Active Elicitation of Imprecise Probability Models

Natan T'Joens, Gert de Cooman, Jasper De Bock, Arthur Van Camp
Data Science Lab, Ghent University, Belgium
{Natan.Tjoens,Gert.DeCooman,Jasper.DeBock,Arthur.VanCamp}@UGent.be

GHENT UNIVERSITY

Introduction

Learning a model We consider the problem of eliciting an imprecise probability model from a domain expert. We will do this by asking specific questions about his beliefs. Answers to these questions will impose *inequality constraints* on the elements of the credal set $\mathscr{M}$ (set of probability mass functions) that models his beliefs, e.g. $P(A) < P(B)$ for all $P \in \mathscr{M}$.

The goal One can distinguish between two different, yet related goals. The first is to construct an imprecise probability model that *captures the expert's beliefs as completely as possible*. The second is to gather information that is aimed specifically *at answering a given question or solving a given decision problem*. We consider the latter, where we have to determine an optimal action among a set of possible actions.

Criteria We consider two important criteria that have to be satisfied.

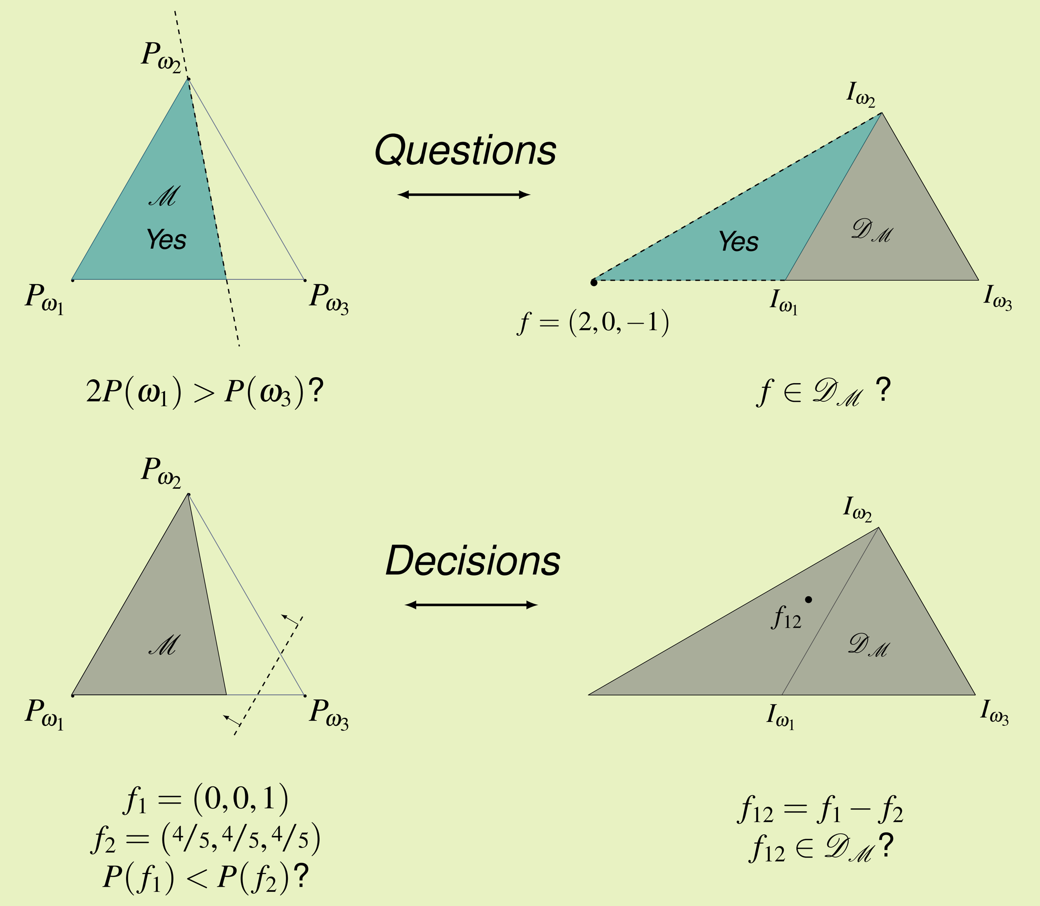
1. *We have to limit ourselves to intuitive questions*

$$\underset{\text{intuitive}}{p(A) < p(B)?} \quad \longleftrightarrow \quad \underset{\text{less intuitive}}{2p(A) + 5p(B) - 7p(C) < 0?}$$

2. *We want to minimise the number of questions we have to ask. This is the optimisation criterion.*

Elicitation in sets of desirable gambles

Sets of desirable gambles allow for more elegant modelling of the elicitation process compared to credal sets or lower/upper previsions. Furthermore, the use of this framework *does not induce any limitations* on the practical feasibility of the problems.



Simplified framework

We assume the expert can only find a gamble f *desirable* or leave it *unresolved*. When f is stated to be desirable, we extend $\mathscr{D}$ with f and consider the natural extension of this set: $\mathscr{D}_{\text{new}} = \mathscr{E}(\mathscr{D}_{\text{old}} \cup f)$. When f is left unresolved, $\mathscr{D}$ is left unchanged. We now search for the minimal sets of questions such that, when these questions are positively answered, we are able to determine a unique optimal action.

$F = \{f_1, f_2, ..., f_p\} \subset \mathscr{L}(\Omega)$ *the set of actions*
$V = \{v_1, v_2, ..., v_m\} \subset \mathscr{L}(\Omega)$ *the set of questions*
$D = \{d_1, d_2, ..., d_a\} \subset \mathscr{L}(\Omega)$ *the initial assessment*

find: *the minimal elements* $V_i \subseteq V$
such that: $\mathscr{D}_i = \mathrm{posi}\,(\mathscr{L}^+(\Omega) \cup D \cup V_i)$ *is coherent and*
$(\exists f^* \in F)(\forall f \in F \setminus \{f^*\})\ f^* - f \in \mathscr{D}_i$.

$|F| > 2$

Algorithm The approach is the same as for the problem where $|F| = 2$. For all $f, g \in F$ such that $f \neq g$ and $(f - g) \notin \mathscr{D}_j$ for all $\mathscr{D}_j \subset \mathscr{D}_i$, we check if $(f - g) \in \mathscr{D}_i$. We use the same working principle as for the basic problem where $|F| = 2$.

Complexity In the worst case, we would have to check $p(p-1)$ *gambles for inclusion in* $\mathscr{D}_i$, where $p := |F|$. We cannot use the Carathéodory Theorem in the same way as before. The computational complexity will remain *exponential* in $|V|$

$|F| = 2$

In a first approach we consider the case where we can choose between *two actions*, so $F = \{f_1, f_2\}$. The algorithm that solves this problem consists of three similar steps explained below, where in each step we use the following linear feasibility problem for a different A:

find: $\lambda \in \mathbb{R}^A$
subject to: $\sum_{g \in A} \lambda_g g = 0$ *and* $\lambda \geq 0$ *and* $\sum_{g \in A} \lambda_g \geq 1$

1. Checking whether $\mathscr{D}_i$ is *coherent*. This is done by $A = D_i = \{I_\omega : \omega \in \Omega\} \cup D \cup V_i$.
   When the problem is feasible, V_i leads to an incoherent situation, so V_i and all its supersets are removed from the search space. When the problem is not feasible, go to step 2.
2. Checking whether f *is included* in $\mathscr{D}_i$. This is done by $A = D_i \cup \{-f\}$.
   When the problem is feasible, V_i is a solution. Remove all supersets of V_i from the search space as these will not be minimal. When the problem is not feasible, go to step 3.
3. Checking whether $-f$ *is included* in $\mathscr{D}_i$. This is done by $A = D_i \cup \{f\}$.
   When the problem is feasible, V_i is a solution. Remove all supersets of V_i from the search space as these will not be minimal. When the problem is not feasible, remove this V_i from the search space and go on.

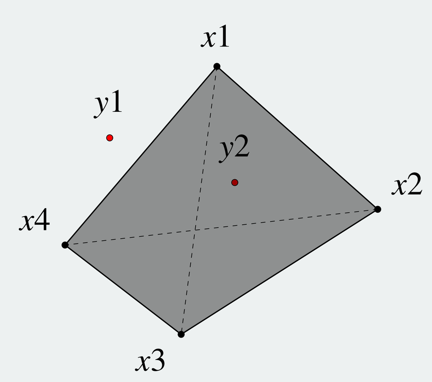
We start with the $V_i \in V$ with $|V_i| = 1$, then $|V_i| = 2$, and so on.

Optimisation

The initial search space is the power set of V. We use Carathéodory's Theorem to reduce it to a search space that is polynomial in the cardinality of V.

Carathéodory's Theorem *For all $x \in \mathbb{R}^n \setminus \{0\}$ and $S \subset \mathbb{R}^n$:*

$$x \in \mathrm{posi}\,(S) \Leftrightarrow (\exists S' \subseteq S)\big(|S'| \leq n \text{ and } x \in \mathrm{posi}\,(S')\big)$$



Example in 3 dimensions
y_2 belongs to the cones generated by $\{x_1, x_2, x_3\}$ and $\{x_1, x_2, x_4\}$ so it lies within the general cone. y_1 does not lie within the general cone, as it does not belong to the cones generated by $\{x_1, x_2, x_3\}$, $\{x_1, x_2, x_4\}$, $\{x_1, x_3, x_4\}$ or $\{x_2, x_3, x_4\}$.

Broader framework

Here we assume the expert can also find a gamble f *undesirable*, meaning he considers the opposite gamble $-f$ to be *desirable*. This is a simplified version of the 'accept & reject statement-based framework' (Quaeghebeur et al., 2015). Every question can now lead to *two* new situations — depending on the answer — whereas in the simplified framework we only had to consider *one* new situation. As we do not know the expert's answers in advance, determining a general optimal question is difficult and compulationally heavy. To limit the complexity of the solving algorithms, we used the following two approaches. We consider $|F| = 2$.

Heuristic decision rules e.g. minimising the uncertainty towards the decision gamble $f = f_1 - f_2$ for the worst case answer on the considered question.

$$v^*_{\max(\Delta)} \in \operatorname*{argmin}_{v \in V} \left[\max_{i \in \{v, -v\}} \left(\overline{P}_{\mathscr{D}'_i}(f) - \underline{P}_{\mathscr{D}'_i}(f) \right) \right] \text{ with } \mathscr{D}'_i := \mathrm{posi}\,(\mathscr{D} \cup \{i\}),$$

and $\mathscr{D}$ the current set of desirable gambles.

Semi-heuristic a hybrid algorithm that combines a heuristic decision rule with the determinative aspect of the algorithm used to solve the problem in the simplified framework. The computational complexity of the algorithm is polynomial in the size of V.

Simulations

We have tested the performance of these algorithms using simulations. A simulation consists in a large number of runs. In each run the process is modelled of asking and answering questions until a decision can be made. The questions are chosen by the mentioned algorithms from a given set of possible questions. This set is changed after each run. The answers are delivered by a fixed belief model that is hidden from the algorithm. We present the most significant results.