Decision-making with Bayesian experts: Can we surpass the best expert?

COS 521 Advanced Algorithm Design

Runzhe Yang[1], Jimmy Wu[2]
Department of Computer Science, Princeton University
[1]runzhey@cs.princeton.edu, [2]jw60@cs.princeton.edu

January 16, 2019

Abstract

This research project studies a novel online decision-making problem of choosing the correct action each round given access to experts that are knowledgeable, rational, and truthful, rather than arbitrary. We ask, in this setting, whether we can surpass the best expert using voting-based algorithms when the experts provide additional information along with their votes. We first show that we can reduce the number of mistakes using the confidence reported by each expert, and rival the performance of the best expert using a weight multiplication algorithm. However, the confidence reported by each expert is insufficient for recovering the correct action, so in order to surpass the best expert, we need more informative cues from these experts to recover the correct action. The experts cannot individually make use of these cues. We show that we can surpass the best expert by having experts provide their predictions of the vote distribution of other experts. We show that with high probability, the correct answer is the one that is "surprisingly popular" relative to their predictions. We then propose an online learning algorithm incorporating this "surprisingly popular" voting strategy, and show that it significantly outperforms the best expert in simulations with real stock data.

1 Motivation

In lecture, we discussed an algorithm for decision-making under total uncertainty. In this setting, the decision maker is allowed to ask votes from n experts to make his/her own decision, while the prior distribution on the set of possible outcomes is unavailable for the decision maker. Without making any assumptions about those experts, we can surprisingly achieve the following bound:

$$\mathbb{E}[\texttt{ERR}^{(T)}] \leq (1+\eta)\texttt{err}^{(T)} + \frac{2\ln n}{\eta}, \tag{1}$$

where $\eta \leq 1/2$ is a fixed learning rate for our multiplicative weights algorithm, $\mathbb{E}[\texttt{ERR}^{(T)}]$ is the he number of mistakes we have made after T rounds, and $\texttt{err}^{(T)}$ is the number of mistakes made by the best among n total experts.

This encouraging result indicates that we can make decisions quite comparable to what the best expert would do, even when we know nothing about the environment and the experts. However, in a more realistic setting, the environment is not entirely uncertain to those experts (i.e., they have some knowledge about the environment), and the experts can be expected to be rational and truthful (i.e., they will truthfully report their prediction based on observed evidence). Can we devise a better decision-making algorithm in this case? If the experts can provide more information to support or explain their votes, e.g., their confidence or their prediction of how other experts will vote, would that information be helpful to improve our decision-making? In this research project, we will address the above questions with algorithmic analysis and experimental validation.

2 Problem Statement

2.1 Bayesian Expert Models

We study decision-making with n experts in a more realistic setting, where the experts have ***knowledge*** about the environment, and this expert is ***rational*** and ***truthful***. To briefly illustrate our assumptions and general idea, we start with the single stock trading example. The daily price movement of the stock is modeled as a sequence of binary events: $\{\texttt{up}, \texttt{down}\}$. Each morning we try to predict whether the price will go up or down on that trading day.

Now we seek help from n experts, who have some knowledge about the stock that we do not have. The expert i will give a vote $V^i \in \{\texttt{up}, \texttt{down}\}$ based on some evidence it discovered, summarized by a private signal $S^i \in \{s_1, \ldots, s_r\}$. Here, we assume these signals are categorical, which can be interpreted as all possible feature combinations that expert can observe, such as news releases on earnings and profits, introductions of a new product, or accounting errors or scandals. The expert i will vote V^i according to a function $V^i = V(S^i)$ that maps private signal to votes for prediction `up` and `down`.

Since they are "experts", we assume they are ***knowledgeable*** about the stock, which means that the joint distribution $p(a, s_k)$ is common knowledge to the experts, where $a \in \{\texttt{up}, \texttt{down}\}$. This is a reasonable assumption since expert can acquire the joint distribution from historical data or from their own investment experience. The experts vote differently since they each observe different private signals each round. Conditional on the true price change to the stock, the experts view the signals exposed to them as being independent, identically distributed with probabilities $p(s_k|\texttt{up})$ according to the knowledge they all have. However, in our online learning setting, the experts receive private signal $s \sim \tilde{p}_i(\cdot|a)$, where $\tilde{p}_i(\cdot|a) := \texttt{normalize}(p(\cdot|a) \cdot (1 + \epsilon_i(\cdot)))$ and $\epsilon_i \sim N^r(0, \epsilon)$ is some *expert-dependent noise* with $\mathbb{E}_i[\epsilon_i] = 0$. Therefore, some experts may have more accurate signals about the environment than others do.

We consider n ***rational*** and ***truthful*** experts in our study, which assumes that they are Bayesian and that the expert i votes accordingly to the maximum of the posterior $V(S^i) = \max_{a \in \{\texttt{up}|\texttt{down}\}} p(a|S^i)$. The confidence of the expert i receiving signal s_k and voting for a is $p(a|s_k)$. The output of our online decision-making algorithm at round t is made based on their votes, $f^{(t)}(V_1^{(t)}, \ldots, V_n^{(t)})$, or other auxiliary information, $f^{(t)}(V_1^{(t)}, I_1^{(t)}, \ldots, V_n^{(t)}, I_n^{(t)})$.

The above setting can be easily extended to scenarios with multiple choices of actions. Suppose there are m possible actions $\{a_1, \ldots, a_m\}$ each round, and the payoffs of taking these actions are unknown. Let a_* be the action with the highest payoff, and our goal is to find this best action a_* based on the expert votes V^i. The experts know the joint probability model $p(s_k, a_\ell)$ about the environment, but they do not know which a_ℓ is the a_*. They will observe the private signal s_k as the evidence, and then give a vote accordingly to the maximum of the ideal posterior $p(a_\ell|s_k)$. Our final decision is made based on their votes or any other auxiliary information they can provide.

2.2 Research Questions

The original weighted majority algorithm guarantees an upper bound on the number of mistakes. Even for irrational and untruthful experts without any knowledge about the environment, we can perform quite compatible to the best expert can do. Now knowing that our experts are all "good", can we outperform this baseline? If even the best expert often votes for the wrong decision, can we do better than the best expert? We break our research goal into the following specific questions:

- The best expert (could be one with the smallest $\|\epsilon_i\|_1$) can make mistakes in our model since its votes depend on signals s_k it received. What is the expected number of mistakes that the best expert in our model will make after T rounds? Can the original weighted majority algorithm perform better than the best expert? What is the expected number of mistakes?

- If experts can also give their *confidence* $p(a_\ell|S^i = s_k)$ about their votes, can we incorporate this information with the multiplicative weight algorithm (e.g., also weighted by their confidence), to make fewer mistakes than the normal majority voting? Can it outperform the best expert? What is the expected number of mistakes after T rounds?

- If experts can also give their *predictions about the distribution of other experts' votes*, based on their private signal, $p(V^{-i}_{pred} = a_j|S^i = s_k)$, would that helpful? Can we incorporate this information with some online learning algorithm, to outperform the best expert? What is the expected number of mistakes after T steps in this case?

This online learning setup is novel and there is little similar literature to our knowledge. However, we tried our best to delve into these questions. For the first two questions, we expect the answers are negative since experts confidence is intuitively inconclusive. For the third questions, the intuition is that the "correct" decision will be more popular than their prediction in our assumed setting. We expect to outperform the best expert with this additional information successfully. We rigorously formulate these three cases, mathematically prove our conjectures, and numerically simulate the online learning procedures in this project.

3 Main Theory

We start by introducing a few notations frequently used in this paper, and clearly state the assumptions that make our discussion meaningful and technically friendly. Our main theory consists of three parts. In the first parts, we will discuss the property of these n Bayesian experts — How the noise in their observation model influences their performance? What can a single expert do? Also, what is the expected number of errors the best expert make? As well as the original randomized majority voting algorithm — How many errors in expectation this algorithm makes? Can we beat the best expert using majority voting with weight update? In the second part, we study the performance of a *confidence majority voting* algorithm, in which we allow the experts not only to vote but also report their confidence, i.e., the posterior probabilities they use to decide their votes. We will show how the provided confidence could be useful information to improve our decision compare to vanilla majority voting, and discuss whether we can beat the best expert, using confidence and learning, in terms of expected number of mistakes. In the last part, we investigate a simple method, *surprisingly popular algorithm*, in which the experts not only vote but also provide their prediction on other people's votes. This method exhibits potential ability to outperform the best expert, based on a few reasonable assumptions, and we will show the reason why. Combined with a sensible weight update scheme, we devise an online "surprisingly popular" algorithm which stably beats the best expert in terms of the total number of mistakes.

Terminology

- n, number of experts; T, total number of rounds.

- $\mathcal{A} = \{a_1, \ldots, a_m\}$ is the *action space*. Each round, we are supposed to take one actions based on actions the n experts, which are called *votes*. The vote of the i-th expert is described as a random variable V^i, which takes value in $\mathcal{A}$. But sometimes for clearance, we use redundant notations $\{v_1, \ldots, v_m\}$ to indicate the value of experts' votes of $\{a_1, \ldots, a_m\}$, correspondingly. In our setting, there is only one action a_{ℓ^*} is *correct* each round, and we denote other actions as $a_{-\ell^*}$. Our goal is to maximize the number of rounds where we take the *correct* actions.

- $\mathcal{S} = \{s_1, \dots, s_r\}$ is the *state space* or the set of *private signals*. Each round, each expert receives a private signal s_k conditioned on the correct action a_{ℓ^*} in the current round. Experts make their votes for the correct action based on their received signals each round, respectively.

- $p(a_\ell, s_k)$ is the *probabilistic model of the environment* describing the joint distribution of current correct action and rendered signal. Furthermore,
  - $p(a_\ell) = \sum_k p(a_\ell, s_k)$ denotes the *prior distribution* of the correct actions, which can be acquired by estimating from historical data in a realistic settings.
  - $p_i(s_k|a_\ell) = p(a_\ell, s_k)/p(a_\ell)$ denotes the probability that the i-th expert receives signal s_k in a world where a_ℓ is the current correct action, according to the probabilistic model of the environment. Since it is the same across all experts, we simply write this value as $p(s_k|a_\ell)$.
  - $p_i(a_\ell|s_k) = \frac{p_i(s_k|a_\ell)p(a_\ell)}{\sum_{\ell'} p_i(s_k|a'_\ell)p(a'_\ell)}$ denotes the *posterior distribution* of the event that action a_ℓ is correct given current private signal s_k for the i-the expert, based on the probabilistic model of the environment. When the current vote of the i-th expert is v_ℓ, $p_i(a_\ell|s_k)$ is also called its *confidence*. We simply note this value as $p(a_\ell|s_k)$ when i is irrelevant.
  - $p_i(v_{\ell'}|s_k)$ is the probability that the i-th expert make vote for $v_{\ell'}$ $(a_{\ell'})$ given received signal s_k. Since the expert's decision making according to private signal can be deterministic, in that case $p_i(v'_\ell|s_k)$ is either 0 or 1.
  - $p_i(v_{\ell'}|a_\ell) = \sum_k p_i(v_{\ell'}|s_k)p(s_k|a_\ell)$ represents the probability that the i-th expert vote for $v_{\ell'}$ when the current correct action is actually a_ℓ, based on model $p(a_\ell, s_k)$. Specially, we use $p_i(v_{-\ell}|a_\ell)$ to denote the probability that the i-th expert's vote is wrong, and $p_i(v_\ell|a_\ell)$ to denote the correct probability.
  - $\kappa_i = \sum_\ell p_i(v_{-\ell}|a_\ell)p(a_\ell)$ be the probability that the i-th expert makes a wrong vote.
  - $p_{-i}(v_\ell|s_k) = \sum_{\ell'} p_i(v_\ell|a_{\ell'})p(a_{\ell'}|s_k)$ denotes the *vote distribution of other experts that i-th expert predicts* based on the its own received private signal s_k.

- $\tilde{p}_i(s_k|a_\ell) = \frac{p(s_k|a_\ell)(1+\epsilon_{ik})}{\sum_{k'} p(s_{k'}|a_\ell)(1+\epsilon_{ik'})}$ is the *real signaling probability* unknown to expert, where $\epsilon_i \sim N^r(0, \epsilon)$ for tiny ϵ is a noise term, which make each experts received signal a bit different from the common model. We replace $p_i(s_k|a_\ell)$ with $\tilde{p}_i(s_k|a_\ell)$ in some of above mentioned terms to obtain $\tilde{p}_i(a_\ell|s_k)$, $\tilde{p}_i(v'_\ell|a_\ell)$, and $\tilde{\kappa}_i$ representing *real posterior distribution*, *real vote probability given action*, and *real wrong vote probability*, respectively. Note that it is not necessary to have $\tilde{p}_i(v'_\ell|s_k)$ and $\tilde{p}_{-i}(v_\ell|s_k)$ since the real signaling probability is unknown to experts for making a vote or prediction.

Assumptions

- Experts are *knowledgable* about the environment. It means the probabilistic model $p(a_\ell, s_k)$ is common knowledge to every expert. However, the correct action and the real signaling probability is unknown to any of them.

- Experts are *rational*. It means they can correctly compute the posterior distribution (confidence) and the vote distribution of other experts with Bayesian rules based on the common knowledge $p(a_\ell, s_k)$ and the private signal s_k they receive.

- Experts are *truthful*. It means that experts honestly report the exact information they have. As for the voting, in this paper we consider each expert vote for the action with maximal posterior probability when receiving s_k, i.e., $V^i = \arg\max_{a_\ell} p(a_\ell|s_k)$. Therefore, their votes are deterministic given the signal, $p_i(v_\ell|s_k)$ is either 0 or 1.

- No communication among n experts.

- Environment is *supportive*. The confusing signals are less likely to be given to experts, i.e. for any two signal $p_i(a_\ell|s_k) > p_i(a_{-\ell}|s_k)$ and $p_i(a_\ell|s'_k) < p_i(a_{-\ell}|s'_k)$, we should have $p(s_k|a_\ell) > p(s'_k|a_\ell)$. This actually assumption implies in expectation majority experts are correct.
- For simplification, we assume $m = 2$ for the most part of this project paper. The generalization from two to more should be straightforward and we leave it as future work.

3.1 Majority Voting with Bayesian Experts

We begin with an analysis of the randomized majority voting under our problem setup. The algorithm is given in Algorithm 1. We will first prove Lemmas 1 and 2.

Lemma 1. *$\kappa < 1/2$, i.e., wrong vote probability is less than half. Bayesian experts with perfect information make more correct votes than wrong votes in expectation.*

Proof. By definition, $\kappa = \sum_\ell p_i(v_{-\ell}|a_\ell)p(a_\ell)$, we have $1 - \kappa = 1 - \sum_\ell p_i(v_{-\ell}|a_\ell)p(a_\ell) = \sum_\ell p(a_\ell) - \sum_\ell p_i(v_{-\ell}|a_\ell)p(a_\ell) = \sum_\ell (1 - p_i(v_{-\ell}|a_\ell))p(a_\ell) = \sum_\ell (p_i(v_\ell|a_\ell))p(a_\ell)$, since $p_i(v_\ell|a_\ell) + p_i(v_{-\ell}|a_\ell) = 1$ hold for all $\ell \in \{1, \dots, m\}$. Thus, if we can prove $\kappa < 1 - \kappa$, which implies $\sum_\ell p_i(v_{-\ell}|a_\ell)p(a_\ell) < \sum_\ell p_i(v_\ell|a_\ell)p(a_\ell)$, we prove $k < 1/2$.

A useful observation is that based on our assumption, we have $p_i(v_{-\ell}|a_\ell) < p_i(v_\ell|a_\ell)$ for all $\ell \in \{1, \dots, m\}$, i.e., the probability of expert i voting correctly is higher than voting wrongly (sounds reasonable for an "expert"). To show this, we write

$$\begin{aligned}
p_i(v_{-\ell}|a_\ell) &= \sum_{1 \le k \le r} p_i(v_\ell|s_k) \cdot p(s_k|a_\ell) \\
&= \sum_{1 \le k \le r} \chi\{v_\ell = \arg\max_{a_{\ell'}}(p_i(a_{\ell'}|s_k))\} \cdot p(s_k|a_\ell) \\
&= \sum_{1 \le k \le r} \chi\{p(a_\ell|s_k) > p(a_{-\ell}|s_k)\} \cdot p(s_k|a_\ell) \\
(*) \quad &< \sum_{1 \le k \le r} \chi\{p(a_\ell|s_k) < p(a_{-\ell}|s_k)\} \cdot p(s_k|a_\ell) \\
&= \sum_{1 \le k \le r} p_i(v_{-\ell}|s_k) \cdot p(s_k|a_\ell) \\
&= p_i(v_\ell|a_\ell)
\end{aligned}$$

where $\chi\{\cdot\} = 1$ if the event in $\{\}$ is true, otherwise 0. The step $(*)$ above holds, because consider two signals s_k and $s_{k'}$ such that $p(a_\ell|s_k) > p(a_{-\ell}|s_k)$ and $p(a_\ell|s_{k'}) < p(a_{-\ell}|s_{k'})$, according to friendly environment assumption, which states the confusing signal is less likely to be received, we have $p(s_k|a_\ell) < p(s'_k|a_\ell)$. It completes our proof.

□

Lemma 2. *$(1 - \epsilon^2)\kappa \le \mathbb{E}_{\epsilon_i}[\tilde{\kappa}_i] \le (1 + \epsilon^2)\kappa$. The real expected vote error rate of experts is very close to its modeled error rate, only with a small distortion.*

Proof. By definition, using the fact that $\sum_{k'} p_i(s_{k'}|a_\ell) = 1$ for any ℓ:

$$\begin{aligned}
\tilde{\kappa} &= \sum_\ell \tilde{p}_i(v_{-\ell}|a_\ell)p(a_\ell) = \sum_\ell \sum_k p_i(v_{-\ell}|s_k)\tilde{p}_i(s_k|a_\ell)p(a_\ell) \\
&= \sum_\ell p(a_\ell) \sum_k p_i(v_{-\ell}|s_k) \frac{p_i(s_k|a_\ell)(1 + \epsilon_{ik})}{\sum_{k'} p_i(s'_k|a_\ell)(1 + \epsilon_{ik'})} \\
&= \sum_\ell p(a_\ell) \sum_k p_i(v_{-\ell}|s_k)p_i(s_k|a_\ell) \cdot \frac{(1 + \epsilon_{ik})}{1 + \sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'}}
\end{aligned}$$

Expanding in Taylor series we know fact that $1 - \sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'} \leq 1/(1 + \sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'}) \leq 1 - \sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'} + (\sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'})^2 \Rightarrow$

$$\mathbb{E}_{\epsilon_i}\left[\frac{1+\epsilon_{ik}}{1+\sum_{k'} p_i(s'_k|a_\ell)}\right] \geq \mathbb{E}_{\epsilon_i}\left[(1+\epsilon_{ik})\left(1 - \sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'}\right)\right] \geq 1 - \epsilon^2 \tag{2}$$

$$\mathbb{E}_{\epsilon_i}\left[\frac{1+\epsilon_{ik}}{1+\sum_{k'} p_i(s'_k|a_\ell)}\right] \leq \mathbb{E}_{\epsilon_i}\left[(1+\epsilon_{ik})\left(1 - \sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'} + \left(\sum_{k'} p_i(s_{k'}|a_\ell)\epsilon_{ik'}\right)^2\right)\right] \leq 1 + \epsilon^2 \tag{3}$$

where we derive inequalities 2 and 3 based on some properties of $\epsilon_i \sim N^r(0, \epsilon)$, including $\mathbb{E}_{\epsilon_i}[\epsilon_{ik}] = 0, \mathbb{E}_{\epsilon_i}[\epsilon_{ik}^2] = Var(\epsilon_{ik}) + \mathbb{E}^2[\epsilon_{ik}] = \epsilon^2$, $\mathbb{E}_{\epsilon_i}[\epsilon_{ik}\epsilon_{ik'}] = 0$ for $k' \neq k$. Thus we have shown that $(1-\epsilon^2)\kappa \leq \mathbb{E}_{\epsilon_i}[\tilde{\kappa}_i] \leq (1+\epsilon^2)\kappa$, which means the real expected wrongly voting probability is very close to its the wrongly voting probability in the common ideal model, only with a small $(1 \pm \epsilon^2)$ distortion. This lemma implies that in many cases we can analyze the expert's behavior just simply using noise-free model then bounded with some small distortion.

□

Note that Lemmas 1 and 2 do not imply the majority is always correct. There is still a chance for more than half of experts to make wrong votes. Next, we prove Theorem 1.

Theorem 1. *The asymptotically upper and lower bound for the expected number of mistakes that the best expert makes in T round are*

$$\kappa(1-\epsilon^2)T - \sqrt{(T/2)\ln n} \leq \mathbb{E}\left[\min_{1 \leq i \leq n} \mathtt{err}_i^{(T)}\right] \leq \kappa(1+\epsilon^2)T - \sqrt{k(1-\epsilon^2)T \ln n}$$

Proof. Each round experts' vote are i.i.d., and the κ is wrongly voting probability in the noise-free model. Therefore, $\mathtt{err}_\mathtt{i}^{(\mathtt{T})}$ is roughly a summation of T random variables drawn from binomial distribution $Bin(k, T)$. When T is large, according to the law of large number, we can use the Gaussian distribution $N(kT, \sqrt{k(1-k)T})$ to approximate the distribution of $\mathtt{err}_\mathtt{i}^{(\mathtt{T})}$. Hence, asymptotically we have

$$\mathbb{E}\left[\min_{1 \leq i \leq n} \mathtt{err}_i^{(T)}\right] \asymp \kappa T - \sqrt{\kappa(1-\kappa)T} \cdot \sqrt{2 \log n}$$

Using the Lemma 2 we derive tight asymptotical bounds for the noisy case:

$$\mathbb{E}\left[\min_{1 \leq i \leq n} \mathtt{err}_i^{(T)}\right] \geq \kappa(1-\epsilon^2)T - \sqrt{2\tilde{\kappa}(1-\tilde{\kappa})} \cdot \sqrt{T \log n} \geq \kappa(1-\epsilon^2)T - \sqrt{(T/2)\ln n},$$

and also use Lemma 1 as well we have

$$\mathbb{E}\left[\min_{1 \leq i \leq n} \mathtt{err}_i^{(T)}\right] \leq \kappa(1+\epsilon^2)T - \sqrt{2\tilde{\kappa}(1-\tilde{\kappa})} \cdot \sqrt{T \log n} \geq \kappa(1+\epsilon^2)T - \sqrt{(1+\epsilon)^2 \kappa T \ln n},$$

We gives our desired bound for the expected number of mistakes that the best expert makes in T rounds. Note that it is roughly $\kappa T - O(\sqrt{kT \log n})$. □

The majority voting algorithm makes the decision with a probability based on the experts' votes. For instance, if there 4 out of 10 experts vote for `up`, 6 out of 10 experts vote for `down` in some round, the probability that our majority voting algorithm choose `up` is 0.4, and 0.6 for `down`. The performance of this algorithm is shown in Theorem 2.

Algorithm 1 Weighted Majority Voting Algorithm with Bayesian Experts

Set a learning rate $\eta \leq 1/2$ ($\approx \sqrt{(\ln n)/T}$).

For each expert i, associate the initial weight $w_i^{(0)} = 1$.

for round $t = 0, \ldots, T-1$ **do**

Experts receive private signals $\{s^i\}_{1 \leq i \leq n}$, and make votes $\{v^i\}_{1 \leq i \leq n}$.

Make a decision $D^{(t)}$ with probability based on is the weighted majority of the experts votes. The weights associated to experts are

$$w_1^{(t)}/\Phi^{(t)}, \ldots, w_n^{(t)}/\Phi^{(t)},$$

where $\Phi^{(t)} = \sum_i w_i^{(t)}$.

For every expert i who vote wrongly, decrease its weight for the next round by multiplying it by a factor of $(1-\eta)$:

$$w_i^{(t+1)} = (1-\eta) w_i^{(t)}$$

end for

Theorem 2. *The expected number of mistakes the majority voting algorithm makes is bounded by*

$$\kappa(1-\epsilon^2)T \leq \mathbb{E}[\mathtt{ERR}_{mv}^{(T)}] \leq \kappa(1+\epsilon^2)T$$

Proof. The random variable $\mathtt{ERR}_{mv}^{(T)}$ can be seen as the sum of T random variables i.i.d. with probability $\sum_i \mathbb{E}[\tilde{\kappa}_i]/n$ to be 1 (indicating our decision is wrong) and $1 - \sum_i \mathbb{E}[\tilde{\kappa}_i]/n$ to be zero. Use Lemma 2

$$k(1-\epsilon^2)T \frac{n(1-\epsilon^2)\kappa}{n} T \leq \mathbb{E}[\mathtt{ERR}_{mv}^{(T)}] = \frac{\sum_i \mathbb{E}[\tilde{\kappa}_i]}{n} \cdot T \leq \frac{n(1+\epsilon^2)\kappa}{n} T = k(1+\epsilon^2)T,$$

which says the performance of majority voting (without learning) is similar to the performance of a single expert. □

From the lecture we know we can improve the majority voting by dynamically adjusting weights assigns to each expert when averaging their votes. Our decision is made with a probability based on the weighted average of their votes. This classic online algorithm is demonstrated in Algorithm 1. We analyze its performance with Bayesian experts in Theorem 3.

Theorem 3. *The expected number of mistakes the weighted majority voting algorithm (online learning) makes is bounded by*

$$(1+\eta)(\kappa(1-\epsilon^2)T - \sqrt{(T/2)\ln n} \leq \mathbb{E}[\mathtt{ERR}_{wmv}^{(T)}] \leq (1+\eta)(\kappa(1+\epsilon^2)T - \sqrt{k(1-\epsilon^2)T \ln n}) + \frac{\ln n}{\eta},$$

where $\eta \leq 1/2$ is the learning rate.

Proof. We can write $\mathbb{E}[\mathtt{ERR}_{wmv}^{(T)}] = \sum_{t=0}^{T-1} \sum_{i=1}^{n} r_i^{(t)} \tilde{w}_i^{(t)}$, where $r_i^{(t)}$ indicating the mistake the i-th expert makes in round t, $r_i^{(t)} = 1$ with probability $\tilde{\kappa}_i$ and otherwise $r_i^{(t)} = 0$ with probability $1 - \tilde{\kappa}_i$ for all $t \in [T]$. In a compact vectorized form, we write $\mathbb{E}[\mathtt{ERR}_{wmv}^{(T)}] = \sum_{t=0}^{T-1} \boldsymbol{r}^{(t)} \cdot \tilde{\boldsymbol{w}}^{(t)}$.

Similar to the proof we saw in class, we investigate the partition function how Φ changes as algorithm runs.

$$\begin{aligned}
\Phi^{(T)} &= \Phi^{(T-1)}(1 - \eta \cdot \boldsymbol{r}^{(T-1)} \cdot \tilde{\boldsymbol{w}}^{(T-1)}) \\
&\leq \Phi^{(T-1)} \exp\left\{-\eta \cdot \boldsymbol{r}^{(T-1)} \cdot \tilde{\boldsymbol{w}}^{(T-1)}\right\} \\
&\leq \prod_{t=0}^{T-1} \Phi^{(0)} \exp\left\{-\eta \cdot \boldsymbol{r}^{(t)} \cdot \tilde{\boldsymbol{w}}^{(t)}\right\} \\
&\leq n \cdot \exp\left\{-\eta \cdot \sum_{t=0}^{T-1} \boldsymbol{r}^{(t)} \cdot \tilde{\boldsymbol{w}}^{(t)}\right\}
\end{aligned}$$

Since the final partition function $\Phi^{(T)}$ is at least the final weight of arbitrary expert i, then we have

$$(1-\eta)^{\sum_{t=0}^{T-1} r_i^{(t)}} \leq \Phi^{(T)} \leq n \cdot \exp\left\{-\eta \cdot \sum_{t=0}^{T-1} \boldsymbol{r}^{(t)} \cdot \tilde{\boldsymbol{w}}^{(t)}\right\}$$

By taking ln on both side we get $\mathtt{err}_i^{(T)} \log(1-\eta) \leq \mathtt{err}_i^{(T)} \eta(1+\eta) \leq \log n - \eta \cdot \mathbb{E}[\mathtt{ERR}_{\mathtt{wmv}}^{(\mathtt{T})}]$, combining with Theorem 1 which implies $\mathbb{E}[\mathtt{ERR}_{wmv}^{(T)}] \leq (1+\eta)\mathtt{err}_i^{(T)} + \frac{\ln n}{\eta} \leq (1+\eta)(\kappa(1+\epsilon^2)T - \sqrt{k(1-\epsilon^2)T \ln n}) + \frac{\ln n}{\eta}$

Note that when taking $\eta \approx \sqrt{(\ln n)/\kappa T}$, we have $\mathbb{E}[\mathtt{ERR}_{wmv}^{(T)}] \leq \kappa(1+\epsilon^2)T - O(\sqrt{kT \ln n})$, which is of the same order as the best expert in Theorem 1, and therefore is better than the majority voting algorithm without learning.

As for the lower bound, since $(1-x) \leq e^{-tx}$ for $t \geq -(\ln(1-x))/x$ for $x < 1$, we have

$$\begin{aligned}
\Phi^{(T)} &= \Phi^{(T-1)}(1 - \eta \cdot \boldsymbol{r}^{(T-1)} \cdot \tilde{\boldsymbol{w}}^{(T-1)}) \\
&\geq \Phi^{(T-1)} \exp\left\{\left(-\frac{\ln(1-\eta)}{\eta}\right)\left(-\eta \cdot \boldsymbol{r}^{(T-1)} \cdot \tilde{\boldsymbol{w}}^{(T-1)}\right)\right\} \\
&\geq \prod_{t=0}^{T-1} \Phi^{(0)} \exp\left\{\ln(1-\eta) \cdot \boldsymbol{r}^{(t)} \cdot \tilde{\boldsymbol{w}}^{(t)}\right\} \\
&\geq n \cdot \exp\left\{\ln(1-\eta) \cdot \sum_{t=0}^{T-1} \boldsymbol{r}^{(t)} \cdot \tilde{\boldsymbol{w}}^{(t)}\right\}
\end{aligned}$$

Because the final partition function $\Phi^{(T)}$ is at most n times the final weight of the best expert i (who masks fewest mistakes), then we have

$$n \cdot (1-\eta)^{\mathtt{err}_{best}^{(T)}} \geq \Phi^{(T)} \geq n \cdot \exp\left\{\ln(1-\eta) \cdot \sum_{t=0}^{T-1} \boldsymbol{r}^{(t)} \cdot \tilde{\boldsymbol{w}}^{(t)}\right\}$$

Thus by taking ln on both side, we have $\mathbb{E}[\mathtt{ERR}_{wmv}^{(T)}] \geq \mathtt{err}_{best}^{(T)} \geq (1+\eta)(\kappa(1-\epsilon^2)T - \sqrt{(T/2)\ln n}$. This indicates that we cannot use weighted majority voting algorithm to surpass the best expert!

□

Remark 1. *The weighted majority voting algorithm is better than the one without learning, but it can not surpass the best expert. (This statement has been shown in the proof of Theorem 3).*

Algorithm 2 Confident Majority Voting Algorithm with Bayesian Experts

Set a learning rate $\eta \leq 1/2$ ($\approx \sqrt{(\ln n)/T}$).

For each expert i, associate the initial weight $w_i^{(0)} = 1$.

for round $t = 0, \ldots, T-1$ **do**

Experts receive private signals $\{s^i\}_{1\leq i\leq n}$, make votes $\{v^i\}_{1\leq i\leq n}$, and report their confidence of voted actions $\{c_i := p_i(a^i|s^i)\}_{1\leq i\leq n}$.

Make a decision $D^{(t)}$ with probability based on is the confidence weighted majority of the experts votes. The weights associated to experts are

$$c_1 w_1^{(t)}/\Phi^{(t)}, \ldots, c_n w_n^{(t)}/\Phi^{(t)},$$

where $\Phi^{(t)} = \sum_i c_i w_i^{(t)}$.

For every expert i who vote wrongly, decrease its weight for the next round by multiplying it by a factor of $(1-\eta)$:

$$w_i^{(t+1)} = (1-\eta) w_i^{(t)}$$

end for

3.2 Rival the Best Expert with Confidence

In section 3.1, we obtain a negative result that even though online learning can improve the randomized majority voting, it cannot beat the best expert. In this and the next section, we try to let experts to provide additional information, to see whether we can use this extra information to surpass the best expert.

What is possible additional information that we can utilize? Confidence that experts feel about their current votes could be one option. Each round, we allow the experts to report their confidence, i.e., the posterior probability of the correct action after observing the signal, along with their votes. Our final decision is based on the average of their votes weighted by their confidence.

How good is this confident majority voting strategy?

Theorem 4. *The expected number of mistakes the majority voting with confidence algorithm makes is no greater than* $\kappa(1+\epsilon^2)T - O(\sqrt{\kappa T \ln n})$

Combined with online learning, we can modify the randomized majority voting by multiplying reported confidence as part of weights each round. The complete algorithm of this confident majority voting algorithm is illustrated in Algorithm 2. We have a similar bound for this algorithm

Theorem 5. *The expected number of mistakes the weighted majority voting with confidence algorithm (online learning) makes is no greater than* $\kappa(1+\epsilon^2)T - O(\sqrt{\kappa T \ln n})$

We only show some intuition to prove Theorem 4 and 5 here. The confident majority voting can be seen as randomized majority voting with some "good" initial weights, Therefore it should be better than vanilla majority voting without learning. However, since partition function changes following a similar claim in the proof of Theorem 3, confident majority voting can only change the additive error $\ln n/\eta$ with a constant. Therefore the bound is still $\kappa(1+\epsilon^2)T - O(\sqrt{\kappa T \ln n})$.

The confidence looks not very helpful, but why? Lemma 3 gives a potential reason that the confidence actually is not informative. Knowing confidence cannot help us know more about the correct action.

Lemma 3. *The confidence is not informative. The same confidence can be generated in some probabilistic model for an arbitrarily selected action.*

Algorithm 3 Weighted Surprisingly Popular Voting Algorithm with Bayesian Experts

Set a small learning rate η.
For each expert i, associate the initial weight $w_i^{(0)} = 1$.
for round $t = 0, \dots, T-1$ **do**
Experts receive private signals $\{s^i\}_{1\le i\le n}$, make votes $\{v^i\}_{1\le i\le n}$, and report their prediction of the distribution of other experts' vote $\{d_i(\cdot) := p_{-i}(\cdot|s^i)\}_{1\le i\le n}$.
Make a decision $D^{(t)}$ based on difference between the real experts votes and the weighed predicted vote distribution:

$$\bar{d}_i^{(t)}(a_\ell) = \sum_i d_i(a_\ell) w_i^{(t)} / \Phi^{(t)},$$

where $\Phi^{(t)} = \sum_i w_i^{(t)}$. If v_ℓ's frequency in $\{v^i\}_{1\le i\le n}$ is greater than $\bar{d}^{(t)}(a_\ell)$, then $D^{(t)} = v_\ell$.
For every expert i who predicts $d(a_{\ell^*}) < freq(v_{\ell^*})$, increase its weight for the next round by adding it with the learning constant η:

$$w_i^{(t+1)} = w_i^{(t)} + \eta$$

end for

Proof. Even when knowing $p(s_k|a_\ell)$ and $p(a_\ell|s_k)$ for all $k \in \{1, \dots, r\}$, we can construct a different joint distribution,

$$q(s_k, a_\ell) = p(s_k|a_{\ell^*}) \left(\frac{p(s_{k'}|a_{\ell^*})}{p(a_\ell|s_{k'})} \right)^{-1}$$

which has $q(s_k|a_\ell) = p(s_k|a_\ell)$ and $q(a_\ell|s_k) = p(a_\ell|s_k)$. Since a_ℓ can be arbitrarily action, the same confidence can be generated in some probabilistic model for an arbitrarily selected action. □

Remark 2. *The majority voting algorithm with confidence is better than vanilla majority voting, and rival to the best expert in terms of the number of errors.*

3.3 Surpass the Best Expert with Predicted Vote Distribution

Now, we will analyze the "surprisingly popular" voting algorithm, in which experts provide their predictions of the vote distributions for other experts. The algorithm is shown in Algorithm 3. We begin by proving Lemmas 4 and 5 below.

Lemma 4. $p_i(v_\ell|a_\ell) > p_i(v_\ell|a_{-\ell})$. *The probability that the expert vote for the correct action is larger than the probability they wrongly vote for the same action in a counterfactual world.*

Proof. We show that actual votes for the correct action exceed counterfactual votes for the correct action,

$$\frac{p_i(v_\ell|a_\ell)}{p_i(v_\ell|a_{-\ell})} = \frac{p_i(a_\ell|v_\ell)p(a_{-\ell})}{p_i(a_{-\ell}|v_\ell)p(a_\ell)} = \frac{p_i(a_\ell|v_\ell)}{1 - p_i(a_\ell|v_\ell)} \cdot \frac{1 - p(a_\ell)}{p(a_\ell)},$$

where $p_i(a_\ell|v_\ell)$ is the probability that when the i-th expert votes for a_{ell} and the correct action is indeed a_ℓ. Note that $p_i(a_\ell|v_\ell) > p_i(a_\ell|v_\ell) \cdot p_i(v_\ell) + p_i(a_\ell|v_{-\ell}) \cdot p_i(v_{-\ell}) = p(a_\ell)$ since $p_i(a_\ell|v_\ell) > p_i(a_\ell|v_{-\ell})$, which can be justified by

$$p_i(a_\ell|v_\ell) = \sum_k p(a_\ell|s_k) p_i(s_k|v_\ell) > \frac{1}{2} > \sum_k p(a_\ell|s_k) p_i(s_k|v_{-\ell}) p_i(a_\ell|v_{-\ell}),$$

as the i-th expert deterministically votes for some a such that $p_i(a|s) > \frac{1}{2}$ (It is the one with max posterior probability, and there are two possible actions as we assume. If there are $m > 2$ actions, we simply modify $1/2$ to $1/m$. The statement still holds.) Hence, we complete the proof for $p_i(v_\ell|a_\ell) >$

$p_i(v_\ell|a_{-\ell})$, the correct votes should be more popular in the real world than some counterfactual world. $\square$

Lemma 5 (The correctness of SP voting)**.** *$p(v_\ell|a_\ell) \geq p_{-i}(v_\ell|s_k)$. The probability of correct voting is underestimated each expert no matter which s_k it receives.*

Proof. The expert receiving signal s_k estimate expected votes distribution of other expert by marginalizing across the two possible worlds where a_ℓ is and is not the correct action

$$p_{-i}(v_\ell|s_k) = p_i(v_\ell|a_\ell)p_i(a_\ell|s_k) + p_i(v_\ell|a_{-\ell})p_i(a_\ell|s_k)$$

According to Lemma 4, the probability of actual vote for the correct action is larger than that in a counterfactual world, $p_i(v_\ell|a_\ell) > p_i(v_\ell|a_{-\ell})$, we have $p_i(v_\ell|a_\ell) \geq p_{-i}(v_\ell|s_k)$, with strict inequality unless $p_i(a_\ell|s_\ell) = 1$. Because weak inequality holds for all signals, and is strict for some, the average predicted vote will be strictly underestimated. $\square$

We now bound the expected number of mistakes made by the algorithm in Theorem 6.

Theorem 6. *The expected number of mistakes the "surprisingly popular" voting algorithm makes is bounded by*

$$\mathbb{E}[\mathtt{ERR}_{sp}^{(T)}] \leq (1+\epsilon^2)^2 \cdot \kappa^2 \cdot T,$$

when there are enough experts.

Proof. Because of Lemma 2, we can just think the total number of wrong votes experts makes each round as a random variable U drawn from binomial distribution $Bin(\kappa, n)$. Therefore, in expectation, there are κn wrong votes each round. When using the "surprisingly popular" voting algorithm, let $\rho = \sum_\ell p_{-i}(v_{-\ell}|a_\ell)p(a_\ell)$. By Lemma 5 we know $\kappa < \rho$ if $U < n \cdot p_{-i}(v_{-\ell}|a_\ell)$ then we will obtain the correct answer. The probability that $U > p_{-i}(v_{-\ell}|a_\ell)$ is very small according to the Chernoff bound

$$\mathtt{Pr}\left[U > \frac{\rho}{\kappa}\kappa n\right] \leq \exp\left\{-\frac{\left(\frac{\rho}{\kappa}-1\right)^2 \kappa n}{3}\right\}$$

Hence, the expected number of mistakes the "surprisingly popular" voting algorithm makes is

$$\mathbb{E}[\mathtt{ERR}_{sp}^{(T)}] = T \cdot \mathtt{Pr}\left[U > \frac{\rho}{\kappa}\kappa n\right] \leq T \cdot \exp\left\{-\frac{\left(\frac{\rho}{\kappa}-1\right)^2 (1-\epsilon^2) \cdot \kappa n}{3}\right\}$$

Now we bound the ratio of ρ to κ.

$$\begin{aligned}
\rho &= \sum_\ell p_{-i}(v_{-\ell}|a_\ell)p(a_\ell) \\
&= \sum_\ell p(a_\ell) \sum_k p_{-i}(v_{-\ell}|s_k)p_i(s_k|a_\ell) \\
&= \sum_\ell p(a_\ell) \sum_k p_i(s_k|a_\ell) \sum_{\ell'} p_i(v_{-\ell}|a_{\ell'})p_i(a_{\ell'}|s_k) \\
&= \sum_\ell p(a_\ell) \sum_k p_i(s_k|a_\ell) \sum_{\ell'} \left[p_i(v_{-\ell}|a_{\ell'})p(a_{\ell'})\right] p_i(a_{\ell'}|s_k)/p(a_{\ell'})
\end{aligned}$$

Therefore, ρ should be some interpolation between κ and $1-\kappa$. Specially, when

$$\rho \geq \kappa + \sqrt{(6\kappa \ln 1/\kappa)/n},$$

holds, we will have

$$\mathbb{E}[\mathtt{ERR}_{sp}^{(T)}] \leq T \cdot \exp\left\{-\frac{6\ln 1/\tilde{\kappa}}{3}\right\} \leq (1+\epsilon^2)^2 \cdot \kappa^2 T.$$

In this case we have $\mathbb{E}[\mathtt{ERR}_{sp}^{(T)}] << \kappa(1+\epsilon^2)T - O(\sqrt{\kappa T \ln n})$ for large T, since the increasing rate of mistakes for "surprisingly popular" algorithm, κ^2, is much lower than that of the best expert, κ. Therefore, the "surprisingly popular" voting algorithm can significantly surpass the best expert! Note that this condition commonly holds when we have enough number of experts. $\square$

Based on this simple "surprisingly popular" voting algorithm, we design an online learning algorithm which further improves the performance. This "surprisingly popular" voting with online learning algorithm is demonstrated in 3. The basic intuition of this learning algorithm is that, as usual, we wish to assign different weights to different experts, but now we will weight more on the experts who tend to underestimate the correct vote, so that ensure the correctness of our "surprisingly popular" voting strategy. Our simulation indicates the learning version "surprisingly popular" voting algorithm can outperform the one without learning, by slightly increasing the accuracy each round. We leave the bound for this "surprisingly popular" voting with learning algorithm as future work.

4 Experiments

We conducted experimental validation of our theoretical analysis on the real-world S&P stock data from Homework 4. We implemented all of the algorithms discussed in c, ran them on the stock data, and compared their performance with that of the best expert. We discuss below our results for various setups.

4.1 Algorithms Implemented

We implemented Algorithms 1, 2, and 3 described in Section 3. We also implemented variations of all three algorithms without the learning component, in which the weights are kept the same the entire time. These six algorithms are compared with the performance of the best expert.

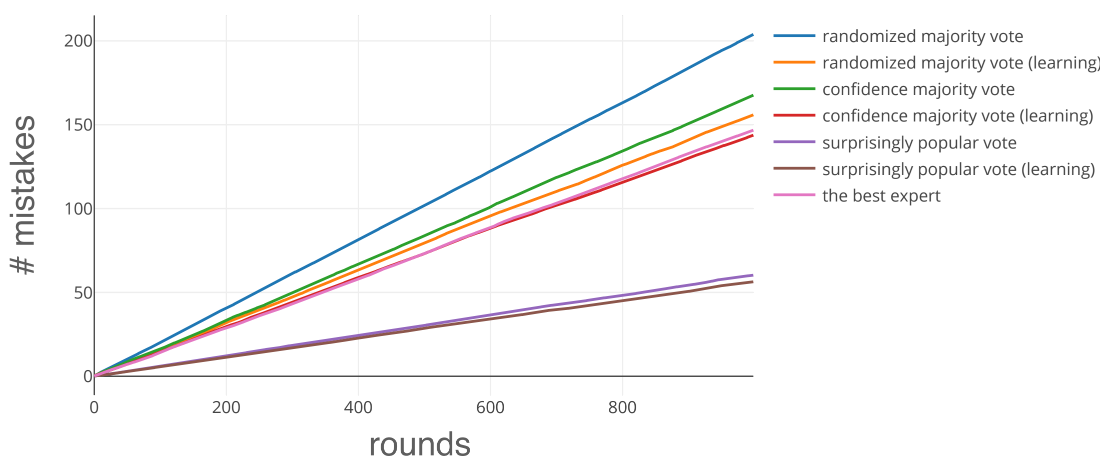

Figure 1: We show the number of mistakes made by each strategy over the span of 1000 days. Empirically, the "surprisingly popular" strategies made far fewer mistakes than the best expert. Also note that the near linearity of the plot is in agreement with our theoretical bound on the number of mistakes. The Best viewed in color.

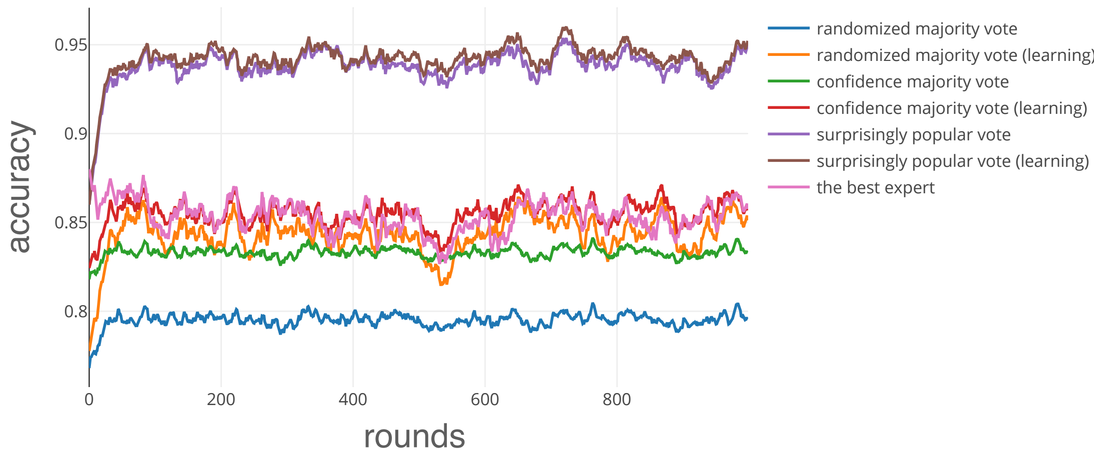

Figure 2: We show the accuracies achieved by each strategy over the 1000 day period. For clarity, we smooth the plots using an exponential moving average with a coefficient of 0.95. Best viewed in color.

4.2 Number of Mistakes

We run each algorithm on the stock data for 1000 days, and each day, we count the cumulative number of mistakes made by each algorithm. This is shown in Figure 1. We note that the near linearity of the plot is in agreement with the asymptotic bounds we derived in Sections 3.1, 3.2, and 3.3, for the number mistakes made by each algorithm.

4.3 Accuracy per Day

We measure the accuracy of each strategy for each day, where the accuracy for a given day and strategy is the value of the posterior for the correct action for that day. Alternatively, this is simply the probability of voting correctly for that day when using a particular strategy. The results are shown in Figure 2. Note that we used an exponential moving average to smooth the curve for more clarity. We observe that the "surprisingly popular" vote strategy outperforms the best expert as well as the other baselines by a large margin.

4.4 Returns

We consider a trading strategy in which we decide whether to buy or sell on each day based on whether we predict the stock will go up or down. When we buy, we convert all of our cash to stock, and when we sell, we convert all of our stock to cash. We assume no fees for our transactions. We apply this trading strategy for all six of the algorithms we test, as well as for the best expert, and show the results in Figure 3. We compute the return of a strategy as the ratio between the final wealth and the initial wealth. The algorithms for "surprisingly popular" vote outperform the other algorithms as well as the best expert by a large margin.

5 Conclusion

In this project, we modified the weighted majority voting algorithm from lecture to see if, given certain assumptions about the experts and certain additional information from the experts, we can do better than the best expert. We considered the special scenario where we have knowledgeable, rational,

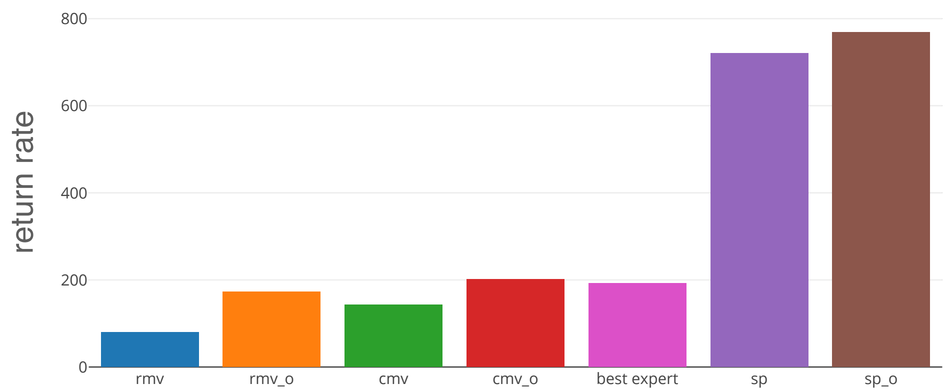

Figure 3: We show the amount of return for each strategy, expressed as the ratio between the initial wealth and the final wealth after 1000 days. The "surprisingly popular" strategies, both with and without online learning, outperform the best expert by a large margin.

and truthful experts, who provide not just their vote, but also an additional piece of information. Under this scenario, we addressed three research questions, each corresponding to a type of experts voting algorithm. We established that (1) randomized majority voting cannot surpass the best expert, (2) confidence majority voting rivals the best expert, and (3) "surprisingly popular" voting is able to outperform the best expert by a large margin. We then implement these algorithms and experimentally validate our theoretical findings on real-world S&P stock data.

6 References

[1] Kothari P. *Decision-making under total uncertainty: the multiplicative weight algorithm.* COS521 Course Lecture (2018)
[2] Arora, S. Hazan, E. & Kale, S. *The multiplicative weights update method: A meta algorithm and its applications.* Theory of Computing, Volume 8, 121164, (2012)
[3] Borodin, A. El Yaniv R.*Online Computation and Competitive Analysis.* Cambridge University Press, 1998.
[4] Prelec, D. Seung, H. S. & McCoy, J. *A solution to the single-question crowd wisdom problem.* Nature 541, 532-535 (2017)
[5] Galton, F. *Vox populi.* Nature 75, 450451 (1907)
[6] Sunstein, C. *Infotopia: How Many Minds Produce Knowledge.* Oxford University Press, 2006
[7] Surowiecki, J. *The Wisdom of Crowds.* Anchor, 2005
[8] Morgan, M. G. *Use (and abuse) of expert elicitation in support of decision-making for public policy.* Proc. Natl Acad. Sci. USA 111, 71767184 (2014).
[9] Koriat, A. *When are two heads better than one and why?* Science 336, 360362 (2012).
[10] Prelec, D. *A Bayesian truth serum for subjective data.* Science. 306, 462466 (2004).
[11] John, L. K., Loewenstein, G. & Prelec, D. *Measuring the prevalence of questionable research practices with incentives for truth-telling.* Psychol. Sci. 23, 524532 (2012).
[12] Arrow, K. J. et al. Economics. *The promise of prediction markets.* Science 320, 877878 (2008).
[13] Lebreton, M., Abitbol, R., Daunizeau, J. & Pessiglione, M. *Automatic integration of confidence in the brain valuation signal.* Nat. Neurosci. 18, 11591167 (2015).
[14] Lee, M. D., Steyvers, M., de Young, M. & Miller, B. *Inferring expertise in knowledge and prediction ranking tasks.* Top. Cogn. Sci. 4, 151163 (2012).