JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 7, December 2010

# A Tool for Prediction of Satellite Future States

Primavera Botelho de Souza* , Mauricio Gonçalves Vieira Ferreira† , and José Demisio Simões da Silva‡
*National Institute for Space Research, São José dos Campos, SP 12227-010*

DOI: 10.2514/1.52045

**The prospect of multiple launches by National Institute for Space Research's satellite program has motivated the development of an application using techniques based on artificial intelligence techniques for automatic generation of flight operation plans to control satellite activities. However, making a critical analysis of these plans before real-world implementation is not possible. We propose a different approach as part of a strategy to validate these plans. This will use a decision support tool based on machine learning concepts to generate prognosis of satellite states for assisting experts in evaluating the performance of the plan. To build the tool, a comparative study of performance between classic data mining classifiers is accomplished to determine the classification model that provides greater accuracy to predict satellite future states.**

## Nomenclature

| | |
|---|---|
| $P(Y)$ | prior probability for $Y$ |
| $P(Y\|X)$ | posterior probability for $Y$ |
| $X$ | attribute set |
| $Y$ | class variable |

## I. Introduction

THERE is general interest in automating satellite control operations related to the task of controlling multiple satellites in National Institute for Space Research's (INPE) space program. In addition, it is generally accepted that the automation of satellite control activities represents a way of reducing in-orbit satellite maintenance costs. At INPE, autonomous systems to control satellite operations employing artificial intelligence (AI) are being developed to automate ground segment operations.

However, this increased autonomy in satellite control operations can lead to distrust of the automatic control system behavior as compared with that of the well known and routine manual control system. In such cases, these systems still require an improvement in reliability to become operational.

In order to achieve this breakthrough in reliability, predictability, and safety, an AI-based strategy for automatic validation of a flight operation plan generated by a planner is presented. This is an architecture composed of software components, resulting from the combination of verification and validation techniques. As a relevant part of this strategy, a decision support tool is proposed in this paper, to assist experts in evaluating the actions of the plan, aiming at guaranteeing the integrity of the satellite. This tool consists of software using AI techniques aimed at predicting the behavior of critical platform satellite subsystems, such as the power supply subsystem (PSS), directly affected by the actions contained in each flight operation plan.

---

Received 20 August 2010; accepted for publication 3 October 2010. 
* Doctorate student, Applied Computing Postgraduate Program (CAP), Av. Dos Astronautas 1758 Jd. Granja, prima@laser.inpe.br.
† Doctor researcher, Satellite Control and Tracking Center (CRC), Av. dos Astronautas 1758 Jd. Granja, mauricio@ccs.inpe.br.
‡ Doctor researcher, Applied Mathematics and Computing Associated Laboratory (LAC), Av. dos Astronautas 1758 Jd. Granja, demisio@lac.inpe.br.

This paper presents in the following section some concepts related to the automation of the control activities of the satellite in orbit. Section 3 describes the strategy for validation of a flight operation plan, an overview of the software architecture and the tool proposed for validation. Section 4 discusses some data mining techniques of classification for data prediction to design the tool. Section 5 presents a comparative study of performance between classifiers algorithms to determine the classification model that provides greater accuracy to predict satellite future states. Conclusions are presented in Section 6.

## II. Satellite Flight Operation Plan

The flight operation plan includes the planning of control operations of space missions and ground segment activities for the planning, execution, and control of the satellite in orbit. Each flight operation plan aims to maintain the satellite in orbit, working to achieve the goals of the mission, containing all the necessary information to control the satellite in orbit, such as: procedures for flight control, procedures for recovery of contingencies, rules, plans, and schedules. All activities included in a flight operation plan have as their starting point the passage of the satellite over the Earth station. The amount of time that a satellite is visible to a given Earth station determines the set of flight operations that should be performed during each pass. Among the activities to control for this period is the sending of commands from the ground (telecommand), and the reception of telemetry which indicates the general state of the satellite.

To meet the growing demand for satellites in orbit and reduce costs significantly, recent studies in AI-based planning have been aimed at the development of tools that automate the tasks of controlling ground operations in INPE. The system, called intelligent planning of flight operation plans (*PlanIPOV*) [1], uses temporal planning AI techniques (temporal planner) applied to the automatic generation of flight operation plans to support the activities of controlling satellites in orbit.

At the same time, the use of automatically generated flight operation plan leads to many doubts. These are partly related to the new technologies involved, but the greatest resistance is related to reliability in the execution of these actions, the predictability and safety of satellites. This increase in autonomy can lead to suspicion about the behavior, often well known and routine. The set of actions contained in a plan acts directly on data critical to maintain of the satellite integrity. Furthermore, depending on the demand for satellites in orbit, a careful validation of these plans can become unviable. In other words, this increased autonomy in satellite control operations still require an improvement in reliability to become operational.

## III. Strategy for Validation of Flight Operation Plan

For this advance in reliability, a strategy for validation of flight operation plans is being proposed. The strategy of validation consists of an architecture composed of several software components for validation of an operation plan generated automatically, to be executed in simulation before actual execution (Fig. 1). Designed with the aim of evaluating the impact of the plan from the simulated state of the satellite, the strategy is designed on the basis of appropriate assurance techniques for space systems [2].

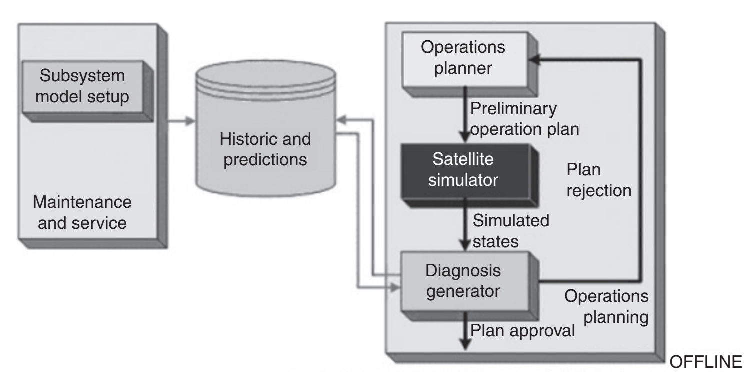


**Fig. 1 Validation of flight operations plan: architecture and situation.**

As the relevant part of this strategy, a validation tool called the diagnosis generator has being developed to provide prediction about future satellite states from the parameters and critical telemetries, indicating how the general satellite state should evolve, suggesting the adoption, or rejection of the plan.

Through an execution offline of the generated plan by the operations planner [1], each action of the plan is executed and a simulation of the satellite behavior is performed by a satellite simulator [3]. The simulator is based on a virtual satellite, with simplified models, which is also part of the strategy for validation of the generated plan [3].

The simulator returns to the diagnosis generator, parameters, and telemetries, containing the simulated state of the satellite, resulting from the execution of the plan's actions by simulator. As a study case, a simplified model of telemetries, parameters, and operational limits of the PSS of a virtual satellite XSAT is being used. The power supply is a critical subsystem for the satellite integrity [3]. Tables 1–5 present a description of these XSAT parameters and telemetries provided by satellite simulator and used as input data for diagnosis generator.

Upon receiving the data from the XSAT virtual satellite PSS model due to an implementation of the plan's actions, the diagnosis generator tool provides prediction from these parameters and telemetries, generating prognosis of the satellite states indicating how the general state of the satellite will evolve, indicating the impact of the plan in the security level of the satellite operation status.

## IV. Techniques for Data Prediction

Computational prediction models are based on probabilistic reasoning over time, interpreting the present and understanding the past and future forecast [4]. The prediction is one of the basic inference tasks in time models, in

**Table 1 Virtual Satellite (XSAT) mission operations summary**

| Payload | Description | Payload data | Data receiving station | Operation criteria | Power consumption |
|---|---|---|---|---|---|
| PL1 | Optical camera | Satellite imagery for land surface monitor | Image receiving station | Over station, at sunlight or at night if calibration requested | PPL1 ON = 800 W OFF = 100 W |
| PL2 | Data collection subsystem | Environmental data acquired by data collection platforms | Data collection station | Over station or continuous, at sunlight and eclipse | PPL2 ON = 15 W OFF = 5 W |

**Table 2 XSAT PSS parameters**

| Identifier | Description | Identifier | Description |
|---|---|---|---|
| SAG | Solar array generator | PAV | Power available to the satellite |
| PSAG | SAG power | IBAT | BAT charging current |
| BAT | Battery | VBAT | BAT voltage |
| QBAT | BAT charge | DOD | BAT depth-of-discharge |

**Table 3 XSAT power values**

| Onboard status | | Description | Generated power (W) | Consumed power (W) |
|---|---|---|---|---|
| SAG | SUN | Sunlight—sun illuminated phase | 1600 | 0 |
| | ECL | Eclipse—eclipse phase | 0 | 0 |
| PL1 | ON | PL1 operating | 0 | 800 |
| | OFF | PL1 standby | 0 | 100 |
| PL2 | ON | PL2 operating | 0 | 15 |
| | OFF | PL2 standby | 0 | 5 |
| SM | – | Service module | 0 | 780 |

**Table 4 XSAT power in each operation mode**

| Operation mode (defined in the plan) | Onboard status | | | Power (W) | | |
|---|---|---|---|---|---|---|
| | SAG | PL1 | PL2 | Consumed | Generated | Available |
| A | SUN | ON | ON | 1595 | 1600 | 5 |
| B | SUN | ON | OFF | 805 | 1600 | 795 |
| C | SUN | OFF | ON | 115 | 1600 | 1485 |
| D | SUN | OFF | OFF | 885 | 1600 | 715 |
| E | ECL | ON | ON | 1595 | 0 | −1595 |
| F | ECL | ON | OFF | 1585 | 0 | −1585 |
| G | ECL | OFF | ON | 895 | 0 | −895 |
| H | ECL | OFF | OFF | 885 | 0 | −885 |

**Table 5 XSAT battery DOD control criteria**

| DOD (%) | DOD status | Operation status |
|---|---|---|
| <15 | Low | Safe |
| 15–20 | High | Unsafe |
| >20 | Extreme | Forbidden |

which the posterior distribution on the future state is calculated, given all the evidence to date. Predictive models have been widely used for building tools to support decision making.

Data mining is a method, in which the ultimate goal is prediction, and represents a process developed to examine routinely large amounts of data collected in search of consistent patterns and systematic relationships between variables. Techniques for finding and describing structural patterns in data have developed within a field known as machine learning, where different styles of learning appear, depending on the data mining application. Those applications where the predictive model requires a judgment needed to inform future decisions, a classification learning scheme takes a set of classified examples (training data) from which it is expected to learn a way of classifying unseen examples (test data) [5].

A classification technique (or classifier) is a systematic approach to building classification models from an input data set. Each technique employs a learning algorithm to identify a model that best fits the relationship between the attribute set (*input*) and class label (*output*) of the input data. The model generated by a learning algorithm should both fit the input data well and correctly predict the class labels of records it has never seen before. Therefore, a key objective of the learning algorithm is to build models with good generalization capability; i.e., models that accurately predict the class labels of previously unknown records [5]. We approach the classical techniques of classification, including decision tree classifiers, Bayesian classifiers, and neural networks.

Following the general approach to solving a classification problem, it was used as a case study, a training data; i.e., a data set with 156 records (instances) of classified examples (Table 6). These input data consist on attribute set of telemetries, parameters, and operational limits of a simplified model of a PSS [3], based on a virtual satellite (see Sec. III), as a result of the action set of a flight operation plan. Each data record is associated with classification of satellite security levels SAFE2 and SAFE3 (STATE class label). For this input data was applied a classifier algorithm, representing each classical classification learning scheme, which each algorithm produces a classification model.

The method used to handle the input data for all classifiers algorithm was one of the methods to random subsampling called cross-validation. We used the 10-fold cross-validation, which the data were segmented into 10 equal-sized partitions. During each run, one of the partitions is chosen for testing, whereas the rest of them are used for training. This procedure is repeated 10 times so that each partition is used for test exactly once.

As mentioned in Sec. III, the diagnosis generator tool should be able to generate data prediction for this satellite subsystem considered critical, based on the classification model that provides greater accuracy to predict satellite future states. So, aiming to provide adequate reasons, the following sections present the main features of these classifiers and associated algorithms used to build the classification models for the diagnosis generator tool.

**Table 6 Input data from XSAT operation**

| DATE/TIME | | SAG | PSAG | PPL1 | PPL2 | PAV | BAT | VBAT | QBAT | CBAT | IBAT | DOD | STATE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 19/4/2010 | 12:30:10 | SUN | 1600 | 100 | 5 | 715 | FULL | 50 | 60 | 1.2 | 0 | 0 | SAFE3 |
| 19/4/2010 | 12:30:40 | SUN | 1600 | 100 | 5 | 715 | FULL | 50 | 60 | 1.2 | 0 | 0 | SAFE3 |
| 19/4/2010 | 12:31:10 | SUN | 1600 | 100 | 5 | 715 | FULL | 50 | 60 | 1.2 | 0 | 0 | SAFE3 |
| 19/4/2010 | 12:31:40 | SUN | 1600 | 100 | 5 | 715 | FULL | 50 | 60 | 1.2 | 0 | 0 | SAFE3 |
| 19/4/2010 | 12:32:10 | SUN | 1600 | 100 | 5 | 715 | FULL | 50 | 60 | 1.2 | 0 | 0 | SAFE3 |
| 19/4/2010 | 12:32:40 | ECL | 0 | 100 | 5 | −885 | DIS | 50 | 59.84 | 1.2 | −19.47 | 0 | SAFE3 |
| 19/4/2010 | 12:33:10 | ECL | 0 | 100 | 5 | −885 | DIS | 49.86 | 59.68 | 1.2 | −19.52 | 0.01 | SAFE3 |
| 19/4/2010 | 12:33:40 | ECL | 0 | 100 | 5 | −885 | DIS | 49.73 | 59.51 | 1.2 | −19.58 | 0.01 | SAFE3 |
| 19/4/2010 | 12:34:10 | ECL | 0 | 100 | 5 | −885 | DIS | 49.59 | 59.35 | 1.2 | −19.63 | 0.01 | SAFE3 |
| 19/4/2010 | 12:34:40 | ECL | 0 | 100 | 5 | −885 | DIS | 49.46 | 59.18 | 1.2 | −19.68 | 0.01 | SAFE3 |
| 19/4/2010 | 12:35:10 | ECL | 0 | 100 | 5 | −885 | DIS | 49.32 | 59.02 | 1.2 | −19.74 | 0.02 | SAFE3 |
| 19/4/2010 | 12:35:40 | SUN | 1600 | 100 | 5 | 715 | CHG | 49.18 | 59.14 | 1.2 | 14.54 | 0.01 | SAFE3 |
| 19/4/2010 | 12:36:10 | SUN | 1600 | 100 | 5 | 715 | CHG | 49.28 | 59.26 | 1.2 | 14.51 | 0.01 | SAFE3 |
| 19/4/2010 | 12:36:40 | SUN | 1600 | 100 | 5 | 715 | CHG | 49.38 | 59.38 | 1.2 | 14.48 | 0.01 | SAFE3 |
| 19/4/2010 | 12:37:10 | SUN | 1600 | 100 | 5 | 715 | CHG | 49.49 | 59.5 | 1.2 | 14.45 | 0.01 | SAFE3 |
| 19/4/2010 | 12:37:40 | SUN | 1600 | 100 | 5 | 715 | CHG | 49.59 | 59.62 | 1.2 | 14.42 | 0.01 | SAFE3 |
| 19/4/2010 | 12:38:10 | SUN | 1600 | 100 | 5 | 715 | CHG | 49.69 | 59.74 | 1.2 | 14.39 | 0 | SAFE3 |
| 19/4/2010 | 12:50:10 | SUN | 1600 | 100 | 5 | 715 | CHG | 49.25 | 59.23 | 1.2 | 14.52 | 0.01 | SAFE3 |
| 19/4/2010 | 12:50:40 | ECL | 0 | 800 | 5 | −1585 | DIS | 49.35 | 58.93 | 1.2 | −35.33 | 0.02 | SAFE3 |
| 19/4/2010 | 12:51:10 | ECL | 0 | 100 | 5 | −885 | DIS | 49.11 | 58.77 | 1.2 | −19.82 | 0.02 | SAFE3 |
| 19/4/2010 | 12:51:40 | ECL | 0 | 100 | 5 | −885 | DIS | 48.97 | 58.6 | 1.2 | −19.88 | 0.02 | SAFE3 |
| 19/4/2010 | 12:52:10 | ECL | 0 | 100 | 5 | −885 | DIS | 48.83 | 58.43 | 1.2 | −19.93 | 0.03 | SAFE3 |
| 19/4/2010 | 12:52:40 | ECL | 0 | 100 | 5 | −885 | DIS | 48.7 | 58.27 | 1.2 | −19.99 | 0.03 | SAFE3 |
| 19/4/2010 | 12:53:10 | ECL | 0 | 100 | 5 | −885 | DIS | 48.56 | 58.1 | 1.2 | −20.05 | 0.03 | SAFE3 |
| 19/4/2010 | 12:53:40 | SUN | 1600 | 100 | 5 | 715 | CHG | 48.42 | 58.22 | 1.2 | 14.77 | 0.03 | SAFE3 |
| 19/4/2010 | 12:54:10 | SUN | 1600 | 100 | 5 | 715 | CHG | 48.52 | 58.35 | 1.2 | 14.74 | 0.03 | SAFE3 |
| 19/4/2010 | 12:54:40 | SUN | 1600 | 100 | 5 | 715 | CHG | 48.62 | 58.47 | 1.2 | 14.71 | 0.03 | SAFE3 |
| 19/4/2010 | 12:55:10 | SUN | 1600 | 100 | 5 | 715 | CHG | 48.72 | 58.59 | 1.2 | 14.67 | 0.02 | SAFE3 |
| 19/4/2010 | 12:55:40 | SUN | 1600 | 100 | 5 | 715 | CHG | 48.83 | 58.71 | 1.2 | 14.64 | 0.02 | SAFE3 |
| 19/4/2010 | 12:56:10 | SUN | 1600 | 100 | 5 | 715 | CHG | 48.93 | 58.84 | 1.2 | 14.61 | 0.02 | SAFE3 |
| 19/4/2010 | 13:44:40 | ECL | 0 | 800 | 15 | −1595 | DIS | 47.25 | 56.39 | 1.2 | −37.13 | 0.06 | SAFE2 |
| 19/4/2010 | 13:45:10 | ECL | 0 | 100 | 5 | −885 | DIS | 47 | 56.22 | 1.2 | −20.71 | 0.06 | SAFE2 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

## A. Decision Tree Classifiers

A decision tree classifier, which is a simple yet widely used classification technique also known as decision tree induction, derives from the simple divide-and-conquer algorithm for producing decision trees [6]. A decision tree contains a root and others internal nodes, beyond attribute test conditions to separate records that have different characteristics.

A decision tree classification learning algorithm was applied to data set (Table 6) to generate the decision tree model for classification of the satellite state. The algorithm chosen for building the decision tree was a well-known and frequently used over the years the C4.5 and J48 as a class for generating a pruned or unpruned C4.5 decision tree [6].

The output of learning algorithm J48 indicating a pruned decision tree model for the training set used with only two (SAFE2 and SAFE3) leaf nodes classification of states (STATE class label). Furthermore, the resulting tree model indicates that the telemetry related with the battery voltage (VBAT) (See Sec. III) is critical to classify the security level of the satellite operation status. Figure 2 shows the decision tree classification model generated used to prognosis of the satellite state for unknown values in a new data record (record test).

## B. Bayesian Classifiers

Following a different approach, we consider the relationship between the attribute set and the class variable being nondeterministic. In other words, it is when the class label of a test record cannot be predicted with certainty, even

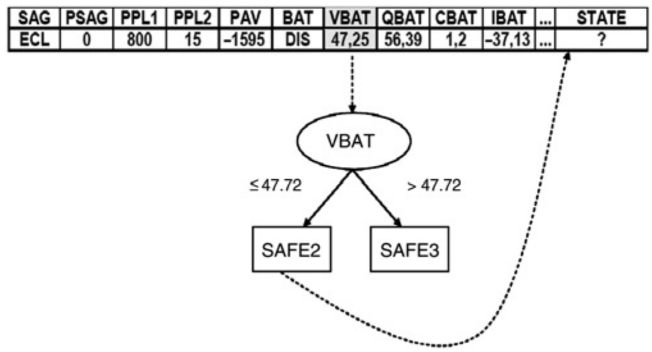


**Fig. 2 Classification model based in decision tree applied a test record.**

though its attribute set is identical to some of the training examples (see Fig. 2). For solving these classification problems, an approach based on the Bayes theorem is used for modeling probabilistic relationships between the attribute set ($X$) and the class variable ($Y$). Consist in a statistical principle for combining prior knowledge of the classes with new evidence gathered from data

$$\frac{P(Y \mid X) = P(X \mid Y P(Y))}{P(X)} \tag{1}$$

Describing how the Bayes theorem was used for classification, let us formalize the classification problem from a statistical perspective. If the class variable ($Y$) has a nondeterministic relationship with the attributes ($X$), then we can treat $X$ and $Y$ as random variables and capture their relationship probabilistically using $P(Y \mid X)$. This conditional probability is also known as the posterior probability for $Y$, as opposed to its prior probability, $P(Y)$. During the training phase, it need to learn the posterior probabilities $P(Y \mid X)$ (Equation (1)) for every combination of $X$ and $Y$ based on information gathered from the training data [7].

The classifier algorithm used to implementation of this model was a naïve Bayes classifier [5], which works using for classification each test record from training data (Table 6), needed to compute the posterior probabilities $P(\text{SAFE2} \mid X)$ and $P(\text{SAFE3} \mid X)$ based on the prior probability obtained for class SAFE3 ($P(\text{SAFE3}) = 67\%$) and the prior probability for class SAFE2 ($P(\text{SAFE2}) = 33\%$). So, the classification is based on the result of the condition: if $P(\text{SAFE3} \mid X) > P(\text{SAFE2} \mid X)$, then the record is classified as SAFE3; otherwise, it is classified as SAFE2.

### C. Vector Quantization

Following one more different approach to build a classification model, we are interested in models of artificial neural networks for classification, because it is a nonparametric and nonlinear technique, which allows the mapping of input data associated with output data. Therefore, the output of the network is the class associated to the sample [8].

For representing a model of artificial neural networks for classification, we choose learning vector quantization (LVQ) networks, which define a family of adaptive algorithms for quantifying vector, originally proposed by Kohonen [9]. LVQ networks define methods for supervised training employing a self-organizing network approach which uses the training vectors to recursively tune placement of competitive hidden units that represent categories of the inputs. Once the network is trained, an input vector is categorized as belonging to the class represented by the nearest hidden unit [8].

The classifier algorithm used to implementation of LVQ networks was the LVQ2_1 classifier algorithm [5]; it consists on iterative algorithm, whose basic principle is to reduce the distance of the input vectors in the same class, and to move away input vector in the wrong class. The classes distribution obtained as output were SAFE3: 16 (80%) and SAFE2: 4 (20%) for the input vectors representing 12 attributes.

In the next section, a performance evaluation of each classification model generated and comparison between three classifiers is accomplished based on performance metrics such as accuracy and error rate values, being the results presented and discussed.

All the classifiers algorithms used are an integral part of the Waikato environment for knowledge analysis (WEKA), a suite of machine learning software was written in Java [6]. WEKA is free software available under the GNU General Public License (GPL), aiming at adding algorithms from different approaches in the subarea of AI, dedicated to the study of learning by machines [6].

## V. Results and Discussion

The performance evaluation of a classification model is based on the counts of test records correctly and incorrectly predicted by the model. These counts are tabulated in a table know as confusion matrix. Table 7 depicts the confusion matrix of classifiers: J48, naïve Bayes, and LVQ2_1.

Each entry $e_{ij}$ in Table 7 denotes the number of records from class SAFE3 predicted to be class SAFE2. For instance, $e_{ji}$ is the number of records from class SAFE2 predicted incorrectly predicted as SAFE3. Thus, based on the entries in the confusion matrix, the total number of correct predictions and total number of incorrect predictions of each model was calculated and presented on Table 8. From these matrix elements is possible also get the performance metrics such as accuracy for each model and the error rate values (Table 8).

The most classification algorithms seek models that attain the highest accuracy, or equivalently, the lowest error rate. Then, evaluating in terms of percentages, the accuracy and error rate values for each classifier, we can say that the classifier naïve Bayes shows the better accuracy value (95%) and minor error rate (5%) followed of the decision tree classifier (91%) and (9%). The worse accuracy and error rate associated was the neural classifier LVQ2_1 (86%) and (13%).

Other key measure for evaluating classifiers is Kappa statistics or Kappa coefficient. A measure of agreement used in nominal scale, that gives us an idea of how much the observations deviate from those expected due to chance, giving us so how legitimate interpretations are. This observer disagreement is indicated by how observers classify individual subjects into the same category on the measurement scale. During in run, each classifier assigned items

**Table 7 Confusion matrix of tree classifiers: J48, naïve Bayes, and LVQ2_1**

| | Class = SAFE3 | Class = SAFE2 | Total |
|---|---|---|---|
| J48 | | | |
| Class = SAFE3 | $e_{ii} = 99$ | $e_{ij} = 6$ | 105 |
| Class = SAFE2 | $e_{ji} = 8$ | $e_{jj} = 43$ | 51 |
| Total | 107 | 49 | 156 |
| Naïve Bayes | | | |
| Class = SAFE3 | $e_{ii} = 99$ | $e_{ij} = 6$ | 105 |
| Class = SAFE2 | $e_{ji} = 2$ | $e_{jj} = 49$ | 51 |
| Total | 101 | 55 | 156 |
| LVQ2_1 | | | |
| Class = SAFE3 | $e_{ii} = 96$ | $e_{ij} = 9$ | 105 |
| Class = SAFE2 | $e_{ji} = 12$ | $e_{jj} = 39$ | 51 |
| Total | 108 | 48 | 156 |

**Table 8 Accuracy and error rate performance metrics for each classifier**

| Classifiers | Accuracy (%) | Error rate (%) |
|---|---|---|
| J48 | 91.02 | 8.97 |
| Naïve Bayes | 94.87 | 5.13 |
| LVQ2_1 | 86.53 | 13.46 |

**Table 9 Kappa coefficient values provided by the three classifiers**

| Classifiers | Kappa | Agreement |
|---|---|---|
| J48 | 0.7940 | Good |
| Naïve Bayes | 0.8858 | Very good |
| LVQ2_1 | 0.6894 | Good |

to one of the two classes SAFE3 and SAFE2, but the number of individuals assigned to each class by classifier are disagree (see Table 7).

The values of Kappa are interpreted as the maximum of 1 when agreement is perfect, 0 when agreement is no better than chance, and negative values when agreement is worse than chance. Other values can be roughly interpreted as [10]:

1) insufficient agreement (<0.20)
2) fair agreement (0.20–0.40)
3) moderate agreement (0.40–0.60)
4) good agreement (0.60–0.80)
5) very good agreement (0.80–1.00)

Kappa measures the percentage of data values in the main diagonal of the confusion matrix (Table 7) and then adjusts these values for the amount of agreement that could be expected due to chance alone. In Table 9, the Kappa coefficient values of each classifier are reported and interpreted.

The Kappa coefficient value obtained form naïve Bayes classifier presented a perfect agreement, whereas the other classifiers present a good agreement. Overall, the classifier algorithm naïve Bayes showed better results, indicating the Bayesian method as the best classification model generated to predict satellite future states.

## VI. Conclusion

This paper presented a comparative study of performance between algorithms classifiers, selected to represent each of the three different methods for classification models generation. The use of methods with different approaches to learning proved to be a significant contribution, in selecting the most appropriate classification model to the set of application data. In addition, the statistic became possible to determine with certainty, which the classification model that provides greater accuracy for predicting satellite future states, once the classification model is the core of the tool designed to assist experts in impact analysis of each plan's action on the satellite behavior to decision support making.

In this study, we simulated data generated from the theoretical model that describe the PSS of a virtual satellite. But, where the data are acquired from a real satellite, which the data may contain missing or anomalies data, the detection and removal of anomalies is often part of the preprocessing of data before the classification model execution. An exception may occur when the anomaly becomes the focus in an application database, requiring the use of specific algorithms to detect anomalies that could be included in future works.

## Acknowledgment

The authors wish to thank CNPq/MCT (an institute of the Brazilian goverment) for supporting this research through grant 142063/2006-1.

## References

[1] Cardoso, L. S., Ferreira, M. G. V., and Orlando, V., "An Intelligent System for Generation of Automatic Flight Operation Plans for the Satellite Control Activities at INPE," *Proceedings of the 9th International Conference on Space Operations*, AIAA, Inc., Reston, Virginia, VA, June 2006.

[2] Blanquart, P., Fleury, S., Hernek, M., Honvault, C., Ingrand, F., Poncet, J. C., Powell, D., Strady-Lécubin, N., and Thevenod-Fosse, P., "Software Safety Supervision On-board Autonomous Spacecraft," *2nd European Congress on Embedded Real Time Software* (*ERTS*-2), Toulouse, France, 2004, 11p.

[3] Tominaga, J., Ferreira, M. G. V., and Silva, J. D. S., "An Implementation of a Satellite Simulator as a Fuzzy Rule-Based Inference Machine," *WORCAP IX*, INPE, São José dos Campos, Brazil, 2009.

[4] Russell, S., and Norvig, P., *Inteligência Artificial*, Tradução da 2nd ed., Editora Campus, Brazil, 2004.
[5] Tan, P.-N., Steinbach, M., and Kumar, V., *Introduction to Data Mining*. Addison-Wesley, Reading, MA, 2005.
[6] Witten, I. H., and Frank, E., *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, 1st ed., The Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann, San Mateo, CA, 1999.
[7] Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H., "The WEKA Data Mining Software: An Update," *SIGKDD Explorations*, Vol. 11, No. 1, 1999, pp. 10–18.
[8] Haykin, S., *Redes Neurais: Princípios e Prática*, 2nd ed., Makron Books, São Paulo, 2001, 900 pp.
[9] Kohonen, T., *Self-Organizing Maps*. 3rd, extended ed., Springer, 2001.
[10] Sheskin, D. J. *Handbook of Parametric and Nonparametric Statistical Procedures*, 3rd ed., Chapman & Hall/CRC, Boca Raton, FL, 2003, ISBN 1584884401.

Reinaldo Perez
*Associate Editor*