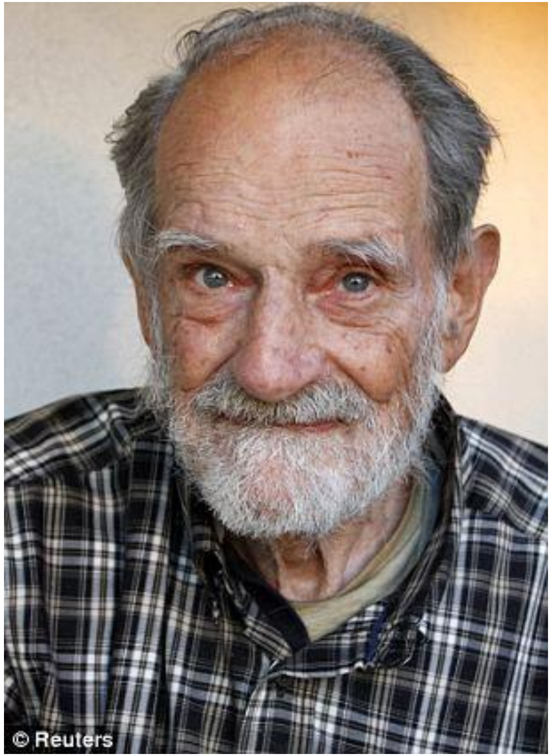
Compact Representations of Coalitional Games

Jackie Liu

CPSC 532L

The Shapley Value is Useful 😀

- A fair way to divide the payoff of a coalition among its players
- Has many ML applications - stay tuned for the next talk



Lloyd S. Shapley

Computing the Shapley Value is Tedious 😔

We have to calculate the marginal contributions of each player, averaged over all possible orderings of how the coalition can be formed

- If there are *n* players, we must average over *n!* orderings of the players
- Computing the Shapley value becomes impractical very fast

Q: Is there a more efficient way to compute the shapley value?

Problem with the Naive Representation 😕

Even before that, how do we input the coalitional game into such programs?

Michael: *How can computing the core be polynomial if the game has an exponential number of subsets of N?*

Kevin: *It is polynomial to the size of the input, if your input size is exponential then too bad.*

Rest of the class: **audible laughter**

Motivations for Compact Representations

- We want more **efficient** ways to compute the solution concepts
- We want more **compact** ways to represent coalitional games

Presentation Outline

Motivation ✔

Overview of Compact Representations

I. Weighted Graph

II. Marginal Contribution Nets

Related Works

Takeaways

Compact Representation 👏

Representations that require at most polynomial space in the number of players

Limitations

- Tradeoff between the compactness of the representation, and the complexity of the associated computational problems
- Representations may not be able to cover all coalitional games

Weighted Graphs

Weighted Graphs

Proposed by Deng and Papadimitriou (1994)

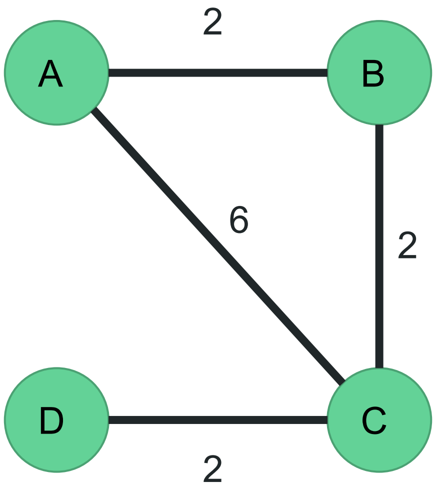
Idea: Represent the coalitional game as an undirected, weighted graph

- Vertices ➔ Players
- Edges ➔ Some integer
- Value of a coalition ➔ The weight of its induced subgraph

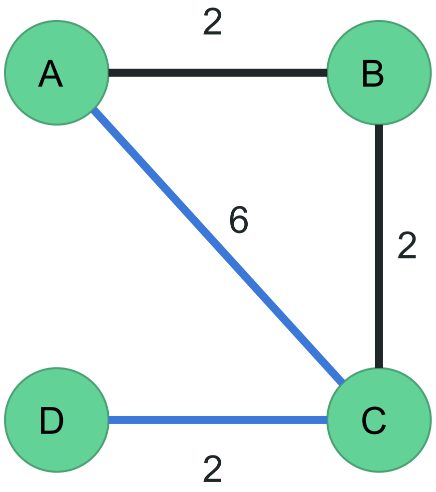
Games that are represented this way are called **induced subgraph games**

Weighted Graphs Representation

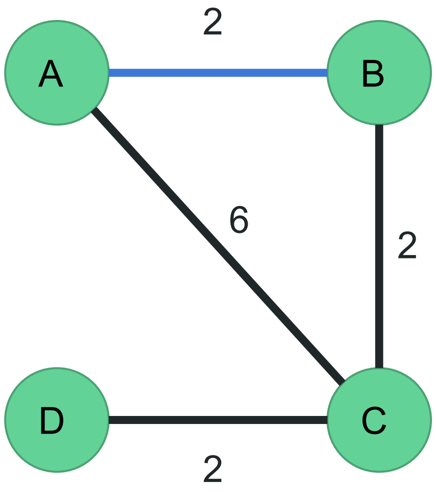
Consider a game with players $N = \{A, B, C, D\}$



Weighted graph representation



v({A, C, D}) = 2 + 6 = 8



v({A, B, D}) = 2

Weighted Graphs Shapley Value Computation

1. Consider every edge in the graph to be a separate game
2. Compute the Shapley value of a player in each edge game and sum them up

   Why does this work?   (Hint: one of Shapley's axioms)

Weighted Graphs Shapley Value Computation

1. Consider every edge in the graph to be a separate game
2. Compute the Shapley value of a player in each edge game and sum them up (Shapley's axiom: Additivity)
   a. Players gets a value of 0 for an edge they are not connected to
   b. Players gets half the weight of an edge they are connected to (Shapley's axiom: Symmetry)

Weighted Graphs Properties

Compact ✔ for a game with n players, we only need $O(n^2)$ space

Not Complete ❌ there are games that the weighted graph can't represent (e.g. a majority voting game)

Computing the **Shapley value**: Polynomial

Marginal-Contribution Nets

Marginal-Contribution Nets

Proposed by Ieong and Shoham (2005)

Idea: Use a set of rules to describe the marginal contributions of the players

- Rules are in the form

$$\textit{Pattern} \mapsto \textit{value}$$

- Patterns ➔ boolean condition over the set of players
- Value of a coalition ➔ sum of the values of all rules that apply to the coalition

MC-Nets Representation

Consider a game with players $N = \{A, B, C, D\}$

$\{A\} \mapsto 1$

$\{A \wedge B\} \mapsto 2$

$\{B \wedge D\} \mapsto 7$

$\{A \wedge D \wedge C\} \mapsto 6$

MC-Net Representation in the basic form (only conjunctions)

$v(\{A, B\}) = 1 + 2 = 3$

$v(\{A, B, D\}) = 1 + 2 + 7 = 10$

$v(\{B, C, D\}) = 7$

MC-Nets (Basic Form) Shapley Value Computation

1. Consider each rule as a separate game
2. Compute the Shapley value of a player in each rule and add them together (Shapley's axiom: additivity)
    a. For each rule a player belongs to, the player gets the value of that rule divided by the number of players in the rule (Shapley's axiom: symmetry)

MC-Nets are a generalization of Weighted Graphs

MC-Nets Properties

Compact ✔ for a game with *m* subgames, where the largest subgame has *n* players, it takes $O(m2^n)$ space

Complete ✔* one rule for every possible coalition

- Trade off between representational power and computation efficiency

Computing the **Shapley value**: Linear in the basic form

Related Work to Consider

Our discussion has been focused on the Shapley value

- What about the core?

The methods described in this presentation rely on the coalitional game to have certain properties

- What about more general methods? (e.g. *read-once MC-nets*)

Takeaways 👋

We use compact representations to:

- Reduce the space required to represent coalitional games
- Improve computational complexity for solution concepts

Trade off of compact representations

- □ Representational power/Compactness ↔ □ Computation efficiency

References

X. Deng and C. Papadimitriou, “On the Complexity of Cooperative Solution Concepts,” Mathematics of Operations Research, vol. 19, no. 2, 1994, pp. 257–266.

S. Ieong and Y. Shoham, “Marginal Contribution Nets: A Compact Representation Scheme for Coalitional Games,” Proc. 6th ACM Conf. Electronic Commerce (ACM-EC 05), ACM, 2005, pp. 193–202.

G. Chalkiadakis, E. Elkind, and M. Wooldridge, “Cooperative Game Theory: Basic Concepts and Computational Challenges,” in IEEE Intelligent Systems, vol. 27, no. 3, 2012, pp. 86-90.