Lectures on Soft Computing and Fuzzy Logic, (2001) Springer-Verlag, 113-127.

Extension principle and probabilistic inferential process.

D. Calabrò, G. Gerla and L. Scarpati
D.M.I. University of Salerno, Via S. Allende, 84084 Baronissi (SA), Italy

1. Introduction.

F. Bacchus in [1] and J. Y. Halpern in [6] compare the propositions:
(i) *"a bird will fly";*
(ii) *"Tweety* (*a particular bird*) *is able to fly"*
and they underline their complete difference when we want to assign them a probability valuation. In fact, if we say that the probability of (i) is, for example, 0.9, this valuation arises from our past experience on the birds. Indeed it coincides with a statistical information about the proportion of fliers among the whole set of birds. The problem is to assign a probability to the second proposition. Indeed a probabilistic valuation seems possible only if we have a collection of elements while Tweety is a particular bird. As a matter of fact one expects only two possibilities:

- Tweety is able to fly and so (ii) is true
- Tweety is not able to fly and so (ii) is false.

G. Gerla in [3] suggests to interpret (ii) as:

"A bird with the same observable properties of Tweety is able to fly".

In accordance, when we assign to (ii) the probability 0.9 we mean that the ninety percent of all birds with the same observable properties of Tweety is able to fly. This idea was born by the conviction that our "degrees of belief" derive from the experience each of us stored in his memory about a class of past cases (the birds) we consider similar to the actual case a_c (Tweety) under consideration. So, given a property α, if we want to know the probability that a_c verifies α, we have:

1. to consider the observable (relevant) properties satisfied by a_c;
2. to consider the set of past cases *similar* to a_c, i.e., the cases satisfying the same properties of a_c (Boolean valuation);
3. to determine the probability of α as the percentage of past cases similar to a_c satisfying α (numerical valuation).

Notice that two notions are on the basis of such an idea. The *"similarity*" relation which is intended as "satisfying the same observable relevant properties" and the notion of *"Boolean valuation"* that is necessary because, as it is well known, the probability valuations are not truth functional.

In this work we start from this idea to sketch out a method to design expert systems, probabilistic in nature. The inferential engine we propose is a data-base storing information about a set of "past cases". The inferential process consists in a querying strategy to investigate about the main observable properties of the "actual case" a_c. The resulting information enables us to isolate the past cases which are similar to a_c to give a probabilistic valuation of a "non-observable" property of a_c.

We also consider the possibility that the information about the past cases is incomplete. To this purpose we use a simple logic, Boolean in nature, we obtain by the extension principle proposed in [2] and [5]. Due to the incompleteness of the information, the resulting valuations are not probabilities but super-additive measures, in general.

Finally, in spite of the theoretical nature of the paper, in order to taste the potentialities of the proposed notions, a prototype shell for expert systems was built up. The relational data-base *Access* is the used languages. Also, *Visual Basic* is used for the interface.

2. Probability and Boolean valuations.

Recall some elementary notions that are on the basis of any approach to probability logic. In the following we denote by F the set of formulas of a zero-order language.

Definition 2.1. A *probability valuation* of F is any map $\mu : F \to [0,1]$ such that:

a) $\mu(\alpha) = 1$ for every tautology α;
b) $\mu(\alpha\vee\beta) = \mu(\alpha) + \mu(\beta)$ if $\alpha\wedge\beta$ is a contradiction;
c) $\mu(\alpha) = \mu(\beta)$ if α is logically equivalent to β.

Observe that if μ is a probability valuation, then $\mu(\alpha) = 0$ for every contradiction α. Indeed, in such a case, since α is logically equivalent to $\alpha\vee\alpha$ and $\alpha\wedge\alpha$ is a contradiction, by b) and c), we have that

$$\mu(\alpha) = \mu(\alpha\vee\alpha) = \mu(\alpha) + \mu(\alpha).$$

This entails that $\mu(\alpha) = 0$. As it is well known, the probability valuations are not truth-functional. In fact, the knowledge of the probability of two formulas α and β doesn't allow to determine the probability of the composed formula $\alpha\wedge\beta$, in general. This is a strong obstacle for a probability logic. Nevertheless, the truth-functionality can be obtained by the notion of Boolean truth-functional valuation, in a sense.

Definition 2.2. Let B be a Boolean algebra. We say that a map $v : F \to B$ is a *truth-functional B-valuation* if the following properties hold for every $x,y \in F$,

a) $v(x\wedge y) = v(x)\wedge v(y)$;
b) $v(x\vee y) = v(x)\vee v(y)$;
c) $v(-x) = -v(x)$.

Observe that a), b) and c) entail that $v(1) = 1$ and $v(0) = 0$. Indeed

$$v(1) = v(-x\vee x) = v(x)\vee -v(x) = 1$$

and

$$v(0) = v(x\wedge -x) = v(x)\wedge -v(x) = 0.$$

If $B = \{0,1\}$, the truth-functional B-valuations coincide with the usual classical interpretations of F.

The more interesting case is when B coincides with the class $\mathcal{P}(S)$ of all the subsets of a set S. Indeed, in such a case we can interpret S as the set of past cases stored in a data-base or as the set of *"possible worlds"* in a Kripke semantics. In accordance, given any sentence α, we can interpret $v(\alpha)$ as the set of past cases (possible worlds) in which α is true.

Definition 2.3. We call *B-probability valuation* a structure (B, v, p) where B is a Boolean algebra, $v : F \to B$ is a truth-functional B-valuation and $p : B \to [0,1]$ is a finitely additive probability.

The following proposition shows that the notion of B-probability valuation is strictly related with the notion of probability valuation.

Proposition 2.1. Let (B,v ,p) be a B-probability valuation and define $\mu : F \to [0,1]$ by setting $\mu(\alpha) = p(v(\alpha))$ for every $\alpha \in F$. Then μ is a probability valuation that we call *associated* with (B, v, p). Conversely, let $\mu : F \to [0,1]$ be any probability valuation in F. Then a Boolean algebra B and a B-probability valuation (B, v, p) exist such that $\mu(\alpha) = p(v(\alpha))$.

Proof. The first part of the proposition is obvious. Let $\mu : F \to [0,1]$ be a probability valuation in F and denote by B the Lindenbaum algebra associated with F. This means that we set, for any formula α,

$$[\alpha] = \{\alpha' \in F : \alpha' \text{ is logically equivalent to } \alpha\}$$

and

$$B = \{[\alpha] : \alpha \in F\}$$

Moreover, for any $[\alpha]$ and $[\beta]$ in B,

$$[\alpha]\wedge[\beta] = [\alpha\wedge\beta], \quad [\alpha]\vee[\beta] = [\alpha\vee\beta], \; -[\alpha] = [-\alpha].$$

Define the function $v : F \to B$ by setting $v(\alpha) = [\alpha]$ for every $\alpha \in F$ and define $p : B \to [0,1]$ by setting $p([\alpha]) = \mu(\alpha)$. Then it is immediate that (B, v, p) is a B-probability valuation whose associated probability valuation is μ. □

3. Knowledge representation system.

The starting point of the inferential process we will define is a data base storing the information about a series of past cases we consider related with the actual case. To represent this, we propose a formalism very near to the formalism proposed by Pawlak in [7]. The first distinction we have to do is between the observable properties and the not observable ones. We call *observable* the properties for which it is possible to detect in a direct way whether they are satisfied or not by the actual case. A property that will be materialized in the future is a typical example of non observable property. Now we can give the following definition.

Definition 3.1. A (*complete*) *knowledge representation system* is a structure $\mathcal{S}$ = (PC, AT, OBS, tr) where:

- PC is a finite set whose elements we call *past cases*;
- AT is a finite set whose elements we call *attributes*;
- OBS is a subset of AT, called the set of the *observable* attributes;
- $tr : PC \times AT \to \{0,1\}$ is a function, called *information function*.

We denote by F (by F_{obs}) the set of formulas of the propositional calculus whose set of propositional variables is AT (is OBS, respectively). Obviously, tr can be extended to the whole set F of formulas by setting

$tr(c,\alpha\wedge\beta) = min\{tr(c,\alpha),tr(c,\beta)\}$,

$tr(c,\alpha\vee\beta) = max\{tr(c,\alpha),tr(c,\beta)\}$,

$tr(c,-\alpha) = 1 - tr(c,\alpha)$.

Given a formula α and a case c, the equation $tr(c,\alpha) = 1$ means that α is true in c, while $tr(c,\alpha) = 0$ means that α is false in c. In other words, tr associates any past case with a classical valuation of the formulas in F. Given a set T of formulas, we say that a past case c is a *model* of T, and we write $c \vDash T$, if $tr(c,\alpha) = 1$ for every $\alpha \in T$.

Now, PC, as the result of the whole past experience, is a too big basis of our inferential process, in general. A more workable tool can be obtained by observing that from the point of view of the inferential apparatus we will define, it is not useful to distinguish two past cases satisfying the same properties. Then, we define an equivalence relation on the set of past cases in the following way.

Definition 3.2. Let $\mathcal{S} = (PC, AT, OBS, tr)$ be a knowledge representation system and A a set of formulas. Then we define a binary relation $\cong_A$ in PC by setting

$$c_1 \cong_A c_2 \quad \text{if and only if} \quad tr(c_1, \alpha) = tr(c_2, \alpha) \text{ for every } \alpha \in A.$$

If $c_1 \cong_A c_2$, we say that c_1 and c_2 are *A-indiscernible*. Then two cases are A-indiscernible if they satisfy the same properties in A. The proof of the following proposition is obvious.

Proposition 3.1. Given any set A of formulas, $\cong_A$ is an equivalence relation. Moreover, if A and B are set of formulas,

$$A \subseteq B \;\Rightarrow\; \cong_A \supseteq \cong_B.$$

In accordance with Proposition 3.1, given a set A of formulas and a case c, we can consider the relative equivalence class

$$[c]_A = \{c' \in PC : c' \cong_A c\}.$$

Also, we can define the quotient

$$PC_A = \{[c]_A : c \in PC\}.$$

In the following proposition, given a set A of formulas, we denote by $\mathcal{L}(A)$ the language generated by A, i.e. the set of formulas obtained from A by an iterated application of the disjunction, conjunction and negation operations.

Proposition 3.2. The relation $\cong_{\mathcal{L}(A)}$ coincides with $\cong_A$. In particular, $\cong_F$ coincides with $\cong_{AT}$. Moreover, $|PC_A| \leq 2^{|A|}$ and, in particular,

$$|PC_F| = |PC_{AT}| \leq 2^{|AT|}.$$

Proof. Obvious.

We write $c_1 \cong c_2$ instead of $c_1 \cong_F c_2$ and in this case we say that c_1 and c_2 are *indiscernible*. Moreover, we write $[c]$ to denote $[c]_F$.

At this point it is natural to consider in spite of $\mathcal{S}$ the related quotient $\mathcal{S}^* = (PC_F, AT, OBS, tr^*)$ where tr^* is defined by setting, for every $c \in PC_F$,

$$tr^*([c], \alpha) = tr(c, \alpha).$$

Obviously, since $|PC_F| = |PC_{AT}| \leq 2^{|AT|}$ the cardinality of $\mathcal{S}$ is not too big. Also, in order to preserve the whole information of the initial knowledge representation system we have to store, for each type-case $[c]$, the number of elements contained in $[c]$. This leads to the following definition.

Definition 3.3. *A* (*complete*) *statistical inferential basis* is a structure $\mathcal{S} = (TC, AT, OBS, tr, w)$ such that:
- (TC, AT, OBS, tr) is a (complete) knowledge representation system such that two elements of TC are always discernible;
- $w : TC \to N$ is a function called *weight function*.

The elements of TC are called *type-cases*.

In a sense, a statistical inferential basis is obtained by an abstraction process from our past experience. For every type-case t, $w(t)$ can be seen as the number of concrete past cases represented by t. It is immediate that $|TC| \leq 2^{|AT|}$. We call *total weight* of $\mathcal{S}$ the number of the past cases represented globally by $\mathcal{S}$, that is

$$w(\mathcal{S})=\Sigma\{w(c) : c\in TC\}.$$

Proposition 3.3. Every statistical inferential basis $\mathcal{S}$ defines a B-probability valuation (B,v,p) in F such that:
- B is the Boolean algebra $\mathcal{P}(TC)$;
- $v(\alpha) = \{c\in TC : tr(c,\alpha) = 1\}$ is the set of type-cases satisfying α;
- $p : B \to [0,1]$ is the probability defined by setting, for any set X of cases,

$$p(X) = \frac{\sum\{w(c) : c \in X\}}{w(TC)}.$$

We call *entropy* of $\mathcal{S}$ the entropy of the probability p, i.e. the number

$$E(\mathcal{S}) = -\sum\{p(c)lg(p(c)) : c\in TC\}.$$

In accordance with Proposition 2.1, we can associate any statistical inferential basis $\mathcal{S}$ with a probability valuation μ of the formulas. It is evident that, for every formula α,

$$\mu(\alpha) = \frac{\sum\{w(c) : tr(c,\alpha) = 1\}}{w(TC)}.$$

In other words, $\mu(\alpha)$ represents the percentage of past cases in which α is true according to the initial dates.

4. Inferential process

Imagine that we have to evaluate the probability that an actual case a_c satisfies a formula α in F. Here we call *actual case* any model of the language F and we denote by $T(a_c)$ the set of observable formulas satisfied by a_c. Obviously, we are interested to the case that α is not observable. Then, we can imagine a step-by-step inferential process resulting from a queering strategy. At step i we obtain a formula α_i as an answer to a "query" about the observable properties of the actual case a_c. After n steps, the information about our actual case is collected in a set $T = \{\alpha_1,\ldots,\alpha_n\} \subseteq T(a_c)$ of formulas. Given a statistical inferential basis $\mathcal{S}$ (basic information), we say that T is *satisfiable* in $\mathcal{S}$ if a typical case exists which satisfies T. The theory T enables us to obtain a new statistical inferential basis $\mathcal{S}(T)$ from $\mathcal{S}$. Namely, we set

$$TC(T) = \{c \in TC : c \models T\},$$

i.e., $TC(T)$ is the set of cases c satisfying T. Equivalently, $TC(T)$ is the set of past cases T-indiscernible from the actual case. Then, we can propose the following definition:

Definition 4.1. Let $\mathcal{S} = (TC, AT, OBS, tr, w)$ be a statistical inferential basis and $T \subseteq F_{obs}$ a theory satisfiable in $\mathcal{S}$. We call *statistical inferential basis defined by T in S* the structure:

$$\mathcal{S}(T) = (TC(T), AT, OBS, tr, w).$$

In accordance with Proposition 3.3, $\mathcal{S}(T)$ defines a B-probability valuation $(\mathcal{P}(TC(T)), v_T, p_T)$ where:

$$v_T(\alpha) = \{c \in TC : c \models T \cup \{\alpha\}\} = TC(T) \cap v(\alpha)$$

and, for any subset X of $TC(T)$,

$$p_T(X) = \frac{\sum\{w(c) : c \in X\}}{w(TC(T))}.$$

We can extend p_T to the whole algebra $\mathcal{P}(TC)$ by setting $p_T(\{c\}) = 0$ for any $c \notin TC(T)$. Notice that in such a way we obtain the conditional probability $p(\_/TC(T))$. In fact, for any $X \subseteq TC$,

$$p_T(X) = \frac{\sum\{w(c) : c \in X \cap TC(T)\}}{w(TC(T))} = \frac{\sum\{w(c) : c \in X \cap TC(T)\}}{\sum\{w(c) : c \in TC\}} \cdot \frac{\sum\{w(c) : c \in TC\}}{w(TC(T))}$$
$$= \frac{p(X \cap TC(T))}{p(TC(T))}.$$

Also, a probability valuation μ_T of the formulas is defined in such a way that

$$\mu_T(\alpha) = \frac{\sum\{w(c) : c \text{ satisfies } \alpha \text{ and } T)}{\sum\{w(c) : c \text{ satisfies } T\}}, \tag{4.1}$$

i.e, $\mu_T(\alpha)$ is the percentage of the past cases verifying α among the cases verifying T.

Now, we are able to give the main definition in this paper.

Definition 4.2. Let α be a formula in F, and $T \subseteq T(a_c)$ the available information on the actual case a_c. Then we call *probability* that a_c satisfies α given T, the probability of α in the statistical inferential basis associated to T.

In conclusion, we imagine an expert system whose inferential engine contains a statistical inferential basis $\mathcal{S}$ obtained by an abstraction process from a knowledge representation system. Given an actual case a_c and a formula α in the language F, the expert system furnishes a probabilistic valuation $\mu(\alpha)$ of α by a step-by-step process as follows:

1. Set $T_0 = \emptyset$ and $\mathcal{S}_0 = \mathcal{S}(\emptyset) = \mathcal{S}$.
2. Given T_i and $\mathcal{S}_i$, put $T_{i+1}=T_i\cup\{\alpha_{i+1}\}$ and $\mathcal{S}_{i+1} = \mathcal{S}(T_{i+1})$, where α_{i+1} as the answer to a query β about the actual case a_c, i.e. $\alpha_{i+1}=\beta$ if the answer is positive, and $\alpha_{i+1}=\neg\beta$ if the answer is negative.
3. If the information is sufficient, goto 4, otherwise goto 2.
4. Set $\mu(\alpha) = \mu_{T_i}(\alpha)$ as defined by (4.1).

In the prototype, β is selected in order to minimize the expected value of the entropy. This is achieved by minimizing the value $|\mu(\beta)-\mu(\neg\beta)|$ where μ is the valuation related to $\mathcal{S}_i$.

Notice that in such a way we gives a precise meaning to the claim:

“ The probability that the actual case a_c satisfies a property α is given by the percentage of the cases indiscernible from a_c that in the past verified α”.

5. An extension principle for incomplete information

The inferential process in the previous section is related to the case of complete information about the past cases stored in the memory. This means that for every case c and α propositional variable, either $tr(c,\alpha)$ is equal to 0 or $tr(c,\alpha)$ is equal to 0. Assume that available information about the truth of the formulas is not complete for some past cases. This means that at least a past case c and an attribute α exist such that we are not able to say whether α is true in c or not. The question arises whether we can propose valuations probabilistic in nature in such a case, too. To deal with such a incomplete information, we need to recall some basic definitions of fuzzy set theory and an extension principle for closure operators proposed in [2] and [5].

Let $L = (L, \wedge, \vee, 0, 1)$ be a complete lattice and S a set. We call *L-fuzzy subset*, or *L-subset* of S any map s from S into L and we denote by L^S the class of L-subsets of S. Usually, L coincides with the lattice [0,1] but in this paper we are interested to the case in which L is a complete Boolean algebra. Given x in S, the value $s(x)$ is called *degree of membership* of x in s. It is immediate that L^S is a complete lattice whose operations are pointwise defined. Then, by using the set-theoretical notations, we have that the *union* of two fuzzy subsets s and s' is defined by setting, for any x in S,

$$(s\cup s')(x) = s(x)\vee s'(x).$$

The *intersection* is defined by setting, for any x in S,

$$(s\cap s')(x) = s(x)\vee s'(x).$$

Moreover, we define the *inclusion* relation by setting

$$s\subseteq s' \Leftrightarrow s(x)\leq s'(x) \text{ for every } x\in S.$$

Given an L-subset s of S and $X\subseteq S$, we define $Incl(X,s)$ by setting

$$Incl(X,s) = Inf\{s(x) : x\in X\}.$$

In particular, we have that $Incl(\emptyset,X) = 1$. The number $Incl(X,s)$ is a multivalued valuation of the statement

“for every $x\in X$, x is an element of s”,

i.e. a measure of the degree of inclusion of X in s. Observe that, by identifying any subset X of S with the related characteristic function, we can consider $\mathcal{P}(S)$ as a sublattice of $\mathcal{F}(S)$. Then any notion in fuzzy set theory have to be an extension of the corresponding notion in classical set theory. For example, notice that if s is the characteristic function of a subset Y of S then

$$Incl(X,s) = 1 \Leftrightarrow \text{ if } X \subseteq Y.$$

We introduce the notion of fuzzy logic in an abstract way by following Tarski point of view. Recall that, given a set F whose elements we call *formulas*, we can define an *abstract crisp logic* as any compact closure operator D in the lattice $\mathcal{P}(F)$. The intended meaning of D is that, given any set X of formulas (the set of axioms), $D(X)$ is the set $\{\alpha \in F : X \vdash \alpha\}$ of logical consequences of X. Recall that a *closure operator* in a set S is any map $J : \mathcal{P}(S) \rightarrow \mathcal{P}(S)$ such that, for any X and Y in $\mathcal{P}(S)$,

i) $X \leq Y \Rightarrow J(X) \leq J(Y)$,

ii) $X \leq J(X)$,

iii) $J(J(X)) = J(X)$.

J is called *compact* provided that

$$J(X) = \cup\{J(X_f) : X_f \text{ is a finite subset of } X\}$$

for any subset X of S.

Now, to obtain a suitable definition of abstract fuzzy logic, we extend the definition of closure operator to any ordered set. Indeed, given an ordered set $(G,\leq)$, we can call *closure operator* any map $J : G \rightarrow G$ such that, for x and y in G,

j) $x \leq y \Rightarrow J(x) \leq J(y)$,

jj) $x \leq J(x)$,

jjj) $J(J(x)) = J(x)$.

In place of the notion of compactness, we consider the notion of continuity. We call *directed* any family $(x_i)_{i\in I}$ of elements in G such that for every i and $j \in I$, there is $h \in I$ such that $x_i \leq x_h$ and $x_j \leq x_h$. If $(G,\leq)$ is complete, we say that J is *continuous* if

$$J(\cup_{i\in I} x_i) = \cup_{i\in I} J(x_i)$$

for every directed family $(x_i)_{i\in I}$ of elements in G. In the case $G = \mathcal{P}(S)$, the continuous closure operators coincides with the compact closure operators. The notion of an abstract fuzzy logic is obtained by substituting $\mathcal{P}(F)$ with the lattice of all the L-subsets of F.

Definition 5.1. An *abstract L-logic* as any continuous closure operator $D : L^F \rightarrow L^F$ is the lattice of all L-subsets of F.

If $v : F \rightarrow L$ is any L-subset of formulas (the L-subset of axioms), then we say that $D(v)$ is the *theory generated by v*.

A simple fuzzy logic can be obtained by the following extension principle enabling to extend any operator $J : \mathcal{P}(S) \rightarrow \mathcal{P}(S)$ into an operator $D^* : L^S \rightarrow L^S$.

Definition 5.2. Let $J : \mathcal{P}(S) \rightarrow \mathcal{P}(S)$ be an operator. Then we call *canonical extension* of J the operator $J^*: L^S \rightarrow L^S$ defined by setting:

$$J^*(s)(x) = Sup\{Incl(X_f,s) : X_f \text{ is a finite subset of } S \text{ and } x \in J(X_f)\}.$$

In particular, we can apply such a principle to the abstract crisp logics, by obtaining an abstract L-logic.

Proposition 5.1. *Let $D : \mathcal{P}(F) \to \mathcal{P}(F)$ be an abstract crisp logic and $D^* : L^F \to L^F$ the related canonical extension. Then D^** is an abstract L-logic (see [2] and [5]).

Trivially, if $v : F \to L$ is any L-subset of formulas, then the L-subset $D^*(v)$ of consequences of v is defined by setting

$$D^*(v)(\alpha) = Sup\{Incl(X_f,v) : X_f \text{ is a finite set of formulas such that } X_f \vdash \alpha\}$$

where we write $X_f \vdash \alpha$ to denote that $\alpha \in D(X_f)$.

In this paper we are interested to the case in which D is the deduction operator of the classical propositional calculus. We say that v is *consistent* if $D^*(v)(\alpha\wedge\neg\alpha) = 0$ for every α formula α .

Proposition 5.2. Let D be the deduction operator of the classical propositional calculus and let v be a consistent L-subset of formulas. Then, for every $\alpha, \beta \in F$:

i) if α is a tautology, then $D^*(v)(\alpha) = 1$,
ii) if α is a contradiction, then $D^*(v)(\alpha) = 0$,
iii) if α entails β, then $D^*(v)(\alpha) \leq D^*(v)(\beta)$,
iv) if α is logically equivalent to β, then $D^*(v)(\alpha) = D^*(v)(\beta)$,
v) $D^*(v)(\beta) \geq D^*(v)(\alpha) \wedge D^*(v)(\alpha\to\beta)$,
vi) $D^*(v)(\alpha\wedge\beta) = D^*(v)(\alpha) \wedge D^*(v)(\beta)$,
vii) $D^*(v)(\alpha\vee\beta) \geq D^*(v)(\alpha) \vee D^*(v)(\beta)$.

Moreover,

$$D^*(v)(\alpha\vee\beta) \neq D^*(v)(\alpha)\vee D^*(v)(\beta),$$

in general.

Proof. Propositions i), ii), iii), iv), v) and vii) are evident. To prove vi) observe that, since $\alpha\wedge\beta \to \alpha$ and $\alpha\wedge\beta \to \beta$ are tautologies, $D^*(v)(\alpha\wedge\beta) \leq D^*(v)(\alpha)$ and $D^*(v)(\alpha\wedge\beta) \leq D^*(v)(\beta)$. Then,

$$D^*(v)(\alpha\wedge\beta) \leq D^*(v)(\alpha) \wedge D^*(v)(\beta).$$

Moreover, by observing that from $\alpha_1,...,\alpha_n \vdash \alpha$ and $\beta_1,...,\beta_m \vdash \beta$ it follows that $\alpha_1,...,\alpha_n, \beta_1,...,\beta_m \vdash \alpha$ and $\alpha_1,...,\alpha_n, \beta_1,...,\beta_m \vdash \beta$, we have that

$$\begin{aligned}
&D^*(v)(\alpha) \wedge D^*(v)(\beta)\\
&\quad = (Sup\{v(\alpha_1)\wedge\ldots\wedge v(\alpha_n) : \alpha_1,...,\alpha_n \vdash \alpha\}) \wedge (Sup\{v(\beta_1)\wedge\ldots\wedge v(\beta_m) : \beta_1,...,\beta_m \vdash \beta\})\\
&\quad = Sup\{v(\alpha_1)\wedge\ldots\wedge v(\alpha_n)\wedge v(\beta_1)\wedge\ldots\wedge v(\beta_m) : \alpha_1,...,\alpha_n \vdash \alpha \text{ and } \beta_1,...,\beta_m \vdash \beta\}\\
&\quad \leq Sup\{v(\gamma_1)\wedge \ldots\wedge v(\gamma_t) : \gamma_1,...,\gamma_t \vdash \alpha \text{ and } \gamma_1,...,\gamma_t \vdash \beta\}\\
&\quad = Sup\{v(\gamma_1)\wedge\ldots\wedge v(\gamma_t) : \gamma_1,...,\gamma_t \vdash \alpha\wedge\beta\}\\
&\quad = D^*(v)(\alpha\wedge\beta).
\end{aligned}$$

Assume that α is undecidable in the support of v. Then, since

$$D^*(v)(\alpha) = D^*(v)(-\alpha) = 0,$$

we have that $D^*(v)(\alpha)\vee D^*(v)(-\alpha) = 0$ while $D^*(v)(\alpha\vee-\alpha) = 1$. □

6. Incomplete information.

Assume that the lattice L coincides with a complete Boolean algebra, as an example, with the Boolean algebra $B = \mathcal{P}(PC)$. Then, we interpret a B-subset $v : F \to B$, by assuming that $v(\alpha)$ is the set of past cases in which <u>we know</u> that α is true. Moreover, it is easy to prove that $D^*(v)(\alpha) = PC$ if α is a tautology and, otherwise,

$$D^*(v)(\alpha) = \cup\{v(\alpha_1)\cap\ldots\cap v(\alpha_n) : \alpha_1,\ldots,\alpha_n \vdash \alpha\}.$$

This means that $D^*(v)(\alpha)$ is the set of past cases in which we can prove that α is true. Moreover, v is consistent if and only if, given any contradiction α, $D^*(v)(\alpha) = \emptyset$, i.e. no case c exists such that a contradiction can be proved.

Definition 6.1. Let B be a Boolean algebra, $p : B \to [0,1]$ a probability and $v : F \to B$ a B-subset of F. Then we call *belief associated with* v the map $Bel(v) : F \to [0,1]$ defined by setting, for every $\alpha \in F$

$$Bel(v)(\alpha) = p(D^*(v)(\alpha)).$$

In accordance with Proposition 5.2, $D^*(v)$ is not a truth-functional B-valuation. In fact, it is possible that $Bel(v)(\alpha) = Bel(v)(\neg\alpha) = 0$. This entails that, differently of the case of complete information, $Bel(v)$ is not a probability valuation, in general. The following proposition shows some basic properties of $Bel(v)$.

Proposition 6.1. Let v be a consistent B-subset. Then the map $Bel(v) : F \to [0,1]$ satisfies the following properties:

i) if α is a tautology, then $Bel(v)(\alpha) = 1$
ii) if α is a contradiction, then $Bel(v)(\alpha) = 0$
iii) if α implies β, then $Bel(v)(\alpha) \leq Bel(v)(\beta)$
iv) if α is equivalent to β, then $Bel(v)(\alpha) = Bel(v)(\beta)$
v) $Bel(v)(\alpha\vee\beta) \geq Bel(v)(\alpha) + Bel(v)(\beta) - Bel(v)(\alpha\wedge\beta)$.

Proof. We confine ourselves to prove v). Indeed,

$$\begin{aligned} Bel(v)(\alpha\vee\beta) &= p(D^*(v)(\alpha\vee\beta)) \\ &\geq p(D^*(v)(\alpha) \cup D^*(v)(\beta)) \\ &= p(D^*(v)(\alpha)) + p(D^*(v)(\beta)) - p(D^*(v)(\alpha)\cap D^*(v)(\beta)) \\ &= Bel(v)(\alpha) + Bel(v)(\beta) - p(D^*(v)(\alpha\wedge\beta)) \\ &= Bel(v)(\alpha) + Bel(v)(\beta) - Bel(v)(\alpha\wedge\beta). \end{aligned}$$ □

In the case of incomplete information, we propose the following obvious extension of Definition 3.1.

Definition 6.2. We call (partial) *knowledge representation system* any structure $\mathcal{S} = (PC, AT, OBS, tr)$ such that $tr : PC\times F \to \{i,1\}$ is a map from $PC\times F$ into $\{i,1\}$.

Notice that tr is defined in the whole set F of formulas and not only in the set of propositional variables. In the case that $tr(c,\alpha) = i$, we say that α is *undetermined* in c. The information "α false in c" is stored by setting $tr(c,\text{-}\alpha) = 1$. We say that two past cases c_1 and c_2 are *indiscernible* if $tr(c_1,\alpha) = tr(c_2,\alpha)$ for every formula α. As in the case of complete information, we denote by $\mathcal{S}^* = (PC^*, AT, OBS, tr^*)$ the quotient of $\mathcal{S}$ modulo the indiscernibility relation. Also, for every equivalence class $[c]$, we consider the number of elements contained in [c]. It is also immediate how define the notion of (incomplete) statistical inferential basis.

Definition 6.3. We call *incomplete statistical inferential basis*, a structure $\mathcal{S}$ = (*TC*, *AT*, *OBS*, *tr*, *w*) such that
i) (*TC*, *AT*, *OBS*, *tr*) is a partial knowledge representation system,
ii) two elements of *TC* are always discernible,
iii) $w : TC \to N$ is a function we call *the weight function*.
The elements of *TC* are called *type-cases*.

Now, given an incomplete statistical inferential basis $\mathcal{S}$, we can set

$$v(\alpha) = \{c \in CT : tr(c,\alpha) = 1\}.$$

Then, the map $v : F \to \mathcal{P}(CT)$ is the function that associates any formula α with the set $v(\alpha)$ of cases in which we know that α was verified. Differently from the case of complete information, the map v is not truth-functional, in general. As a matter of fact, v is any map from F to $\mathcal{P}(TC)$. We interpret the B-subset v as a system of axioms (the available information). A natural semantics is obtained if we call *model* any truth-functional B-valuation m. We say that m is a model of v if $m(\alpha) \supseteq v(\alpha)$. Consider the deduction operator $D : \mathcal{P}(F) \to \mathcal{P}(F)$ of the classical propositional calculus and the related extension $D^* : \mathcal{P}(TC)^F \to \mathcal{P}(TC)^F$. To face the problem of the incomplete information, the idea is to consider D^* as a tool to complete in part our information. We can interpret the set $D^*(v)(\alpha)$ as the set of the cases in which the available information is sufficient to prove α. Moreover, we can consider the belief $Bel(v) : F \to [0,1]$ associated with v. It is easy to prove that, given any formula α, the number $Bel(v)(\alpha)$ denotes the frequency of cases in which we have information sufficient to prove α, i.e.,

$$Bel(v)(\alpha) = \frac{\sum\{w(c) : c \in D^*(v(\alpha))\}}{w(S)}.$$

The inferential process from an incomplete statistical inferential basis $\mathcal{S}$ runs as in the complete statistical inferential basis. Indeed, given an actual case a_c and a formula α in the language F, we obtain a probabilistic valuation of the claim that a_c satisfies α as follows:
- we obtain a set T of formulas in F_{obs} satisfied by a_c as a result of a sequence of queries (tests)
- we define a new representation system $\mathcal{S}(T)$ by considering only the cases satisfying T
- we consider the quantity $Bel(v)(\alpha)$.
We conclude by observing that from i) and v) in Proposition 6.1, it follows that

$$1 = Bel(v)(\alpha \vee \text{-}\alpha) \geq Bel(v)(\alpha) + Bel(v)(\text{-}\alpha).$$

By setting $Bel(v)^*(\alpha) = 1\text{-}\ Bel(v)(\text{-}\alpha)$, we have that

$$Bel(v)(\alpha) \leq Bel^*(v)(\alpha).$$

Then, our inferential apparatus furnishes, for every formula α, an interval valuation $[Bel(v)(\alpha), Bel^*(v)(\alpha)]$ of α, probabilistic in nature. The intended interpretation is that the "actual" probability of α is a number in such an interval. In the case of complete information $Bel(v) = Bel^*(v)$ is a probability valuation and the interval valuation coincides with it.

References.

[1] F. Bacchus, *Lp*, a logic for representing and reasoning with statistical Knowledge", *Comput. Intell.*, 6 (1990) 209-231.
[2] L. Biacino, G. Gerla, An extension principle for closure operators, *J. of Math. Anal. Appl.*, 198 (1996) 1-24.
[3] G. Gerla, The probability that Tweety is able to fly, *International Journal of Intelligent Systems*, 9 (1994) 403-409.
[4] G. Gerla, *Fuzzy Logic: Mathematical Tools for Approximate Reasoning*, Kluwer Academic Publishers (2000) Dordrecht.
[5] G. Gerla, L. Scarpati, Extension principles for fuzzy set theory, *Journal of Information Sciences*, 106 (1998) 49-69.
[6] J. Y. Halpern, An analysis of first-order logic of probability, *Artificial Intelligence*, 46 (1990) 331-350.
[7] Z. Pawlak, Rough sets, *International Journal of Information and Computer Science*, 11 (1982) 344-356.