Uncertainty In Artificial Intelligence
Theory

Christian Rossi

Academic Year 2023-2024

Abstract

The topics of the course are:

- Uncertainty sources that affect models: typology, issues, and modeling approaches.
- Measure-based uncertainty modeling.
- Logic-based uncertainty modeling.
- Fuzzy models: fuzzy sets, fuzzy logic, fuzzy rules, motivations for fuzzy modeling, tools for fuzzy systems, design of fuzzy systems, applications.
- Bayesian networks: basics, design, learning, evaluation, applications.
- Hidden Markov Models: basics, design, learning, evaluation, applications.
- Applications: motivations, choices, models, case studies.

Contents

CHAPTER 1

Introduction

1.1 Definition

Definition (*Uncertainty*)

> *Uncertainty* refers to epistemic situations involving imperfect or unknown information. It applies to predictions of future events, to physical measurements that are already made, or to the unknown.

Uncertainty arises in partially observable or stochastic environments, as well as due to ignorance, indolence, or both. It arises in any number of fields, including insurance, philosophy, physics, statistics, economics, finance, medicine, psychology, sociology, engineering, metrology, meteorology, ecology and information science.

The lack of certainty, a state of limited knowledge where it is impossible to exactly describe the existing state, a future outcome, or more than one possible outcome. This puts in evidence that uncertainty is related to the need of describing a piece of reality.

1.2 Modelling

Modelling is at the base of our life: the way we interact with the world is through models that interpret data coming from sensors and generate knowledge and actions. Modelling is also the way we may represent entities in a computer and possibly making it reasoning on them.

Definition (*Model*)

> A *model* is a representation of some entity, defined for a specific purpose. A model captures only the aspects of the entity modelled that are relevant for the purpose. A model is necessarily different from the modelled entity. So, intrinsic to modelling are all sort of uncertainties.

All Artificial Intelligence applications are based on models, either defined by somebody or learned. These models are represented In different ways, but share uncertainty issue mainly on inputs.

1.3 Uncertainty classification

The uncertainty can be of two main types:

- Epistemic uncertainty: it is due to things one could in principle know but does not in practice. This may because a measurement is not accurate, because the model neglects certain effects, or because particular data have been deliberately hidden. It is also known as systematic uncertainty and can in principle be reduced by enriching the model.

- Aleatoric uncertainty: it is representative of unknown unknowns that differ each time we run the same experiment. Aleatoric uncertainty is also known as statistical uncertainty, since only statistical information can describe it. This may also depend on the way we get and elaborate data. In general, it is present when the model is missing some aspects.

The sources of uncertainty can be:

- Parameter: it comes from the model parameters, whose exact values are unknown to experimentalists and cannot be controlled in experiments, or whose values cannot be inferred by statistical methods.

- Parametric variability: it comes from the variability of input variables of the model.

- Structural: also known as model inadequacy, model bias, or model discrepancy, this comes from the lack of knowledge of the problem.

- Algorithmic: also known as numerical uncertainty, or discrete uncertainty. This type comes from numerical errors and numerical approximations in the implementation of the computer model.

- Experimental: also known as observation error, this comes from the variability of experimental measurements.

- Interpolation: this comes from a lack of variable data collected from computer model simulations and/or experimental measurements.

1.4 Uncertainty modelling

The type of uncertainty model depends on the type of uncertainty, its sources and the information we have in uncertainty and mostly has to do with qualification and quantification of uncertainty. The possible models for uncertainty are: statistical, logical and cognitive.

Artificial Intelligence and Machine Learning technologies are based on models that include uncertainty models of these sorts, essential not only for the implementation of effective models, but also to define learning models able to cope with complex situations, and to evaluate the quality of learned/developed models. There are two major types of problems in uncertainty quantification:

- Forward propagation of uncertainty: the various sources of uncertainty are propagated through the model to predict the overall uncertainty in the system response:
  - To evaluate low-order moments of the outputs (mean and variance).
  - To evaluate the reliability of the outputs.

  - To assess the complete probability distribution of the outputs.

  This is what is done also in Bayesian networks and graphical models.

- Inverse assessment of model uncertainty and parameter uncertainty, where the model parameters are calibrated simultaneously using test data: given some experimental measurements of a system and some results from its mathematical model, inverse uncertainty quantification estimates the discrepancy between the experiment and the mathematical model (bias correction) and estimates the values of unknown parameters in the model if there are any (parameter calibration).

The models used in Artificial Intelligence can be classified in three main types:

- Symbolic models: elements of the models are expressed as terms related to entities to be modelled. The state of the world is represented by facts expressed in formal languages close to natural languages.
- Sub-symbolic models: elements of the models are expressed by code.
- Black-box models: the model can be computed and possibly investigated, but it is only regarded as a computational way to map inputs to outputs.

For symbolic models a fact is true in a model if it is possible to collect enough evidence to support it. The only really true facts are the ones true by definition. All the others may be supported by evidence.

1.5 Ignorance management

There are many potential sources of ignorance when reasoning in the real world:

- Insufficient data.
- Biased data: data are collected by sensors affected by errors.
- Variable data: data are collected by imprecise sensors.
- Reliability of data.
- Fuzzyness.
- Reliability of the model: depends on the model design, implementation and parametrization.
- Incompleteness of the model.

Example: Let's consider the sentence "The elephant weighs 2 tons". This can be interpreted in various ways, each slightly different:

- The elephant weighs exactly 2 tons.
- The elephant weighs 2 tons ± 10 kg, given the resolution of the weight scales of the instrument.
- The elephant weighs approximately 2 tons, but we cannot say anything more precise.
- We are not sure about any previous sentence because we do not have enough evidence.

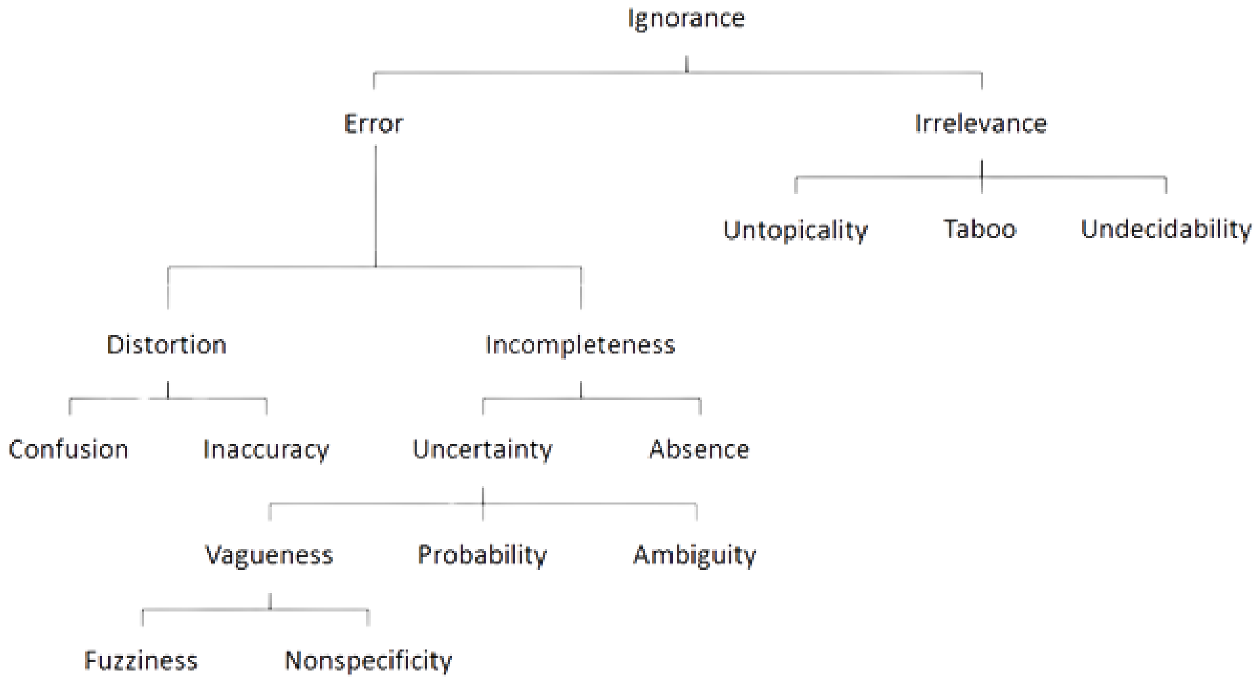


Figure 1.1: Smithson's taxonomy of ignorance and uncertainty

To model ignorance most often it is decided to associate measures of some aspects. Let's distinguish between two aspects:

- The type of representation: numbers, labels, intervals, ...
- The represented ignorance that we would like to model: i.e, probability, reliability, subjective evaluation, ...

The probability is represented with numbers between zero and one, and a well-established set of rules and properties are associated to its management, among which, given a set of alternative hypothesis:

- The sum of their probabilities should be one.
- The probability a posteriori of a hypothesis h_i given some evidence e is given by the Bayes theorem:
$$P(h_i \mid e) = \frac{P(e \mid h_i)P(h_i)}{P(e)}$$

Probability was used, for example, in the MYCIN that was one of the first expert systems, aimed at diagnosing blood illness. They modeled certainty by considering two numerical factors:

- Measure of increased Belief: $MB = \frac{P(\frac{h}{e})-P(h)}{1-P(h)}$.
- Measure of decreased Disbelief: $MD = \frac{P(h)-P(\frac{h}{e})}{P(h)}$.

The measure of a statement is given by the certain factor:

$$CF = MB - MD \in [-1;1]$$

The main hypothesis for this solution is that the number given as MB and MD are not statistical probabilities, but subjective probabilities, provided by different experts and combined by rules (this may be ambiguous).

Compared to probabilities, linguistic terms are less ambiguous than numbers. Using a limited set of labels it is possible to associate to statements subjective evaluation, on which it is relatively easy to make subjective judgements converge. Then, a computational mechanism is needed to define how to combine labels. This is done by using fuzzy systems, that are a representation of truth of a statement in linguistic terms, as evaluation of its fuzzyness.

CHAPTER 2

Fuzzy sets

2.1 History

Fuzzy sets have been defined by Lotfi Zadeh in 1965 as a tool to model approximate concepts. In 1972 the first linguistic fuzzy controllers has been implemented. Around 1980 the fuzzy were used frequently worldwide. In the Nineties there were a massive diffusion of fuzzy controllers in various end-user goods. Today, fuzzy systems are the kernel of many intelligent devices.

2.2 Fuzzy membership function

A ”crisp” set is defined by a boolean membership function on some property on the considered elements. Instead, a ”fuzzy” set is a set whose membership function that ranges in the values between zero and one.

Definition (*Membership function*)

> A *membership function* defines a set, by defining the degree of membership of an element of the universe of discourse to the set. A name is given to the set to make it possible to refer to it: this is usually called *label*. Fuzzy sets can also be defined with a variable with discrete values.

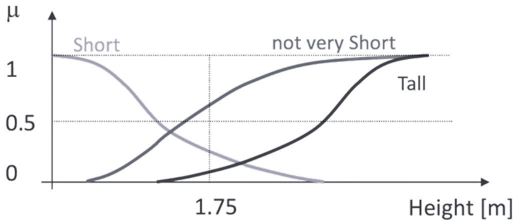


Figure 2.1: Example of a membership function

To define a membership function we have to (according to the purpose of the model and the available data):

1. Select a variable on which the membership function will be defined.

2. Define the range of the variable.

3. Identify the fuzzy sets needed for the application and define the labels.

4. For each fuzzy set identify characteristic points for the membership function.

5. Define the shape of the membership function.

6. Check if the membership function is correct.

The shapes of the membership function can be chosen arbitrarily. The choice of the shape modify the smoothness of the transition between two labels (i.e., in intervals (horizontal shape) the transition is immediate).

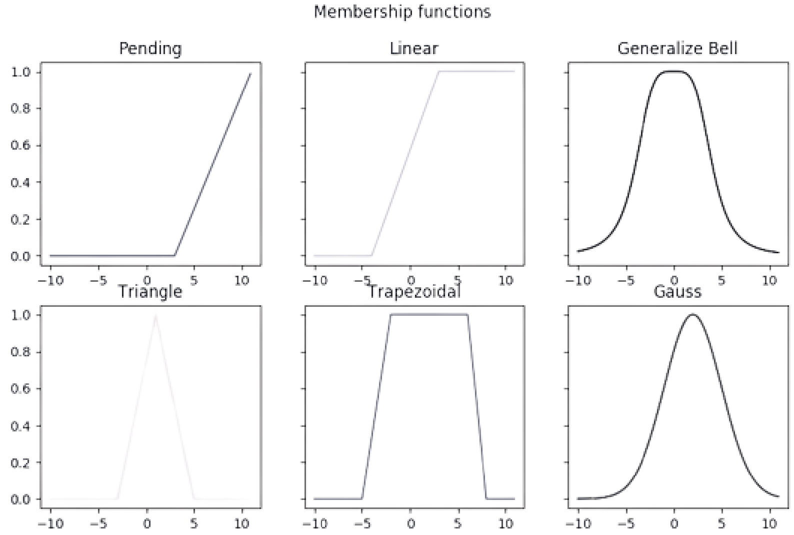


Figure 2.2: Possible shapes for a membership function

Definition

A set of fuzzy sets fully covering the universe of discourse is called *frame of cognition.* The properties of this set are:

- Coverage: each element of the universe of discourse us assigned to at least one granule with membership greater or equal than zero.

- Uni-modality of fuzzy sets: there is a unique set of values for each granule with maximum membership.

Definition

A frame of cognition for which the sum of the membership values of each value of the base variable is equal to one is called a *fuzzy partition.*

Definition

The *α-cut* of a fuzzy set is the ”crisp” set of the values of x such that $\mu(x) \geq \alpha$:

$$\alpha_\mu(x) = \{x \mid \mu(x) \geq \alpha\}$$

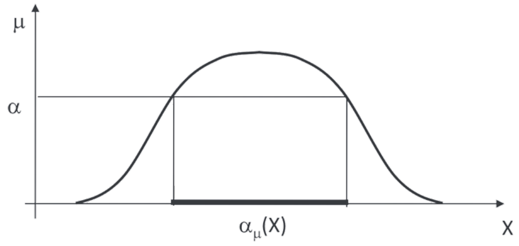


Figure 2.3: Alpha-cut of a membership function

Definition (*H*)

The *support* of a fuzzy set is the "crisp" set of values x such that $\mu_f(x) > 0$ is the *support* of the fuzzy set f on the universe X.

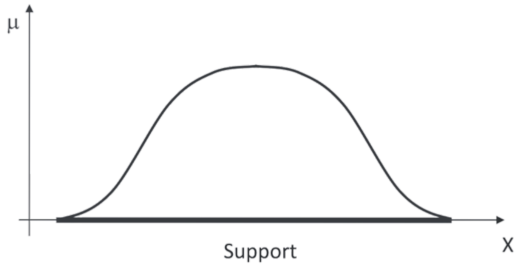


Figure 2.4: Support of a membership function

Definition

The height h_f of a fuzzy set f on the universe X is the highest membership degree of an element of X to the fuzzy set:

$$h_f(X) = \max_{x \in X} \mu_f(x)$$

A fuzzy set is normal if, and only if, $h_f(X) = 1$.

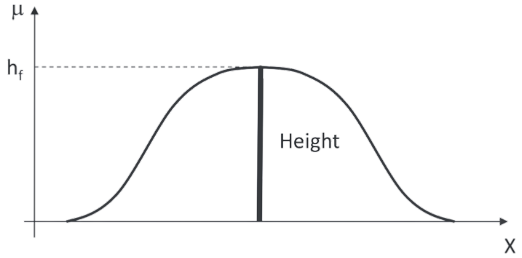


Figure 2.5: Height of a membership function

Definition

A fuzzy set is *convex* if and only if

$$\mu[\lambda x_1 + (1 - \lambda)x_2] \geq \min[\mu(x_1), \mu(x_2)]$$

for any $(x_1, x_2) \in \mathbb{R}$ and any $\lambda \in [0, 1]$.

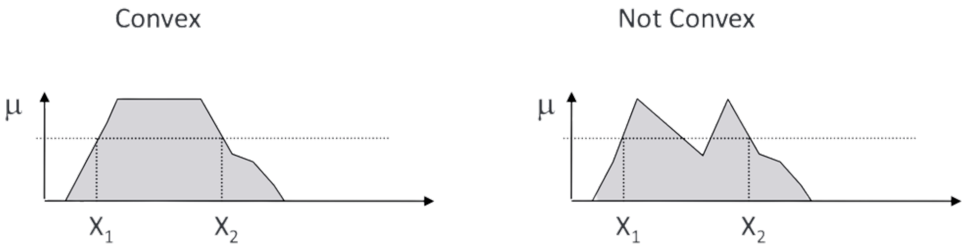


Figure 2.6: Graphical difference between a convex and a not convex set

The particular fuzzy sets are: singleton (a fuzzy set with exactly one member) and interval (a fuzzy set whose members have all membership equals to one). The possible operations on the fuzzy sets are:

- Complement: $\mu_{\bar{f}}(x) = 1 - \mu_f(x)$.
- Union: $\mu_{f_1 \cup f_2}(x) = \max[\mu_{f_1}(x), \mu_{f_2}(x)]$.
- Intersection: $\mu_{f_1 \cap f_2}(x) = \min[\mu_{f_1}(x), \mu_{f_2}(x)]$.

CHAPTER 3

Fuzzy logic

3.1 Introduction

Logic is a tool that has been used since a thousand of years to formally represent knowledge. There are many types of logic:

- Propositional: truth values for proposition.
- First order: truth values for predicates (with variables and quantifiers).
- Second order: predicates of predicates.

These types of logic are binary. We may notice that the meaning of the terms in these logics is not defined together with the formalism, and this is not needed to make the logic work.

3.2 Propositional logic

Propositional logics are concerned with propositional operators which may be applied to one or more propositions giving new propositions. The accent is on the truth value of propositions and on how these truth values are composed.

Definition

A logic is *truth functional* if the truth value of a compound sentence depends only on the truth values of the consistent atomic sentences, not on their meaning or structure. For such a logic the only important question about propositions is what truth values may have.

In a classical, boolean or two-valued logic every proposition is either true or false and no other feature of the proposition is relevant.

The main operators in the propositional logics are: conjunction ($\wedge$), disjunction ($\vee$) and negation ($\neg$).

3.3 First order predicate logic

The first order logic is the same as propositional logic augmented with the possibility to define predicates on variables. Furthermore, existential ($\exists$) and universal ($\forall$) quantifiers are defined.

In predicate logics it is possible to infer the truth value of a proposition by inferential mechanisms, such as Modus Ponens.

Inference: Given the sentences: "All man are mortal" and "Socrates is a man" we can infer that "Socrates is mortal".

Inference is used to model a mechanism that we have in our minds to store a reduced amount of information and set a mechanism that can be applied to derive from information other information, to face everyday situations.

Definition

Information and potential relationship together compose what we call *knowledge.*

3.4 Many-valued logics

Aristotle already had put in evidence problems about the validity of classical logic as a knowledge representation tool. For instance, it is difficult to state the truth value of a proposition in the future. To solve this problem, let's introduce a third value (i.e., 0.5) for the undefined situation and define a three-valued logic. From this to an infinite set of truth values there is just a small step.

Infinite-value logics considers a continuum of truth values between zero and one for example.

Logic L1, Łukasiewicz(1930): The main rules in this type of infinite-value logic are:

- $T(\neg a) = 1 - T(a)$.
- $T(a \wedge b) = \min(T(a), T(b))$.
- $T(a \vee b) = \max(T(a), T(b))$.
- $T(a \implies b) = \min(1, 1 + T(b) - T(a))$.
- $T(a \Leftrightarrow b) = 1 - |T(a) - T(b)|$.

This innovations bring a change in the society: things are no longer stated as true or false, probability (kolmogorov, 1929) and stochasticity (Markov, 1906) became the way to represent the new approach to science and life.

The difference between classical logic L2 and many-valued logic L1 are the following:

- L1 is isomorphic to the fuzzy set theory with standard operators as the classical logic L2 is isomorphic to the set theory.
- Tautologies are true by definition, and are used to prove theorems, so to prove the truth of an inferential chain. Some tautologies valid in L2 are no longer valid in L1, for example:
  - Third excluded law ($T(a \vee \neg a) = 1$)
  - Non-contradiction law ($T(a \wedge \neg a) = 0$).

The sentence "I'm a liar" would be a paradox in classical logic, if we give a meaning to the term "liar", since no formula can have the same truth value of its negation. This may not be so in many-valued logics. In Łukasiewicz logic, for instance, it can be that the truth value of a sentence is 0.5, and that its negation is the same, so the proposition is consistent with the axioms, and it is not a paradox.

3.5 Fuzzy logic

Fuzzy logic is an infinite-valued logic, with truth values in $[0 \dots 1]$ and prepositions are expressed as:

$$A\, is\, L$$

where:

- A is a linguistic variable.
- L is a label denoting a fuzzy set.

Formally, a linguistic variable is defined by a 5-tuple $(X, T(X), U, G, M)$, where:

- X is the name of the variable.
- $T(X)$ is the set of term for X, each corresponding to a fuzzy variable denoted by $T(X)$ and ranging on U.
- U is the universe of discourse defined on a base variable u.
- G is the syntactic rule used to generate the interpretation X of each value u.
- M is the semantic rule used to associate to X its meaning.

Linguistic variable for age : We can define a linguistic variable for the age in the following way:

- X is a linguistic variable labelled "age".
- $U = [0 \dots 100]$.
- $T(X) = old, middle - aged, young, \dots.$
- $u = [0 \cdots + \infty]$.
- M is the definition in terms of fuzzy sets of the values of X.
- G is the fuzzy matching interpretation of u.

Now that we have defined the linguistic variable it is possible to write a simple proposition in the following way:

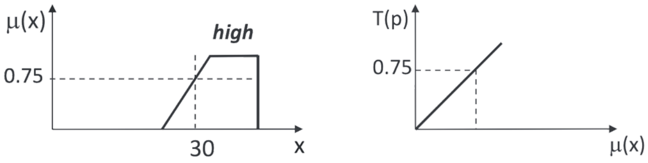
$$p \;:\; X\, is\, F$$

where:

- X is a linguistic variable.
- F is the label of a fuzzy set, defined on U, which represent a fuzzy predicate.
- $\mu_F(x)$ is the membership function defining F, and it is interpreted as truth value for the preposition p $(T(p) = \mu_F(x))$.

Therefore, the truth value of the preposition P is a fuzzy set defined on $[0 \dots 1]$.

Example : Given the simple proposition "p:temperature is high", where X is temperature and F is high we can find the truth value of this preposition using the graph of the membership function given:



So, the truth value of the given proposition is 0.75.

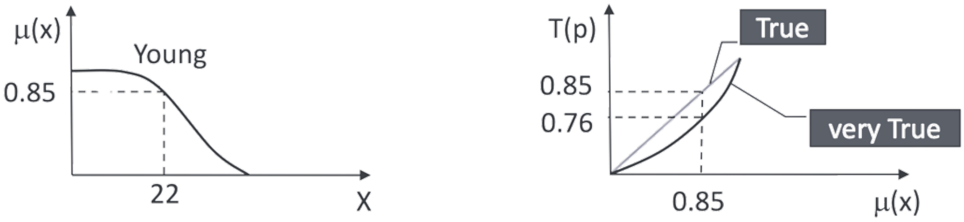
It is also possible to define qualified, non-conditional propositions with this syntax:

$$p \ : \ (X \ is \ F) \ is \ S$$

where:

- S is a fuzzy truth qualifier.
- F is a fuzzy set.
- p is truth qualified.

Example : Given the conditional proposition "p:age of Tina is young is very true", where X is age, F is young and S is very true we can find the truth value of this preposition using the graph of the membership function given:



In the fuzzy logic it is possible to use fuzzy modifiers to modify the truth values of the propositions. The modifiers can be of two main types:

- Strong ($m(a) \leq a \, \forall a \in [0 \ldots 1]$): they make the predicate stronger, so they reduce the truth of the preposition.
- Weak($m(a) \geq a \forall a \in [0 \ldots 1]$): they make the predicate weaker, so they increase the truth of the preposition.

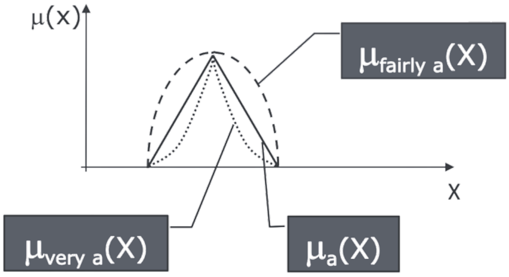
The properties of the fuzzy modifiers are:

- $m(0) = 0$ and $m(1) = 1$.
- m is a continuous function.
- If m is strong m^{-1} is weak, and the other way around.
- Given another modifier g, the composition of g and m and the other way round are modifiers, too, and, if both are strong (or weak), so it's their composition.

Example : The sentence ”x is young” actually means ”(x is young) is true”. This sentence can be modified in the following ways with fuzzy modifiers:

- ”x is very young is true”.
- ”x is young is very true”.
- ”x is very young is very true”.

Graphically we can draw the modified membership function as:



where:

- $\mu_{very\,a}(x) = \mu_a(x)^2$.
- $\mu_{fairly\,a}(x) = \mu_a(x)^{\frac{1}{2}}$.

3.6 Inference rules

Definition

An *inference rule* is a model. In other words it is a way to define a mapping from input to output. Rules are used to represent inferential relationships among pieces of knowledge.

We consider forward chaining rules, having the shape:

$$IF\ antecedent\ THEN\ consequent$$

where:

- *antecedent* is a set of clauses related by logical operators.
- *consequent* is a set of clauses related by logical operators.

The clauses used in the inference rules are either a proposition (sequence of symbols) or a pattern (sequence of symbols and variables).

Inference rules are used to implement Knowledge-Based Systems, among which Expert Systems are mostly known as successful Artificial Intelligence applications. An Expert System is designed upon the experience of somebody to replicate, or improve his performance in solving a problem. Knowledge Acquisition is a complex process bringing to the definition of rule-based systems, implemented and running on computers.

A system can generate new information using rules and other related information using these steps:

1. Pattern matching: identify the rules whose antecedents match the known facts (saved in the fact base). These can be considered for activation, given the corresponding assignment to variables.

2. Selection of the rules to be activated: among the rules identified with pattern matching (candidate rules), select the rules that should be activated.

3. Activation of the selected rules: assert the consequent of the selected rules in the fact base.

Example : Suppose that the rule base contains the following four rules:

1. "If X croaks and X eats flies, then X is a frog".
2. "If X chirps and X sings, then X is a canary".
3. "If X is a frog, then X is green".
4. "If X is a canary, then X is yellow".

Now suppose to observe the following facts (fact base):

- Fritz croaks.
- Fritz eats flies.

From rule 1 and facts a and b we can add to the fact base the fact:

$$Fritz\ is\ a\ frog$$

Given the new fact base, we can use rule 3 to deduce the fact:

$$Fritz\ is\ green$$

3.7 Fuzzy rules

Definition

A *fuzzy rule* is a rule whose clauses have the shape

$$(V\ is\ L)$$

where V is a linguistic variable and L is a label, a value for V associated to a fuzzy set. This is a *linguistic clause.*

Often, clauses in the antecedent are only related by the AND operator which is not explicitly written. The antecedent is usually matched against facts that are represented as values of the base variables corresponding to the linguistic variables. The consequent may be one of two types:

- Linguistic rules: the consequent is a conjunction of linguistic clauses. These rules can be considered as a mapping between the interpretation of an input configuration and a symbolic description of the desired output. The general formula is:

$$IF\ (A\ is\ LA_i)\ AND\ (B\ is\ LB_k)\ AND\ \ldots\ THEN\ (U\ is\ LU_m)\ AND\ \ldots$$

- Model rules: bind a model to the linguistic interpretation of its applicability conditions. This can be considered as a mapping between the interpretation of an input configuration and a model to be applied to the input real values to obtain the output. The general formula is:

$$IF\ (A\, is\, LAn)\ AND\ (B\, is\, LBk)\ AND\ \ldots\ THEN\, U\, is\, f(A, B)$$

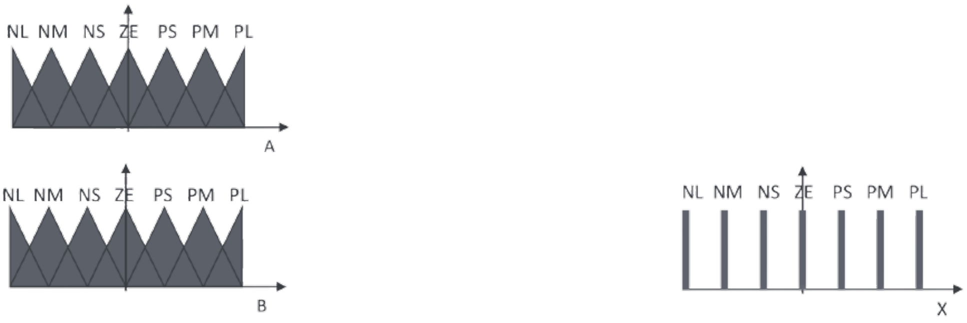
The steps to use the fuzzy rules are the following:

1. Input matching.
2. Combination of matching degrees.
3. Combination with rule weight, if present.
4. Aggregation of output from different rules.
5. Eventual defuzzyfication of output.

To defuzzyficate the output it is possible to consider various operators other than the weighted mean, for example: centroid, bisector, average of maxima, the lowest maximum, the highest maximum, center of the highest area, . . . Depending on the choice the system change the output and the level of optimization.

Linguistic rules : Let's consider:

- Two input variables (fuzzy partition) A and B equally distributed from Negative Large to Positive Large.
- One output variable U (equally distributed fuzzy set) from Negative Large to Positive Large. The fuzzy sets are all singletons.



We now have to define the rules of the rule base (each one with a weight) as follows:

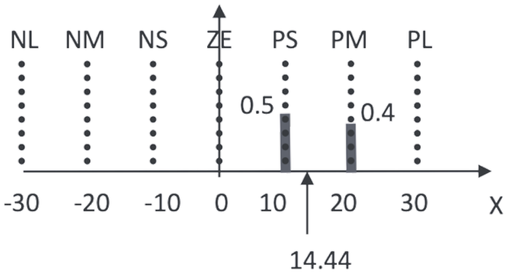
1. IF A is PL AND B is PS THEN X is PM (weight 1).
2. IF A is PM AND B is PS THEN X is PS (weight 0.5).
3. IF A is PL AND B is PM THEN X is PM (weight 1).

Let's now set $A = 22$ and $B = 140$. The steps used to calculate the output value are the following:

1. For the first step we have to check the corresponding value for each label. In this case we have:

- (A is PL) has a truth value of 0.2.
- (B is PS) has a truth value of 0.6.
- (A is PM) has a truth value of 0.8.
- (B is PM) has a truth value of 0.4.

2. To consider the degree of truth of each predicate we simply take the minimum between the two values (because there is an AND operator). So, the result will be 0.2 for the first rule, 0.6 for the second and 0.4 for the third one.
3. Now we have to consider the rule weight. To do so we simply select the minimum between the selected value and the weight value. So the final value for the consequent are: 0.2 for the first rule, 0.5 for the second and 0.4 for the third one.
4. Now we have to aggregate the output. If we have a repeated expression we take the maximum value between the possible ones. In this case we obtain that (X is PM) has a truth value of 0.4 (maximum between 0.2 and 0.4) and (X is PS) has a truth value of 0.5. This result can be visualized graphically by cutting the initial graph as follows.



5. Now we can defuzzyficate the result by obtaining a number. We need to do this operation because the input is a number and the desired output needs to be a number. To calculate the exact value of the output it is possible to use a simple weighted mean:

$$X = \frac{10 \cdot 0.5 + 20 \cdot 0.4}{0.5 + 0.4} = 14.44$$

Model rules : The variables are the same as the previous example. The rules we are going to consider are the following:

1. IF A is PL and B is PS THEN X is $A + 2B$.
2. IF A is PM and B is PS THEN X is $A + 3$.
3. IF A is PL and B is PM THEN X is $A + B$.

The models considered in these rules are all linear. Pattern matching is the same as the previous example, and so we have that the subsequent have the following degree of truth: the first one has a value of 0.2, the second a value of 0.5 and the third a value of 0.4. For the output aggregation we consider again the weighted mean and use the initial value of $A = 22$ and $B = 140$ in the resulting formula:

$$X = \frac{0.2 \cdot (A + 2B) + 0.5 \cdot (A + 3) + 0.4 \cdot (A + B)}{0.2 + 0.5 + 0.4} = 125.18$$

3.8 Fuzzy system design

To design a fuzzy system we have to follow those steps:

1. Problem definition.

2. Parametrization of the model: concepts.
3. Mapping definition: rules.
4. Implementation.
5. Testing.

In the problem definition phase we have to choose all the input and output variables and the goal of the model. In principle, input variables are numerical or ordinal (like colors) variables so that it is possible to define fuzzy sets on them. Variables can be either:

- Perceived values coming directly from sensors, data, or users.
- Computed from perceived variables.

The selection of the variables is up to designers, so there aren't best or worst input variables to select. Output variables are the result of the models, so come directly from the modeler needs. The goals of the fuzzy models depend on the specification. The goals should always be stated in advance and guide the design.

The system parametrization is based on:

- Selection of the membership functions for all variables. These function can be defined by a:
  - Single expert, with objective evaluation or interviews.
  - Multiple expert, which is more reliable.
  - Automatic systems working on data (like Neural Networks).

  The number of membership functions for each variable varies between three and seven. Any point in the range of input variables has to be covered by at least one fuzzy set participating to at least one rule and boundaries should be covered with maximum value.
- Selection of the inferential mechanism. The inferential engine depends on the operators selected for:
  - AND of antecedent clauses (minimum: the worst degree of matching is the most relevant; product: all the degrees of matching are relevant).
  - Detachment: combination with the rule weight (minimum or product).
  - Aggregation of the degrees of the same consequent (max: the best degree is the most relevant; probabilistic sum: all the knowledge is considered).
- Selection of eventual fuzzyfication and defuzzyfication.

The rules can be defined:

- From experience.
- From another model.
- By using Machine Learning, or self-tuning techniques (like Neural Networks).

The testing can be done with: dynamic simulation, static simulation or directly on the process, possibly under safe conditions.

3.9 Applications of fuzzy systems

The fuzzy controls are systems able to control the behavior of another system. In most cases it is a PID controller, where the output depends on the difference between the desired and the observed behavior.

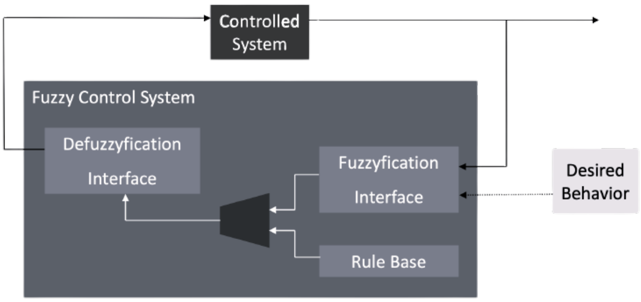


Figure 3.1: General schema of a fuzzy control system

The main features of a fuzzy control system are:

- Robustness with reference to noise.
- Control rules defined over a wide range of applicability.
- Possibility to model heuristics from experts.
- Smoothness of action.
- Non-linearity.

The fuzzy system can also be used to make database queries that are flexible with human like sensibility. For example with fuzzy sets it is possible to make queries like ”Give me the names of the people that have recently invested a lot” and give a meaning to ”recently” and ”a lot”.

The fuzzy systems can be also used in Artificial Intelligence systems like Expert Systems, scheduling, and Decision Support Systems.

CHAPTER 4

Evidence theory

4.1 Fuzzy mathematics

Fuzzy numbers are fuzzy sets defined over the set of real numbers, which model our concept of approximate value. The constraints used to define fuzzy numbers are:

1. Normal fuzzy sets.

2. Convex fuzzy sets (all α-cut intervals should be closed).

3. The support of A should be bounded.

The first constraint captures the concept of approximate value corresponding to a number. The other two constraints are needed to define arithmetic.

With fuzzy sets it is possible to define fuzzy numbers (first graph), fuzzy intervals (second graph), defined intervals (third graph) and crisp numbers (fourth graph).

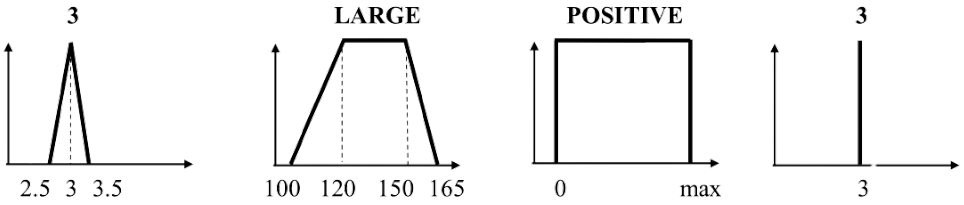


Figure 4.1: Possible representation of numbers

.

The arithmetic of fuzzy numbers is based on two properties:

- Each fuzzy number can be completely represented by its α-cuts uniquely.

- The α-cuts of fuzzy numbers are closed intervals of real numbers.

The four main operators are defined as the union of the operations on intervals (α-cut) that compose the fuzzy number and are:

- $[a, b] + [d, e] = [a + d, b + e]$.

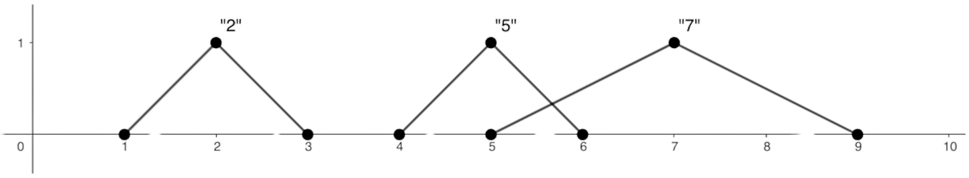
- $[a, b] - [d, e] = [a - e, b - d]$.
- $[a, b] \times [d, e] = [\min(ad, ae, bd, be), \max(ad, ae, bd, be)]$
- $[a, b] \div [d, e] = [\min(\frac{a}{d}, \frac{a}{e}, \frac{b}{d}, \frac{b}{e}), \max(\frac{a}{d}, \frac{a}{e}, \frac{b}{d}, \frac{b}{e})]$ with $[d, e] \neq [0, 0]$

Example : Given the fuzzy numbers $[1, 3]$ and $[4, 6]$ we have that:

- The sum of the numbers is given by the sum of the minimum and the maximum of each interval:

$$[1, 3] + [4, 6] = [1 + 4, 3 + 6] = [5, 9]$$

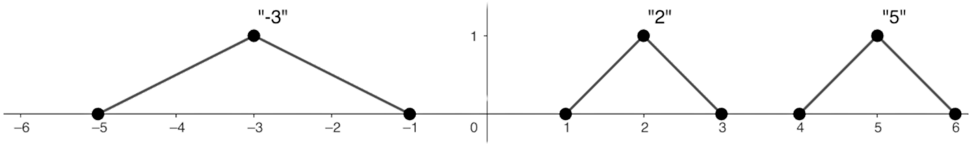
Graphically, we obtain the following graph:



- The subtraction of the numbers is given by the subtraction of the minimum of the first interval with the maximum of the second and the subtraction of the maximum of the first interval with the minimum of the second interval:

$$[1, 3] + [4, 6] = [1 - 6, 3 - 4] = [-5, -1]$$

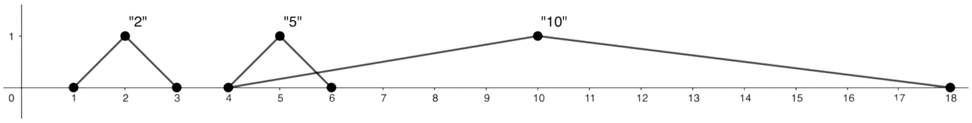
Graphically, we obtain the following graph:



- The multiplication of the numbers is given by the minimum and the maximum of the multiplication of each element:

$$[1, 3] \times [4, 6] = [\min(1 \cdot 4, 1 \cdot 6, 3 \cdot 4, 3 \cdot 6), \max(1 \cdot 4, 1 \cdot 6, 3 \cdot 4, 3 \cdot 6)] = [4, 18]$$

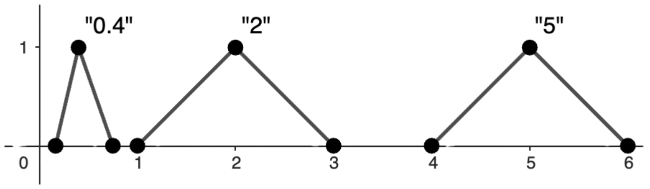
Graphically, we obtain the following graph:



- The division of the numbers is given by the minimum and the maximum of the division of each element:

$$[1, 3] \times [4, 6] = [\min(\frac{1}{4}, \frac{1}{6}, \frac{3}{4}, \frac{3}{6}), \max(\frac{1}{4}, \frac{1}{6}, \frac{3}{4}, \frac{3}{6})] = [\frac{1}{6}, \frac{3}{4}]$$

Graphically, we obtain the following graph:



From fuzzy arithmetic is also possible to define fuzzy functions, fuzzy integrals and fuzzy derivatives. In general, fuzzy numbers are used to represent approximation.

4.2 Fuzzy measure and probability assignment

Definition

If a field has the property that, if the sets $A_1, \dots, A_n$ belong to the fields, then also the union and the intersection of the sets belong to the field, this is named a *Borel field.* A function g defined on a Borel field B of the universe of discourse X is a *fuzzy measure* if it has the following properties:

1. $g(\varnothing) = 0$ and $g(X) = 1$.
2. If $A, B \in B$ and $A \subseteq B$, then $g(A) \leq g(B)$.
3. If $A_n \in B$ and $A_1 \subseteq A_2 \subseteq \dots$ then $\lim_{n\to\infty} g(A_n) = g(\lim_{n\to\infty} A_n)$.

The fuzzy measure is different from classical measure, since the additivity property is relaxed.

Definition

The *basic probabilistic assignment* is defined as:

- $m : \mathcal{P}(X) \to [0, 1]$.
- $m(\varnothing) = 0$.
- $\sum_{A\in\mathcal{P}(X)} m(A) = 1$.

Where m gives, for any set A belonging to the power set of $X(\mathcal{P}(X))$, how much the available and relevant evidence supports the fact that a given element belongs to A.

Note that: it is not needed that $m(X) = 1$, it is not needed that $m(A) \leq m(B)$ when $A \subseteq B$ and no relationship holds between $m(A)$ and $m(\neg A)$.

4.3 Evidence theory

We would like to define a measure of evidence for or against a proposition. To do that we will use two fuzzy measures: Belief and Plausibility.

Definition

Belief is an estimation of the minimum probability that can be assigned to an element, given the collected evidence. The *Belief* has the following definition:

- $Bel : \mathcal{P}(X) \to [0, 1]$.
- $Bel(\varnothing) = 0$ and $Bel(X) = 1$.
- $Bel(A_1 \cup A_2 \cup \dots \cup A_n) \geq \sum_j Bel(A_j) - \sum_{j<k} Bel(A_j \cap A_k) + \dots + (-1)^{n+1} Bel(A_1 \cap A_2 \cap \dots \cap A_n)$
- $Bel(A) + Bel(\neg A) \leq 1$.

Definition

Plausibility is an estimation of the maximum probability that can be assigned to an element, given the collected evidence. The *Plausibility* has the following definition:

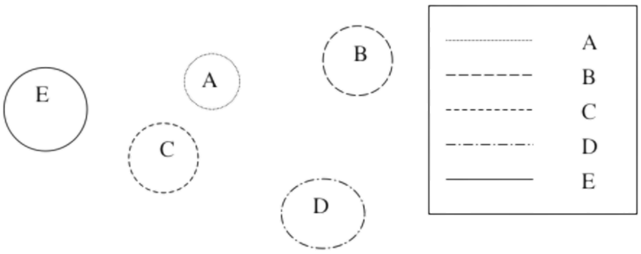
- $Pl : \mathcal{P}(X) \rightarrow [0, 1]$.
- $Pl(\varnothing) = 0$ and $Pl(X) = 1$.
- $Pl(A_1 \cap A_2 \cap \cdots \cap A_n) \geq \sum_j Pl(A_j) - \sum_{j<k} Pl(A_j \cup A_k) + \cdots + (-1)^{n+1} Pl(A_1 \cup A_2 \cup \cdots \cup A_n)$
- $Pl(A) + Pl(\neg A) \geq 1$.

The evidence theory can be used when we have multiple sources of knowledge and when the basic probability assignment is distributed on different sets of statements, or intervals. In these cases, we may exploit the features of evidence theory to collect the basic probability assignments and to combine them to evaluate upper and lower bounds for probability of a single statement. Evidence theory has the following implications:

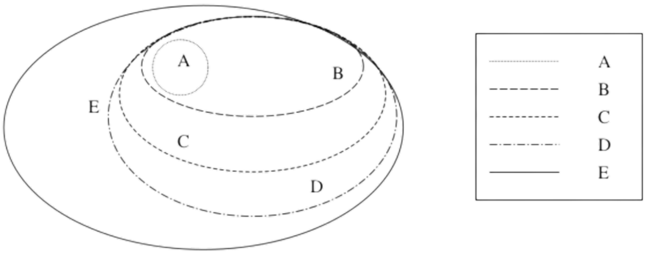
- It is not necessary to elicit a precise measurement from a knowledge source or an experiment if it is not realistic or feasible to do so.
- The principle of insufficient reason is not imposed. Statements can be made about the likelihood of multiple events together without having to resort to assumptions about the probabilities of the individual events under ignorance.
- The axiom of additivity is not imposed. The measures can:
  - Add to exactly one: it corresponds to a traditional probabilistic representation.
  - Add to less than one (sub-additive case): incompatibility between multiple sources of information providing conflicting information.
  - Add to more than one (super-additive): cooperative effect between multiple sources of information (multiple sensors providing the same information).

Given five sources (A, B, C, D, E where A is the target) of information it is possible to have four cases:

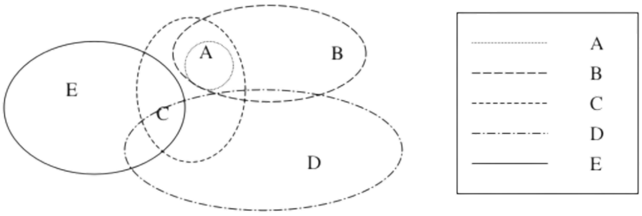
- Conflict: each source provides evidence for disjoint sets.



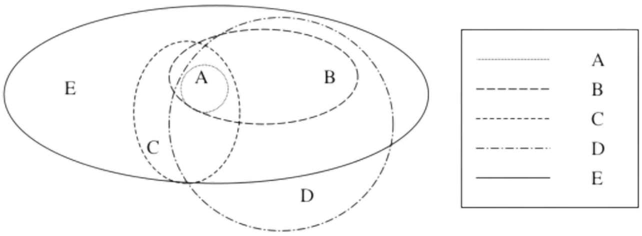
- Consonance: sources provide some evidence on nested sets converging on the target



- Arbitrary: each source provides evidence for sets, only some of which include the target hypothesis.



- Consistent: all sources provide some evidence for sets that include the same hypothesis.



To combine the basic probability assignments it is possible to use the Dempster rule of combination:

$$m_{1,2}(A) = \frac{\sum_{B \cap C = A} m_1(B) m_2(C)}{1 - K}$$

where K is the basic probability mass associated with conflict. Its role in the denominator has the effect of completely ignoring conflict and attributing any probability mass associated with conflict to the null set. The value of this variable is the following:

$$K = \sum_{B \cap C = O} m_1(B) m_2(C)$$

Klir, Yuan, 1995 : Let's assume that an old painting was discovered which strongly resembles paintings by Raphael. Such a discovery is likely to generate various questions regarding the status of the painting. Let's assume the following three questions:

1. Is the discovered painting a genuine painting by Raphael?
2. Is the discovered painting a product of one of Raphael's many disciples?
3. Is the discovered painting a counterfeit?

Let R, D, and C denote subsets of our universal set X-the set of all paintings, which contain the set of all paintings by Raphael, the set of all paintings by disciples of Raphael, and the set of all counterfeits of Raphael's paintings, respectively. Assume now that two experts performed careful examinations of the painting and subsequently provided us with basic assignments m_1 and m_2. By applying the above introduced formulas, it is possible to compute plausibility and belief of all the subsets of hypotheses.

4.4 Possibility and necessity

Definition

Possibility is another fuzzy measure working on sets. A *possibility measure* is given by the equation $\Pi : \mathcal{P}(X) \to [0,1]$, for which the following properties hols:

1. $\Pi(\varnothing) = 0$ and $\Pi(X) = 1$.
2. $A \subseteq B \implies \Pi(A) \leq \Pi(B)$.
3. $\Pi(A) = \sup_{x \in A} f(x)$ where $A \subset X$.

It can be uniquely defined by a possibility relationship $f : X \to [0,1]$ so that:

$$\Pi\left(\bigcup_{i \in I} A_i\right) = \sup_{i \in I} \Pi\left(A_i\right)$$

Therefore, f is defined as $\Pi\left(\{X\}\right)\ \forall x \in X$.

Possibility : Given the set $x = \{0,1,2,3,4,5,6,7,8,9,10\}$ and $\Pi(x)$, the possibility that x is close to the value 8:

x	0	1	2	3	4	5	6	7	8	9	10
$\Pi\left(\{X\}\right)$	0	0	0	0	0	0.1	0.5	0.8	1	0.8	0.5

Now we need to compute $\Pi(A)$ that is the possibility that A includes an integer close to 8. For a given set $A = \{2,5,9\}$, it is possible to compute its possibility:

$$\Pi(A) = \sup\left[\Pi\left(\{2\}\right), \Pi\left(\{5\}\right), \Pi\left(\{9\}\right)\right] = \sup\left[0, 0.1, 0.8\right] = 0.8$$

Definition

Necessity is the dual of possibility, and it is defined as:

$$\Pi(A) = N(\neg A)$$

Necessity also satisfy the condition:

$$\min\left[N(A), N(\neg A)\right] = 0$$

Those two parameters are connected by the following relations:

- $\Pi(A) \geq N(A)$.
- $N(A) > 0 \implies \Pi(A) = 1$.
- $\Pi(A) < 1 \implies N(A) = 0$.

Definition

The *confirmation degree* is a value that puts together possibility and necessity:

$$C(A) = N(A) + \Pi(A) - 1$$

Negatives values of $C(A)$ correspond to a disconfirmation degree.

It is possible to demonstrate that if the focal set of elements of possibility (the one with values of m different from 0) is composed of sets with a single element, then Bel and Pl have the same value and this is equal to the sum of the probabilities of the elements of the set A to which they are applied, given by m.

The evidence in the probabilistic model is on single elements, in the possibility model it can also be on sets.

Both have distribution functions, even if they are normalized differently: probabilities add up to one, for possibilities the maximum value is one.

In possibility theory ignorance is expressed by assigning all the evidence to the total set (i.e., anything is possible), in probability, instead, by distributing a uniform fraction of the evidence to each element (each element is equiprobable).

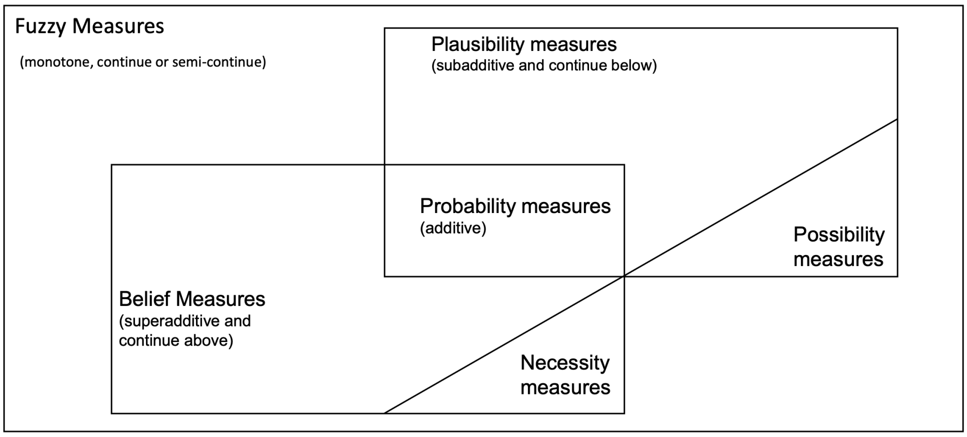


Figure 4.2: Inclusion relationships among fuzzy

4.5 Fuzzyness measure

The fuzzyness measures provide the fuzzyness degree of a fuzzy set. A fuzzyness measure is the entropy of a fuzzy set.

Definition

Given a fuzzy set $A = \{x, \mu_A(x)\}$ the fuzzyness measure (entropy) is defined as:

$$d(A) = K \sum_{i=1}^{n} S(\mu_A(x_i))$$

where $S(x)$ is the Shannon's function:

$$S(x) = -x \ln(x) - (1 - x) \ln(1 - x)$$

Example : Let's define A as "the set of integer close to ten" we have that:

x	7	8	9	10	11	12	13
$\mu_A(x)$	0.1	0.5	0.8	1	0.8	0.5	0.1

The entropy of this set has a value of:

$$d(A) = 0.325 + 0.693 + 0.673 + 0.501 + 0 + 0.501 + 0.693 + 0.325 = 3.711$$

Let's define B as "the set of integer quite close to ten" we have that:

x	6	7	8	9	10	11	12	13	14
$\mu_A(x)$	0.1	0.3	0.4	0.7	1	0.8	0.5	0.3	0.1

The entropy of this set has a value of:

$$d(A) = 4.35$$

It is possible to notice that B is more fuzzy than A, since $d(B) > d(A)$.

CHAPTER 5

Probabilistic reasoning

5.1 Basic probability

Definition

A is a *boolean-valued random variable* if A denotes an event, and there is some degree of uncertainty as to whether A occurs.

Probability of A is the fraction of possible worlds in which A is true.

Theorem

The axioms of the probability theory are:

- $0 \leq P(A) \leq 1$.
- $P(A = true) = 1 \ \wedge P(A = false) = 0$.
- $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$.

From the axioms it is possible to derive some other formula such as:

1. $P(\overline{A}) = 1 - P(A)$.
2. $P(A) = P(A \wedge B) + P(\overline{A} \wedge B)$

Definition

A *multivalued random variable* A is a random variable of arity k if it can take on exactly one values out of $\{v_1, v_2, v_3, \dots, v_k\}$.

Theorem

The axioms for multivalued random variable are:

- $P(A = v_i \wedge A = v_j) \quad i \neq j.$
- $P(A = v_1 \vee A = v_2 \vee A = v_3 \vee \cdots \vee A = v_k) = 1.$

With those new axioms it is possible to derive other formulas that are useful with multivalued variable:

1. $P(A = v_1 \vee A = v_2 \vee \cdots \vee A = v_i) = \sum_{j=1}^{i} P(A = v_j).$
2. $\sum_{j=1}^{k} P(A = v_j) = 1.$
3. $P(B \wedge (A = v_1 \vee A = v_2 \vee \cdots \vee A = v_i)) = \sum_{j=1}^{i} B \wedge A = v_j.$
4. $P(B) = \sum_{j=1}^{k} P(B \wedge A = v_j).$

Definition

The *conditional probability* of A given B is the fraction of possible worlds in which B is true that also have A true.

The inference can be done mainly using the following rules:

- Chain rule: $P(A \wedge B) = P(A|B)P(B)$
- Bayes theorem: $P(A|B) = \dfrac{P(B|A)P(A)}{P(B)}.$
- Sum rule (marginalization): $P(A) = \sum_b (A \wedge B = b).$

Definition

Assume A and B are boolean random variables; A and B are independent (denote it with $A \perp B$) if and only if:

$$P(A|B) = P(A)$$

Given two random variables A and B, the *joint distribution* of A and B is the distribution of A and B together: $P(A, B)$

It is possible to represent a joint distribution of M (binary) variables with the following steps:

1. Make a truth table listing all combination of values (2^M entries).
2. For each combination compute how probable it is.
3. Check that all probabilities sum up to one.

5.2 Probabilistic reasoning

The graphical models are used to describe factorization for the join distribution. The graphical models are used for backward reasoning mainly using the Bayes' theorem. The models used to compute the probability are graph-based algorithms. The graph used by these algorithms can of three types: directed, undirected, and factor.

Example : A fiendish murder has been committed. There are two suspects: the butler and the cook. There are three possible murder weapons: a butcher's knife, a pistol, and a fireplace poker.

We know that the butler has served family well for many years, and the cook, hired recently, rumors of dodgy history, so we can say that:

$$\text{P}(\text{Culprit} \rightarrow \textit{butler}) = 20\% \qquad \text{P}(\text{Culprit} \rightarrow \textit{cook}) = 80\%$$

Culprit is a binary random variable which probabilities add to 100%.

The butler is ex-army, and keeps a gun in a locker drawer. The cook has access to a lot of knives. The butler is older, and getting frail. We know that the weapons are:

$$\text{Weapon} = \{pistol, knife, poker\}$$

And given the evidence we can state that:

$$\text{P}(\text{Weapon}|\text{Culprit} \rightarrow \textit{butler}) = \begin{bmatrix} 80\% & 10\% & 10\% \end{bmatrix}$$

$$\text{P}(\text{Weapon}|\text{Culprit} \rightarrow \textit{cook}) = \begin{bmatrix} 5\% & 65\% & 30\% \end{bmatrix}$$

Using the chain rule we can finally compute the joint distribution, that is:

	pistol	**knife**	**poker**
cook	4%	52%	24%
butler	16%	2%	2%

Using the sum rule we can compute the marginal distribution of culprits: cook (80%), and butler (20%). And the marginal distribution of weapons: pistol (20%), knife (54%), and poker (26%). If at a certain point we discover the weapon, we can use the Bayes' theorem to compute who is the culprit.

5.3 Density estimation

To compute the probability for logic expression we can use the sum:

$$P(E) = \sum_{row \backsim E} P(row)$$

To compute the inference we can use the formula:

$$P(E_1|E_2) = \frac{P(E_1 \wedge E_2)}{E_2} = \frac{\sum_{row \backsim E_1 \wedge E_2} P(row)}{\sum_{row \backsim E_2} P(row)}$$

Example : Given the following truth table:

A	B	C	$\mathbf{P(A,B,C)}$
0	0	0	0.30
0	0	1	0.05
0	1	0	0.10
0	1	1	0.05
1	0	0	0.05
1	0	1	0.10
1	1	0	0.25
1	1	1	0.10

$P(A)$ can be found by summing all the probability where $A = 1$, that is:

$$P(A) = 0.05 + 0.10 + 0.25 + 0.10 = 0.5$$

$P(A \wedge B)$ can be found by summing all the probability where $A = 1$ and $B = 1$, that is:

$$P(A \wedge B) = 0.25 + 0.10 = 0.35$$

$P(\overline{A} \vee B)$ can be found by summing all the probability where $A = 0$ or $B = 1$, that is:

$$P(\overline{A} \vee B) = 0.30 + 0.05 + 0.10 + 0.05 + 0.25 + 0.10 = 0.85$$

$P(A|B)$ can be found by dividing the probability of $A = 1$ and $B = 1$ by the probability where $B = 1$, that is:

$$P(A|B) = \frac{(0.25 + 0.10)}{(0.10 + 0.05 + 0.25 + 0.10)} = 0.7$$

$P(C|A \wedge B)$ can be found by dividing the probability of $A = 1$ and $B = 1$ and $C = 1$ by the probability where $A = 1$ and $B = 1$, that is:

$$P(C|A \wedge B) = \frac{(0.10)}{(0.25 + 0.10)} = 0.285$$

$P(\overline{A}|C)$ can be found by dividing the probability of $A = 0$ and $C = 1$ by the probability where $C = 1$, that is:

$$P(\overline{A}|C) = \frac{(0.05 + 0.05)}{(0.05 + 0.05 + 0.10 + 0.10)} = 0.333$$

Definition

A *density estimator* learns a mapping from a set of attributes to a probability distribution over the attributes space:

$$M : \{0, 1\}^I \to [0, 1]$$

We can use likelihood for evaluating density estimation: given a record x, a density estimator M tells you how likely it is $\widehat{P}(x|M)$. Given a dataset with N records, a density estimator can tell how likely data is under the assumption that all records were independently generated from it:

$$\widehat{P}(\text{dataset}) = \widehat{P}(x_1, x_2, \ldots, x_N) = \prod_{n=1}^{N} \widehat{P}(x|M)$$

Since likelihood can get too small we usually use log-likelihood:

$$\log \widehat{P}(\text{dataset}) = \log \prod_{n=1}^{N} \widehat{P}(x_n|M) = \sum_{n=1}^{N} \log \widehat{P}(x_n|M)$$

Density estimators can do many good things: sort the records by probability, and thus spot weird records, inference and can be used for Bayes classifiers. The main problem about joint density estimators is that they can badly overfit.

The naïve Bayes estimator model assumes that each attribute is distributed independently of the other attributes. Let $x[i]$ denote the i^{th} field of record x, the naïve density estimator says that:

$$x[i] \perp \{x[1], x[2], \ldots, x[i-1], x[i], x[i+1], \ldots, x[I]\}$$

It is important to know that all attributes are equally important and that knowing one attribute says nothing about the value of another. This last assumption is almost never correct, but works quite well when used in practice.

Example: Given four variables A, B, C, D. For the naïve Bayes estimator model they are all independent thanks to the assumptions. This means that:

$$P(A, \overline{B}, C, \overline{D}) = P(A)P(\overline{B})P(C)P(\overline{D})$$

To learn a naïve Bayes estimator we have to suppose that $x[1], x[2], \ldots, x[n]$ are independently distributed. With this hypothesis it is possible to construct any row of the implied joint distribution on demand:

$$\widehat{P}(x[1] = u_1, x[2] = u_2, \ldots, x[I] = u_I) = \prod_{k=1}^{I} \widehat{P}(x[k] = u_k)$$

	Joint estimator	**Naïve estimator**
Modelling	Anything	Boring distributions
Attributes	Few Boolean	Many multivalued
Overfitting	Yes	Quite robust

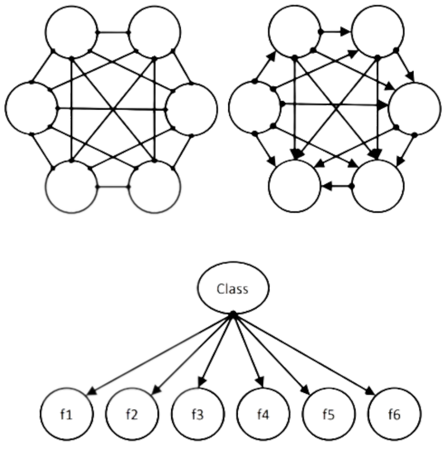


Figure 5.1: Comparison between a joint and a naïve estimator

CHAPTER 6

Graphical models

6.1 Introduction

In the real world the random variables are usually correlated. To overcome this fact it is possible to represent a probability distribution via:

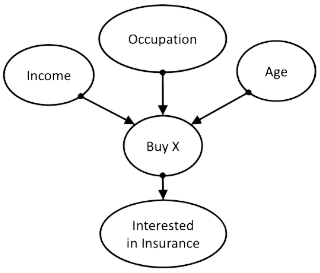
- Conditional independence assumptions that apply on a subset of them.
- A set of conditional probabilities with their priors.

The models based on these rules are called graphical models.

6.2 Bayesian network

A Bayesian network is a method to describe the joint probability distribution of a set of variables. It can be represented as a directed graph where the nodes represent random variables and the edges represent direct influence.

Example : Given the following Bayesian network:



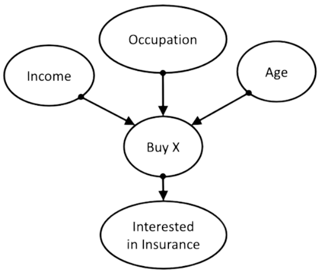
We can see that the random variables "age", "income", and "occupation" are independent while "buy X" and "Interested in insurance" are conditional probability distributions.

Let $x_1, x_2, \ldots, x_n$ be a set of variables. We have that a Bayesian network can tell any combination probability in which the full joint distribution will require:

$$2^N - 1 \text{ parameters}$$

To represent the probabilities in the network we need only the priors and conditional parameters. We simply multiply the number of nodes for 2^k, where k is the number of incoming edges, and we obtain the numbers of parameters needed.

Example : Given the following Bayesian network:



In this case we have that the full joint distribution will need:

$$2^N - 1 = 2^5 - 1 = 31 \text{ parameters}$$

And to represent the Bayesian network we will need

$$3 \cdot 2^0 + 1 \cdot 2^3 + 1 + \cdot 2^1 = 13 \text{ parameters}$$

Definition

We say X_1 is *conditionally independent* of X_2 given X_3 if the probability of X_1 is independent of X_2 given some knowledge about X_3:

$$P(X_1|X_2, X_3) = P(X_1|X_3)$$

The same can be said for a set of variables: X_1, X_2, X_3 is independent of Y_1, Y_2, Y_3 given Z_1, Z_2, Z_3:

$$P(X_1, X_2, X_3|Y_1, Y_2, Y_3, Z_1, Z_2, Z_3) = P(X_1, X_2, X_3|Z_1, Z_2, Z_3)$$

Example : Martin and Norman toss the same coin. Let be A “Norman’s outcome”, and B “Martin’s outcome”. Assume the coin might be biased; in this case A and B are not independent: observing that B is heads causes us to increase our belief in A being heads. So it holds:

$$P(A|B) \neq P(A)$$

Variables A and B are both dependent on C “The coin is biased towards Heads with probability θ”. Once we know for C then any evidence about B cannot change our belief about A:

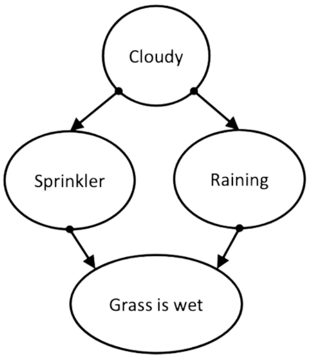
$$P(A|B, C) = P(A|C)$$

Definition

A *prior probability* is a probability with zero incoming edges.

Note that in a Bayesian network a node is independent of its ancestors given its parent.

Example : The event "Grass is wet" ($W = true$) has two possible causes: either the water "Sprinkler" is on ($S = true$) or it is "Raining" ($R = true$). The related Bayesian network is the following.



Where the probabilities for cloudy are:

Cloudy	**P(C)**
0	0.5
1	0.5

Where the probabilities for sprinkler are:

Sprinkler	**Cloudy**	**P(S|C)**
0	0	0.1
0	1	0.5
1	0	0.9
1	1	0.5

Where the probabilities for raining are:

Raining	**Cloudy**	**P(R|C)**
0	0	0.8
0	1	0.5
1	0	0.2
1	1	0.5

Where the probabilities for the wet grass are:

Wet	**Sprinkler**	**Raining**	**P(W|S,R)**
0	0	0	1
0	0	1	0.1
0	1	0	0.1
0	1	1	1
1	0	0	0
1	0	1	0.9
1	1	0	0.9
1	1	1	0.99

With all these values it is possible to compute all the probabilities with the formula:

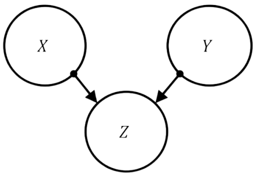
$$\begin{aligned}P(C,S,R,W) &= P(W|S,R,C)P(S,R,C) = \\ &= P(W|S,R)P(S,R,C) = \\ &= P(W|S,R)P(S|R,C)P(R,C) = \\ &= P(W|S,R)P(S|C)P(R,C) = \\ &= P(W|S,R)P(S|C)P(R|C)P(C)\end{aligned}$$

With this formula we can compute all the joint probabilities.

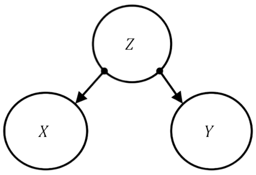
C	**S**	**W**	**R**	**P(C,S,W,R)**
0	0	0	0	0.04
0	0	0	1	0
0	0	1	0	0.001
0	0	1	1	0.009
0	1	0	0	0.036
0	1	0	1	0.324
0	1	1	0	0.0009
0	1	1	1	0.0891
1	0	0	0	0.125
1	0	0	1	0
1	0	1	0	0.0125
1	0	1	1	0.1125
1	1	0	0	0.125
1	1	0	1	0.1125
1	1	1	0	0.00125
1	1	1	1	0.12375

Explaining away is known in statistics as Berkson's paradox, or selection bias, and it describes two variable which become dependent because you observe a third one. In general, the independencies in a Bayesian network can be:

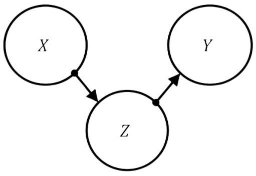
- $P(X,Y,Z) = P(X)P(Y)P(Z|X,Y)$ and $P(X,Y|Z) = P(X)P(Y)$ if the nodes are connected in the following way.



- $P(X,Y,Z) = P(X|Z)P(Y|Z)P(Z)$ and $P(X,Y|Z) = P(X|Z)P(Y|Z)$ if the nodes are connected in the following way.



- $P(X,Y,Z) = P(X)P(Z|X)P(Y|Z)$ and $P(X,Y|Z) = P(X|Z)P(Y|Z)$ if the nodes are connected in the following way.



Definition

> Two sets of nodes A and B are *conditionally independent* (also called d-separated) given C if and only if all the path from A to B are shielded by C.
>
> C is a *root* if C is hidden, children are dependent due to a hidden common cause. If C is observed, they are conditionally independent.
>
> C is a *leaf* if C is hidden, its parents are marginally independent, but if C, or any descendant, is observed parents become dependent.
>
> C is a *bridge*: nodes upstream and downstream of C are dependent if and only if C is hidden, because conditioning breaks the graph at that point.

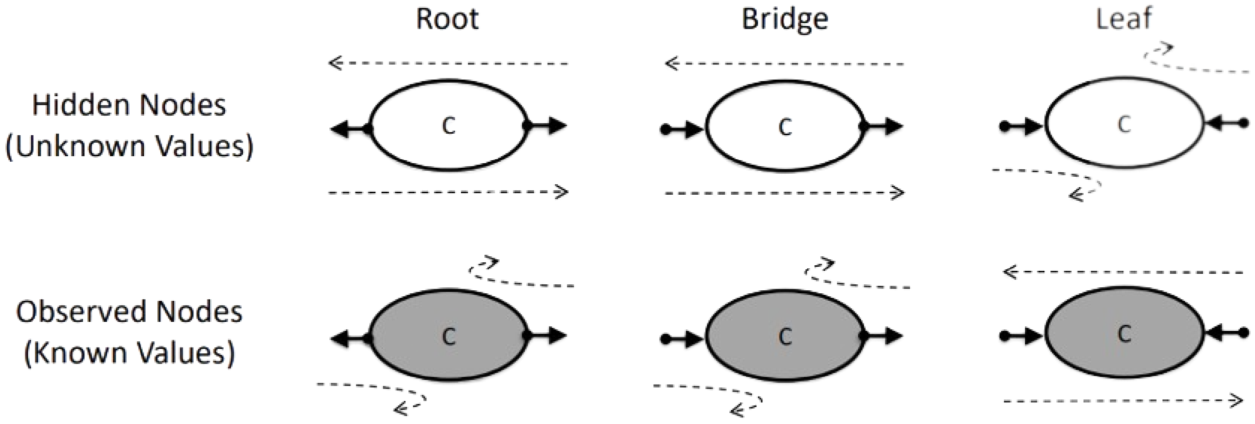


Figure 6.1: Graphical representation of root, bridge, and leaf

It is possible to make two types of reasoning with Bayesian networks:

- Bottom-Up: infer the cause given an evidence.
- Top-Down: we can compute the probability of an event given another event; this is predictive use of Bayesian Networks as "generative" models.

The most interesting property of Bayesian Networks is that they can be used to reason about causality on a solid mathematical basis. We can have Bayesian Networks with both real and discrete nodes. Using these we can obtain a rich toolbox for probabilistic modeling.

6.3 Introduction

The inference on a Bayesian network has an exponential complexity of $O(|X_i|^N)$. To reduce the complexity is possible to do:

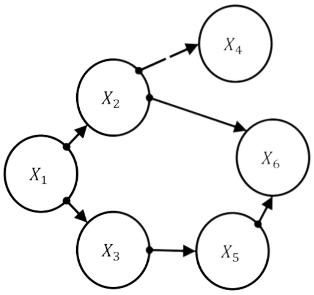
- Variable elimination.
- Belief propagation (message passing/sum product algorithm).
- Junction trees (will see only how to build).
- Loopy belief propagation.
- Sampling based methods.

The first three methods are exact, while the last two are approximate.

6.4 Variable elimination

We can use the factored representation of joint probability to do marginalization efficiently by pushing sums in as far as possible. As we perform innermost sums we create new terms.

Example : Given the following Bayesian network:



We have that:

$$\begin{aligned}
P(X_5) =& \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_4}\sum_{X_6} P(X_1)P(X_2|X_1)P(X_3|X_1)P(X_4|X_2)P(X_5|X_3)P(X_6|X_5,X_2)\\
=& \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}P(X_4|X_2)\\
=& \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}P(X_6|X_5,X_2)\\
=& \sum_{X_2}\sum_{X_3} P(X_5|X_3)\mu_1(X_2)\mu_2(X_5,X_2)\sum_{X_1}P(X_1)P(X_2|X_1)P(X_3|X_1)\\
=& \sum_{X_3} P(X_5|X_3)\sum_{X_2}\mu_1(X_2)\mu_2(X_5,X_2)\mu_3(X_2,X_3)\\
=& \sum_{X_3} P(X_5|X_3)\mu_4(X_3,X_5)\\
=& \mu_5(X_5)
\end{aligned}$$

The variable elimination procedure is based on dynamic programming. To apply this technique we have to divide the main problem in many small problems by using the factorization of the joint distribution. This factorization allows us to determine in which order variable elimination is efficient and to determine the functions for μ.

To automate this process we will use the factor graphs models, that is a form of graphical model in which the box notation indicates terms that depends on some variables.

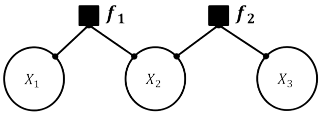


Figure 6.2: A simple example of a factor graph

The main properties of this type of graph are:

- It is a bipartite graph.
- Each circle node represents a random variable X_i.
- Each box node represents a factor f_k, which is a function $f_k(X_{C_k})$.
- The joint probability distribution is given as:

$$P(X_1, X_2, \ldots, X_N) = \prod_{k=1}^{K} f_k(X_{C_k})$$

To transform a Bayesian network into a factor graph we need to apply the moralization. In the general case this operation requires to create a link between the parents, maintaining the maximum number of independence properties

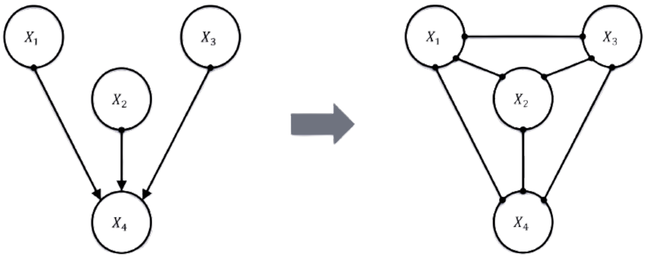
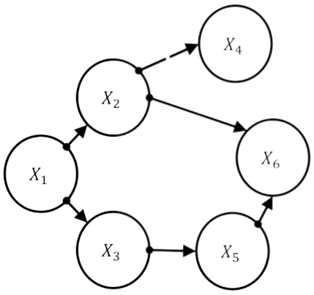


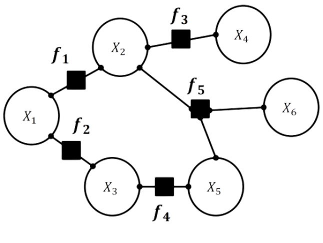
Figure 6.3: An example of moralization

For the transformation in factor graph we have to join unlinked parents into a single factor.

Example : Given the following Bayesian network:



The correspondent factor graph is:



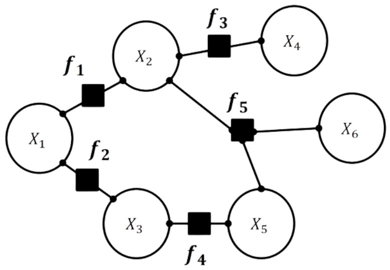
Definition

A graph is a *perfect map* if and only if every independence property of a distribution is reflected in the graph and vice versa.

Note that:

- Not all distributions can be represented as a directed/undirected graphs.
- Not all directed graphs can be represented as undirected graphs.
- Not all undirected graphs can be represented as directed graphs.

Example : The chain can be represented as a factor graph as follows:



- $f_1(X_1) = P(X_1)$
- $f_2(X_1, X_2) = P(X_1|X_2)$
- $f_3(X_2, X_3) = P(X_3|X_2)$
- $f_4(X_3, X_4) = P(X_4|X_3)$
- $f_5(X_4, X_5) = P(X_5|X_4)$
- $f_5(X_5, X_6) = P(X_6|X_5)$

Note that the factor graphs are not unique: every Bayesian network can be written in different ways.

The inputs of the variable elimination algorithm are: a list F of factors, and a tuple C_0 of output variables to keep. The output is a single factor m, over variables X_{C_0}.

Algorithm 1 Variable elimination algorithm

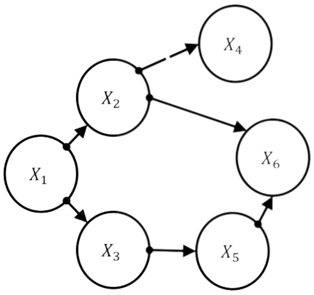
```
define all variables present in $F$ as $V \leftarrow \text{vars}(F)$
define all variables to be eliminated as $E \leftarrow V - C_0$
for all $i \in E$ do
    call eliminate_single_variable$(F, i)$
end for
for all remaining factors do
    $m \leftarrow \prod_{f \in F} f$
end for
```

The function eliminate_single_variable(F, i) of the previous algorithm takes as inputs the list F of factors and the variable with identifier i. The output is the list F of factors.

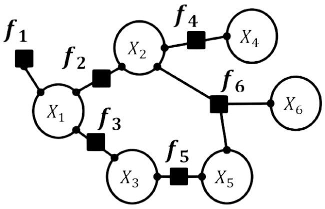
Algorithm 2 eliminate_single_variable(F, i)

```
find relevant subset $f \subset F$ of factors over $i$ as $f \leftarrow \{C|i \in C\}$
define the remaining clique as $C_t \leftarrow \text{vars}(f) - \{i\}$
compute the temporary factor as $\mu(X_{C_t}) = \sum_{X_i} \prod_{f \in F} f$
remove old factors $f$ and append new temporary factor $t$ to $F$
return $F$
```

Example : Given the following Bayesian network:



We can compute the factor graph with moralization, obtaining:



Compute the marginal $P(X_1, X_6) = \mu(X_1, X_6)$. The input of the algorithm will be:

- $F = \{f_1, f_2, f_3, f_4, f_5, f_6\}$
- $C_0 = \{X_1, X_6\}$

At the first step the algorithm defines all the variables present in F, that is:

$$V = \{X_1, X_2, X_3, X_4, X_5, X_6\}$$

At the second step it computes the set of variables to be eliminated:

$$E = V - C_0 = \{X_1, X_2, X_3, X_4, X_5, X_6\} - \{X_1, X_6\} = \{X_2, X_3, X_4, X_5\}$$

Now we have to eliminate all the single variables that are contained in the set E with the function eliminate_single_variable.

- eliminate_single_variable($F, 4$). For the first step we check the connected functions:

  $$f = \{f_4\}$$

  The clique containing the node X_4 (except itself) is:

  $$C_t = \{X_2\}$$

  The temporary factor appended to F is:

  $$\mu_1(X_2) = \sum_{X_4} P(X_4|X_2)$$

So, the set now become:

$$F = \{f_1, f_2, f_3, f_5, f_6, \mu_1\}$$

We now remove the factors f from E, that is now $E = \{X_2, X_3, X_5\}$

- eliminate_single_variable($F, 3$). For the first step we check the connected functions:

  $$f = \{f_3, f_5\}$$

  The clique containing the node X_3 (except itself) is:

  $$C_t = \{X_1, X_5\}$$

  The temporary factor appended to F is:

  $$\mu_2(X_1, X_5) = \sum_{X_3} P(X_3|X_1)P(X_5|X_3)$$

  So, the set now become:

  $$F = \{f_1, f_2, f_6, \mu_1, \mu_2\}$$

  We now remove the factors f from E, that is now $E = \{X_2, X_5\}$

- eliminate_single_variable($F, 5$). For the first step we check the connected functions:

  $$f = \{f_6, \mu_2\}$$

  The clique containing the node X_5 (except itself) is:

  $$C_t = \{X_1, X_2, X_6\}$$

  The temporary factor appended to F is:

  $$\mu_3(X_1, X_2, X_6) = \sum_{X_5} \mu_2(X_1, X_5)P(X_6|X_2, X_5)$$

  So, the set now become:

  $$F = \{f_1, f_2, \mu_1, \mu_3\}$$

  We now remove the factors f from E, that is now $E = \{X_2\}$

- eliminate_single_variable($F, 2$). For the first step we check the connected functions:

  $$f = \{f_2, \mu_1, \mu_3\}$$

  The clique containing the node X_2 (except itself) is:

  $$C_t = \{X_1, X_6\}$$

  The temporary factor appended to F is:

  $$\mu_4(X_1, X_6) = \sum_{X_2} \mu_1(X_2)P(X_2|X_1)\mu_3(X_1, X_2, X_6)$$

  So, the set now become:

  $$F = \{f_1, \mu_4\}$$

  We now remove the factors f from E, that is now $E = \varnothing$

After all these iterations we have that:

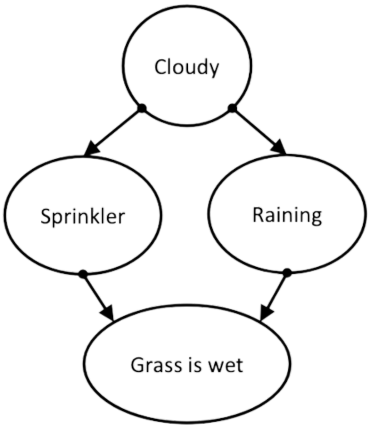
- $F = \{f_1, \mu_4\}$
- $C_0 = \{X_1, X_6\}$

The last steps require to multiply all the elements in f:

$$P(X_1, X_6) = \mu_4 \cdot f_1 = P(X_1)\mu_4(X_1, X_6)$$

Note that the order of the variables used as input for the function liminate_single_variable can be chosen with a heuristic function (we can choose the nodes ordered by ascending numbers of connections).

Example : Given the following Bayesian network.



And the following truth tables.

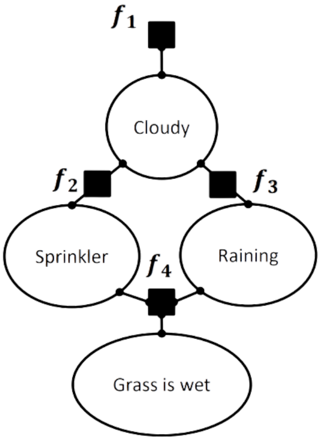
Cloudy	P(C)
0	0.5
1	0.5

Sprinkler	Cloudy	P(S|C)
0	0	0.1
0	1	0.5
1	0	0.9
1	1	0.5

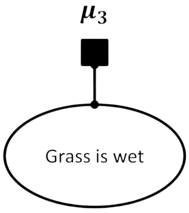
Raining	Cloudy	P(R|C)
0	0	0.8
0	1	0.5
1	0	0.2
1	1	0.5

Wet	**Sprinkler**	**Raining**	**P(W|S,R)**
0	0	0	1
0	0	1	0.1
0	1	0	0.1
0	1	1	1
1	0	0	0
1	0	1	0.9
1	1	0	0.9
1	1	1	0.99

We want to compute $P(W)$. To do so we start by transforming the Bayesian network into a factor graph where we put $f_1(C) = P(C), f_2(S, C) = P(S|C), f_3(R, C) = P(R|C), f_4(W, S, R) = P(W|S, R)$.



After applying the variable elimination algorithm we obtain the following graph:



With the corresponding truth table.

Wet	$\boldsymbol{\mu_3}$**(W)**
0	0.22915
1	0.77085

The variable elimination has good properties such as: it is very simple to implement, does exactly what you would do on paper, and its complexity with optimal ordering complexity is $O(N2^K)$. But it also has some drawbacks: finding the optimal ordering is an $\mathcal{NP}$-hard problem, it computes only one marginal at the time, and requires N executions to compute all marginals.

6.5 Belief propagation