Meta-inductive Probability Aggregation and Optimal Scoring

Christian J. Feldbacher-Escamilla and **Gerhard Schurz**
Duesseldorf Center for Logic and Philosophy of Science (DCLPS)
University of Duesseldorf

Abstract

In this paper we combine the theory of probability aggregation with results of machine learning theory concerning the optimality of predictions under expert advice. In probability aggregation theory several characterisation results for linear aggregation exist. However, in linear aggregation weights are not fixed, but free parameters. We show how fixing such weights by success-based scores allows for transferring the mentioned optimality results to the case of probability aggregation.

Introduction

Probability aggregation is the theory of how to adequately aggregate several probability distributions to a single one. It is an expansion of the theory of judgment aggregation that combined problems studied by social choice theory and logic as, e.g., problems of preference aggregation and questions of voting theory. In past, research in judgment aggregation centred around the disciplines of economics and political science (cf., e.g., Arrow 1963), law (cf., e.g., Kornhauser and Sager 1986), and philosophy (cf., e.g., List and Pettit 2002). Recently, however, increasing work in judgment aggregation stems also from computer science and research on artificial intelligence (cf., e.g., Rossi, Venable, and Walsh 2011; and Grossi and Pigozzi 2014).

Already in the 1980s several characterisation results have been proven for families of probability aggregation rules (cf. Genest and Zidek 1986). However, these characterisations leave some parameters still free and uninterpreted. In this paper we provide a new approach to fix these parameters. Following suggestions of the literature on scoring rules for probabilistic forecasts, we suggest to interpret the weights in a success-based way. By cashing out results on no-regret algorithms for prediction under expert advice in another field of computer science, namely online machine learning, we show that fixing the parameters in a success-based way allows for optimal probability aggregation.

The structure of the paper is as follows: In the following section we briefly present the basics of the framework of meta-inductive probability aggregation we are interested in. Afterwards we indicate the main result of research on prediction under expert advice employed by us. Then we implement this result into the aggregation framework and show how it allows for optimal probability aggregation. We conclude in final section.

Linear Probability Aggregation

The theory of probability aggregation deals with the problem of how to aggregate a set of probability distributions. Abstractly speaking, the question is how to characterise a probability aggregation rule f which takes as input a set of n probability distributions $P_1, \dots, P_n$ and generates as output a/*the* aggregated probability distribution P_{aggr}:

$$P_{aggr} = f(P_1, \dots, P_n)$$

Usually, several constraints are put forward for such an aggregation rule. Quite common are the following three constraints:

(U) *Universal domain*: f allows as input any P that satisfies the laws of probability theory

(A) *Permutation*: f is invariant under permutation of the input: $P_{aggr} = f(P_1, \dots, P_i, P_{i+1}, \dots, P_n) = f(P_1, \dots, P_{i+1}, P_i, \dots, P_n)$

(I) *Irrelevance of Alternatives*: f aggregates propositionwise: There is an f^* such that for all propositions p: $P_{aggr}(p) = f(P_1, \dots, P_n)(p) = f^*(P_1(p), \dots, P_n(p))$

As is discussed and shown in (Lehrer and Wagner 1981, chpt. 6; and Genest and Zidek 1986, sect. 3), these three conditions characterise the family of linear probability aggregation rules which have the form of a weighted arithmetic mean:

$$P_{aggr} = \sum_{i=1}^{n} w_i \cdot P_i \tag{AM}$$

$$\text{(where } w_i \geq 0 \text{ and } w_1 + \cdots + w_n = 1\text{)}$$

It is clear that different interpretations of the weights allow for different specifications. Here we want to argue for interpreting the weights in a regret-based way, because such an interpretation allows for optimal probability aggregation. In the next section we will present such a result.

Optimality in an Expert Advice Setting

In online machine learning regret bounds of algorithms for making predictions under expert advice are studied (cf. Cesa-Bianchi and Lugosi 2006). The idea is to consider a series of events whose outcomes have to be predicted by so-called *experts* or *candidate* methods. Given these predictions the task is to construct a prediction algorithm that uses the candidate method's forecast as input and aims at approaching the predictive success of the best expert in the setting, even if the best expert in changing in time in irregular ways. Since a prediction method under expert advice combines the expert's predictions by inductively projecting the observed regrets to the future, it is called a *meta-inductive method* (Schurz 2008). The difference between a candidate method's accumulated success and that of the meta-inductive algorithm is called *regret*. The algorithm approaches the best candidate method's success if its per-round regret decreases and vanishes in the limit.

The setting of online learning are so-called *prediction games* that have the following ingredients (cf. Schurz 2008, notation adjusted):

- E is an infinite series of events consisting of variables $E_1, E_2, \dots$ whose outcomes $val_1(E), val_2(E), \dots$ are elements of the normalised interval $[0, 1]$.
- $P_{1,t}, \dots, P_{n,t}$ are the predictions of E_t (also elements of $[0, 1]$) of all n candidate methods.
- $P_{mi,t}$ is the prediction of E_t of the algorithm under investigation.

As we have indicated above, the meta-inductive algorithm"cooks up" a prediction from the present predictions and past success rates of the candidate methods. In order to keep track of the success rate of a method i one identifies the score of i's prediction about event E_t with 1 minus the loss l of this prediction and then sums up all of its scores up to round t and divides by t (cf. Schurz 2008, sect. 1):

$$s_{i,t} = \frac{\sum_{u=1}^{t} 1 - l(P_{i,u}, val_u(E))}{t}$$

The measure $s_{i,t}$ represents the average per-round success rate of candidate method i up to round t. The only assumption we make about the loss function l is that it is within $[0, 1]$, and that it is *convex* in its first argument, i.e. that the loss of a weighted average of two predictions is lower or equal to the weighted average of the losses of these two predictions. Or formally: $l(w \cdot x + (1 - w) \cdot y, z) \leq w \cdot l(x, z) + (1 - w) \cdot l(y, z)$ holds for all x, y and $w \in [0, 1]$.

Now, based on this measure for the success rate up to round t one can define a so-called *attractivity measure at figures as weight function*. The idea of such a measure is that the higher the past success of an attractive method, the higher is also its weight. Moreover, the attractivity measure cuts off those candidate methods that are not attractive, i.e., that have a lower average per-round success rate as the algorithm. Thus the weight of a candidate method P_i for the algorithm P_{mi} regarding event E_t is defined as follows (where $s_{mi,t}$ is the success per round of the meta-inductive method up to round t):

$$w_{i,t} = \frac{max(0, s_{i,t} - s_{mi,t})}{\sum_{j=1}^{n} max(0, s_{j,t} - s_{mi,t})}$$

Candidate methods that are performing worse than P_{mi} get weight 0. If P_{mi} outperforms all candidate methods, then $s_{mi,t} \geq s_{i,t}$ for all $i \in \{1, \dots, n\}$, and we stipulate $w_{i,t} = 1/n$. So, the weights are always positive and sum up to 1;

Based on these weights, we can define a weighted-average algorithm (WM) which weights the predictions of the candidate methods according to their attractivities. Such an algorithm generates predictions by the method of linear (arithmetic) aggregation as follows (cf. Cesa-Bianchi and Lugosi 2006, sect. 2.1) and (cf. Schurz 2008, sect. 7):

$$P_{mi,t+1} = \sum_{i=1}^{n} w_{i,t} \cdot P_{i,t+1} \qquad \text{(WM)}$$

In case there are no attractive methods, but also at the very beginning (E_1) the algorithm's prediction consists of the mean of all predictions. What we called here *attractivities* corresponds to positive *per-round regrets*.

The algorithm (WM) proves to be very powerful regarding the task of approaching the best candidate methods' per-round success rates: There are quite narrow bounds of P_{mi} regarding the worst-case per-round regret, i.e., the difference of their success rates compared to the success rate of the actually best candidate method. The basic result of the machine learning literature we want to employ in this paper is the following theorem on the upper bounds of the regret (cf. Cesa-Bianchi and Lugosi 2006, sect. 2.1f; and Schurz 2008, sect. 7):

Theorem. *Given the loss function l is convex it holds:*

$$s_{i,t} - s_{mi,t} \leq \sqrt{n/t} \quad \forall i \in \{1, \dots, n\}$$

This theorem shows that (WM) is a no-regret algorithm in the sense that:

$$\lim_{t \to \infty} max(s_{1,t}, \dots, s_{n,t}) - s_{mi,t} \quad \leq \quad 0$$

So, the meta-inductive algorithm's success rate and that of the best performing candidate methods converge in the limit. In the machine learning literature such prediction methods are also known as *online learnable* (cf. Shalev-Shwartz and Ben-David 2014).

The guaranteed performance of (WM) can be enhanced further by exponentially weighting the absolute regrets such that the upper bound of the per-round regret is $\sqrt{c \cdot \log(n)/t}$ with $c \geq .5$. Up to now the algorithm with the best known general upper is such an algorithm using exponentially absolute regret-weighting which guarantees such an upper bound with $c = 3.125$. To design an algorithm which has the minimal upper bound that is achievable in principle, namely $\sqrt{\log(n)/2t}$ (cf. Cesa-Bianchi and Lugosi 2006, p. 62, thrm. 3.7), is still an open task of online machine learning theory.

In the next section we are going to utilise this result in order to fix the weights of linear probability aggregation and provide rationale doing so.

Optimal Probability Aggregation

In *probabilistic* prediction games each forecaster or candidate method identifies the predicted real value with its credence of the predicted event conditional on her information about the past. In the following part of this paper we are implementing the optimality result of the foregoing section into the framework of probability aggregation.

In order to cash out the no-regret optimality result of meta-induction presented above for probability aggregation we have to change our framework: It contains:

- Again, a series of events represented by random variables $E_1, E_2, \ldots$, but now the events do not have outcomes within $[0, 1]$, but within a space of discrete (non-numerical), mutually disjoint and exhaustive values v_i, $Val = \{v_1, \ldots, v_k\}$. In order to indicate which value a random variable took on at a specific round, we assume a valuation function val to be given by:

$$val_t(v_m) = \begin{cases} 1, & \text{if the value of } E_t \text{ is } v_m \\ 0, & \text{otherwise} \end{cases}$$

- Predictions are the credences of n candidate methods for each event variable E_t in the series, represented by probability distributions $P_1, \ldots, P_n$:

$$\forall\, t, i \in \{1, \ldots, n\} \; \sum_{m=1}^{k} P_{i,t}(v_m) = 1 \; \text{ and } P_{i,t}(v_m) \geq 0$$

  So, for each event, at each round, the candidate methods provide a full probability distribution about the outcome of the event in question.

- The meta-inductive algorithm P_{mi} is also represented by a probability distribution and defined as an arithmetically weighted average of the $P_1, \ldots, P_n$; details are presented below.

The attempt to expand the framework of prediction games introduced in the foregoing section to the probabilistic setting faces the problem that the predictions are real numbers, i.e. probabilities, but the event's values are not numbers but non-numeric mutually exclusive and exhaustive values $v_1, \ldots, v_k$. However, as we will show now, there is a possibility to apply the meta-inductive framework of prediction games to this case.

Since each of these values has two possible truth values, 0 and 1, we can score probabilistic predictions by comparing them with these truth values for each of the possible values. This means in effect that we mimic a prediction game about a random variable with k values $v_1, \ldots, v_k$ by launching k prediction games about k binary events, v_m versus not-v_m, in parallel.

We can define a measure for the predictive success regarding a value v_m as follows:

$$s_{i,t}(v_m) = \frac{\sum_{u=1}^{t} 1 - l(P_{i,u}(v_m), val_u(v_m))}{t}$$

It is reasonable though not mandatory to assume that l is the quadratic loss function ($(P_{i,u}(v_m) - val_u(v_m))^2$), because according to a well-known result of (Brier 1950) the quadratic loss function minimizes the forecaster's expected success if she identifies her predictions with her credences.

The decisive difference of this setting compared to the previous one is that now the success rates of the candidate methods and the meta-inductive algorithm are relative to elements of the value space: Each method has a success rate for each value v_m. Based on this we can define a weight $w_{i,t}(v_m)$ of method i for predicting event value v_m up to time t as follows (where $s_{aggr,t}$ is the per-round success rate of P_{aggr} as defined below):

$$w_{i,t}(v_m) = \frac{max(0, s_{i,t}(v_m) - s_{aggr^*,t}(v_m))}{\sum_{j=1}^{n} max(0, s_{j,t}(v_m) - s_{aggr^*,t}(v_m))}$$

Finally, based on these weights we might define a probabilistic aggregating algorithm as follows:

$$P_{aggr^*,t+1}(v_m) = \sum_{i=1}^{n} w_{i,t}(v_m) \cdot P_{i,t+1}(v_m) \qquad \text{(Aggr*)}$$

It is easy to see that the no-regret optimality result of the foregoing section holds for such a meta-level method for each value v_m of E's value space: The probabilistic aggregating forecasting algorithm P_{aggr^*} will approximate the maximum of the success rates of the best candidate methods accessible in the setting regarding each v_m. However, there is a problem: It can easily happen that the method which is best at a given round depends on the value of the value space. In other words, the meta-inductive forecaster uses weights resulting from different prediction games which can lead to the result that its aggregated probabilities are *incoherent*. To see this, consider the following example:

- Let E be a series of discrete random variables $E_1, E_2, \ldots$.
- $k = 3$, i.e. the value space consists of v_1, v_2, v_3.
- Let $n = 2$, i.e. the accessible candidate methods are P_1 and P_2. Now, let up to round u candidate method P_1 be a perfect expert in predicting v_1 and P_2 be a perfect expert in predicting v_2. Let up to round u P_1 completely fail regarding the predictions of v_2, v_3 and P_2 completely fail regarding predictions of v_1, v_3. Thus for all $t \leq u$: if $val_t(v_1) = 1$, then $P_{1,t}(v_1) = 1$ and $P_{2,t}(v_1) = 0$; and if $val_t(v_2) = 1$, then $P_{2,t}(v_2) = 1$ and $P_{1,t}(v_2) = 0$. Moreover if $val_t(v_3) = 0$ both fail, i.e. $P_{1,t}(v_3) = P_{2,t}(v_3) = 0$.
- So, the candidate predictions are such that their success rates at each round $t \leq u$ (for all convex loss functions without an additive term) are:

	$s_{1,t}(v_i)$	$s_{2,t}(v_i)$
v_1	100%	0%
v_2	0%	100%
v_3	0%	0%

- But then $w_{1,u+1}(v_1) = 1$, thus $P_{aggr^*,u+1}(v_1) = P_{1,u+1}(v_1)$ and $w_{2,u+1}(v_2) = 1$, thus $P_{aggr^*,u+1}(v_2) = P_{2,u+1}(v_2)$. Now assume that at round $u + 1$ both of the candidate methods predict the value they were absolute experts up to round u, i.e. $P_{1,u+1}(v_1) = 1$ and

$P_{2,u+1}(v_2) = 1$. Then the predictions of the algorithm are

$$P_{aggr^*,u+1}(v_1) = 1 \text{ and } P_{aggr^*,u+1}(v_2) = 1$$

which is probabilistically inconsistent.

So, although each individual provides a probabilistic forecast, pooling the forecasts according to this idea ends up with a forecast that is no longer probabilistically consistent. Regarding each value of the value space such a forecast is no-regret optimal, however this optimality comes at cost of consistency.

One can restore consistency by normalising P_{aggr^*}. Here the idea is to still calculate for each candidate method success rates that depend on the method's success regarding a specific value v_m of the value space. These success rates are then, in a second step, used for defining value-dependent weights for each candidate method. And these weights are again, in a third step, used to construct a prediction as above. However, additionally as a fourth step these predictions are normalised in order to guarantee probabilistic consistency. Such a normalised average probability aggregation algorithm (Aggr**) can be defined as follows:

$$P_{aggr^{**},t+1}(v_m) = \frac{P_{aggr^*,t+1}(v_m)}{\sum_{j=1}^{k} P_{aggr^*,t+1}(v_j)} \qquad (\text{Aggr}^{**})$$

A schema of such an implementation is illustrated in figure 1 (more details on this figure see below): Probabilistic forecasts consist no longer of parallel prediction games, but of combining parallel predictions by help of normalisation to a single probabilistic forecast.

By help of an example one can show that the probabilistic aggregating forecasting algorithm is not access optimal with respect to the single values. To see this, consider the following probabilistic prediction game:

- Let us assume that we have three values v_1, v_2, v_3, two forecasters P_1, P_2 and for simplicity reasons let us assume that each of them gives at each round full probability to one of the values. Now, let us assume that the forecasts and the outcome are as follows:

t	1	2	3	4	
P_1	$v_1 : 1.0$	$v_1 : 1.0$	$v_1 : 0.0$	$v_1 : 0.0$	
	$v_2 : 0.0$	$v_2 : 0.0$	$v_2 : 0.0$	$v_2 : 0.0$	
	$v_3 : 0.0$	$v_3 : 0.0$	$v_3 : 1.0$	$v_3 : 1.0$	
P_2	$v_1 : 0.0$	$v_1 : 0.0$	$v_1 : 0.0$	$v_1 : 0.0$	
	$v_2 : 1.0$	$v_2 : 0.0$	$v_2 : 1.0$	$v_2 : 0.0$	
	$v_3 : 0.0$	$v_3 : 1.0$	$v_3 : 0.0$	$v_3 : 1.0$	
val	v_1	v_1	v_2	v_3	

t	5	6	7	8	$\ldots$
P_1	$v_1 : 1.0$	$v_1 : 1.0$	$v_1 : 0.0$	$v_1 : 0.0$	$\ldots$
	$v_2 : 0.0$	$v_2 : 0.0$	$v_2 : 0.0$	$v_2 : 0.0$	$\ldots$
	$v_3 : 0.0$	$v_3 : 0.0$	$v_3 : 1.0$	$v_3 : 1.0$	$\ldots$
P_2	$v_1 : 0.0$	$v_1 : 0.0$	$v_1 : 0.0$	$v_1 : 0.0$	$\ldots$
	$v_2 : 1.0$	$v_2 : 0.0$	$v_2 : 1.0$	$v_2 : 0.0$	$\ldots$
	$v_3 : 0.0$	$v_3 : 1.0$	$v_3 : 0.0$	$v_3 : 1.0$	$\ldots$
val	v_2	v_1	v_2	v_3	$\ldots$

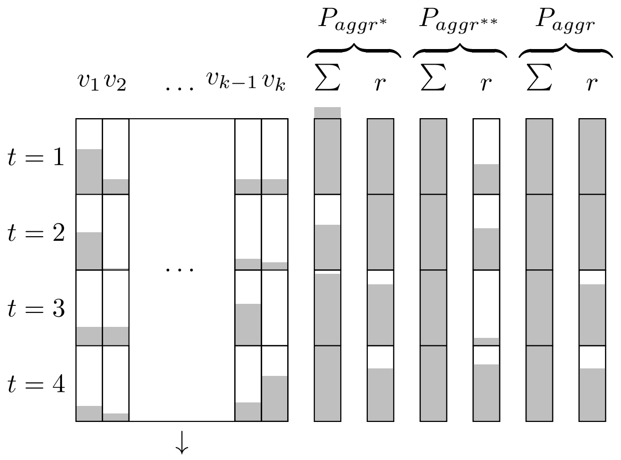

Figure 1: Example of launching k prediction games about single events parallel (P_{aggr^*}), one for each value of the value space. Out of the parallel prediction games a probabilistic forecast about all values is constructed by normalisation ($P_{aggr^{**}}$); P_{aggr} constructs its predictions out of averaging the success-rates among the values. The bars under $\sum$ indicate the sum of the meta-inductive algormithm's probability forecast. The bars under r (regret) indicate proven upper bounds for the regrets. As can be seen, P_{aggr^*}'s regret vanishes in the long run, hower its forecast is probabilistcally inhoherent (does not sum up to 1). $P_{aggr^{**}}$ is probabilistcally coherent through normalisation, however, its average per-round regret does not vanish in the long run. And finally, P_{aggr} has advantages of both worlds: it is probabilistically coherent and no-regret optimal.

- Let us furthermore assume a linear loss function (similar counterexamples are possible with other convex loss functions). Then the success rates will converge to $s_{1,t\to\infty}(v_1) = s_{2,t\to\infty}(v_2) = 7/8$, $s_{1,t\to\infty}(v_2) = s_{2,t\to\infty}(v_1) = 5/8$, $s_{1,t\to\infty}(v_3) = s_{2,t\to\infty}(v_3) = 7/8$. Thus, after some point in time t^*, P_1 will gain full attractivity and weight in predicting v_1, P_2 full attractivity and weight in predicting v_2, and both get equal weight in predicting v_3. Hence, starting at $t^* + 1$ the unnormalised and the normalised predictions of the meta-level agents are:

t	t^*	$t^* + 1$	$t^* + 2$	$t^* + 3$	$\ldots$
P_{aggr^*}	$v_1 : 1.0$	$v_1 : 1.0$	$v_1 : 0.0$	$v_1 : 0.0$	$\ldots$
	$v_2 : 1.0$	$v_2 : 0.0$	$v_2 : 1.0$	$v_2 : 0.0$	$\ldots$
	$v_3 : 0.0$	$v_3 : 0.5$	$v_3 : 0.5$	$v_3 : 1.0$	$\ldots$
$P_{aggr^{**}}$	$v_1 : 0.5$	$v_1 : 0.\overline{66}$	$v_1 : 0.0$	$v_1 : 0.0$	$\ldots$
	$v_2 : 0.5$	$v_2 : 0.0$	$v_2 : 0.\overline{66}$	$v_2 : 0.0$	$\ldots$
	$v_3 : 0.0$	$v_3 : 0.\overline{33}$	$v_3 : 0.\overline{33}$	$v_3 : 1.0$	$\ldots$
val	v_1/v_2	v_1	v_2	v_3	$\ldots$

- But then—given, e.g., the natural loss function—the success rates of $P_{aggr^{**},t\to\infty}$ are: $s_{aggr^{**},t\to\infty}(v_1) = 19/24 < 7/8 = s_{1,t\to\infty}(v_1)$, $s_{aggr^{**},t\to\infty}(v_2) = 19/24 < 7/8 = s_{2,t\to\infty}(v_2)$, and $s_{aggr^{**},t\to\infty}(v_3) = 10/12 < 7/8 = s_{1,t\to\infty}(v_3) = s_{2,t\to\infty}(v_3)$.

- Hence, regarding all three values $P_{aggr^{**}}$ is no-regret *sub*optimal.

As the examples above show, one cannot have both, consistency and optimality with respect to each value of the value space. However we construct a probabilistic aggregation method, call it P_{aggr}, that is both coherent and no-regret optimal. We can do so simply by averaging the success-rates for the individual values of the value space.

To recognise this possibility, we just have to hint to the mathematical fact that if the loss function l is convex with respect to all values of the value space, then also averaging among the losses with respect to all values of the value space is convex. Let us first define such an average loss measure l_{av}:

$$l^{av}_{i,t} = \frac{\sum_{m=1}^{k} l(P_{i,t}(v_m), val_t(v_m))}{k}$$

Note that if l is the quadratic loss function, then l^{av} is the Brier score for a particular round (Brier 1950). The Brier score can be calculated then by summing up all the scores up to round t and dividing them by t (that is the average per round loss averaged over all values of the value space).

Now, since we assumed that l is convex, also l^{av} is convex. So, we can define a measure for average success s^{av} which is not relativised to a single value of the value space:

$$s^{av}_{i,t} = \frac{\sum_{u=1}^{t} 1 - l^{av}_{i,t}}{t}$$

Based on these per-round average success rate we can define average success-based weights w^{av} that are also not relativised to a single value of the value space (where $s^{av}_{aggr,t}$ is the per-round success rate of the aggregation algorithm P_{aggr} as defined below):

$$w^{av}_{i,t} = \frac{max(0, s^{av}_{i,t} - s^{av}_{aggr,t})}{\sum_{j=1}^{n} max(0, s^{av}_{j,t} - s^{av}_{aggr,t})}$$

Now, we can define the meta-inductive algorithm for weighted average probability aggregation (Aggr) based on these weights in accordance with (WM):

$$P_{aggr,t+1} = \sum_{i=1}^{n} w^{av}_{i,t} \cdot P_{i,t+1} \qquad \text{(Aggr)}$$

Since (Aggr) is an instance of (WM) and since l^{av} used to determine the weights w^{av} is convex, given the underlying loss function l is convex, we can transfer the no-regregt/optimality result of P_a to P_{aggr} in a straightforward way:

Theorem. *Given the loss function l is convex it holds:*

$$s^{av}_{i,t} - s^{av}_{aggr,t} \leq \sqrt{n/t} \quad \forall i \in \{1, \dots, n\}$$

So, Pr_{aggr} is a no-regret algorithm for aggregating probabilities:

$$\lim_{t\to\infty} max(s^{av}_{1,t}, \dots, s^{av}_{n,t}) - s^{av}_{aggr,t} \;\leq\; 0$$

To illustrate this fact we can come back to the last example on the sub-optimality of P_{aggr} regarding each value of the value space: Here it was the case that the candidate method P_1 was better than P_{aggr} regarding v_1 and v_3, P_2 was better than P_{aggr} regarding v_2 and v_3. However, as calculating the average success-rates demonstrates as an instance of our general result above, both of them are not better than P_{aggr} in averaging over their per-round success-rates regarding all values of the value space v_1, v_2, v_3: $s^{av}_{aggr,t\to\infty} \geq s^{av}_{1,t\to\infty}$ and $s^{av}_{aggr,t\to\infty} \geq s^{av}_{2,t\to\infty}$.

Conclusion

In this paper we have argued for a new solution to the problem of weighted probability aggregation. We have seen that some general constraints determine families of aggregation rules like linear aggregation rules. In order to address the problem of specifying such rules by fixing weights we have argued for a success-based calculation of weights as is suggested also in the literature on scoring probabilistic forecasts (cf. Genest and McConway 1990, pp.57ff). As we have shown, such an approach can be justified by help of results on predictions under expert advice since a success-based calculation of weights allows for no-regret optimal probabilistic aggregation.

References

Arrow, Kenneth Joseph (1963). *Social Choice and Individual Values*. 2nd Edition. Yale: Yale University Press.

Brier, Glenn W. (1950). “Verification of Forecasts Expressed in Terms of Probability”. In: *Monthly Weather Review* 78.1, pp. 1–3.

Cesa-Bianchi, Nicolo and Gabor Lugosi (2006). *Prediction, Learning, and Games*. Cambridge: Cambridge University Press.

Genest, Christian and Kevin J. McConway (1990). “Allocating the Weights in the Linear Opinion Pool”. In: *Journal of Forecasting* 9.1, pp. 53–73. URL: http://dx.doi.org/10.1002/for.3980090106.

Genest, Christian and James V. Zidek (1986). “Combining Probability Distributions: A Critique and an Annotated Bibliography”. In: *Statistical Sciences* 1.1, pp. 114–135.

Grossi, Davide and Gabriella Pigozzi (2014). *Judgment Aggregation: A Primer*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Williston: Morgan & Claypool.

Kornhauser, Lewis A. and Lawrence G. Sager (1986). “Unpacking the Court”. In: *The Yale Law Journal* 96.1, pp. 82–117. URL: http://www.jstor.org/stable/796436.

Lehrer, Keith and Carl Wagner (1981). *Rational Consesus in Science and Society. A Philosophical and Mathematical Study*. Dordrecht: Reidel Publishing Company.

List, Christian and Philip Pettit (2002). “Aggregating Sets of Judgments: An Impossibility Result”. In: *Economics and Philosophy* 18.01, pp. 89–110.

Rossi, Francesca, Kristen Brent Venable, and Toby Walsh (2011). *A Short Introduction to Preferences. Between Artificial Intelligence and Social Choice*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Williston: Morgan & Claypool.

Schurz, Gerhard (2008). "The Meta-Inductivist's Winning Strategy in the Prediction Game: A New Approach to Hume's Problem". In: *Philosophy of Science* 75.3, pp. 278–305.

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