

Single and Multiple Failures Diagnostics of Pneumatic Valves using Machine Learning

Humberto Hayashi Sano¹, João Pedro P. Malere², Lilian Berton¹

¹Institute of Science and Technology – Federal University of São Paulo (UNIFESP)
São José dos Campos, SP, Brazil

²Technological Institute of Aeronautics (ITA)
São José dos Campos, SP, Brazil.

humbertosano79@gmail.com, jpmalere@gmail.com, lberton@unifesp.br

Abstract. *Predictive maintenance of aeronautical systems is an important field of study to reduce airlines operational costs and non-scheduled events. Valves are important components of several aircraft systems. This fact motivates the need for health monitoring and prognosis. In this context, this work compares machine learning methods to predict pneumatic valves health conditions. A multilayer perceptron artificial neural network model was able to discriminate the three single failures modes with a mean accuracy of 99.9% and a support vector machine model was able to diagnose single and concurrent failure modes with an average accuracy of 94.3%. The results reveal an accuracy improvement compared to a previous pneumatic valves health assessment study.*

1. Introduction

The air transport industry is a competitive market, especially at the commercial segment where aircraft direct operational costs, which are composed of fuel consumption, maintenance and other operational costs are critical for a profitable operation. One relevant portion of the operational cost is directed related to maintenance. Unscheduled failures and troubleshooting activities have a great impact on the airliners regarding logistic costs and flight schedules. One important aspect to reduce this impact is to minimize the non-scheduled maintenance of aircraft systems. One way to achieve this goal is the development of *Prognostics and Health Monitoring* (PHM) programs.

The aerospace industry is investing in *Integrated Vehicle Health Management* (IVHM) systems to reduce the impacts of the negative effects of unscheduled faults [Jennions 2014]. The main idea of such systems is to get real aircraft systems data to evaluate the health condition of the aircraft components and use this information to schedule maintenance actions [1]. The use of PHM in other industries, such as listed on [PHMSociety 2019], is evidence that not just the aerospace industry is researching this technology, but also manufacturing, power generation and automotive.

The aircraft is composed of many systems which are responsible for its functions. The *Environmental Control System* (ECS) is responsible for providing environmental conditioning to cabin and cockpit. The ECS is composed of some valves which regulate the hot air extracted from the engines to other subsystems as air conditioning, anti-ice, etc. These valves are called *Pressure Regulated Shutoff Valves* (PRSOV). Figure 1 represents these valves in the ECS architecture. Valves internal components are more susceptible to

failures due to their operation at high pressures and temperature environments. PRSOV internal sub-components are shown in Figure 2.

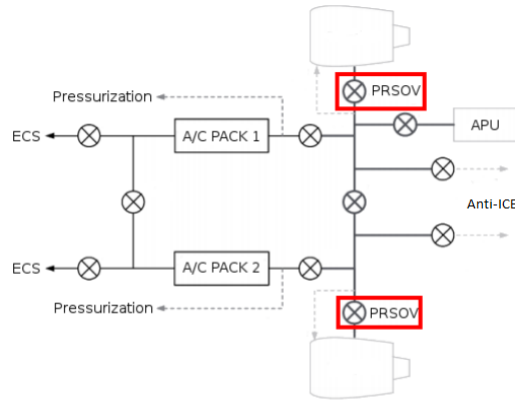


Figure 1. Pressure Regulated Shutoff Valves (PRSOV) in the Environmental Control System (ECS) system.

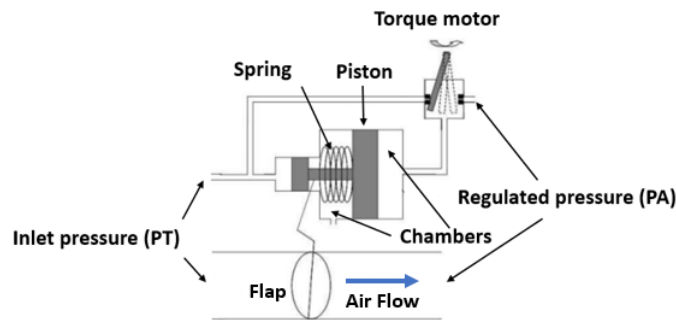


Figure 2. Internal scheme of Pressure Regulated Shutoff Valves (PRSOV) [Turcio et al. 2013].

There is a lack of studies considering a multi-fault classification of the PRSOV using machine learning. In [Castilho et al. 2018] only single-fault has been analyzed. Moreover, the evaluation of other machine learning algorithms to predict the PRSOV's health is demanded as well as an analysis of which features influence the PRSOV prediction.

The objective of this paper is to evaluate the application of machine learning techniques to classify the health of PRSOV. We considered the output responses due to their most relevant parameters variation. As far as we know, this is the first work that analyzes single and multiple failures on PRSOV. The contributions of this work are:

- Presenting the effectiveness of different types of machine learning algorithms to predict the PRSOV health;
- Analyzing PRSOV multi-fault classification;
- Presenting the relevance of considering operational condition variables, as pressures and temperature, to be part of the learning methods attributes;
- Studying the influence of PRSOV input command in the classification results.

The results show that the input command contributes to a more accurate diagnostics as well as the environmental conditions for the multi-fault case. Machine learning

algorithms can also diagnose single faults with 99.9% accuracy and multi-fault with a 94.3% accuracy.

The remaining of this paper is organized as follows: Section 2 presents other works that have studied machine learning to PHM purposes in some industrial applications and with a focus to determine the PRSOV health also. Section 3 presents the main concepts of machine learning used in this paper. Section 4 describes the dataset and its statistical analysis, the inputs type evaluation, the process of machine learning algorithm parameter tuning, the sampling and the criteria to compare the algorithms. Section 5 describes the results obtained employing the concepts presented in Section 4. Finally, Section 6 summarizes the main conclusions of this paper.

2. Related work

Some works already explore PHM of PRSOV valves using machine learning techniques. In [Castilho et al. 2018], the PRSOV has been analyzed using output data from a Simulink model with a specific valve maneuver to obtain the PRSOV times as attributes for the machine learning algorithms. The author used Support Vector Machines (SVM) and Classification and Regression Trees (CART) to estimate the health state of the PRSOV. This work uses these techniques to classify some PRSOV individual failures independently.

[Baptista et al. 2017a] investigated the use of the algorithms k -NN, feed-forward ANNs, RFs and linear SVM to predict the End of Life (EOL) and Remaining Useful Life (RUL) of engineering systems. The authors also showed that data-driven models based on different data sources (sensors data, fault messages and reliability data) can provide better prognostics than traditional prognostics based on historical time-to-failure data [Baptista et al. 2017b]. The following algorithms have been used in this study, k -Nearest Neighbors (k -NN), Artificial Neural Networks (ANN), CART, SVM, Bayesian generalized linear models (Bayes), Gradient Boosting with Regression Trees (Boosted Trees), Linear Regression (LR) and Random Forests (RF).

In [Caesarendra et al. 2010], the authors obtained the failure degradation of bearing machine using data from a Matlab model simulation and experiment bearing run-to-failure. The authors calculated a logistic regression (LR) model and compare this model with the results of a trained relevance vector machine (RVM). Concurrent damage health monitoring was investigated by [Daigle and Goebel 2013], where the author proposed a prognostics methodology based on the state-parameter estimation which describes the damage progression of a component. In [Vianna and Yoneyama 2018], the author studied degradation trends and future wear values estimation considering a multiple model approach of the extended Kalman filter technique.

3. Machine learning algorithms

Machine Learning (ML) has been defined by [Mitchell 1997] as “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T , as measured by P , improves with experience E ”. ML are classified in different categories, where the *Supervised Learning* aims to build a mathematical model from a set of data that contains both the inputs and the desired outputs [Russell and Norvig 2009]. The supervised algorithms employed in this work are: *k-nearest neighbors algorithm* (k -NN), *Naive Bayes* (NB), *Classification*

And *Regression Tree* (CART), *Support-Vector machines* (SVMs), *Multilayer Perceptron* (MLP) and *Multinomial Logistic Regression* (MLR). Following a summary from these methods are presented [Hastie et al. 2009].

K-nearest neighbors algorithm (k -NN) is a non-parametric method that selects the k closest training examples in the feature space and classifies a new example by a plurality vote of its neighbors. The parameter k is a positive integer, usually small. If $k = 1$, then the example is assigned to the class of that single nearest neighbor. A commonly used distance metric for continuous variables is the Euclidean distance.

Naive Bayes classifier is a probabilistic method based on Bayes' theorem with independence assumptions between the features. Given an example to be classified, represented by a vector x , it assigns to this instance probabilities for each of possible class C_n as $P(C_n|x) = p(C_n)p(x|C_n)/p(x)$.

Decision tree algorithm builds a tree by splitting the classification set into the root node of a tree and successor "children". The splitting is based on a set of rules based on the classification feature. This process is repeated on each derived subset recursively. Each interior node corresponds to one of the input variables. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf. There are many specific decision-tree algorithms, here we used *Classification And Regression Tree* (CART).

Support-Vector machines (SVMs) represent the examples as points in space and separate them by categories with a hyperplane that maximizes the distance from each example to the nearest data point on each side. Any hyperplane can be written as the set of points \vec{x} satisfying $\vec{w} \cdot \vec{x} - b = 0$ where \vec{w} is the normal vector to the hyperplane. If the training data is linearly separable, two parallel hyperplanes can separate the two classes of data, so that the distance between them is as large as possible. However, it can efficiently perform a non-linear classification using a kernel trick that implicitly maps the inputs into high-dimensional feature spaces.

Artificial Neural Network (ANN) is an interconnected group of artificial neurons that uses a mathematical or computational model for information processing. A *Multilayer Perceptron* (MLP) has multiple layers (an input and an output layer with one or more hidden layers), non-linear activation and uses backpropagation for training. Backpropagation is an efficient method for calculating the weights updates in the network until it can perform the task for which it is being trained.

A binary logistic model has a dependent variable with two possible values represented by an indicator variable, where the two values are labeled 0 and 1. In the logistic model, the log-odds (the logarithm of the odds) for the value labeled 1 is a linear combination of one or more independent variables. The corresponding probability of the value labeled 1 can vary between 0 and 1, hence the labeling. The binary logistic regression model has extensions to more than two levels of the dependent variable: categorical outputs with more than two values are modeled by *Multinomial Logistic Regression* (MLR).

4. Methodology

This section presents the materials and methods used in the work. Subsection 4.1 presents the generation process of the dataset. Subsection 4.2 presents some statistics for the at-

tributes. Subsection 4.3 presents the attributes selection process. Subsection 4.4 presents the experimental validation and parameter tuning.

4.1. PRSOV model simulation

In this study, a Simulink model of PRSOV was used to obtain samples of the valve with variations at their internal characteristics. The constructive characteristics of the PRSOV which includes the torque motor dynamics and its parts as the chamber volumes, the spring and piston with its mechanical interference are represented in this model. We also represent the airflow which passes through the valve characteristics as pressure and temperature. This model has been validated with experimental data obtained from the laboratory tests.

The model input is the command which controls the valve torque motor and modulates the differential pressure applied in the piston. This piston movement results in a motion of the valve (shown in Figure 2). The model output is the valve flap position. We have obtained PRSOV samples varying the valve parameters as friction coefficient, leakage percentage and spring constant. The three parameters have a direct influence in the piston dynamical movement which is mechanically linked to the valve flap. The interval values of these parameters have been chosen based on the experience of a technical specialist.

We performed simulations with two types of PRSOV commands. The first command (command 1) is the same shape used in [Castilho et al. 2018] and the second (command 2) has a shape with a higher rate transition based on the natural dynamic of the PRSOV. In each simulation, we have collected some output information to classify the valve's health. These parameters are instant times to the PRSOV that achieves 5% and 85% of the fully open position (t_1 and t_2) and 5% and 85% of the full close position (t_3 and t_4). We also collected the total and ambient pressure (P_a and P_t) and temperature (T). The pseudo-code which describes the simulation of the model for each valve sample is described in Algorithm 1.

Algorithm 1 Pseudocode to obtain valves responses

```

1: procedure PRSOV OUT(inputType, leakageDist, springDist, frictionDist)
2:   for valve from 1 to 40000 do
3:     Leak  $\leftarrow$  getLeakageValue(leakageDist)
4:     Friction  $\leftarrow$  getFrictionValue(frictionDist)
5:     Spring  $\leftarrow$  getSpringValue(springDist)
6:     [t1, t2, t3, t4]  $\leftarrow$  Simulate(model, inputType, Leak, Friction, Spring)
7:   return t1, t2, t3, t4

```

We obtained the number of 40000 samples for each PRSOV command type. The analysis of these dataset has been conducted in two parts. Firstly, we considered only isolated failures as possible labels (Healthy, Spring Fault, Leakage Fault, Friction Fault). Based on this information we evaluated the different types of input regarding its shape (command 1 or command 2). The second part consisted of including the possibility of simultaneous failures (Leakage and Spring Simultaneous Faults, Leakage and Friction Simultaneous Faults, Spring and Friction Simultaneous Faults and All Faults occurring simultaneously).

Table 1 presents the main information of the generated dataset. There are the same number of samples (5000) which results in 12.5 % samples for each class.

Table 1. Summary of the dataset attributes.

# Examples	# Attributes (simb./num.)	% Absences	% Majority
40000	7 (0/7)	0	12.5

4.2. Dataset analysis

Analyzing the statistics of the attributes, we can observe differences at the magnitude of values between the time and the operational attributes as pressures and temperatures (see Table 2). Therefore, there is a necessity to normalize the attributes where the machine learning methods demand. We normalize the data to be mean zero and standard deviation one (Equation 1) where \bar{X}_i is the mean and $sd(X_i)$ is the standard deviation of the X_i data. Another aspect observed in the absence of outliers in the dataset due to the median and the mean have almost the same value for the attributes.

$$X_{Ni} = \frac{X_i - \bar{X}_i}{sd(X_i)} \quad (1)$$

Table 2. Dataset attributes summary.

	t1	t2	t3	t4	Pa	Pt	T
Min	1.273	3.327	21.57	23.4	90000	230000	290
1st Quartil	1.338	3.583	21.98	24.21	92496	232504	295
Median	1.411	3.807	22.15	24.54	94990	235037	300
Mean	1.427	3.882	22.12	24.55	94990	235021	300
3rd Quartil	1.504	4.078	22.25	24.94	97480	237546	305
Max	1.734	5.457	22.71	25.75	99999	240000	310

4.3. Input commands and attribute selection

We can notice in the boxplot of Figure 3 that one-time variable is not enough to separate classes. In the case of two variables shown in Figure 4, it is possible in some time combinations to separate the healthy class (green) from the others. In Figure 5, considering three-time variables, two classes can be separated and the other two have a little overlap between them. There is a tendency of increasing the classification performance with the number of attributes. Due to this fact, in our study, we compared two groups of attributes, the first with four attributes (t1, t2, t3 and t4) and the other with the addition of the pressures and temperature. For this comparison, we analyzed the accuracy and the confusion matrix of both groups of attributes applied at the CART algorithm.

4.4. Validation and parameter settings

The dataset has been divided into training and test set with the proportion of 90% and 10% respectively. The test partition represents new data which has not being used in the training process of the machine learning algorithms.

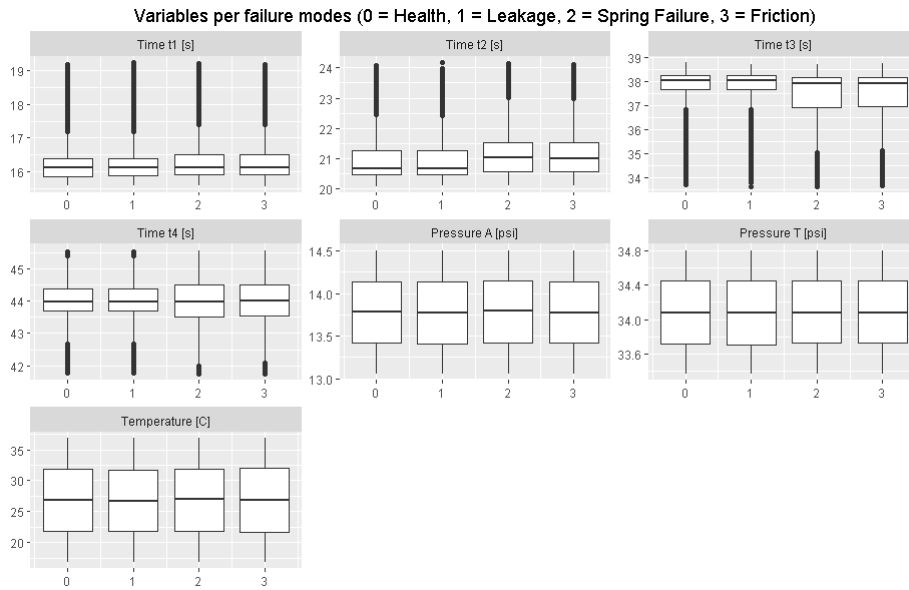


Figure 3. Attributes boxplot.

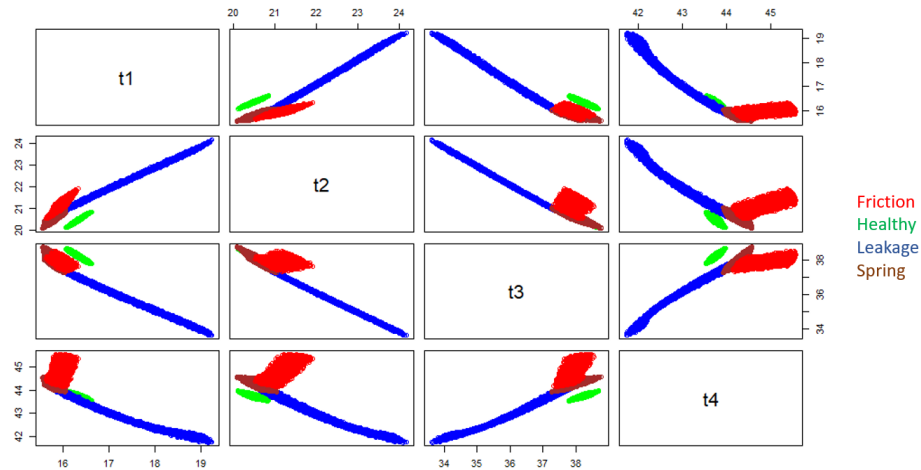


Figure 4. Correlation times 2d.

After the machine learning training process, the best algorithm tuned with the hyper-parameter with higher accuracy has been trained and tested using the training and test partitions respectively. As a result, we obtain the accuracy of this test and the respective confusion matrix to analyze the classification errors.

The training process used a k -fold cross-validation method with $n = 10$. We use the training portion (a subset of 90% of the total dataset) which is divided into n folds. For the k -NN, SVM, and MLP hyper-parameter selection, the algorithm is trained and validated n times. For all methods, it is calculated a mean accuracy over the n runs of the cross-validation. In the k -NN, SVM and MLP cases, the hyper-parameter with highest mean accuracy value is selected. We applied this 10-fold cross-validation for the following algorithms with the configuration below:

- k -NN: Euclidean distance and k varied in the range of 1, 3, 5, 7, 9. The used im-

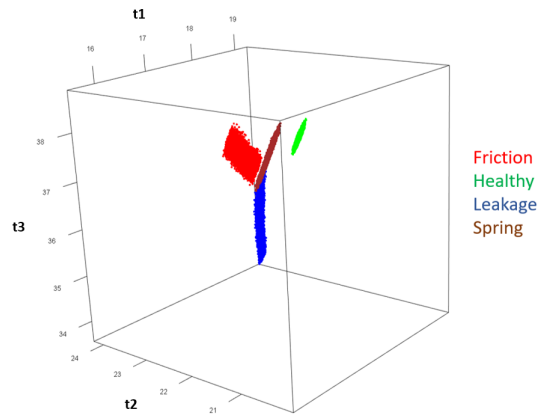


Figure 5. Correlation times 3d.

plementation is the one described in [Ripley and Venable 2019].

- **SVM:** linear kernel, type= C-Classification, $\epsilon = 0.1$, tolerance = 0.001, cost varied in the range of 0.1, 1, 2. The used implementation is the one described in [Dimitriadou et al. 2006].
- **MLP:** configurations (neurons on the two hidden layers): (3, 2), (4, 0), (3, 0), (6, 0), (7, 0), learning rate from 0.5 to 1.2, maximum number of step = $10e5$, number of training repetitions = 10, no start weights, algorithm: resilient back-propagation (rprop-), activation function: logistic, error function = sum of square error of prediction, linear output=false. The threshold parameter is customized during the training process to allow the network training to converge. The used implementation is the one described in [Günther and Fritsch 2010].
- **CART:** Gini split criteria, maximum number of surrogate splits = 5, minimum number of observation to split = 20, minimum number of observations on terminal nodes = 7, complexity parameter = 0.01. The used implementation is the one described in [M. Therneau and Atkinson 1997].
- **NB:** gaussian distribution. The used implementation is the one described in [Dimitriadou et al. 2006].
- **MLR:** $\lambda = 0$. The used implementation is the one described in [Tibshirani et al. 2010].

Since the MLR is estimated through least square error minimization and no regularization, no hyperparameter is selected on the cross-validation. The assumption of a Gaussian distribution is adopted for the NB case and also no hyperparameter is selected for this case. Finally, to compare this work with [Castilho et al. 2018], the default parameters for the CART implementation of [M. Therneau and Atkinson 1997] is used.

5. Results

The results are presented as follows. Subsection 5.1 analyzes the influence of the commands (command 1 and command 2) to produce better outputs of PRSOV classification and the number of variables (times t1, t2, t3 and t4, temperature and pressure) that are

suitable to be considered for the fault diagnostics. Subsection 5.2 presents the classification results for single-faults and Subsection 5.3 presents the classification results for multi-faults.

5.1. Input types evaluation

The first experiment is the comparison between a command used in [Castilho et al. 2018] (command 1) and a higher rate transition command input (command 2). The objective of this experiment is to verify the hypotheses that a command with a rate higher than used in [Castilho et al. 2018] is more effective to produce outputs (times) which reflect better the PRSOV healthy to classification analysis.

We applied the CART algorithm following the configuration described in subsection 4.4 for the attributes obtained from the command 1 and command 2. The tree trained from the attributes obtained from command 2 input provides a more accurate diagnostics information than the command 1 used in [Castilho et al. 2018]. Furthermore, the tree trained from command 2 has a better classification than command 1 as shown in Table 4. The error in predicting Spring instead of Friction is higher in the tree from command 1. For these reasons, the attributes from command 2 input will be considered in further experiments.

Table 3. Accuracy (Acc) and standard deviation (Dev) comparison between command 1 and command 2 inputs.

Input Type	Method	Mean Acc	Dev
command1	CART	97.64	0.36
command2	CART	99.42	0.16

Table 4. Confusion matrix comparison between command 1 and command 2 inputs for CART classification of single-fault (Friction (F), Healthy (H), Leak (L) and Spring (S)).

	Command1				Command2			
	F	H	L	S	F	H	L	S
Pred F	461	0	0	0	495	0	0	0
Pred H	0	500	0	0	0	500	0	0
Pred L	0	0	497	1	0	0	495	2
Pred S	39	0	3	499	5	0	5	498

The second experiment is related to the number of variables that are suitable to be considered for the fault diagnostics. Only the times' variables were considered in [Castilho et al. 2018]. The accuracy of the CART tree trained using the environmental conditions inputs (pressures and temperature) in addition to the time attributes is the same as the tree trained without those parameters as shown in Table 5. The addition of new sensors for diagnostics is costly, so the remaining experiments of this work will only consider the t1, t2, t3 and t4 times without the pressure and temperature environmental condition. The other operation inputs (pressures and temperature) only has been used in case of an accuracy lower than 90% or the non-convergence of an algorithm during training.

Table 5. Variables evaluation.

Variables	Method	Mean Acc	Dev
Times (t1, t2, t3, t4)	CART	99.42	0.16
Times (t1 to t4), temp and press	CART	99.42	0.16

5.2. Single-fault experiments

Based on the results in subsection 5.1, the time attributes (t1, t2, t3 and t4) have been used as inputs to the machine learning algorithms. The results show all the algorithms have an accuracy superior to 90% as shown in Table 6. It seems the classification of the individual fault is simple for the algorithms due to the defined separated classes. This behavior can be observed in the 3d plot (Figure 5). The MLP algorithm had a superior accuracy, for this reason, the confusion matrix of MLP after the test set being applied to this model is shown in Table 7 considering the hidden layer with the best performance (3,2).

Table 6. Accuracy (Acc) and standard deviation (Dev) results for machine learning algorithms in single-fault classification.

Method	Mean Acc	Dev
<i>k</i> -NN (k=1)	99.75	0.14
SVM (cost=2)	99.76	0.086
MLP (3,2)	99.91	0.01
CART	99.42	0.16
NB	98.73	0.26
MLR	99.87	0.03

Table 7. Confusion Matrix of MLP with hidden layer (3,2) for single-fault classification (Friction (F), Healthy (H), Leak (L) and Spring (S)).

	F	H	L	S
Pred F	1000	0	0	0
Pred H	0	1000	0	0
Pred L	0	0	999	1
Pred S	0	0	1	999

5.3. Multi-fault experiments

Table 8 shows the accuracy comparison of several models with the step input and the times (t1, t2, t3 and t4) as features and also with the environmental condition (temperature and pressure) in case of accuracy lower than 90%. The input type and the predictors were defined based on the analysis shown in Tables 3 and 5, respectively. The SVM algorithm had a superior accuracy, for this reason, the confusion matrix of SVM after the test set being applied to this model is shown in Table 9 considering the cost = 2. Analyzing the confusion matrix, we can notice that there is an overlap between the LSF and FL classes.

6. Conclusion

This work compared machine learning methods to diagnose failures of a simulated open-loop bleed valve. Based on this study we can conclude:

Table 8. Accuracy (Acc) and standard deviation (Dev) results for machine learning algorithms in multi-fault classification.

4 inputs			7 inputs		
Method	Mean Acc	Dev	Method	Mean Acc	Dev
k -NN ($k=9$)	89.25	0.37	k -NN ($k=3$)	85.08	0.43
SVM (cost=2)	90.14	0.51	SVM (cost=2)	94.28	0.30
MLP (3,2)	84.69	0.41	MLP (7,0)	87.7	1.85
CART	80.25	0.69	CART	80.25	0.69
NB	82.91	0.45	NB	82.94	0.49
MLR	49.74	0.08	MLR	49.88	0.06

Table 9. Confusion Matrix of SVM with cost = 2 for multi-fault classification (Friction (F), Healthy (H), Leak (L), Spring (S), Leakage and Friction (LF), Leakage and Spring (LS), Spring and Friction (SF) and All Faults occurring simultaneously (LSF)).

	F	H	L	LF	LS	LSF	S	SF
Pred F	500	0	0	0	0	0	0	0
Pred H	0	500	0	0	16	0	0	0
Pred L	0	0	497	0	0	0	0	0
Pred LF	0	0	0	445	0	151	0	1
Pred LS	0	0	0	0	484	0	0	0
Pred LSF	0	0	0	52	0	330	0	0
Pred S	0	0	3	0	0	0	500	0
Pred SF	0	0	0	3	0	19	0	499

- The type of PRSOV input signal is relevant to produce attributes which allow a better classification performance of the machine learning algorithms. An input command with a higher rate than the input used in [Castilho et al. 2018] increased the mean accuracy of diagnostics methods when compared with the accuracy obtained in that previous study.
- The environmental conditions variables are relevant mostly to the diagnostics of concurrent failures.
- Regarding the learning algorithms for the single fault scenario, all methods presented an accuracy greater than 90% and the MLP method presented the greatest accuracy (99.9%).
- In the multi-fault diagnostics, where even simpler approaches (such as NB) has an accuracy greater than 80%. The SVM algorithm provided the model with the greatest mean accuracy for the concurrent failure diagnostics (94.3%). The confusion matrix for this method also shows that there is an overlap between the LSF and LF classes.

Further studies could evaluate the machine learning methods on a closed-loop operation of this valve considering the normal operational commands which the ECS system applies. This other approach could result in a simpler and cheaper ECS system.

References

Baptista, M., Nascimento Jr, C., Prendinger, H., and Henriques, E. (2017a). A case for the use of data-driven methods in gas turbine prognostics. In *Annual Conference of the*

Prognostics and Health Management Society, pages 1 – 10.

- Baptista, M., P. de Medeiros, I., Malere, J., Nascimento Jr, C., Prendinger, H., and Henriques, E. (2017b). Comparative case study of life usage and data-driven prognostics techniques using aircraft fault messages. *Computers in Industry*, 86:1–14.
- Caesarendra, W., Widodo, A., and Yang, B.-S. (2010). Application of relevance vector machine and logistic regression for machine degradation assessment. *Mechanical Systems and Signal Processing*, 24(4):1161 – 1171.
- Castilho, H. M., Nascimento, C. L., and Vianna, W. O. L. (2018). Aircraft bleed valve fault classification using support vector machines and classification trees. In *2018 Annual IEEE International Systems Conference (SysCon)*, pages 1–7.
- Daigle, M. J. and Goebel, K. (2013). Model-based prognostics with concurrent damage progression processes. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(3):535–546.
- Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., Weingessel Maintainer, A., Leisch@ci, f., Tuwien, A., and , A. (2006). The e1071 package. pages 32 – 48.
- Günther, F. and Fritsch, S. (2010). neuralnet: Training of neural networks. *R Journal*, 2:30–38.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 edition.
- Jennions, I. K. (2014). *Integrated Vehicle Health Management: Implementation and Lessons Learned*. SAE International.
- M. Therneau, T. and Atkinson, E. (1997). An introduction to recursive partitioning using the rpart routines. *Mayo Clinic*, 61:5–9.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition.
- PHMSociety (2019). Phmsociety. <https://www.phmsociety.org>. Accessed: 2019-05-25.
- Ripley, B. and Venable, W. (2019). Functions for classification. <https://cran.r-project.org/web/packages/class/class.pdf>. Accessed: 2019-06-01.
- Russell, S. and Norvig, P. (2009). *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition.
- Tibshirani, R., Hastie, T., and Friedman, J. (2010). Regularized paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33:9–10.
- Turcio, W., Yoneyama, T., and Moreira, F. (2013). Quasi-l6pv gain-scheduling control of a nonlinear aircraft pneumatic system. In *21st Mediterranean Conference on Control and Automation*, pages 341–350.
- Vianna, W. O. L. and Yoneyama, T. (2018). Predictive maintenance optimization for aircraft redundant systems subjected to multiple wear profiles. *IEEE Systems Journal*, 12(2):1170–1181.