X WORCAP
20 e 21 de outubro
Instituto Nacional de Pesquisas Espaciais - INPE
São José dos Campos - SP

A Classification Model to Generate Prognosis of Satellite States

Primavera Botelho de Souza[1], Mauricio Gonçalves Vieira Ferreira[2],

José Demisio Simões da Silva[3]

[1]Applied Computing Postgraduate Program – National Institute for Space Research, São José dos Campos, SP, 12227-010, Brazil

[2]Control and Tracking Center – National Institute for Space Research, São José dos Campos, SP, 12227-010, Brazil

[3]Applied Mathematics and Computing Associated Laboratory – National Institute for Space Research, São José dos Campos, SP, 12227-010, Brazil

`(prima)@laser.inpe.br, mauricio@ccs.inpe.br, demisio@lac.inpe.br`

Abstract. *The prospect of multiple launches by INPE's satellite program, has motivated the development of an application using techniques based on Artificial Intelligence (AI) concepts 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 decision support tool AI-based data mining technique 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.*

Keywords: *Classification Model, Data Prediction, Satellite.*

1. Introduction

There is general interest in automating satellite control operations related to the task of controlling multiple satellites in INPE ´s 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 are being studied 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 to 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 article, to assist experts in evaluating the actions of the plan, aiming at guaranteeing the integrity of the satellite. This tool consists of software using Artificial Intelligence techniques aimed at predicting the behavior of critical platform satellite subsystems, such as the power supply subsystem, directly affected by the actions contained in each flight operation plan.

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.

2. 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*) [Cardoso 2006], 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 others words, this increased autonomy in satellite control operations still require an improvement in reliability to become operational.

3. 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 (Figure 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 [Blanquart et al. 2004].

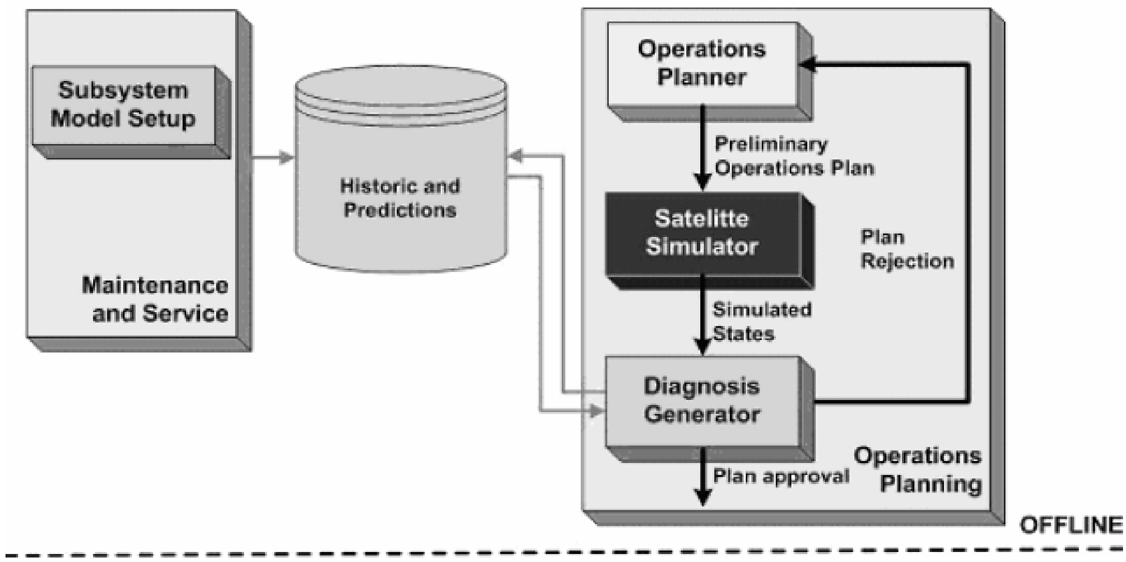

Figure 1. Validation of Flight Operations Plan: architecture and situation

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 off-line of the generated plan by the planner, each action of the plan is executed and a simulation of the behavior of the satellite is performed by a satellite simulator. The simulator is based on a virtual satellite, with simplified models, which is also part of the strategy for validation of the generated plan [Tominaga et al. 2009].

The simulator returns to the Diagnosis Generator, parameters and telemetries (see section 2) containing the simulated state of the satellite, resulting from the execution of the plan's actions. As a study case, a simplified model of telemetries, parameters and operational limits of the power supply subsystem of a virtual satellite XSAT is being used. The power supply is a critical subsystem for the satellite integrity [Tominaga et al. 2009]. The tables 1, 2, 3, 4 and 5 below present a description of these XSAT parameters and telemetries used as input data for Diagnosis Generator:

Table 1. XSAT mission operations summary.

Payload	Description	Payload data	Data receiving station	Operation criteria	Power Consumption
PL1	Optical Camera	Satellite imagery for land surface monitoring	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 Power Supply Subsystem 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 consumed in each operation mode.

Operation Mode (defined in the plan)	Onboard Status				Power (W)		
	SAG	PL1	PL2	SM	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**

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.

4. Techniques for Data Prediction

Computational prediction models are based on probabilistic reasoning over time, interpreting the present and understanding the past and future forecast [Russell and Norvig 2005]. Prediction is one of the basic inference tasks in time models, in 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/or 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) [Frank et al. 2009].

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 [Tan et al. 2005]. 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 dataset 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 Power Supply Subsystem (PSS) [Tominaga et al. 2009], based on a virtual satellite (see section 3), 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 was segmented into 10 equal-sized partitions. During each run, one of the partitions is chosen for testing, while 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 a section 3, 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 the virtual satellite XSAT.

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
...	...	...	...	...	...	...	...	...	...	...	...	...

4.1. 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 [Witten and Frank 1999]. A decision tree includes the root and others internal nodes, contain attribute test conditions to separate records that have different characteristics.

A decision tree classification learning algorithm was applied to dataset (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 [Witten and Frank 1999].

The output of classification learning algorithm J48, indicating a pruned decision tree model for the training set used with only 2 (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 section 3) is critical to classify the security level of the satellite operation status. The 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).

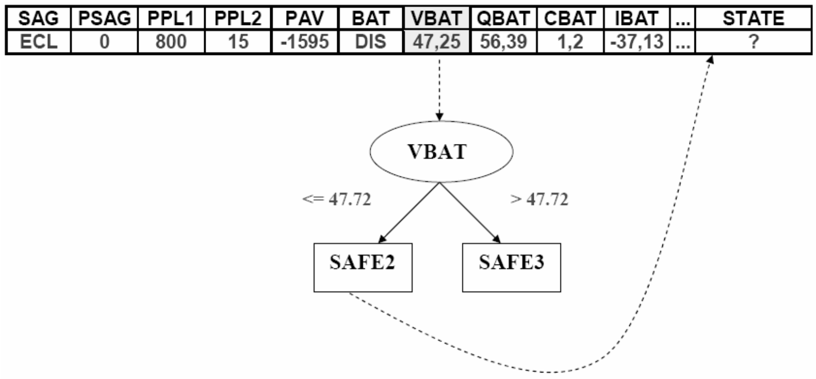

Figure 2. Classification model based in decision tree applied a test record.

4.2. Bayesian Classifiers

Following a different approach, we consider the relationship between the attribute set and the class variable being non-deterministic. In other words, it is when the class label of a test record cannot be predicted with certainty, even though its attribute set is identical to some of the training examples (see Figure 2). For solving these classification problems, an approach based on the Bayes theorem is used for modeling probabilistic relationships between the attribute set and the class variable. 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. Let X denotes the attribute set and Y denote the class variable. If the class variable has a non-deterministic relationship with the attributes, then we can treat X and Y as random variables and capture their relationship probabilistically using P(Y|X). This conditional probability is also known as the posterior probability for Y, as opposed to its prior probability, P(Y) (see equation 1). During the training phase, it need to learn the posterior probabilities P(Y|X) for every combination of X and Y based on information gathered from the training data [Tan et al. 2005].

The classifier algorithm used to implementation of this model was a naive Bayes classifier [Frank et al. 2009], which works using for classification each test record from training data (Table 6), needed to compute the posterior probabilities P(SAFE2|X) and P(SAFE3|X) based on the prior probability obtained for class SAFE3 (P(SAFE3)=67%) and the prior probability for class SAFE2 (P(SAFE2)=33%). So, the classification is based on the result of the condition: if P(SAFE3|X) > P(SAFE2|X), then the record is classified as SAFE3; otherwise, it is classified as SAFE2.

4.3. Artificial Neural Networks

Analogous to human brain structure, an Artificial Neural Networks is composed of an interconnected assembly of nodes and directed links. Consist on set of individual processing elements (formal neurons), grouped under diverse topologies and governed by mathematical procedures clustering vectors, discrete optimization, minimizing errors and others [Haykin 2001].

Following one more different approach to build a classification model, we became interested in models of artificial neural networks for classification, because it is a non-parametric and non-linear 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.

For representing a model of artificial neural networks for classification, we chose Networks LVQ (Learning Vector Quantization), which define a family of adaptive algorithms for quantifying vector, originally proposed by Kohonen. 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 [Haykin 2001].

The classifier algorithm used to implementation of LVQ networks was the LVQ2_1 classifier algorithm [Frank et al. 2009]; 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 written in Java [Frank et al. 2009]. WEKA is free software available under the GNU General Public License, aiming at adding algorithms from different approaches in the sub-area of Artificial Intelligence, dedicated to the study of learning by machines [Witten and Frank 1999].

5. Results and Discussion

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. The Table 7 depicts the confusion matrix of classifiers: J48, naive Bayes and LVQ2_1.

Each entry e_{ij} in the 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).

Table 7. Confusion matrix of tree classifiers: J48, naive Bayes, lvq2_1.

J48	Class = SAFE3	Class = SAFE2	Total
Class = SAFE3	e_{ii} = 99	e_{ij} = 6	105
Class = SAFE2	e_{ji} = 8	e_{jj} = 43	51
Total	107	49	**156**
NAIVE BAYES	**Class = SAFE3**	**Class = SAFE2**	**Total**
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**	**Class = SAFE2**	**Total**
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
NAIVE BAYES	94.87	5.13
LVQ2_1	86.53	13.46

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 naive 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 to one of 2 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 [Sheskin 2003]:

- Poor agreement = Less than 0.20
- Fair agreement = 0.20 to 0.40
- Moderate agreement = 0.40 to 0.60
- Good agreement = 0.60 to 0.80
- Very good agreement = 0.80 to 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.

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

Classifiers	Kappa	Agreement
J48	0.7940	Good
NAIVE BAYES	0.8858	Very good
LVQ2_1	0.6894	Good

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

6. Conclusion

This paper presented a comparative study of performance between classifiers algorithms used in data prediction to determine the classification model that provides greater accuracy to predict satellite future states. The classification model consist on the design of a prediction tool, that is being developed as a relevant part of the validation strategy for a flight operation plan generated automatically to control and track satellites. The tool performs data prediction of a critical platform subsystem, directly affected by the actions contained in each satellite flight plan. In addition, the tool assists experts in impact analysis of each plan's action on the satellite behavior, suggesting the adoption or rejection of the plan.

The most significant contribution of the Diagnosis Generator tool is related to the possibility of evaluating the impact of the plan from simulated satellite states, when integrated with the simulator or from real data to decision support making, providing effective support to experts, and representing an advance in reliability, predictability and safety of the satellite control activities generated automatically, especially considering multiple launchings planned for the near future, when a careful evaluation of these plans, before real execution would be impossible.

7. References

Cardoso, L. S., Ferreira, M. G. V, and Orlando, V. (2006) “An Intelligent System for Generation of Automatic Flight Operation Plans for the Satellite Control Activities at INPE”, *In Proceedings of the 9th International Conference on Space Operations*, Rome, Italy, June.

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

Russell, S. and Norvig, P. (2004). Inteligência Artificial, Tradução da 2a. Edição, Editora Campus, Brazil.

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

Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (1999). The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, Volume 11, Issue 1, University Mark Hall.

Witten, I.H. and Frank E. (1999). Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (The Morgan Kaufmann Series in Data Management Systems), 1st ed. Morgan Kaufmann, October.

Haykin, S. (2001). Redes Neurais: princípios e prática, 2a. ed, Makron Books, 900 p.

Tan, P-N, Steinbach, M. and Kumar, V. (2005). Introduction to Data Mining. Addison-Wesley.

Sheskin, D. J. (2003). Handbook of Parametric and Nonparametric Statistical Procedures, Third Edition**,** Chapman & Hall/CRC, ISBN 1584884401.