Learning Cooperative Solution Concepts From Voting Behavior

A Case Study on the Israeli Parliament

Lu Wei

Department of Computer Science
National University of Singapore

Jan 2020

Outline

Background

- Coalition formation games:
  - Mostly theoretical analysis
  - Most models require full information on player preference
  - Lack of large dataset with ground truth
- Clustering & community detection:
  - Data-driven analysis
  - Missing strategic behavior modelling

Data

- Israeli parliament (the Knesset) voting data
- Available since March 2017
- Contains 147 parliament members' votes on over 7500 bills in ~ 4 years
- Ground truth clusters: political party affiliations
  - 10 parties aligned along a left-right axis
  - Ideological agreement among the government (right) and the opposition (left) parties respectively

Probably Approximately Correct (PAC) Stability

Given m observations of formed coalitions $S_1, \dots, S_m$ sampled i.i.d. from some distribution $\mathcal{D}$, and the cardinal valuations of players in S_j $(v_i(S_j))_{i \in S}$
Local loss given a coalition structure π and a coalition $S \subseteq N$:

$$\lambda(\pi, S) = \begin{cases} 1 & \text{if } \forall i \in S : v_i(S) > v_i(\pi(i)) \\ 0 & \text{otherwise.} \end{cases}$$

The *expected loss* of π w.r.t. $\mathcal{D}$:

$$L_{\mathcal{D}}(\pi) = \Pr_{S \sim \mathcal{D}} [\lambda(\pi, S) = 1]$$

A *PAC stabilizing* algorithm:

- input: a set of i.i.d. samples $S_1, \dots, S_m \sim \mathcal{D}^m$
- output: a coalition structure π^* with the following guarantee: $\Pr_{(S_1,\dots,S_m)\sim\mathcal{D}^m}[L_{\mathcal{D}}(\pi^*) \geq \varepsilon] < \delta$
- The number of samples needed m grows linearly in the number of players, and polynomially in $\frac{1}{\varepsilon}$ and $\log \frac{1}{\delta}$

Research Questions

- Can we use hedonic games to model real-world collaborative activities?
- How well does the outcome compare to ground truth?
- How well does the outcome compare to that of canonical clustering and community detection models?

Methodology

Game Theoretic Models

- Assumes hedonic preferences
  - Top responsive
  - Bottom responsive
  - Boolean

Methodology

Game Theoretic Models

- Assumes hedonic preferences
  - Top responsive
  - Bottom responsive
  - Boolean
- Two approaches:
  - Full-information models: construct complete preference profile, then derive core stable solution
  - PAC models: learn a probably stable solution directly from partial preference relations

Methodology

Game Theoretic Models

- Assumes hedonic preferences
  - Top responsive
  - Bottom responsive
  - Boolean
- Two approaches:
  - Full-information models: construct complete preference profile, then derive core stable solution
  - PAC models: learn a probably stable solution directly from partial preference relations
    - Simulate i.i.d.: sampling with replacement 3/4 of all bills
    - Repeat 50 times to check consistency of solution between runs
    - Select the "centroid" to represent model output: partition with minimum sum of information distance from other 49 partitions

Methodology

Comparisons

- Comparison machine learning models:
  - K-Means
  - Stochastic Block Model
- Comparing every hedonic & ML model output partition to ground truth party affiliations
  - Quantitatively: information theoretic measures
  - Qualitatively: political analysis

How different are two partitions, quantitatively?

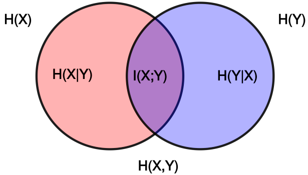
Objectives

We want a measure that...

- has strong mathematical foundation: information theoretic measures
- is intuitive: satisfing metric property
  - non-negativity
  - symmetry
  - triangle inequality

How different are two partitions, quantitatively?

Definitions

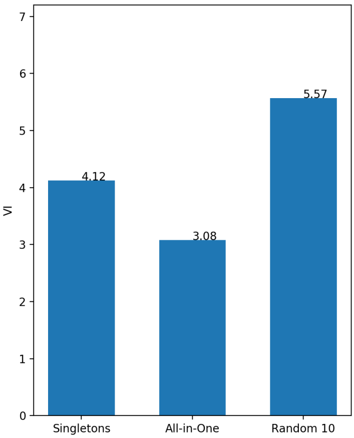


Venn diagram showing additive and subtractive relationships various information measures associated with correlated variables X and Y

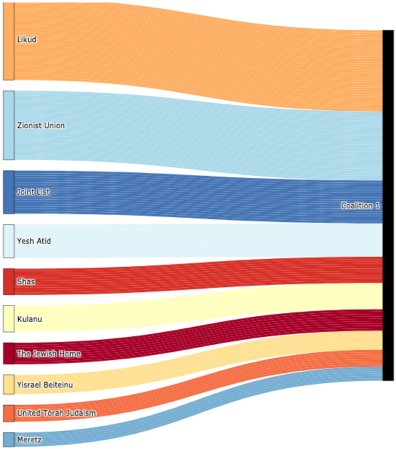
- Entropy: $H(\pi) = -\sum_{j=1}^{J} \frac{|S_j|}{|N|} \log \frac{|S_j|}{|N|}$
- Conditional entropy: $H(\pi|\pi') = -\sum_{j=1}^{J} \sum_{k=1}^{K} \frac{|S_j \cap S'_k|}{|N|} \log \frac{|S_j \cap S'_k|/|N|}{|S_k|/|N|}$
- Mutual Information (MI): $I(\pi, \pi') = H(\pi) - H(\pi|\pi')$
- Variation of Information (VI): $VI(\pi, \pi') = H(\pi) + H(\pi') - 2I(\pi, \pi')$, metric!

How different are two partitions, quantitatively?

Baseline Values



The Knesset partition baseline VI values



All-in-One partition of the Knesset

How different are two partitions, quantitatively?

Additional Measure: AMI

Adjusted Mutual Information (AMI):

$$AMI(\pi, \pi') = \frac{I(\pi, \pi') - E(I(\pi, \pi'))}{\max(H(\pi), H(\pi')) - E(I(\pi, \pi'))}$$

- Adjusted for chance
- Normalized: $[0, 1]$
- Not metric

Good for detecting "bad" (very different) partitions:

	Ajusted Mutual Information
Singletons	3e-14
All-in-One	-5e-16
Randome 10	0.007

The Knesset partition baseline AMIs

Top Responsive Games

Definitions

Idea: for every player, the value of a coalition depends on the most preferred subset of players

- *Choice sets*:
  $Ch(i,S) = \{S' \subseteq S : (i \in S') \wedge (S' \succeq_i S'' \forall S'' \subseteq S)\}$
  When $|Ch(i,S)| = 1$, the unique choice set is $ch(i,S)$
- A *top responsive* preference profile requires that for any player $i \in N$, and any coalition that may contain player i:
  $S, T \in \mathcal{N}_i$:
  1. $|Ch(i,S)| = 1$.
  2. if $ch(i,S) \succ_i ch(i,T)$ then $S \succ_i T$
  3. if $ch(i,S) = ch(i,T)$ and $S \subset T$ then $S \succ_i T$

Top Responsive Games

Example

Example 1

Consider a game of three players with the following choice sets:
$ch(1,\{1,2,3\}) = ch(1,\{1,2\}) = \{1,2\}, ch(1,\{1,3\}) = \{1,3\}, ch(1,\{1\}) = \{1\}$
$ch(2,\{1,2,3\}) = ch(2,\{2,3\}) = \{2,3\}, ch(2,\{1,2\}) = \{1,2\}, ch(2,\{2\}) = \{2\}$
$ch(3,\{1,2,3\}) = ch(3,\{1,3\}) = \{1,3\}, ch(3,\{1,2\}) = \{1,2\}, ch(3,\{3\}) = \{3\}$
Then the resulting preference profile is top responsive:
player 1: $\{1,2\} \succ_1 \{1,2,3\} \succ_1 \{1,3\} \succ_1 \{1\}$
player 2: $\{2,3\} \succ_2 \{1,2,3\} \succ_2 \{1,2\} \succ_2 \{2\}$
player 3: $\{1,3\} \succ_3 \{1,2,3\} \succ_3 \{2,3\} \succ_3 \{3\}$

Top Responsive Games

Core Finding Algorithms - Full Information

Algorithm 1 Top Covering Algorithm

Input: A hedonic game satisfying top responsiveness.

1: $R^1 \leftarrow N$; $\pi \leftarrow \emptyset$.
2: **for** $k = 1$ to $|N|$ **do**
3:     Select S^k
4:     $\pi \leftarrow \pi \cup \{S^k\}$ and $R^{k+1} \leftarrow R^k \setminus S^k$
5:     **if** $R^{k+1} = \emptyset$ **then**
6:         **return** π
7:     **end if**
8: **end for**
9: **return** π

Let $C^1(i,S) = ch(i,S) \quad C^{t+1}(i,S) = \bigcup_{j \in C^t(i,S)} ch(j,S)$

The *connected component* of i with respect to S: $CC(i,S) = C^{|N|}(i,S)$
Step 3: select $i \in R^k$ such that $|CC(i,R^k)| \leq |CC(j,R^k)|$ for each $j \in R^k$; and $S^k \leftarrow CC(i,R^k)$

Top Responsive Games

Algorithm 2 Top Covering Algorithm - PAC

Input: A hedonic game satisfying top responsiveness.

1: **return** π

Top Responsive Games

Core Finding Algorithms - PAC

Algorithm 2 PAC Top Covering Algorithm

Input: ε, δ, set $\mathcal{S}$ of $m = (2n^4 + 2n^3)\lceil \frac{1}{\varepsilon} \log \frac{2n^3}{\delta} \rceil$ samples from $\mathcal{D}$

1: $R^1 \leftarrow N$, $\pi \leftarrow \emptyset$
2: $\omega \leftarrow \lceil 2n^2 \frac{1}{\varepsilon} \log \frac{2n^3}{\delta} \rceil$
3: **for** $k = 1$ to $|N|$ **do**
4:     $\mathcal{S}' \leftarrow$ take and remove ω samples from $\mathcal{S}$
5:     $\mathcal{S}' \leftarrow \{T : T \in \mathcal{S}', T \subseteq R^k\}$
6:     **for** $i \in R^k$ **do**
7:         **if** $i \notin \bigcup_{X \in \mathcal{S}'} X$ **then**
8:             $B_{i,k} \leftarrow \{i\}$
9:         **else**
10:             $B_{i,k} \in \arg\max_{T \in \mathcal{S}'} v_i(T)$
11:             $B_{i,k} \leftarrow \bigcap_{\{T \in \mathcal{S}' : ch(i,T) = ch(i, B_{i,k})\}} T.$
12:         **end if**
13:     **end for**
14:     **for** $j = 1, \ldots, |R^k|$ **do**
15:         $\mathcal{S}'' \leftarrow$ take and remove ω samples from $\mathcal{S}$
16:         $\mathcal{S}'' \leftarrow \{T : T \in \mathcal{S}'', T \subseteq R^k\}$
17:         **for** $i \in R^k$ **do**
18:             $B_{i,k} \leftarrow B_{i,k} \cap \bigcap_{T \in \mathcal{S}'' : ch(i,T) = ch(i, B_{i,k})} T.$
19:         **end for**
20:     **end for**
21:     Select S^k
22:     $\pi \leftarrow \pi \cup \{S^k\}$; and $R^{k+1} \leftarrow R^k \setminus S^k$
23:     **if** $R^{k+1} = \emptyset$ **then**
24:         **return** π
25:     **end if**
26: **end for**
27: **return** π

- Steps 1-3, 21-27: the same structure as Algorithm 1
- Steps 4-20: approximate player preferences from sample observations of coalitions formed
- [1] Step 21: select $i \in R^k$ such that $|CC(i, R^k)| \leq |CC(j, R^k)|$ for each $j \in R^k$; and $S^k \leftarrow CC(i, R^k)$
- Improved Step 21: select the largest Strongly Connected Component (SCC) in the graph induced by R^k as vertices and directed edges E, $(i, j) \in E$ if $j \in ch(i, R^k)$ for all $j \in R^k$; and $S^k \leftarrow SCC(R^k)$

Top Responsive Games

Imporoved Core Finding Algorithm - PAC

- correctness proof hiteboarding, proof by picture
- running time improvements:
  - each iteration: from finding smallest CC's $\mathcal{O}(|V|(|V|+|E|))$ to finding the largest SCC's $\mathcal{O}(|V|+|E|)$
  - removing more players in the earlier iterations also reduces the amount of computation required for the later iterations

Top Responsive Games - Handcrafted Value Function

Let S_f be the set of members who voted "for" and S_a be the set of members who voted "against". $S_p = S_f \cup S_a$.

$$v_i(S) = \begin{cases} 1 + \frac{1}{|S|} + \frac{|S_p|}{|N|}, & \text{if } S \text{ is the winning majority} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

- A winning coalition is always worth more than a losing coalition
- $\frac{1}{|S|}$ reflects that a win is more valuable when achieved with fewer members
- The participation term $\frac{|S_p|}{|N|}$ gives a win more value when there are more effective votes for a given bill
- Assign all unobserved coalition the value of zero

Top Responsive Games - Handcrafted Value Function

Let S_f be the set of members who voted "for" and S_a be the set of members who voted "against". $S_p = S_f \cup S_a$.

$$v_i(S) = \begin{cases} 1 + \frac{1}{|S|} + \frac{|S_p|}{|N|}, & \text{if } S \text{ is the winning majority} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

- Core stable partition: apply Algorithm 1 with partial reference profile as input
- PAC stable partition: apply improved Algorithm 2, sampling with replacement to make up for insufficient samples

Top Responsive Games - Appreciation of Friends

Let G_i be player i's set of friends, and B_i the set of enemies. $G_i \cup B_i \cup i = N$ and $G_i \cap B_i = \emptyset$. A preference profile P^f is based on *appreciation of friends* if for all player $i \in N$, $S \succeq_i T$ if and only if

1. $|S \cap G_i| > |T \cap G_i|$ or
2. $|S \cap G_i| = |T \cap G_i|$ and $|S \cap B_i| \leq |T \cap B_i|$

- Friends: anyone whose votes agreed with the given player's more often than they disagreed
- Agreed votes are only counted if the given player voted "for" or "against"
- Disagreed votes:
  1. Narrow disagreement (general friends): the other player's vote is different from mine, and is either "for" or "against"
  2. Broad disagreement (selective friends): the other player's vote is different from mine

Example 2

Given 3 players and 3 bills, their votes are as follow:

	player 1	player 2	player 3
bill A	for	for	against
bill B	abstained	for	abstained
bill C	abstained	against	abstained

General friends preference profile:
player 1: $\{1,2\} \succ_1 \{1,2,3\} \succ_1 \{1\} \succ_1 \{1,3\}$
player 2: $\{1,2\} \succ_2 \{1,2,3\} \succ_2 \{2\} \succ_2 \{2,3\}$
player 3: $\{3\} \succ_3 \{1,3\} \sim \{2,3\} \succ_3 \{1,2,3\}$
Selective friends preference profile:
player 1: $\{1,2\} \succ_1 \{1,2,3\} \succ_1 \{1\} \succ_1 \{1,3\}$
player 2: $\{2\} \succ_2 \{1,2\} \sim \{2,3\} \succ_2 \{1,2,3\}$
player 3: $\{3\} \succ_3 \{1,3\} \sim \{2,3\} \succ_3 \{1,2,3\}$

Bottom Responsive Games

Definitions

Idea: for every player, the value of a coalition depends on **the absence of the least preferred** subset of players

- *Avoid sets*:
  $Av(i, S) = \{S' \subseteq S : (i \in S') \wedge (S' \preceq_i S'' \forall S'' \subseteq S)\}$
- A *bottom responsive* preference profile requires:
  1. if for all $S' \in Av(i, S)$ $T' \in Av(i, T)$, $S' \succ_i T'$ then $S \succ_i T$
  2. if $Av(i, S) \cap Av(i, T) \neq \emptyset$ and $|S| \geq |T|$ then $S \succeq_i T$

Bottom Responsive Games - Aversion to Enemies

A preference profile P^e is based on *aversion to enemies* if for every player $i \in N$, $S \succeq_i T$ if and only if

1. $|S \cap B_i| < |T \cap B_i|$ or
2. $|S \cap B_i| = |T \cap B_i|$ and $|S \cap G_i| \geq |T \cap G_i|$

It is a proper subclass of bottom responsive games

- Friends: anyone whose votes agreed with the given player's more often than they disagreed
- Agreed votes are only counted if the given player voted "for" or "against"
- Disagreed votes:
    1. Narrow disagreement (general friends/selective enemies): the other player's vote is different from mine, and is either "for" or "against"
    2. Broad disagreement (selective friends/general enemies): the other player's vote is different from mine

Example 3

Given 3 players and 3 bills, their votes are as follow:

	player 1	player 2	player 3
bill A	for	for	against
bill B	abstained	for	abstained
bill C	abstained	against	abstained

General friends / selective enemies preference profile:
player 1: $\{1,2\} \succ_1 \{1\} \succ_1 \{1,2,3\} \succ_1 \{1,3\}$
player 2: $\{1,2\} \succ_2 \{2\} \succ_2 \{1,2,3\} \succ_2 \{2,3\}$
player 3: $\{3\} \succ_3 \{1,3\} \sim \{2,3\} \succ_3 \{1,2,3\}$
Selective friends / general enemies preference profile:
player 1: $\{1,2\} \succ_1 \{1\} \succ_1 \{1,2,3\} \succ_1 \{1,3\}$
player 2: $\{2\} \succ_2 \{1,2\} \sim \{2,3\} \succ_2 \{1,2,3\}$
player 3: $\{3\} \succ_3 \{1,3\} \sim \{2,3\} \succ_3 \{1,2,3\}$

Bottom Responsive Games

Core Finding Algorithms - Full Information

Algorithm 3 Bottom Responsive Game Core Finding Algorithm

Input: A bottom responsive game

$S \leftarrow N$; $\pi \leftarrow \emptyset$.
while $S \neq \emptyset$ **do**
  Set $\Gamma \leftarrow \{S\}$
  Set $\Phi \leftarrow \{X \in \Gamma | \{i\} \in Av(i, X)$ for each $i \in X\}$
  while $\Phi = \emptyset$ **do**
    $\Gamma \leftarrow \bigcup_{X \in \Gamma} \bigcup_{i \in X} \{X \backslash \{j\} | j \in Y \text{ for some } Y \in Av(i, X)\}$
    $\Phi \leftarrow \{X \in \Gamma | \{i\} \in Av(i, X)$ for each $i \in X\}$
  end while
  Select a coalition $X \in \Phi$
  Set $\pi \leftarrow \pi \cup \{X\}$ and $S \leftarrow S \setminus X$
end while
return π

Bottom Responsive Games

Core Finding Algorithms - PAC

- Same structure as Algorithm 3
- Approximate player preferences $Av(i, X)$ from samples
  - Find friends from each set of sample coalitions
  - Take intersection of friend sets as "true friends"
  - Let each player's avoid set be players outside the "true friends" set

Boolean Hedonic Games

Idea: a player either likes to be a member of a coalition or hates it

- A player is indifferent among all satisfactory coalitions, same for unsatisfactory coalitions
- Strictly prefers any satisfactory coalition over any unsatisfactory coalition
- Within each bill, "for" and "against" groups each forms a satisfactory coalition
- Assume unobserved coalitions as unsatisfactory

Example 4

Given a parliament with 3 players and 3 bills, their votes are as follow:

	player 1	player 2	player 3
bill A	for	against	for
bill B	abstained	for	for
bill C	for	against	against

Boolean preference profile:
player 1: $\{1,3\} \sim \{1\} \succ_1 \{1,2\} \sim \{1,2,3\}$
player 2: $\{2\} \sim \{2,3\} \succ_2 \{1,2\} \sim \{1,2,3\}$
player 3: $\{1,3\} \sim \{2,3\} \succ_3 \{3\} \sim \{1,2,3\}$

Boolean Hedonic Games

Core Finding Algorithms - Full Information

Algorithm 4 Boolean Hedonic Game Core Finding Algorithm

Input: A Boolean hedonic game

1: $N' \leftarrow N$; $\pi \leftarrow \emptyset$.
2: **while** $N' \neq \emptyset$ **do**
3: Find $S \subset N'$ where all players in S find S satisfactory, and the size of S is the largest if there are multiple such coalitions.
4: $\pi \leftarrow \pi \cup \{S\}$ and $N' \leftarrow N' \setminus S$
5: **end while**
6: **return** π

- Symmetry in preference profile implies the bill with the broadest support/disapproval also yields the largest coalition
- Symmetry further implies largest cross-party coalition will be part of the output partition
- Selecting any satisfactory coalition (not necessarily the largest) in Step 3 maintains core stability
- Our implementation: replace largest with median-sized coalition

Boolean Hedonic Games

Core Finding Algorithms - PAC

- Same as Algorithm 4
- Only difference: the input is satisfactory coalitions derived from sample bills
- The output is consistent with the observed samples, therefore PAC stable [2]

k-Means Clustering

k-means clustering[3] divides a given set of samples $x_1, \cdots, x_n$ into k disjoint sets C, each described by the mean μ_j of the samples in the cluster; it produces a partition minimizing the *within-cluster sum-of-squares* (WCSS):

$$\sum_{i=1}^{n} \min_{\mu_j \in C} (||x_i - \mu_j||^2)$$

- General purpose
- Only need to find the best k
- Runs fast
- Assumes similar sized clusters

k-Means Clustering

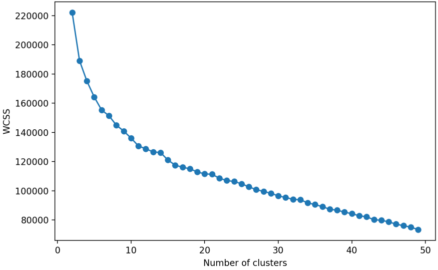
Model Construction

Distance between points

- Each politician correspond to a point
- Each bill acts as a feature
- A "for" vote takes value of 1, "against" -1, others 0

Finding the best k

- Elbow method: $k = 10$
- Average silhouette: $k = 2$



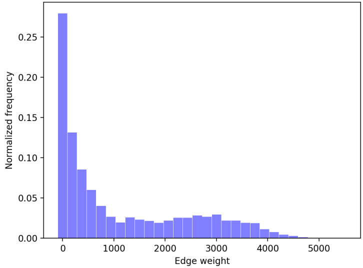
Stochastic Block Model

- A benchmark model in community detection
- Models dataset as a graph: parliament members as nodes, same/different votes as edge weights
- Assumes nodes in the same block shares same probability of being connected to other nodes
- Using Bayesian inference to find a partition that maximizes the likelihood of the observed network

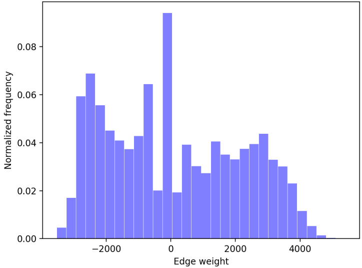
Stochastic Block Model

Modeling Edge Weights

- Positive edge weight: the number of times a pair of politicians voted together, either "for" or "against" a bill.
- Modeled as a geometric distribution



- Possibly negative edge weight: the difference between the number of times their votes agree and the number of times their votes disagree
- Modeled as a normal distribution



Variability among PAC Partitions

Model	Partition Size Mean (SD)	CV
Value Function	87 (0)	0
General Friends	13 (1.29)	0.10
Selective Friends	20 (1.83)	0.09
Selective Enemies	10 (0)	0
General Enemies	34 (0.84)	0.02
Boolean	85 (16.92)	0.2

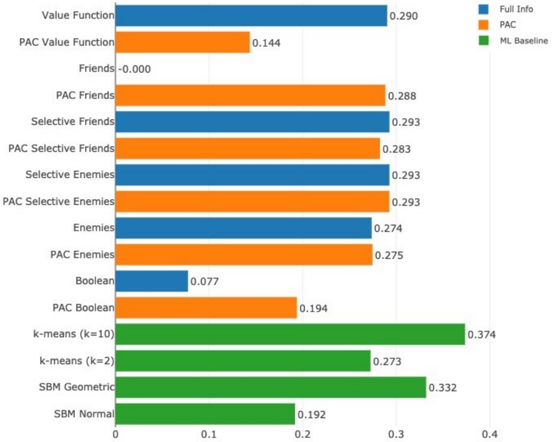
PAC model partition size statistics across 50 runs per model

Model	Pairwise AMI Mean (Min)	CV
Value Function	1 (1)	0
General Friends	0.78 (0.6)	0.09
Selective Friends	0.84 (0.66)	0.08
Selective Enemies	0.99 (0.97)	0.01
General Enemies	0.97 (0.93)	0.01
Boolean	0.18 (-0.06)	0.86

PAC Model Partition Pairwise AMI Statistics over 50 Runs per Model

Quantitative Analysis

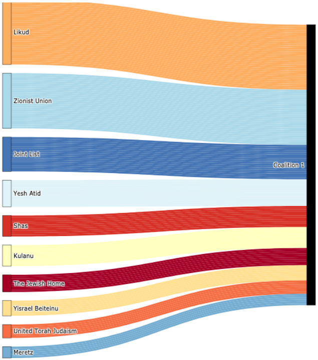
AMI



AMI between model partition and party affiliations

Quantitative Analysis

Friends Models - Full Information



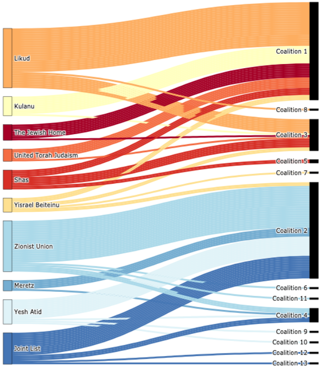
General friends model output

Full-information models

- $AMI_{\text{friends}} = 0$
- $AMI_{\text{selective friends}} = 0.293$
- More friendly $\rightarrow$ more likely to have grand coalition as final partition
- Major drawback of the full-information friends models: sensitive to the definition of "friends"

Quantitative Analysis

Friends Models - PAC



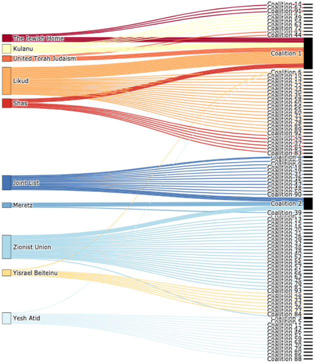
PAC general friends model output

PAC models

- $AMI_{\text{PAC friends}} = 0.288$
- $AMI_{\text{PAC selective friends}} = 0.283$
- PAC models dampen their sensitivity to the definition of friends through sampling

Quantitative Analysis

Boolean Models



Boolean model output

Full-info model

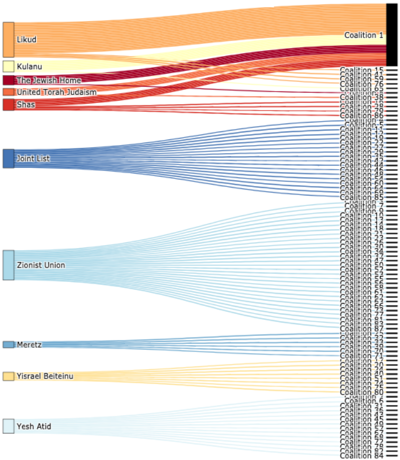
- $AMI_{\text{Boolean}} = 0.077$
- Too many "stranded" singleton coalitions

PAC model

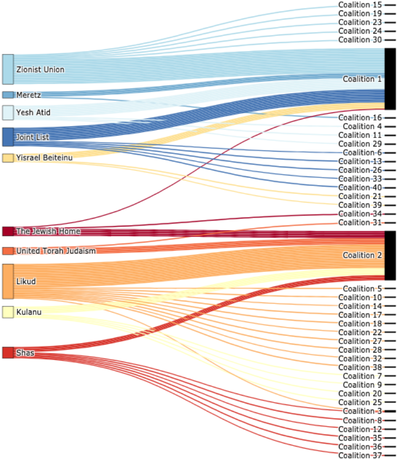
- $AMI_{\text{PAC Boolean}} = 0.194$
- Slightly better performance
- But of limited representativeness due to high variability

Quantitative Analysis

Handcrafted Value Function Models



PAC value function model output



Value function model output

Quantitative Analysis

Handcrafted Value Function Models

Full-info model

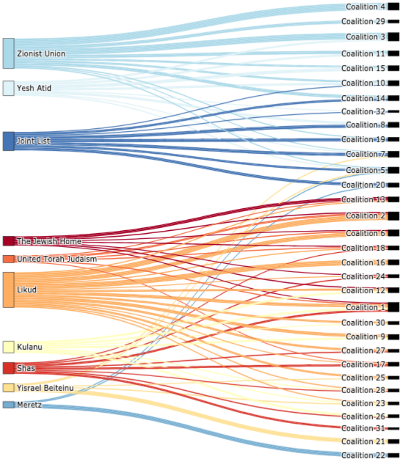
- $AMI_{\text{Value Fuction}} = 0.290$
- Identified a government coalition and an opposition coalition
- Precense of Yisrael Beiteinu members in coalition 1 (opposition parties) not in line with reality

PAC model

- $AMI_{\text{PAC Value Fuction}} = 0.144$
- Worst performing PAC model
- Many singleton coalitions
- Identified a government coalition
- Missing opposition coalition due to sampling more likely picking government majority bills — limitation of value function formulation

Quantitative Analysis

SMB - Normal Edge Weights

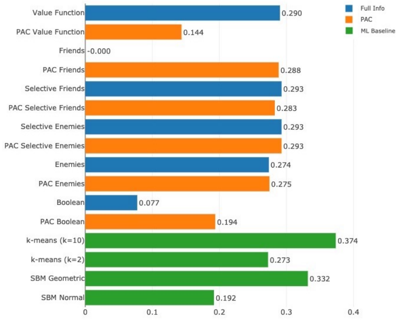


SBM normal model output

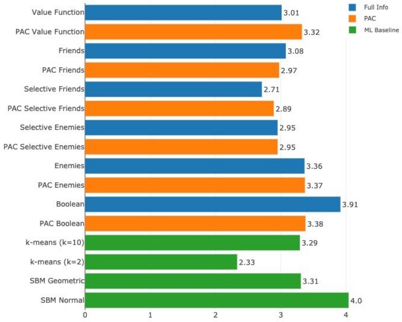
- $AMI_{\text{SBM normal}} = 0.192$
- Many small coalitions
- Left-wing/right-wing grouping observed, but fails to distinguish the government from the opposition

Quantitative Analysis

Model Selection



AMI: higher means closer to ground truth



VI: lower means closer to ground truth

Qualitative Analysis

Selected Models

PAC Models

- Selective Friends
- Selective Enemies
- Boolean

Comparison ML Models

- 2-group k-means
- 10-group k-means
- SBM Geometric

Qualitative Analysis

Criteria

- Coherence: separate coalition and opposition parties cleanly? (no mixed coalitions)
- Overal Structure: able to distinguish between the main government and opposition groups?
- Party Sub-groups: able to identify subgroups within parties?

Model	Coherence	Structure	Sub-group
PAC Selective Friends	✓	✓	✗
PAC Selective Enemies	✓	✓	✗
PAC Boolean	✓	✓	✗
2-group k-means	✗	✓	✗
10-group k-means	✗	✗	✗
SBM Geometric	✗	✗	✗

Conclusion

Summary

- ML methods: do not consider players' preferences and strategic behavior
- Game theoretic research: mostly theoretical or prescriptive with simulated data
- This thesis: a "descriptive" study of hedonic game theoretical models using real-world data of scale, and with ground truth
  - PAC models are able to recover overall structure & more coherent than ML models
  - PAC approach result is more robust than full-info approach

Future Research

- Apply PAC models on other parliaments, e.g. Dutch, Brazilian, US congress
- Apply other hedonic uncertainty models, e.g. Baysian core, on the Knesset dataset

Bibliography

[1] Jakub Sliwinski and Yair Zick
Learning Hedonic Games
Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI), pp.2730-2736, 2017

[2] Tushant Jha and Yair Zick
A Learning Framework for Distribution-Based Game-Theoretic Solution Concepts
2019. arXiv: 1903.08322.

[3] Tan, Pang-Ning and Steinbach, Michael and Karpatne, Anuj and Kumar, Vipin
Introduction to Data Mining (2Nd Edition)
Pearson, 2018.