Artificial Intelligence 231 (2016) 1–16

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Voting rules as error-correcting codes ☆

Ariel D. Procaccia, Nisarg Shah *, Yair Zick

Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA

A R T I C L E I N F O

Article history:
Received 21 January 2015
Received in revised form 13 October 2015
Accepted 20 October 2015
Available online 27 October 2015

Keywords:
Social choice
Voting
Ground truth
Adversarial noise
Error-correcting codes

A B S T R A C T

We present the first model of optimal voting under adversarial noise. From this viewpoint, voting rules are seen as error-correcting codes: their goal is to correct errors in the input rankings and recover a ranking that is close to the ground truth. We derive worst-case bounds on the relation between the average accuracy of the input votes, and the accuracy of the output ranking. Empirical results from real data show that our approach produces significantly more accurate rankings than alternative approaches.

1. Introduction

Social choice theory develops and analyzes methods for aggregating the opinions of individuals into a collective decision. The prevalent approach is motivated by situations in which opinions are *subjective*, such as political elections, and focuses on the design of voting rules that satisfy normative properties [1].

An alternative approach, which was proposed by the marquis de Condorcet in the 18th Century, had confounded scholars for centuries (due to Condorcet's ambiguous writing) until it was finally elucidated by Young [34]. The underlying assumption is that the alternatives can be *objectively* compared according to their true quality. In particular, it is typically assumed that there is a ground truth *ranking* of the alternatives. Votes can be seen as noisy estimates of the ground truth, drawn from a specific *noise model*. For example, Condorcet proposed a noise model where — roughly speaking — each voter (hereinafter, *agent*) compares every pair of alternatives, and orders them correctly (according to the ground truth) with probability $p > 1/2$; today an equivalent model is attributed to Mallows [26]. Here, it is natural to employ a voting rule that always returns a ranking that is *most likely* to coincide with the ground truth, that is, the voting rule should be a *maximum likelihood estimator (MLE)*.

Although Condorcet could have hardly foreseen this, his MLE approach is eminently applicable to crowdsourcing and human computation systems, which often employ voting to aggregate noisy estimates; EteRNA [24] is a wonderful example, as explained by Procaccia et al. [30]. Consequently, the study of voting rules as MLEs has been gaining steam in the last decade [16,15,19,33,32,25,30,2–4,27,12,13].

Despite its conceptual appeal, a major shortcoming of the MLE approach is that the MLE voting rule is specific to a noise model, and that noise model — even if it exists for a given setting — may be difficult to pin down [27]. Caragiannis et al. [12,13] have addressed this problem by relaxing the MLE constraint: they only ask that the probability of the voting

☆ A preliminary version of the paper appears in the Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI), 2015.

* Corresponding author.
E-mail addresses: arielpro@cs.cmu.edu (A.D. Procaccia), nkshah@cs.cmu.edu (N. Shah), yairzick@cmu.edu (Y. Zick).

http://dx.doi.org/10.1016/j.artint.2015.10.003
0004-3702/

rule returning the ground truth go to one as the number of votes goes to infinity. This allows them to design voting rules that uncover the ground truth in a wide range of noise models; however, they may potentially require an infinite amount of information.

Our approach. In this paper, we propose a fundamentally different approach to aggregating noisy votes. Instead of assuming probabilistic noise, we assume a known upper bound on the "total noise" in the input votes, and allow the input votes to be *adversarial* subject to the upper bound. We emphasize that in potential application domains there is no adversary that actively inserts errors into the votes; we choose an adversarial error model to be able to correct errors even in the *worst case*. This style of worst-case analysis — where the worst case is assumed to be generated by an adversary — is prevalent in many branches of computer science, e.g., in the analysis of online algorithms [10], and in machine learning [23,9].

We wish to design voting rules that do well in this worst-case scenario. From this viewpoint, our approach is closely related to the extensive literature on *error-correcting codes*. One can think of the votes as a *repetition code*: each vote is a transmitted noisy version of a "message" (the ground truth). The task of the "decoder" is to correct adversarial noise and recover the ground truth, given an upper bound on the total error. The question is: how much total error can this "code" allow while still being able to recover the ground truth?

In more detail, let d be a distance metric on the space of rankings. As an example, the well-known *Kendall tau (KT)* distance between two rankings measures the number of pairs of alternatives on which the two rankings disagree. Suppose that we receive n votes over the set of alternatives $\{a, b, c, d\}$, for an even n, and we know that the average KT distance between the votes and the ground truth is at most $1/2$. Can we always recover the ground truth? No: in the worst-case, exactly $n/2$ agents swap the two highest-ranked alternatives and the rest report the ground truth. In this case, we observe two distinct rankings (each $n/2$ times) that only disagree on the order of the top two alternatives. Both rankings have an average distance of $1/2$ from the input votes, making it impossible to determine which of them is the ground truth.

Let us, therefore, cast a larger net. Inspired by *list decoding* of error-correcting codes (see, e.g., [20]), our main research question is:

> *Fix a distance metric d. Suppose that we are given n noisy rankings, and that the average distance between these rankings and the ground truth is at most t. We wish to recover a ranking that is guaranteed to be at distance at most k from the ground truth. How small can k be, as a function of n and t?*

Our results. We observe that for any metric d, one can always recover a ranking that is at distance at most $2t$ from the ground truth, i.e., $k \le 2t$. We also show that one can pick, in polynomial time, a ranking from the given noisy rankings that provides a weaker $3t$ upper bound. We complement the upper bounds by providing a lower bound of (roughly) $k \ge t/2$ that holds for every distance metric. We also show that an extremely mild assumption on the distance metric improves the lower bound to (roughly) $k \ge t$. In addition, we consider the four most popular distance metrics used in the social choice literature, and prove a tight lower bound of (roughly) $k \ge 2t$ for each metric. This lower bound is our main theoretical result; the construction makes unexpected use of Fermat's Polygonal Number Theorem.

The worst-case optimal voting rule in our framework is defined with respect to a *known* upper bound t on the average distance between the given rankings and the ground truth. However, we show that the voting rule which returns the ranking minimizing the total distance from the given rankings — which has strong theoretical support in the literature — serves as an approximation to our worst-case optimal rule, irrespective of the value of t. We leverage this observation to provide theoretical performance guarantees for our rule in cases where the error bound t given to the rule is an underestimate or overestimate of the tightest upper bound.

Finally, we test our worst-case optimal voting rules against many well-known voting rules, on two real-world datasets [27], and show that the worst-case optimal rules exhibit superior performance as long as the given error bound t is a reasonable overestimate of the tightest upper bound.

Related work. Our work is related to the extensive literature on error-correcting codes that use permutations (see, e.g., [5], and the references therein), but differs in one crucial aspect. In designing error-correcting codes, the focus is on two choices: i) the *codewords*, a subset of rankings which represent the "possible ground truths", and ii) the *code*, which converts every codeword into the message to be sent. These choices are optimized to achieve the best tradeoff between the number of errors corrected and the rate of the code (efficiency), while allowing unique identification of the ground truth. In contrast, our setting has fixed choices: i) every ranking is a possible ground truth, and ii) in coding theory terms, our setting constrains us to the repetition code. Both restrictions (inevitable in our setting) lead to significant inefficiencies, as well as the impossibility of unique identification of the ground truth (as illustrated in the introduction). Our research question is reminiscent of coding theory settings where a bound on adversarial noise is given, and a code is chosen with the bound on the noise as an input to maximize efficiency (see, e.g., [21]).

List decoding (see, e.g., [20]) relaxes classic error correction by guaranteeing that the number of possible messages does not exceed a small quota; then, the decoder simply lists all possible messages. The motivation is that one can simply scan the list and find the correct message, as all other messages on the list are likely to be gibberish. In the voting context, one cannot simply disregard some potential ground truths as nonsensical; we therefore select a ranking that is close to every possible ground truth.