

Approximating the Shapley Value via Multi-Issue Decompositions

Hossein Azari Soufiani
SEAS Harvard
Cambridge, MA, US
azari@fas.harvard.edu

David M. Chickering
Microsoft Research
Redmond, WA, US
dmax@microsoft.com

Denis X. Charles
Microsoft Research
Redmond, WA, US
cdx@microsoft.com

David C. Parkes
SEAS Harvard
Cambridge, MA, US
parkes@eecs.harvard.edu

ABSTRACT

The Shapley value provides a fair method for the division of value in coalitional games. Motivated by the application of crowdsourcing for the collection of suitable labels and features for regression and classification tasks, we develop a method to approximate the Shapley value by identifying a suitable decomposition into multiple issues, with the decomposition computed by applying a graph partitioning to a pairwise similarity graph induced by the coalitional value function. The method is significantly faster and more accurate than existing random-sampling based methods on both synthetic data and data representing user contributions in a real world application of crowdsourcing to elicit labels and features for classification.

Keywords

Coalitional game theory, Shapley value, Machine learning, Crowdsourcing

1. INTRODUCTION

There is a recent trend to use crowdsourcing tools for machine learning, such as crowd prediction [1] and machine-learning markets [20]. An example is to use crowd workers to label whether webpages contain recipes or not. These labels, in conjunction with features of the webpage (e.g., the presence or absence of specific words such as “ingredient”), are used to train a classifier. An extension can allow workers to recommend certain words (e.g., “recipe”, “teaspoon” etc.) as features, to be used in determining whether or not a webpage contains the recipe.

When considering methods to reward participants for the value that they contribute to the system, it is useful to model this as a cooperative problem, and understand what would be a fair division of value, that is, a division that reflects the contributions of individuals to the system. Indeed, the performance of such a system may not be a simple function of the inputs (e.g., labels, features) from individual partic-

ipants. Rather, the performance will likely depend on the interaction between the various inputs received from different participants. For example, it may not be useful for two participants to suggest the same feature or to provide duplicate labels. In these cases, we would like the allocation of value to reflect the effective contribution of each participant, considering the way it interacts with other contributions.

Two questions naturally arise: (1) what is a good approach for assigning value to participants (e.g., in the form of payments), and (2) what is the corresponding computational complexity of the approach?

In this paper, we provide answers to both questions using coalitional game theory for modeling the multi-agent system; this approach has been used, for instance, to address the problem of feature selection in Cohen et al. [4]. A well-studied solution concept for fair division in coalitional games is the *Shapley value* [21]. The Shapley value is the unique value function that satisfies a set of easily justifiable axioms. A challenge with the Shapley value, however, is that it can be computationally hard to compute. Our focus is on the development of an efficient method to approximate the Shapley value, and in particular to take advantage of structure that exists in the value provided by the contributions of different users in an application of crowdsourcing to elicit labels and features used to train a classifier.

1.1 Our Contributions

We start by introducing a general *similarity* measure among sets of agents for a given coalitional value function and, as is done by Jeong and Shoham [11], we show how to represent exactly the value function in terms of this measure through inclusion-exclusion arguments.

Our main contribution is to formalize the problem of approximating the Shapley value in a graph-theoretic way by focusing on pairwise similarity between individual participants. In particular, we construct a *pairwise similarity graph* with agents as vertices, and undirected edges representing the pairwise agent similarity. We use this graph to identify a good decomposition of agents into groups, and we compute the Shapley value exactly for the coalitional game associated with each group. Following the phrasing introduced by Conitzer and Sandholm [5], who discussed exact multi-issue decompositions, we view this as an approximate *multi-issue decomposition*. In particular, we can adopt various algorithms to find a good partition of agents; our experimen-

Appears in: *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014)*, Lomuscio, Scerri, Bazzan, Huhns (eds.), May, 5–9, 2014, Paris, France.

Copyright © 2014, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

tal results adopt a spectral clustering approach. When the coalitional value function satisfies *strong sub-additivity* we have a simple upper bound for the error introduced by a particular partition of agents, and the computational run time of the approach can be tuned by deciding on the maximal size of a coalitional game that is acceptable for computing exact Shapley values.

We also present empirical results that demonstrate the effectiveness of our approximations, using both synthetic and real-world data from the domain of crowdsourcing for solving classification problems.

1.2 Related Work

The Shapley value is commonly used in cooperative settings [14, 9, 19, 10, 17, 9] to evaluate participants. There are applications of the Shapley value in feature selection; e.g., [4] where they use a randomized sub-sampling method for approximating the Shapley value. Sometimes a domain has specific structure that allows for the Shapley value to be computed efficiently; e.g., Ma et al. [14] compute Shapley value exactly for ISP games and Bachrach et al. [2] propose an approximation of Shapley value for Reliability Games. Fatima et al. [7] propose an approximation for Shapley value in voting games.

Conitzer and Sandholm [5] propose an exact multi-issue decomposition that, when available, can reduce the complexity of computation. The idea is that the interaction between agents may be limited to smaller groups of agents that all care about the same issue. (E.g., in our setting they might contribute to classifiers of cats but not dogs.) Jeong and Shoham [11] propose an orthogonal decomposition approach that generalizes the weighted-graph model of Deng and Papadimitriou [6] to allow for value that is generated in a way that depends on more than pairs of agents being present in a system. Grabisch et al. [8] proposes an interaction term among agents in a cooperative setting. Neither of these works consider an approximation scheme, with decompositions or without, for computing Shapley values.

As far as we know, the only general approach for approximating Shapley value is based on taking random samples of different orders on agents; e.g., [3]. This approach is problematic, however, because of the very large number of possible orders. In contrast, our approach identifies an approximate decomposition of the game into a set of smaller coalitional games, keeping agents together (for the purpose of Shapley value computation) where it matters for achieving good accuracy.

1.3 Outline

In the next section (§2) we give formal definitions for the necessary terms from coalitional game theory. In §3 we define the K-group similarity metric that forms the basis of the decomposition and provides the exact form of the decomposition. In §4 we give the proof of our main theorem characterizing the approximation obtained by partitioning the agents. Following this idea we propose a pairwise similarity graph in §5, and use min-cuts in this graph to provide partitions for our approximation. We illustrate exact forms for the Shapley value based on our decomposition using two examples from machine learning, and introduce some details of the crowd-learning setting in §6. Finally, in §7 we present experiments with synthetic and real-world data.

2. PRELIMINARY DEFINITIONS

A coalitional game is defined for a set of agents N and a coalitional value function $v : 2^N \rightarrow \mathbb{R}$, which defines the total value that can be achieved when a coalition of agents $S \subseteq N$ work together. If the agents are being used to train a classifier, for example, the coalitional value function could be any of the standard metrics (e.g., precision, recall, AUC, F1-score, etc.) used to measure the performance of the classifier trained using these labels. We call the coalition of all agents (i.e., $S = N$) the *grand coalition*.

2.1 Shapley Value

Suppose we want to design a reward system $\psi_i(v)$ for a coalitional game defined by a set of agents N with coalitional value function v . We start with the following desirable properties of the evaluation system:

- **Efficiency:** The total sum of reward is equal to the coalitional value of the grand coalition. $\sum_i \psi_i(v) = v(N)$.
- **Symmetry:** For any coalitional value function v , if for all S that $i, j \notin S$ we have $v(S \cup \{i\}) = v(S \cup \{j\})$, then $\psi_i(v) = \psi_j(v)$.
- **Dummy Player:** For any coalitional value function v , if for all S we have $v(S \cup \{i\}) = v(S) + v(\{i\})$, then $\psi_i(v) = v(\{i\})$.
- **Strong Positivity:** For any agent i and for two games v and w , if for all $S \in N$ we have $v(S \cup \{i\}) - v(S) \leq w(S \cup \{i\}) - w(S)$ then $\psi_i(v) \leq \psi_i(w)$.

Given the Efficiency property, such an evaluation system can be considered a way to distribute the total value generated by the grand coalition among the participants; the first three properties seem reasonable at face value.

The Strong Positivity property posits a meaningful relationship between performance and the value share assigned to an agent (or payment): an agent who provides relatively higher marginal value for a game will receive a corresponding higher reward.

THEOREM 1. (Uniqueness of Shapley [16]) *The unique reward system that satisfies Efficiency, Symmetry, Dummy Player and Strong Positivity is the Shapley value:*

$$Sh_v(N, i) \equiv \frac{1}{|N|!} \sum_{S \subseteq N} |S|!(|N| - |S| - 1)!(v(S \cup \{i\}) - v(S))$$

A fifth property will prove useful:

- **Additivity:** For any agent i and for two coalitional value functions v and w , if we define the addition of games as $(v + w)(S) = v(S) + w(S)$ then we have $\psi_i(v + w) = \psi_i(v) + \psi_i(w)$.

The following is well known:

PROPOSITION 1. (Additivity of Shapley) *The Shapley Value satisfies the Additivity property.*

Indeed, the Shapley value is also unique amongst methods for value division that satisfy Efficiency, Symmetry, Dummy and Additivity. One way to understand additivity is that the Shapley value for an agent is the expectation, taken with respect to the uniform distribution on ordered coalitions, of the value increase brought by that agent to a coalition. Since it is an expectation, the Shapley value is linear in the value function.

3. K-GROUP SIMILARITY METRIC

In this section, we define a similarity metric among two or more sets of agents. In addition to leading to an exact representation of any coalitional value function, the special case of pairwise similarity on agents will be used in Section 4 for our main results.

DEFINITION 1 (2-GROUP SIMILARITY). *For a coalitional value function v , the 2-group similarity metric on sets of agents, $S_1, S_2 \subseteq N$, is:*

$$\mathcal{M}_2(S_1, S_2) = v(S_1) + v(S_2) - v(S_1 \cup S_2).$$

More generally, K -group similarity metric for K sets of agents, is:

$$\mathcal{M}_K(S_1, \dots, S_K) = \sum_{I \subseteq \{1, \dots, K\}} (-1)^{|I|+1} v(\cup_{i \in I} S_i).$$

For example, for $K = 3$, the similarity metric is,

$$\begin{aligned} \mathcal{M}_3(S_1, S_2, S_3) = & v(S_1) + v(S_2) + v(S_3) - v(S_1 \cup S_2) \\ & - v(S_1 \cup S_3) - v(S_2 \cup S_3) + v(S_1 \cup S_2 \cup S_3). \end{aligned}$$

3.1 An Exact Decomposition

In this section, we propose an exact decomposition of a coalitional value function, which leads by the additivity of the Shapley value to a characterization of the Shapley value in coalitional games.

LEMMA 1. *The following is an exact decomposition of any coalitional value function:*

$$v(S) = \sum_{K=1}^{|S|} (-1)^{K+1} Q_K(S) \quad (1)$$

Where, $Q_K(S) = 0$ for $|S| < K$ and,

$$Q_K(S) = \sum_{\{i_1, \dots, i_K\} \subseteq S} \mathcal{M}_K(\{i_1\}, \dots, \{i_K\}) = \sum_{I \subseteq S, |I|=K} \mathcal{M}_{|I|}(I)$$

and $\mathcal{M}_{|I|}(I)$ is a compact form of $\mathcal{M}_K(\{i_1\}, \dots, \{i_K\})$. Equivalently, we can write,

$$\begin{aligned} v(S) = & \sum_{i \in S} v(\{i\}) - \sum_{i \neq j \in S} \mathcal{M}_2(\{i\}, \{j\}) \\ & + \sum_{i \neq j \neq k \in S} \mathcal{M}_3(\{i\}, \{j\}, \{k\}) - \dots \end{aligned}$$

PROOF.

$$\begin{aligned} \sum_{K=1}^{|S|} (-1)^{K+1} Q_K(S) &= \sum_{K=1}^{|S|} (-1)^{K+1} \sum_{I \subseteq S, |I|=K} \mathcal{M}_K(\{i_1\}, \dots, \{i_K\}) \\ &= \sum_{I \subseteq S} (-1)^{|I|+1} \mathcal{M}_{|I|}(I) = \sum_{I \subseteq S} (-1)^{|I|+1} \sum_{J \subseteq I} (-1)^{|J|+1} v(\cup_{j \in J} \{j\}) \\ &= \sum_{I \subseteq S} \sum_{J \subseteq I} (-1)^{|I|+|J|} v(\cup_{j \in J} \{j\}) = v(\cup_{j \in \{1, 2, \dots, |S|\}} \{j\}) = v(S) \end{aligned}$$

The last step follows because if $|J| < |S|$ then there is an equal number of I 's with odd size and even size that contain J , and they cancel out in the summation. $\square$

By the additivity of the Shapley value, we immediately have:

$$Sh_v(N, i) = \sum_K (-1)^{K+1} Sh_{Q_K}(N, i).$$

Recognizing that the $\mathcal{M}_K(\{i\}, \{j_1\}, \dots, \{j_{K-1}\})$ term is a *positive-literal rule* in the language of MC-nets [11], we immediately have that:

$$Sh_{Q_K}(N, i) = \frac{1}{K} \sum_{\{j_1, \dots, j_{K-1}\} \subseteq N \setminus \{i\}} \mathcal{M}_K(\{i\}, \{j_1\}, \dots, \{j_{K-1}\}),$$

Intuitively, the value $\mathcal{M}_K(\{i\}, \{j_1\}, \dots, \{j_{K-1}\}) \in \mathbb{R}$ is only realized when a coalition S includes all of $\{i, j_1, \dots, j_{K-1}\}$ agents, and by the symmetry of the Shapley value this value must accrue evenly to the agents who participate. From this, we immediately have the following by additivity:

THEOREM 2. *The Shapley-value in any coalitional game can be exactly computed as:*

$$\begin{aligned} Sh_v(N, i) = & \sum_K \frac{(-1)^{K+1}}{K} \sum_{\{j_1, \dots, j_{K-1}\} \subseteq N \setminus \{i\}} \mathcal{M}_K(\{i\}, \{j_1\}, \dots, \{j_{K-1}\}). \end{aligned}$$

4. MULTI-ISSUE DECOMPOSITION AND APPROXIMATING SHAPLEY

In this section, we obtain an approximation to the Shapley value by forming a partition of agents; we view this partition as an approximate multi-issue decomposition. Whereas Conitzer and Sandholm [5] study exact multi-issue representations, we study approximate decompositions. In a sense, by forming a partition we identify implicit issues around which to group agents for the purpose of approximating the Shapley value.

In order to obtain a bound on the quality of the approximation we will need to assume a subadditivity structure to the coalitional game. The algorithmic approach is well defined without these assumptions and we will see in the experimental section that we obtain a very good approximation even when the property does not quite hold.

We start by defining the following properties for the coalitional value function.

DEFINITION 2. *For a valuation function $v : 2^N \rightarrow \mathbb{R}$,*

- v is sub-additive if for any $S_1, S_2 \subseteq N$ we have:

$$v(S_1 \cup S_2) \leq v(S_1) + v(S_2)$$

- v is strongly sub-additive if it is sub-additive and for any three sets $S_1, S_2, S_3 \subseteq N$, we have:

$$\begin{aligned} v(S_1 \cup S_2 \cup S_3) \geq & v(S_1 \cup S_2) + v(S_1 \cup S_3) + v(S_2 \cup S_3) \\ & - v(S_1) - v(S_2) - v(S_3) \end{aligned}$$

The above definitions can be seen as an imposition of the Benferroni inequalities on the value function [15].

For a sub-additive coalitional value function, we immediately have $\mathcal{M}_2(S_1, S_2) \geq 0$. Similarly, for a strongly sub-additive coalitional value function, we have $\mathcal{M}_3(S_1, S_2, S_3) \geq 0$.

DEFINITION 3. *The vector of coalitional value functions $(v_1, v_2, \dots, v_K)$, with each $v_i : 2^N \rightarrow \mathbb{R}$ is an ϵ -approximate decomposition for coalitional value function v , if:*

$$v(S) = \epsilon(S) + \sum_{k=1}^K v_k(S),$$

where $\epsilon(S) \in \mathbb{R}$ for every $S \subseteq N$.

The following is immediate from the additivity of the Shapley value:

LEMMA 2. For ϵ -approximate decomposition $v = \epsilon + \sum_k v_k$ we have:

$$Sh_v(N, i) = Sh_\epsilon(N, i) + \sum_k Sh_{v_k}(N, i)$$

DEFINITION 4. [5] Coalitional value function v_k concerns only $C_k \subseteq N$ if $v_k(S_1) = v_k(S_2)$ whenever $C_k \cap S_1 = C_k \cap S_2$. In this case v_k can be defined only over 2^{C_k} .

LEMMA 3. [5] If v_k concerns only $C_k \subseteq N$, then:

$$Sh_{v_k}(N, i) = Sh_{v_k}(C_k, i)$$

In the following lemma, we derive upper bounds for the 2-group similarity metric between two coalitions in terms of the 2-group similarity between the agents in the coalitions. This leads to the concept of a *pairwise similarity graph*, which is used for partitioning agents for the purpose of approximating the Shapley value.

LEMMA 4. For a strongly sub-additive coalitional valuation function v , we have:

$$\mathcal{M}_2(S_1, S_2) \leq \sum_{i \in S_1, j \in S_2} \mathcal{M}_2(\{i\}, \{j\}) \quad (2)$$

Where $\mathcal{M}_2$ is the 2-group similarity for the coalitional value function v .

PROOF. We use induction on the size of both S_1 and S_2 . For the base case of $|S_1| = 1$ and $|S_2| = 1$ we have:

$$\mathcal{M}_2(\{i\}, \{j\}) = \mathcal{M}_2(\{i\}, \{j\}).$$

Now, suppose we have $\mathcal{M}_2(S_1, S_2) \leq \sum_{i \in S_1, j \in S_2} \mathcal{M}_2(\{i\}, \{j\})$ for any $|S_1| \leq s_1$ and $|S_2| \leq s_2$, and due to symmetry, we only need to show (2) holds for $|S_1| = s_1 + 1$ and $|S_2| = s_2$. Suppose $k \in S_1$ then define $S_3 = S_1 \setminus \{k\}$, $|S_3| = s_1$, and from the definition of $\mathcal{M}_2$, we have:

$$\begin{aligned} \mathcal{M}_2(S_1, S_2) &= v(S_1) + v(S_2) - v(S_1 \cup S_2) \\ &= v(S_3 \cup \{k\}) + v(S_2) - v(S_3 \cup \{k\} \cup S_2). \end{aligned}$$

Using the following equality to expand the last term,

$$\begin{aligned} -\mathcal{M}_3(S_3, \{k\}, S_2) &= -v(S_3 \cup \{k\} \cup S_2) + v(S_3 \cup \{k\}) \\ &\quad + v(S_3 \cup S_2) + v(\{k\} \cup S_2) \\ &\quad - v(S_3) - v(S_2) - v(\{k\}), \end{aligned}$$

we get,

$$\begin{aligned} \mathcal{M}_2(S_1, S_2) &= v(S_3) + v(S_2) - v(S_3 \cup S_2) + v(\{k\}) + v(S_2) \\ &\quad - v(\{k\} \cup S_2) - \mathcal{M}_3(S_3, \{k\}, S_2) \\ &= \mathcal{M}_2(S_3, S_2) + \mathcal{M}_2(\{k\}, S_2) - \mathcal{M}_3(S_3, \{k\}, S_2). \end{aligned}$$

Given that $\mathcal{M}_3(S_3, \{k\}, S_2) \geq 0$ (due to strong sub-additivity), we have:

$$\begin{aligned} \mathcal{M}_2(S_1, S_2) &= \mathcal{M}_2(S_3, S_2) + \mathcal{M}_2(\{k\}, S_2) - \mathcal{M}_3(S_3, \{k\}, S_2) \\ &\leq \mathcal{M}_2(S_3, S_2) + \mathcal{M}_2(\{k\}, S_2) \\ &\leq \sum_{i \in S_3, j \in S_2} \mathcal{M}_2(\{i\}, \{j\}) + \sum_{j \in S_2} \mathcal{M}_2(\{k\}, \{j\}) \\ &= \sum_{i \in S_1, j \in S_2} \mathcal{M}_2(\{i\}, \{j\}). \end{aligned}$$

The induction hypothesis is used in deriving the inequality in the third step above. $\square$

We use Lemma 4 to derive upper bounds on errors obtained by having an ϵ -approximate decomposition of the value function.

THEOREM 3. For ϵ -approximate decomposition, $v = \epsilon + \sum_k v_k$, where v is strongly sub-additive and v_k concerns a set $C_k \subseteq N$, given that C_k s for $k \in \{1, \dots, K\}$ form a partition of N , we have:

$$\epsilon(S) \leq \sum_{1 \leq \ell \neq k \leq K} \sum_{i \in C_k \cap S, j \in C_\ell \cap S} \mathcal{M}_2(\{i\}, \{j\})$$

PROOF. The proof is by induction on the size of the decomposition K . Let $S_k = S \cap C_k$, and S_k 's form a partition of S . First, note that:

$$\begin{aligned} v(S) &= \epsilon(S) + \sum_{k=1}^K v_k(S) = \epsilon(S) + \sum_{k=1}^K v(S \cap C_k) \\ &= \epsilon(S) + \sum_{k=1}^K v(S_k). \end{aligned}$$

Now, suppose $K = 2$, then we have:

$$\epsilon(S) = v(S) - v(S_1) - v(S_2).$$

Applying Lemma 4, we have,

$$\begin{aligned} \epsilon(S) &= v(S) - v(S_1) - v(S_2) = \mathcal{M}_2(S_1, S_2) \\ &\leq \sum_{i \in S_1, j \in S_2} \mathcal{M}_2(\{i\}, \{j\}). \end{aligned}$$

Suppose the bound holds for K . Then, we show that it holds for $K + 1$. For this, we let $C'_K = C_K \cup C_{K+1}$. Hence we have $S'_K = S_K \cup S_{K+1}$, and:

$$\begin{aligned} \epsilon(S) &= v(S) - v(S_1) - \dots - v(S_{K-1}) - v(S_K) - v(S_{K+1}) \\ &= v(S) - v(S_1) - \dots - v(S_{K-1}) - v(S'_K) + v(S'_K) \\ &\quad - v(S_K) - v(S_{K+1}). \end{aligned}$$

We apply the induction hypothesis to the K partition, C'_K :

$$\begin{aligned} \epsilon(S) &= [v(S) - v(S_1) - \dots - v(S_{K-1}) - v(S'_K)] \\ &\quad + [v(S'_K) - v(S_K) - v(S_{K+1})] \\ &\leq [\sum_{1 \leq \ell \neq k \leq K-1} \sum_{i \in S_k, j \in S_\ell} \mathcal{M}_2(\{i\}, \{j\})] \\ &\quad + [\sum_{1 \leq \ell \leq K-1} \sum_{i \in S'_K, j \in S_\ell} \mathcal{M}_2(\{i\}, \{j\})] \\ &\quad + [\sum_{i \in S_K, j \in S_{K+1}} \mathcal{M}_2(\{i\}, \{j\})] \end{aligned}$$

Combining the last two terms, and using,

$$\begin{aligned} \sum_{1 \leq \ell \leq K} \sum_{i \in S'_K, j \in S_\ell} \mathcal{M}_2(\{i\}, \{j\}) &= \\ \sum_{1 \leq \ell \leq K-1} \sum_{i \in S_K, j \in S_\ell} \mathcal{M}_2(\{i\}, \{j\}) &+ \sum_{i \in S_{K+1}, j \in S_\ell} \mathcal{M}_2(\{i\}, \{j\}), \end{aligned}$$

we conclude that,

$$\epsilon(S) \leq \sum_{1 \leq \ell \neq k \leq K} \sum_{i \in S_k, j \in S_\ell} \mathcal{M}_2(\{i\}, \{j\}).$$

□

Based on this, the following result bounds the effect that weights on edges between groups in the partition have on the approximation to Shapley value.

COROLLARY 1. *For ϵ -approximate decomposition, $v = \epsilon + \sum_k v_k$, where v is strongly sub-additive and v_k concerns a set $C_k \subseteq N$, and $k \in \{1, \dots, K\}$, we have:*

$$|Sh(N, i) - \sum_k Sh_{v_k}(C_k, i)| = |Sh_\epsilon(N, i)| \leq 2 \sum_{1 \leq \ell \neq k \leq K} \sum_{i \in C_k, j \in C_\ell} \mathcal{M}_2(\{i\}, \{j\})$$

This result follows by using Theorem 3, with which we can bound the marginal contributions using $|\epsilon(S') - \epsilon(S)| \leq 2 \sum_{1 \leq \ell \neq k \leq K} \sum_{i \in C_k, j \in C_\ell} \mathcal{M}_2(\{i\}, \{j\})$.

5. CONSTRUCTING THE PAIRWISE SIMILARITY GRAPH

In this section, we define a graph based on the 2-similarity metric among pairs of agents and, on the basis of this graph, we identify a useful partition of agents into groups, with the exact Shapley value computed for each group. We will call the 2-similarity metric among pairs of agents the *pairwise similarity*, and view the resulting partition as an approximate multi-issue decomposition.

The pairwise similarity graph associated with a set of agents N in a coalitional game is a graph with $n = |N|$ vertices, and edges whose weights are given by $\mathcal{M}_2(\{i\}, \{j\})$ for vertices $1 \leq i, j \leq n$. Following the direction suggested in Corollary 1, we have:

COROLLARY 2. *Suppose we have a partitioning $C_1, C_2, \dots, C_K$ of the pairwise similarity graph of a game and $v_k(S) = v(S \cap C_k)$ for all k . Let*

$$F(C_1, \dots, C_K) = \sum_{1 \leq \ell \neq k \leq K} \sum_{i \in C_k, j \in C_\ell} \mathcal{M}_2(\{i\}, \{j\})$$

be the weight of the inter-partition edges. We can compute an approximation to the Shapley values with the following error bound:

$$|Sh(N, i) - \sum_k Sh_{v_k}(C_k, i)| \leq 2F(C_1, \dots, C_K),$$

in $O(K2^{\max_k |C_k|})$ time.

The claim on the running time follows because we compute the Shapley value exactly for K smaller games with the size at most $\max_k |C_k|$.

Our goal, then, is to find a partition of agents that minimizes the inter-partition weights in the pairwise similarity graph. There are multiple ways of partitioning the graph. We describe an approach for finding a partition consisting of two components, but smaller partitions can be obtained by either recursively partitioning the graph or more direct methods.

For partitioning the graph we use *Spectral clustering* [12]. This method partitions the graph in $O(n^3)$ time steps, by computing the eigenvector of the adjacency matrix of the graph corresponding to the second largest eigenvalue and choosing the two partitions by separating the positively and negatively signed elements.

Once we have such a split of agents into two sets, we can apply the above corollary to conclude that:

THEOREM 4. *For partitioning the pairwise similarity graph on a set N of agents ($n = |N|$) into two groups of agents C_1, C_2 with min-cut flow ζ , meaning that:*

$$\sum_{i \in C_1, j \in C_2} \mathcal{M}_2(\{i\}, \{j\}) = \zeta,$$

then we have:

$$|Sh(N, i) - \sum_{k=1}^2 Sh_{v_k}(C_k, i)| \leq 2\zeta,$$

where $v_1(S) = v(S \cap C_1)$ and $v_2(S) = v(S \cap C_2)$ for all $S \subseteq N$ with corresponding complexity:

$$O(2^{\max(|C_1|, |C_2|)} + n^3).$$

To further reduce the running time of computing Shapley we can proceed by recursively sub-dividing the partitions, and accumulate the error from the partitioning. This motivates Algorithm 1, which partitions agents until $\max_k |C_k|$ (the maximal cardinality of any group) is small enough that exact Shapley value computation on each partition is feasible.

Algorithm 1 Recursive Partitioning for Approximating Shapley (RPAS) for a game G with agents in N_G with the coalitional value function v_G and with maximum acceptable set cardinality set to K_{max} .

Input: Coalitional game $G = (N_G, v_G)$

Variables: K_{max}

RPAS(G):

if $|N_G| \leq K_{max}$ **then**

return Shapley(G)

end if

 Compute $\mathcal{M}_2$ for v_G

$(G_1, G_2) := \text{Partition}(\mathcal{M}_2)$

return Concatenate(RPAS(G_1), RPAS(G_2))

The pairwise similarity graph can be computed in $O(n^2)$ time. The recursive clustering can be done in $O(n^4)$ time and we can apply Corollary 2 to compute the approximation to Shapley in $O(K2^{\max_k |C_k|} + n^4)$ time. The error in approximating the Shapley value through the recursive algorithm can be bounded by:

$$|Sh(N, i) - \sum_k Sh_{v_k}(C_k, i)| \leq 2 \sum_k \zeta_k$$

where $v_k(S) = v(S \cap C_k)$ for all k and C_k is the k th partition generated in RPAS algorithm. ζ_k corresponds to the cut capacity for the clustering of the agents in the k th recursive application of the algorithm.

6. A CROWD-LEARNING SETTING

Consider an application where a crowd of users is used to help with a machine-learning task. We refer to this as a *crowd-learning setting*.

We consider three variants. First, users in the crowd are tasked with providing item labels (e.g., topic labels for web pages). Second, users in the crowd are tasked with providing features (e.g., unigram text features for web pages).

Third, users are tasked with providing both item labels and features.

6.1 Labels Game

In this section we illustrate a simple, stylized example, where the exact Shapley value can be efficiently computed. Suppose we define the following game for a set of items $\mathcal{B}$, where the items can be labeled by a set of agents $N = \{1, \dots, n\}$.

- Each agent provides you a subset of the items $B_i \subseteq \mathcal{B}$.
- The game is represented as a function $v : 2^N \rightarrow \mathbb{R}$, and for simplicity we imagine that this maps each coalition (combination of items gathered by agents for agents in $S \in N$) to the number of unique positive labeled items, with

$$v(S) = |\cup_{i \in S} B_i|.$$

The pairwise similarity metric for this game is simply,

$$\mathcal{M}_2(\{i\}, \{j\}) = |B_i \cap B_j|.$$

Let's apply the decomposition (1),

$$Q_1(S) = \sum_{i \in S} |B_i|, \quad Q_2(S) = \sum_{i \neq j \in S} |B_i \cap B_j|$$

$$Q_K(S) = \sum_{i_1, \dots, i_K \in S, i_l \neq i_k} \left| \bigcap_{l \in \{1, \dots, K\}} B_{i_l} \right|$$

Then we have the following decomposition,

$$v(S) = \sum_{K=1}^{|S|} (-1)^{K+1} Q_K(S)$$

And using the result in Theorem 2, the Shapley value is,

$$Sh_v(N, i) = \sum_K \frac{(-1)^{K+1}}{K} \sum_{j_1, \dots, j_{K-1}} |B_i \cap \bigcap_{l \in \{1, \dots, K-1\}} B_{j_l}| \quad (3)$$

$$= |B_i| - \frac{1}{2} \sum_{i \neq j} |B_i \cap B_j| + \frac{1}{3} \sum_{i \neq j \neq k} |B_i \cap B_j \cap B_k| - \dots \quad (4)$$

This game is equivalent to the *recommendation game* in Kleinberg et al. [13], who derived the simplified expression for the equation(3) above as:

$$Sh_v(N, i) = \sum_{j \in B_i} \frac{1}{|\{k : j \in B_k\}|}$$

6.2 Features Game

In this section, we define a coalitional game where agents correspond to features in a regression setting as follows.¹ For a set of agents, $N = \{1, \dots, n\}$, we have:

- Each agent has a vector $X_{\{i\}} \in \mathbb{R}^m$
- For a vector $Y_v \in \mathbb{R}^m$ and a set coefficients $\alpha \in \mathbb{R}^n$, we have:

$$Y_v = X_N \alpha$$

$$\text{for } X_N = X_{\{i|i \in N\}}.$$

¹The setup is similar to the regression game framework proposed by Pinter [18].

- The regression is used as a projection to find α_S to approximate Y_v using a coalition S of features:

$$\hat{Y}_v = X_S \alpha_S$$

$$\alpha_S = (X_S^T X_S)^{-1} X_S^T Y_v$$

for $X_S = X_{\{i|i \in S\}}$.

- The game is represented as a function $v : 2^N \rightarrow \mathbb{R}$ which maps each coalition (set of vectors) $S \in N$ to a real number,

$$v(S) = \|\hat{Y}_v\|_2^2$$

$$= \|X_S \alpha_S\|_2^2 = \alpha_S^T X_S^T X_S \alpha_S = \alpha_S^T \Sigma_S \alpha_S,$$

where $\Sigma_S = X_S^T X_S$ and Σ_N is an invertible matrix, and $\|\cdot\|_2$ is the L_2 norm. $v(S)$ represents the size of the component of Y_v which is explainable with the features in the coalition S .

THEOREM 5. For the regression game, the value:

$$\psi_i(v) = \alpha_{N_i}^2 + \sum_{j \neq i} \alpha_{N_j} \alpha_{N_i} \sigma_{ij} = \alpha_{N_i} \cdot \langle Y_v, X_i \rangle$$

is the Shapley value for $v(S)$.

Where $\langle Y_v, X_i \rangle$ is the inner product of the vectors Y_v and X_i . The proof establishes the four axioms for the proposed value function. We skip the proof in the interest of the space.

6.3 Label and Feature Game

A more general crowd-learning problem involves eliciting both labels and features from participants, and building a classifier.

Suppose we have a set of agents $N = \{1, \dots, n\}$, where:

- Each agent has a set of labels and a set of features.
- We have a classifier, that adopts the following design elements:
 - A logistic regression is trained on the basis of the labels and features suggested by a coalition $S \subseteq N$ of agents.
 - Regularization is accomplished through cross validation.
 - The performance is measured as the area-under-the-curve (AUC), using a testing set selected from combination of labels and features from all the agents in N .
- The coalitional value function is $v(S) = AUC(S)$, adopting the performance metric for the classifier.

The generality of the above setting does not allow for analytical or closed form solutions for Shapley. Rather, we will approximate the Shapley value using our proposed pairwise similarity graph and agent-partitioning method.

For example, suppose that we have a webpage classification problem for the topic *science-math*. Participants search among a set of webpages and find webpages that they are comfortable with labeling and provide labels. Moreover, they provide a set of features for the webpages. Then we consider their labels and features in a setting similar to the above label and feature game. We will describe this setting in more detail in the experimental section.

7. EXPERIMENTAL RESULTS

In this section, we compare the efficiency and accuracy of our approximation method with random sampling approaches to estimating the Shapley value. We consider both synthetic and real-world experiments.

7.1 Synthetic Experiments

In the first set of experiments, we generate synthetic data for a feature game setting defined in Section 6.2 where we have n features (X_i 's) with dimension m .

Features (X_i 's) are generated from a Dirichlet distribution with uniform density on the simplex. The independent variable (Y vector) is a score that is generated synthetically by multiplying X vectors to the coefficients in the regression (which are randomly samples from Normal distribution with mean 0 and variance 1). Y is normalized such that $\|Y\| = 1$, and we set number of document to $m = 1000$ and number of dictionaries to $n = 16$.

We compute the (approximate) Shapley values using four different methods. The first method is the exact computation of Shapley, and is performed by enumerating over all agent orders. The second and third methods approximate Shapley values using the random-sample approach, which samples either 1/2 or 1/4 of all possible agent orders for the purpose of approximating the Shapley values. The fourth method approximates Shapley values by clustering the agents into two clusters, using the partitioning through the eigenvector for the second largest eigenvalue of the pairwise similarity matrix, respectively using the spectral clustering.

We illustrate the computational time and accuracy of the methods for the mean squared error between the exact Shapley and approximated Shapley in Figure 1. We further illustrate the error in the ranking of agents using a Kendall correlation metric between the exact ranking of agents from exact Shapley and approximate rankings.

7.2 Real World Experiments

In real world studies we are considering a crowd-learning classification problem as described in § 6.3, in which the participants provide labels along with features for webpages, and we use these inputs to train and evaluate a classifier.

We repeated the experiment for classification of webpages for ten different topics such as (Science-Math, Home-Cooking, Comics etc.). Each topic involved different number of agents (because participants tend to provide different number of features.). For each topic, we have the same set of four meth-

	MSE	MSE Top 3	Kendall
Clustering-2	.06(.01)	.04(.02)	.71(.54)
Random-2	.34(.17)	.21(.23)	.61(.44)

Table 1: Performance of different Approximation methods for real world experiment

ods for computing Shapley as in the synthetic experiments. The results for computational time versus the number of agents are illustrated in Figure 2, showing the efficiency of the proposed approximation. In Table 1 we have computed the average of the mean squared errors between the exact Shapley and approximated Shapley with different methods. We look into the the same average for mean squared error for only top 3 and also to the error in the ranking us-

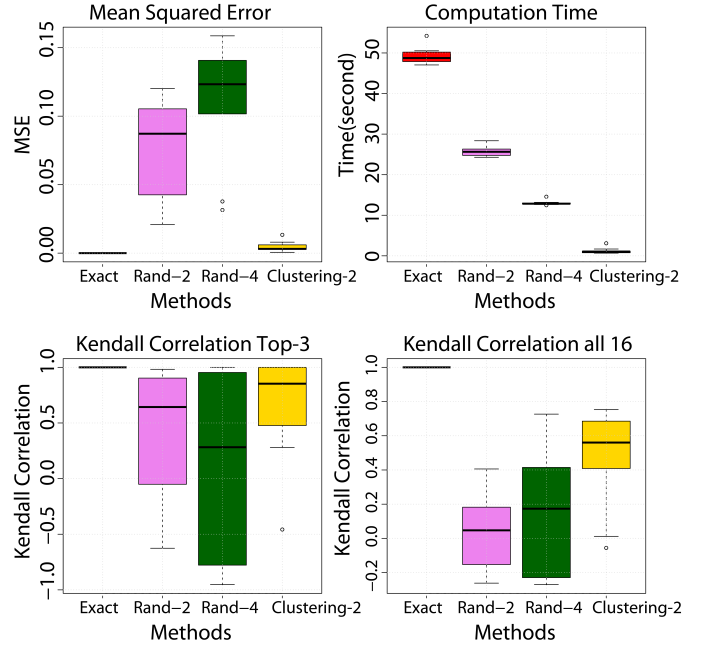


Figure 1: Computational time and accuracy comparisons for synthetic experiments with $n = 16$ agents for vectors with dimension $m = 1000$

ing Kendall correlation (the more the better). All the error metrics indicate that our method out performs the random approximation by some margin.

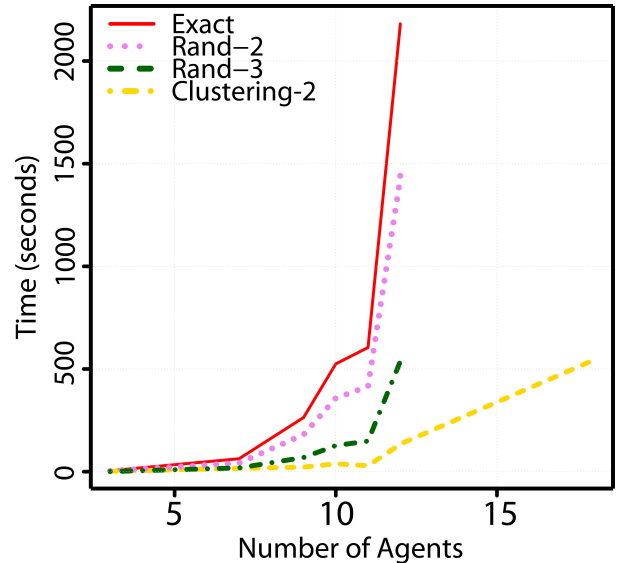


Figure 2: Computation time for the classification tasks versus the size of the agents in the tasks.

In addition, to show the underlying structure we display a k -means clustering of the pairwise similarity graph of agents for two topics (science-math and comics) in Figure 3. The figures reveal that the similarity graph partitions into tight clusters.

Though the AUC metric for a classifier built with these

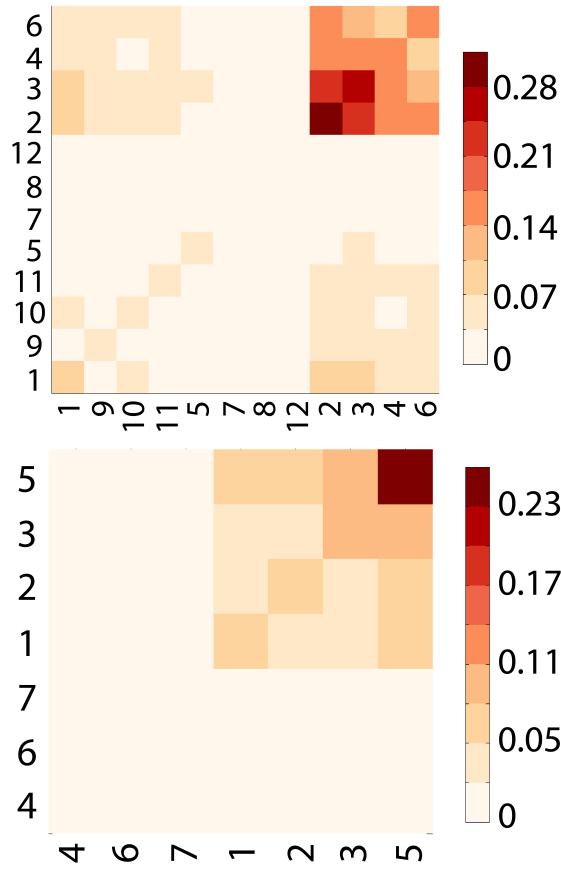


Figure 3: Similarity Matrix for two classification tasks, upper panel is Science-Math with 11 agents and lower panel is the Comics with 7 agents.

labels and features is not strongly sub-additive, the empirical results still yield very good approximations. This indicates that results obtained in this paper perhaps hold under more lenient assumptions.

8. ACKNOWLEDGEMENT

This work is supported in part by NSF Grants No. CCF-0915016 and No. AF-1301976.

9. REFERENCES

- [1] Jacob D. Abernethy and Rafael M. Frongillo. A collaborative mechanism for crowdsourcing prediction problems. In J. Shawe-taylor, R. S. Zemel, P. Bartlett, F. C. N. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2600–2608. 2011.
- [2] Yoram Bachrach, Reshef Meir, Michal Feldman, and Moshe Tennenholtz. Solving cooperative reliability games. *arXiv preprint arXiv:1202.3700*, 2012.
- [3] Javier Castro, Daniel Gómez, and Juan Tejada. Polynomial calculation of the Shapley value based on sampling. *Computers & Operations Research*, 36(5):1726–1730, 2009.
- [4] Shay Cohen, Gideon Dror, and Eytan Ruppin. Feature selection via coalitional game theory. *Neural Computation*, 19(7):1939–1961, 2007.
- [5] Vincent Conitzer and Tuomas Sandholm. Computing Shapley values, manipulating value division schemes, and checking core membership in multi-issue domains. In *AAAI*, volume 4, pages 219–225, 2004.
- [6] Xiaotie Deng and Christos H. Papadimitriou. On the complexity of cooperative solution concepts. *Mathematics of Operations Research*, 19(2):257–266, 1994.
- [7] S. Shaheen Fatima, Michael Wooldridge, and Nicholas R. Jennings. A linear approximation method for the Shapley value. *Artif. Intell.*, 172(14):1673–1699, 2008.
- [8] Michel Grabisch and Marc Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of Game Theory*, 28(4):547–565, 1999.
- [9] Claus-Jochen Haake, Akemi Kashiwada, and Francis Edward Su. The Shapley value of phylogenetic trees. *Journal of mathematical biology*, 56(4):479–497, 2008.
- [10] Sergiu Hart and Andreu Mas-Colell. Bargaining and value. *Econometrica: Journal of the Econometric Society*, pages 357–380, 1996.
- [11] Samuel Ieong and Yoav Shoham. Marginal contribution nets: a compact representation scheme for coalitional games. In *Proceedings of the 6th ACM conference on Electronic commerce*, pages 193–202. ACM, 2005.
- [12] Ravi Kannan, Santosh Vempala, and Adrian Vetta. On clusterings: Good, bad and spectral. *Journal of the ACM (JACM)*, 51(3):497–515, 2004.
- [13] Jon Kleinberg, Christos H. Papadimitriou, and Prabhakar Raghavan. On the value of private information. In *Proceedings of the 8th conference on Theoretical aspects of rationality and knowledge*, pages 249–257. Morgan Kaufmann Publishers Inc., 2001.
- [14] Richard T. B. Ma, Dah Ming Chiu, John C. S. Lui, Vishal Misra, and Dan Rubenstein. The Shapley profit for content, transit and eyeball internet service providers.
- [15] Daniel Q. Naiman and Henry P. Wynn. Inclusion-exclusion-Bonferroni identities and inequalities for discrete tube-like problems via Euler characteristics. *The Annals of Statistics*, pages 43–76, 1992.
- [16] Abraham Neyman. Uniqueness of the Shapley value. *Games and Economic Behavior*, 1(1):116–118, 1989.
- [17] David Pérez-Castrillo and David Wettstein. Bidding for the surplus: a non-cooperative approach to the Shapley value. *Journal of Economic Theory*, 100(2):274–294, 2001.
- [18] Miklós Pintér. Regression games. *Annals of Operations Research*, 186(1):263–274, 2011.
- [19] Roberto Serrano. Cooperative games: Core and Shapley value. Technical report, CEMFI, 2007.
- [20] Amos Storkey. Machine learning markets. *arXiv preprint arXiv:1106.4509*, 2011.
- [21] Eyal Winter. The Shapley value. *Handbook of game theory with economic applications*, 3:2025–2054, 2002.