Learning to Explain Voting Rules

Inwon Kang [1] Qishen Han [1] Lirong Xia [1]

[1]Rensselaer Polytechnic Institute

Motivation

- Providing a convincing *explanation* of the an election outcome is an important but difficult task.
- While the most popular voting rules (e.g. Borda, Plurality) are simple enough to understand, some others (e.g. Ranked Pairs, Kemeny-Young) can be more difficult for the general public to understand.
- Past works [1, 2] have approached this problem with deep models – however, deep models themselves are difficult to explain.
- Decision trees are an effective way to represent the decision paths taken by a voting rule for an election.
- **Question:** *How can we automate this process?*

Example

- Local government holds an election with some pre-determined voting rule.
- The government decides to generate an explanation for the voting outcome to get better support from the public.
- The public has no expertise on how the voting rule works, but they can understand simple notions. (e.g. "a swimming pool is more welcomed than a stadium")
- The government wants their explanation to be as simple and correct as possible.

Experimental Settings

Voting Rules: Copeland, Kemeny-Young, Ranked Pairs, Schulze

ML Models: Classification and Regularization Tree (CART), Generalized Optimal Sparse Decision Tree (GOSDT) [3], Hierarchical Shrinkage

To train the decision trees, we extract a feature called *pairwise margin* from the voting profiles

Pairwise Margin: the difference between every possible pair of pairs of candidates, i.e., the difference between the *pairwise victories of two pairs*

Example of pairwise margin: Given a profile with three votes of $[A \succ B \succ C]$, two votes of $[A \succ C \succ B]$, and two votes of $[B \succ A \succ C]$. Table 1 and Table 2 show the pairwise victories and (a subset of) pairwise margins of the profile respectively.

$A \succ B$	$A \succ C$	$B \succ A$	$B \succ C$	$C \succ A$	$C \succ B$
5	7	2	5	0	2

Table 1. Example of Pairwise Victory

$\lvert A \succ B\rvert - \lvert A \succ C\rvert$	-2
$\lvert A \succ B\rvert - \lvert B \succ A\rvert$	3
$\lvert A \succ B\rvert - \lvert B \succ C\rvert$	2
$\lvert A \succ B\rvert - \lvert C \succ A\rvert$	5
...	...

Table 2. Example of Pairwise Margin

Experiments

- A set of random preference profiles are generated for each voting rule.
- Each preference profile is then converted into the *pairwise margin* features.
- A decision tree is trained for each candidate to predict their victory.

	Copeland	Kemeny-Young	Ranked Pairs	Schulze
XGBoost	1.0	0.99	1.0	1.0
GOSDT	1.0	0.99	1.0	1.0
Scikit-Learn	0.82	0.96	0.95	0.95
HSTree	0.81	0.96	0.95	0.95

Table 3. Final averaged accuracy score of best trees learned by model for $m = 3, n = 100$

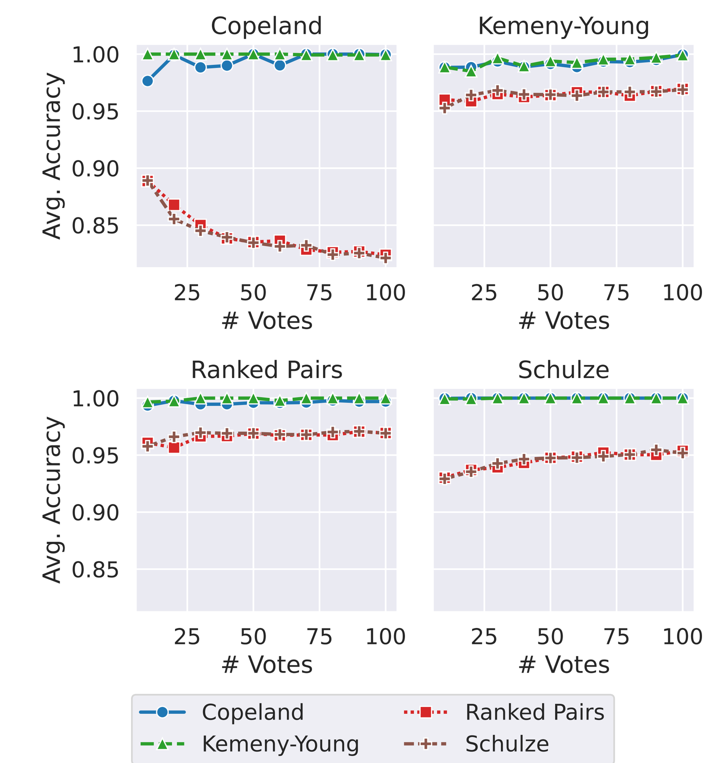

Figure 1. Averaged performance of models trained using randomly generated data over $n \in [10, 100]$

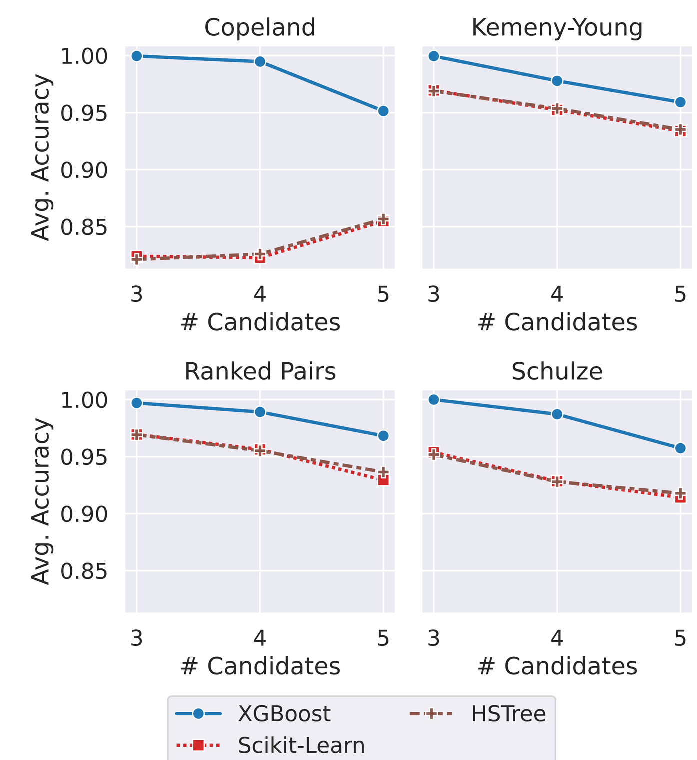

Figure 2. Averaged performance of models over $m \in [3, 5]$

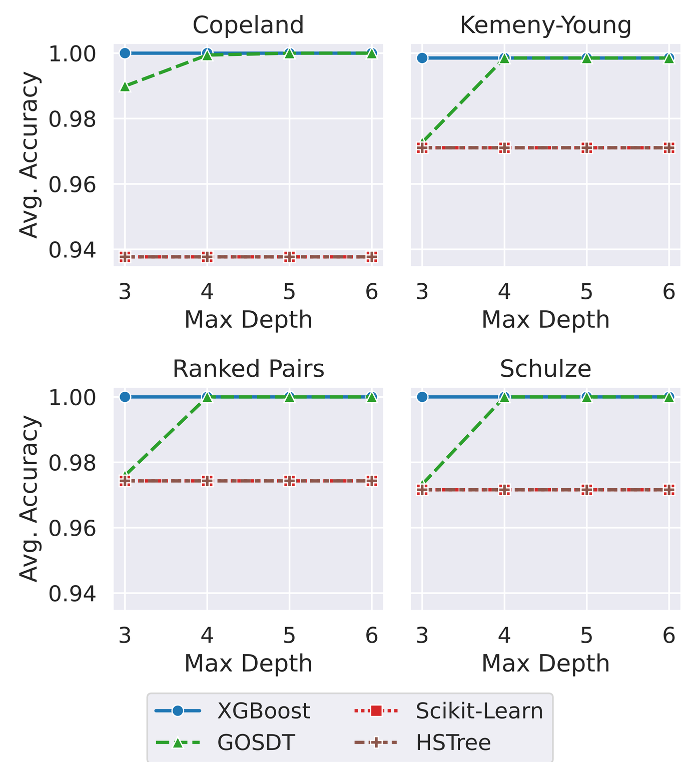

Figure 3. Averaged performance of models trained with $max\_depth \in [3, 6]$

Example Tree

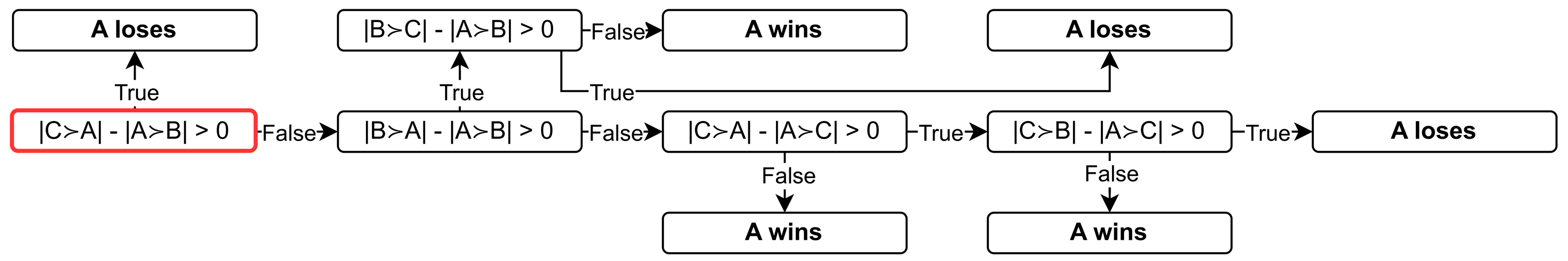

Figure 4. Example GOSDT model trained on Schulze with $m = 3$

Conclusion

- We proposed a methodology to learn to explain different voting rules using tree-based ML classifiers.
- We find that voting rules that satisfy the condorcet criterion could be well estimated by decision trees using the *pairwise margin* feature.
- For elections with $m = 3$, we were able to produce a provably correct decision tree for each outcome for KY, RP and Schulze.
- GOSDT algorithm was able to produce the most succinct trees – however, scaling issues were faced with larger profiles.
- As the candidate size grow past 3, the depth of the tree needed to be expanded much beyond 5 to be correct.

Future Work

- Can a new voting rule be created based on decision trees? (Instead of top-down, bottom up approach to designing a voting rule)
- Can a voter's preference be represented in a decision tree? If so, how can they be aggregated?

References

[1] Cem Anil and Xuchan Bao. Learning to elect. In *Advances in Neural Information Processing Systems*, pages 8006–8017, 2021.

[2] Dávid Burka, Clemens Puppe, László Szepesváry, and Attila Tasnádi. Voting: a machine learning approach. *European Journal of Operational Research*, 2021.

[3] Jimmy Lin, Chudi Zhong, Diane Hu, Cynthia Rudin, and Margo Seltzer. Generalized and scalable optimal sparse decision trees. In *Proceedings of the 37th International Conference on Machine Learning*, pages 6150–6160, 2020.