Extracting the collective wisdom of experts in probabilistic judgments

Cem Peker*

July 2021

Abstract

How should we combine disagreeing expert judgments on the likelihood of an event? A simple average of predictions exhibits a bias when experts have shared information. Optimal weights for weighted averaging are typically unknown and require large amount of past data to estimate reliably. This paper proposes an algorithm to aggregate probabilistic expert judgments effectively in a single prediction problem. Experts are asked to report a probabilistic prediction and a *meta-prediction*. The latter is an estimate on the average of other individuals' predictions. In a Bayesian setup I show that if average prediction is unbiased, the percentage of predictions and meta-predictions that overshoot the average prediction should be the same. An *overshoot surprise* occurs when the two measures differ, which is an indicator of a bias in the average prediction. The Surprising Overshoot (SO) algorithm uses the information revealed in an overshoot surprise to produce a more accurate estimator for the unknown probability. No past data or any other external information is required. Evidence from three experimental studies suggest that the SO algorithm consistently outperforms basic aggregates such as unweighted average and median prediction. Furthermore, the SO algorithm compares favorably to alternative aggregation mechanisms when there is a stronger disagreement between experts.

*peker@ese.eur.nl

1 Introduction

In episode 'Errand of Mercy' of *Star Trek: The Original Series*, Captain Kirk and his first officer Mr. Spock find themselves in a dangerous situation. Captain Kirk plans to infiltrate an enemy camp and asks Mr. Spock the odds of getting out alive. Although being humanoid in appearance, Mr. Spock is an alien and his race is known for enhanced abilities in logical and quantitative reasoning. He estimates the odds to be '7824.7 to 1', which Captain Kirk half-jokingly calls 'a pretty close approximation'. In his usual seriousness, Mr. Spock responds by saying 'I endeavor to be accurate'.

The days of starships and deep space exploration might still be far away, but researchers and decision makers today find themselves facing uncertain decision problems, some of which may have stakes as high as Captain Kirk's. Scientists make probabilistic projections on natural phenomena, such as occurrence of a major earthquake or the effects of anthropogenic climate change. Strategists assess the likelihood of important geopolitical events. Investors form judgments on investment opportunities. Economists are interested in predicting the chances of downturns and crashes.

In many such problems, the decision maker needs a well-calibrated probabilistic forecast. Unfortunately, a superior mind such as Mr. Spock's is not available (yet). Individual judgments might be subject to biases such as optimism, overconfidence, being anchored on an initial estimate, focusing too much on easily available information, neglecting an event's base rate and many more (9, 22, 8). Combining multiple judgments to leverage 'the wisdom of crowds' is known to be an effective approach in improving accuracy (21, 11). To illustrate, consider a decision maker who aims to predict the chances of an armed conflict between two hostile neighboring countries in the next few months. She elicits estimates from a panel of 10 experts. Suppose the actual (latent) probability is 60%. Figure 1 below depicts individual expert predictions:

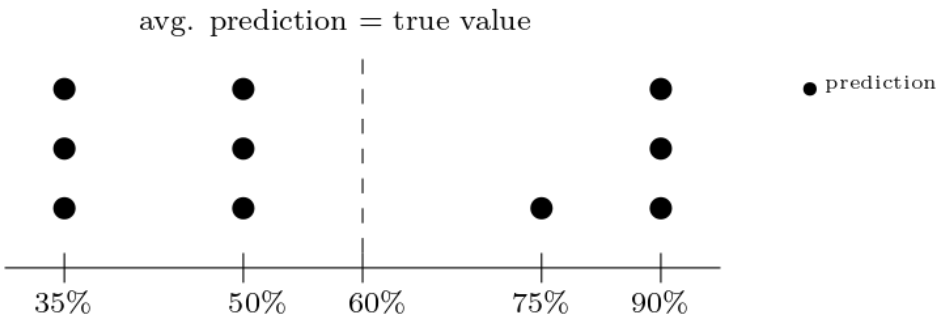


Figure 1: Example expert predictions on the chances of armed conflict

In Figure 1, $35\%, 50\%$ and 90% are reported by three experts each while one experts predicts 75%. Note that none of the experts is perfectly accurate in her prediction. However, the average of all predictions gives 60%. Figure 1 presents an example of the 'wisdom of crowds' effect (21). Individual experts either over or underestimate the likelihood of the event. However, when judgments are aggregated, individual errors cancel out and the average prediction provides a perfectly calibrated estimate.

The use of collective wisdom involves choosing an aggregation method that combines individual expert predictions into a single aggregate prediction (1, 4, 16). In the example in Figure 1, simple averaging provides an accurate prediction. Previous work found simple averaging to be surprisingly effective, typically outperforming more sophisticated aggregation methods and showing robustness across various settings (11, 12, 24, 6). Even though it is difficult to beat simple averaging consistently across many tasks, there are cases where we may expect simple average to be inaccurate. To illustrate, consider the same problem as in Figure 1. Suppose the countries in question have had recurrent conflicts and historically in about 40% of instances, tensions escalate into an armed conflict. So, the experts who study this conflict conflict have some shared information suggesting that the chances of an armed conflict is around 40%. Upon observing the more case-specific evidence, individual experts might reach different conclusions. But their judgments are still affected by the shared information. Figure 2 illustrates one such case:

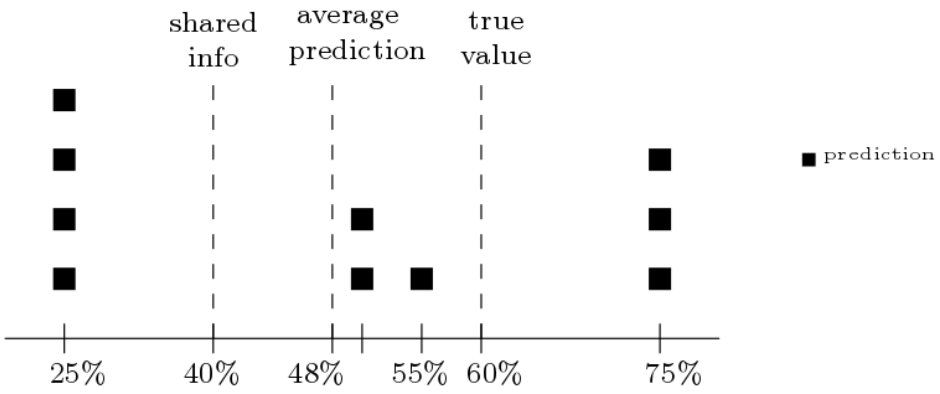


Figure 2: Predictions in the example case where experts have shared information in the form of historical data

In Figure 2, the average prediction underestimates the true value. In this example, the decision maker would be better off using a weighted average where the three 75% predictions are weighted more heavily.

Why does simple averaging fail in Figure 2? Intuitively, simple averaging allows individual forecasting errors to cancel (10), leading to a more accurate prediction. In Figure 2, three experts overshot the true probability even though the shared information underestimates the likelihood. Nevertheless, most experts ended up underestimating the likelihood. When there is shared information that deviates from the true value, experts are more likely to err in the same direction, causing a positive correlation in prediction errors and smaller error-cancellation. Since the shared information in Figure 2 underestimates the chances of the event, most individual predictions in Figure 2 undershoot the true probability. As a result, the simple average of predictions exhibits a bias (3, 18).

In general, the optimal aggregation method in a particular estimation task depends on the information structure and differences in expertise among forecasters (5, 15). For example, in Figure 2, the decision maker could improve the aggregate estimate by weighting the overshooting predictions more heavily compared to the simple unweighted average. In theory, the decision maker in a given task can select and weight judgments such that the errors perfectly cancel out (13, 2). However, optimal weights relate to how experts' prediction errors are correlated and are typically unknown to decision maker. Conventional approach aims to

estimate appropriate weights using past data from similar tasks (2, 13). The effectiveness of this approach is limited by the availability and reliability of past data.

This paper develops an algorithm to aggregate judgments on the likelihood of an event. I consider a setup where experts form their judgments by combining shared and private information on the probability of an event. When the shared information differs from the true probability, experts are likely to err on the same direction, resulting in a miscalibrated average prediction. The algorithm relies on augmented elicitation procedure commonly found in recent work (19, 20, 18, 17, 23): Experts report a prediction on the probability as well as an estimate on the average of others' predictions, which is referred to as a *meta-prediction*. I establish that when average prediction is an unbiased estimator, the percentage of predictions and meta-predictions that overshoot the average prediction should be the same. Whenever the two measures differ an *overshoot surprise* occurs, which is an indicator of a bias in the average prediction. I develop the Surprising Overshoot (SO) algorithm which produces a more accurate estimator. The SO algorithm uses the information in the size and direction of the overshoot surprise. It does not require the use of past data.

The simple example below illustrates the intuition behind an overshoot surprise. Let the cases in Figures 1 and 2 be the 'unbiased' and 'biased' worlds respectively. In the unbiased world the average prediction is accurate while in the biased world the average prediction underestimates the chances of occurrence. Figure 3 below shows the two worlds, which describe two possible scenarios. The empty circles and squares in each figure depict example meta-predictions. An expert's reports (prediction and meta-prediction) are paired in rectangles.

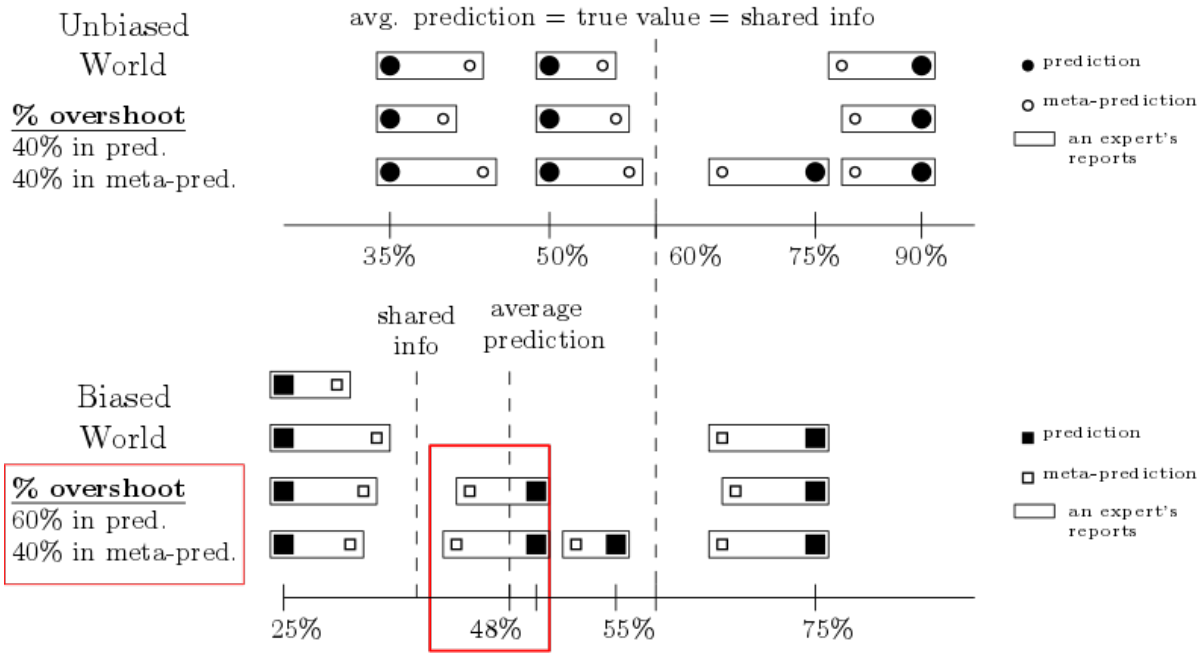


Figure 3: Example cases of unbiased and biased average prediction

In the unbiased world, 4 out of 10 predictions and meta-predictions overshoot the average prediction. Then, we can say the overshoot rate in both reports is 40%. In the biased world, 4 out of 10 predictions overshoot the average prediction, so the overshoot rate in meta-predictions is 40%. However, the overshoot rate in predictions differ. Since 6 out of 10 meta-predictions are higher than the average prediction, the overshoot rate is 60%. The difference is due to the two experts in the biased world, captured in the solid red rectangle. For these experts, the predictions undershoot the average prediction while the meta-predictions overshoot.

Why does a discrepancy between the overshoot rates in the biased world occur? Observe that in both worlds, an expert's meta-prediction is depicted in-between her prediction and the shared information. I will later show that such reporting in meta-predictions is optimal for Bayesian individuals. In the unbiased world, experts' shared information coincides with the average prediction. An expert's prediction and meta-prediction are on the same side of the average prediction. Thus, we do not expect an overshoot surprise. Indeed the overshoot rates in predictions and meta-predictions are both 40%. In contrast the shared information

and the true probability differ in the biased world, resulting in a difference between the actual probability of the event and the average prediction. Then, for some experts prediction and meta-prediction may fall on different sides of the average prediction. In Figure 3, two experts undershoot in their meta-predictions even though they overshoot in their predictions, resulting in an overshoot surprise of 20%. The SO algortihm exploits this information to produce a more accurate estimate of the true probability. We see a negative overshoot surprise when the average prediction is biased downwards (as in the biased world in Figure 3) while a positive overshoot surprise occurs when the average prediction is biased upwards. The SO algorithm infers the direction and size of the bias from the sign and magnitude of the overshoot surprise. In practice, a decision maker typically would not know if the true state of the world is biased or unbiased. The SO algorithm does not distort an unbiased average prediction while removing the bias when it is present.

I test the SO algorithm using experimental data from two sources. Palley and Soll (18) conducted an experimental study where subjects are asked to predict the number of heads in 100 flips of a biased coin. The SO algorithm shares the same Bayesian framework as Palley and Soll (18) and the experiment implements shared and private signals as sample flips from the biased coin. This study allows us to test the SO algorithm in a controlled setup. The second source is Wilkening et al. (23), who conducted two experimental studies. The first experiment replicates the earlier study by Prelec et al. (20) which asked subjects true/false questions about the capital cities of U.S. states. However, unlike Prelec et al. (20) they also ask subjects to report probabilistic predictions and meta-predictions, which allows an implementation of the SO algorithm. In the second experiment, Wilkening et al. (23) generate 500 basic science statements and ask subjects to report probabilistic predictions and meta-predictions on the likelihood of a given statement being true. I use the data from these two experiments to investigate if the SO algorithm produces an aggregate probabilistic prediction closer to the correct answer. Results suggest that the SO algorithm outperforms simple benchmarks such as unweighted averaging and median prediction. I also

compare the SO algorithm to alternative solutions for aggregating probabilistic judgments, which elicit similar information from individuals (18, 14, 17, 23). The SO algorithm never underperforms and compares favorably to alternative aggregation mechanisms in prediction tasks where individual predictions are more dispersed. Experimental evidence suggests that the SO algorithm is especially effective in extracting the collective wisdom from disagreeing probabilistic judgments in moderate to large samples of experts.

This paper contributes to the growing literature of judgment aggregation mechanisms that utilize meta-beliefs to improve prediction accuracy. Prelec et al. (20)'s Surprisingly Popular (SP) algorithm uses meta-beliefs to identify the true answer to a multiple choice question even when the majority opinion is inaccurate. In a similar setup, Wilkening et al. (23) develops the Surprisingly Confident (SC) algorithm, which uses meta-beliefs to determine weights that leverage more informed judgments. The SP and SC algorithms aim to identify the correct answer to a binary or multiple-choice question while the SO algorithm produces an estimate of the latent probability of an uncertain event.

Recent work developed aggregation algorithms for probabilistic judgments as well. As mentioned before, the SO algorithm shares the same setup as Palley and Soll (18), in which information commonly available to individuals causes the shared-information bias in the average prediction. Palley and Soll (18) develop the pivoting method which uses meta-predictions to recover and recombine shared and private information in an optimal manner. Palley and Satopää (17) consider a similar setup and propose knowledge-weighting method. Knowledge-weighting constructs a weighted crowd based on the accuracy of combined meta-prediction. The resulting weights are expected to minimize the error in combined prediction as well. The meta-probability weighting algorithm (14) considers a slightly different framework. The absolute difference between an individual's prediction and meta-prediction is considered as an indicator of her expertise. In testing the performance of the SO algorithm, pivoting, knowledge-weighting and meta-probability weighting are considered as benchmarks. As mentioned above the SO algorithm performs especially well when individual judgments

are more dispersed. In practice, such problems are likely to be the most challenging ones where expert judgments disagree substantially and it is not clear how judgments should be aggregated for maximum accuracy.

The rest of this paper is organized as follows: Section 2 introduces the formal framework. Section 3 develops the SO algorithm and illustrates how it produces an aggregate estimate. Section 4 introduces the data sets and benchmarks we consider in testing the SO algorithm empirically. The same section also presents some preliminary evidence on how overshoot surprises relate to biases. Section 5 presents experimental evidence testing the SO algorithm. Section 6 provides a discussion on the effectiveness of the SO algorithm. Section 7 concludes.

2 The Framework

The framework follows the definition of a *linear aggregation problem* in Palley and Soll (18) and Palley and Satopää (17) with a binary outcome. Let $Y \in \{0, 1\}$ be a random variable denoting the outcome of an event. Also let $\theta = P(Y = 1)$ be the latent probability of the outcome 1, representing the occurrence. A decision maker (DM) would like to estimate θ. She elicits judgments from a sample of N agents to develop an estimator, where $N \to \infty$ represents the whole population. Agents observe a *shared signal* $s \in [0, 1]$ and each agent i receives a *private signal* $t_i \in [0, 1]$. All signals follow the same expectation θ and are conditionally independent given θ.

Each agent i updates her beliefs according to the Bayes' rule. Let $E[\theta|s, t_i]$ denote an agent i's posterior expectation on θ. In a linear aggregation problem, the posterior expectation combines her signals linearly:

$$E[\theta|s, t_i] = (1 - \omega)s + \omega t_i$$

where $\omega \in [0, 1]$ represents weight an agent puts on her private signal relative to the shared signal s. The weight parameter ω is common for all agents. Palley and Soll (18) define

such a linear aggregation problem where θ is a latent probability. Agents share the common prior on θ given by a Beta density and each signal is the average of multiple independent realizations of Y. The parameter ω represents the relative informativeness of an agent's private signal t_i compared to the shared signal s.

Suppose a DM considers the simple average of agents' predictions as an estimator for θ. Let x_i be agent i's reported prediction on θ. Suppose all agents report their best guesses, i.e. $x_i = E[\theta|s, t_i]$. Then the average prediction is given by

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i = (1-\omega)s + \omega\frac{1}{N}\sum_{i=1}^{N} t_i$$

Note that $plim\, \bar{x} = (1-\omega)s + \omega\theta \neq \theta$ if $s \neq \theta$, i.e. the average prediction is not a consistent estimator of θ unless the shared information is perfectly accurate. Palley and Soll (18) refer to this as the *shared-information problem*. Section 3 develops the Surprising Overshoot algorithm, which constructs an estimator that accounts for the shared-information problem. Then, I will investigate the empirical effectiveness of the algorithm using experimental data from Palley and Soll (18) and Wilkening et al. (23).

3 The Surprising Overshoot algorithm

The decision maker asks each agent i to submit two reports. In the first, she is asked to make a *prediction* on θ, denoted by x_i. In the second, she reports a *meta-prediction* $z_i \in \mathbb{R}$, which is an estimate on the average of other agents' predictions, denoted by $\bar{x}_{-i} = \frac{1}{N-1}\sum_{j\neq i} x_j$. I assume that agents report their best estimates in each report, given by $(x_i, z_i) = (E[\theta|s, t_i], E[\bar{x}_{-i}|s, t_i])$. Such truthful elicitation can be achieved by incentivizing each report

by a proper scoring rule (7). Then, we have

$$\begin{aligned} x_i = E[x_i|s,t_i] &= (1-\omega)s + \omega t_i \\ z_i = E[\bar{x}_{-i}|s,t_i] &= (1-\omega)s + \omega E\left[\frac{1}{N-1}\sum_{j\neq i} t_j \middle| s, t_i\right] \\ &= (1-\omega)s + \omega x_i \end{aligned}$$

where the expression for z_i comes from $E[t_j|s,t_i] = (1-\omega)s + \omega t_i = x_i$ for all $j \neq i$.

A prediction or meta-prediction is said to *overshoot* the average prediction $\bar{x}$ if it exceeds $\bar{x}$. For any arbitrary agent i, there are two overshoot indicators. For example, if $x_i > \bar{x} > z_i$, agent i's prediction overshoots the average prediction while her meta-prediction does not overshoot. The SO algorithm develops crowd measures based on instances of individual overshoots. Observe that

$$\begin{aligned} x_i &> \bar{x} \\ (1-\omega)s + \omega t_i &> (1-\omega)s + \omega \frac{1}{N}\sum_{j=1}^{N} t_j \\ t_i &> \frac{1}{N}\sum_{i=1}^{N} t_j \end{aligned}$$

Recall that θ is the mean of the signal distribution. We have $\frac{1}{N}\sum_{i=1}^{N} t_i \to \theta$ for $N \to \infty$. At the limit, an agent i's prediction is higher than the average prediction if and only if agent i receives a higher than average signal, that is

$$x_i > \bar{x} \iff t_i > \theta \tag{1}$$

Now consider agent i's meta-prediction z_i. The following holds for z_i:

$$\begin{aligned} z_i &> \bar{x} \\ (1-\omega)s + \omega x_i &> (1-\omega)s + \omega \frac{1}{N}\sum_{i=j}^{N} t_j \\ x_i &> \frac{1}{N}\sum_{j=1}^{N} t_j = \theta \quad \text{for} \quad N \to \infty \end{aligned}$$

Then, we get

$$z_i > \bar{x} \iff x_i > \theta \tag{2}$$

Equations 1 and 2 suggest a pattern for $N \to \infty$. In equation 1, an agent i's prediction x_i is higher than $\bar{x}$ if and only if $t_i > \theta$. However, for her meta-prediction z_i to overshoot $\bar{x}$, we must have $x_i = (1-\omega)s + \omega t_i > \theta$. Since $s < \theta$, there could an agent i such that x_i exceeds the average prediction $\bar{x}$ but still undershoots θ. Thus, we do not necessarily have $z_i > \bar{x}_i$ whenever $x_i > \bar{x}$ is satisfied.

Now consider the following measures computed using predictions and meta-predictions:

$$\begin{aligned} p_x &= \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(x_i > \bar{x}) \\ p_z &= \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(z_i > \bar{x}) \end{aligned}$$

The measures p_x and p_z represent the population proportion of predictions and meta-predictions that overshoot the average prediction $\bar{x}$. I refer to p_x and p_z as the *overshoot rate* in predictions and meta-predictions respectively. From equation 2, we can infer that p_z also corresponds the population proportion of predictions such that $x_i > \theta$.

Let $f(x)$ denote the population density of predictions given the signal distribution. Figure 4 illustrates two scenarios with an example distribution:

(a) Case $s = \theta = 0.25$ (b) Case $\theta = 0.25 < s = 0.6$

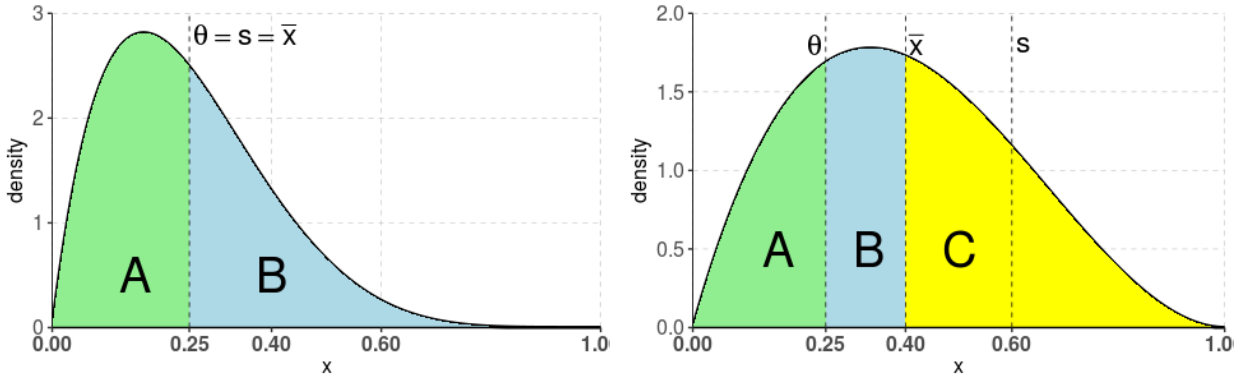


Figure 4: An example theoretical density f of forecasts in two example cases. A, B and C represent the shaded regions.

Figure 4a represents the case where $\bar{x}$ and θ coincides, which follows from $s = \theta$. Then, p_x and p_z both represent the region B and we have $p_x = p_z$. The average prediction $\bar{x}$ reflects θ. In contrast, Figure 4b shows an example density of predictions where the shared signal s differs from θ, resulting in $\bar{x} > \theta$. Observe that in Figure 4b, p_z represents the region $B + C$ while p_x corresponds to the area marked by C only. In this case, we see $\bar{x} > \theta$ due to $s > \theta$. As a result, the percentage of predictions that exceed $\bar{x}$ is lower and we have $p_z > p_x$.

Definition 1 (**Overshoot surprise**). *An overshoot surprise occurs when $p_x \neq p_z$. The overshoot surprise is positive if $p_z > p_x$ and negative if $p_z < p_x$. The size of the overshoot surprise is given by $\Delta p = p_z - p_x$.*

In Figure 4b, we observe a positive overshoot surprise, which indicates an upward bias in $\bar{x}$. A negative overshoot surprise would ensue if $\bar{x}$ is biased downwards instead. In contrast, there is no overshoot surprise in Figure 4a where $\bar{x}$ is unbiased.

Figure 4 suggests that p_z reveals important information on the density of predictions. Consider the cumulative density F associated with the density of predictions in Figure 4b:

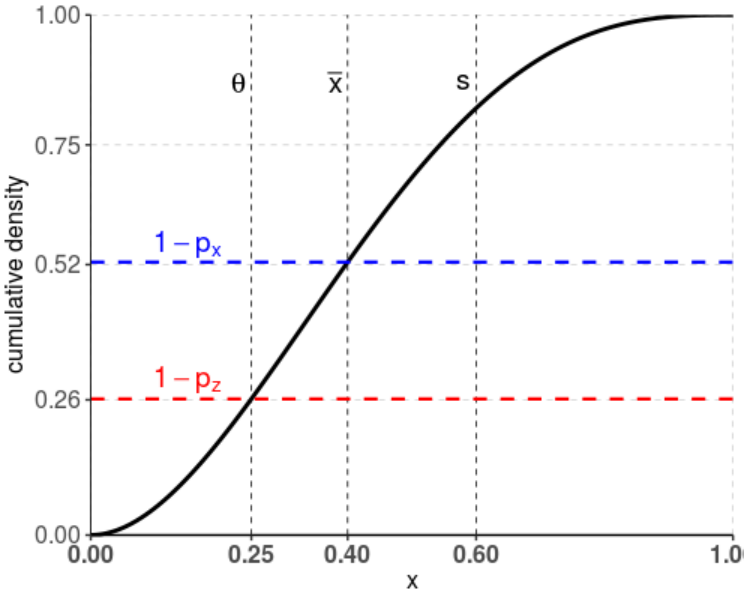


Figure 5: Cumulative density F of predictions for the example in Figure 4b

Since p_z measures the proportion of predictions that overshoot θ, $1-p_z$ gives the quantile of the cumulative density that corresponds θ. We can generalize this finding as follows:

Lemma 1. *Let F denote the continuous population density of predictions. Then, $\theta = F^{-1}(1-p_z)$*

Lemma 1 suggests that when the predictions and meta-predictions in the whole population ($N \to \infty$) are known and the population density of predictions is continuous, the exact θ can be recovered using p_z and F. The measure p_z simply reveals the cumulative density of predictions that overshoot θ, implying that $1-p_z$ gives the quantile of F that corresponds to θ.

In practice, a DM can only recruit a finite sample of agents. The population distribution of predictions and the population quantiles are unknown. Let $\hat{F}$ be the empirical cumulative density of the predictions from a random finite sample of agents of size N. Also let $\hat{p}_z = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(z_i > \bar{x})$ be the sample overshoot rate in meta-predictions in a finite sample of size N. The definition below introduces the Surprising Overshoot (SO) algorithm:

Definition 2 (The Surprising Overshoot algorithm). *The Surprising Overshoot algo-*

rithm constructs the estimator x^{SO} for θ following the steps below:

1. *Elicit sample predictions and meta-predictions.*
2. *Calculate* $\hat{p}_z = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(z_i > \bar{x})$
3. *Set* $x^{SO} = inf\{y|\hat{F}(y) \geq 1 - \hat{p}_z\})$.

The SO algorithm simply locates the $1 - \hat{p}_z$ quantile of the sample predictions. Lemma 1 established that when predictions follow a continuous density, the unknown probability is simply the population quantile $1 - p_z$. Since the population density and the corresponding quantile are unknown, the SO algorithm relies on the sample counterparts.

In finite samples or in prediction problems where the underlying population is not continuous, x^{SO} may have a non-zero bias. However, we could still expect x^{SO} to have a higher accuracy compared to $\bar{x}$. A positive (negative) overshoot surprise indicates an upward (downward) bias in $\bar{x}$, in which case x^{SO} is expected to be lower (higher) than $\bar{x}$ as $1 - \hat{p}_z$ is expected to be a smaller (larger) quantile than $1 - \hat{p}_x$. Section 4 presents empirical evidence suggesting that overshoot surprises strongly correlate with the forecasting errors of average prediction as predicted by the theoretical framework. Thus, the SO estimator adjusts the average prediction in the right direction. When there is no overshoot surprise, we expect $\hat{p}_x$ and $\hat{p}_z$ to be the same. The SO algorithm is expected to produce $x^{SO} = \bar{x}$ and not distort the average prediction if it is unbiased.

The SO estimator relies on the empirical distribution of predictions as well as agents' meta-predictions. This property has implications about the prediction problems where we may expect the SO algorithm to be more effective. To illustrate, consider the two example empirical densities below. Both figures depict predictions from a sample of 10 agents where the sample average prediction is 0.4 while $\theta = 0.25$. In Figure 6a agents report one of $0.5, 0.3$ or 0.1 as prediction. The distribution of predictions in Figure 6b is more dispersed around the average prediction. Suppose the meta-predictions in each example (not shown on figures)

are such that $\hat{p}_z = 0.2$ in both cases. Then the SO estimate is $1 - \hat{p}_z = 0.8$ quantile of the empirical density of predictions. The orange bar in each figure locates the SO estimate.

(a) Low dispersion (b) High dispersion

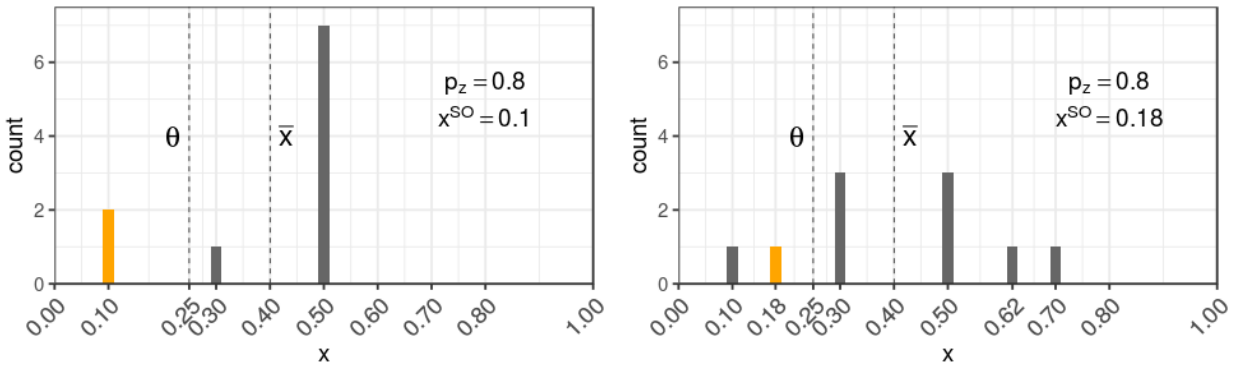


Figure 6: Two examples of empirical density of predictions

The SO estimate is more accurate in the high dispersion case simply because the 0.2 quantile falls closer to θ. The SO algorithm simply picks a reported prediction that corresponds to the sample quantile $1 - \hat{p}_z$. So the set of values x^{SO} can take depens on the empirical density of predictions. Even when $1 - \hat{p}_z$ provides an accurate estimate of the cumulative density at θ, the SO estimate may not be more accurate than $\bar{x}$ simply because $1 - \hat{p}_z$ quantile of the sample predictions is not close to θ. Such cases are less likely when the sample size is higher and/or the empirical density of predictions is more dispersed, as in Figure 6b. Therefore, we may expect the SO algorithm to perform better in larger samples and when the predictions are more dispersed. Intuitively, high dispersion can be considered as representing prediction tasks where individual judgments disagree, which could occur when the event of interest is highly uncertain and there is no strong expert consensus. The following sections test the SO algorithm using experimental data. In the analyses below, sample size and dispersion of predictions are considered as factors of interest.

4 Testing the SO algorithm

This section outlines the empirical methodology and presents some preliminary evidence on overshoot surprises. I use data from various experimental studies to test the SO algorithm. Section 4.1 provides information on the data sets. In testing the SO algorithm, I follow a comparative approach. The analysis will implement various alternative methods as a benchmark and test if the SO algorithm performs significantly better. Section 4.2 introduces the benchmarks. Finally, Section 4.3 provides some preliminary findings on overshoot surprises and how they relate to biases in the simple average.

4.1 Data sets

In testing the SO algorithm, I use data from three experimental studies. The first data set comes from Study 1 in Palley and Soll (18). In this study, Palley and Soll (18) conducted an online experiment where subjects reported their prediction and meta-prediction on the number of heads in 100 flips of a biased two-sided coin. The bias (the actual probability of heads) is unknown to the subjects, who are recruited on Amazon Mechanical Turk. Prior to submitting her report on a coin, each subject observed two independent samples of flips. One sample is common to all subjects and represents the shared signal. The second sample is subject-specific and constitutes a subject's private signal. A subject's best guess on the number of heads in 100 new flips a coin is effectively her best guess on the unknown bias. Thus, the prediction task elicits predictions on a probability unknown to subjects.

Study 1 in Palley and Soll (18) implements 3 different information structures. All subjects observe the shared signal and a private signal in the 'Symmetric' setup while only a subset of subjects observe a private signal in the 'Nested-Symmetric' structure. Private signals are subject-specific and unbiased in both structures, which agrees with the theoretical framework of the SO algorithm. The other setup is referred to as the 'Nested' structure, in which private signals are not subject-specific. The average of private signals do not converge to the true

value, which deviates from the theoretical framework of the SO algorithm. Thus, I exclude the 'Nested' structure and use the 48 prediction tasks (48 distinct coins) from the 'Symmetric' and 'Nested-Symmetric' structures only.

The Coin Flips data set from Palley and Soll (18)'s Study 1 presents an opportunity to test the SO algorithm in a controlled setup. Since latent probabilities (biases of the coins) are known to the analyst, it is possible to calculate prediction errors directly. The number of subjects per coin vary between 101 and 125, which allows an analysis on the effect of sample size. Section 5 elaborates on this analysis. Palley and Soll (18) run a second study where they use the same tasks as in Study 1 but vary subjects' incentives. Certain experimental conditions in the second study replicates Study 1. However the sample sizes are much smaller. Thus, I use data from Study 1 only.

The second source of data is the two experimental studies conducted by Wilkening et al. (23). In their first experiment, Wilkening et al. (23) replicated the experiment initially conducted by Prelec et al. (20). For each U.S. state, subjects are asked if the most populated city is the capital of that state. Prelec et al. (20) required subjects to pick true or false and report the percentage of other subjects who would agree with them. Wilkening et al. (23) asked subjects to report probabilistic predictions and meta-predictions on the statement (largest city being the capital city), which allows us to implement the SO algorithm. There are 89 subjects in total and each subject answered 50 questions (one per state). In the second experiment, subjects are presented with U.S. grade school level true/false general science statements. Examples include 'Water boils at 100 degrees Celsius at sea level', 'Materials that let electricity pass through them easily are called insulators' and 'Voluntary muscles are controlled by the cerebrum' etc. The experiments elicits judgments on 500 statements in total. Each subject reports her prediction and meta-prediction on the probability of a statement being true for 100 such statements. The number of subjects reporting on a given statement varies between 89 to 95. The State Capital and General Knowledge data sets allow us to test the SO algorithm in relatively more practical settings where only binary

outcomes can be observed.

4.2 Benchmarks

The benchmarks in testing the SO algorithm can be categorized in two groups. I will first consider *simple benchmarks*, namely the simple average and median prediction. Simple averaging is an easy and intuitive aggregation method. The median forecast is also popular because it is more robust to outliers. These simple aggregation methods do not require meta-predictions, which makes them easier to implement. However, as shown in Section 2 with simple averaging, these methods may produce biased aggregate judgments. As discussed in Section 1, there exists a growing literature which provide more sophisticated solutions to the aggregation problem utilizing meta-beliefs. I consider three *advanced benchmarks*: Palley and Soll (18)'s pivoting, Palley and Satopää (17)'s knowledge-weighting (KW) and Martinie et al. (14)'s meta-probability weighting (MPW).

The pivoting method first computes simple average of predictions and meta-predictions, $\bar{x}$ and $\bar{z}$ in our notation respectively. Then the mechanism pivots from $\bar{x}$ in different directions. The pivot in the direction of $\bar{z}$ provides an estimate for the shared information while the step in the opposite direction gives an estimate for the average of private signals. These estimates are combined using Bayesian weights to produce the optimal aggregate estimate. The canonical pivoting method requires the knowledge of Bayesian weight ω to determine the optimal pivot size and optimal aggregation. Palley and Soll (18) propose *minimal pivoting* (MP) as a simple variant which adjusts $\bar{x}$ by $\bar{x} - \bar{z}$. The adjustment moves the aggregate estimate away from the shared information and alleviates the shared-information bias. MP does not require the knowledge of ω but it only partially corrects for the bias in $\bar{x}$.

The KW mechanism proposes a weighted crowd average as the aggregate prediction. The weights are estimated by minimizing the peer prediction gap, which measures the accuracy of weighted crowds' aggregate meta-prediction in estimating the average prediction. In a similar framework to Section 2, Palley and Satopää (17) show that minimizing the peer

prediction gap is a proxy for minimizing the mean squared error of a weighted aggregate prediction. Intuitively, KW builds on the idea that a weighted crowd that is accurate in predicting others could be more accurate in predicting the unknown quantity itself as well. The mechanism's aggregate prediction is simply the weighted average prediction of such a crowd.

The MPW algorithm aims to construct a weighted average of probabilistic predictions. Martinie et al. (14) consider a slightly different Bayesian setup where agents receive a private signal from one of the two signal technologies, one for experts and the other for novices. The absolute difference between an agent's optimal prediction and meta-prediction is higher if the agent's signal is more informative. Based on this result, the MPW algorithm assigns weights proportional to the absolute differences between their prediction and meta-prediction. It is expected that agents with more informative private signals receive higher weights and the weighted average becomes more accurate as a result.

Similar to the advanced benchmarks listed above, the SO algorithm relies on an augmented elicitation procedure which elicits meta-predictions in addition to predictions. In contrast, the mechanisms in simple benchmarks do not require information from meta-predictions. Thus we may expect the SO algorithm to significantly outperform simple benchmarks. The advanced benchmarks have similar information demands to the SO algorithm. Thus, comparisons with the advanced benchmarks may produce different results in different tasks.

4.3 Preliminary evidence on overshoot surprises

Section 3 established a relationship between the size and direction of overshoot surprises and biases. The more p_z differs from p_x, higher the overshoot surprise, suggesting a stronger bias in the crowd average. Presence of an overshoot surprise relates to the performance of SO algorithm as well. We may expect a larger error reduction from using the SO algorithm when $|p_z - p_x|$ is larger as x^{SO} will be further away than the biased $\bar{x}$.

The Coin Flips data set presents an opportunity to investigate whether overshoot surprises correlate with biases. In this experiment, both the shared signal s and the unknown probability θ in each coin are generated by the experimenter. The shared-information bias in $\bar{x}$ is proportional to the absolute difference between s and θ. Furthermore, the bias would be downwards (upwards) if $s < \theta$ ($s > \theta$). Recall that positive (negative) overshoot surprises are associated with upward (downward) biases and we expect no overshoot surprise if $\bar{x}$ is unbiased, which corresponds to the case $s = \theta$. Since the information on s and θ are available, we can investigate if this pattern is observed in the data. Figure 7 shows the relationship between overshoot surprises and $s - \theta$. Each dot represents an item and the blue lines shows the best linear fit.

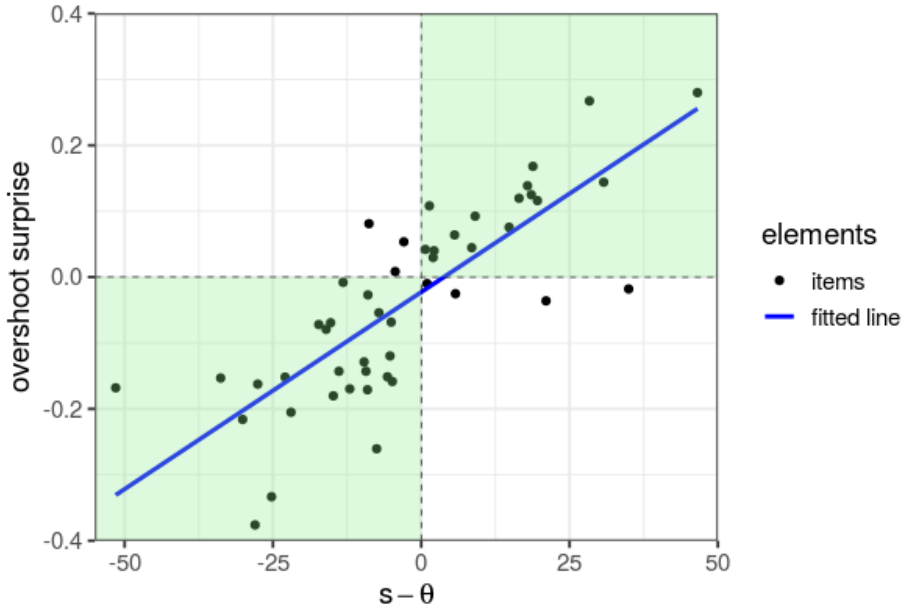


Figure 7: The relationship between the shared-information bias and overshoot surprises in prediction tasks. Shaded areas show the regions where the direction of bias and overshoot surprise is as predicted by the theory.

Figure 7 shows a strong linear association between biases and overshoot surprises. Also observe that most of the points are within the shaded regions. A positive (negative) overshoot surprise is much more likely to occur when we have an upward (downward) bias in $\bar{x}$, which corresponds to $s > \theta$ ($s < \theta$). In addition, the magnitude of overshoot surprises are higher

when biases are more substantial. An overshoot surprise is indeed a strong indicator of the size and direction of a bias in the crowd average. The SO estimator can be thought of as $\bar{x}$ adjusted away from the direction of the bias, identified by the sign of the overshoot surprise. Furthermore, $|x^{SO} - \bar{x}|$ is proportional to the magnitude of the overshoot surprise. Thus, the evidence from overshoot surprises suggests potential error reduction from using the SO algorithm. Section 5 explores whether the SO algorithm improves over various benchmarks.

5 Results

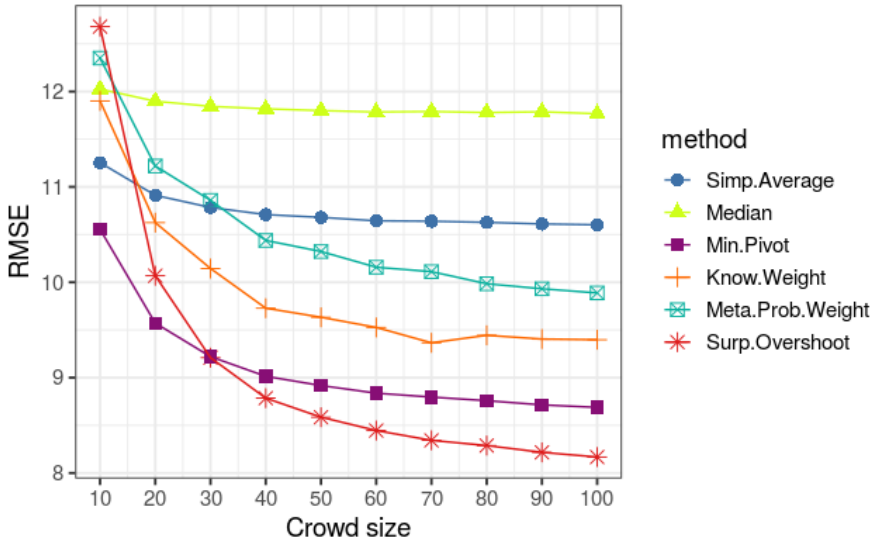
5.1 Coin Flips data

In testing the SO estimator with Coin Flips data, I follow a bootstrap approach similar to Palley and Satopää (17). For each item in a given data set, a subset of subjects of size M is randomly selected to construct a bootstrap sample. For each sample and item, I compute the absolute errors of aggregate predictions from the benchmarks and the SOA. The average of absolute errors across the items gives a measure of the corresponding method's error in that task. This procedure is run 1000 times for each crowd size to obtain 1000 data points on each method's errors for a given crowd size. The observations from bootstrap samples allow us to test for differences in errors between the SO algorithm and a benchmark. I consider two measures for comparison. Firstly, I calculate average RMSE across all iterations for each method. Secondly, I log transform the errors and calculate pairwise differences for each iteration to construct 95% bootstrap confidence intervals for statistical inference. The differences in log-transformed errors can be interpreted as percentage error reduction. The bootstrap approach also allows us to see the effect of crowd size on the relative performance of the SO algorithm.

Figure 8 presents the results of the bootstrap analysis. Figure 8a depicts the average RMSE across iterations. Figure 8b shows the bootstrap confidence intervals for reduction in log absolute error (the SO estimator vs benchmark). The two panels in 8b depict the error

reductions compared to simple and advanced benchmarks respectively.

(a) Average RMSE (across iterations) vs crowd size



(b) Reduction in log absolute error (averaged across items) in Bootstrap samples

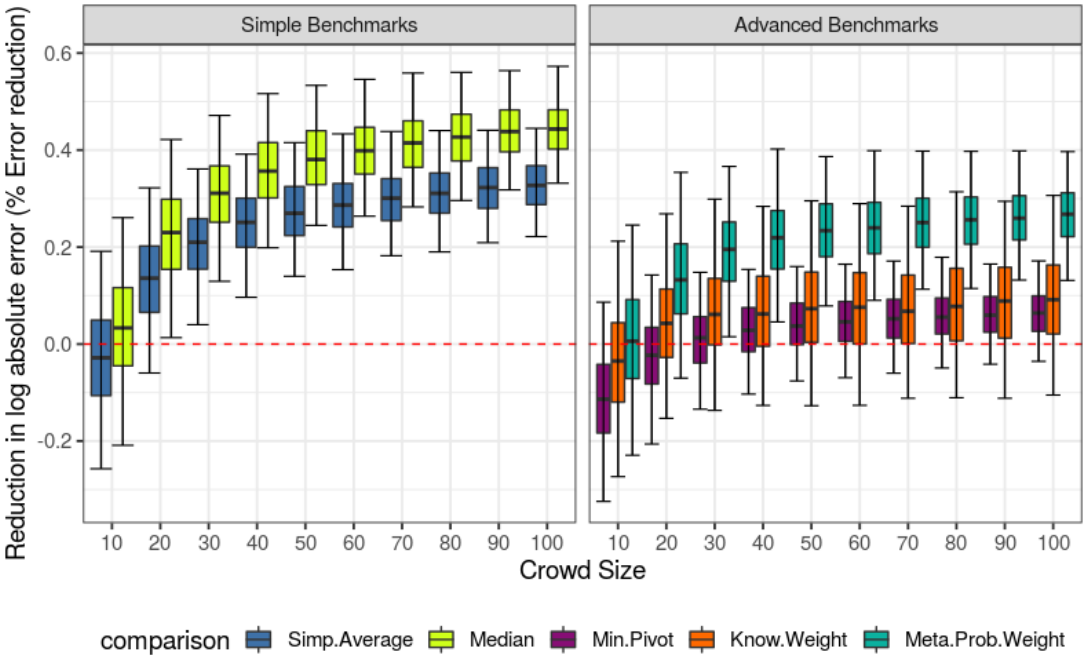


Figure 8: Results of bootstrap analysis on Coin Flips data

Figure 8a shows that the SO algorithm achieves the lowest errors in samples of more than 30 subjects. Observe that increasing the sample size has a stronger effect on the SO estimator. Almost all aggregation methods benefit from larger samples due to the wisdom

of crowds effect. For the SO algorithm, benefits of a larger crowd could be twofold. Not only the wisdom of crowds effect becomes more pronounced, but also a larger sample of predictions is typically more dispersed and allows a more accurate approximation of the population density of predictions.

Figure 8b indicates that the SO algorithm significantly outperforms the simple benchmarks. We also see that the SO algorithm achieves lower errors in most bootstrap samples compared to the advanced benchmarks. The differences are significant only for the MPW method. Appendix A provides the 95% bootstrap confidence intervals depicted in Figure 8b. The SO algorithm improves the accuracy by 20-40% relative to the simple benchmarks. Compared to the MPW, KW and MP methods, the median percentage error reduction in large samples is around 25%, 9% and 6% respectively.

The Coin Flips study elicits judgments on a controlled prediction task where all private signals are equally informative and the data generation process of signals induces a distribution of predictions. The following section presents evidence from General Knowledge and State Capital data. In those tasks, subjects may differ in knowledge and the distribution of predictions varies across items.

5.2 State Capital and General Knowledge data

I now consider the State Capital and General Knowledge data sets from the experimental studies in Wilkening et al. (23). As discussed in Section 4.1, items in these data sets have a binary truth. The analysis in this section tests if the SO algorithm produces a probabilistic estimate closer to the true answer. I follow a similar approach to Budescu and Chen (2) and Martinie et al. (14) and calculate transformed Brier scores associated with the aggregate estimates of each method in each data set. The transformed Brier score of a method i in a

given data set is defined as

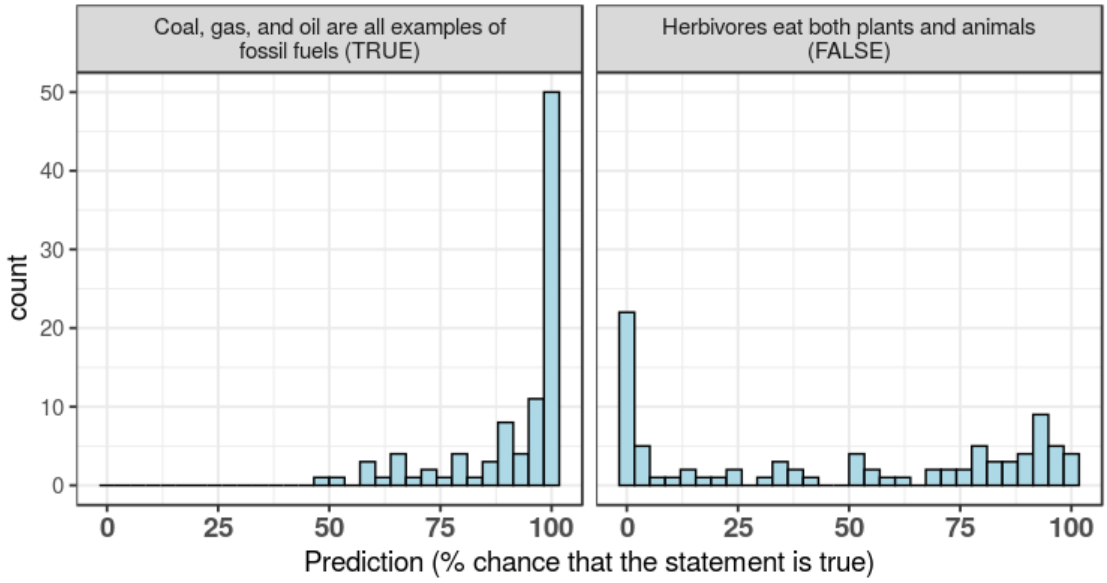
$$S_i = 100 - 100 \sum_{j=1}^{J} \frac{(o_j - x_j^i)^2}{J}$$

where $o_j \in \{0, 1\}$ be the outcome of event j, J is the total number of events in the data set and $x_j^i \in [0, 1]$ is the aggregate probabilistic prediction of method i on event j. The transformed Brier score is strictly proper and assigns a score within $[0, 100]$.

The main question of interest is whether the SO algorithm achieves significantly higher transformed Brier scores compared to the benchmarks. For statistical inference, I will construct 95% confidence intervals for paired differences in Transformed Brier scores using a bootstrap approach. I generate 1000 bootstrap samples for each method in each data set. A bootstrap sample consists of items sampled with replacement and provides a measurement on S_i. Then, I calculate the paired differences between the scores of the SO estimator and the other benchmarks for each bootstrap sample. Similar to the analysis in Coin Flips data, the 2.5% and 97.5% quantiles of pairwise differences provide a 95% bootstrap confidence interval for testing SO algorithm against a benchmark.

Section 3 discussed why the SO algorithm could be more effective when predictions are more dispersed. The tasks in General Knowledge and State Capital data sets differ in terms of difficulty and the presence of a strong consensus among the predictions. This allows us to investigate how the extent of disagreement in predictions relates to the relative performance of SO algorithm. To illustrate, consider the two items from the General Knowledge data in Figure 9 below:

Figure 9: Predictions on two example items from the General Knowledge data

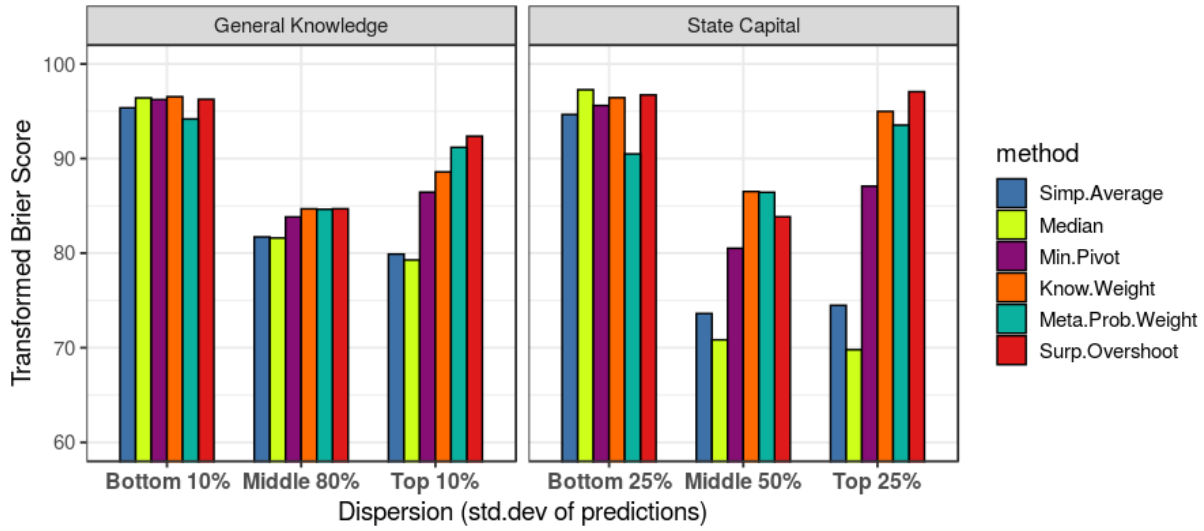


For the item in the left panel, a large proportion of predictions are at 100% and almost all predictions are 50% or higher. The dispersion of predictions is relatively smaller compared to the item in the right panel, where prediction vary from 0% to 100%. Similar examples can be found in the State Capital data. Given such variety, we can categorize the items in terms of the dispersion of predictions and calculate transformed Brier scores for each category of items. For the main results below, I use standard deviation of predictions as the measure of dispersion in an item. Appendix B replicates the same analysis using kurtosis as the measure and finds very similar results. In the General Knowledge data, I categorize the items in three groups in terms of the standard deviation of predictions: bottom 10%, middle 80% and top 10%. The bottom and top 10% items represent the low and high dispersion items respectively. The State Capital data includes a lower number of items. In order to have a reasonable number of items in each category, the thresholds are set at 25% and 75%. Thus, the categories in the State capital data are bottom 25%, middle 50% and top 25%. The transformed brier score will be calculated separately for each dispersion category.

The discussion in Section 3 suggests that the SO algorithm would be relatively more effective in the high-dispersion items. Figure 10 below shows the transformed Brier scores

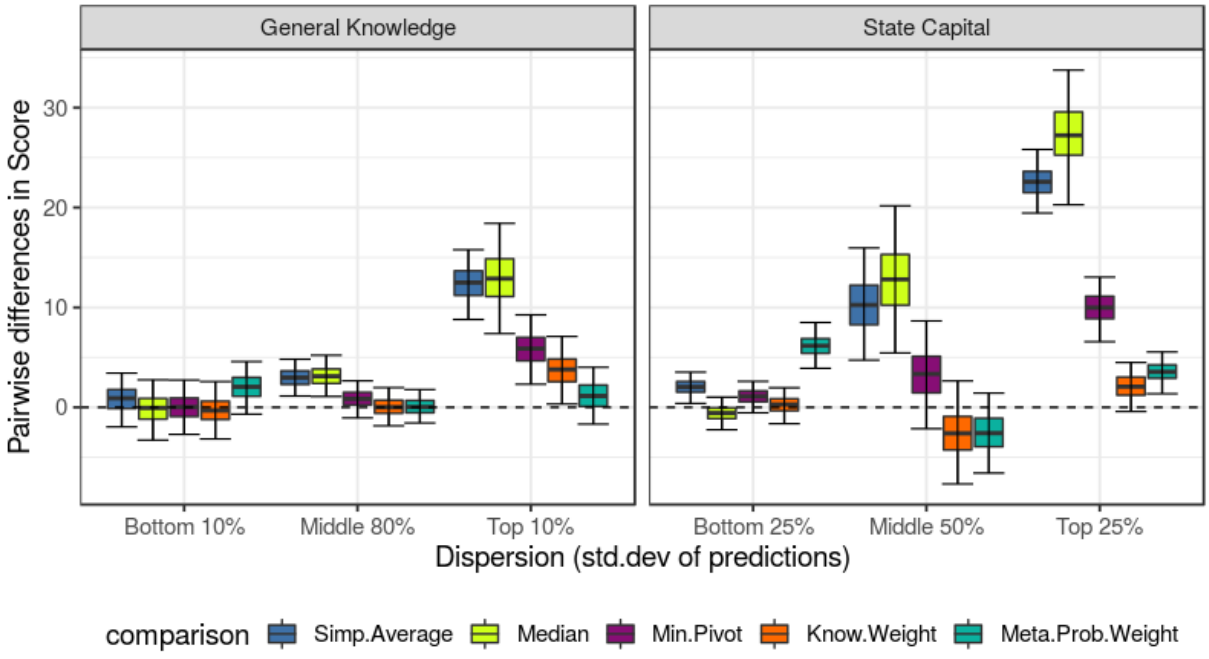
for each method across various levels of dispersion in the General Knowledge and State Capital items.

Figure 10: Transformed Brier Scores in the two data sets



In both data sets, the transformed Brier scores are comparable across all methods in low-dispersion items. Individual predictions are more spread in high-dispersion items, implying a higher frequency of inaccurate judgments. As in Figure 9 (the right panel), there could be many predictions putting a high probability on 'True' even though the correct answer is 'False', reducing the accuracy of simple aggregates. Figure 10 demonstrates the the advanced aggregation algorithms strongly outperform simple aggregates in such cases. In addition, the SO algorithm achieves the best transformed Brier score among items with highest dispersion in either data set. So, in high-dispersion items the SO algorithm is more effective relative to other advanced benchmarks as well. Figure 11 below presents 95% bootstrap confidence intervals for pairwise differences in transformed Brier scores. An observation above the 0-line indicates that the SO estimator achieved a higher transformed Brier score than the corresponding benchmark in that particular bootstrap sample.

Figure 11: Bootstrap differences in Transformed Brier Scores



Appendix A provides the Bootstrap confidence intervals depicted in Figure 11. The confidence intervals suggest that the SO estimator significantly outperforms simple benchmarks in moderate and high dispersion items. Furthermore, the SO algorithm compares favorably to the advanced benchmarks in high-dispersion items. Similar to the results in Coin Flips study, the SO algorithm is never worse than the advanced benchmarks while being significantly more accurate in some cases.

To summarize, results indicate that the SO algorithm is relatively more effective in samples of more than 30 experts and when individual predictions disagree greatly, resulting in a more dispersed empirical density of predictions. provides a further discussion on the merits and limitations of the SO algorithm.

6 When and why is the SO algorithm effective?

The findings in Section 5 not only document the effectiveness of the SO algorithm but also provides a 'user's manual' for a DM who intends to use an aggregation algorithm to

combine probabilistic judgments. Two important features of the aggregation problem allow a characterization of the situations where the SO algorithm could be an effective solution. Firstly, the decision maker knows the size of the sample of experts, which is relevant for the SO algorithm. Secondly the empirical density of predictions, which is an input for the SO algorithm, is observable to the DM prior to the resolution of the uncertain event.

Section 5.1 showed that, compared to the benchmarks, the forecast errors of the SO algorithm decreases even more rapidly as the sample size increases. The SO algorithm has the highest average error in samples 10 experts while in samples of 30 or more experts, the SO algorithm achieves the lowest errors. Intuitively, the SO algorithm is more sensitive to the sample size because it relies on the sample density of predictions. The sample quantiles may overlap in very small samples. As the sample size increases, the sample density becomes more representative of the underlying population density and the quantiles could become more distinct. Then the SO algorithm can produce a more fine-tuned aggregate prediction. The DM might prefer the SO algorithm over the benchmarks if she has to access to a moderate to large sample of experts. In small samples, simple aggregation methods or the MP method could be a better alternative.

The disagreement between experts is also a factor in the effectiveness of the SO algorithm. Consider a situation where there is a strong consensus among experts: individual predictions are clustered around a certain value (low dispersion). We can imagine two scenarios in which the DM would observe such a pattern. Experts could be highly accurate individually, in which case a simple average of predictions would perform sufficiently well. The lowest categories of dispersion depicted in Figure 10 represent such cases. The simple aggregation methods are as accurate as the more sophisticated aggregation algorithms. The second scenario is where experts predictions are clustered around an inaccurate value. Previous work developed aggregation algorithms to pick a correct answer to a multiple choice question when the majority vote is inaccurate (20, 23). An analogous solution in aggregating probabilistic judgments when most experts are inaccurate could be identifying a contrarian

but accurate prediction and discarding other predictions. As discussed in Section 4.2, the KW and MPW mechanisms determine individual weights for aggregation. Other than some extreme cases, these mechanisms are highly unlikely to attach 0 weight to a very high proportion of predictions. The MP method makes an adjustment based on average prediction and meta-prediction. It does not attempt to locate more accurate experts. In theory, the SO algorithm can pick the sample quantile that corresponds to the contrarian prediction. Thus, the DM might prefer the SO algorithm. However, the sample quantiles are close to each other when predictions are highly clustered. Thus, the SO algorithm's adjustment may not be sufficiently extreme. Alternatively, if the DM expects a strong consensus with reasonably well-calibrated individual expert predictions, she might prefer eliciting the predictions only and using a simple aggregation method.

Now consider a situation of high dispersion in predictions instead. Experts disagree in their predictions and some experts are less accurate (ex-post) compared to others. The top dispersion categories in General Knowledge and State Capital studies represent this case. Figure 11 suggests that the SO algorithm not only outperforms the simple aggregation methods, but also it is relatively more effective compared to the advanced benchmarks. The SO algorithm performs well under more disagreement because, similar to the intuition earlier, the sample quantiles become more distinct and the SO algorithm has a higher room for improvement. High dispersion in predictions allows more precision in the SO estimator. Thus, a DM who observes strong disagreement among individual predictions might prefer the SO algorithm when it comes to constructing an aggregate prediction. Note that the aggregation problem is more tricky when experts disagree. The absence of a clear consensus is not uncommon despite the fact that experts typically have shared information through their field of expertise. The SO algorithm is particularly effective in problems where the DM might need an effective aggregation algorithm the most.

The SO algorithm differs from the other alternative aggregation algorithms in its use of the empirical density of predictions. For a given level of overshoot surprise, the absolute

difference between the SO estimator and the average prediction depends on the dispersion in the empirical density of predictions. However, the SO algorithm always sets an aggregate estimate that lies within the range of individual predictions. Recall that the MP method uses a fixed step size to adjust the average prediction. In contrast, the SO algorithm's adjustment on the aggregate prediction is informed and restrained by the empirical density. This makes the SO estimator more robust to potential over-adjustments, which would reduce the accuracy even when the aggregate estimate is adjusted in the correct direction (i.e. away from the shared-information bias).

7 Conclusion

Very few decision problems have total absence of uncertainty. Decision makers frequently face the problem of predicting the likelihood of an uncertain event as part of a decision making process. Leveraging the collective wisdom of many experts is shown to be a promising solution. However, the use of collective wisdom is not a trivial solution because there are typically no guidelines as to how individual judgments should be aggregated for maximum accuracy. Experts typically have shared information through their training, overlapping past experiences, knowledge of the same academic works etc. In such cases, the simple average of predictions exhibits the shared-information problem (18). Recent work developed aggregation algorithms that rely on an augmented elicitation procedure (19, 20, 18, 17, 23). These algorithms use individuals' meta-beliefs to aggregate predictions more effectively. This paper proposes a novel algorithm to aggregate probabilistic judgments on the likelihood of an event. The Surprising Overshoot algorithm uses experts' probabilistic meta-predictions to aggregate their probabilistic predictions. Unlike other aggregation methods, the SO algorithm also utilizes the information in the empirical density of predictions.

Experimental evidence suggests that the SO algorithm consistently outperforms simple averaging and median prediction. Furthermore, the SO algorithm compares favorably to

alternative aggregation algorithms that elicit meta-beliefs (18, 17, 14). The SO algorithm is particularly effective when the empirical density of predictions is highly dispersed. Such high dispersion is more likely in moderate to large samples of experts and in prediction tasks where experts disagree in their individual assessment.

In practice, a DM is more likely to need a judgment aggregation algorithm when expert predictions lack a clear consensus. Experts frequently disagree in their judgments even though they have overlapping information. In such decision problems, the DM could find herself with conflicting expert judgments with no straightforward way to combine them. Because of its effectiveness in aggregating disagreeing judgments, the SO algorithm could be especially powerful in such challenging aggregation problems. The dispersion in predictions that result from the disagreement among experts works in the algorithm's favor.

The empirical evidence in this paper comes from three controlled experiments. In General Knowledge and State Capital data sets, a subset of subjects might know the correct answer in a given item. In many practical prediction problems, the actual state of the world is associated with an outcome that is not realized yet. Experts will be asked to report a prediction for the latent probability of occurrence and even the most extreme experts may not be absolutely sure. Future work may implement the SO algorithm in such prediction tasks to test its effectiveness in other practical forecasting problems.

References

[1] Armstrong, J. S. (2001). Combining forecasts. In *Principles of forecasting*, pages 417–439. Springer.

[2] Budescu, D. V. and Chen, E. (2015). Identifying Expertise to Extract the Wisdom of Crowds. *Management Science*, 61(2):267–280.

[3] Chen, K.-Y., Fine, L. R., and Huberman, B. A. (2004). Eliminating public knowledge biases in information-aggregation mechanisms. *Management Science*, 50(7):983–994.

[4] Clemen, R. T. (1989). Combining forecasts: A review and annotated bibliography. *International journal of forecasting*, 5(4):559–583.

[5] Clemen, R. T. and Winkler, R. L. (1986). Combining economic forecasts. *Journal of Business & Economic Statistics*, 4(1):39–46.

[6] Genre, V., Kenny, G., Meyler, A., and Timmermann, A. (2013). Combining expert forecasts: Can anything beat the simple average? *International Journal of Forecasting*, 29(1):108–121.

[7] Gneiting, T. and Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378.

[8] Kahneman, D., Slovic, S. P., Slovic, P., and Tversky, A. (1982). *Judgment under uncertainty: Heuristics and biases*. Cambridge university press.

[9] Kahneman, D. and Tversky, A. (1973). On the psychology of prediction. *Psychological review*, 80(4):237.

[10] Larrick, R. P. and Soll, J. B. (2006). Intuitions about combining opinions: Misappreciation of the averaging principle. *Management science*, 52(1):111–127.

[11] Makridakis, S. and Winkler, R. L. (1983). Averages of Forecasts: Some Empirical Results. *Management science*, 29(9):987–996.

[12] Mannes, A. E., Larrick, R. P., and Soll, J. B. (2012). The social psychology of the wisdom of crowds. In Krueger, J. I., editor, *Frontiers of social psychology. Social judgment and decision making*, pages 227–242. Psychology Press.

[13] Mannes, A. E., Soll, J. B., and Larrick, R. P. (2014). The wisdom of select crowds. *Journal of personality and social psychology*, 107(2):276.

[14] Martinie, M., Wilkening, T., and Howe, P. D. (2020). Using meta-predictions to identify experts in the crowd when past performance is unknown. *Plos one*, 15(4):e0232058.

[15] Min, C.-k. and Zellner, A. (1993). Bayesian and non-bayesian methods for combining models and forecasts with applications to forecasting international growth rates. *Journal of Econometrics*, 56(1-2):89–118.

[16] Palan, S., Huber, J., and Senninger, L. (2019). Aggregation mechanisms for crowd predictions. *Experimental economics*, pages 1–27.

[17] Palley, A. and Satopää, V. (2020). Boosting the wisdom of crowds within a single judgment problem: Selective averaging based on peer predictions. *Available at SSRN 3504286*.

[18] Palley, A. B. and Soll, J. B. (2019). Extracting the wisdom of crowds when information is shared. *Management Science*, 65(5):2291–2309.

[19] Prelec, D. (2004). A Bayesian Truth Serum for Subjective Data. *Science*, 306(5695):462–466.

[20] Prelec, D., Seung, H. S., and McCoy, J. (2017). A solution to the single-question crowd wisdom problem. *Nature*, 541(7638):532–535.

[21] Surowiecki, J. (2004). *The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations*. Doubleday & Co, New York, NY, US.

[22] Tversky, A. and Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *science*, 185(4157):1124–1131.

[23] Wilkening, T., Martinie, M., and Howe, P. D. (forthcoming). Hidden experts in the crowd: Using meta-predictions to leverage expertise in single-question prediction problems. *Management Science*.

[24] Winkler, R. L., Grushka-Cockayne, Y., Lichtendahl, K. C., and Jose, V. R. R. (2019). Probability forecasts and their combination: A research perspective. *Decision Analysis*, 16(4):239–260.

Appendix A

Table A1: 95% Confidence intervals depicted in Figure 8b

C.Size	Comparison	Low.B.	Upp.B.	C.Size	Comparison	Low.B.	Upp.B.
10	Simp.Average	-0.26	0.19	60	Simp.Average	0.15	0.43
10	Median	-0.21	0.26	60	Median	0.26	0.55
10	Min.Pivot	-0.32	0.09	60	Min.Pivot	-0.07	0.16
10	Know.Weight	-0.27	0.21	60	Know.Weight	-0.13	0.29
10	Meta.Prob.Weight	-0.23	0.25	60	Meta.Prob.Weight	0.09	0.40
20	Simp.Average	-0.06	0.32	70	Simp.Average	0.18	0.44
20	Median	0.01	0.42	70	Median	0.28	0.56
20	Min.Pivot	-0.21	0.14	70	Min.Pivot	-0.06	0.17
20	Know.Weight	-0.15	0.27	70	Know.Weight	-0.11	0.28
20	Meta.Prob.Weight	-0.07	0.35	70	Meta.Prob.Weight	0.11	0.40
30	Simp.Average	0.04	0.36	80	Simp.Average	0.19	0.44
30	Median	0.13	0.47	80	Median	0.30	0.56
30	Min.Pivot	-0.13	0.15	80	Min.Pivot	-0.05	0.18
30	Know.Weight	-0.14	0.30	80	Know.Weight	-0.11	0.31
30	Meta.Prob.Weight	0.02	0.37	80	Meta.Prob.Weight	0.11	0.40
40	Simp.Average	0.10	0.39	90	Simp.Average	0.21	0.44
40	Median	0.20	0.52	90	Median	0.32	0.56
40	Min.Pivot	-0.10	0.15	90	Min.Pivot	-0.04	0.16
40	Know.Weight	-0.13	0.28	90	Know.Weight	-0.11	0.29
40	Meta.Prob.Weight	0.05	0.40	90	Meta.Prob.Weight	0.13	0.40
50	Simp.Average	0.14	0.42	100	Simp.Average	0.22	0.44
50	Median	0.24	0.53	100	Median	0.33	0.57
50	Min.Pivot	-0.08	0.16	100	Min.Pivot	-0.04	0.17
50	Know.Weight	-0.13	0.30	100	Know.Weight	-0.11	0.31
50	Meta.Prob.Weight	0.08	0.39	100	Meta.Prob.Weight	0.13	0.40

Table A2: 95% Bootstrap confidence intervals depicted in Figure 11, General Knowledge data

Comparison	Dispersion	Low.B.	Upp.B.
Simp.Average	Bottom 10%	-1.96	3.42
Median	Bottom 10%	-3.28	2.75
Min.Pivot	Bottom 10%	-2.71	2.73
Know.Weight	Bottom 10%	-3.17	2.57
Meta.Prob.Weight	Bottom 10%	-0.68	4.58
Simp.Average	Middle 80%	1.13	4.81
Median	Middle 80%	1.09	5.21
Min.Pivot	Middle 80%	-1.06	2.65
Know.Weight	Middle 80%	-1.85	1.98
Meta.Prob.Weight	Middle 80%	-1.56	1.78
Simp.Average	Top 10%	8.80	15.78
Median	Top 10%	7.38	18.42
Min.Pivot	Top 10%	2.31	9.26
Know.Weight	Top 10%	0.33	7.09
Meta.Prob.Weight	Top 10%	-1.69	4.00

Table A3: 95% Bootstrap confidence intervals depicted in Figure 11, State Capital data

Comparison	Dispersion	Low.B.	Upp.B.
Simp.Average	Bottom 25%	0.39	3.52
Median	Bottom 25%	-2.25	1.02
Min.Pivot	Bottom 25%	-0.55	2.59
Know.Weight	Bottom 25%	-1.65	1.96
Meta.Prob.Weight	Bottom 25%	3.90	8.49
Simp.Average	Middle 50%	4.72	15.95
Median	Middle 50%	5.44	20.18
Min.Pivot	Middle 50%	-2.14	8.65
Know.Weight	Middle 50%	-7.67	2.63
Meta.Prob.Weight	Middle 50%	-6.59	1.42
Simp.Average	Top 25%	19.45	25.82
Median	Top 25%	20.28	33.76
Min.Pivot	Top 25%	6.57	13.03
Know.Weight	Top 25%	-0.42	4.49
Meta.Prob.Weight	Top 25%	1.36	5.54

Appendix B

Figure B1: Transformed Brier Scores (measure of dispersion: kurtosis)

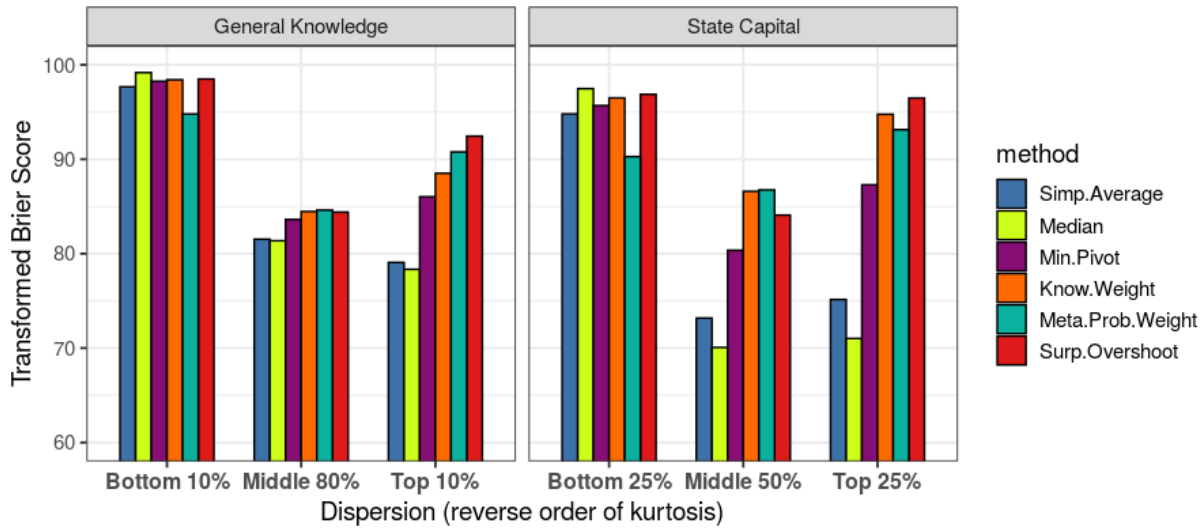


Figure B2: Bootstrap differences in Transformed Brier Scores (measure of dispersion: kurtosis)