An empirical classification-based framework for the safety criticality assessment of energy production systems, in presence of inconsistent data

Tairan Wang, Vincent Mousseau, Nicola Pedroni, Enrico Zio

An empirical classification-based framework for the safety criticality assessment of energy production systems, in presence of inconsistent data

Tai-Ran WANG[a], Vincent MOUSSEAU[b], Nicola PEDRONI[a], Enrico ZIO[a,c]

[a]*Chair System Science and the Energy Challenge, Fondation Electricité de France (EDF), CentraleSupélec, Université Paris Saclay, Grande Voie des Vignes, 92290 Chatenay-Malabry, France*

[b]*Laboratoire Genie Industriel, CentraleSupélec, Université Paris-Saclay, Grande Voie des Vignes, 92290 Chatenay-Malabry, France*

[c]*Politecnico di Milano, Energy Department, Nuclear Section, c/o Cesnef, via Ponzio 33/A , 20133, Milan, Italy, Fax: 39-02-2399.6309, Phone: 39-02-2399.6340, enrico.zio@polimi.it*

ABSTRACT

The technical problem addressed in the present paper is the assessment of the safety criticality of energy production systems. An empirical classification model is developed, based on the Majority Rule Sorting method, to evaluate the class of criticality of the plant/system of interest, with respect to safety. The model is built on the basis of a (limited-size) set of data representing the characteristics of a number of plants and their corresponding criticality classes, as assigned by experts.

The construction of the classification model may raise two issues. First, the classification examples provided by the experts may contain contradictions: a validation of the consistency of the considered dataset is, thus, required. Second, uncertainty affects the process: a quantitative assessment of the performance of the classification model is, thus, in order, in terms of accuracy and confidence in the class assignments.

In this paper, two approaches are proposed to tackle the first issue: the inconsistencies in the data examples are "resolved" by deleting or relaxing, respectively, some constraints in the model construction process. Three methods are proposed to address the second issue: (i) a model retrieval-based approach, (ii) the Bootstrap method and (iii) the cross-validation technique.

Numerical analyses are presented with reference to an artificial case study regarding the classification of Nuclear Power Plants.

KEYWORDS: Safety-criticality, classification model, data consistency validation, confidence estimation, MR-Sort, nuclear power plants

1. INTRODUCTION

The ever-growing attention to Energy and Environmental (E&E) issues has led to emphasizing a systemic view of the trilemma of energy systems' safety and security, sustainable development and cost effectiveness [(1)]. In particular, the assessment of the level of criticality of existing energy production systems in relation to safety is strongly demanded. This has sparked a number of efforts to guide designers, managers and stakeholders in (i) the definition of the criteria for the evaluation of safety criticality, (ii) its qualitative and quantitative assessment [(2)(3)] and (iii) the selection of actions to reduce criticality. In this paper, we mainly address the central issue (ii) above, i.e., the quantitative assessment of the level of safety-related criticality of energy production systems. We use Nuclear Power Plants (NPPs) as reference systems, as the study is motivated by the need of the Research and Development (R&D) Department of Industrial Risk Management of Electricité de France (EdF) of developing a methodology for aiding decisions on the selection of alternative safety barriers, maintenance options etc, which have an impact on different system attributes and performance indicators.

In practice, it is unavoidable that the analysis of the safety criticality of a system be affected by uncertainty [(4)], due to the long time frame considered, the intensive investment of capital and the involvement of multiple stakeholders with different views and preferences [(5)(6)]. Thus, it is difficult to proceed with traditional risk/safety assessment methods, such as statistical analysis or probabilistic modeling [(7)].

In this paper, we adopt an empirical classification approach and develop a classification model based on the Majority Rule Sorting (MR-Sort) method [(10)] (which is a simplified version of the ELECTRE-Tri method [(8)(9)]). The MR-Sort classification model contains a set of parameters that have to be calibrated based on a set of *empirical* classification examples (also called training set), i.e., a set of systems (called alternatives in the terminology of the method) with known classifications to which correspond criticality classes, as assigned by *experts*.

Two practical issues may arise in the construction of the classification model. First, the classification provided by the experts on the systems of the training set may contain contradictions: a validation of the consistency of the dataset is, thus, required. In this paper, two approaches are introduced to address this issue: the inconsistencies in the training data are "resolved" by *deleting* or *relaxing*, respectively, some constraints in the process of model construction [(10)]. Second, due to the finite (typically small) size of the training set of classification examples usually available for the analysis of real systems, the performance of the classification model may be affected by: (i) a low (resp., high) classification *accuracy* (resp., error); (ii) significant uncertainty, which affects the *confidence* of the classification-based evaluation model. In our work, we define the confidence in a classification assignment as in Ref. 10, i.e., as the probability that the class assigned by the model to a system is correct. The performance of the classification model (i.e., the classification accuracy – resp., error – and the confidence in the classification) needs to be assessed: this is of paramount importance for taking robust decisions informed by the evaluation of the level of safety criticality [(11)(12)]. In this paper, three different approaches are proposed to assess the performance of a classification-based MR-Sort evaluation model in the presence of small training datasets. The first is a model-retrieval based approach [(10)], which is used to assess the expected percentage error in assigning new alternatives. The second is Cross-Validation (CV): a given number of alternatives from the entire dataset is randomly selected to form the training set and generate the corresponding model, which is, then, used to classify the rest of the alternatives in the dataset. By so doing, the expected percentage model error is estimated as the fraction of alternatives incorrectly assigned (as an average over the left-out data). The third, is based on *bootstrapping* the available training set in order to build an ensemble of evaluation models [(13)]; the method can be used to assess both the accuracy and the confidence of the model: in particular, the confidence in the assignment of a given alternative to a class is given in terms of the full (probability) distribution of the possible classes for that alternative (built on the bootstrapped ensemble of evaluation models) [(14)].

The methods are applied on an exemplificative case study concerning the assessment of the overall level of safety criticality of NPPs: the characteristics of the plants as well as their categorizations are provided by experts of the R&D Department of Industrial Risk Management of EdF.

The contribution of this work is threefold:

- classification models are used in a variety of fields including finance, marketing, environmental and energy management, human resources management, medicine, risk analysis, fault diagnosis etc. [(15)]: to the best of the authors' knowledge, this is the first time that a classification-based framework is applied for the evaluation of the safety-related criticality of complex energy production systems (e.g., Nuclear Power Plants);
- two approaches are developed for the verification of the consistency of the classification examples provided by the experts: on the basis of the verification, the training dataset is modified before model construction;
- to the best of the authors' knowledge, it is the first time that the confidence in the assignments provided by an MR-Sort classification model is quantitatively assessed by the bootstrap method, in terms of the probability that a given alternative is correctly classified.

The paper is organized as follows. The next Section presents the basic framework for system criticality evaluation. Section 3 shows the classification model applied within the proposed framework. Section 4 describes the learning process of a classification model by the disaggregation method. Section 5 deals with the inconsistency study of the pre-assigned dataset. In Section 6, three approaches are proposed to analyze the performance of the classification model. Then, the proposed approaches are applied in Section 7 to a case study involving a set of nuclear power plants. Finally, Sections 8 and 9 present the discussion of the results and the conclusions of this research, respectively.

2 GENERAL FRAMEWORK FOR THE EVALUATION OF SYSTEM SAFETY-RELATED CRITICALITY

Without loss of generality, we consider that the overall level of criticality of the system can be characterized in terms of a set of six criteria $x' = \{x_1', x_2', x_3', x_4', x_5', x_6'\}$: its level of safety, its level of security and protection, its possible impact on the environment, its long-term performance, its operational performance and its possible impact on the communication and reputation of the operating company (Figure 1.). These six criteria are used as the basis to assess the level of criticality of the system. Each criterion is evaluated by experts in 4 grades, ranging from best (grade '0') to worst (grade '3'). Further details about the "scoring" of the criticality of each criterion are given in Appendix A. Four levels (or categories) of criticality are considered: satisfactory (0), acceptable (1), problematic (2) and serious (3). Then, the assessment of the level of criticality can be performed within a classification framework: find the criticality category (or class) corresponding to the evaluation of the system in terms of the six criteria above. A description of the algorithm used to this purpose is given in the following Section.

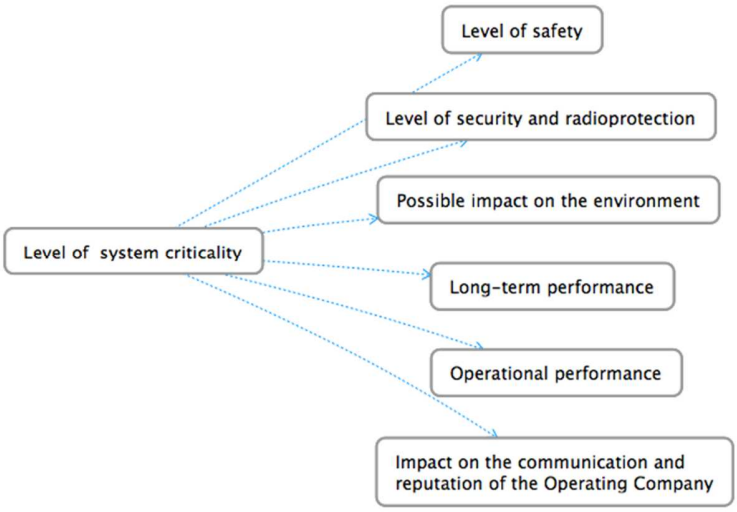

Figure 1. Criteria used to characterize the overall level of criticality of an energy production system or plant.

3 CLASSIFICATION MODEL FOR THE EVALUATION OF THE SYSTEM CRITICALITY: THE MAJORITY RULE SORTING (MR-SORT) METHOD

The Majority Rule Sorting Model (MR-Sort) method is a simplified version of ELECTRE Tri, an outranking sorting procedure in which the assignment of an alternative to a given category is determined using a complex concordance non-discordance rule [(8)(9)]. We assume that the alternative to be classified (in this paper, a safety-critical energy production system, e.g., a nuclear power plant) can be evaluated with respect to an *n*-tuple of elements $x' = \{x_1', x_2', x_3', x_4', x_5', x_6'\}$ (see the previous Section 2 and Figure 1), in 4 grades, from best ('0') to worst ('3'). In the present paper, the n=6 criteria used to evaluate the safety-related criticality of NPPs include safety, security, impact on the environment etc, as described in Section 2 and shown in Figure 1).

The MR-Sort model allows assigning an alternative $x = \{x_1, x_2, ..., x_i, ..., x_n\} \in X = X_1 \times X_2 \times ... \times X_i \times ... \times X_n$ to a particular pre-defined category (in this paper, a class of overall criticality), in a given ordered set of categories, $\{A^h : h = 1, 2, ..., k\}$. As mentioned in Section 2, k = 4 categories are considered in this work: A^1 = satisfactory, A^2 = acceptable, A^3 = problematic, A^4 = serious.

The model is further specialized in the following way:

-We assume that X_i is a subset of $\mathcal{R}$ for all $i \in N$ and the sub-intervals $(X_i^1, X_i^2, ..., X_i^h, ..., X_i^k)$ of X_i are compatible with the order on the real numbers, i.e., for all $x_i^1 \in X_i^1, x_i^2 \in X_i^2, ..., x_i^h \in X_i^h, ..., x_i^k \in X_i^k$, we have $x_i^1 > x_i^2 > ... > x_i^h > ... > x_i^k$. We assume, furthermore, that each interval $X_i^h, h = 2, 3, ..., k$ has a smallest element b_i^h, which implies that $x_i^{h-1} \geq b_i^h > x_i^h$. The vector $b^h = \{b_1^h, b_2^h, ..., b_i^h, ..., b_n^h\}$ (containing the lower bounds of the intervals X_i^h of criteria $i = 1, 2, ..., n$ in correspondence of category *h*) represents the lower limit profile of category A^h.

-There is a weight ω_i associated with each criterion $i = 1, 2, ..., n$, quantifying the relative importance of criterion *i* in the evaluation assessment process; notice that the weights are normalized such that $\sum_{i=1}^{n} \omega_i = 1$.

In this framework, a given alternative $x = \{x_1, x_2, ...x_i, ..., x_n\}$ is assigned to category $A^h, h = 1, 2, ..., k$, iff

$$\sum_{i \in N: x_i \geq b_i^h} \omega_i \geq \lambda \quad \text{and} \quad \sum_{i \in N: x_i \geq b_i^{h+1}} \omega_i < \lambda, \qquad (1)$$

where λ is a threshold $(0 \leq \lambda \leq 1)$ chosen by the analyst. Parameter λ can be considered as an

indicator of how confident the experts would like to be in the assignment: the higher the value of λ the stronger the evidence supporting the assignment needs to be. Actually, rule *(1)* is interpreted as follows. An alternative x belongs to category A^h if: (1) its evaluations in correspondence of the *n* criteria (i.e., the values $\{x_1, x_2, ..., x_i, ..., x_n\}$) are at least as good as b_i^h (lower limit of category A^h with respect to criterion *i*), $i = 1, 2, ..., n$, on a subset of criteria that has sufficient importance (in other words, on a subset of criteria that has a "total weight" larger than or equal to the threshold λ chosen by the analyst); and at the same time (2) the total weight of the subset of criteria on which the evaluations $\{x_1, x_2, ..., x_i, ..., x_n\}$ are at least as good as b_i^{h+1} (lower limit of the successive category A^{h+1} with respect to criterion *i*), $i = 1, 2, ..., n$, is not sufficient to justify the assignment of x to the successive category A^{h+1}. Notice that alternative x is assigned to the best category A_1 if $\sum_{i \in N: x_i \geq b_i^1} \omega_i \geq \lambda$ and it is assigned to the worst category A_k if $\sum_{i \in N: x_i \geq b_i^{k-1}} \omega_i < \lambda$.

The parameters of the model are the $(k-1) \cdot n$ lower limit profiles (*n* limits for the *k-1* categories, since the worst category does not need one), the *n* weights of the criteria $\omega_1, \omega_2, ..., \omega_i, ..., \omega_n$, and the threshold λ, for a total of $k \cdot n + 1$ parameters.

For illustration purpose, a numerical example of category assignment with *n=6* criteria and *h=2* categories is described in what follows, as shown in Figure 2:

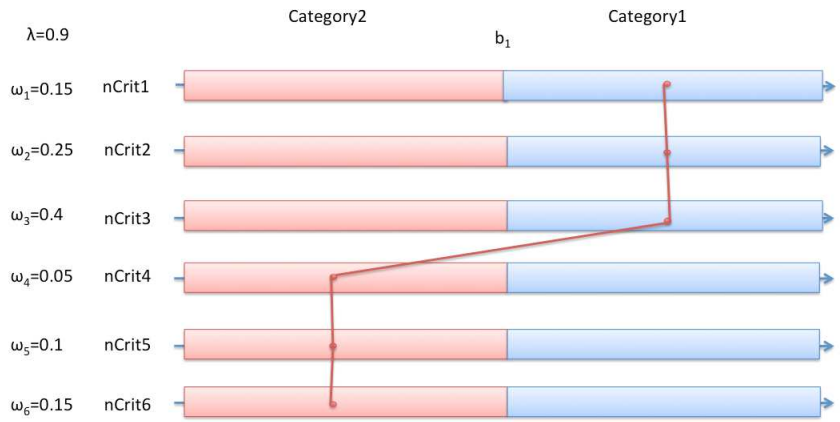

Figure 2. Representation of illustrative example of MR-Sort model

For each of the *n=6* criteria, a weight $\{\omega_i, i = 1,2, \dots ,6\}$ is assigned to represent its importance. The lower bound b_1 is used to "separate" the *h=2* categories. The points connected by lines represent the values (x_i) of the 6 criteria describing the alternative to be classified. In order to judge if this alternative can be assigned to "Category1" (best category, as indicated by the arrows), we have to compare the value of the threshold $\lambda = 0.9$ with the sum of the weights (ω_i) of the corresponding points (criteria) that are larger than the profile b_1. If the sum is larger, then the alternative should be assigned to the best category, "Category1", otherwise "Category2". In this particular case (Figure 2), the sum ($\omega_1 + \omega_2 + \omega_3$ =0.15+0.25+0.4=0.8) is smaller than the pre-defined threshold λ $(= 0.9)$: the alternative is, thus, assigned to "Category2".

4 CONSTRUCTING THE MR-SORT CLASSIFICATION MODEL

In order to construct an MR-Sort classification model, we need to determine the set of $k \cdot n + 1$ parameters, i.e., the weights $\omega = \{\omega_1, \omega_2, \dots, \omega_n\}$, the lower profiles $b = \{b^1, b^2, \dots, b^h, \dots, b^k\}$, with $b^h = \{b_1^h, b_2^h, \dots, b_i^h, \dots, b_n^h\}, h = 1, 2, \dots, k$, and the threshold λ; in this paper, λ is considered a fixed, constant value chosen by the analyst (e.g., λ=0.9 provides a strong confidence in the assignments, as suggested in [6]).

To this aim, the expert provides a training set of "classification examples" $D_{TR} = \{(x_p, \Gamma_p^t), p = 1, 2, \dots, N_{TR}\}$, i.e., a set of N_{TR} alternatives (in this case, NPPs of given, known characteristics) $x_p = (x_1^p, x_2^p, \dots x_i^p, \dots, x_n^p)$, $p = 1, 2, \dots, N_{TR}$, together with the corresponding real pre-assigned categories (i.e., criticality classes) Γ_p^t (the superscript 't' indicates that Γ_p^t represents the true, a priori-known class of alternative x_p).

The calibration of the $k \cdot n$ parameters is done through the learning process detailed in [6]. In extreme synthesis, the information contained in the training set D_{TR} is used to restrict the set of MR-Sort models compatible with such information, and to finally select one among them [6]. The a priori-known assignments generate constraints on the parameters of the MR-Sort model. In [6], such constraints have a linear formulation and are integrated into a Mixed Integer Program (MIP) that is designed to select one (optimal) set of such parameters ω^* and b^* (in other words, to select one classification model $M(\cdot \mid \omega^*, b^*)$)

that is coherent with the data available and maximizes a defined *objective function*. In [6], the optimal parameters $\omega^*$and b^* are those that maximize the value of the minimal slack in the constraints generated by the given set of data D_{TR}. Once the (optimal) classification model $M(\cdot \mid \omega^*, b^*)$ is constructed, it can be used to assign a new alternative x (i.e., a new nuclear power plant) to one of the performance classes $A^h, h = 1, 2, ..., k$: in other words, $M(x \mid \omega^*, b^*) = \Gamma_x^M$ where Γ_x^M is the class assigned by model $M(\cdot \mid \omega^*, b^*)$ to alternative x and assumes one value among $\{A^h : h = 1, 2, ..., k\}$. Further mathematical details about the training algorithm are not given here for brevity: the reader is referred to [6] for more detailed information.

There are two main issues related to this disaggregation process and to the construction by the MR-Sort classification model. First, for the given set of pre-assigned alternatives, it is possible that some of the class assignments are not consistent, due to fact that different experts may give different judgments (which causes an internal inconsistency); for obtaining a compatible classification model, the given training dataset must be made consistent. Second, in most real applications, because of the finite (and typically small) number N_{TR} of classification examples available, the model $M(\cdot \mid \omega^*, b^*)$ can only give a partial representation of reality and its class assignments are affected by uncertainty, which needs to be quantified to build confidence in the decision process based on the criticality level assessment.

In the following Section, the methods used in this paper to study the consistency of a given training dataset are described in detail; then, in Section 6 three different methods are presented to assess the performance of the MR-sort classification model.

5 CONSISTENCY STUDY: VALIDATION AND MODIFICATION OF THE SET OF CLASSIFIED ALTERNATIVES PRE-ASSIGNED BY EXPERTS

As explained before, a sorting model assigns alternatives to ordered categories based on the evaluation of a set of criteria. To develop such a model, it is necessary to set the values of the preference parameters used in the model, by inference from class assignment examples provided by experts. However,

assignment examples provided by experts can be *inconsistent* under two perspectives: either the examples provided contradict each other, or it is the preference model that is not flexible enough to account for the way alternatives are classified. In the first case, the expert would acknowledge a misjudgment and would agree to reconsider his/her examples; in the second case, the expert would not agree to change the examples and the preference model should be changed. In both cases, we refer to an inconsistency situation. In any case, the expert needs to know what causes inconsistency, i.e., which judgments should be changed if the aggregation model is to be kept (which is the perspective taken in our case) [(16)].

The MIP algorithm summarized in the previous Section may prove infeasible in case the class assignments of the alternatives in the training set are incompatible with all MR-sort models. In order to help the experts to understand how their inputs are conflicting and to question previously expressed judgments to learn about their preferences as the interactive process evolves, we formulate two MIPs that are able to: (i) find one MR-sort model that maximizes the number of training set alternatives correctly classified and (ii) propose accordingly a possible modification for each of the conflicting alternatives.

5.1 Inconsistency resolution via constraints deletion

Resolving the inconsistencies can be performed by deleting a subset of constraints related to the inconsistent alternatives. As shown in Figure 3, each alternative x_p can provide one or two constraints with respect to its assignment: for example, alternatives assigned to extreme categories, i.e., A_1 and A_4, provide one constraint, whereas alternatives assigned to intermediate categories, i.e., A_2 and A_3, introduce two constraints. Let us introduce a binary variable γ_p for each alternative x_p, which is equal to “1” if *all* the constraints associated to x_p are fulfilled, and equal to “0” otherwise.

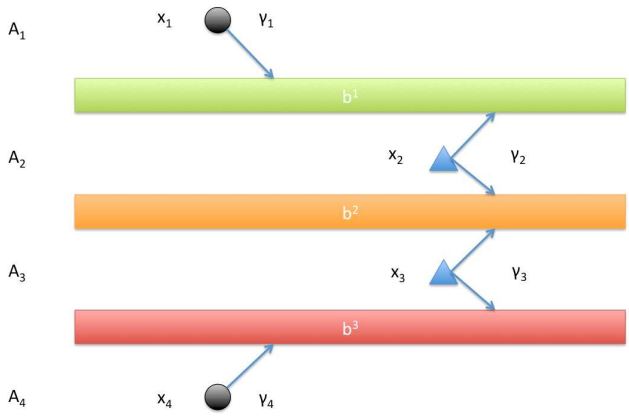

Figure 3. Representation of constraints deletion algorithm

The algorithm proceeds by "deleting" (i.e., removing) those constraints (i.e., those alternatives) that do not allow the creation of a compatible classification model, while maximizing the number of alternatives retained in the training set (i.e., minimizing the number of alternatives that are not taken into account): by so doing, we maximize the quantity of information that can be used to generate a classification model correctly. In other words, we obtain a MIP that yields a subset $D^*_{TR} \subseteq D_{TR}$ of maximal cardinality that can be represented by an MR-sort model. The reader is referred to [16] for more mathematical details.

5.2 Inconsistency resolution via constraints relaxation

Based on the algorithm presented in the previous subsection, a subset of maximal cardinality that can be represented by an MR-sort model is obtained. At the same time, its complementary set is *deleted*. However, in order to help the experts understand in what way the identified inconsistent inputs conflict with the others, and guide them to reconsider and possibly modify their judgments, a constraints relaxation algorithm is here proposed.

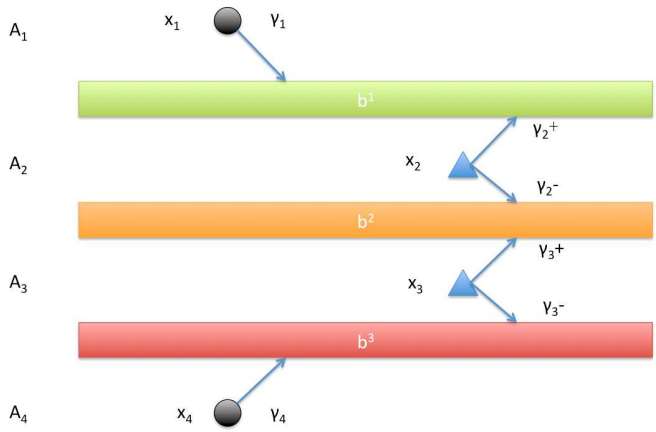

Figure 4. Representation of constraints relaxation algorithm

As presented in Section 5.3, each alternative x_p can provide one or two constraints with respect to its assignment. As presented in Figure 4, we introduce the following binary variables: γ_p, for the alternatives originally assigned to extreme categories, i.e., A_1 and A_4; γ_p^+ and γ_p^- for the alternatives originally assigned to intermediate categories, i.e., A_2 and A_3: In particular, γ_p^+ refers to the fulfillment of the constraint associated to the best category low profiles, whereas γ_p^- refers to the fulfillment of the constraint associated to the worst category low profiles.

As in the previous case, the algorithm identifies a subset $D_{TR}^* \subseteq D_{TR}$ of maximal cardinality that can generate an MR-sort model with proper formulation. In addition, for each of the alternatives that are not accepted into the subset D_{TR}^*, the corresponding inconsistent constraints are also targeted: for example, if for one alternative x_p we obtain $\gamma_p^+ = 0$ (resp., $\gamma_p^- = 0$), then this alternative should be classified in the best (resp., worst) category; in other words, its original assignment is underestimated (resp., overestimated). The same criterion is applied to the alternatives that are originally assigned to the best or worst category.

6 METHODS FOR ASSESSING THE PERFORMANCE OF THE CLASSIFICATION-BASED MODEL FOR CRITICALITY EVALUATION

6.1 Model Retrieval-Based Approach

The first method of performance assessment is based on the model-retrieval approach proposed in [(6)]. A

fictitious set D_{TR}^{rand} of N_{TR} alternatives $\{x_p^{rand} : p = 1, 2, ..., N_{TR}\}$ is generated by random sampling within the ranges X_i of the criteria, $i = 1, 2, ..., n$. Notice that the size N_{TR} of the fictitious set D_{TR}^{rand} has to be the same as the real training set D_{TR} available, for the comparison to be fair. Also, a MR-Sort classification model $M(\cdot \mid \omega^{rand}, b^{rand})$ is constructed by randomly sampling possible values of the internal parameters, $\{\omega_i : i = 1, 2, ..., n\}$ and $\{b_h : h = 1, 2, ..., k-1\}$. Then, we simulate the behavior of an expert by letting the (random) model $M(\cdot \mid \omega^{rand}, b^{rand})$ assign the (randomly generated) alternatives $\{x_p^{rand} : p = 1, 2, ..., N_{TR}\}$. In other words, we construct a training set D_{TR}^{rand} by assigning the (randomly generated) alternatives using the (randomly generated) MR-Sort model, i.e., $D_{TR}^{rand} = \{(x_p^{rand}, \Gamma_p^M) : p = 1, 2, ..., N_{TR}\}$, where Γ_p^M is the class assigned by model $M(\cdot \mid \omega^{rand}, b^{rand})$ to alternative x_p^{rand}, i.e., $\Gamma_p^M = M(x_p^{rand} \mid \omega^{rand}, b^{rand})$. Subsequently, a new MR-Sort model $M'(\cdot \mid \omega', b')$, compatible with the training set D_{TR}^{rand}, is inferred using the MIP formulation summarized in Section 3. Although models $M(\cdot \mid \omega^{rand}, b^{rand})$ and $M'(\cdot \mid \omega', b')$ may be quite different, they coincide on the way they assign elements of D_{TR}^{rand}, by construction. In order to compare models M and M′, we randomly generate a (typically large) set D_{test}^{rand} of *new* alternatives $D_{test}^{rand} = \{x_p^{test,rand} : p = 1, 2, ..., N_{Test}\}$ and we compute the percentage of "assignment errors", i.e., the proportion of these N_{Test} alternatives that models M and M′ assign to different criticality categories.

In order to account for the randomness in the generation of the training set D_{TR}^{rand} and of the model $M(\cdot \mid \omega^{rand}, b^{rand})$, and to provide robust estimates for the assignment errors ε, the procedure outlined above is repeated for a large number N_{sets} of random training sets $D_{TR}^{rand,j}, j = 1, 2, ..., N_{sets}$; in addition, for each set *j* the procedure is repeated for different random models $M(\cdot \mid \omega^{rand,l}, b^{rand,l}), l = 1, 2, ..., N_{models}$. The sequence of assignment errors thereby generated, $e_{jl}, j = 1, 2, ..., N_{sets}, l = 1, 2, ..., N_{models}$, is, then, averaged to obtain a robust estimate for ε. The procedure is sketched in Figure 5.

Notice that this method does not make any use of the original training set D_{TR} (i.e., of the training set constituted by real-world classification examples). In this view, the model retrieval-based approach can be interpreted as a tool to obtain an absolute evaluation of the expected error that an 'average' MR-Sort classification model $M(\cdot \mid \omega, b)$ with *k* categories, *n* criteria and trained by means of an 'average' dataset of given size N_{TR} makes in the task of classifying a new generic (unknown) alternative.

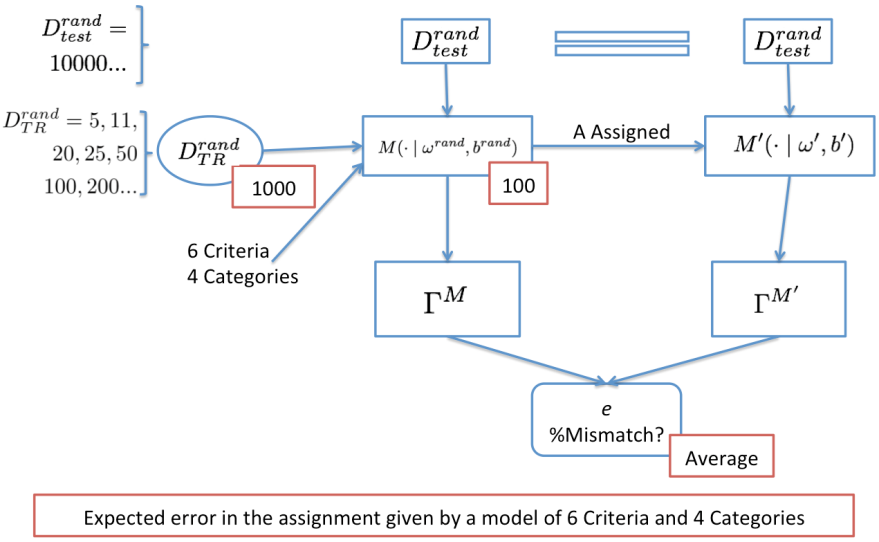

Figure 5. The general structure of the model-retrieval approach

6.2 Cross-Validation Technique [17][18][19]

This technique characterizes the performance of the MR-Sort model in terms of average classification accuracy (resp., error).

The procedure is as follows:

0. Set the iteration number q=1;

1. For a dataset $D = \{(x_p, \Gamma_p^t), p = 1, 2, ..., N_{total}\}$ with pre-assigned alternatives, select a learning set $D_{TR}^q = \{(x_s, \Gamma_s^t), s = 1, 2, ..., N_{TR}\}$ (with $\frac{1}{2}N_{Total} < N_{TR} < N_{Total}$) by performing random sampling without replacement from the given D. The remaining alternatives are used to form a test set $D_{TS}^q = \{(x_r, \Gamma_r^t), s = 1, 2, ..., N_{TS}\}$, with $N_{TS} = N_{Total} - N_{TR}$.

2. Build a classification model $\{M_q(\cdot|\omega_q, b_q)\}$ on the basis of the training set $D_{TR} = \{(x_s, \Gamma_s^t), s = 1, 2, ..., N_{TR}\}$.

3. Use the classification model $\{M_q(\cdot|\omega_q, b_q)\}$ to provide a class Γ_r^q to the elements of the corresponding test set $D_{TS}^q = \{(x_r, \Gamma_r^t), s = 1, 2, ..., N_{TS}\}$.

4. The classification error ϵ^q on test set D_{TS}^q is computed as the fraction of alternatives of D_{TS}^q that are incorrectly classified.

Steps 1-4 are repeated for $q = 1, 2, ..., B$ times (in this paper, *B*=1000). Finally, the expected classification

error of the algorithm is obtained as the average of the classification errors $\epsilon^q, q = 1, 2, ..., B$, obtained on the *B* test sets D^q_{TS}, $q = 1, 2, ..., B$. The general structure of the algorithm is as shown in Figure 6.

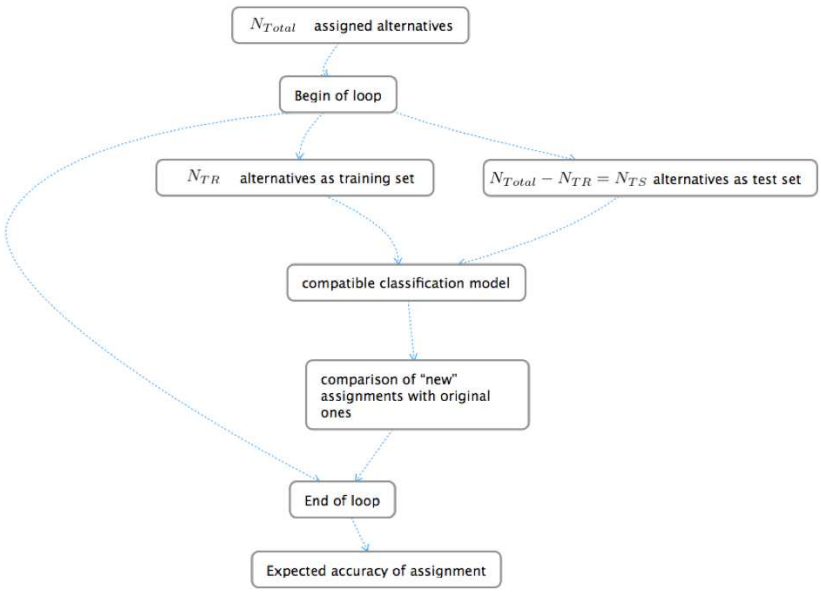

Figure 6. The general structure of the Cross-Validation Technique

6.3 The Bootstrap Method

A way to assess both the accuracy (i.e., the expected fraction of alternatives correctly classified) and the confidence of the classification model (i.e., the probability that the category assigned to a given alternative is the correct one) is by resorting to the bootstrap method [(20)], which is used to create an ensemble of classification models constructed on different datasets bootstrapped from the original one [(21)]. The final class assignment provided by the ensemble is based on the combination of the individual output of classes provided by the ensemble of models [(13)].

The basic idea is to generate different training datasets by random sampling with replacement from the original one [(22)]. The different training sets are used to build different individual classifications. The individual classifiers of the ensemble perform well possibly in different regions of the training space and, thus, they are expected to make errors on alternatives with different characteristics; these errors are

balanced out in the combination, so that the performance of the ensemble is, in general, superior to that of the single classifiers [21][22].

In this paper, the output classes of the single classifiers are combined by *majority voting*: the class chosen by most classifiers is the ensemble final assignment. The bootstrap-based empirical distribution of the assignments given by the different classification models of the ensemble is used to measure the confidence in the classification of a given alternative x, that represents the probability that such alternative is correctly assigned [13][22].

In more details, the main steps of the bootstrap algorithm here developed are as follows (Figure 7):

1. Build an ensemble of *B* (typically of the order of 500-1000) classification models $\{M_q(\cdot \mid (\omega_q, b_q) : q = 1, 2, ..., B\}$ by random sampling with replacement from the original dataset D_{TR} and use each of the bootstrapped models $M_q(\cdot \mid \omega_q, b_q)$ to assign a class Γ_x^q, *q* = 1, 2,..., *B*, to a given alternative x of interest (notice that Γ_x^q takes a value in $A_h, h = 1, 2, ..., k$). By so doing, a bootstrap-based empirical probability distribution $P(A_h \mid x), h = 1, 2, ..., k$ for category A_h of alternative x is produced, which is the basis for assessing the confidence in the assignment of alternative x. In particular, repeat the following steps for *q* = 1, 2, ... , *B*:

    a. Generate a bootstrap dataset $D_{TR,q} = \{(x_p, \Gamma_p^t) : p = 1, 2, ..., N_{TR}\}$, by performing random sampling with replacement from the original dataset $D_{TR,q} = \{(x_p, \Gamma_p^t) : p = 1, 2, ..., N_{TR}\}$ of N_{TR} input/output patterns. The dataset $D_{TR,q}$ is, thus, constituted by the same number N_{TR} of input/output patterns drawn among those in D_{TR}, although due to the sampling with replacement some of the patterns in D_{TR} will appear more than once in $D_{TR,q}$, whereas some will not appear at all.

    b. Build a classification model $\{M_q(\cdot \mid \omega_q, b_q) : q = 1, 2, ..., B\}$, on the basis of the bootstrap dataset $D_{TR,q} = \{(x_p, \Gamma_p^t) : p = 1, 2, ..., N_{TR}\}$.

    c. Use the classification model $M_q(\cdot \mid \omega_q, b_q)$ to provide a class $\Gamma_x^q, q = 1, 2, ..., B$ to a given alternative of interest, i.e., $\Gamma_x^q = M_q(x \mid \omega_q, b_q)$.

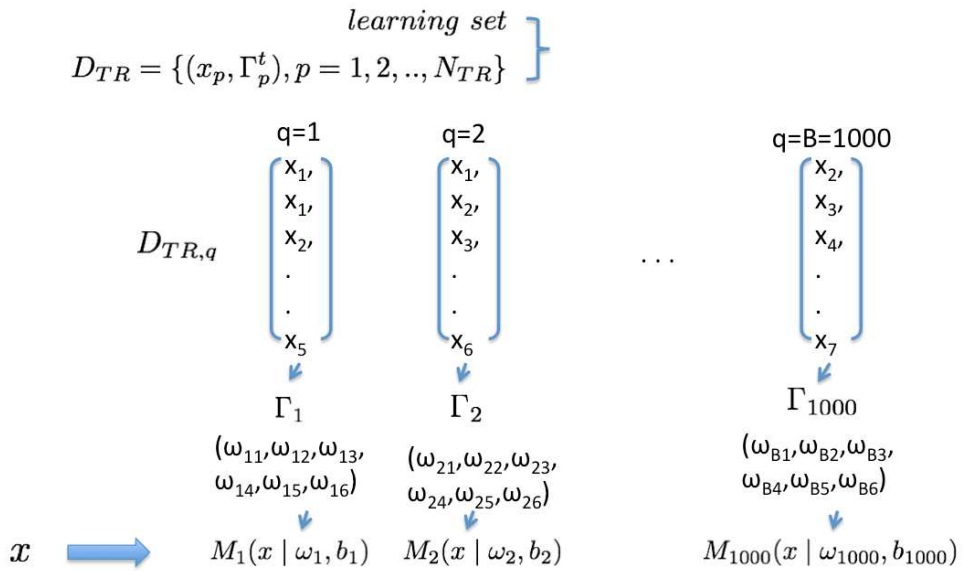

Figure 7. The bootstrap algorithm

2. Combine the output classes $\Gamma^q, q = 1, 2, ..., B$ of the individual classifiers by majority voting: the class chosen by most classifiers is the ensemble assignment Γ_x^{ens}, i.e., $\Gamma_x^{ens} = argmax_{A^h}[card_q\{\Gamma_x^q = A^h\}]$.

3. As an estimation of the confidence in the majority-voting assignment Γ_x^{ens} (step 2, above), consider the bootstrap-based empirical probability distribution $P(A_h \mid x), h = 1, 2, ..., k$, i.e., the probability that category A_h is the correct category given that the (test) alternative is x [(6)]; the estimator of $P(A_h \mid x)$ here employed is: $P(A_h \mid x) = \frac{\sum_{q=1}^{B} I\{\Gamma_q = A_h\}}{B}$, where $I\{\Gamma_q = A_h\} = 1$, if $\Gamma_q = A_h$, and *0* otherwise.

4. Finally, the accuracy of classification is represented by the estimator $P(A_h \mid x)$ (ratio of the number of alternatives correctly assigned by the classification models to the total number of alternatives). The error of the classification model is defined as the complement to 1 to the accuracy.

7 APPLICATION

The methods presented in Sections 4 - 6 are applied on an exemplificative case study concerning the assessment of the overall level of safety-related criticality of Nuclear Power Plants (NPPs) [(9)]. The

characteristics of the plants and their categorization are provided by experts belonging to the R&D Department of Industrial Risk Management of EdF. We identify n = 6 main criteria $i = 1, 2, ..., n = 6$ by means of the approach presented in [9] (see Section 2): x_1 = level of safety, x_2 = level of security and radioprotection, x_3 = possible impact on the environment, x_4 = long-term performance, x_5 = operational performance and x_6 = impact on the communication and reputation of the company. Then, k = 4 criticality categories $A_h, h = 1, ..., k = 4$ are defined as: A_1 = satisfactory, A_2 = acceptable, A_3 = problematic and A_4 = dangerous (Section 2). The entire original dataset is constituted by a group of 35 NPPs x_p with the corresponding a priori-known category Γ_p^t (Table I).

In what follows, first we apply the two approaches for data consistency validation (Section 7.1); then, we use the three techniques of Section 6 to assess the performance of the MR-Sort classification-based model built using the training set D_{TR} (Section 7.2).

Table I. Original training dataset

	Criticality evaluation criteria						
Alternatives (NPPs)	Safety	Secrurity and Radioprotection	Impact on the Environment	Long-term Performance	Operational Performance	Communication and reputation of the Operating Company	Category Assignment (original set)
x1	3	0	3	3	0	2	3
x2	1	0	1	1	0	2	1
x3	1	0	1	2	0	1	2
x4	2	2	3	0	0	1	2
x5	3	1	2	3	0	1	3
x6	1	3	2	2	0	1	2
x7	2	0	3	2	0	3	4
x8	2	2	3	2	0	0	1
x9	1	0	2	0	0	0	1
x10	2	0	3	0	0	2	3
x11	2	0	3	2	0	2	3
x12	1	0	3	1	0	1	3
x13	1	0	2	0	0	1	1
x14	2	0	0	0	0	1	2
x15	1	0	0	0	0	0	1
x16	1	0	0	0	0	1	3
x17	2	0	0	2	0	1	3
x18	1	2	2	0	0	1	2
x19	0	0	0	0	0	1	3
x20	0	3	0	0	1	0	4
x21	1	0	2	1	1	0	4
x22	1	3	0	0	1	0	2
x23	1	0	1	0	1	0	1
x24	1	0	2	0	0	0	4
x25	1	0	0	0	1	0	1
x26	1	0	0	0	0	0	2
x27	1	0	0	0	0	1	2
x28	1	0	0	0	0	1	2
x29	2	2	3	0	0	0	3
x30	2	2	3	2	0	0	2
x31	2	2	2	1	0	0	1
x32	3	0	3	0	0	3	2
x33	1	0	1	0	0	0	3
x34	3	0	0	1	0	3	3
x35	3	0	0	0	0	3	2

7.1 Consistency study results

The application of the MR-sort disaggregation algorithm on the given set of alternatives $D = \{(x_p, \Gamma_p^t) : p = 1, 2, ..., N = 35\}$ (Table I) does not lead to the generation of any classification model (infeasible solution by the MIP algorithm), because there are inconsistencies within the given data. There may exist different types of inconsistencies, as illustrated in Table II by two examples:

Table II. Examples of inconsistent assignments

Case 1:

	Criticality evaluation criteria						
Alternatives (NPPs)	Safety	Secrurity and Radioprotection	Impact on the Environment	Long-term Performance	Operational Performance	Communication and Reputation of the Operating Company	Category Assignment (original set)
x16	1	0	0	0	0	1	3
x27	1	0	0	0	0	1	2

Case 2:

	Criticality evaluation criteria						
Alternatives (NPPs)	Safety	Secrurity and Radioprotection	Impact on the Environment	Long-term Performance	Operational Performance	Communication and Reputation of the Operating Company	Category Assignment (original set)
x13	1	0	2	0	0	1	1
x19	0	0	0	0	0	1	3

In Case 1, two alternatives (x_{16} and x_{27}) with same value for all the six criteria are assigned to different categories (resp., 3 and 2). In Case 2, an alternative (x_{19}) with better characteristics than another (x_{13}) with respect to the six criteria, is assigned to a worse category (3).

Such inconsistencies are solved below via constraints deletion (Section 7.1.1) and constraints relaxation (Section 7.1.2).

7.1.1 Inconsistency resolution via constraints deletion

We first consider finding out the consistent dataset with maximized number of pre-assigned alternatives. We analyze the given dataset by the constraints deletion algorithm. In the given set D of 35 alternatives, 14 are deleted, which leaves a consistent dataset of 21 alternatives. The new consistent set $D_{ad} = \{(x_p, \Gamma_p^t) : p = 1, 2, ..., N_{ad} = 21\}$ is, then, used to generate a compatible classification model $\{M_{ad}(\cdot|(\omega_{ad}, b_{ad})\}$ by the MR-sort disaggregation algorithm. Then, all the alternatives in the original dataset D are assigned a class by model M_{ad}: such assignments agree with the results of the constraints deletion process, i.e., only the deleted alternatives are not correctly assigned (see Table III, where the deleted

alternatives are highlighted).

7.1.2 Inconsistency resolution via constraints relaxation

In the previous Section, we succeeded in obtaining a consistent dataset from a given inconsistent one by deleting the inconsistent alternatives of a "wrong" assignment. However, from the point of view of the experts, it would be ideal to retain as many alternatives as possible in the training set, especially when the size is limited (as is always the case for real systems). This can be done by modifying the pre-defined (wrong) assignments of the inconsistent alternatives.

We examine the same set D by means of the constraints relaxation algorithm presented in Section 5.2. After the application of the algorithm, we obtain the set $D_{ar} = \{(x_p, \Gamma_p^t) : p = 1, 2, ..., N_{ar} = 21\}$, which is identical to the set $D_{ad} = \{(x_p, \Gamma_p^t) : p = 1, 2, ..., N_{ad} = 21\}$ obtained in the previous subsection (for the alternatives in this set, the corresponding generated constraints are consistent). The remaining alternatives form the set $D_r = D - Dar$. However, this algorithm also allows the identification of two more sets: (i) $D_{up} = \{(x_p, \Gamma_p^t) | \gamma_p^+ = 0\}$ (i.e., the set of alternatives whose assignments should be better than the original one, indicated in Table III by a "+" in the shadowed Table cells in column "Constraint relaxation"); (ii) $D_{down} = \{(x_p, \Gamma_p^t) | \gamma_p^- = 0\}$ (i.e., the set of alternatives whose assignments should be worse than the original one, indicated in Table III by a "-" in the shadowed Table cells in the column "Constraints relaxation").

Based on the indications given by the sets D_{up} and D_{down}, we have modified each of the alternatives in D_r by one category in the direction suggested by the relaxation algorithm. Combining the alternatives thereby modified in D_r with the ones in D_{ar} , we obtain a new dataset of 35 alternatives $D_{relax} = \{(x_p, \Gamma_p^{relax}) : p = 1, 2, ..., N_{relax} = 35\}$. A group of $N_{TR} = 25$ data of D_{relax} (marked as *"TR"* in the first column of Table III) is used to build the training set D_{TR} for the model, i.e., $D_{TR} = \{(x_p, \Gamma_p^{relax} : p = 1, 2, ..., N_{TR} = 25)\}$; the remaining 10 alternatives (marked as *"TS"* in the first column of Table III) are used for testing the model generated. In what follows, we consider the classification

model generated using dataset D_{relax} and we assess its performance in terms of accuracy and confidence in the assignments.

Table III. Original inconsistent dataset and the corresponding modifications operated by the constraint deletion and relaxation algorithms

	Criticality evaluation criteria								
Alternatives (NPPs)	Safety	Secrurity and Radioprotection	Impact on the Environment	Long-term Performance	Operational Performance	Communication and Reputation of the Operating Company	Category Assignment (original set)	Constraints deletion	Constraints relaxation
x1 (TR)	3	0	3	3	0	2	3	3	3
x2 (TR)	1	0	1	1	0	2	1	2	-
x3 (TR)	1	0	1	2	0	1	2	2	2
x4 (TR)	2	2	3	0	0	1	2	3	-
x5 (TR)	3	1	2	3	0	1	3	3	3
x6 (TR)	1	3	2	2	0	1	2	2	2
x7 (TR)	2	0	3	2	0	3	4	4	4
x8 (TR)	2	2	3	2	0	0	1	3	-
x9 (TR)	1	0	2	0	0	0	1	1	1
x10 (TR)	2	0	3	0	0	2	3	3	3
x11 (TR)	2	0	3	2	0	2	3	3	3
x12 (TR)	1	0	3	1	0	1	3	3	3
x13 (TR)	1	0	2	0	0	1	1	2	-
x14 (TR)	2	0	0	0	0	1	2	2	2
x15 (TR)	1	0	0	0	0	0	1	1	1
x16 (TR)	1	0	0	0	0	1	3	2	+
x17 (TR)	2	0	0	2	0	1	3	3	3
x18 (TR)	1	2	2	0	0	1	2	2	2
x19 (TR)	0	0	0	0	0	1	3	1	+
x20 (TR)	0	3	0	0	1	0	4	1	+
x21 (TR)	1	0	2	1	1	0	4	1	+
x22 (TR)	1	3	0	0	1	0	2	2	2
x23 (TR)	1	0	1	0	1	0	1	1	1
x24 (TR)	1	0	2	0	0	0	4	1	+
x25 (TR)	1	0	0	0	1	0	1	1	1
x26 (TS)	1	0	0	0	0	0	2	1	+
x27 (TS)	1	0	0	0	0	1	2	2	2
x28 (TS)	1	0	0	0	0	1	2	2	2
x29 (TS)	2	2	3	0	0	0	3	3	3
x30 (TS)	2	2	3	2	0	0	2	3	-
x31 (TS)	2	2	2	1	0	0	1	1	1
x32 (TS)	3	0	3	0	0	3	2	3	-
x33 (TS)	1	0	1	0	0	0	3	1	+
x34 (TS)	3	0	0	1	0	3	3	3	3
x35 (TS)	3	0	0	0	0	3	2	3	-

7.2 Assessment of the classification performance

7.2.1 Application of the Model Retrieval-Based Approach

We generate $N_{sets} = 1000$ different training sets $D_{TR}^{rand,j}, j = 1, 2, ..., N_{sets}$, and for each set *j*, we randomly generate $N_{models} = 100$ models $M(\cdot \mid \omega^{rand,l}, b^{rand,l}), l = 1, 2, ..., N_{models} = 100$. By so doing, the expected accuracy (1-ε) of the corresponding MR-Sort model is obtained as the average of $N_{sets} \cdot N_{models} = 1000 \cdot 100 = 100000$ values $(1 - e_{jl}), j = 1, 2, ..., N_{sets}, l = 1, 2, ..., N_{models}$ (see Section 6.1). The size N_{test} of the random test set D_{TR}^{rand} is $N_{test} = 10000$. Finally, we perform the procedure of Section 6.1 for different sizes N_{TR} of the random

training set D_{TR}^{rand} (even if the chosen size of the training set in our following case study is $N_{TR} = 25$, see Section 7.1.2): in particular, we choose $N_{TR} = 5, 11, 20, 25, 50, 100\ and\ 200$. This analysis serves the purpose of outlining the behavior of the accuracy (1-ε) as a function of the amount of classification examples available.

The results are summarized in Figure 8, where the average percentage assignment error ε is shown as a function of the size N_{TR} of the training set (from 5 to 200). As expected, the assignment error ε tends to decrease when the size of the training set N_{TR} increases: the higher the cardinality of the training set, the higher (resp. lower) the accuracy (resp. the expected error) in the corresponding assignments. Comparing these results with those obtained by Leroy et al [(6)] using MR-Sort models with k = 2 and 3 categories and n = 3-5 criteria, it can be seen that for a given size of the learning set, the error rate (resp. the accuracy) grows (resp. decreases) with the number of model parameters to be determined, equal to $k \cdot n + 1$. It can be seen that for our model with n = 6 criteria and k = 4 categories, in order to guarantee an error rate smaller than 10% we would need training sets consisting of more than N_{TR}= 100 alternatives. Typically, for a learning set of N_{TR} = 25 alternatives (as chosen in Section 7.1.2), the average assignment error ε is around 24%; correspondingly, the accuracy of the MR-Sort classification model trained with the dataset D_{TR} of size $N_{TR} = 25$ available in the present case is around (1-ε) = 76%: in other words, there is a probability of 76% that a new alternative (i.e., a new NPP) is assigned to the correct category of performance.

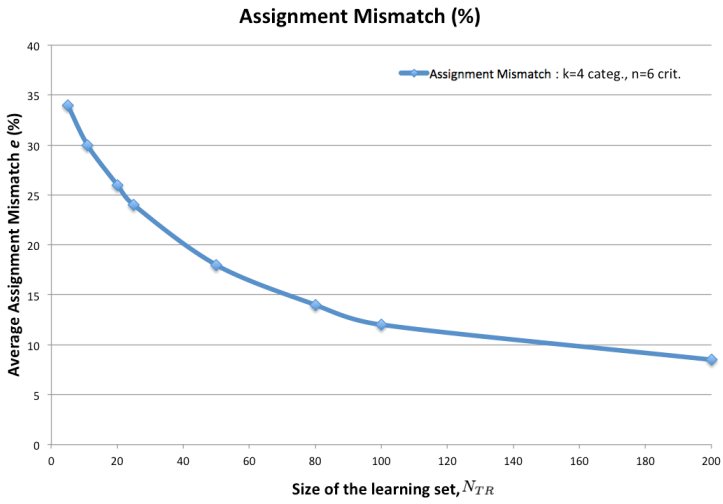

Figure 8. Average Assignment error ε (%) as a function of the size N_{TR} of the learning set according to the model retrieval-based approach of Section 5.1

In order to assess the randomness intrinsic in the procedure used to obtain the accuracy estimate above, we have also calculated the 95% confidence intervals for the average assignment error ε of the models trained with N_{TR} = 11, 20, 25 and 100 alternatives in the training set. The 95% confidence interval for the error associated to the models trained with 11, 20, 25 and 100 alternatives in the training set are [25.4%, 33%], [22.2%, 29.3%], [12.8%, 27.6%] and [10%, 15.5%], respectively. For illustration purposes, Figure 9 shows the distribution of the assignment mismatch built using the $N_{sets} \cdot N_{models} = 100000$ values $e_{jl}, j = 1, 2, ..., N_{sets} = 1000, l = 1, 2, ..., N_{models} = 100$, generated as described in Section 5.1 for the case of 25 alternatives.

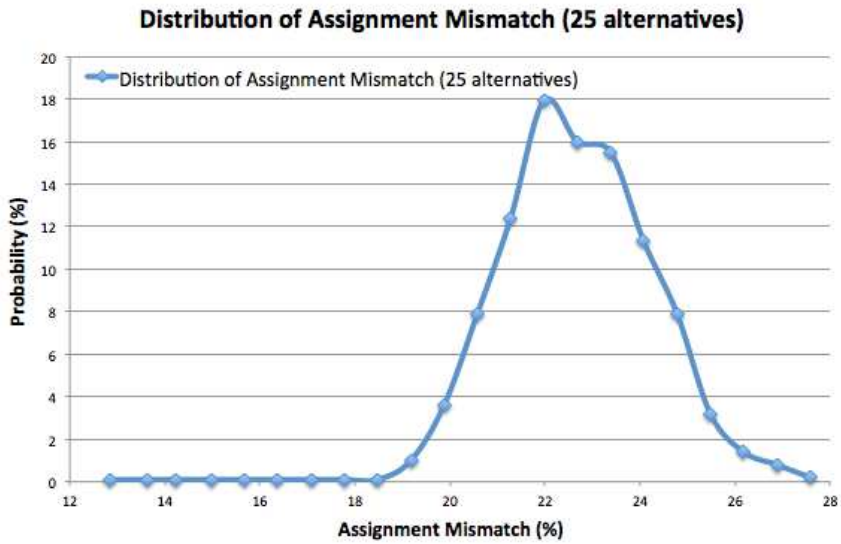

Figure 9. Distribution of the assignment mismatch for a MR-Sort model trained with N_{TR} = 25 alternatives (%)

7.2.2 Application of the Cross-Validation Technique

A loop of *B* (=1000) iterations is performed, as presented in Section 6.2. We take D_{relax} as the training set and generate a training set $D_{TR} = \{(x_p, \Gamma_p^{relax}) : p = 1, 2, ..., N_{TR} = 25\}$ for each loop by performing random sampling without replacement from it. The test set is formed by the corresponding complimentary set of D_{TR}. The average error calculated is around 18%.

7.2.3 Application of the Bootstrap Method

A number *B* (= 1000) of bootstrapped training sets $D_{TR,q}, q = 1, 2, ..., 1000$ of size N_{TR} = 25 is built by random sampling with replacement from D_{TR} (see Section 7.1.2). The sets $D_{TR,q}$ are, then, used to train *B* = 1000 different classification models $\{M_1, M_2, ..., M_{1000}\}$. Then, all the data available (both the training and test elements) are classified by the ensemble.

Notice that all the training patterns are assigned by majority voting to the correct class [(13)]: in other words, the accuracy of the ensemble of models on the training set is 100%. Then, a confidence in the assignment is also provided. In this respect, Table IV reports the distribution of the confidence values associated to the class to which each of the 25 alternatives has been assigned.

Table IV: Number of patterns classified with a given confidence value

Confidence range	(0.5,0.6]	(0.6,0.7]	(0.7,0.8]	(0.8,0.9]	(0.9,1]
Number of patterns	1	3	1	11	9

Thus, a fraction of $20/25 \cong 80\%$ of all the alternatives (i.e., the critical plants) of the training set are correctly assigned with confidence bigger than 0.8.

The ensemble of models can also be used to classify new alternatives, e.g., the alternatives in the test set D_{TS} (see Section 7.1.2). Figure 10 shows the probability distributions of the 10 elements of $P(A_h|x_p), h = 1, 2, ..., k-4, p = 1, 2, ..., N_{TS} = 10$, empirically generated by the ensemble of B = 1000 bootstrapped MR-Sort classification models in the task of classifying the N_{TS} = 10 alternatives of the test set $D_{TS} = \{x_1, x_2, ..., x_{N_{TS}}\}$. The categories highlighted by the rectangles are the correct ones, as obtained by the constraints relaxation algorithm (Section 7.1.2, Table III). It can be seen that six alternatives (x_{26}, x_{27}, x_{28}, x_{29}, x_{30} and x_{33}) over 10 are correctly assigned: in other words, the accuracy of the informed bootstrapped ensemble is around $6/10 \cong 60\%$.

Then, for each specific test pattern x_i, the distribution of the assignments by the B = 1000 classifiers is analyzed to obtain the corresponding confidence. By way of example, it can be seen that alternative x_{28} is assigned to Class A^2 (the correct one) with a confidence of $P(A^2|x_{28}) = 0.931$, whereas alternative x_{26} is assigned to Class A^1 but with a confidence of only $P(A^1|x_{26}) = 0.856$.

More importantly, it can be seen that the 4 alternatives incorrectly classified (x_{31}, x_{32}, x_{34} and x_{35}) are assigned a class close to the correct one; in addition, the "true" class is given the second highest confidence in the distribution. For example, alternative x_{35} is assigned to class A^4 instead of A^3 with 68% confidence; however, the true Class A^3 is still given a confidence of 32%.

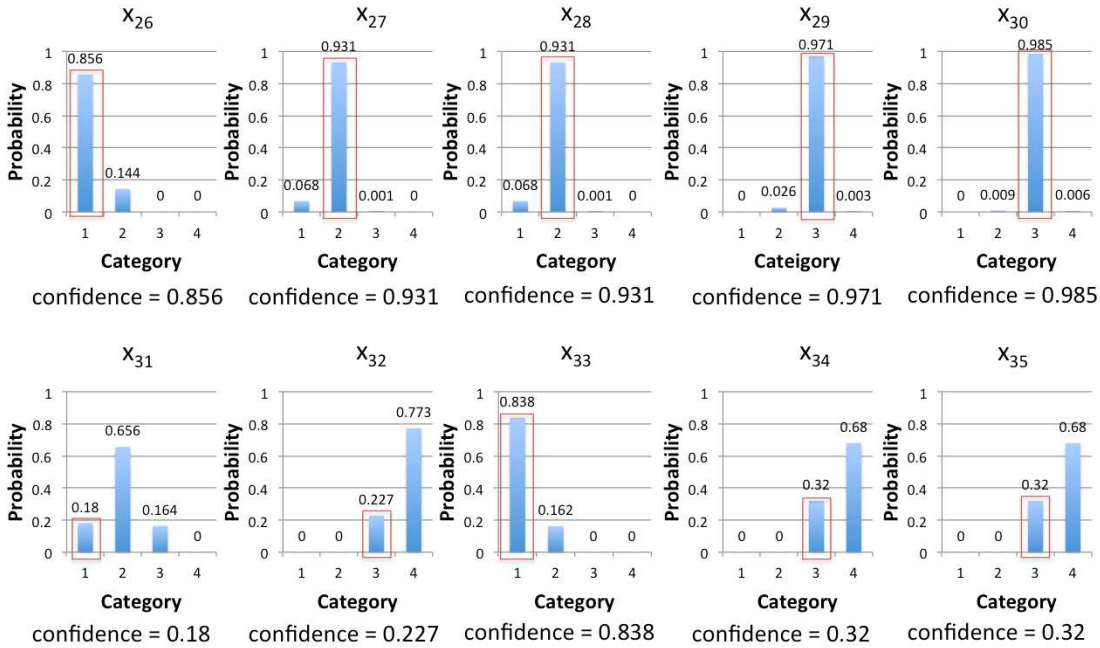

Figure 10. Probability distributions examples of $P(A_h|x_p), h = 1, 2, ..., k-4, p = 1, 2, ..., N_{TS} = 10$ obtained by the ensemble of B = 1000 bootstrapped MR-Sort models in the classification of the alternatives x_p contained in the training set D_{TR}

8 DISCUSSION OF THE RESULTS

The analysis of the inconsistencies of the original dataset has ensured the generation of a coherent training set and, correspondingly, of a compatible classification model for system criticality evaluation: $D_{TR} = \{(x_p, \Gamma_p^{relax} : p = 1, 2, ..., N_{TR} = 25)\}$, generated by constraints relaxation.

Then, three methods have been used to assess the performance of the classification model thereby generated: the three methods provide conceptually and practically different estimates of the performance of the MR-Sort classification model.

The model retrieval-based approach provides a quite general indication of the classification capability of an evaluation model with given characteristics. Actually, in this approach the only constant, fixed parameters are the size N_{TR} of the training set (given by the number of real-world classification examples available), the number of criteria *n* and the number of categories *k* (given by the analysts according to the characteristics of the systems at hand). On this basis, the space of all possible training sets of size N_{TR}

and the space of all possible models with the above mentioned structure (*n* criteria and *k* categories) are randomly explored (again, notice that no use is made of the original training set): the classification performance is obtained as an average over the possible random training sets (of fixed size) and random models (of fixed structure). Thus, the resulting accuracy estimate is a realistic indication of the expected classification performance of an 'average' model (of given structure) trained with an 'average' training set (of given size). In the case study considered, the average assignment error (resp. accuracy) is around 24% (resp. 76%).

The cross-validation method has also been used to quantify the expected classification performance in terms of accuracy. In order to maximally exploit the information contained in the available dataset, B=1000 training sets of size $N_{TR} = 25$ are generated by random sampling without replacement from the original set. Each training set is used to build a model whose classification performance is evaluated on the ten elements correspondingly left out. The average error rate (resp. accuracy) turns out to be 18% (resp. 82%).

On the contrary, the bootstrap method uses the training set available to build an ensemble of models compatible with the dataset itself. In this case, we do not explore the space of all possible training sets as in the model retrieval-based approach, but rather the space of all the classification models compatible with that particular training set constituted by real-world examples. In this view, the bootstrap approach serves the purpose of quantifying the uncertainty intrinsic in the particular (training) dataset available when used to build a classification model of given structure (i.e., with given numbers *n* and *k* of criteria and categories, respectively). In this case study, the accuracy evaluated by the bootstrap method is slightly lower than that estimated by the model retrieval-based approach, with an error (accuracy) rate equals 40% (60%). However, notice that differently from the model retrieval-based approach, the bootstrap method does not provide only the global classification performance of the evaluation model, but also the confidence that for each test pattern a class assigned by the model is the correct one: this is given in terms of the full probability distribution of the performance classes for each alternative to be classified.

9 CONCLUSIONS

In this paper, the issue of assessing the criticality of energy production systems (in the case study considered, nuclear power plants) with respect to different safety-related criteria has been tackled within an empirical framework of classification. An MR-Sort model has been trained by means of a small-sized set of training data representing a priori-known criticality classification examples provided by experts (in our case study, from the Research and Development (R&D) Department of Industrial Risk Management of Electricité de France (EdF)).

Inconsistencies and contradictions in the initial dataset have been resolved by resorting to constraint deletion and relaxation algorithms that have maximized the number of consistent examples in the training set that can be coherently used to build a compatible classification model.

The performance of the MR-sort model has been evaluated with respect to: (i) its classification *accuracy* (resp., error), i.e., the expected fraction of patterns correctly (resp., incorrectly) classified; (ii) the *confidence* associated to the classification assignments (defined as the probability that the class assigned by the model to a given system is the correct one). In particular, the performance of the empirically constructed classification model has been assessed by resorting to three approaches: a model retrieval-based approach, the cross-validation technique and the bootstrap method. To the best of the authors' knowledge, it is the first time that:

- a classification-based framework is applied for the criticality assessment of energy production systems (e.g., Nuclear Power Plants) from the point of view of safety-related criteria;
- the confidence in the assignments provided by the MR-Sort classification model developed is assessed by the bootstrap method in terms of the probability that a given alternative is correctly classified.

From the results obtained in the case study, it can be concluded that although the model retrieval-based approach may be useful for providing an upper bound on the error rate of the classification model (obtained by exploring the space of all possible random models and training sets), for practical

applications the bootstrap method seems to be advisable for the following reasons: (i) it makes use of the training dataset available from the particular case study at hand, thus characterizing the uncertainty intrinsic in it; (ii) for each alternative (i.e., safety-critical system) to be classified, it is able to assess the confidence in its classification by providing the probability that the selected performance class is the correct one. This seems of paramount importance in the decision-making processes performed on the basis of the assessed safety-criticality, since it provides a metric for the 'robustness' of the decision.

In the future, the methodology could be further developed for applications applied to other problems, e.g. the NRC's Risk-Informed Regulatory Oversight Program, in which reactors are assigned to different classes with reference to the amount of regulatory oversight performed.

Acknowledgements:

The authors are thankful to François Beaudouin and Dominique Vasseur of EDF R&D for providing input classification examples and guidance throughout the work. The authors also thank the anonymous reviewers for their valuable comments, which have helped improving the paper significantly.

APPENDIX A. Criticality levels associated to the criteria used for the integrated assessment of a system from the point of view of safety criteria (Section 2)

In what follows, the criticality "scores" associated to each classification criterion introduced in Section 2 are specified.

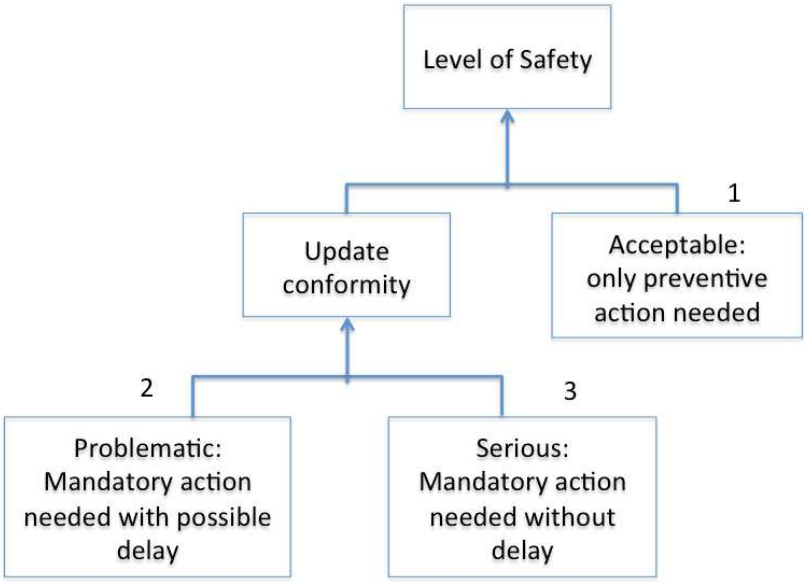

Figure A.1 "Scoring" of criticality for criterion "Level of Safety"

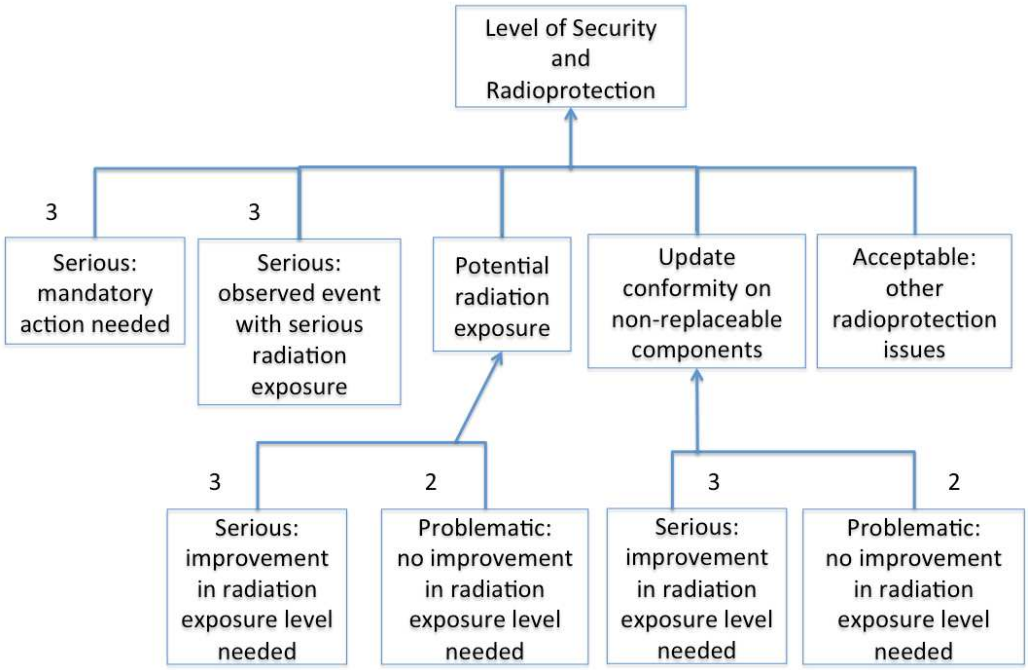

Figure A.2 “Scoring” of criticality for criterion “Level of Security and Radioprotection”

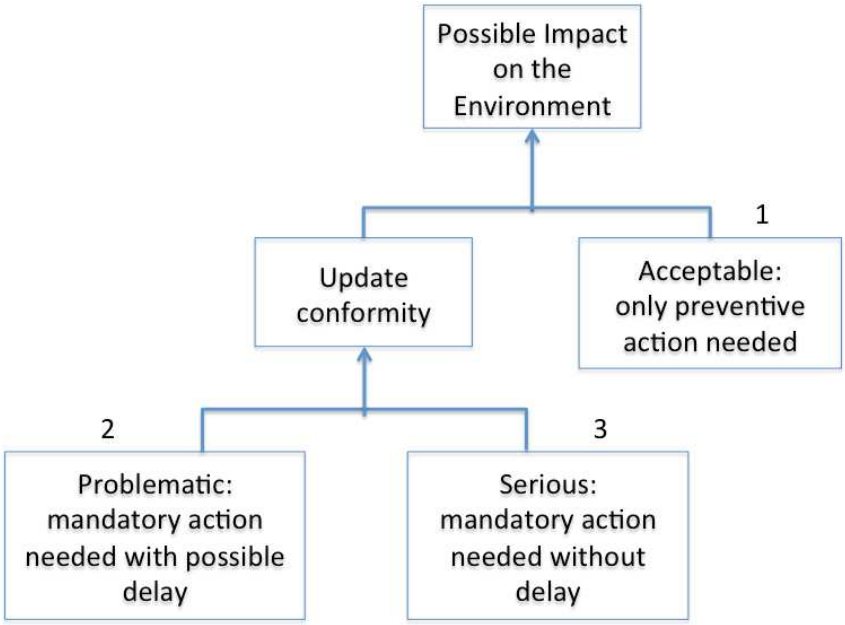

Figure A.3 “Scoring” of criticality for criterion “Level of Possible Impact on the Environment”

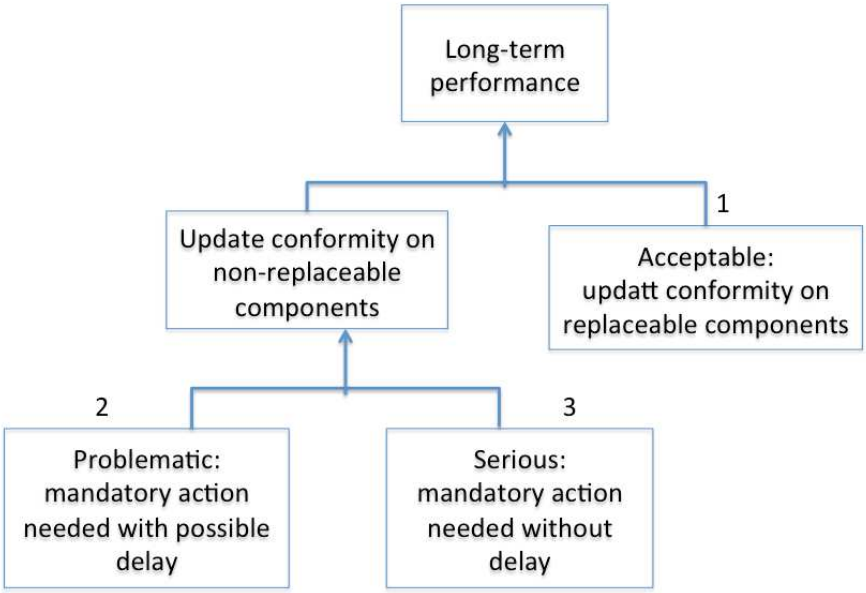

Figure A.4 “Scoring” of criticality for criterion “Level of Long-term performance”

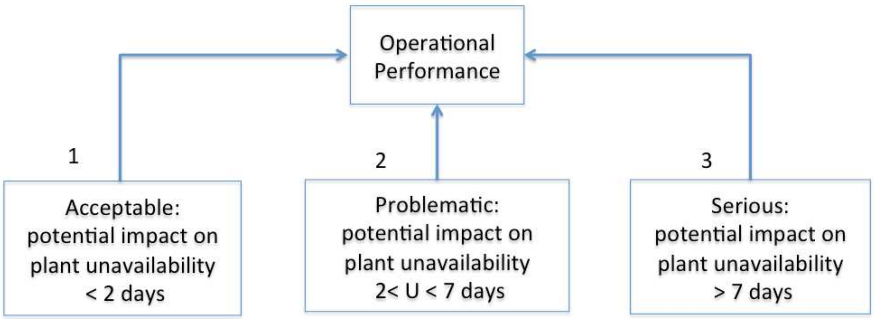

Figure A.5 “Scoring” of criticality for criterion “Level of Operational performance”

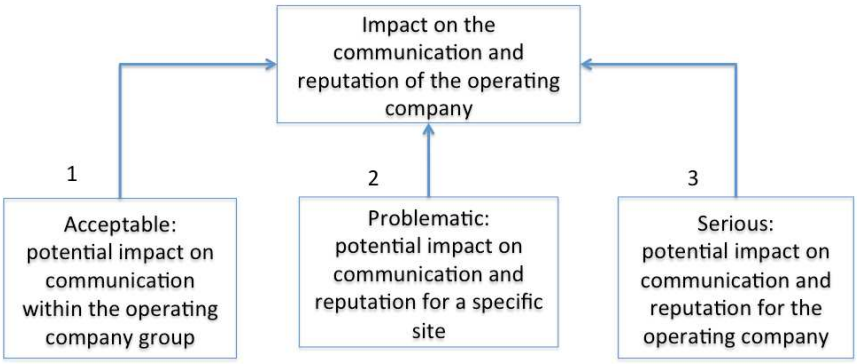

Figure A.6 “Scoring” of criticality for criterion “Level of Impact on the Communication and reputation of the Operational Enterprise”

REFERENCES

1 Wang Q, Poh K.L. A survey of integrated decision analysis in energy and environmental modeling. Energy, 2014; 77, pp. 691-702.

2 Aven T. Foundations of Risk Analysis. Germany, Berlin: Wiley, N.J, 2003.

3 Aven T. Some reflections on uncertainty analysis and management. Reliability Engineering and System Safety, 2010; 95, pp. 195-201.

4 Kröger W, Zio E. Vulnerable Systems. UK, London: Springer, 2001.

5 Huang J.P., Poh K.L. and Ang B.W. Decision analysis in energy and environmental modeling. Energy, 1995; 20, pp. 843-855.

6 Leroy A, Mousseau V, Pirlot M. Learning the parameters of a multiple criteria sorting method, The Second International Conference on Algorithmic Decision Theory, Algorithmic Decision Theory, R.I. Brafman, F. Roberts, and A. Tsoukiàs (Eds.): ADT 2011, LNAI 6992, pp. 219–233, Germany, Berlin: Springer, 2011

7 Wang T-R, Mousseau V, Zio E. A hierarchical decision making framework for vulnerability analysis. pp. 1–8. Proceedings of ESREL2013, Amsterdam, The Netherlands, 2013.

8 Roy B. The outranking approach and the foundations of ELECTRE methods. Theory and Decision 31, 1991, pp. 49-73.

9 Mousseau V., Slowinski R. Inferring an ELECTRE TRI Model from Assignment Examples. Journal of Global Optimization, vol. 12, 1998, pp. 157-174.

10 Aven T, Flage R. Use of decision criteria based on expected values to support decision-making in a production assurance and safety setting. Reliability Engineering and System Safety, 2009; 94, pp. 1491-1498.

11 Milazzo MF, Aven T. An extended risk assessment approach for chemical plants applied to a study

related to pipe ruptures. Reliability Engineering and System Safety, 2012; 99, pp. 183-192.

12 Rocco C, Zio E. Bootstrap-based techniques for computing confidence intervals in Monte Carlo system reliability evaluation. pp. 303–307. Proceedings of the Annual Reliability and Maintainability Symposium, 2005. IEEE

13 Baraldi, P., Razavi-Far, R., Zio, E., 2010. A Method for Estimating the Confidence in the Identification of Nuclear Transients by a Bagged Ensemble of FCM Classifiers. Seventh American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control and Human-Machine Interface Technologies NPIC&HMIT 2010, Las Vegas, Nevada, November 7-11, 2010, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2010).

14 Doumpos M., Zopounidis C., Multricriteria Decision Aid Classification Methods, Kluwer Academic Publishers, Netherlands. 2002, ISBN 1- 4020-0805-8.

15 NWRA, N. W. R. A. Risk assessment methods for water infrastructure systems, 2012. Rhode Island Water Resources Center, University of Rhode Island, Kingston, RI.

16 Mousseau V., Dias C.L., Figueira J. Dealing with inconsistent judgments in multiple criteria sorting models. 4OR: A Quarterly Journal of Operations Research, 2005; 4, pp. 145-158.

17 Baraldi P, Razavi-Fra R, Zio E. Bagged ensemble of fuzzy C means classifiers for Nuclear Transient Identification. Annals of Nuclear Energy, Elsevier Masson, 2011, 38, pp. 1161-1171.

18 Wilson R, Martinez TR. Combining cross-validation and confidence to measure fitness. Proceedings of the International Joint Conference on Neural Networks (IJCNN'99), 1999; pp. 1409-1416. Washington D.C. IEEE

19 Gutierrez-Osuna, R. Pattern analysis for machine olfaction: A review. IEEE SENSORS JOURNAL, 2002, 10.1109/JSEN.2002.800688

20 Efron B, Thibshirani RJ. An introduction to the bootstrap. Monographs on statistics and applied probability 57, 1993. Chapman and Hall, New York.

21 Zio E, A study of the bootstrap method for estimating the accuracy of artificial neural networks in predicting nuclear transient processes. IEEE Transactions on Nuclear Science, 53(3), 2006; pp. 1460-

1470.

22 Cadini F, Zio E, Kopustinskas V, Urbonas R. An empirical model based bootstrapped neural networks for computing the maximum fuel cladding temperature in a RBMK-1500 nuclear reactor accident. Nuclear Engineering and Design, 238, 2008; pp. 2165-2172.