathscr{C}$. The NSP where $\mathscr{C} = \{N\}$ is a trivial NSP and any other NSP is called non trivial NSP. A non trivial NSP $\mathscr{C} = \{C_1, C_2, \ldots, C_k\}$ is called a k-size NSP (or k-NSP for short).*

The obvious question that arises next is that whether an NSP always exists and if not then under what conditions it exists. In general, it is possible that a given hedonic game may not have any NSP. However, for some classes of hedonic games, existence of an NSP is guaranteed; for example, the games with *additively separable (AS) and symmetric preferences* (defined next).

Definition 3 (AS and Symmetric Preferences). *A player i's preferences are additively separable if there exists a function $v_i : N \rightarrow R$ such that $\forall$ $C \subseteq N$ for which $C \ni i$, we have $u_i(C) = \sum_{j \in C} v_i(j)$. Without loss of generality, we* *can set $v_i(i) = 0$ which would imply that $u_i(C) = u_i(C \cup \{i\})\ \forall C \subseteq N$. Thus, we can say that any additively separable preferences can be equivalently represented by an $n \times n$ matrix $v = [v_i(j)]$. Further, we say that the AS preferences satisfy symmetry iff the matrix v is symmetric, that is $v_i(j) = v_j(i) = v_{ij}\ \forall i, j$.*

It turns out that the problem of deciding whether an NSP exists in a pure hedonic game with the AS preferences is NP-complete [1], [16]. However, Bogomolnaia and Jackson [2] have shown that an NSP exists in every hedonic game having the AS and *symmetric* preferences. It is worth mentioning that Bogomolnaia and Jackson proved the existence of an NSP. But their result does not specify anything regarding the structure of the NSP, for example whether the NSP would be trivial or nontrivial, whether there is a unique NSP or many, what is the size of such NSP(s), etc. However, Olsen [16] has shown that the problem of deciding whether a non-trivial NSP exists for the AS hedonic games with non-negative and symmetric preferences is NP-complete. In rest of the paper, we will be working under the hedonic setting with the AS and symmetric preference relation and hence we do not have to worry about the existence of an NSP (although it could be a trivial NSP).

In what follows, we extend the notion of an NSP to an approximate NSP which we prefer to call as ϵ-Regret NSP. The motivation behind this notion comes from the facts that (1) a hedonic game with the AS and symmetric preferences may not have a nontrivial NSP (i.e. a k-NSP with $k > 1$) but for any value of $1 \leq k \leq n$, there always exists an $\epsilon \geq 0$ such that the game has an ϵ-Regret k-NSP, and moreover, (2) checking for the existence of a nontrivial NSP is NP-complete (under non-negative preferences) [16].

Definition 4 (ϵ-Regret k-NSP). *A given k-size partition of the players in a hedonic game is said to be ϵ-regret k-NSP if no player can increase his utility by more than ϵ by switching his current coalition given that no other player is switching his current coalition.*

3.1 NSP Computation via Integer Linear Program

In what follows, we give an integer linear program (ILP) whose solution would give us an NSP. This ILP can be directly obtained from the existence proof of Bogomolnaia and Jackson [2].

Theorem 1. *Consider an n-person hedonic game having the AS and symmetric preferences given by a matrix $v = [v_{ij}]$. Let $\mathscr{C}^*$ be a partition of the set N of players. If $\mathscr{C}^*$ is a solution of the following ILP then $\mathscr{C}^*$ is an NSP of this game. In this ILP we have $v(C) = \sum_{i,j \mid i,j \in C} v_{ij}$.*

$$\begin{aligned} &\textit{maximize} \sum_{C \subset N} \alpha(C) v(C) \\ &\textit{subject to} \sum_{C \subset N \mid i \in C} \alpha(C) = 1\ \forall\ i \in N; \alpha(C) \in \{0,1\}\ \forall C \subset N \end{aligned} \tag{1}$$

In the above ILP, $\alpha(C)$ is an indicator variable that decides whether the coalition C is a part of the partition or not. The constraints basically ensure a partition of the players. The objective function is nothing but the sum of values of all the coalitions in a partition, which we call as *partition's value*. Thus, we can say that any value maximizing partition is always an NSP. Note that the above theorem provides a sufficiency condition for an NSP. However, it is not necessary that if a non trivial partition $\mathscr{C}$ of N is an NSP then that would also be a solution of the above ILP. For example, consider a 4×4 symmetric matrix v having $v_{ii} = 0 \; \forall i$; $v_{12} = 10$; $v_{13} = v_{14} = 1$; $v_{23} = v_{24} = 1$; $v_{34} = 5$. It is easy to see that for this matrix we have $\mathscr{C} = \{\{1,2\},\{3,4\}\}$ as an NSP but this is not the optimal solution of the ILP. The optimal solution of the ILP is a trivial NSP. This observation can be generalized by saying that if matrix $v = [v_{ij}]$ is nonnegative then the above ILP will give only trivial partition as the solution.

Note that solving an ILP is in general hard and quite impractical (especially for a large number of variables). Hence, one can work with an LP relaxation ($0 \leq \alpha(C) \leq 1$) of the above ILP and get an approximate solution. In fact, in our feature clustering approach (proposed in the next section), we use an LP relaxation of this ILP.

4 Feature Clustering via Nash Stable Partition

In this section, we show that the feature clustering problem can be effectively posed as a Nash stable partitioning problem. For this, let us consider the binary classification problem $\{x_l, y_l\}_{l=1}^{m}$ where we have m training examples and the input vector x_l is specified in the form of n real valued features $\{f_1, f_2, \ldots, f_n\}$ and $y_l \in \{-1, +1\}$. Let ρ_{ij} be the estimate of the Pearson correlation coefficient between feature f_i and feature f_j. Similarly, ρ_{iy} is the Pearson correlation coefficient between the feature f_i and the class label y. Now let us set up an analogy between a feature cluster and an NSP as follows. View each feature as a player in the game and define a payoff function $v_i(j) = |\rho_{ij}| \; \forall i, j \in N$. Define $v_i(i) = 0 \; \forall \; i$. It is easy to see that $v_i(j) = v_j(i) = v_{ij} = |\rho_{ij}|$. Assume that each feature f_i has a preference relation $\succeq_i$ over the feature subsets such that $\forall C_1, C_2 \ni f_i$, we have $C_1 \succeq_i C_2 \Leftrightarrow \sum_{j \in C_1} v_i(j) \geq \sum_{j \in C_2} v_i(j)$. If the features are allowed to form the coalitions then every feature would tend to join a feature group which maximizes its payoff function. It is easy to see that for the chosen function $v_{ij} = |\rho_{ij}|$, substitutable features would tend to group together. The above situation can be viewed as a coalitional game with the AS and symmetric preferences for which an NSP is a reasonable solution concept to predict the final coalitional structure. Therefore, if we use an NSP of the above game as clusters of the features then it would have the property that the features are stable in their own clusters and don't want to move across the clusters (which is a desired property of any clustering scheme). Thus, we can say that any NSP of the above game would be a reasonable clustering of the features where each cluster would contain substitutable features and the features across clusters would be complementary to each other. A Nash stable partition of the above game can be obtained by solving the ILP given in (1). Recall, if v_{ij} is nonnegative for all i and j then ILP does not have a nontrivial solution and moreover computing a non-trivial NSP is NP-complete due to the result of Olsen [16]. In order to tackle this situation, we can modify the payoff functions slightly as follows: $v_{ij} = |\rho_{ij}| - \beta$ where $0 < \beta < 1$. β can be viewed as a parameter which decides a threshold (in an implicit manner) such that any two features having v_{ij} values higher (lower) than the threshold would qualify as substitutable (complementary) features. With this interpretation, it is easy to see that as we increase β, the threshold increases and hence we get more and more fine grained clusters of the substitutable features.

It is interesting to note that the function $v_{ij} = |\rho_{ij}| - \beta$ is not the only payoff function but there exist many other choices and depending upon the function we decide to choose, we may either get substitutable features grouped together or complementary features grouped together. For example, if we use $v_{ij} = |\rho_{iy}| + |\rho_{jy}| - |\rho_{ij}| - \beta$ where $0 < \beta < 2$ then the complementary features would tend to group together and each cluster will contain the relevant and complementary features in its own. Note that the maximum possible value of the function $|\rho_{iy}| + |\rho_{jy}| - |\rho_{ij}|$ is 2 and having $\beta > 2$ would make all $v_{ij},\ i \neq j$ negative. Because $v_{ii} = 0\ \forall i$, this would result in a solution where each single feature would form its own cluster. This is the reason why we have restricted the value of β upto 2. One can also work with the function $v_{ij} = |\rho_{iy}| + |\rho_{jy}| + |\rho_{ij}| - \beta$ where $0 < \beta < 3$. This will again give the substitutable features grouped together. In each one of these functions, one can replace $|\rho_{ij}|$ with m_{ij} and $|\rho_{iy}|$with m_{iy} where m_{ij} is the estimate of the mutual information between the features f_i and f_j. Below, we suggest a few v_{ij} functions along with their clustering nature.
Substitutable Features: $(|\rho_{ij}| - \beta)$, $(|\rho_{iy}| + |\rho_{jy}| + |\rho_{ij}| - \beta)$, $(m_{ij} - \beta)$, and $(m_{iy} + m_{jy} + m_{ij} - \beta)$.
Complementary Features: $(|\rho_{iy}| + |\rho_{jy}| - |\rho_{ij}| - \beta)$ and $(m_{iy} + m_{jy} - m_{ij} - \beta)$.

5 Feature Selection Approaches

Here we discuss a few approaches for selecting a given number of features. In the first approach, we choose a payoff function that puts the substitutable features in one cluster. Next, we tune the parameter β in such a way that we obtain m feature clusters as a solution of the ILP (or relaxed ILP). Now we pick the most relevant feature from each cluster. The most relevant feature of a cluster can be chosen by several schemes such as maximum $|\rho_{iy}|$ or maximum m_{iy}. Note that in this approach, there is a risk of getting an irrelevant feature selected if all the features in that group have very low value of $|\rho_{iy}|$. Therefore, we propose an alternative approach next.

In the second approach, we again choose a payoff function that puts the substitutable features in one cluster. Next, we choose some value of the parameter β and obtain the feature clusters as the solution of ILP (or relaxed ILP). Note that, in this approach, we have no control over the number of clusters (because β is not tuned). Now, we perform some kind of sampling without replacement on these clusters. In particular, we compute a ranking score $r(C)$ for each cluster C in the partition and then pick the top cluster. From this top cluster, we pick the most relevant feature (using the criterion discussed in the previous approach) and then delete that feature from this cluster. Now we recompute the ranking score for this cluster and perform the feature selection step as before. We repeat this whole process until we obtain the desired number of features. A few suggested ranking functions are $r(C) = \frac{\sum_{i \in C} |\rho_{iy}|}{|C|}$; $r(C) = \frac{\sum_{i \in C} m_{iy}}{|C|}$. One can develop similar approaches by using a payoff function for grouping the complementary features together.

6 Handling of Large Feature Set Size

In section 4, we suggested that in most of the cases, LP relaxation of the ILP gives an approximate clustering based on NSP. However, it is easy to see that even the relaxed LP would have 2^n variables and hence running time of any LP solver is large even for as small a number of features as $n = 20$. In such a scenario, the ILP or relaxed ILP based approaches are not feasible for feature clustering. To tackle such situations, we propose a hierarchical feature clustering approach. This approach is based on an interesting result on the equivalence between a k-NSP of a coalitional game and minimum k-cut of an appropriately constructed graph (as we prove below).

6.1 Equivalence between a k-NSP and Minimum k-Cut

Consider an n-person hedonic game with the AS, symmetric, and non-negative preferences given by an $n \times n$ symmetric and non-negative matrix $v \geq 0$. This game can be represented by an undirected graph $G = (N, E)$ where N is the set of nodes which is the same as the set of players. We put an undirected edge between node i and j iff $v_{ij} = v_{ji} > 0$ and assign a weight v_{ij} to that edge.

Definition 5 (Minimum k-Cut). *A k-cut of a graph G is defined as any partitioning of the nodes into k $(1 < k \leq n)$ nonempty disjoint subsets, say $\{C_1, C_2, \ldots, C_k\}$. The capacity of such a k-cut is defined as the sum of the weights of all those edges whose end points are in different partitions and is denoted by $cap(C_1, C_2, \ldots, C_k)$. A minimum k-cut of this graph is a k-cut whose capacity is minimum across all the possible k-cuts of the graph. For any given feasible value of k, there could be multiple minimum k-cuts and the capacities of all those cuts would be the same and we denote that by $cap^*(k)$.*

Definition 6 (Support Size). *The support size s of a k-cut $\{C_1, C_2, \ldots, C_k\}$ is defined as follows: $s = \min_{i=1,\ldots,k} |C_i|$.*

Theorem 2 (k-NSP and Minimum k-Cut). *Let G be a graph representation of a hedonic game with the AS and symmetric preferences given by a matrix $v \geq 0$. Then the following holds true: $\boxed{\text{minimum } k\text{-cut of } G \text{ having } s > 1} \Rightarrow \boxed{k\text{-NSP}}$.*

Proof: The proof is by contradiction. If possible, let $\{C_1, C_2, \ldots, C_k\}$ be a minimum k-cut with support $s > 1$ and it is not a k-NSP. This would mean that there exists some player $i \in N$ who would gain by switching to some other coalition given that no other player is switching. Let us assume that $i \in C_t$ for some $t \in \{1, 2, \ldots, k\}$. Let $u_i(C_z) > u_i(C_t)$ for some $z \neq t$ and hence player i would prefer to switch to the coalition C_z from his current coalition C_t.[5] Because of the AS and symmetry preferences, we have $u_i(C) = \sum_{j \in C} v_{ij}$ for any coalition C, and hence the cut capacity of this newly formed k-cut after the switching is given by $cap(\text{new cut}) = cap(C_1, \ldots, C_k) + u_i(C_t) - u_i(C_z)$. Because $u_i(C_z) > u_i(C_t)$, we have $cap(\text{new cut}) < cap(C_1, C_2, \ldots, C_k)$ which is a contradiction to the assumption that $\{C_1, C_2, \ldots, C_k\}$ is a minimum k-cut.[6] *(Q.E.D.)*

In the above proof, $s > 1$ is required to ensure that even after node i switches, the resulting partition is a k-way cut. Theorem 2 gives a sufficiency condition for existence of a k-NSP and hence becomes useful in the case when computing a minimum k-cut of a graph is an easy problem. However, it is a well known fact that computing a minimum k-cut of a graph is NP-hard for $k > 2$ [20]. Therefore, this theorem would not help much for computing a k-NSP of a given game if $k > 2$. To handle that case, it is useful to define an approximate minimum cut in the form of ϵ-Regret k-cut of a graph (defined below). In Theorem 3, we show that there exists a connection between an ϵ-Regret k-cut of a graph and an ϵ-Regret k-NSP. The proof of this theorem follows the similar line of arguments as proof of the Theorem 2. Hence, we skip the proof.

Definition 7 (ϵ-Regret k-Cut). *Given a graph G, a k-cut $\{C_1, C_2, \ldots, C_k\}$ is said to be an ϵ-Regret k-cut iff it satisfies the following condition.*

$$cap^*(k) \leq cap(C_1, C_2, \ldots, C_k) \leq cap^*(k) + \epsilon \tag{2}$$

where $cap^*(k)$ is the capacity of the minimum k-cut.

Theorem 3. $\boxed{\epsilon\text{-Regret } k\text{-cut having } s > 1} \Rightarrow \boxed{\epsilon\text{-Regret } k\text{-NSP}}$

6.2 Hierarchical Feature Clustering

From Theorem 3, it is apparent that any scheme that efficiently computes an ϵ-Regret k-cut of a graph would be an efficient scheme for computing the ϵ-Regret k-NSP also. Therefore, we propose a simple hierarchical scheme (without any theoretical bounds on the value of ϵ) for computing an ϵ-Regret k-cut and hence an ϵ-Regret k-NSP. Note that when v has only positive entries (and under some cases with negative entries also), it becomes a polynomial time solvable problem to compute a minimum 2-cut [19]. Hence, our approach works by computing minimum 2-cuts of a graph in hierarchical manner.

In this approach, we begin with the whole feature set and just compute a minimum 2-cut of the underlying graph. Then we recursively compute the

[5] Note that i can't gain by isolating itself (i.e. $C_z = \emptyset$) because in that case $u_i(C_z) = v_{ii} = 0$ and $0 > u_i(C_t)$ which is a contradiction because v is non-negative.

[6] A similar result was proved by [7] in the context of web communities for $k = 2$.

minimum 2-cut of each partition obtained in the previous step. This gives us a binary tree structure (not necessarily balanced) where each node is the set of features and two children of a node correspond to a minimum 2-cut of that node. We stop splitting a node further if the number of features in that node are less than some threshold value. All such nodes would form the leaf nodes. Finally, we take all the leaf nodes of such a tree as feature clusters and then apply any one of the feature selection approaches discussed earlier.

Note that at every step of the above scheme, one can use an exact algorithm for computing a minimum 2-cut (for example, $O(mn + n^2 log(n))$ algorithm by Stoer and Wagner [19] where m is the number of edges). However, it is possible to have $m = O(n^2)$ in which case finding an exact solution becomes expensive $(O(n^3))$. In addition to this, if a graph has multiple minimum 2-cuts then we would be interested in focusing on those cuts for which support size is more than 1. Therefore, in practice one may prefer a fast randomized algorithm (although approximate) as opposed to an exact algorithm because we can run it several times and generate various minimum 2-cut solutions and then pick the best solution. Some of the methods that are fast and can generate multiple solutions include Karger and Stein's [10] randomized mincut algorithm ($O(n^2 log^3 n)$ for a cut) and spectral clustering (SC) algorithms ($O(n^3)$ for a cut). We conducted experiments with a SC algorithm for ease of implementation and to get a feel of the performance achievable using an approximate method. Although the SC algorithm has same $O(n^3)$ complexity as the Stoer and Wagner's algorithm but we found that SC algorithm was significantly faster in our experiments. The SC algorithm which we use here is as follows [13][5]: (1) Construct a diagonal matrix D of the size same as the matrix v in such a way that each diagonal entry of D is the sum of the entries of the corresponding row of the matrix v; (2) Compute the graph Laplacian matrix $L = D - v$; (3) Find the two eigenvectors corresponding to the two lowest eigenvalues of the matrix L; (4) Apply 2-means clustering on these two eigenvectors. These clusters would correspond to an approximate minimum 2-cut. It is important to note that in this approximate algorithm, the 2-means clustering algorithm involves a random initialization of the clusters. Depending on how the initialization is done, we may get different solutions. Among the different solutions, the final solution can be picked by using an appropriate cluster quality measure.

7 Experiments

We illustrate our approach by conducting several experiments on the feature selection problems for one synthetic dataset and two real world datasets - *Splice* and *Arrhythmia* (details given later), which are binary classification problems.

Synthetic Datasets (No. of Features = 20, Training Set Size = 1000, Test Set Size = 300): We generated synthetic datasets corresponding to 20-dimensional zero mean correlated Gaussian random variables of 15 different co-variance matrices. Each of these 15 datasets consisted 1000 training samples, denoted by X_{tr}, and 300 testing samples, denoted by X_{ts}. In our experiment, we generated a coefficient vector w of size 20 for a linear classifier and used it to obtain the class labels for the training data as follows: $Y_{tr} = sign(w' \cdot X_{tr})$. In the same way, we generated the class labels for the test data. The vector w was generated as 20 i.i.d. uniform random variables over the interval $[-1, 1]$. Next we computed the correlation matrix $\rho = [\rho_{ij}]$ for the training data. We set the payoff function $v_{ij} = |\rho_{ij}| - \beta$ and then varied β from 0 to 1 so that we get different size NSPs by solving the corresponding relaxed LPs for ILP (1). For each of these NSPs, we used $|\rho_{iy}|$ to pick the relevant features and trained a linear least squares classifier on the training data using these selected features. We computed the accuracy of the test data (with the selected features) on this new trained classifier. We performed this experiment 1000 times by changing the vector w. We computed the average accuracy over these realizations. We repeated the same experiment for 15 different covariance matrices. The results of our experiments are summarized in Figure 1; we have plotted the variation of average accuracy for 15 different covariance matrices in the form of blue colored box plots (for our method). We have also plotted a red colored box plot along with each blue box plot. The red box plot corresponds to the variation of the average accuracy when the features were picked from the set of features ranked in decreasing order according to $|\rho_{iy}|$. We conducted statistical significance test using Wilcoxon sign rank test at the significance level of 0.05 to compare the two schemes of feature selection and found that for the feature set sizes of 3 - 6, the game theoretic based feature clustering approach (followed by feature selection) yielded better performance than selecting the features using $|\rho_{iy}|$ values.

Splice Datasets (No. of Features = 60, Training Set Size = 1000, Test Set Size = 2175): The splice datasets are taken from `http://theoval.cmp.uea.ac.uk/~gcc/matlab/default.html`. In this experiment our goal was to test the *relevant feature identification capability* of our method. Because the number of features is large, we used the hierarchical clustering approach (described in Section 6) for obtaining the clusters of the features and then we picked top 7 features using the second approach for feature selection (proposed in Section 5) by making use of the function $r(S) = \frac{\sum_{i \in S} |\rho_{iy}|}{|S|}$. We repeated this experiment 1000 times with random initialization for 2-means clustering. We found that most of the times, 6 out of the selected 7 features belong to the following set: $\{f_{28}, f_{29}, f_{30}, f_{31}, f_{32}, f_{33}, f_{34}\}$. It is interesting to note that this feature set is almost the same as the one identified as the relevant features (as they are all located in the vicinity of the splice junction that lies between 30 and 31) for classification by Meilă and Jordan [14]. This demonstrates the effectiveness of the proposed method on real world problems.

Arrhythmia Dataset (No. of Features = 246, Training Set Size =280, Test Set Size = 140): The Arrhythmia dataset can be obtained from the UCI repository. We used a version of this dataset that was modified by Perkins et al [18] and was also used by Cohen et al [3]. For this dataset, we again performed the hierarchical feature clustering followed by the feature selection (varying size from 10 to 30) in the same way as we did for the Splice dataset case. We repeated the whole process for 100 different initial conditions as was done in the Splice

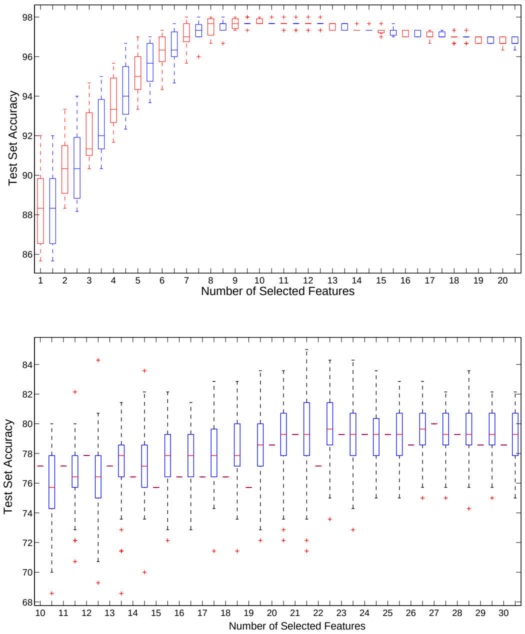

Fig. 1. Results for Synthetic and Arrhythmia Datasets

datasets case. For each case, we trained a simple linear least squares classifier on the selected features and then recomputed the accuracy of the test set. We have plotted the boxplot for the variation in accuracy in Figure 1. Along with each box plot, we have indicated the test set accuracy (in the form of red dash) when the same number of features are selected from the set of features ranked in decreasing order according to $|\rho_{iy}|$ and the linear least squares classifier is trained on those features. We see that the median of the test set accuracy with the feature clustering is higher than 75% for each case (which was the reported performance of Perkins et al [18]). Specifically, for 21 features, we see that the median of the performance is 79% whereas the highest performance reaches beyond 85% (which is higher than 84.2% reported by Cohen et al [3]).

8 Discussion

Although the proposed approach is complete in its own, there are several avenues for further investigations. For example, all the experiments were conducted using the payoff functions that put substitutable features in one cluster. One can experiment with other payoff functions which would drive complementary features in one cluster. In the case of multiple NSPs, we assumed that it is fine to choose any one of them. However, it is worth investigating the quality of other NSPs.

References

1. C. Ballester. NP-completeness in hedonic games. *Games and Economic Behavior*, 49(1):1–30, 2004.
2. A. Bogomolnaia and M. O. Jackson. The stability of hedonic coalition structures. *Games and Economic Behavior*, 38:201–230, 2002.
3. S. Cohen, G. Dror, and E. Ruppin. Feature selection via coalitional game theory. *Neural Computation*, 19:1939–1961, 2007.
4. I. Dhillon, Y. Guan, and B. Kulis. A unified view of kernel k-means, spectral clustering, and graph partitioning. Technical report, Univ. of Texas, Austin, 2005.
5. C. Ding. A tutorial on spectral clustering. *Talk presented at ICML. (Slides available at http://crd.lbl.gov/∼cding/Spectral/).*
6. J. Drèze and J. Greenberg. Hedonic coalitions: Optimality and stability. *Econometrica*, 48:987–1003, 1980.
7. G. Flake, R. Tarjan, and K. Tsioutsiouliklis. Graph clustering and minimum cut trees. *Internet Mathematics*, 1(4):385–408, 2004.
8. J. Greenberg. Coalition structures. In R.J. Aumann and Hart S., editors, *Handbook of Game Theory, Volume 2*. Elsevier Science, B.V., 1994.
9. I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.
10. D. R. Karger and C. Stein. An $\widetilde{O}(n^2)$ algorithm for minimum cuts. In *STOC*, 1993.
11. A Keinan, B. Sandbank, C. Hilgetag, I. Meilijson, and E. Ruppin. Axiomatic scalable neurocontroller analysis via shapley value. *Artificial Life*, 12, 2006.
12. R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 77:273–324, 1997.
13. U. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4), 2007.
14. M. Meilă and M. I. Jordan. Learning with mixtures of trees. *Journal of Machine Learning Research*, 1:1–48, 2000.
15. R. B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, Cambridge, Massachusetts, 1997.
16. M. Olsen. Nash stability in additively separable hedonic games is NP-hard. In *CiE '07: Proc. of 3rd Conference on Computability in Europe*, 2007.
17. H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. on PAMI*, 27(8):1226–1237, 2005.
18. S. Perkins, K. Lacker, and J. Theiler. Grafting: Fast, incremental feature selection by gradient descent in function space. *JMLR*, 3:1333–1356, 2003.
19. M. Stoer and F. Wagner. A simple min-cut algorithm. *J. of the ACM*, 44(4), 1997.
20. V. V. Vazirani. *Approximation Algorithms*. Springer, 2004.