CrossMark

Aggregating incoherent agents who disagree

Richard Pettigrew[1]

Abstract In this paper, we explore how we should aggregate the degrees of belief of a group of agents to give a single coherent set of degrees of belief, when at least some of those agents might be probabilistically incoherent. There are a number of ways of aggregating degrees of belief, and there are a number of ways of fixing incoherent degrees of belief. When we have picked one of each, should we aggregate first and then fix, or fix first and then aggregate? Or should we try to do both at once? And when do these different procedures agree with one another? In this paper, we focus particularly on the final question.

Amira and Benito are experts in the epidemiology of influenza. Their expertise, therefore, covers a claim that interests us, namely, that the next 'flu pandemic will occur

---

[1] An agent's *credence* in a proposition is her *degree of belief* in it. That is, it measures how confident she is in the proposition. Sometimes these are called her *subjective probabilities* or *forecasts of probabilities*. I avoid the latter terminology since it might suggest to the reader that these credences are probabilistically coherent, and we are interested in this paper in cases in which they are not. As I write them, credences are real numbers in the unit interval $[0, 1]$. Others write them as percentages. Thus, where I write that Amira has credence 0.5 in X, others might write that she has credence 50% in X or that she is 50% confident in X. Translating between the two is obviously straightforward.

✉ Richard Pettigrew
  Richard.Pettigrew@bris.ac.uk

1   Department of Philosophy, University of Bristol, Cotham House, Cotham Hill, Bristol BS6 6JL,
    United Kingdom

in 2019. Call that proposition X and its negation $\overline{X}$. Here are Amira's and Benito's credences or degrees of belief in that pair of propositions:[1]

	X	$\overline{X}$
Amira	0.5	0.1
Benito	0.2	0.6

We would like to arrive at a single coherent pair of credences in X and $\overline{X}$. Perhaps we wish to use these to set our own credences; or perhaps we wish to publish them in a report of the WHO as the collective view of expert epidemiologists; or perhaps we wish to use them in a decision-making process to determine how medical research funding should be allocated in 2018. Given their expertise, we would like to use Amira's and Benito's credences when we are assigning ours. However, there are two problems. First, Amira and Benito disagree—they assign different credences to X and different credences to $\overline{X}$. Second, Amira and Benito are incoherent—they each assign credences to X and $\overline{X}$ that do not sum to 1. How, then, are we to proceed? There are natural ways to aggregate different credence functions; and there are natural ways to fix incoherent credence functions. Thus, we might fix Amira and Benito first and then aggregate the fixes; or we might aggregate their credences first and then fix up the aggregate, if it is incoherent. But what if these two disagree, as we will see they are sometimes wont to do? Which should we choose? To complicate matters further, there is a natural way to do both at once—it makes credences coherent and aggregates them all at the same time. What if this one-step procedure disagrees with one or other or both of the two-step procedures, fix-then-aggregate and aggregate-then-fix? In what follows, I explore when such disagreements arise and what the conditions are that guarantee that they will not. Then I will explain how these results may be used in philosophical arguments. I begin, however, with an overview of the paper.

To begin, we consider only the case in which the propositions to which our disagreeing agents assign credences form a partition. Indeed, in Sects. 1–7, we consider only two-cell partitions—that is, our agents have credences only in a proposition and its negation. Having illustrated the central ideas of the paper in this simple setting, we then consider what happens when we move to n-cell partitions in Sect. 8. Finally, in Sect. 9, we consider agents who have credences in propositions that don't form a partition at all. Throughout, we assume that all agents have credences in exactly the same propositions. We leave the fully general case, in which the disagreeing agents may have credences in different sets of propositions, for another time.[2]

In Sect. 1, we present the two most popular methods for aggregating credences: linear pooling (LP) takes the aggregate of a set of credence functions to be their weighted arithmetic average, while geometric pooling (GP) takes their weighted geometric average and then normalises that. Then, in Sect. 2 we describe a natural method for fixing incoherent credences: specify a measure of how far one credence function lies from the other, and fix an incoherent credence function by taking the coherent function that is closest to it according to that measure. We focus particularly on two of the most popular such measures: *squared Euclidean distance* (SED) and *generalized*

---

[2] But see Predd et al. (2008) for some initial work on this question in the spirit of the present paper.

Kullback-Leibler divergence (GKL). In Sect. 3, we begin to see how the methods for fixing interact with the methods for aggregating: if we pair our measures of distance with our pooling methods carefully, they commute; otherwise, they do not. And we begin to see the central theme of the paper emerging: LP pairs naturally with SED (if anything does), while GP pairs with GKL (if anything does). In Sect. 4, we note that, just as we can fix incoherent credence functions by minimizing distance from or to coherence, so we can aggregate credence functions by taking the aggregate to be the credence function that minimizes the weighted average distance from or to those credence functions. The aggregation methods that result don't necessarily result in coherent credence functions, however. To rectify this, in Sect. 5 we introduce the Weighted Coherent Aggregation Principle, which takes the aggregate to be the *coherent* credence function that minimizes the weighted average distance from or to the credence functions to be aggregated. Up to this point, we have been talking generally about measures of the distance from one credence function to another, or only about our two favoured examples. In Sect. 6, we introduce the class of additive Bregman divergences, which is the focus for the remainder of the paper. Our two favoured measures, SED and GKL, belong to this class, as do many more besides. In Sect. 7 we come to the central results of the paper. They vindicate the earlier impression that linear pooling matches with squared Euclidean distance (if anything does), while geometric pooling matches with generalized Kullback-Leibler divergence (if anything does). Theorems 10 and 12 show that the only methods of fixing or fixing-and-aggregating-together that commute with LP are those based on SED, while the only methods that commute with GP are those based on GKL. And Theorems 11 and 13 describe the aggregation rules that result from minimising the weighted average distance from or to the credence functions to be aggregated. In Sect. 8, we move from two-cell partitions to many-cell partitions. Some of our results generalise fully—in particular, those concerning GP and GKL—while some generalise only to restricted versions—in particular, those concerning LP and SED. As mentioned above, in Sect. 9, we ask what happens when we consider disagreeing agents who assign credences to propositions that do not form a partition. Here, we meet a dilemma that GP and GKL face, but which LP and SED do not. Finally, by Sect. 10, we have all of our results in place and we can turn to their philosophical significance. I argue that these results can be used as philosophical booster rockets: on their own, they support no philosophical conclusion; but paired with an existing argument, either in favour of a way of aggregating or in favour of a particular measure of distance between credence functions, they can extend the conclusion of those arguments significantly. They say what measure of distance you should use if you wish to aggregate by LP or by GP, for instance; and they say what aggregation method you should use if you favour SED or GKL over other measures of distance. In Sect. 11, we conclude. The Appendix provides proofs for all of the results.

1 Aggregating credences

As advertised, we will restrict attention in these early sections to groups of agents like Amira and Benito, who assign credences only to the propositions in a partition $\mathcal{F} = \{X_1, X_2\}$. Let $C_{\mathcal{F}}$ be the set of credence functions over $\mathcal{F}$—that is, $C_{\mathcal{F}} = \{c : \mathcal{F} \to [0, 1]\}$. And let $P_{\mathcal{F}} \subseteq C_{\mathcal{F}}$ be the set of *coherent* credence functions over

$\mathcal{F}$—that is, $P_{\mathcal{F}} = \{c \in C_{\mathcal{F}} \mid c(X_1) + c(X_2) = 1\}$. Throughout, we take an agent's credence function to record her true credences. It doesn't record her reports of her credences, and it doesn't record the outcome of some particular method of measuring those credences. It records the credences themselves. Thus, we focus on cases in which our agent is genuinely incoherent, and not on cases in which she appears incoherent because of some flaw in our methods of measurement.

An aggregation method is a function $T : (C_{\mathcal{F}})^n \to C_{\mathcal{F}}$ that takes n credence functions—the agents—and returns a single credence function—the aggregate. Both aggregation methods we consider in this section appeal to a set of weights $\alpha_1, \ldots, \alpha_n$ for the agents, which we denote $\{\alpha\}$. We assume $\alpha_1, \ldots, \alpha_n \geq 0$ and $\sum_{k=1}^{n} \alpha_k = 1$.

First, *linear pooling*. This says that we obtain the aggregate credence for a particular proposition X_j in $\mathcal{F}$ by taking a weighted arithmetic average of the agents' credences in X_j; and we use the same weights for each proposition. The weighted arithmetic average of a sequence of numbers $r_1, \ldots, r_n$ given weights $\alpha_1, \ldots, \alpha_n$ is $\sum_{k=1}^{n} \alpha_k r_k = \alpha_1 r_1 + \cdots + \alpha_n r_n$. Thus:

Linear Pooling (LP) Let $\{\alpha\}$ be a set of weights. Then

$$\mathrm{LP}^{\{\alpha\}}(c_1, \cdots, c_n)(X_j) = \alpha_1 c_1(X_j) + \cdots + \alpha_n c_n(X_j) = \sum_{k=1}^{n} \alpha_k c_k(X_j)$$

for each X_j in $\mathcal{F}$.

Thus, to aggregate Amira's and Benito's credences in this way, we first pick a weight $0 \leq \alpha \leq 1$. Then the aggregate credence in X is $0.5\alpha + 0.2(1-\alpha)$, while the aggregate credence in $\overline{X}$ is $0.1\alpha + 0.6(1 - \alpha)$. Thus, if $\alpha = 0.4$, the aggregate credence in X is 0.32, while the aggregate credence in $\overline{X}$ is 0.4. (See Fig. 1 for an illustration of the effect of linear pooling on Amira's and Benito's credences.) Notice that, just as the two agents are incoherent, so is the aggregate. This is typically the case, though not universally, when we use linear pooling.

Second, we consider *geometric pooling*. This uses weighted *geometric* averages where linear pooling uses weighted *arithmetic* averages. The weighted geometric average of a sequence of numbers $r_1, \ldots, r_n$ given weights $\alpha_1, \ldots, \alpha_n$ is $\prod_{i=1}^{n} r_i^{\alpha_i} = r_1^{\alpha_1} \times \cdots \times r_n^{\alpha_n}$. Now, when all of the agents' credence functions are coherent, so is the credence function that results from taking weighted *arithmetic* averages of the credences they assign. That is, if $c_k(X_1) + c_k(X_2) = 1$ for all $1 \leq k \leq n$, then

$$\sum_{k=1}^{n} \alpha_k c_k(X_1) + \sum_{k=1}^{n} \alpha_k c_k(X_2) = \sum_{k=1}^{n} \alpha_k \left(c_k(X_1) + c_k(X_2) \right) = \sum_{k=1}^{n} \alpha_k = 1$$

However, the same is not true of weighted *geometric* averaging. Even if $c_k(X_1) + c_k(X_2) = 1$ for all $1 \leq k \leq n$, there is no guarantee that

$$\prod_{k=1}^{n} c_k(X_1)^{\alpha_k} + \prod_{k=1}^{n} c_k(X_2)^{\alpha_k} = 1$$

Thus, in geometric pooling, after taking the weighed geometric average, we need to normalize. So, for each cell X_j of our partition, we first take the weighted geometric
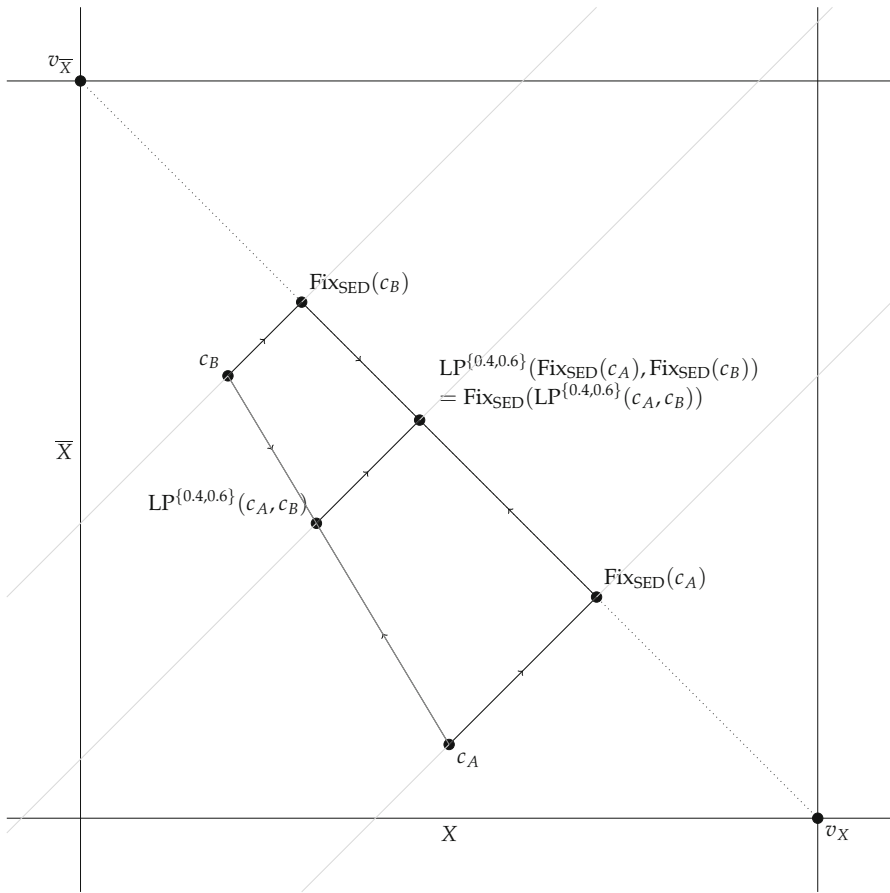
Fig. 1 *Linear pooling and* SED*-fixing applied to Amira's and Benito's credences.* If $\mathcal{F} = \{X, \overline{X}\}$, we can represent the set of all credence functions defined on X and $\overline{X}$ as the points in the unit square: we represent $c : \{X, \overline{X}\} \to [0, 1]$ as the point $(c(X), c(\overline{X}))$, so that the x-coordinate gives the credence in X, while the y-coordinate gives the credence in $\overline{X}$. In this way, we represent Amira's credence function as c_A and Benito's as c_B in the diagram above. And $\mathcal{P}_{\mathcal{F}}$, the set of coherent credence functions, is represented by the thick diagonal line joining the omniscient credence functions v_X and $v_{\overline{X}}$. As we can see, $\mathrm{Fix}_{\mathrm{SED}}(c_A)$ is the orthogonal projection of c_A onto this set of coherent credence functions; and similarly for $\mathrm{Fix}_{\mathrm{SED}}(c_B)$ and $\mathrm{Fix}_{\mathrm{SED}}(\mathrm{LP}^{\{0.4,0.6\}}(c_A, c_B))$. The straight line from c_A to c_B represents the set of linear pools of c_A and c_B generated by different weightings. The arrows indicated that you can reach the same point—$\mathrm{LP}^{\{0.4,0.6\}}(\mathrm{Fix}_{\mathrm{SED}}(c_A), \mathrm{Fix}_{\mathrm{SED}}(c_B)) = \mathrm{Fix}_{\mathrm{SED}}(\mathrm{LP}^{\{0.4,0.6\}}(c_A, c_B))$—from either direction. That is, LP and $\mathrm{Fix}_{\mathrm{SED}}$ commute

average of the agents' credences $c_k(X_j)$, and then we normalize the results. So the aggregated credence for X_j is

$$\frac{\prod_{k=1}^{n} c_k(X_j)^{\alpha_k}}{\prod_{k=1}^{n} c_k(X_1)^{\alpha_k} + \prod_{k=1}^{n} c_k(X_2)^{\alpha_k}}$$
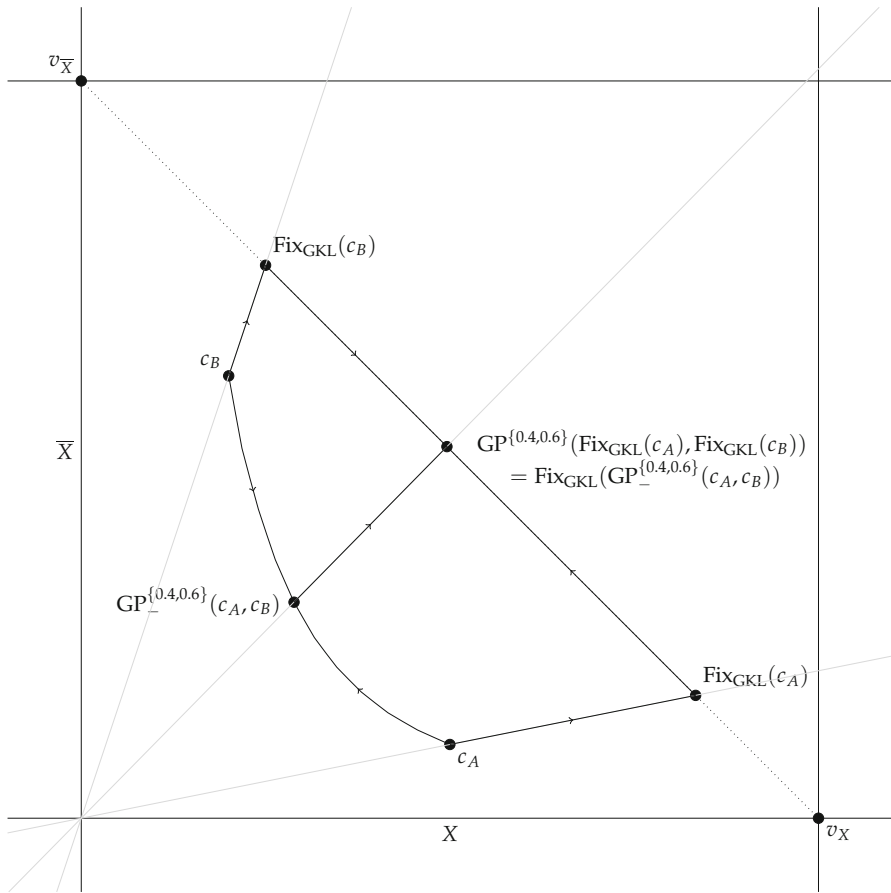
That is,

Fig. 2 Here, we see that $\mathrm{Fix}_{\mathrm{GKL}}(c_A)$ is the projection from the origin through c_A onto the set of coherent credence functions; and similarly for $\mathrm{Fix}_{\mathrm{GKL}}(c_B)$ and $\mathrm{Fix}_{\mathrm{GKL}}(\mathrm{GP}_-^{\{0.4,0.6\}}(c_A, c_B))$. The curved line from c_A to c_B represents the set of geometric pools of c_A and c_B generated by different weightings. Again, the arrows indicate that $\mathrm{GP} = \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{GP}_- = \mathrm{GP} \circ \mathrm{Fix}_{\mathrm{GKL}}$

Geometric Pooling (GP) Let $\{\alpha\}$ be a set of weights. Then

$$\mathrm{GP}^{\{\alpha\}}(c_1, \ldots, c_n)(X_j) = \frac{\prod_{k=1}^{n} c_k(X_j)^{\alpha_k}}{\sum_{X \in \mathcal{F}} \prod_{k=1}^{n} c_k(X)^{\alpha_k}}$$

for each X_j in $\mathcal{F}$.

Thus, to aggregate Amira's and Benito's credences in this way, we first pick a weight α. Then the aggregate credence in X is $\frac{0.5^\alpha 0.2^{1-\alpha}}{0.5^\alpha 0.2^{1-\alpha} + 0.1^\alpha 0.6^{1-\alpha}}$, while the aggregate credence in $\overline{X}$ is $\frac{0.1^\alpha + 0.6^{1-\alpha}}{0.5^\alpha 0.2^{1-\alpha} + 0.1^\alpha 0.6^{1-\alpha}}$. Thus, if $\alpha = 0.4$, the aggregate credence in X is 0.496, while the aggregate credence in $\overline{X}$ is 0.504. (Again, see Fig. 2 for an illustration.) Note that, this time, the aggregate is guaranteed to be coherent, even though the agents are incoherent.

2 Fixing incoherent credences

Amira has incoherent credences. How are we to fix her up so that she is coherent? And Benito? In general, how do we fix up an incoherent credence function so that it is coherent? A natural thought is that we should pick the credence function that is as similar as possible to the incoherent credence function whilst being coherent—we might think of this as a method of minimal mutilation.[3]

For this purpose, we need a measure of distance between credence functions. In fact, since the measures we will use do not have the properties that mathematicians usually require of distances—they aren't typically *metrics*—we will follow the statisticians in calling them *divergences* instead. A divergence is a function $\mathfrak{D} : C_{\mathcal{F}} \times C_{\mathcal{F}} \to [0, \infty]$ such that (i) $\mathfrak{D}(c, c) = 0$ for all c, and (ii) $\mathfrak{D}(c, c') > 0$ for all $c \neq c'$. We do not require that $\mathfrak{D}$ is symmetric: that is, we do not assume $\mathfrak{D}(c, c') = \mathfrak{D}(c', c)$ for all c, c'. Nor do we require that $\mathfrak{D}$ satisfies the triangle inequality: that is, we do not assume $\mathfrak{D}(c, c'') \leq \mathfrak{D}(c, c') + \mathfrak{D}(c', c'')$ for all c, c', c''.

Now, suppose $\mathfrak{D}$ is a divergence. Then the suggestion is this: given a credence function c, we fix it by taking the coherent credence function c^* such that $\mathfrak{D}(c^*, c)$ is minimal; or perhaps the coherent credence function c^* such that $\mathfrak{D}(c, c^*)$ is minimal. Since $\mathfrak{D}$ may not be symmetric, these two ways of fixing c might give different results. Thus:[4]

Fixing Given a credence function c, let

$$\mathrm{Fix}_{\mathfrak{D}_1}(c) = \underset{c' \in P_{\mathcal{F}}}{\arg \min} \, \mathfrak{D}(c', c)$$

and

$$\mathrm{Fix}_{\mathfrak{D}_2}(c) = \underset{c' \in P_{\mathcal{F}}}{\arg \min} \, \mathfrak{D}(c, c')$$

Throughout this paper, we will be concerned particularly with fixing incoherent credence functions using the so-called *additive Bregman divergences* (Bregman 1967). I'll introduce these properly in Sect. 6, but let's meet two of the most famous Bregman divergences now:

Squared Euclidean Distance (SED)

$$\mathrm{SED}(c, c') = \sum_{X \in \mathcal{F}} (c(X) - c'(X))^2$$

This is the divergence used in the least squares method in data fitting, where we wish to measure how far a putative fit to the data, c, lies from the data itself c'. For arguments in its favour, see Selten (1998), Leitgeb and Pettigrew (2010a), D'Agostino and Sinigaglia (2010) and Pettigrew (2016a).

---

[3] Such a fixing procedure is at least suggested by the second central result of De Bona and Staffel (2017, 204). We will meet the principle of minimal mutilation again in Sect. 10.

[4] Recall: $P_{\mathcal{F}}$ is the set of coherent credence functions over $\mathcal{F}$.

Generalized Kullback-Leibler (GKL)

$$\text{GKL}(c, c') = \sum_{X \in \mathcal{F}} \left(c(X) \log \frac{c(X)}{c'(X)} - c(X) + c'(X) \right)$$

This is most famously used in information theory to measure the information gained by moving from a prior distribution, c', to a posterior, c. For arguments in its favour, see Paris and Vencovská (1990), Paris and Vencovská (1997) and Levinstein (2012).

Let's see the effect of these on Amira's and Benito's credences. SED is symmetric—that is, $\text{SED}(c, c') = \text{SED}(c', c)$, for all c, c'.[5] Therefore, both fixing methods agree—that is, $\text{Fix}_{\text{SED}_1} = \text{Fix}_{\text{SED}_2}$. GKL isn't symmetric. However, its fixing methods nonetheless always agree for credences defined on a two-cell partition—that is, as we will see below, we also have $\text{Fix}_{\text{GKL}_1} = \text{Fix}_{\text{GKL}_2}$.

	X	$\overline{X}$
Amira (original)	0.5	0.1
Benito (original)	0.2	0.6
Amira (SED-fixed)	0.7	0.3
Benito (SED-fixed)	0.3	0.7
Amira (GKL-fixed)	0.83	0.17
Benito (GKL-fixed)	0.25	0.75

In general:

Proposition 1 *Suppose $\mathcal{F} = \{X_1, X_2\}$ is a partition. Then, for all c in $C_{\mathcal{F}}$ and X_j in $\mathcal{F}$,*

(i) $\text{Fix}_{\text{SED}_1}(c)(X_j) = \text{Fix}_{\text{SED}_2}(c)(X_j) = c(X_j) + \frac{1-(c(X_1)+c(X_2))}{2}$

(ii) $\text{Fix}_{\text{GKL}_1}(c)(X_j) = \text{Fix}_{\text{GKL}_2}(c)(X_j) = \frac{c(X_j)}{c(X_1)+c(X_2)}$

In other words, when we use SED to fix an incoherent credence function c over a partition X_1, X_2, we add the same quantity to each credence. That is, there is K such that $\text{Fix}_{\text{SED}}(c)(X_j) = c(X_j) + K$, for $j = 1, 2$. Thus, the difference between a fixed credence and the original credence is always the same—it is K. In order to ensure that the result is coherent, this quantity must be $K = \frac{1-(c(X_1)+c(X_2))}{2}$. On the other hand, when we use GKL to fix c, we multiply each credence by the same quantity. That is, there is K such that $\text{Fix}_{\text{GKL}}(c)(X_j) = K \cdot c(X_j)$, for $j = 1, 2$. Thus, the ratio between a fixed credence and the original credence is always the same—it is K. In order to ensure that the result is coherent in this case, this quantity must be $K = \frac{1}{c(X_1)+c(X_2)}$.

There is also a geometric way to understand the relationship between fixing using SED and fixing using GKL. Roughly: $\text{Fix}_{\text{SED}}(c)$ is the orthogonal projection of c onto the set of coherent credence functions, while $\text{Fix}_{\text{GKL}}(c)$ is the result of projecting from the origin through c onto the set of coherent credence functions. This is illustrated in Figs. 1 and 2. One consequence is this: if $c(X) + c(\overline{X}) < 1$, then fixing using SED is more conservative than fixing by GKL, in the sense that the resulting credence function

---

[5] Indeed, SED is the only symmetric Bregman divergence.

is less opinionated—it has a lower maximum credence. But if $c(X) + c(\overline{X}) > 1$, then fixing using GKL is more conservative.

3 Aggregate-then-fix versus fix-then-aggregate

Using the formulae in Proposition 1, we can explore what differences, if any, there are between fixing incoherent agents and then aggregating them, on the one hand, and aggregating incoherent agents and then fixing the aggregate, on the other. Suppose $c_1, \ldots, c_n$ are the credence functions of a group of agents, all defined on the same two-cell partition $\{X_1, X_2\}$. Some may be incoherent, and we wish to aggregate them. Thus, we might first fix each credence function, and then aggregate the resulting coherent credence functions; or we might aggregate the original credence functions, and then fix the resulting aggregate. When we aggregate, we have two methods at our disposal—linear pooling (LP) and geometric pooling (GP); and when we fix, we have two methods at our disposal—one based on squared Euclidean distance (SED) and the other based on generalized Kullback-Leibler divergence (GKL). Our next result tells us how these different options interact. To state it, we borrow a little notation from the theory of function composition. For instance, we write $\mathrm{LP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{SED}}$ to denote the function that takes a collection of agents' credence functions $c_1, \ldots, c_n$ and returns $\mathrm{LP}^{\{\alpha\}}(\mathrm{Fix}_{\mathrm{SED}}(c_1), \ldots, \mathrm{Fix}_{\mathrm{SED}}(c_n))$. So $\mathrm{LP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{SED}}$ might be read: $\mathrm{LP}^{\{\alpha\}}$ *following* $\mathrm{Fix}_{\mathrm{SED}}$, or $\mathrm{LP}^{\{\alpha\}}$ *acting on the results of* $\mathrm{Fix}_{\mathrm{SED}}$. Similarly, $\mathrm{Fix}_{\mathrm{SED}} \circ \mathrm{LP}^{\{\alpha\}}$ denotes the function that takes $c_1, \ldots, c_n$ and returns $\mathrm{Fix}_{\mathrm{SED}}(\mathrm{LP}^{\{\alpha\}}(c_1, \ldots, c_n))$. And we say that two functions are equal if they agree on all arguments, and unequal if they disagree on some.

Proposition 2 *Suppose $\mathcal{F} = \{X_1, X_2\}$ is a partition. Then*

(i) $\mathrm{LP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{SED}} = \mathrm{Fix}_{\mathrm{SED}} \circ \mathrm{LP}^{\{\alpha\}}$.

That is, linear pooling commutes with SED-*fixing.*

That is, for all $c_1, \ldots c_n$ in $C_{\mathcal{F}}$,

$$\mathrm{LP}^{\{\alpha\}}(\mathrm{Fix}_{\mathrm{SED}}(c_1), \ldots, \mathrm{Fix}_{\mathrm{SED}}(c_n)) = \mathrm{Fix}_{\mathrm{SED}}(\mathrm{LP}^{\{\alpha\}}(c_1, \ldots, c_n))$$

(ii) $\mathrm{LP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} \neq \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{LP}^{\{\alpha\}}$.

That is, linear pooling does not commute with GKL-*fixing.*

That is, for some $c_1, \ldots, c_n$ in $C_{\mathcal{F}}$,

$$\mathrm{LP}^{\{\alpha\}}(\mathrm{Fix}_{\mathrm{GKL}}(c_1), \ldots, \mathrm{Fix}_{\mathrm{GKL}}(c_n)) \neq \mathrm{Fix}_{\mathrm{GKL}}(\mathrm{LP}^{\{\alpha\}}(c_1, \ldots, c_n))$$

(iii) $\mathrm{GP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} = \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{GP}^{\{\alpha\}}$.

That is, geometric pooling commutes with GKL-*fixing.*

That is, for all $c_1, \ldots, c_n$,

$$\mathrm{GP}^{\{\alpha\}}(\mathrm{Fix}_{\mathrm{GKL}}(c_1), \ldots, \mathrm{Fix}_{\mathrm{GKL}}(c_n)) = \mathrm{Fix}_{\mathrm{GKL}}(\mathrm{GP}^{\{\alpha\}}(c_1, \ldots, c_n))$$

(iv) $\mathrm{GP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{SED}} \neq \mathrm{Fix}_{\mathrm{SED}} \circ \mathrm{GP}^{\{\alpha\}}$.

That is, geometric pooling does not commute with SED*-fixing.*

That is, for some $c_1, \ldots, c_n$,

$$\mathrm{GP}^{\{\alpha\}}(\mathrm{Fix}_{\mathrm{SED}}(c_1), \ldots, \mathrm{Fix}_{\mathrm{SED}}(c_n)) \neq \mathrm{Fix}_{\mathrm{SED}}(\mathrm{GP}^{\{\alpha\}}(c_1, \ldots, c_n))$$

With this result, we start to see the main theme of this paper emerging: SED naturally accompanies linear pooling, while GKL naturally accompanies geometric pooling. In Sect. 7, we'll present further results that support that conclusion, as well as some that complicate it a little. In Sects. 8 and 9, these are complicated further. But the lesson still roughly holds.

4 Aggregating by minimizing distance

In the previous section, we introduced the notion of a divergence and we put it to use fixing incoherent credence functions: given an incoherent credence function c, we fix it by taking the coherent credence function that minimizes divergence *to* or *from* c. But divergences can also be used to aggregate credence functions.[6] The idea is this: given a divergence and a collection of credence functions, take the aggregate to be the credence function that minimizes the weighted arithmetic average of the divergences *to* or *from* those credence functions. Thus:

$\mathfrak{D}$**-aggregation** Let $\{\alpha\}$ be a set of weights. Then

$$\mathrm{Agg}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in C_{\mathcal{F}}}{\arg\min} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c', c_k)$$

and

$$\mathrm{Agg}_{\mathfrak{D}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in C_{\mathcal{F}}}{\arg\min} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c_k, c')$$

Let's see what these give when applied to the two divergences we introduced above, namely, SED and GKL.

---

[6] When the agents are represented as having categorical doxastic states, such as full beliefs or commitments, this method was studied first in the computer science literature on *belief merging* (Konieczny and Pino-Pérez 1999; Konieczny and Grégoire 2006). It was studied first in the judgment aggregation literature by Pigozzi (2006).

Proposition 3 *Let $\{\alpha\}$ be a set of weights. Then, for each X_j in $\mathcal{F}$,*

(i) $$\text{Agg}_{\text{SED}}^{\{\alpha\}}(c_1, \ldots, c_n)(X_j) = \sum_{k=1}^{n} \alpha_k c_k(X_j) = \text{LP}^{\{\alpha\}}(c_1, \ldots, c_n)$$

(ii) $$\text{Agg}_{\text{GKL}_1}^{\{\alpha\}}(c_1, \ldots, c_n)(X_j) = \prod_{k=1}^{n} c_k(X_j)^{\alpha_k} = \text{GP}_{-}^{\{\alpha\}}(c_1, \ldots, c_n)$$

(iii) $$\text{Agg}_{\text{GKL}_2}^{\{\alpha\}}(c_1, \ldots, c_n)(X_j) = \sum_{k=1}^{n} \alpha_k c_k(X_j) = \text{LP}^{\{\alpha\}}(c_1, \ldots, c_n)$$

Thus, Agg_{SED} and $\text{Agg}_{\text{GKL}_2}$ are just linear pooling—they assign to each X_j the (unnormalized) weighted arithmetic average of the credences assigned to X_j by the agents. On the other hand, $\text{Agg}_{\text{GKL}_1}$ is just geometric pooling without the normalization procedure—it assigns to each X_j the (unnormalized) weighted geometric average of the credences assigned to X_j by the agents. I call this aggregation procedure GP_{-}. Given a set of coherent credence functions, GP returns a coherent credence function, but GP_{-} typically won't. However, if we aggregate using GP_{-} and then fix using GKL, then we obtain GP:

Proposition 4 *Let $\{\alpha\}$ be a set of weights. Then*

$$\text{GP}^{\{\alpha\}} = \text{Fix}_{\text{GKL}} \circ \text{GP}_{-}^{\{\alpha\}}$$

5 Aggregate and fix together

In this section, we meet our final procedure for producing a single coherent credence function from a collection of possibly incoherent ones. This procedure fixes and aggregates together: that is, it is a one-step process, unlike the two-step processes we have considered so far. It generalises a technique suggested by Osherson and Vardi (2006) and explored further by Predd et al. (2008).[7] Again, it appeals to a divergence $\mathfrak{D}$; and thus again, there are two versions, depending on whether we measure distance *from* coherence or distance *to* coherence. If we measure distance from coherence, the *weighted coherent approximation principle* tells us to pick the coherent credence function such that the weighted arithmetic average of the divergences from that coherent credence function to the agents is minimized. And if we measure distance to coherence, it picks the coherent credence function that minimizes the weighted arithmetic average of the divergences from the agents to the credence function. Thus, it poses a minimization problem similar to that posed by $\mathfrak{D}$-aggregation, but in this case, we

---

[7] Osherson and Vardi (2006) and Predd et al. (2008) consider only what we call $\text{WCAP}_{\text{SED}_1}^{\{\frac{1}{n}\}}$ below. They do not consider the different Bregman divergences $\mathfrak{D}$; they do not consider the two directions; and they do not consider the possibility of weighting the distances differently. This is quite understandable—their interest lies mainly in the feasibility of the procedure from a computational point of view. We will not address this issue here.

wish to find the credence function *amongst the coherent ones* that does the minimizing; in the case of $\mathfrak{D}$-aggregation, we wish to find the credence function *amongst all the credence ones* that does the minimizing.

Weighted Coherent Approximation Principle Let $\{\alpha\}$ be a set of weights. Then

$$\mathrm{WCAP}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in P_{\mathcal{F}}}{\arg\min} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c', c_k)$$

and

$$\mathrm{WCAP}_{\mathfrak{D}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in P_{\mathcal{F}}}{\arg\min} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c_k, c')$$

How does this procedure compare to the fix-then-aggregate and aggregate-then-fix procedures that we considered above? Our next result gives the answer:

Proposition 5 *Suppose* $\mathcal{F} = \{X_1, X_2\}$ *is a partition. Let* $\{\alpha\}$ *be a set of weights. Then*

(i)
$$\begin{aligned} \mathrm{WCAP}_{\mathrm{SED}}^{\{\alpha\}} &= \mathrm{LP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{SED}} &&= \mathrm{Fix}_{\mathrm{SED}} \circ \mathrm{LP}^{\{\alpha\}} \\ &= \mathrm{Agg}_{\mathrm{SED}}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{SED}} &&= \mathrm{Fix}_{\mathrm{SED}} \circ \mathrm{Agg}_{\mathrm{SED}}^{\{\alpha\}} \end{aligned}$$

(ii)
$$\begin{aligned} \mathrm{WCAP}_{\mathrm{GKL}_1}^{\{\alpha\}} &= \mathrm{GP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} &&= \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{GP}^{\{\alpha\}} \\ &= \mathrm{Agg}_{\mathrm{GKL}_1}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} &&= \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{Agg}_{\mathrm{GKL}_1}^{\{\alpha\}} \\ &= \mathrm{GP}^{\{\alpha\}} \end{aligned}$$

(iii)
$$\begin{aligned} \mathrm{WCAP}_{\mathrm{GKL}_2}^{\{\alpha\}} &\neq \mathrm{GP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} &&= \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{GP}^{\{\alpha\}} \\ &= \mathrm{GP}^{\{\alpha\}} \end{aligned}$$

(iv)
$$\begin{aligned} \mathrm{WCAP}_{\mathrm{GKL}_2}^{\{\alpha\}} &= \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{LP} &&= \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{Agg}_{\mathrm{GKL}_2}^{\{\alpha\}} \\ &\neq \mathrm{Agg}_{\mathrm{GKL}_2}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} \end{aligned}$$

(i) and (ii) confirm our picture that linear pooling naturally pairs with SED, while geometric pooling pairs naturally with GKL. However, (iii) and (iv) complicate this. This is a pattern we will continue to encounter as we progress: when we minimize distance *from* coherence, the aggregation methods and divergence measures pair up reasonably neatly; when we minimize distance *to* coherence, they do not.

These, then, are the various ways we will consider by which you might produce a single coherent credence function when given a collection of possibly incoherent ones: fix-then-aggregate, aggregate-then-fix, and the weighted coherent approximation principle. Each involves minimizing a divergence at some point, and so each comes in two varieties, one based on minimizing distance *from* coherence, the other based on minimizing distance *to* coherence.

Are these the only possible ways? There is one other that might seem a natural cousin of WCAP, and one that might seem a natural cousin of $\mathfrak{D}$-aggregation, which

we might combine with fixing in either of the ways considered above. In WCAP, we pick the coherent credence function that minimizes the weighted *arithmetic* average of the distances from (or to) the agents. The use of the weighted arithmetic average here might lead you to expect that WCAP will pair most naturally with linear pooling, which aggregates by taking the weighted arithmetic average of the agents' credences. You might expect it to interact poorly with geometric pooling, which aggregates by taking the weighted geometric average of the agents' credences (and then normalizing). But, in fact, as we saw in Theorem 5(ii), when coupled with the divergence GKL, and when we minimize distance from coherence, rather than distance to coherence, WCAP entails geometric pooling. Nonetheless, we might think that if it is natural to minimize the weighted *arithmetic* average of distances from coherence, and if both linear and geometric pooling are on the table, revealing that we have no prejudice against using geometric averages to aggregate numerical values, then it is equally natural to minimize the weighted *geometric* average of distances from coherence. This gives:

Weighted Geometric Coherent Approximation Principle Let $\{\alpha\}$ be a set of weights. Then

$$\text{WGCAP}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in P_{\mathcal{F}}}{\arg\min} \prod_{k=1}^{n} \mathfrak{D}(c', c_k)^{\alpha_k}$$

and

$$\text{WGCAP}_{\mathfrak{D}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in P_{\mathcal{F}}}{\arg\min} \prod_{k=1}^{n} \mathfrak{D}(c_k, c')^{\alpha_k}$$

However, it is easy to see that:

Proposition 6 *For any divergence $\mathfrak{D}$, any set of weights $\{\alpha\}$, any $i \in \{1, 2\}$, and any coherent credence functions $c_1, \ldots, c_n$,*

$$c^* = \text{WGCAP}_{\mathfrak{D}_i}^{\{\alpha\}}(c_1, \ldots, c_n) \quad \text{iff} \quad c^* = c_1 \text{ or } \ldots \text{ or } c_n$$

That is, WGCAP gives a dictatorship rule when applied to coherent agents: it aggregates a group of agents by picking one of those agents and making her stand for the whole group. This rules it out immediately as a method of aggregation.

Similarly, we might define a geometric cousin to $\mathfrak{D}$-aggregation:

Geometric $\mathfrak{D}$-aggregation Let $\{\alpha\}$ be a set of weights. Then

$$\text{GAgg}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in C_{\mathcal{F}}}{\arg\min} \prod_{k=1}^{n} \mathfrak{D}(c', c_k)^{\alpha_k}$$

and

$$\text{GAgg}_{\mathfrak{D}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min_{c' \in C_{\mathcal{F}}} \prod_{k=1}^{n} \mathfrak{D}(c_k, c')^{\alpha_k}$$

However, we obtain a similar result to before, though this time the dictatorship arises for *any* set of agents, not just coherent ones.

Proposition 7 *For any divergence $\mathfrak{D}$, any set of weights $\{\alpha\}$, any $i \in \{1, 2\}$, and any credence functions $c_1, \ldots, c_n$,*

$$c^* = \text{GAgg}_{\mathfrak{D}_i}^{\{\alpha\}}(c_1, \ldots, c_n) \quad \textit{iff} \quad c^* = c_1 \textit{ or } \ldots \textit{ or } c_n$$

Thus, in what follows, we will consider only fix-then-aggregate, aggregate-then-fix, and WCAP.

6 Bregman divergences

In the previous section, we stated our definition of *fixing* and our definition of the *weighted coherent approximation principle* in terms of a divergence $\mathfrak{D}$. We then identified two such divergences, SED and GKL, and we explored how those ways of making incoherent credences coherent related to ways of combining different credence functions to give a single one. This leaves us with two further questions: Which other divergences might we use when we are fixing incoherent credences? And how do the resulting ways of fixing relate to our aggregation principles? In this section, we introduce a large family of divergences known as the *additive Bregman divergences* (Bregman 1967). SED and GKL are both additive Bregman divergences, and indeed Bregman introduced the notion as a generalisation of SED. They are widely used in statistics to measure how far one probability distribution lies from another (Csiszár 1991; Banerjee et al. 2005; Gneiting and Raftery 2007; Csiszár 2008; Predd et al. 2009); they are used in social choice theory to measure how far one distribution of wealth lies from another (D'Agostino and Dardanoni 2009; Magdalou and Nock 2011); and they are used in the epistemology of credences to define measures of the inaccuracy of credence functions (Pettigrew 2016a). Below, I will offer some reasons why we should use them in our procedures for fixing incoherent credences. But first let's define them.

Each additive Bregman divergence $\mathfrak{D} : C_{\mathcal{F}} \times C_{\mathcal{F}} \to [0, \infty]$ is generated by a function $\varphi : [0, 1] \to \mathbb{R}$, which is required to be (i) strictly convex on $[0, 1]$ and (ii) twice differentiable on $(0, 1)$ with a continuous second derivative. We begin by using φ to define the divergence from x to y, where $0 \leq x, y \leq 1$. We first draw the tangent to φ at y. Then we take the divergence from x to y to be the difference between the value of φ at x—that is, $\varphi(x)$—and the value of that tangent at x—that is, $\varphi(y) + \varphi'(y)(x - y)$. Thus, the divergence from x to y is $\varphi(x) - \varphi(y) - \varphi'(y)(x - y)$. We then take the divergence from one credence function c to another c' to be the sum of the divergences from each credence assigned by c to the corresponding credence assigned by c'. Thus:

Definition 1 Suppose $\varphi : [0, 1] \to \mathbb{R}$ is a strictly convex function that is twice differentiable on $(0, 1)$ with a continuous second derivative. And suppose $\mathfrak{D} : C_{\mathcal{F}} \times C_{\mathcal{F}} \to [0, \infty]$. Then $\mathfrak{D}$ *is the additive Bregman divergence generated by* φ if, for any c, c' in $C_{\mathcal{F}}$,

$$\mathfrak{D}(c, c') = \sum_{X \in \mathcal{F}} \varphi(c(X)) - \varphi(c'(X)) - \varphi'(c'(X))(c(X) - c'(X))$$

And we can show:

Proposition 8 (i) SED *is the additive Bregman divergence generated by* $\varphi(x) = x^2$. (ii) GKL *is the additive Bregman divergence generated by* $\varphi(x) = x \log x - x$.

Why do we restrict our attention to additive Bregman divergences when we are considering which divergences to use to fix incoherent credences? Here's one answer.[8] Just as beliefs can be true or false, credences can be more or less accurate. A credence in a true proposition is more accurate the higher it is, while a credence in a false proposition is more accurate the lower it is. Now, just as some philosophers think that beliefs are more valuable if they are true than if they are false (Goldman 2002), so some philosophers think that credences are more valuable the more accurate they are (Joyce 1998) and (Pettigrew 2016a). This approach is sometimes called *accuracy-first epistemology*. These philosophers then provide mathematically precise ways to measure the *in*accuracy of credence functions. They say that a credence function c is more inaccurate at a possible world w the further c lies from the omniscient credence function v_w at w, where v_w assigns maximal credence (i.e. 1) to all truths at w and minimal credence (i.e. 0) to all falsehoods at w. So, in order to measure the inaccuracy of c at w we need a measure of how far one credence function lies from another, just as we do when we want to fix incoherent credence functions. But which divergences are legitimate measures for this purpose? Elsewhere, I have argued that it is only the additive Bregman divergences (Pettigrew 2016a, Chapter 4).[9] I won't rehearse the argument here, but I will accept the conclusion.

Now, on its own, my argument that only the additive Bregman divergences are legitimate for the purpose of measuring inaccuracy does not entail that only the additive Bregman divergences are legitimate for the purpose of correcting incoherent credences. But the following argument gives us reason to take that further step as well. One of the appealing features of the so-called accuracy-first approach to the epistemology of credences is that it gives a neat and compelling argument for the credal norm of *probabilism*, which says that an agent should have a coherent credence function (Joyce 1998; Pettigrew 2016a). Having justified the restriction to the additive Bregman divergences on other grounds, the accuracy-first argument for probabilism is based on the following mathematical fact:

---

[8] See Bona and Staffel (2017) for a similar line of argument.

[9] If we use Bregman divergences to measure the distance from the omniscient credence function to another credence function, the resulting measure of inaccuracy is a *strictly proper scoring rule*. These measures of inaccuracy have been justified independently in the accuracy-first literature (Oddie 1997; Gibbard 2008; Joyce 2009). And conversely, given a strictly proper scoring rule, we can easily recover an additive Bregman divergence.

Theorem 9 (Predd et al. 2009) *Suppose* $\mathcal{F} = \{X_1, \ldots, X_m\}$ *is a partition, and* $\mathfrak{D} :$ $C_{\mathcal{F}} \times C_{\mathcal{F}} \to [0, \infty]$ *is an additive Bregman divergence. And suppose c is an incoherent credence function. Then, if* $c^* = \arg\min_{c' \in P_{\mathcal{F}}} \mathfrak{D}(c', c)$, *then* $\mathfrak{D}(v_i, c^*) < \mathfrak{D}(v_i, c)$ *for all* $1 \leq i \leq m$, *where* $v_i(X_j) = 1$ *if* $i = j$ *and* $v_i(X_j) = 0$ *if* $i \neq j$.

That is, if c is incoherent, then the closest coherent credence function to c is closer to all the possible omniscient credence functions than c is, and thus is more accurate than c is at all possible worlds. Thus, if we fix up incoherent credence functions by using an additive Bregman divergence and taking the nearest coherent credence function, then we have an explanation for why we proceed in this way, namely, that doing so is guaranteed to increase the accuracy of the credence function. To see this in action, consider $\text{Fix}_{\text{SED}}(c_A)$ and $\text{Fix}_{\text{SED}}(c_B)$ in Fig. 1. It is clear from this picture that $\text{Fix}_{\text{SED}}(c_A)$ is closer to v_X than c_A is, and closer to $v_{\overline{X}}$ than c_A is.

7 When do divergences cooperate with aggregation methods?

7.1 Minimizing distance from coherence

From Proposition 5(i) and (ii), we learned of an additive Bregman divergence that fixes up incoherent credences in a way that cooperates with linear pooling—it is SED. And we learned of an additive Bregman divergence that fixes up incoherent credences in a way that cooperates with geometric pooling, at least when you fix by minimizing distance *from* coherence rather than distance *to* coherence—it is GKL. But this leaves open whether there are other additive Bregman divergences that cooperate with either of these rules. The following theorem shows that there are not.

Theorem 10 *Suppose* $\mathcal{F} = \{X_1, X_2\}$ *is a partition. And suppose* $\mathfrak{D}$ *is an additive Bregman divergence. Then:*

(i) $\text{WCAP}_{\mathfrak{D}_1} = \text{Fix}_{\mathfrak{D}_1} \circ \text{LP} = \text{LP} \circ \text{Fix}_{\mathfrak{D}_1}$ *iff* $\mathfrak{D}$ *is a positive linear transformation of* SED.

(ii) $\text{WCAP}_{\mathfrak{D}_1} = \text{Fix}_{\mathfrak{D}_1} \circ \text{GP} = \text{GP} \circ \text{Fix}_{\mathfrak{D}_1}$ *iff* $\mathfrak{D}$ *is a positive linear transformation of* GKL.

Thus, suppose you fix incoherent credences by minimizing distance *from* coherence. And suppose you wish to fix and aggregate in ways that cooperate with one another—we will consider an argument for doing this in Sect. 10. Then, if you measure the divergence between credence functions using SED, then Proposition 5 says you should aggregate by linear pooling. If, on the other hand, you wish to use GKL, then you should aggregate by geometric pooling. And, conversely, if you aggregate credences by linear pooling, then Theorem 10 says you should fix incoherent credences using SED. If, on the other hand, you aggregate by geometric pooling, then you should fix incoherent credences using GKL. In Sect. 10, we will ask whether we have reason to fix and aggregate in ways that cooperate with one another.

We round off this section with a result that is unsurprising in the light of previous results:

Theorem 11 *Suppose* $\mathfrak{D}$ *is an additive Bregman divergence. Then,*

(i) $\text{Agg}_{\mathfrak{D}_1} = \text{LP}$ *iff* $\mathfrak{D}$ *is a positive linear transformation of* SED.
(ii) $\text{Agg}_{\mathfrak{D}_1} = \text{GP}_-$ *iff* $\mathfrak{D}$ *is a positive linear transformation of* GKL.

7.2 Minimizing distance to coherence

Next, let us consider what happens when we fix incoherent credences by minimizing distance *to* coherence rather than distance *from* coherence.

Theorem 12 *Suppose* $\mathfrak{D}$ *is an additive Bregman divergence generated by* φ. *Then,*

(i) $\text{WCAP}_{\mathfrak{D}_2} = \text{Fix}_{\mathfrak{D}_2} \circ \text{LP}$.
(ii) $\text{WCAP}_{\mathfrak{D}_2} = \text{Fix}_{\mathfrak{D}_2} \circ \text{LP} = \text{LP} \circ \text{Fix}_{\mathfrak{D}_2}$, *when the methods are applied to coherent credences.*
(iii) $\text{WCAP}_{\mathfrak{D}_2} = \text{Fix}_{\mathfrak{D}_2} \circ \text{LP} = \text{LP} \circ \text{Fix}_{\mathfrak{D}_2}$, *if* $\varphi''(x) = \varphi''(1-x)$, *for* $0 \le x \le 1$.

Theorem 12(iii) corresponds to Theorem 10(i), but in this case we see that a much wider range of Bregman divergences give rise to fixing methods that cooperate with linear pooling when we measure distance *to* coherence. Theorem 12(i) and (ii) entail that there is no analogue to Theorem 10(ii). There is no additive Bregman divergence that cooperates with geometric pooling when we fix by minimizing distance *to* coherence. That is, there is no additive Bregman divergence $\mathfrak{D}$ such that $\text{WCAP}_{\mathfrak{D}_2} = \text{Fix}_{\mathfrak{D}_2} \circ \text{GP} = \text{GP} \circ \text{Fix}_{\mathfrak{D}_2}$. This result complicates our thesis from above that SED pairs naturally with linear pooling while GKL pairs naturally with geometric pooling.

We round off this section with the analogue of Theorem 11:

Theorem 13 *Suppose* $\mathfrak{D}$ *is an additive Bregman divergence. Then,* $\text{Agg}_{\mathfrak{D}_2} = \text{LP}$.

8 Partitions of any size

As we have seen, there are three natural ways in which we might aggregate the credences of disagreeing agents when some are incoherent: we can fix-then-aggregate, aggregate-then-fix, or fix-and-aggregate-together. In the preceding sections, we have seen, in a restricted case, when these three methods agree for two standard methods of pooling and two natural methods of fixing. In this restricted case, where the agents to fixed or aggregated have credences only over a two-cell partition, both methods of pooling seem viable, as do both methods of fixing—the key is to pair them carefully. In this section, we look beyond our restricted case. Instead of considering only agents with credences in two propositions that partition the space of possibilities, we consider agents with credences over partitions of any (finite) size. As we will see, in this context, geometric pooling and GKL continue to cooperate fully, but linear pooling and SED do not. This looks like a strike against linear pooling and SED, but we should not write them off so quickly, for in Sect. 9, we will consider agents with credences in propositions that do not form a partition, and there we will see that geometric pooling and GKL face a dilemma that linear pooling and SED avoid. So the scorecard evens out.

Suppose, then, that $\mathcal{F} = \{X_1, \ldots, X_m\}$ is a partition, and $c_1, \ldots, c_n$ are credence functions over $\mathcal{F}$. We'll look at geometric pooling and GKL first, since there are no surprises there.

First, Proposition 1(ii) generalizes in the natural way: to fix a credence function over any partition using GKL, you simply normalise it in the usual way—see Proposition 15 below. Propositions 2(ii–iv) also generalise, as do Propositions 3(ii-iii), 5(ii-iv), 6, and 7, as well as Theorems 10(ii) and 11(ii). Thus, as for agents with credences over two-cell partitions, GKL fully cooperates with geometric pooling for agents with credences over many-cell partitions.

Things are rather different, however, for linear pooling and SED. The initial problem is that Proposition 1(i) does not generalise in the natural way. Suppose c is an incoherent credence function over the partition $\mathcal{F} = \{X_1, \ldots, X_m\}$. We wish to fix c by taking the credence function that minimizes distance from it when we measure distance using SED. We might expect that, as before, there is some constant K such that we fix c by adding K to each of the credences that c assigns—that is, we might expect that the fixed credence in X_j will be $c(X_j) + K$, for all $1 \leq j \leq m$. The problem with this is that, sometimes, there is no K such that the resulting function is a coherent credence function—there is sometimes no K such that (i) $\sum_{i=1}^{m} c(X_j) + K = 1$, and (ii) $c(X_j) + K \geq 0$, for all $1 \leq j \leq m$. Indeed, $\sum_{i=1}^{m} c(X_j) + K = 1$ holds iff $K = \frac{1 - \sum_{i=1}^{m} c(X_i)}{m}$, and often there is X_j such that $c(X_j) + \frac{1 - \sum_{i=1}^{m} c(X_i)}{m} < 0$. Consider, for instance, the following credence function over $\{X_1, X_2, X_3\}$: $c(X_1) = 0.9$, $c(X_2) = 0.9$, and $c(X_3) = 0.1$. Then $c(X_3) + \frac{1 - (c(X_1) + c(X_2) + c(X_3))}{3} = -0.3$.

So, if this is not what happens when we fix an incoherent credence function over a many-cell partition using SED, what does happen? In fact, there *is* some constant that we add to the original credences to obtain the fixed credences. But we don't necessarily add that constant to *each* of the original credences. Sometimes, we fix some of the original credences by setting them to 0, while we fix the others by adding the constant. The following fact is crucial:

Proposition 14 *Suppose $0 \leq r_1, \ldots, r_m \leq 1$. Then there is a unique K such that*

$$\sum_{r_i : r_i + K \geq 0} r_i + K = 1$$

With this in hand, we are now ready to state the true generalization of Proposition 1:

Proposition 15 *Suppose c is a credence function over a partition $\mathcal{F} = \{X_1, \ldots, X_m\}$. Then*

(i) *For all $1 \leq j \leq m$,*

$$\text{Fix}_{\text{SED}_1}(c)(X_j) = \text{Fix}_{\text{SED}_2}(c)(X_j) = \begin{cases} c(X_j) + K & \text{if } c(X_j) + K \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

where K is the unique number such that

$$\sum_{i : c(X_i) + K \geq 0} c(X_i) + K = 1$$

(ii) *For all $1 \leq j \leq m$,*

$$\mathrm{Fix}_{\mathrm{GKL}_1}(c)(X_j) = \mathrm{Fix}_{\mathrm{GKL}_2}(c)(X_j) = \frac{c(X_j)}{\sum_{i=1}^{m} c(X_i)}$$

Having seen the effects of $\mathrm{Fix}_{\mathrm{SED}}$, we can now see why Proposition 2(i) does not generalise to the many-cell partition case. The following table provides the credences of two agents over a partition $\{X_1, X_2, X_3\}$. Both are incoherent. As we can see, fixing using SED and then linear pooling gives quite different results from linear pooling and then fixing using SED.

	X_1	X_2	X_3
c_1	0.9	0.9	0.1
c_2	0.9	0.1	0.9
$\mathrm{Fix}_{\mathrm{SED}}(c_1)$	0.5	0.5	0
$\mathrm{Fix}_{\mathrm{SED}}(c_2)$	0.5	0	0.5
$\mathrm{LP}^{\frac{1}{2}}(\mathrm{Fix}_{\mathrm{SED}}(c_1), \mathrm{Fix}_{\mathrm{SED}}(c_2))$	0.5	0.25	0.25
$\mathrm{Fix}_{\mathrm{SED}}(\mathrm{LP}^{\frac{1}{2}}(c_1, c_2))$	0.6	0.2	0.2
$\mathrm{WCAP}_{\mathrm{SED}}^{\frac{1}{2}}(c_1, c_2)$	0.6	0.2	0.2

We can now state the true generalization of Proposition 5:

Proposition 16 *Suppose $\mathcal{F} = \{X_1, \ldots, X_m\}$ is a partition. Let $\{\alpha\}$ be a set of weights. Then*

(i) $\mathrm{WCAP}_{\mathrm{SED}}^{\{\alpha\}} = \mathrm{Fix}_{\mathrm{SED}} \circ \mathrm{LP}^{\{\alpha\}} \neq \mathrm{LP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{SED}}$
(ii) $\mathrm{WCAP}_{\mathrm{GKL}_1}^{\{\alpha\}} = \mathrm{GP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} = \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{GP}^{\{\alpha\}}$
(iii) $\mathrm{WCAP}_{\mathrm{GKL}_2}^{\{\alpha\}} \neq \mathrm{GP}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}} = \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{GP}^{\{\alpha\}} = \mathrm{GP}^{\{\alpha\}}$
(iv) $\mathrm{WCAP}_{\mathrm{GKL}_2}^{\{\alpha\}} = \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{LP} = \mathrm{Fix}_{\mathrm{GKL}} \circ \mathrm{Agg}_{\mathrm{GKL}_2}^{\{\alpha\}} \neq \mathrm{Agg}_{\mathrm{GKL}_2}^{\{\alpha\}} \circ \mathrm{Fix}_{\mathrm{GKL}}$

Thus, when we move from two-cell partitions to many-cell partitions, the cooperation between geometric pooling and GKL remains, but the cooperation between linear pooling and SED breaks down. Along with Propositions 2(i) and 5(i), Theorems 10(i) and 12 also fail in full generality. Proposition 3(i) and Theorem 11(i), however, remain—they are true for many-cell partitions just as they are for two-cell partitions.

However, the situation is not quite as bleak as it might seem. There is a large set of credence functions such that, if all of our agents have credence functions in that set, then Propositions 2(i) and 5(i) and Theorem 10(i) holds. Let

$$S_{\mathcal{F}} = \left\{ c \in C_{\mathcal{F}} : \forall 1 \leq j \leq m, c(X_j) + \frac{1 - \sum_{i=1}^{m} c(X_i)}{m} \geq 0 \right\}$$

Then, if c is in $S_{\mathcal{F}}$,

$$\mathrm{Fix}_{\mathrm{SED}_1}(c)(X_j) = \mathrm{Fix}_{\mathrm{SED}_2}(c)(X_j) = c(X_j) + \frac{1 - \sum_{i=1}^{m} c(X_i)}{m}$$

and, if $c_1, \ldots, c_n$ are in $S_{\mathcal{F}}$, then

$$\text{WCAP}_{\text{SED}}^{\{\alpha\}} = \text{LP}^{\{\alpha\}} \circ \text{Fix}_{\text{SED}} = \text{Fix}_{\text{SED}} \circ \text{LP}^{\{\alpha\}}$$

as Propositions 2(i) and 5(i) say. What's more, $\text{WCAP}_{\mathfrak{D}_1}$, $\text{Fix}_{\mathfrak{D}_1} \circ \text{LP}$, and $\text{LP} \circ \text{Fix}_{\mathfrak{D}_1}$ agree for all credence functions in $S_{\mathcal{F}}$ iff $\mathfrak{D}$ is a positive linear transformation of SED, as Theorem 10(i) says. Note the following corollary: there is no Bregman divergence $\mathfrak{D}$ such that $\text{WCAP}_{\mathfrak{D}_1}$, $\text{LP} \circ \text{Fix}_{\mathfrak{D}_1}$, and $\text{Fix}_{\mathfrak{D}_1} \circ \text{LP}$ agree for *all* credence functions over a many-cell partition.

Thus, while linear pooling and SED don't always cooperate when our agents have credences over a many-cell partition, there is a well-defined set of situations in which they do. What's more, these situations are in the majority—they occupy more than half the volume of the space of possible credence functions. Thus, while it is a strike against linear pooling and SED that they do not cooperate—and indeed that there is no aggregation method that cooperates with SED and no Bregman divergence that cooperates with linear pooling—it is not a devastating blow.

9 Beyond partitions

So far, we have restricted attention to credence functions defined on partitions. In this section, we lift that restriction. Suppose Carmen and Donal are two further expert epidemiologists. They have credences in a rather broader range of propositions than Amira and Benito do. They consider the proposition, X_1, that the next 'flu pandemic will occur in 2019, but also the proposition, X_2, that it will occur in 2020, the proposition X_3 that it will occur in neither 2019 nor 2020, and the proposition, $X_1 \vee X_2$, that it will occur in 2019 or 2020. Thus, they have credences in X_1, X_2, X_3, and $X_1 \vee X_2$, where the first three propositions form a partition but the whole set of four does not. Unlike Amira and Benito, Carmen and Donal are coherent. Here are their credences:

	X_1	X_2	X_3	$X_1 \vee X_2$
Carmen (c_1)	0.2	0.3	0.5	0.5
Donal (c_2)	0.6	0.3	0.1	0.9

Since they are coherent, the question of how to fix them does not arise. So we are interested here only in how to aggregate them. If we opt to combine SED and linear pooling, there are three methods:

(LP1) Apply the method of linear pooling to the most fine-grained partition, namely, X_1, X_2, X_3, to give the aggregate credences for those three propositions. Then take the aggregate credence for $X_1 \vee X_2$ to be the sum of the aggregate credences for X_1 and X_2, as demanded by the axioms of the probability calculus.

For instance, suppose $\alpha = \frac{1}{2}$. Then
- $c^*(X_1) = \frac{1}{2}0.2 + \frac{1}{2}0.6 = 0.4$
- $c^*(X_2) = \frac{1}{2}0.3 + \frac{1}{2}0.3 = 0.3$
- $c^*(X_3) = \frac{1}{2}0.5 + \frac{1}{2}0.1 = 0.3$

- $c^*(X_1 \vee X_2) = c^*(X_1) + c^*(X_2) = 0.4 + 0.3 = 0.7$.

(LP2) Extend the method of linear pooling from partitions to more general sets of propositions in the natural way: the aggregate credence for a proposition is just the weighted arithmetic average of the credences for that proposition.

Again, suppose $\alpha = \frac{1}{2}$. Then
- $c^*(X_1) = \frac{1}{2}0.2 + \frac{1}{2}0.6 = 0.4$
- $c^*(X_2) = \frac{1}{2}0.3 + \frac{1}{2}0.3 = 0.3$
- $c^*(X_3) = \frac{1}{2}0.5 + \frac{1}{2}0.1 = 0.3$
- $c^*(X_1 \vee X_2) = \frac{1}{2}0.5 + \frac{1}{2}0.9 = 0.7$.

(LP3) Apply WCAP$_\text{SED}$, so that the aggregate credence function is the coherent credence function that minimizes the arithmetic average of the squared Euclidean distances to the credence functions.

Again, suppose $\alpha = \frac{1}{2}$. Then

$$\underset{c' \in P_{\mathcal{F}}}{\arg \min} \; \frac{1}{2}\text{SED}(c', c_1) + \frac{1}{2}\text{SED}(c', c_2) = \frac{1}{2}c_1 + \frac{1}{2}c_2.$$

It is easy to see that these three methods agree. And they continue to agree for any number of agents, any weightings, and any set of propositions. Does the same happen if we opt to combine GKL and geometric pooling? Unfortunately not. Here are the analogous three methods:

(GP1) Apply the method of geometric pooling to the most fine-grained partition, namely, X_1, X_2, X_3, to give the aggregate credences for those three propositions. Then take the aggregate credence for $X_1 \vee X_2$ to be the sum of the aggregate credences for X_1 and X_2, as demanded by the axioms of the probability calculus.

Suppose $\alpha = \frac{1}{2}$. Then
- $c^*(X_1) = \frac{\sqrt{0.2}\sqrt{0.6}}{\sqrt{0.2}\sqrt{0.6}+\sqrt{0.3}\sqrt{0.3}+\sqrt{0.5}\sqrt{0.1}} \approx 0.398$
- $c^*(X_2) = \frac{\sqrt{0.3}\sqrt{0.3}}{\sqrt{0.2}\sqrt{0.6}+\sqrt{0.3}\sqrt{0.3}+\sqrt{0.5}\sqrt{0.1}} \approx 0.345$
- $c^*(X_3) = \frac{\sqrt{0.5}\sqrt{0.1}}{\sqrt{0.2}\sqrt{0.6}+\sqrt{0.3}\sqrt{0.3}+\sqrt{0.5}\sqrt{0.1}} \approx 0.257$
- $c^*(X_1 \vee X_2) = c^*(X_1) + c^*(X_2) \approx 0.743$

(GP2) Extend the method of geometric pooling from partitions to more general sets of credence functions.

The problem with this method is that it isn't clear how to effect this extension. After all, when we geometrically pool credences over a partition, we start by taking weighted geometric averages and then we normalize. We can, of course, still take weighted geometric averages when we extend beyond partitions. But it isn't clear how we would normalize. In the partition case, we take a cell of the partition, take the weighted geometric average of the credences in that

cell, then divide through by the sum of the weighted geometric averages of the credences in the various cells of the partition. But suppose that we try this once we add $X_1 \vee X_2$ to our partition X_1, X_2, X_3. The problem is that the normalized version of the weighted geometric average of the agents' credences in $X_1 \vee X_2$ is not the sum of the normalized versions of the weighted geometric averages of the credences in X_1 and in X_2. But how else are we to normalize?

(GP3) Apply WCAP_{SED}, so that the aggregate credence function is the coherent credence function that minimizes the arithmetic average of the generalized Kullback-Leibler divergence from that credence function to the credence functions.

Again, suppose $\alpha = \frac{1}{2}$. Now, we can show that, if $c^* = \underset{c' \in P_{\mathcal{F}}}{\arg \min} \frac{1}{2}\text{GKL}(c', c_1) + \frac{1}{2}\text{GKL}(c', c_2)$, then

- $c^*(X_1) = 0.390$
- $c^*(X_2) = 0.338$
- $c^*(X_3) = 0.272$
- $c^*(X_1 \vee X_2) = 0.728$

Thus, (GP2) does not work—we cannot formulate it. And (GP1) and (GP3) disagree. This creates a dilemma for those who opt for the package containing GKL and geometric pooling. How should they aggregate credences when the agents have credences in propositions that don't form a partition? Do they choose (GP1) or (GP3)? The existence of the dilemma is a strike against GKL and geometric pooling, and a point in favour of SED and linear pooling, which avoid the dilemma.

10 The philosophical significance of the results

What is the philosophical upshot of the results that we have presented so far? I think they are best viewed as supplements that can be added to existing arguments. On their own, they do not support any particular philosophical conclusion. But, combined with an existing philosophical argument, they extend its conclusion significantly. They are, if you like, philosophical booster rockets.

There are two ways in which the results above might provide such argumentative boosts. First, if you think that the aggregate of a collection of credence functions should be the credence function that minimizes the weighted average divergence from or to those functions, then you might appeal to Proposition 3 or Theorems 11 and 13 either to move from a way of measuring divergence to a method of aggregation, or to move from an aggregation method to a favoured divergence—recall: each of these results holds for any size of partition. Thus, given an argument for linear pooling, and an argument that you should aggregate by minimizing weighted average distance *from* the aggregate to the agent, you might cite Theorem 11(i) and argue for measuring how far one credence function lies from another using SED. Or, given an argument that you should aggregate by minimizing weighted average divergence *to* the agents, and an argument in favour of GKL, you might cite Theorem 11(ii) and conclude further that you should aggregate by GP_. Throw in an argument that you should fix incoherent

credence functions by minimizing distance from coherence and this gives an argument for GP.

Second, if you think that the three possible ways of producing a single coherent credence function from a collection of possibly incoherent ones should cooperate—that is, if you think that aggregate-then-fix, fix-then-aggregate, and the weighted coherent approximation principle should all give the same outputs when supplied with the same input—then you might appeal to Theorems 10 and 12, or to the restricted versions that hold for any size of partition, to move from aggregation method to divergence, or vice versa. For instance, if you think we should fix by minimizing distance from coherence, you might use Theorem 10(ii) to boost an argument for geometric pooling to give an argument for GKL. And, if the agents you wish to aggregate have credence functions in $S_{\mathcal{F}}$, you might use Theorem 10(i) to boost an argument for linear pooling so that it becomes also an argument for SED. And so on.

We begin, in this section, by looking at the bases for these two sorts of argument. Then we consider the sorts of philosophical argument to which our boosts might be applied. That is, we ask what sorts of arguments we might give in favour of one divergence over another, or one aggregation method over another, or whether we should fix by minimizing distance to or from coherence.

10.1 Aggregating as minimizing weighted average distance

Why think that we should aggregate the credence functions of a group of agents by finding the single credence function from or to which the weighted average distance is minimal? There is a natural argument that appeals to a principle that is used elsewhere in Bayesian epistemology. Indeed, we have used it already in this paper in our brief justification for fixing incoherent crecences by minimizing distance from or to coherence. It is *the principle of minimal mutilation*. The idea is this: when you are given a collection of credences that you know are flawed in some way, and from which you wish to extract a collection that is not flawed, you should pick the unflawed collection that involves the least possible change to the original flawed credences.

The principle of minimal mutilation is often used in arguments for credal updating rules. Suppose you have a prior credence function, and then you acquire new evidence. Since it is new evidence, your prior likely does not satisfy the constraints that your new evidence places on your credences. How are you to respond? Your prior is now seen to be flawed—it violates a constraint imposed by your evidence—so you wish to find credences that are not flawed in this way. A natural thought is this: you should move to the credence function that does satisfy those constraints and that involves the least possible change in your prior credences; in our terminology, you should move to the credence function whose distance from or to your prior amongst those that satisfies the constraints is minimal. This is the principle of minimal mutilation in action. And its application has lead to a number of arguments for various updating rules, such as Conditionalization, Jeffrey Conditionalization, and others (Williams 1980; Diaconis and Zabell 1982; Leitgeb and Pettigrew 2010b).

As we have seen in Sect. 2, the principle of minimal mutilation is also our motivation for fixing an incoherent credence function c by taking $\mathrm{Fix}_{\mathfrak{D}_1}(c)$ or $\mathrm{Fix}_{\mathfrak{D}_2}(c)$, for some divergence $\mathfrak{D}$. And the same holds when you have a group of agents, each possibly

incoherent, and some of whom disagree with each other. Here, again, the credences you receive are flawed in some way: within an individual agent's credence functions, the credences may not cohere with each other; and between agents, there will be conflicting credence assignments to the same proposition. We thus wish to find a set of credences that are not flawed in either of these ways. We want one credence per proposition, and we want all of the credences to cohere with one another. We do this by finding the set of such credences that involves as little change as possible from the original set. The weightings in the weighted average of the divergences allow us to choose which agent's credences we'd least like to change (they receive highest weighting) and whose we are happiest to change (they receive lowest weighting).

10.2 The No Dilemmas argument

As we noted above, in order to use Theorem 10(ii), say, to extract a reason for using GKL from a reason for aggregating by geometric pooling, we must argue that the three possible ways of producing a single coherent credence function from a collection of possibly incoherent credence functions *should* cooperate. That is, we must claim that aggregate-then-fix, fix-then-aggregate, and the weighted coherent approximation principle *should* all give the same outputs when supplied with the same input. The natural justification for this is a *no dilemmas argument*. The point is that, if the three methods don't agree on their outputs when given the same set of inputs, we are forced to pick one of those different outputs to use. And if there is no principled reason to pick one or another, whichever we pick, we cannot justify using it rather than one of the others. Thus, for instance, given any decision where the different outputs recommend different courses of action, we cannot justify picking the action recommended by one of the outputs over the action recommended by one of the others. Similarly, given any piece of statistical reasoning in which using the different outputs as prior probabilities results in different conclusions at the end, we cannot justify adopting the conclusion mandated by one of the outputs over the conclusion mandated by one of the others.

Does this no dilemmas argument work? Of course, you might object if you think that there are principled reasons for preferring one method to another. That is, you might answer the no dilemmas argument by claiming that there is no dilemma in the first place, because one of the options is superior to the others. For instance, you might claim that it is more natural to fix first and then aggregate than to aggregate first and then fix. You might say that we can only expect an aggregate to be epistemically valuable when the credences to be aggregated are epistemically valuable; and you might go on to say that credences aren't epistemically valuable if they're incoherent.[10] But this claim is compatible with aggregating first and then fixing. I can still say that aggregates are only as epistemically valuable as the credence functions they aggregate, and I can still say that the more coherent a credence function the more epistemically valuable it is, and yet also say that I should aggregate and then fix. After all, while the aggregate won't be very epistemically valuable when the agents are incoherent, once I've fixed it and made it coherent it will be. And there's no reason to think it will be epistemically

---

[10] Thanks to Ben Levinstein for urging me to address this line of objection.

worse than if I first fixed the agents and then aggregated them. So I think this particular line of argument fails.

Here's another. There are many different reasons why an agent might fail to live up to the ideal of full coherence: the computations required to maintain coherence might be beyond their cognitive powers; or coherence might not serve a sufficiently useful practical goal to justify devoting the agent's limited cognitive resources to its pursuit; or an agent with credences over a partition might only ever have considered each cell of that partition on its own, separately, and never have considered the logical relations between them, and this might have lead her inadvertently to assign incoherent credences to them. So it might be that, while there is no reason to favour aggregating-then-fixing over fixing-then-aggregating or the weighted coherent approximation principle *in general*, there is reason to favour one or other of these methods *once we identify the root cause of the agent's incoherence*.

For instance, you might think that, when her incoherence results from a lack of attention to the logical relations between the propositions, it would be better to treat the individual credences in the individual members of the partition separately for as long as possible, since they were set separately by the agent. And this tells in favour of aggregating via LP or GP_ first, since the aggregate credence each assigns to a given proposition is a function only of the credences that the agents assign to that proposition. I don't find this argument compelling. After all, it is precisely the fact that the agent has considered these propositions separately that has given rise to their flaw. Had they considered them together as members of one partition, they might have come closer to the ideal of coherence. So it seems strange to wish to maintain that separation for as long as possible. It seems just as good to fix the flaw that has resulted from keeping them separate so far, and then aggregate the results. However, while I find the argument weak, it does show how we might look to the reasons behind the incoherence in a group of agents, or perhaps the reasons behind their disagreements, in order to break the dilemma and argue that the three methods for fixing and aggregating need not agree.

10.3 Minimizing divergence from or to coherence

As we have seen in Propositions 3 and 5 and Theorems 10 and 12, it makes a substantial difference whether you fix incoherent credence functions by minimizing distance from or to coherence, and whether you aggregate credences by minimizing distance from or to the agents' credence functions when you aggregate them. Do we have reason to favour one of these directions or the other?

Here is one argument, at least in the case of fixing incoherent credences. Recall Theorem 9 from above. Suppose c is an incoherent credence function. Then let c^* be the coherent credence function for which the divergence *from c^* to c* is minimal, and let $c^\dagger$ be the coherent credence function for which the distance *to $c^\dagger$ from c* is minimal. Then c^* is guaranteed to be more accurate than c, while $c^\dagger$ is not. Now, this gives us a reason for fixing an incoherent credence function by minimizing the distance *from* coherence rather than the distance *to* coherence. It explains why we should use $\text{Fix}_{\mathfrak{D}_1}$ rather than $\text{Fix}_{\mathfrak{D}_2}$ to fix incoherent credence functions. After all, when $\mathfrak{D}$ is a Bregman

divergence, $\mathrm{Fix}_{\mathfrak{D}_1}(c)$ is guaranteed to be more accurate than c, by Theorem 9, whereas $\mathrm{Fix}_{\mathfrak{D}_2}(c)$ is not.

10.4 Linear pooling versus geometric pooling

In this section, we briefly survey some of the arguments for and against linear or geometric pooling. For useful surveys of the virtues and vices of different aggregation methods, see Genest and Zidek (1986), Russell et al. (2015) and Dietrich and List (2015).

In favour of aggregating by linear pooling (LP): First, McConway (1981) and Wagner (1982) show that, amongst the aggregation methods that always take coherent credence functions to coherent aggregates, linear pooling is the only one that satisfies what Dietrich and List (2015) call *eventwise independence* and *unanimity preservation*. Eventwise Independence demands that aggregation is done proposition-wise using the same method for each proposition. That is, an aggregation methods T satisfies Eventwise Independence if there is a function $f : [0, 1]^n \to [0, 1]$ such that $T(c_1, \ldots, c_n)(X_j) = f(c_1(X_j), \ldots, c_n(X_j))$ for each cell X_j in our partition $\mathcal{F}$. Unanimity Preservation demands that, when all agents have the same credence function, their aggregate should be that credence function. That is, $T(c, \ldots, c) = c$, for any coherent credence function c. It is worth noting, however, that GP_- also satisfies both of these constraints; but of course it doesn't always take coherent credences to coherent aggregates.

Second, in a previous paper, I showed that linear pooling is recommended by the accuracy-first approach in epistemology, which we met in Sect. 6 (Pettigrew 2016b). Suppose, like nearly all parties to the accuracy-first debate, you measure the accuracy of credences using what is known as a strictly proper scoring rule; this is equivalent to measuring the accuracy of a credence function at a world as the divergence from the omniscient credence function at that world to the credence function, where the divergence in question is an additive Bregman divergence. Suppose further that each of the credence functions you wish to aggregate is coherent. Then, if you aggregate by anything other than linear pooling, there will be an alternative aggregate credence function that each of the agents expects to be more accurate than your aggregate. I argue that a credence function cannot count as the aggregate of a set of credence functions if there is some alternative that each of those credence functions expects to do better epistemically speaking.

Third, as we saw in Sect. 9, linear pooling remains a sensible aggregation method when we wish to aggregate agents with credences over propositions that don't form a partition.

Against linear pooling: First, Dalkey (1975) notes that it does not commute with conditionalization.[11] Thus, if you first conditionalize your agents on a piece of evidence and then linear pool, this usually gives a different result from linear pooling first and then conditionalizing (at least if you use the same weights before and after the evidence is accommodated). That is, typically,

---

[11] For responses to this objection to linear pooling, see Madansky (1964), McConway (1981) and Pettigrew (2016b).

$$\mathrm{LP}^{\{\alpha\}}(c_1, \ldots, c_n)(-|E) \neq \mathrm{LP}^{\{\alpha\}}(c_1(-|E), \ldots, c_n(-|E))$$

Second, Laddaga ([1977]) and Lehrer and Wagner ([1983]) note that linear pooling does not preserve relationships of probabilistic independence.[12] Thus, usually, if A and B are probabilistically independent relative to each c_i, they will not be probabilistically independent relative to the linear pool. That is, if $c_i(A|B) = c_i(A)$ for each c_i, then usually

$$\mathrm{LP}^{\{\alpha\}}(c_1, \ldots, c_n)(A|B) \neq \mathrm{LP}^{\{\alpha\}}(c_1, \ldots, c_n)(A).$$

Third, as we saw in Sect. [8], there is no Bregman divergence that always cooperates with linear pooling. While SED cooperates when the agents to be aggregated have credence functions in $S_{\mathcal{F}}$, it does not necessarily do so otherwise.

Geometric pooling (GP) succeeds where linear pooling fails, and fails where linear pooling succeeds. The accuracy-first argument tells against it; and it violates Eventwise Independence. But it commutes with conditionalization. What's more, while linear pooling typically returns an incoherent aggregate when given incoherent agents, geometric pooling always returns a coherent aggregate, whether the agents are coherent or incoherent. Of course, this is because we build in that coherence by hand when we normalize the geometric averages of the agents' credences. Geometric pooling faces a dilemma when we move beyond partitions, but there is a Bregman divergence that always cooperates with it when we restrict attention to partitions.

10.5 Squared Euclidean distance versus generalized Kullback-Leibler divergence

There are a number of different ways in which we might argue in favour of SED or GKL.

In favour of squared Euclidean distance (SED): First, there is an argument that I have offered elsewhere that proceeds in two steps (Pettigrew [2016a], Chapter 4): (i) we should measure how far one credence function lies from another using additive Bregman divergences, because only by doing so can we capture two competing senses of accuracy—the alethic and the calibrationist—in one measure; (ii) the distance from one credence function to another should be the same as the distance to the first credence from the second, so that our divergence should be symmetric. Since SED is the only symmetric Bregman divergence, this gives an argument in its favour.

Second, D'Agostino and Sinigaglia ([2010]) argue for SED axiomatically. SED is the only way of measuring how far one credence function lies from another that satisfies certain plausible formal constraints. Csiszár ([1991], [2008]) offers axiomatic characterizations of SED and GKL that allow us to tell between them on the basis of their formal features.

---

[12] For responses to this objection to linear pooling, see Pettigrew ([2016b]); Genest and Wagner ([1987]) and Wagner ([2010]).

Third, some argue in favour of SED indirectly. They argue primarily in favour of the so-called Brier score.[13] This is a particular inaccuracy measure that is widely used in the accuracy-first epistemology literature. It is a strictly proper scoring rule. And it is the inaccuracy measure that you obtain by using SED and taking inaccuracy to be divergence from omniscient credences. Thus, arguments for the Brier score can be extended to give arguments for SED. How might we argue for the Brier score? First, Schervish (1989) showed that agents with different practical ends and different opinions about what decisions they are likely to face will value their credences for pragmatic purposes using different strictly proper scoring rules. Thus, we might argue for the Brier score if we have particular practical ends and if we hold a certain view about the sorts of decisions we'll be asked to make (Levinstein 2017). Second, you might argue for the Brier score because of the way that it scores particular credences. It is more forgiving of extreme inaccuracy than is, for instance, the logarithmic scoring rule associated with GKL (Joyce 2009).

Against SED: if we use it to say how we should update in response to a certain sort of evidence, it gives updating rules that seem defective (Levinstein 2012; Leitgeb and Pettigrew 2010b). It does not justify either Bayesian Conditionalization or Jeffrey Conditionalization; and the alternative rules that it offers have undesirable features.

This argument against SED is also the primary argument in favour of GKL. As I show elsewhere, it is difficult to find a Bregman divergence other than GKL that warrants updating by Conditionalization and Jeffrey Conditionalization (Pettigrew 2016a, Theorem 15.1.4).

11 Conclusion

This completes our investigation into the methods by which we might produce a single coherent credence function from a group of possibly incoherent expert credence functions. At the heart of our investigation is a set of results that suggest that squared Euclidean distance pairs naturally with linear pooling (if anything does), while the generalized Kullback-Leibler divergence pairs naturally with geometric pooling. I suggested that these results might be used by philosophers to argue for an aggregation method if they have reason to favour a particular divergence, or to argue for a particular divergence if they have reason to favour one aggregation method over another.

---

[13] According to the Brier score, the inaccuracy of a credence function c at a world w is

$$\mathfrak{B}(c, w) = \sum_{i=1}^{m} (v_w(X_i) - c(X_i))^2$$

where v_w is the omniscient credence function at world w.

Appendix: Proofs

Some useful lemmas

We begin by stating and proving some useful lemmas to which we will appeal in our proofs. Throughout, we suppose:

- $\mathfrak{D}$ is the additive Bregman divergence generated by φ;
- $\alpha_1, \ldots, \alpha_n \geq 0$ and $\sum_{k=1}^{n} \alpha_k = 1$;
- c, c' are credence functions defined on the partition $\mathcal{F} = \{X_1, \ldots, X_m\}$;
- $C_\mathcal{F} = \{c : \{X_1, \ldots, X_m\} \to [0, 1]\}$
- $P_\mathcal{F} = \{c : \{X_1, \ldots, X_m\} \to [0, 1] \mid \sum_{i=1}^{m} c(X_i) = 1\}$.

Lemma 17 (i) *If* $c^* = \mathrm{Agg}_{\mathrm{SED}}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min_{c' \in C_\mathcal{F}} \sum_{k=1}^{n} \alpha_k \mathrm{SED}(c', c_k)$, *then, for all* $1 \leq j \leq m$,

$$c^*(X_j) = \sum_{k=1}^{n} \alpha_k c_k(X_j)$$

(ii) *If* $c^* = \mathrm{WCAP}_{\mathrm{SED}}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min_{c' \in P_\mathcal{F}} \sum_{k=1}^{n} \alpha_k \mathrm{SED}(c', c_k)$, *then, for all* $1 \leq j \leq m$,

$$c^*(X_j) = \begin{cases} \sum_{k=1}^{n} \alpha_k c_k(X_j) + K & \text{if } \sum_{k=1}^{n} \alpha_k c_k(X_j) + K \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

where K is the unique number such that

$$\sum_{i : \alpha_k c_k(X_i) + K \geq 0} \alpha_k c_k(X_i) + K = 1.$$

Lemma 18 (i) *If* $c^* = \mathrm{Agg}_{\mathrm{GKL}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min_{c' \in C_\mathcal{F}} \sum_{k=1}^{n} \alpha_k \mathrm{GKL}(c', c_k)$, *then, for all* $1 \leq j \leq m$,

$$c^*(X_j) = \prod_{k=1}^{n} c_k(X_j)^{\alpha_k}$$

(ii) *If* $c^* = \mathrm{WCAP}_{\mathrm{GKL}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min_{c' \in P_\mathcal{F}} \sum_{k=1}^{n} \alpha_k \mathrm{GKL}(c', c_k)$, *then, for all* $1 \leq j \leq m$,

$$c^*(X_j) = \frac{\prod_{k=1}^{n} c_k(X_j)^{\alpha_k}}{\sum_{i=1}^{m} \prod_{k=1}^{n} c_k(X_i)^{\alpha_k}}$$

Lemma 19 (i) *If* $c^* = \mathrm{Agg}_{\mathrm{GKL}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min\limits_{c' \in C_{\mathcal{F}}} \sum_{k=1}^{n} \alpha_k \mathrm{GKL}(c_k, c')$, *then,*

for all $1 \leq j \leq m$,

$$c^*(X_j) = \sum_{k=1}^{n} \alpha_k c_k(X_j)$$

(ii) *If* $c^* = \mathrm{WCAP}_{\mathrm{GKL}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min\limits_{c' \in P_{\mathcal{F}}} \sum_{k=1}^{n} \alpha_k \mathrm{GKL}(c_k, c')$, *then, for all*

$1 \leq j \leq m$,

$$c^*(X_j) = \frac{\sum_{k=1}^{n} \alpha_k c_k(X_j)}{\sum_{i=1}^{m} \sum_{k=1}^{n} \alpha_k c_k(X_i)}$$

To prove these, we appeal to the Karush-Kuhn-Tucker conditions, which are summarised in the following theorem (Karush 1939):

Theorem 20 (KKT conditions) *Suppose* $f, g_1, \ldots, g_k, h_1, \ldots, h_n : \mathbb{R}^m \to \mathbb{R}$ *are smooth functions. Consider the following minimization problem. Minimize* $f(x_1, \ldots, x_m)$ *relative to the following constraints:*

$$g_i(x_1, \ldots, x_m) \leq 0 \ \text{for } i = 1, \ldots, k$$
$$h_j(x_1, \ldots, x_m) = 0 \ \text{for } j = 1, \ldots, n$$

If $x^* = (x_1^*, \ldots, x_m^*)$ *is a (nonsingular) solution to this minimization problem, then there exist* $\mu_1, \ldots, \mu_k, \lambda_1, \ldots, \lambda_n$ *in* $\mathbb{R}$ *such that*

(i) $\nabla f(x^*) + \sum_{i=1}^{k} \mu_i \nabla g_i(x^*) + \sum_{j=1}^{n} \lambda_j \nabla h_j(x^*) = 0$
(ii) $\mu_i g_i(x^*) = 0$, *for* $i = 1, \ldots, k$,
(iii) $\mu_i \geq 0$, *for* $i = 1, \ldots, k$,
(iv) $g_i(x^*) \leq 0$, *for* $i = 1, \ldots, k$,
(v) $h_j(x^*) = 0$, *for* $j = 1, \ldots, n$.

Proof of Lemma 17

(i) We appeal to Theorem 20 with:

$$f(x_1, \ldots, x_m) = \sum_{k=1}^{n} \alpha_k \sum_{i=1}^{m} (x_i - c_k(X_i))^2$$
$$g_i(x_1, \ldots, x_m) = -x_i$$

Then let
- $\mu_i = 0$, for $1 \leq i \leq m$.

Then the KKT conditions are satisfied for

$$x_j^* = \sum_{k=1}^{n} \alpha_k c_k(X_j)$$

as required.

(ii) We appeal to Theorem 20 with:

$$f(x_1, \ldots, x_m) = \sum_{k=1}^{n} \alpha_k \sum_{i=1}^{m} (x_i - c_k(X_i))^2$$

$$g_i(x_1, \ldots, x_m) = -x_i$$

$$h(x_1, \ldots, x_m) = 1 - \sum_{i=1}^{m} x_i$$

Then let

-
$$\mu_i = \begin{cases} 0 & \text{if } \sum_{k=1}^{n} \alpha_k c_k(X_i) + K > 0 \\ -2(\sum_{k=1}^{n} \alpha_k c_k(X_i) + K) & \text{otherwise} \end{cases}$$

for $1 \le i \le m$.

- $\lambda = 2K$

Then the KKT conditions are satisfied for

$$x_j^* = \begin{cases} \sum_{k=1}^{n} \alpha_k c_k(X_j) + K & \text{if } \sum_{k=1}^{n} \alpha_k c_k(X_j) > 0 \\ 0 & \text{otherwise} \end{cases}$$

as required.

Proof of Lemma 18

(i) We appeal to Theorem 20 with:

$$f(x_1, \ldots, x_m) = \sum_{k=1}^{n} \alpha_k \sum_{i=1}^{m} x_i \log \frac{x_i}{c_k(X_i)} - x_i + c_k(X_i)$$

$$g_i(x_1, \ldots, x_m) = -x_i$$

Then let

- $\mu_i = 0$, for $1 \le i \le m$.

Then the KKT conditions are satisfied for

$$x_j^* = \prod_{k=1}^{n} c_k(X_j)^{\alpha_k}$$

as required.

(ii) We appeal to Theorem 20 with:

$$f(x_1, \ldots, x_m) = \sum_{k=1}^{n} \alpha_k \sum_{i=1}^{m} x_i \log \frac{x_i}{c_k(X_i)} - x_i + c_k(X_i)$$

$$g_i(x_1, \ldots, x_m) = -x_i$$

$$h(x_1, \ldots, x_m) = 1 - \sum_{i=1}^{m} x_i$$

Then let
- $\mu_i = 0$, for $1 \le i \le m$.
- $\lambda = -\log \sum_{i=1}^{m} \prod_{k=1}^{n} c_k(X_i)^{\alpha_k}$

Then the KKT conditions are satisfied for

$$x_j^* = \frac{\prod_{k=1}^{n} c_k(X_j)^{\alpha_k}}{\sum_{i=1}^{m} \prod_{k=1}^{n} c_k(X_i)^{\alpha_k}}$$

as required.

Proof of Lemma 19

(i) We appeal to Theorem 20 with:

$$f(x_1, \ldots, x_m) = \sum_{k=1}^{n} \alpha_k \sum_{i=1}^{m} c_k(X_i) \log \frac{c_k(X_i)}{x_i} - c_k(X_i) + x_i$$

$$g_i(x_1, \ldots, x_m) = -x_i$$

Then let
- $\mu_i = 0$, for $1 \le i \le m$.

Then the KKT conditions are satisfied for

$$x_j^* = \sum_{k=1}^{n} \alpha_k c_k(X_j)$$

as required.

(ii) We appeal to Theorem 20 with:

$$f(x_1, \ldots, x_m) = \sum_{k=1}^{n} \alpha_k \sum_{i=1}^{m} c_k(X_i) \log \frac{c_k(X_i)}{x_i} - c_k(X_i) + x_i$$

$$g_i(x_1, \ldots, x_m) = -x_i$$

$$h(x_1, \ldots, x_m) = 1 - \sum_{i=1}^{m} x_i$$

Then let
- $\mu_i = 0$, for $1 \le i \le m$.
- $\lambda = 1 - \sum_{k=1}^{n} \sum_{i=1}^{m} \alpha_k c_k(X_i)$

Then the KKT conditions are satisfied for

$$x_j^* = \frac{\sum_{k=1}^{n} \alpha_k c_k(X_j)}{\sum_{i=1}^{m} \sum_{k=1}^{n} \alpha_k c_k(X_i)}$$

as required.

Lemma 21 *Suppose $\mathcal{F} = \{X_1, X_2\}$ is a partition, and $\mathfrak{D}$ is an additive Bregman divergence generated by φ. Then*

(i) $(x, 1 - x) = \text{WCAP}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in P_{\mathcal{F}}}{\arg\min} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c', c_k)$ *iff*

$$\varphi'(x) - \varphi'(1 - x) = \sum_{k=1}^{n} \alpha_k \varphi'(c_k(X_1)) - \sum_{k=1}^{n} \alpha_k \varphi'(c_k(X_2))$$

(ii) $c^* = \text{WCAP}_{\mathfrak{D}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in P_{\mathcal{F}}}{\arg\min} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c_k, c')$ *iff*

$$\varphi''(x)(x - \sum_{k=1}^{n} \alpha_k c_k(X_1)) = \varphi''(1 - x)(1 - x - \sum_{k=1}^{n} \alpha_k c_k(X_2))$$

Proof of Lemma 21. Straightforward calculus.

Proof of Propositions 2–5, 15, 16

- Since $\text{Fix}_{\mathfrak{D}_i}(c) = \text{WCAP}^{\{1\}}(c)$, Propositions 1(i) and 15(i) follow from Lemma 17(ii).
- Since $\text{Fix}_{\mathfrak{D}_i}(c) = \text{WCAP}^{\{1\}}(c)$, Proposition 1(ii) and 15(ii) follow from Lemmas 18(ii) and 19(ii).
- Proposition 2 is straightforward, given the definitions of LP and GP, together with Proposition 1.
- Proposition 3 follows from Lemmas 17(i), 18(i), and 19(i).
- Proposition 4 is straightforward, given the definition GP, together with Lemmas 18 and 19.
- Propositions 5 and 16 follow from Lemmas 17, 18, 19.

Proof of Propositions 6 and 7

Recall:

$$\text{WGCAP}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \underset{c' \in P_{\mathcal{F}}}{\arg\min} \prod_{k=1}^{n} \mathfrak{D}(c', c_k)^{\alpha_k}$$

First, suppose $c' \neq c_k$, for all $1 \leq k \leq n$. Then, by the definition of a divergence, for all $1 \leq k \leq n$, $\mathfrak{D}(c', c_k) > 0$. Thus, $\prod_{k=1}^{n} \mathfrak{D}(c', c_k)^{\alpha_k} > 0$. Next, suppose $c' = c_k$ for some $1 \leq k \leq n$. Then, again by the definition of a divergence, $\mathfrak{D}(c', c_k) = 0$. Thus, $\prod_{k=1}^{n} \mathfrak{D}(c', c_k)^{\alpha_k} = 0$. Thus, $\prod_{k=1}^{n} \mathfrak{D}(c', c_k)^{\alpha_k}$ is minimized iff $c' = c_k$ for some $1 \leq k \leq n$. And similarly for $\text{WGCAP}_{\mathfrak{D}_2}^{\{\alpha\}}$, $\text{GAgg}_{\mathfrak{D}_1}$, and $\text{GAgg}_{\mathfrak{D}_2}$.

Proof of Theorem 10

Proof of Theorem 10(i) Suppose $\text{Fix}_{\mathfrak{D}_1} \circ \text{LP} = \text{WCAP}_{\mathfrak{D}_1}$. Given $0 \leq a, b \leq 1$ and $0 \leq \alpha \leq 1$, let $(x, 1 - x)$ be the coherent credence function that results from applying both procedures to the credence functions $(a, 0)$ (which assigns a to X and 0 to $\overline{X}$) and $(b, 0)$ (which assigns b to X and 0 to $\overline{X}$)—it assigns x to X and $1 - x$ to $\overline{X}$. That is,

$$(x, 1 - x) = \text{Fix}_{\mathfrak{D}_1}(\text{LP}^\alpha((a, 0), (b, 0)) = \text{WCAP}^\alpha_{\mathfrak{D}_1}((a, 0), (b, 0))$$

By Lemma 21,

$$\varphi'(x) - \varphi'(1 - x) = \varphi'(\alpha a + (1 - \alpha)b) - \varphi'(\alpha \cdot 0 + (1 - \alpha) \cdot 0)$$

And, again by Lemma 21,

$$\varphi'(x) - \varphi'(1 - x) = (\alpha\varphi'(a) + (1 - \alpha)\varphi'(b)) - (\alpha\varphi'(0) + (1 - \alpha)\varphi'(0))$$

So

$$\varphi'(\alpha a + (1 - \alpha)b) - \varphi'(\alpha \cdot 0 + (1 - \alpha) \cdot 0) =$$
$$(\alpha\varphi'(a) + (1 - \alpha)\varphi'(b)) - (\alpha\varphi'(0) + (1 - \alpha)\varphi'(0))$$

So

$$\varphi'(\alpha a + (1 - \alpha)b) = \alpha\varphi'(a) + (1 - \alpha)\varphi'(b)$$

Thus, $\varphi'(x) = kx + c$ for some constants k, c. And so $\varphi(x) = mx^2 + kx + c$, for some constants m, k, c. Since φ is strictly convex, $m > 0$. Now, it turns out that, if ψ is a strictly convex function and $\theta(x) = \psi(x) + kx + c$, then φ and ψ generate the same Bregman divergence. After all,

$$\theta(x) - \theta(y) - \theta'(y)(x - y) = (\psi(x) + kx + c) - (\psi(y) + ky + c) - (\psi'(x) + k)(x - y)$$
$$= \psi(x) - \psi(y) - \psi'(y)(x - y)$$

So $\mathfrak{D}$ is a positive linear transformation of SED, as required.

Proof of Theorem 10(ii) Suppose $\text{Fix}_{\mathfrak{D}_1} \circ \text{GP} = \text{GP} \circ \text{Fix}_{\mathfrak{D}_1} = \text{WCAP}_{\mathfrak{D}_1}$. And let (a, b) be a credence function on $\{X, \overline{X}\}$. Then note that $\text{Fix}_{\mathfrak{D}_1}(\text{GP}((a, b))) = \text{GP}((a, b))$, since geometric pooling also fixes; and, $\text{GP}(\text{Fix}_{\mathfrak{D}_1}((a, b))) = \text{Fix}_{\mathfrak{D}_1}((a, b))$, since pooling a single coherent credence function leaves it as it is. Thus, $\text{GP}((a, b)) = \text{Fix}_{\mathfrak{D}_1}((a, b))$. But $\text{GP}((a, b)) = \left(\frac{a}{a+b}, \frac{b}{a+b}\right)$. And, by Lemma 21, $(x, 1 - x) = \text{Fix}_{\mathfrak{D}_1}((a, b))$ iff

$$\varphi'(x) - \varphi'(1 - x) = \varphi'(a) - \varphi'(b)$$

Thus,

$$\varphi' \left(\frac{a}{a+b} \right) - \varphi' \left(\frac{b}{a+b} \right) = \varphi'(a) - \varphi'(b) \tag{1}$$

for all $0 \leq a, b \leq 1$. We will use this identity below.

Next, since $\mathrm{Fix}_{\mathfrak{D}_1} \circ \mathrm{GP} = \mathrm{WCAP}_{\mathfrak{D}_1}$, and since $\mathrm{Fix}_{\mathfrak{D}_1} \circ \mathrm{GP}$ is just geometric pooling, and since geometric pooling takes two credence functions (a, b) and (a', b') and returns

$$\left(\frac{a^\alpha a'^{1-\alpha}}{a^\alpha a'^{1-\alpha} + b^\alpha b'^{1-\alpha}}, \frac{b^\alpha b'^{1-\alpha}}{a^\alpha a'^{1-\alpha} + b^\alpha b'^{1-\alpha}} \right)$$

then $\mathrm{WCAP}_{\mathfrak{D}_1}$ must return that too when given (a, b) and (a', b'). Thus, by Lemma 21,

$$\varphi' \left(\frac{a^\alpha a'^{1-\alpha}}{a^\alpha a'^{1-\alpha} + b^\alpha b'^{1-\alpha}} \right) - \varphi' \left(\frac{b^\alpha b'^{1-\alpha}}{a^\alpha a'^{1-\alpha} + b^\alpha b'^{1-\alpha}} \right)$$
$$= (\alpha \varphi'(a) + (1-\alpha)\varphi'(a')) - (\alpha \varphi'(b) + (1-\alpha)\varphi'(b'))$$

Now, by the identity (1) proved above, we have

$$\varphi' \left(\frac{a^\alpha a'^{1-\alpha}}{a^\alpha a'^{1-\alpha} + b^\alpha b'^{1-\alpha}} \right) - \varphi' \left(\frac{b^\alpha b'^{1-\alpha}}{a^\alpha a'^{1-\alpha} + b^\alpha b'^{1-\alpha}} \right) = \varphi'(a^\alpha a'^{1-\alpha}) - \varphi'(b^\alpha b'^{1-\alpha})$$

So

$$\varphi'(a^\alpha a'^{1-\alpha}) - \varphi'(b^\alpha b'^{1-\alpha}) = (\alpha \varphi'(a) + (1-\alpha)\varphi'(a')) - (\alpha \varphi'(b) + (1-\alpha)\varphi'(b')) \tag{2}$$

for all $0 \leq a, b, a', b' \leq 1$ and $0 \leq \alpha \leq 1$. So let $b = a' = b' = 1$. Then

$$\varphi'(a^\alpha) - \varphi'(1) = (\alpha \varphi'(a) + (1-\alpha)\varphi'(1)) - (\alpha \varphi'(1) + (1-\alpha)\varphi'(1))$$

So

$$\varphi'(a^\alpha) = \alpha \varphi'(a) + (1-\alpha)\varphi'(1) \tag{3}$$

for all $0 \leq a \leq 1$ and $0 \leq \alpha \leq 1$. Now, take any $0 \leq a, b \leq 1$. Then there are $0 \leq c, d \leq 1$ and $0 \leq \alpha \leq 1$ such that $a = c^\alpha$ and $b = d^{1-\alpha}$ (in fact, you can always take $\alpha = \frac{1}{2}$). Then, by identity (2) from above,

$$\varphi'(ab) - \varphi'(1) = \varphi'(c^\alpha d^{1-\alpha}) - \varphi'(1) = \alpha \varphi'(c) + (1-\alpha)\varphi'(d) - \varphi(1)$$

But by identity (3) from above,

- $\alpha \varphi'(c) = \varphi'(c^\alpha) - (1-\alpha)\varphi'(1) = \varphi'(a) - (1-\alpha)\varphi'(1)$
- $(1-\alpha)\varphi'(d) = \varphi'(d^{1-\alpha}) - \alpha \varphi'(1) = \varphi'(b) - \alpha \varphi'(1)$

So

$$\varphi'(ab) = \varphi'(a) - (1-\alpha)\varphi'(1) + \varphi'(b) - \alpha \varphi'(1)$$

iff

$$\varphi'(ab) = \varphi'(a) + \varphi'(b) - \varphi'(1)$$

for all $0 \leq a, b \leq 1$. And this is the Cauchy functional equation for the logarithmic function. So $\varphi'(x) = m \log x + k$, for some constants m, k. Hence, $\varphi(x) = m(x \log x - x) + kx + c$, for some constant c. As we noted above, if ψ is a strictly convex function and $\theta(x) = \psi(x) + kx + c$, then θ generates the same Bregman divergence as ψ. And thus, $\mathfrak{D}$ is a positive linear transformation of GKL, as required.

Proof of Theorem 12

Proof of Theorem 12(i) The crucial fact is this: Let

$$f_1(x_1, \ldots, x_m) = \mathfrak{D}((\sum_{k=1}^{n} \alpha_k c_k(X_1), \ldots, \sum_{k=1}^{n} \alpha_k c_k(X_m)), (x_1, \ldots, x_m))$$

and

$$f_2((x_1, \ldots, x_n) = \sum_{k=1}^{n} \alpha_k \mathfrak{D}((c_k(X_1), \ldots, c_k(X_m)), (x_1, \ldots, x_m))$$

Then,

$$\frac{\partial}{\partial x_i} f_1(x_1, \ldots, x_m) = \varphi''(x_i) \left(x_i - \sum_{k=1}^{n} \alpha_k c_k(x_i) \right) = \frac{\partial}{\partial x_i} f_2(x_1, \ldots, x_m)$$

Thus, whatever minimizes f_1 relative to side constraints also minimizes f_2 relative to those same side constraints, and vice versa.

Proof of Theorem 12(ii) If $c_1, \ldots, c_n$ are coherent, then

$$\text{Fix}_{\mathfrak{D}_2}(\text{LP}^{\{\alpha\}}(c_1, \ldots, c_n)) = \text{LP}^{\{\alpha\}}(c_1, \ldots, c_n) = \text{LP}^{\{\alpha\}}(\text{Fix}_{\mathfrak{D}_2}(c_1), \ldots, \text{Fix}_{\mathfrak{D}_2}(c_n))$$

as required.

Proof of Theorem 12(iii) Suppose $\varphi''(x) = \varphi''(1-x)$. Now, by Lemma 21, $(x, 1-x) = \text{WCAP}_{\mathfrak{D}_2}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg \min_{c' \in P_{\mathcal{F}}} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c_k, c')$ iff

$$\varphi''(x)(x - \sum_{k=1}^{n} \alpha_k c_k(X_1)) = \varphi''(1-x)(1 - x - \sum_{k=1}^{n} \alpha_k c_k(X_2))$$

iff

$$x - \sum_{k=1}^{n} \alpha_k c_k(X_1) = 1 - x - \sum_{k=1}^{n} \alpha_k c_k(X_2)$$

iff

$$x = \sum_{k=1}^{n} \alpha_k c_k(X_1) + \frac{1 - \sum_{k=1}^{n} \alpha_k c_k(X_1) - \sum_{k=1}^{n} \alpha_k c_k(X_2)}{2}$$

Thus, $(x, 1 - x) = \mathrm{Fix}_{\mathfrak{D}_2}(c)$ iff

$$x = c(X_1) + \frac{1 - c(X_1) - c(X_2)}{2}$$

Using these, it is easy to verify that $\mathrm{WCAP}_{\mathfrak{D}_2}^{\{\alpha\}} = \mathrm{LP} \circ \mathrm{Fix}_{\mathfrak{D}_2} = \mathrm{Fix}_{\mathfrak{D}_2} \circ \mathrm{LP}$.

Proof of Theorems 11 and 13

Proof of Theorem 11(i) By Theorem 20, $c^* = \mathrm{Agg}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) = \arg\min_{c' \in C_{\mathcal{F}}} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c', c_k)$ iff, for all $1 \le j \le m$,

$$\varphi'(c^*(X_j)) = \sum_{k=1}^{n} \alpha_k \varphi'(c_k(X_j))$$

And of course $c^* = \mathrm{LP}^{\{\alpha\}}(c_1, \ldots, c_n)$ iff, for all $1 \le j \le m$,

$$c^*(X_j) = \sum_{k=1}^{n} \alpha_k c_k(X_j)$$

Thus, $\mathrm{Agg}_{\mathfrak{D}_1} = \mathrm{LP}$ iff, for any $\alpha_1, \ldots, \alpha_n$, and $c_1, \ldots, c_n$,

$$\varphi'\left(\sum_{k=1}^{n} \alpha_k c_k(X_j)\right) = \sum_{k=1}^{n} \alpha_k \varphi'(c_k(X_j))$$

iff, for any $0 \le x, y, \le 1$, and $0 \le \alpha \le 1$,

$$\varphi'(\alpha x + (1 - \alpha)y) = \alpha \varphi'(x) + (1 - \alpha)\varphi'(y)$$

And thus, $\varphi'(x) = kx + c$, for some constants k, c. From this point, the proof proceeds in the same fashion as the proof of Theorem 10(i).

Proof of Theorem 11(ii) Again, by Theorem 20, $c^* = \text{Agg}_{\mathfrak{D}_1}^{\{\alpha\}}(c_1, \ldots, c_n) =$ $\arg\min_{c' \in C_{\mathcal{F}}} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c', c_k)$ iff, for all $1 \leq j \leq m$,

$$\varphi'(c^*(X_j)) = \sum_{k=1}^{n} \alpha_k \varphi'(c_k(X_j))$$

And of course $c^* = \text{GP}_{-}^{\{\alpha\}}(c_1, \ldots, c_n)$ iff, for all $1 \leq j \leq m$,

$$c^*(X_j) = \prod_{k=1}^{n} c_k(X_j)^{\alpha_k}$$

Thus, $\text{Agg}_{\mathfrak{D}_1} = \text{GP}_{-}$ iff, for any $\alpha_1, \ldots, \alpha_n$, and $c_1, \ldots, c_n$,

$$\varphi'\left(\prod_{k=1}^{n} c_k(X_j)^{\alpha_k}\right) = \sum_{k=1}^{n} \alpha_k \varphi'(c_k(X_j))$$

iff, for any $0 \leq x, y \leq 1$, and $0 \leq \alpha \leq 1$,

$$\varphi'(x^\alpha y^{1-\alpha}) = \alpha \varphi'(x) + (1 - \alpha)\varphi'(y)$$

And thus, $\varphi'(x) = m \log x + k$, for some constants m, k. From this point on, the proof proceeds in the same fashion as the proof of Theorem 10(ii).

Proof of Theorem 13 Again by Theorem 20, $c^* = \text{Agg}_{\mathfrak{D}_2}^{\{\alpha\}}(c_1, \ldots, c_n) =$ $\arg\min_{c' \in C_{\mathcal{F}}} \sum_{k=1}^{n} \alpha_k \mathfrak{D}(c_k, c')$ iff, for all $1 \leq j \leq m$,

$$\left(c^*(X_j) - \sum_{k=1}^{n} \alpha_k c_k(X_j)\right) \varphi''(c^*(X_j)) = 0$$

And of course $c^* = \text{LP}^{\{\alpha\}}(c_1, \ldots, c_n)$ iff, for all $1 \leq j \leq m$,

$$c^*(X_j) = \sum_{k=1}^{n} \alpha_k c_k(X_j)$$

Thus, $\text{Agg}_{\mathfrak{D}_2} = \text{LP}$ iff

$$\left(\sum_{k=1}^{n} \alpha_k c_k(X_j) - \sum_{k=1}^{n} \alpha_k c_k(X_j)\right) \varphi''(c^*(X_j)) = 0$$

And that is true for any $\mathfrak{D}$.

References

Banerjee, A., Guo, X., & Wang, H. (2005). On the optimality of conditional expectation as a bregman predictor. *IEEE Transactions of Information Theory*, *51*, 2664–69.

Bregman, L. M. (1967). The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, *78*(384), 200–217.

Csiszár, I. (1991). Why least squares and maximum entropy? An axiomatic approach to inference for linear inverse problems. *Annals of Statistics*, *19*, 2032–2066.

Csiszár, I. (2008). Axiomatic characterizations of information measures. *Entropy*, *10*, 261–273.

D'Agostino, M., & Dardanoni, V. (2009). What's so special about Euclidean distance? A characterization with applications to mobility and spatial voting. *Social Choice and Welfare*, *33*(2), 211–233.

D'Agostino, M., & Sinigaglia, C. (2010). Epistemic accuracy and subjective probability. In M. Suárez, M. Dorato, & M. Rédei (Eds.), *EPSA epistemology and methodology of science: Launch of the european philosophy of science association* (pp. 95–105). Netherlands: Springer.

Dalkey, N. C. (1975). Toward a theory of group estimation. In H. A. Linstone & M. Turoff (Eds.), *The Delphi method: Techniques and applications*. Reading, MA: Addison-Wesley.

De Bona, G., & Staffel, J. (2017). Graded incoherence for accuracy-firsters. *Philosophy of Science*, *84*(2), 189–213.

Diaconis, P. & Zabell, S. (1982). Updating Subjective Probability. *Journal of the American Statistical Association*, *77*(380), 822–830.

Dietrich, F. & List, C. (2015). Probabilistic Opinion Pooling. In A. Hájek & C. R. Hitchcock (Eds.), *Oxford Handbook of Philosophy and Probability*. Oxford: Oxford University Press.

Genest, C., & Wagner, C. (1987). Further evidence against independence preservation in expert judgement synthesis. *Aequationes Mathematicae*, *32*(1), 74–86.

Genest, C., & Zidek, J. V. (1986). Combining probability distributions: A critique and an annotated bibliography. *Statistical Science*, *1*(1), 114–135.

Gibbard, A. (2008). Rational credence and the value of truth. In T. Gendler & J. Hawthorne (Eds.), *Oxford studies in epistemology* (Vol. 2). Oxford: Oxford University Press.

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

Goldman, A. (2002). The unity of the epistemic virtues. In *Pathways to knowledge: Private and public*. New York: Oxford University Press.

Joyce, J. M. (1998). A nonpragmatic vindication of probabilism. *Philosophy of Science*, *65*(4), 575–603.

Joyce, J. M. (2009). Accuracy and coherence: Prospects for an alethic epistemology of partial belief. In F. Huber & C. Schmidt-Petri (Eds.), *Degrees of belief*. Berlin: Springer.

Karush, W. (1939). *Minima of functions of several variables with inequalities as side constraints*. Master's thesis, Department of Mathematics, University of Chicago, Chicago.

Konieczny, S., & Grégoire, E. (2006). Logic-based approaches to information fusion. *Information Fusion*, *7*, 4–18.

Konieczny, S., & Pino-Pérez, R. (1999). Merging with integrity constraints. In *Fifth European conference on symbolic and quantitative approaches to reasoning with uncertainty (ECSQARU'99)* (pp. 233–244).

Laddaga, R. (1977). Lehrer and the consensus proposal. *Synthese*, *36*, 473–477.

Lehrer, K. & Wagner, C. (1983). Probability Amalgamation and the independence issue: A reply to Laddaga. *Synthese*, *55*(3), 339–346.

Leitgeb, H., & Pettigrew, R. (2010a). An objective justification of Bayesianism I: Measuring inaccuracy. *Philosophy of Science*, *77*, 201–235.

Leitgeb, H. & Pettigrew, R. (2010b). An Objective Justification of Bayesian II: The consequences of minimizing inaccuracy. *Philosophy of Science*, *77*, 236–272.

Levinstein, B. (2017). A Pragmatist's guide to epistemic utility. *Philosophy of Science*, *84*(4), 613–638.

Levinstein, B. A. (2012). Leitgeb and pettigrew on accuracy and updating. *Philosophy of Science*, *79*(3), 413–424.

Madansky, A. (1964). *Externally Bayesian groups. Memorandum rm-4141-pr*. Santa Monica: The RAND Corporation.

Magdalou, B., & Nock, R. (2011). Income distributions and decomposable divergence measures. *Journal of Economic Theory*, *146*(6), 2440–2454.

McConway, K. J. (1981). Marginalization and linear opinion pools. *Journal of the American Statistical Association*, *76*, 410–414.

Oddie, G. (1997). Conditionalization, cogency, and cognitive value. *British Journal for the Philosophy of Science*, *48*, 533–41.

Osherson, D., & Vardi, M. Y. (2006). Aggregating disparate estimates of chance. *Games and Economical Behavior*, *56*(1), 148–173.

Paris, J. B., & Vencovská, A. (1990). A note on the inevitability of maximum entropy. *International Journal of Approximate Reasoning*, *4*, 181–223.

Paris, J. B., & Vencovská, A. (1997). In defense of the maximum entropy inference process. *International Journal of Approximate Reasoning*, *17*, 77–103.

Pettigrew, R. (2016a). *Accuracy and the laws of credence*. Oxford: Oxford University Press.

Pettigrew, R. (2016b). On the accuracy of group credences. In T. S. Gendler & J. Hawthorne (Eds.), *Oxford studies in epistemology* (Vol. 6). Oxford: Oxford University Press.

Pigozzi, G. (2006). Belief merging and the discursive dilemma: An argument-based approach to paradoxes of judgment aggregation. *Synthese*, *152*, 285–298.

Predd, J., Seiringer, R., Lieb, E. H., Osherson, D., Poor, V., & Kulkarni, S. (2009). Probabilistic coherence and proper scoring rules. *IEEE Transactions of Information Theory*, *55*(10), 4786–4792.

Predd, J. B., Osherson, D., Kulkarni, S., & Poor, H. V. (2008). Aggregating probabilistic forecasts from incoherent and abstaining experts. *Decision Analysis*, *5*(4), 177–189.

Russell, J. S., Hawthorne, J., & Buchak, L. (2015). Groupthink. *Philosophical Studies*, *172*, 1287–1309.

Schervish, M. (1989). A general method for comparing probability assessors. *The annals of statistics*, *17*, 1856–1879.

Selten, R. (1998). Axiomatic characterization of the quadratic scoring rule. *Experimental Economics*, *1*(1), 43–61.

Wagner, C. (1982). Allocation, Lehrer models, and the consensus of probabilities. *Theory and Decision*, *14*, 207–220.

Wagner, C. (2010). Peer disagreement and independence preservation. *Erkenntnis*, *74*(2), 277–288.

Williams, P. M. (1980). Bayesian Conditionalisation and the Principle of minimum information. *British Journal for the Philosophy of Science*, *31*, 131–144.