The Complexity of Learning Approval-Based Multiwinner Voting Rules

Ioannis Caragiannis, Karl Fehrs

Department of Computer Science, Aarhus University, Åbogade 34, 8200 Aarhus N, Denmark.
$\{$iannis, karl$\}$@cs.au.dk

Abstract

We study the PAC learnability of multiwinner voting, focusing on the class of approval-based committee scoring (ABCS) rules. These are voting rules applied on profiles with approval ballots, where each voter approves some of the candidates. According to ABCS rules, each committee of k candidates collects from each voter a score, that depends on the size of the voter's ballot and on the size of its intersection with the committee. Then, committees of maximum score are the winning ones. Our goal is to learn a target rule (i.e., to learn the corresponding scoring function) using information about the winning committees of a small number of sampled profiles. Despite the existence of exponentially many outcomes compared to single-winner elections, we show that the sample complexity is still low: a polynomial number of samples carries enough information for learning the target rule with high confidence and accuracy. Unfortunately, even simple tasks that need to be solved for learning from these samples are intractable. We prove that deciding whether there exists some ABCS rule that makes a given committee winning in a given profile is a computationally hard problem. Our results extend to the class of sequential Thiele rules, which have received attention due to their simplicity.

1 Introduction

Voting has been used for centuries to aggregate individual preferences into a common decision. In addition to its traditional use for electing governments or for decision making in management boards, it has also been proved useful in novel applications where individual ratings need to be summarized as collective knowledge. But, is there a general recipe on how preferences should be aggregated? Fortunately, there is no "golden" voting rule and this has led social choice theory —and, in particular, its modern computational branch (Brandt et al. 2016)— onto exciting research endeavours.

A popular approach has aimed, quite successfully, to evaluate voting rules in terms of desirable axioms they must satisfy. Well-known impossibilities, e.g., see Arrow (1951), showcase the limitations of this approach. Deviating from this *axiomatic* treatment, recent works view voting rules as optimized decision making methods, perhaps tailored to particular applications. In this context, the *data-driven design* of voting rules is a very natural approach. The goal is to derive a voting rule from a set of known preferences with the hope that the rule is equally well-suited to more general preferences. The current paper aims to study the potentials and limitations of this approach.

We focus on *multiwinner* voting rules (Faliszewski et al. 2017), which on input the preferences of n voters over m available candidates, return as outcome one or more committees of candidates of fixed size k. In particular, we study *approval-based* voting (Laslier and Sanver 2010; Lackner and Skowron 2021a), where the preference of a voter is simply the set of candidates she approves. And, more concretely, we consider the class of *approval-based committee scoring* (*ABCS*) rules, defined by Lackner and Skowron (2021b). An ABCS rule follows a common format. It employs a scoring function, according to which each voter awards a score to each committee of k candidates. This score depends on the ballot size (the number of candidates the voter approves) and the size of its intersection with the committee. Different scoring functions define different voting rules.

Deciding on the best ABCS rule depends on the application at hand. For example, under a rule that favours *individual excellence*, each voter assigns to each committee a score that is equal to the number of candidates in the committee the voter approves. Another rule could give just one point to each committee that has non-empty intersection with the voter's ballot; such a rule would promote *representation of voters*. In practice, situations with such a clear objective for a voting rule are extremely rare. Instead, it is usually easier to derive the characteristics of the desired rule from available data, in the form of preference profiles and corresponding desired winning committees. Arguably, the best rule for the particular application should at least agree with these data points, and, ideally, produce desirable outcomes for unknown preference profiles. Can such a data-driven selection of an ABCS rule be effective?

We explore this question using the *PAC* (probably approximately correct) *learning* framework. We follow a similar methodological approach with Procaccia et al. (2009), who addressed the same question for single-winner voting rules. In the terminology of PAC learning, we would like to determine the *sample complexity* of the class of ABCS rules. How many samples (profiles and corresponding winning committees) are necessary and sufficient so that an ABCS rule that agrees with these data points can be learnt? However, the answer to this question addresses our challenge only partially. Indeed, low sample complexity does not necessarily imply *efficient* learning, as the computational problem of finding an ABCS rule that fits the given data can be hard.

Our contribution and techniques. Our first main result states that the class of ABCS rules has only polynomial sample complexity (Section 4). Using the *multiclass fundamental theorem* in PAC learning (Theorem 4), we obtain our sample complexity bounds by proving asymptotically tight bounds on the *Natarajan dimension* of the class of ABCS rules. In our proofs,[1] we establish a connection between the Natarajan dimension and the number of different sign patterns of a set of linear functions. Then, a result in algebraic combinatorics —originally proved by Warren (1968) and later refined by Alon (1996)— is used to bound this number of sign patterns and, consequently, the Natarajan dimension and the sample complexity of the class of ABCS rules. To prove our main result, we need to make a simple but important observation that improves the use of the multiclass fundamental theorem (see the discussion at the end of Section 3).

On the negative side, we give strong evidence that efficient PAC learnability of ABCS rules is not possible. We show that given a profile of approval votes and a committee, deciding whether there is an ABCS rule that makes this committee winning is a coW$[1]$-hard[2] problem, when parameterized by the committee size k (Section 5). Our proof uses a quite involved reduction from INDEPENDENTSET, which, on input a graph, defines a profile consisting of several parts and a committee. Some of the parts of the profile guarantee that the only ABCS rule that can make the committee winning has a very particular form: it takes into account only votes with two candidates (ignoring the rest), and mimics the approval-based CC rule (henceforth, the CC rule), a famous rule that is inspired by the work of Chamberlin and Courant (1983). Then, the main part of the profile guarantees that the committee is indeed winning under this rule if and only if the graph does not have a large independent set. Our reduction can be modified to give coW$[1]$-hardness for the following *winner verification* problem: given a profile and a committee, is the committee winning under the CC rule? This result strengthens a recent one by Sonar, Dey, and Misra (2020).

We also consider sequential Thiele rules (Section 6). These can be thought of as greedy approximations of a subclass of ABCS rules which originate from the work of Thiele (1895). However, their definition is considerably different from ABCS rules, so that our sample complexity analysis techniques need revision. Still, we are able to show polynomial sample complexity bounds for learning sequential Thiele rules. Interestingly, the problem of deciding whether there is some sequential Thiele rule that makes a given committee winning in a given profile is now fixed-parameter tractable (parameterized by the committee size). Despite this seemingly positive result, we provide evidence that efficient learning is out of reach for sequential Thiele rules as well, by showing NP-hardness. We do so by a novel reduction from a structured version of 3SAT, which equates the ordering in which several candidates are greedily included in the winning committee with a boolean assignment to the 3SAT variables. As a corollary, our reduction can be modified to yield the first NP-hardness result for the winner verification problem for the sequential CC rule.

[1]Due to lack of space, many proofs are omitted.

[2]We follow standard notions from parameterized complexity theory, such as W-hierarchy hardness and fixed-parameter tractability; e.g., see Cygan et al. (2015).

Related work. The paper by Procaccia et al. (2009) is the most related to ours. Among other results, they prove that the class of single-winner positional voting rules is efficiently PAC-learnable. We remark that our setting is much more demanding. In particular, the number of possible outcomes is doubly exponential in our case, i.e., $2^{\binom{m}{k}}-1$, the number of all possible non-empty sets of winning committees, while it is just m in theirs (where fixed tie-breaking is used to produce a single winning candidate). Hence, even though we have not been able to prove efficiency of learning, the low sample complexity of ABCS rules is rather surprising.

PAC learning in voting has been considered, among other economic paradigms, by Jha and Zick (2020) and, in relation to the notion of the distortion, by Boutilier et al. (2015). Actually, the use of sign patterns has been inspired by the latter paper, even though the particular way in which we employ the result of Alon (1996) here is different.

More distantly related to our setting, the data-driven approach in the design of voting rules has been followed by a series of papers which focus on particular applications like rating (Caragiannis et al. 2019), evaluation of online surveys (Baumeister and Hogrebe 2019), and peer grading (Caragiannis, Krimpas, and Voudouris 2015, 2020). Other foundational work in this direction includes the papers by Faliszewski, Szufa, and Talmon (2018) and Xia (2013).

The computational complexity of multiwinner voting rules has received much attention; see the survey by Lackner and Skowron (2021a). The CC rule has been central in most related studies regarding ABCS rules. Procaccia, Rosenschein, and Zohar (2008) proved NP-hardness for the problem of deciding whether there is a committee that exceeds a given threshold under the CC rule on a given profile. The problem was later proved to be W$[2]$-hard by Betzler, Slinko, and Uhlmann (2013). Sonar, Dey, and Misra (2020) considered the question of whether a given candidate belongs to a winning committee for a given profile. They also prove that winner verification for the CC rule is coNP-hard, using a reduction from a variant of 3-HITTINGSET. We are not aware of published hardness results (of a similar spirit) for sequential Thiele rules.

2 Preliminaries

We consider approval-based voting with a set $\mathcal{N}$ of n *voters* (or *agents*), each approving a subset from a set Σ of m *candidates* (or *alternatives*). An approval-based multiwinner voting rule is defined for an integer k with $1 < k < m$. It takes as input a profile $P = \{\sigma_i\}_{i\in\mathcal{N}}$, where $\sigma_i \subset \Sigma$ is the non-empty set of alternatives approved by agent $i \in \mathcal{N}$ (or, her *approval vote*), and returns one or more k-sized subsets of Σ. We use the term *committee* to refer to any k-sized set of alternatives; then, the outcome of a multiwinner voting rule is one or more *winning* committees. We are interested in a specific class of multiwinner voting rules called *approval-based committee scoring* (ABCS) rules, defined by Lackner and Skowron (2021b). These rules are specified by a set of scoring parameters. Using these parameters, an agent's approval vote gives a score to each committee and the winning committees are those that receive the highest total score from all agents.

More formally, an ABCS rule is specified by a bivariate scoring function f. The parameter $f(x, y)$ denotes the non-negative score that an approval vote σ gives to the committee C when σ consists of y alternatives and has x alternatives in common with C. Notice that, under this interpretation, the function f needs only be defined over the set of pairs

$$\begin{aligned}\mathcal{X}_{m,k} = \{(x, y) : y \in [m-1], \\ x \in \max\{0, y-m+k\}, ..., \min\{k, y\}\}.\end{aligned}$$

Indeed, an approval vote with y alternatives can intersect with a committee in at least $\max\{0, y-m+k\}$ and at most $\min\{k, y\}$ alternatives.

Hence, formally $f : \mathcal{X}_{m,k} \to \mathbb{R}_{\geq 0}$. By definition, f is monotone non-decreasing in its first argument. To keep the presentation concise, we slightly overload notation and use f to refer both to the scoring function f and the ABCS rule specified by f. On input a profile $P = \{\sigma_i\}_{i\in\mathcal{N}}$, the ABCS rule f assigns a score of

$$\mathrm{sc}_f(C, P) = \sum_{i\in\mathcal{N}} f(|C \cap \sigma_i|, |\sigma_i|)$$

to each committee C; then, any committee of maximum score is winning in profile P under rule f. We write $f(P)$ for the set of all winning committees in profile P. We denote by $\mathcal{F}^{m,k}$ the class of ABCS rules with m alternatives and committee size k. We use the term *trivial* to refer to the ABCS rule f with $f(x, y) = 0$ for every $(x, y) \in \mathcal{X}_{m,k}$; obviously, all committees are winning in any profile under this rule.

An important subclass of ABCS rules is that of *Thiele* rules. Thiele rules use scoring functions f where the scoring parameter $f(x, y)$ does not depend on y. In this case, we can assume that f is univariate, defined over $\{0, 1, ..., k\}$, non-negative, and monotone non-decreasing. A specific Thiele rule that we use extensively is the CC rule that uses $f(0) = 0$ and $f(x) = 1$ for $x > 0$.

To bypass the necessity of computing the scores of all committees, *sequential Thiele rules* have been introduced to approximate ABCS rules by computing a winning committee in a greedy manner. Starting from an empty subcommittee, such rules build a winning committee gradually in k steps; in each step, they include an alternative that increases the score of the current subcommittee the most. The sequential Thiele rule that uses the univariate scoring function f computes the intermediate score of a set of alternatives A of size *up to* k on profile $P = \{\sigma_i\}_{i\in\mathcal{N}}$ as $\mathrm{sc}_f(A, P) = \sum_{i\in\mathcal{N}} f(|A \cap \sigma_i|)$. Then, given a profile P, a committee C is winning under the sequential Thiele rule f in profile P if there is an ordering of the alternatives in C, e.g., as $C = \{c_1, c_2, ..., c_k\}$, so that

$$c_i \in \operatorname*{arg\,max}_{c\in\Sigma\setminus\{c_1,...,c_{i-1}\}} \mathrm{sc}_f(\{c_1, ..., c_{i-1}\} \cup \{c\}, P),$$

for every $i \in [k]$. By this definition, a sequential Thiele rule can return more than one winning committees. We denote by $\mathcal{F}^k_{\text{seq}}$ the class of sequential Thiele rules for committee size k and any number of alternatives m higher than k. Again, the term "trivial" is reserved for the sequential Thiele rule that uses a scoring function f with $f(x) = 0$ for every x.

We conclude this section by defining the two decision problems we study: TARGETABCS and TARGETSEQTHIELE. In both, we are given a profile of approval votes $P = \{\sigma_i\}_{i\in\mathcal{N}}$ over the set Σ of m alternatives and a k-sized subset C of Σ. Our goal is to find a non-trivial rule f from $\mathcal{F}^{m,k}$ (for TARGETABCS) or $\mathcal{F}^k_{\text{seq}}$ (for TARGETSEQTHIELE), so that C is a winning committee in profile P according to f, or to return that no such rule exists.

3 PAC Learning Background

We follow a standard PAC learning model. In this model, a learning algorithm has to learn a target function from a hypothesis class $\mathcal{H}$ of functions which assign labels from the set Y to the points of a set Z. The learning algorithm is given a *training set of examples* T consisting of points from the sample space Z and labels from Y, which are sampled i.i.d. according to some probability distribution D over Z. We consider the *realizable case* and assume that there exists a function $h^* \in \mathcal{H}$ that is used to label the examples in the training set as $\{(z, h^*(z))\}_{z\in T}$. The learning algorithm outputs a function $h \in \mathcal{H}$. The error of function h is defined as

$$\mathrm{err}(h) = \Pr_{z\sim D}[h(z) \neq h^*(z)].$$

Clearly, $\mathrm{err}(h^*) = 0$. The terms "probably" and "approximately correct" refer to the existence of two parameters $\delta, \epsilon \in (0, 1)$, indicating the required *confidence* and *accuracy* of learning, respectively.

Definition 1 (PAC learnability)**.** *A hypothesis class $\mathcal{H}$ of functions from set Z to set Y is PAC-learnable if there exist a function $s : (0, 1)^2 \to \mathbb{N}$ —the sample complexity of $\mathcal{H}$— and a learning algorithm $\mathcal{A}$ with the following property: For every $\delta, \epsilon \in (0, 1)$, every distribution D over Z, and every function h^* from $\mathcal{H}$, on input a training set of at least $s(\delta, \epsilon)$ examples generated by D and labelled by h^*, the probability (over the choice of the training examples) that algorithm $\mathcal{A}$ returns a hypothesis h of error more than ϵ is at most δ.*

Extending the relation of the well-known VC dimension with the PAC learnability of boolean functions, Natarajan (1989) relates the sample complexity of a hypothesis class $\mathcal{H}$ to the notion of generalized (or Natarajan) dimension, which captures the combinatorial richness of $\mathcal{H}$. To define the Natarajan dimension, we need to define the notion of *shattering* first.

Definition 2 (shattering). *Let $\mathcal{H}$ be a class of functions from Z to Y. We say that $\mathcal{H}$* shatters *$T \subseteq Z$ if there exist two functions $g_1, g_2 \in \mathcal{H}$ such that*

1. *For all $z \in T$, $g_1(z) \neq g_2(z)$.*
2. *For all $S \subseteq T$, there exists $h_S \in \mathcal{H}$ such that $h_S(z) = g_1(z)$ for all $z \in S$, and $h_S(z) = g_2(z)$, for all $z \in T \setminus S$.*

Definition 3 (generalized dimension; Natarajan 1989, 1991). *Let $\mathcal{H}$ be a class of functions from a set Z to a set Y. The* generalized dimension *(or* Natarajan dimension*) of $\mathcal{H}$, denoted by $D_G(\mathcal{H})$, is the greatest integer d such that there exists a set of cardinality d that is shattered by $\mathcal{H}$.*

The relation of Natarajan dimension to sample complexity is given by the next statement. The upper bound depends also on the quantity $\Psi(\mathcal{H}) = \max_{z \in Z} |\{h(z) : h \in \mathcal{H}\}|$, which denotes the maximum number of labels in Y a point in Z can have according any function from $\mathcal{H}$.

Theorem 4 (multiclass fundamental theorem; see Shalev-Shwartz and Ben-David 2014). *There exist constants $C_1, C_2 > 0$ such that the hypothesis class $\mathcal{H}$ is PAC-learnable (assuming realizability) with sample complexity*

$$s(\delta, \epsilon) \geq C_1 \cdot \frac{D_G(\mathcal{H}) + \ln(1/\delta)}{\epsilon},$$

$$s(\delta, \epsilon) \leq C_2 \cdot \frac{D_G(\mathcal{H}) \cdot \ln\left(\frac{\Psi(\mathcal{H}) \cdot D_G(\mathcal{H})}{\epsilon}\right) + \ln(1/\delta)}{\epsilon}.$$

We remark that the upper bound in Theorem 4 is rather non-standard and, usually, the quantity $\Psi(\mathcal{H})$ is replaced by the larger quantity $|Y|$. However, observe that every function h in the hypothesis class $\mathcal{H}$ can be encoded by a function $\widehat{h}$ which maps the points of Z to positive integers. Assume an ordering of the elements in set Y. We say that an element y of Y is feasible for point $z \in Z$ if there exists a function $h \in \mathcal{H}$ so that $h(z) = y$. Now, for $z \in Z$, define $\widehat{h}(z)$ so that $h(z)$ is the $\widehat{h}(z)$-th (according to the ordering of Y) feasible element of Y for point z. Let $\widehat{\mathcal{H}}$ be the hypothesis class consisting of functions $\widehat{h}$ for each function $h \in \mathcal{H}$. Now, PAC learning in hypothesis classes $\mathcal{H}$ and $\widehat{\mathcal{H}}$ are equivalent tasks (ignoring the computational burden of encoding an outcome of function $h \in \mathcal{H}$ to the outcome of the corresponding function $\widehat{h}$ and vice-versa). The range of the functions in $\widehat{\mathcal{H}}$ has size only $\Psi(\mathcal{H})$ and our version of Theorem 4 follows by standard statements in PAC learning theory; e.g., see Shalev-Shwartz and Ben-David (2014), Theorem 29.3.

4 The Learnability of ABCS Rules

We are ready to prove that the class $\mathcal{F}^{m,k}$ of ABCS rules is PAC-learnable with sample complexity that depends polynomially on the number of alternatives m and the committee size k.

Theorem 5. *The class $\mathcal{F}^{m,k}$ of ABCS rules with m alternatives and committee size k is PAC-learnable with sample complexity $s(\delta, \epsilon)$ such that $s(\delta, \epsilon) \in \Omega\left(\epsilon^{-1}\left(|\mathcal{X}_{m,k}| + \ln(1/\delta)\right)\right)$ and $s(\delta, \epsilon) \in \mathcal{O}\left(\epsilon^{-1}\left(|\mathcal{X}_{m,k}|^2 k \ln m + |\mathcal{X}_{m,k}| \ln(1/\epsilon) + \ln(1/\delta)\right)\right)$.*

Notice that $|\mathcal{X}_{m,k}| \in \Theta(k(m-k))$; so the sample complexity grows only polynomially in m, k, and $1/\epsilon$, and logarithmically in $1/\delta$. Our proof of Theorem 5 will follow by Theorem 4 using upper and lower bounds on the Natarajan dimension of $\mathcal{F}^{m,k}$ and an upper bound on the maximum number of labels per example $\Psi(\mathcal{F}^{m,k})$ used in Theorem 4.

Upper-bounding the Natarajan dimension. To bound the Natarajan dimension, we will use an important result in algebraic combinatorics that bounds the number of different sign patterns a set of polynomials may have. Consider a set $\mathcal{L}$ of K polynomials $p_1, p_2, ..., p_K$, each defined over the ℓ real variables $x_1, x_2, ..., x_\ell$ (i.e., $p_i : \mathbb{R}^\ell \to \mathbb{R}$ for $i \in [K]$). A sign pattern $\mathbf{s}$ is just a vector of values in $\{-1, 0, +1\}$ with K entries. We say that the set of polynomials $\mathcal{L}$ realizes the sign pattern $\mathbf{s}$ if there exists values $x_1^*, x_2^*, ..., x_\ell^*$ for the variables $x_1, x_2, ..., x_\ell$ such that $\operatorname{sgn}(p_i(x_1^*, x_2^*, ..., x_\ell^*)) = \mathbf{s}_i$, for $i = 1, 2, ..., K$. Here, sgn is the signum function returning -1, 0, or $+1$, depending on whether its argument is negative, zero, or positive.

Clearly, the number of different sign patterns K polynomials may realize is at most 3^K. Usually, this is a very weak upper bound; Alon (1996) provides a much better bound, extending a previous statement due to Warren (1968).

Theorem 6 (Alon 1996, Warren 1968). *The number of different sign patterns a set of K polynomials of degree τ over ℓ real variables may realize is at most $\left(\frac{8e\tau K}{\ell}\right)^\ell$.*

Using Theorem 6, we can prove the next upper bound.

Lemma 7. $D_G(\mathcal{F}^{m,k}) \in \mathcal{O}(|\mathcal{X}_{m,k}|)$.

Proof. Assume that the Natarajan dimension of $\mathcal{F}^{m,k}$ is N. Thus, we have a set of N different profiles $\{P_j\}_{j \in [N]}$ and two voting rules $f, g \in \mathcal{F}^{m,k}$ such that $f(P_j) \neq g(P_j)$ for every $j \in [N]$. Hence, the two sets of committees $f(P_j)$ and $g(P_j)$ differ in at least one committee. For any profile P_j, this allows us to pick two committees $C_j \in f(P_j), D_j \in g(P_j)$ such that either $C_j \notin g(P_j)$ or $D_j \notin f(P_j)$ (not necessarily exclusively). We now define

$$L^j_{C_j, D_j}(s) = \mathrm{sc}_s(C_j, P_j) - \mathrm{sc}_s(D_j, P_j),$$

where s is any scoring function specifying a voting rule in $\mathcal{F}^{m,k}$. Let $P_j = \{\sigma_i^j\}_{i \in \mathcal{N}}$; then

$$L^j_{C_j, D_j}(s) = \sum_{i \in \mathcal{N}} s\left(|C_j \cap \sigma_i^j|, |\sigma_i^j|\right) - \sum_{i \in \mathcal{N}} s\left(|D_j \cap \sigma_i^j|, |\sigma_i^j|\right).$$

Hence, $L^j_{C_j, D_j}(s)$ is a linear function (a polynomial of degree 1) on the variables $s(x, y)$ for $(x, y) \in \mathcal{X}_{m,k}$. Let $\mathcal{L} = \{L^j_{C_j, D_j}(s) : j \in [N]\}$ be the set of linear functions defined for the N different profiles.

By our assumption of a Natarajan dimension of N and Definitions 2 and 3, we know that for each set $S \subseteq [N]$, there exists a voting rule $h_S \in \mathcal{F}^{m,k}$ such that $h_S(P_j) = f(P_j)$ for all $j \in S$ and $h_S(P_j) = g(P_j)$ for all $j \in [N] \setminus S$. Now, consider two different subsets S and S' of $[N]$ such that $S \not\subseteq S'$ (notice that this is without loss of generality). Let j^* be such that $j^* \in S$ and $j^* \notin S'$. Then,

$$\begin{aligned}
&\text{sgn}\left(L^{j^*}_{C_{j^*},D_{j^*}}(h_S)\right)\\
&= \text{sgn}\left(\text{sc}_{h_S}(C_{j^*},P_{j^*}) - \text{sc}_{h_S}(D_{j^*},P_{j^*})\right)\\
&= \begin{cases} 0, & \text{if } D_{j^*} \in f(P_{j^*})\\ +1, & \text{if } D_{j^*} \notin f(P_{j^*}) \end{cases}
\end{aligned}$$

and

$$\begin{aligned}
&\text{sgn}\left(L^{j^*}_{C_{j^*},D_{j^*}}(h_{S'})\right)\\
&= \text{sgn}\left(\text{sc}_{h_{S'}}(C_{j^*},P_{j^*}) - \text{sc}_{h_{S'}}(D_{j^*},P_{j^*})\right)\\
&= \begin{cases} -1, & \text{if } C_{j^*} \notin g(P_{j^*})\\ 0, & \text{if } C_{j^*} \in g(P_{j^*}) \end{cases}
\end{aligned}$$

Since, by the definition of committees C_{j^*} and D_{j^*}, it is either $C_{j^*} \notin g(P_{j^*})$ or $D_{j^*} \notin f(P_{j^*})$, we get that $\text{sgn}\left(L^{j^*}_{C_{j^*},D_{j^*}}(h_S)\right) \neq \text{sgn}\left(L^{j^*}_{C_{j^*},D_{j^*}}(h_{S'})\right)$. Hence, each of the 2^N voting rules h_S for $S \subseteq [N]$ —corresponding to a distinct assignment of values $s(x,y)$ for $(x,y) \in \mathcal{X}_{m,k}$— yields a different sign pattern to the set of polynomials $\mathcal{L}$.

We now apply Theorem 6 to $\mathcal{L}$ for $K = N$, $\tau = 1$, $\ell = |\mathcal{X}_{m,k}|$. This gives an upper bound of $\left(\frac{8eN}{|\mathcal{X}_{m,k}|}\right)^{|\mathcal{X}_{m,k}|}$ on the number of different sign patterns with entries in $\{-1, 0, +1\}$ for the set of polynomials $\mathcal{L}$. Hence,

$$2^N \leq \left(\frac{8eN}{|\mathcal{X}_{m,k}|}\right)^{|\mathcal{X}_{m,k}|}$$

and, equivalently,

$$|\mathcal{X}_{m,k}| \cdot 2^{N/|\mathcal{X}_{m,k}|} - 8eN \leq 0. \tag{1}$$

The derivative of the LHS of (1) with respect to N is $2^{N/|\mathcal{X}_{m,k}|} \cdot \ln 2 - 8e$, i.e., increasing in N. For $N = 8|\mathcal{X}_{m,k}|$, its value is $2^8 \cdot \ln 2 - 8e$, i.e., already positive. Hence, for $N > 8|\mathcal{X}_{m,k}|$, the LHS of (1) is larger than $(2^8 - 64e) \cdot |\mathcal{X}_{m,k}| > 0$, contradicting inequality (1). Thus, the condition $N \leq 8|\mathcal{X}_{m,k}|$ is necessary so that inequality (1) holds and the proof of Lemma 7 is complete. □

A tight lower bound. We now prove an asymptotically tight lower bound on $D_G(\mathcal{F}^{m,k})$. In our proof, we construct a large set of profiles that can be shattered by the set of ABCS rules $\mathcal{F}^{m,k}$.

Lemma 8. $D_G(\mathcal{F}^{m,k}) \in \Omega(|\mathcal{X}_{m,k}|)$.

Proof. For a given $m \geq 3$ and k such that $2 \leq k \leq m-1$, consider the set of alternatives

$$\Sigma = \{a, b_1, \ldots, b_{k-1}, c, d_1, \ldots, d_{m-k-1}\}.$$

Our goal is to define a set of profiles, where for each profile we are able to pick rules from $\mathcal{F}^{m,k}$ such that either committee $A = \{a, b_1, \ldots, b_{k-1}\}$ or committee $C = \{b_1, \ldots, b_{k-1}, c\}$ is the single winning committee under the respective rule.

Let $\mathcal{T}_{m,k}$ be the following set of pairs:

$$\begin{aligned}
\mathcal{T}_{m,k} = \Big\{&(x,y) : y \in \{2, \ldots, m-1\},\\
&x \in \{1 + \max\{0, y-m+k\}, \ldots, \min\{k,y\}\}\Big\}\\
&\setminus \{(k,k)\}.
\end{aligned}$$

Even though some of the pairs of set $\mathcal{X}_{m,k}$ have been omitted from $\mathcal{T}_{m,k}$, they have asymptotically the same size as the next lemma indicates.

Lemma 9. $|\mathcal{T}_{m,k}| \in \Omega(|\mathcal{X}_{m,k}|)$.

We now define the set of profiles $\{P_{xy}\}_{(x,y)\in\mathcal{T}_{m,k}}$. Each profile P_{xy} contains four approval votes:

$$\sigma_1^{xy} = \{a\}, \sigma_2^{xy} = A, \sigma_3^{xy} = C,$$

and

$$\sigma_4^{xy} = \{b_1, \ldots, b_{x-1}, c, d_1, \ldots, d_{y-x}\}.$$

We introduce the family of rules $F \subseteq \mathcal{F}^{m,k}$ which, for every subset $S \subseteq \mathcal{T}_{m,k}$, contains the voting rule h_S defined as:

$$\begin{aligned}
h_S(1,1) &= 1,\\
h_S(k,k) &= 4k-1,\\
h_S(\max\{0, y-m+k\}, y) &= 0, \qquad \text{for } y \in [m-1],\\
h_S(x,y) - h_S(x-1,y) &= \begin{cases} 0, & (x,y) \in S,\\ 2, & (x,y) \in \mathcal{T}_{m,k} \setminus S. \end{cases}
\end{aligned}$$

Notice that the function h_S is monotonically non-decreasing in its first argument, as required by the definition of voting rules in $\mathcal{F}^{m,k}$.

We will show that the family F shatters the set of profiles $\{P_{xy}\}_{(x,y)\in\mathcal{T}_{m,k}}$. To do so, we will make use of the following two lemmas. Lemma 10 guarantees that no other committee besides A and C is ever winning in any profile of $\{P_{xy}\}_{(x,y)\in\mathcal{T}_{m,k}}$. Lemma 11 identifies the winning committee among A and C in each of these profiles for every voting rule in set F.

Lemma 10. *For every committee $X \neq A, C$, every profile $P_{xy} \in \{P_{xy}\}_{(x,y)\in\mathcal{T}_{m,k}}$ and any voting rule $h \in F$, it holds that $sc_h(A, P_{xy}) - sc_h(X, P_{xy}) \geq 1$.*

Lemma 11. *Let $S \subseteq \mathcal{T}_{m,k}$. For every profile $P_{xy} \in \{P_{xy}\}_{(x,y)\in\mathcal{T}_{m,k}}$, it holds that*

$$sc_{h_S}(A, P_{xy}) - sc_{h_S}(C, P_{xy}) = \begin{cases} 1, & (x,y) \in S\\ -1, & (x,y) \in \mathcal{T}_{m,k} \setminus S \end{cases}$$

Together, Lemmas 10 and 11 imply that, when applied to profile P_{xy}, the voting rule h_S returns

- A as the unique winning committee if $(x,y) \in S$, and
- C as the unique winning committee if $(x,y) \in \mathcal{T}_{m,k} \setminus S$.

By Definition 2, this implies that the family F (and, consequently, the family $\mathcal{F}^{m,k}$) shatters the set of profiles $\{P_{xy}\}_{(x,y)\in\mathcal{T}_{m,k}}$. Indeed, it suffices to define functions g_1 and g_2 as $g_1 = h_{\mathcal{T}_{m,k}}$ and $g_2 = h_\emptyset$, while the set of profiles $\{P_{xy}\}_{(x,y)\in\mathcal{T}_{m,k}}$ plays the role of set T in Definition 2. By Definition 3, we conclude that the Natarajan dimension of $\mathcal{F}^{m,k}$ is at least $|\mathcal{T}_{m,k}|$. Lemma 8 now follows from Lemma 9. □

Bounding the maximum number of labels. To bound $\Psi(\mathcal{F}^{m,k})$, we use again sign patterns and Theorem 6. Let P be a profile and consider the square matrix M_s whose columns and rows correspond to committees. The entry $M_s(A, C)$ at the row corresponding to committee A and the column corresponding to committee C has the outcome of the linear function

$$M_s(A, C) = \mathrm{sc}_s(A, P) - \mathrm{sc}_s(C, P).$$

The important observation is the following: A set $\mathcal{W}$ of committees is the set of winning committees for some voting rule s only if the following is true:

- for every $A \in \mathcal{W}$, $M_s(A, C) \geq 0$ for every committee C, and
- for every $A \notin \mathcal{W}$, there exists a committee C such that $M_s(A, C) < 0$.

Hence, the number of different winning sets is upper-bounded by the number of different sign patterns the entries of matrix M_s can realize. By Theorem 6, we get that this number, which upper-bounds $\Psi(\mathcal{F}^{m,k})$, is

$$\Psi(\mathcal{F}^{m,k}) \leq \left(\frac{8e\binom{m}{k}^2}{|\mathcal{X}_{m,k}|} \right)^{|\mathcal{X}_{m,k}|}.$$

Taking logarithms in both sides, we obtain the following.

Lemma 12. $\ln \Psi(\mathcal{F}^{m,k}) \in \mathcal{O}(|\mathcal{X}_{m,k}| \cdot k \cdot \ln m)$.

Putting all together. Now, the lower bound on sample complexity in Theorem 5 follows by the first inequality in Theorem 4 using our lower bound on the Natarajan dimension of $\mathcal{F}^{m,k}$ in Lemma 8. The upper bound in Theorem 5 follows by the second inequality in Theorem 4 using the upper bound on the Natarajan dimension from Lemma 7 and the bound on quantity $\ln \Psi(\mathcal{F}^{m,k})$ from Lemma 12.

5 The Complexity of TARGETABCS

Unfortunately, despite the low sample complexity of the class of ABCS rules, learning from samples is notoriously hard. We prove this for TARGETABCS, which captures the elementary task of learning from a single sample. The next statement uses a polynomial-time reduction from (the complement of) the INDEPENDENTSET problem.

Definition 13 (INDEPENDENTSET)**.** *Given a graph G and a positive integer K, decide whether G contains a set of at least K nodes that form an independent set.*

INDEPENDENTSET is known to be W[1]-hard, parameterized by the independent set size (Cygan et al. 2015, Theorem 13.18).

Theorem 14. TARGETABCS *parameterized by the committee size k is* coW[1]*-hard.*

Proof. For a given instance of INDEPENDENTSET consisting of a graph G and an integer K, we construct an instance of TARGETABCS with $k = K$ such that there is a non-trivial rule $f \in \mathcal{F}^{m,k}$ that outputs A as a winning committee in P if and only if G contains no independent set of size K.

Let Δ denote the maximum degree among the vertices of G. We can assume that $\Delta \geq 2$, since INDEPENDENTSET would be trivially solvable in polynomial time otherwise. As a first step in our construction, we modify G to another graph G' as follows. For every vertex $v \in V$, we add $\Delta - \deg(v)$ dummy vertices that are adjacent only to v. Let $G' = (V', E')$ be the resulting graph and let $|V'| = r$.

Without loss of generality, we can assume that V' is the set of positive integers in $[r]$. The set of alternatives Σ consists of alternatives a_i and b_i for every vertex $i \in V'$, and the additional alternatives c and d. Let $A = \{a_1, a_2, ..., a_k\}$. The profile P consists of three parts:

- Part 1 consists of vote $\{b_i, b_j\}$ for every edge $(i, j) \in E'$.
- Part 2 consists of $k\Delta - 1$ copies of each of the following votes: vote $\{a_i, b_j\}$ for every $i, j \in [r]$, votes $\{a_i, c\}$ and $\{b_i, d\}$ for every $i \in [r]$, vote $\{a_1, d\}$, and vote $\{c, d\}$.
- Part 3 consists of a vote containing alternatives d, a_1, a_2, ..., a_{x-1}, and $y - x$ additional alternatives among a_{k+1}, a_{k+2}, ..., a_r, b_1, ..., b_r, for every $(x, y) \in \mathcal{X}_{m,k} \setminus (\{(\max\{0, y - m + k\}, y) : y \in [m-1]\} \cup \{(1, 2), (2, 2)\})$.

We use P_1, P_2, and P_3 to denote the three subprofiles of votes in part 1, 2, and 3, respectively.[3]

Parts 2 and 3 of profile P have important properties that are given in Lemmas 15 and 16.

Lemma 15. *Let $f \in \mathcal{F}^{m,k}$ and $C = \{a_1, ..., a_{k-1}, d\}$. Then, $sc_f(A, P_3) = sc_f(C, P_3)$ if $f(x, y) = 0$ for every $(x, y) \in \mathcal{X}_{m,k} \setminus \{(1, 2), (2, 2)\}$ and $sc_f(A, P_3) < sc_f(C, P_3)$, otherwise.*

Lemma 16. *Let $f \in \mathcal{F}^{m,k}$ and $C = \{a_1, ..., a_{k-1}, d\}$. Then, $sc_f(A, P_2) = sc_f(C, P_2)$ if $f(1, 2) = f(2, 2)$ and $sc_f(A, P_2) < sc_f(C, P_2)$, otherwise.*

As no vote in part 1 of profile P includes any alternatives in committees A and C, Lemmas 15 and 16 imply that a non-trivial rule $f \in \mathcal{F}^{m,k}$ can make committee A winning in P only if it satisfies $f(1, 2) = f(2, 2) > 0$ and $f(x, y) = 0$ for any pair (x, y) of $\mathcal{X}_{m,k}$ different than $(1, 2)$ and $(2, 2)$. We complete the proof assuming —without loss of generality— that f furthermore satisfies $f(1, 2) = f(2, 2) = 1$.

Claim 17. *It holds that $sc_f(A, P) = (k\Delta - 1)(kr + k + 1)$.*

Consider a committee B and let t be the number of its alternatives from $\{b_1, ..., b_r\}$.

Lemma 18. *If $t < k$, $sc_f(B, P) \leq (k\Delta - 1)(kr + k + 1)$.*

By Claim 17 and Lemma 18, if committee B has score higher than $\mathrm{sc}_f(A, P)$, then it must be $t = k$. We conclude the proof by reasoning about $\mathrm{sc}_f(B, P)$ in this case.

Claim 19. *Let B be a committee with $t = k$. Then, $sc_f(B, P_2) = (k\Delta - 1)(kr + k)$.*

Lemma 20. *Consider any committee B with $t = k$. If G has no independent set of size k, then $sc_f(B, P_1) \leq k\Delta - 1$.*

[3] By restricting profiles to have only parts 1 and 2, our reduction yields coW[1]-hardness of winner verification for the CC rule.

Proof. Let S be the set of vertices in G' to which the alternatives in B correspond. Then, $\mathrm{sc}_f(B, P_1)$ is equal to the number of edges in G' that are incident to the vertices of S. These vertices have degree either 1 or Δ. If one of them has degree 1, then $\mathrm{sc}_f(B, P_1) \leq (k-1)\Delta + 1 \leq k\Delta - 1$. Otherwise, if all of them have degree Δ in G', then they correspond to vertices of G. Since G has no independent set of size k, at least two vertices of S are connected by an edge in G and, consequently, in G'. Hence, the number of edges incident to the vertices of S and, consequently, $\mathrm{sc}_f(B, P_1)$ is at most $k\Delta - 1$. □

By Claim 19 and Lemma 20, we obtain that if G has no independent set of size k, $\mathrm{sc}_f(B, P) \leq (k\Delta-1)(kr+k+1)$. Thus, by Claim 17, A is a winning committee in this case.

Now, assume that G has an independent set of size k. This implies that G' has an independent set S of k vertices of degree Δ. Now, consider the committee B consisting of the alternatives that correspond to the vertices of S. As the number of edges that are incident to vertices of S is $k\Delta$, $\mathrm{sc}_f(B, P_1) = k\Delta$ as well. Then, by Claims 17 and 19, we have $\mathrm{sc}_f(B, P) = 1 + (k\Delta - 1)(kr + k + 1) > \mathrm{sc}_f(A, P)$ indicating that A is not winning. The proof of correctness of our reduction is now complete. □

6 Sequential Thiele Rules

We now turn our attention to the PAC learnability of sequential Thiele rules and related complexity questions. In Section 4, we saw how the sign of a single linear function can be used to compare the score of two committees in a profile according to an ABCS rule. Due to the different definition of sequential Thiele rules, such a direct comparison is not possible. Still, deciding whether a committee is winning can be done by examining the signs of a *block* of linear functions. This will be our main tool to show that TARGETSEQTHIELE is in FPT and that the class $\mathcal{F}^k_{\text{seq}}$ is PAC-learnable.

Assume a generic ordering of the alternatives in Σ. For a committee A and integer $i \in [k]$, we denote by $A(i)$ the i-th alternative of committee A (according to the generic ordering). For a committee A, permutation $\pi : [k] \rightarrow [k]$, and integer $i \in [k]$, the notation $A[\pi, i]$ is used to denote the set of alternatives $\cup_{j=1}^{i}\{A(\pi(j))\}$.

Now, assume that the sequential Thiele rule s returns committee A as winning when applied on profile P. Assume that the order in which rule s decides the alternatives in A as winning is given by permutation π: at step i, the rule includes alternative $A(\pi(i))$ in the winning committee. By the definition of the sequential Thiele rule s, this decision can be expressed by the set of inequalities

$$\mathrm{sc}_s(A[\pi, i], P) - \mathrm{sc}_s(A[\pi, i-1] \cup \{a\}, P) \geq 0, \qquad (2)$$

for every alternative $a \in \Sigma \setminus A[\pi, i]$. Non-negativity is necessary and sufficient so that alternative $A(\pi(i))$ is (weakly) preferred for inclusion in the winning committee at step i over any alternative $a \in \Sigma \setminus A[\pi, i]$.

For a sequential Thiele rule s, committee A, and permutation π, we define the block $B^P_{A,\pi}(s)$ consisting of the LHS expression of equation (2) for every $i \in [k]$ and every alternative $a \in \Sigma \setminus A[\pi, i]$. By the discussion above, committee A is winning in profile P under rule s if and only if there is a permutation π so that all expressions in block $B^P_{A,\pi}(s)$ are non-negative. Otherwise, if the block $B^P_{A,\pi}(s)$ contains a negative expression for every permutation π, committee A is not winning.

We can use this observation to show that TARGETSEQTHIELE can be solved in time $k! \cdot \mathrm{poly}(m, n)$ and, hence, is fixed-parameter tractable. This can be done as follows. For each of the $k!$ permutations π, consider the linear program that has parameters $s(1)$, ..., $s(k)$ as variables (assuming $s(0) = 0$) and its constraints require that each expression of block $B^P_{A,\pi}(s)$ —each of which is a linear function of the variables— is non-negative and, furthermore, $0 \leq s(1) \leq ... \leq s(k)$ and $s(k) \geq 1$ to ensure non-negativity, monotonicity, and non-triviality. If the linear program is feasible for some permutation π, then the corresponding scoring function s gives a sequential Thiele rule that makes A a winning committee in profile P. Otherwise, no such rule exists. Checking feasibility can be done in polynomial time using well-known algorithms for linear programming. The next statement summarizes this discussion.

Theorem 21. TARGETSEQTHIELE *is in FPT.*

By adapting our analysis in Section 4 and using blocks of linear functions to witness winning committees as discussed above, we can prove Theorem 22. The sample complexity of sequential Thiele rules is polynomial too, but potentially higher compared to ABCS rules. The notation $\widetilde{\mathcal{O}}$ hides logarithmic terms in m, k, $1/\delta$, and $1/\epsilon$.

Theorem 22. *The class $\mathcal{F}^k_{seq}$ of sequential Thiele rules with m alternatives and committee size k is PAC-learnable with sample complexity $s(\delta, \epsilon)$ such that $s(\delta, \epsilon) \in \Omega\left(\epsilon^{-1}\left(k + \ln(1/\delta)\right)\right)$ and $s(\delta, \epsilon) \in \widetilde{\mathcal{O}}\left(\epsilon^{-1}k^4\right)$.*

Again, the proof of Theorem 22 relies on Theorem 4 and exploits bounds on quantities $D_G(\mathcal{F}^k_{\text{seq}})$ and $\Psi(\mathcal{F}^k_{\text{seq}})$.

The last statement of this section is negative and provides evidence that learning in class $\mathcal{F}^k_{\text{seq}}$ is hard as well. The proof employs a novel reduction from a structured version of 3SAT, known to be NP-hard (Yoshinaka 2005).

Theorem 23. TARGETSEQTHIELE *is* NP*-hard.*

7 Concluding Remarks

We studied complexity aspects of learning ABCS and sequential Thiele rules. In a nutshell, our results suggest that learning from these classes is feasible in the PAC learning framework but —in a worst-case sense— only in computational inefficient ways. We believe that our techniques for assessing PAC learnability can be extended to other rules. Faliszewski et al. (2019) define a hierarchy of classes of ranking-based multiwinner voting rules that are specified using scoring functions. These are natural candidates for extending our analysis. We also remark that, en route to proving hardness of TARGETABCS and TARGETSEQTHIELE, parts of our reductions show hardness of winner verification for the CC and the sequential CC rule. The next statement summarizes these byproduct results.

Theorem 24. *Winner verification is* coW[1]*-hard for the CC rule and* NP*-hard for the sequential CC rule.*

Acknowledgements

We thank Piotr Faliszewski for his insights on the complexity of sequential Thiele rules and for pointers to the literature. We also thank Allan Grønlund for helpful discussions on the multiclass fundamental theorem and Krzysztof Sornat for his comments on our AAAI preprint. Furthermore, we thank the AAAI reviewers for their feedback.

References

Alon, N. 1996. Tools from Higher Algebra. In *Handbook of Combinatorics (Vol. 2)*, 1749–1783. MIT Press.

Arrow, K. J. 1951. *Social Choice and Individual Values*. Wiley.

Baumeister, D.; and Hogrebe, T. 2019. How Hard Is the Manipulative Design of Scoring Systems? In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, 74–80.

Betzler, N.; Slinko, A.; and Uhlmann, J. 2013. On the Computation of Fully Proportional Representation. *Journal of Artificial Intelligence Research*, 47: 475–519.

Boutilier, C.; Caragiannis, I.; Haber, S.; Lu, T.; Procaccia, A. D.; and Sheffet, O. 2015. Optimal Social Choice Functions: A Utilitarian View. *Artificial Intelligence*, 227: 190–213.

Brandt, F.; Conitzer, V.; Endress, U.; Lang, J.; and Procaccia, A. D., eds. 2016. *Handbook of Computational Social Choice*. Cambridge University Press.

Caragiannis, I.; Chatzigeorgiou, X.; Krimpas, G. A.; and Voudouris, A. A. 2019. Optimizing Positional Scoring Rules for Rank Aggregation. *Artificial Intelligence*, 267: 58–77.

Caragiannis, I.; Krimpas, G. A.; and Voudouris, A. A. 2015. Aggregating Partial Rankings with Applications to Peer Grading in Massive Online Open Courses. In *Proceedings of the 14th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 675–683.

Caragiannis, I.; Krimpas, G. A.; and Voudouris, A. A. 2020. How Effective Can Simple Ordinal Peer Grading Be? *ACM Transactions on Economics and Computation*, 8(3).

Chamberlin, J. R.; and Courant, P. N. 1983. Representative Deliberations and Representative Decisions: Proportional Representation and the Borda Rule. *American Political Science Review*, 77(3): 718–733.

Cygan, M.; Fomin, F. V.; Kowalik, L.; Lokshtanov, D.; Marx, D.; Pilipczuk, M.; Pilipczuk, M.; and Saurabh, S. 2015. *Parameterized Algorithms*. Springer.

Faliszewski, P.; Skowron, P.; Slinko, A.; and Talmon, N. 2019. Committee Scoring Rules: Axiomatic Characterization and Hierarchy. *ACM Transactions on Economics and Computation*, 7(1).

Faliszewski, P.; Skowron, P.; Slinko, A. M.; and Talmon, N. 2017. Multiwinner Voting: A New Challenge for Social Choice Theory. In Endriss, U., ed., *Trends in Computational Social Choice*, 27–47. AI Access.

Faliszewski, P.; Szufa, S.; and Talmon, N. 2018. Optimization-Based Voting Rule Design: The Closer to Utopia the Better. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, 32–40.

Jha, T.; and Zick, Y. 2020. A Learning Framework for Distribution-Based Game-Theoretic Solution Concepts. In *Proceedings of the 21st ACM Conference on Economics and Computation (EC)*, 355–377.

Lackner, M.; and Skowron, P. 2021a. Approval-Based Committee Voting. arXiv:2007.01795.

Lackner, M.; and Skowron, P. 2021b. Consistent Approval-Based Multi-Winner Rules. *Journal of Economic Theory*, 192: 105173.

Laslier, J.-F.; and Sanver, M. R., eds. 2010. *Handbook on Approval Voting*. Springer.

Natarajan, B. K. 1989. On Learning Sets and Functions. *Machine Language*, 4(1): 67–97.

Natarajan, B. K. 1991. *Machine Learning: A Theoretical Approach*. Morgan Kaufmann Publishers Inc.

Procaccia, A. D.; Rosenschein, J. S.; and Zohar, A. 2008. On the Complexity of Achieving Proportional Representation. *Social Choice and Welfare*, 30: 353–362.

Procaccia, A. D.; Zohar, A.; Peleg, Y.; and Rosenschein, J. S. 2009. The Learnability of Voting Rules. *Artificial Intelligence*, 173(12): 1133–1149.

Shalev-Shwartz, S.; and Ben-David, S. 2014. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press.

Sonar, C.; Dey, P.; and Misra, N. 2020. On the Complexity of Winner Verification and Candidate Winner for Multiwinner Voting Rules. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, 89–95.

Thiele, T. N. 1895. On Multiple Choice (in Danish). *Bulletin of the Royal Danish Academy of Sciences and Letters*, 415–441.

Warren, H. E. 1968. Lower Bounds for Approximation by Nonlinear Manifolds. *Transactions of the American Mathematical Society*, 133(1): 167–178.

Xia, L. 2013. Designing Social Choice Mechanisms Using Machine Learning. In *Proceeedings of the 12th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 471–474.

Yoshinaka, R. 2005. Higher-Order Matching in the Linear λ Calculus in the Absence of Constants Is NP-Complete. In *Proceedings of the 16th International Conference on Term Rewriting and Applications*, 235–249.