On the potential of ruled-based machine learning for disruption prediction on JET

On the Potential of Ruled-Based Machine Learning for Disruption Prediction on JET

EUROfusion Consortium, JET, Culham Science Centre, Abingdon, OX14 3DB, UK

by M.Lungaroni[1], A.Murari[2], E.Peluso[1], J.Vega[3], G.Farias[4] and M.Gelfusa[1] and JET Contributors*

1) Department of Industrial Engineering, University of Rome "Tor Vergata", via del Politecnico 1, Roma, Italy
2) Consorzio RFX (CNR, ENEA, INFN, Universita' di Padova, Acciaierie Venete SpA), Corso Stati Uniti 4, 35127 Padova, Italy
3) Laboratorio Nacional de Fusión, CIEMAT. Av. Complutense 40. 28040 Madrid. Spain
4) Pontificia Universidad Catolica de Valparaiso, Av. Brasil 2147, Valparaiso, Chile

*See the author list of "X. Litaudon *et al* 2017 *Nucl. Fusion* 57 102001"

Abstract

In the last years, it has become apparent that detecting disruptions with sufficient anticipation time is an essential but not exclusive task of predictors. It is also important that the prediction is accompanied by appropriate qualifications on its reliability and it is formulated in mathematical terms appropriate for the task at hand (mitigation, avoidance, classification etc.). In this paper, a wide series of rule-based predictors, of the Classification and Regression Trees (CART) family, have been compared to assess their relative merits. An original refinement of the training, called noise-based ensembles, has allowed not only to obtain significantly better performance but also to increase the interpretably of the results. The final predictors can indeed be represented by a tree or a series of specific and clear rules. Such performance has been proved by analysing large databases of shots at JET with both the carbon wall and the ITER Like Wall. In terms of performance, the developed tools are therefore very competitive with other machine learning techniques, with the specificity of formulating the final models in terms of trees and simple rules.

Keywords: Disruptions, Machine Learning Predictors, Decision Support Systems, Classification And Regression Trees, Boosting, Bagging, Random Forests

Corresponding author: emmanuele.peluso@uniroma2.it

1 Rule-based Machine learning for disruption prediction in Tokamaks

Since they can compromise the integrity of large Tokamaks, particularly in the parameter range of the next generation of devices, disruptions have been intensively studied in the last decades [1,2]. These studies range from mitigation techniques, such as massive gas injection, to prediction and avoidance strategies. Of course, reliable forecasting tools are an essential ingredient in the implementation of any mitigation or avoidance intervention. Unfortunately, the theoretical understanding of disruption causes is not sufficient to program reliable simulation models for forecasting. Consequently, in the last decades, many efforts have been devoted to deriving empirical models from experiments, to identify the boundary between the safe and disruptive regions of the operational space. Among these empirical models, the most performing are based on machine learning tools. On JET two generations of machine learning predictors, APODIS and SPAD [3-7], have been implemented in the real time network. These classifiers, and the others tested offline, are based on various machine learning techniques, ranging from the distance based ones (SVM and Neural Networks), to clustering and fuzzy logic [8-10]. A family of techniques not significantly explored are the rule based ones, which are the subject of this paper.

In the field of computer science, the term rule-based machine learning (RBML) indicates the machine learning methods that extract “rules” to solve a problem directly from the data available. The defining aspect of rule-based machine learners, in their application to data mining, is their capability to identify a set of relational rules that best represent the relevant knowledge in the data, for the solution of the problem at hand. This is in contrast to traditional rule-based systems, which are hand-crafted and therefore simply encode already available, prior human knowledge. Expressing the data driven knowledge as rules is a significant advantage for both the interpretation and the implementation of the results, as will become clear in the next sections. The methods implemented and refined to perform the studies described in the rest of the paper are based on the Classification And Regression Tree (CART) technology. This technique allows producing a tree summarising the rules as the final output.

Rule-based classifiers of the CART family are very powerful and easy to interpret. On the other hand, one of their main problems is the sensitivity to the details of the training set. Their final trees are indeed not very stable; small changes in the training set can result in major differences in the final trees. To alleviate this problem, the approach of ensemble rule-based classifiers has proved to be very successful. It consists of training many even not very performing classifiers and then somehow average their results in order to obtain the final classification of the new examples.

In order to follow the evolution of the operational space during the campaigns, an adaptive form of training has been adopted. Such a training has also the advantage of optimising the computational efforts by minimising the training set. This procedure implements a "*learning from scratch*" approach so that all the proposed predictors can start working with just one disruptive and one non disruptive example [11, 12]. The last model is updated as the campaign progresses, by refining the training with additional cases.

Regarding the structure of the paper, next section gives an overview of the rule-based classifiers of the CART family. Section 3 introduces the methodology of the ensemble rule-based classifiers, Section 4 discusses in detail the adaptive method adopted to train the various versions of the predictors and describes the main characteristics of JET database investigated. The results obtained for the ILW and a Carbon wall are reviewed in Sections 5 and 6. The conclusions and lines of future work are the subject of the last Section 7 of the paper.

2 The basics of Classification Tree analysis

Nowadays the reference, basic rule-based machine learning tools are the so called Classification and Regression Trees (CART). They have been widely implemented for constructing prediction models from data [13]. Such models are derived directly from the available databases by recursively partitioning the data space and fitting a simple prediction rule at each partition. The final partitioning, once properly optimised, consists therefore of a series of rules that can be represented graphically by a decision tree. Classification trees, the subject of this paper, have been conceived to classify dependent variables that take a finite number of unordered values. Their performance are therefore typically quantified in terms of misclassification costs. Regression trees are an extension used to handle dependent variables that take continuous or ordered discrete values, with prediction error typically measured by the squared difference between the observed and predicted values.

Decision trees are supervised techniques and therefore require the a priori definition of the number of classes and a sufficient number of examples. In the applications described in this paper, decision trees are used to solve classification problems, which mathematically can be formalised as follows. Given a training sample of n observations, the class variable is indicated by Y and can in general take a finite set of discrete values 1, 2, ... , k. In our application, the number of classes is typically 2. The set of p features used as predictor variables are indicated by $X_1,..., X_p$. The objective of the analysis consists of finding a model, which can predict the class Y from new X values. The method to identify the best model consists of partitioning the database one node at the time starting from the root. The algorithm exhaustively searches the whole database to determine which variable and which value minimize the total impurity of its two child nodes. To quantify the purity of a node, the version of CART implemented for the studies of this paper uses a generalization of the binomial variance called the Gini index. As a metric to split the nodes, the Gini impurity calculates how often a randomly chosen element from the training set would be incorrectly labelled, under the assumption that the labels are allocated as the distribution of labels in the subset. The Gini factor is typically computed by summing the probability p_i of the item being correctly classified by the probability $(1-p_i)$ of the item being wrongly classified

$$\mathrm{GINI} = \Sigma\, p_i(1-p_i) \qquad (1)$$

Where the sum is extended over the number of classes. The GINI impurity reaches its minimum (zero) when all cases in the node fall into a single target category.

3 Ensemble rule-based classifiers

Ensemble rule-based classifiers implement the concept of weak learners. A 'weak' learner (either classifier or predictor) is just a machine learning tool, which produces a model that performs relatively poorly but is often, but not always, computationally simple. The relatively limited computational resources required allow training various versions of such weak learners which can then be pooled (via Bagging, Random Forests etc) together to create a "strong" ensemble classifier. The basic elements of the ensembles used in this paper are decision trees of the type described in the previous section. The next subsections provide some details about the various weak learners trained and pooled to obtain the results reported in the rest of the paper. These techniques are nowadays quite standard; the original methodological development introduced in this treatment is the category of so called *noise-based ensemble classifiers*, which take into account the effects of the noise on the measurements.

3.1 Bagging

One of the main weaknesses of decision trees is the sensitivity of their results to the specific data used for their training. A small change in the inputs (for example even using a subset of the training data) can imply a major variation in the resulting decision tree and in turn quite different predictions. Bagging is an application of ensemble weak learners to reduce this a high-variance of decision trees. Bagging of the CART algorithm would consist of the following steps:

1. Generation of many random sub-samples of the original dataset with replacement.
2. Training of a CART model for each subset of samples.
3. Given a new example, calculate the average prediction from each model and select the class with a form of majority vote.

When Bagging, individual tree overfitting the training data is less of a concern. For this reason, the individual decision trees can be grown deep and not or less pruned. Of course, these trees will tend to have both high variance and low bias. The high variance is then remedied by using Bagging. The main adjustable parameter of Bagging is the number of trees; this parameter can be chosen by increasing the number of trees until the accuracy begins to saturate instead of improving. Very large numbers of models will take longer to train but will not overfit the training data.

3.2 Random Forests

Random Forests or random decision forests are another ensemble learning method for classification based on constructing a multitude of decision trees. In a certain sense, Random Forests extend bootstrapping since they build multiple CART models with different sample and different initial variables. For a given number of trees T Random Forests are trained as follows:

1. Sample the original dataset at random with replacement to create a subset of the data. Indicatively each subset should have a size of about 66% of the total set.
2. At each node:
    1. select at random a subset of predictor variables from all the predictor variables.
    2. The predictor variable that provides the best split, according to some objective function, is used to do a binary split on that node.
    3. At the next node, another subset of predictor variables is chosen at random from all predictor variables and the best one is used to split the node.

The final prediction is a function of each prediction obtained again with some sort of majority voting.

3.3 Noise-based Ensembles

One of the main issues of the measurements in Tokamaks is the high levels of noise. The resulting uncertainties in the data can therefore reach 30% of the measured values, even if the range 10-20% is more common. This noise is very difficult to reduce; the source of noise are many and independent. Even if these uncertainties are a potential issue, they suggest an alternative approach to the method of building ensembles of weak classifiers, which is an innovation proposed for the first time in this paper. The idea consists again of collecting ensembles but not with subsets of the original data; on the contrary the various training sets are obtained by the original one summing random noise to the measurements. The random noise is generated from Gaussian distributions with variance equal to the error bar of the measurements. Again the number of trees can be increased until the accuracy begins to saturate instead of improving. This approach of Noised-based Ensembles can be applied directly to CART trees. It can also be combined with Bagging and Random Forests; to this end, for each member of the ensemble obtained with traditional Bagging or random forest methods various weak classifiers are trained by adding different noise realizations to the inputs (see Section 5).

4 The Adaptive Training and JET Databases

This section discusses the method used to train the various RBML tools (Subsection 4.1) and provides a general overview of JET databases to which they have been applied (Subsection 4.2). Information about the computational requirements of the implemented methods is provided in Subsection 4.3.

4.1 Training of the adaptive predictor from scratch

As already mentioned, large devices cannot afford collecting many disruptive examples to train machine learning predictors with the traditional data mining algorithms. It is therefore important to devise efficient training methods, which can start predicting with a limited amount of examples; they have also to be sufficiently adaptive to preserve good performance as the campaigns progress.

Various solutions can be adopted to train an adaptive predictor. In harmony with previous instances [9-12], a quite simple strategy has been implemented for the training. The predictors are designed with a "*from scratch*" approach and therefore need only one disruptive and one non disruptive case to build the first model. In the campaigns analysed, the first disruption occurred after many safe discharges and therefore the first model was obtained after the first disruption. For the disruptive discharge, 15 ms before the beginning of the current quench have been divided in 5 intervals of 3 ms each and the averages of the selected signal features over these five intervals have been used as input to the training. The 5 discharges prior to the first disruptive one have been used as examples for the safe case. For each of these discharges, two random periods of 20 ms, with plasma current above 750 kA, have been averaged and the averages over these intervals have been used as inputs for the training.

The model derived as previously described needs to adapt by learning how a) disruptivity conditions vary and b) the safe space of operation changes. In other words, the model has to be automatically updated to follow the evolution of the boundary between the disruptive and non disruptive regions of the operational space as the experimental campaigns evolve. To this end a model, starting with the one obtained after the first training described above, is used for the following discharges until the first missed alarm. When the previous model misses a disruption, the shot not properly classified is included in the training set. In this way a new model is determined, which is deployed to analyse the following discharges until the next error, which provides an example for a new retraining.

If the error is a false alarm, it is not appropriate to insert that example in the training set and retrain the predictor. In closed loop real time applications, indeed, it is not necessarily the case that false alarms can be always recognised, after the discharge has been shutdown following the received alarm. Of course post pulse investigations by the experts and analysis of possible technical faults can provide indications that the discharge was stopped prematurely, but this cannot be assumed to happen systematically. On the other hand, retraining only on the basis of disruptive examples can cause the number of false alarms to increase unnecessarily, particularly during the course of long campaigns and/or when the scenarios evolve and new regions of the operational space are explored. As a compromise solution, adopted to obtain the results described in the following, the retraining has been performed with a new safe example every time the model launches an alarm. After each alarm the previous discharge, if safe as it is normally the case, is used as an example of a non disruptive discharge to retrain. Now for the training two 10 ms intervals, around the maximum value of the locked mode, have been averaged and these averages are the features for the new training.

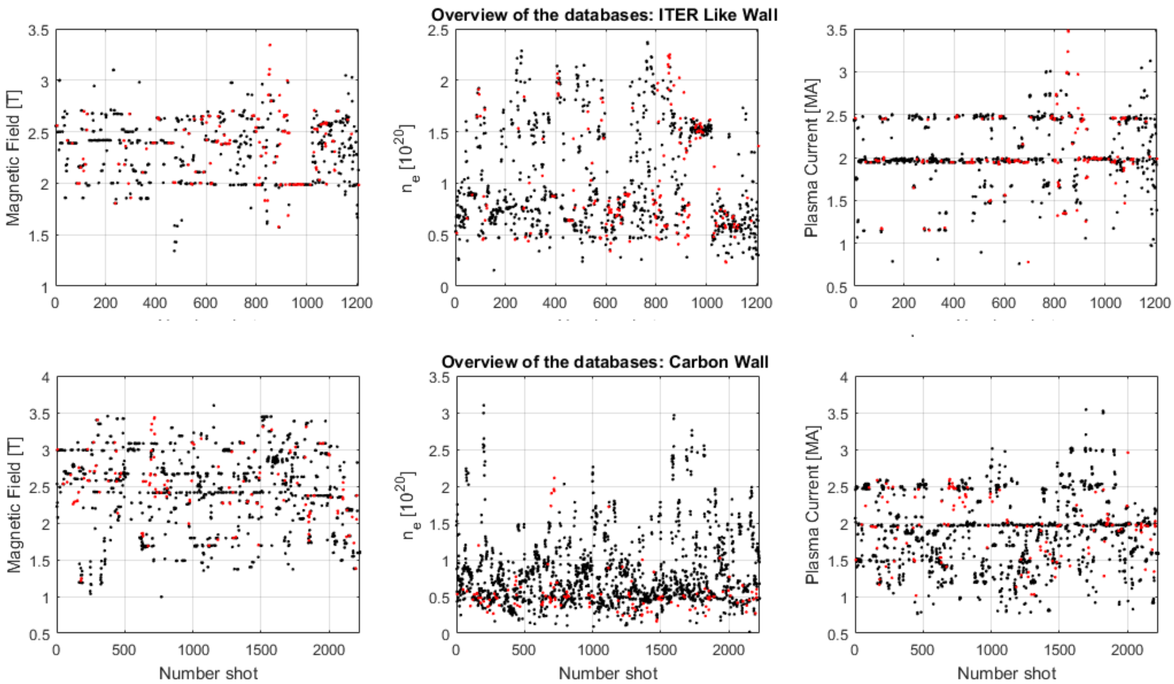

Figure 1. Overview of the databases for the Carbon wall and ILW- A characteristic point for each shot in the database has been reported. The red points belong to disruptive shots.

Of course, in the case of closed loop applications of the predictors, more sophisticated strategies could be implemented, such as retraining when new scenarios are developed and run or by identifying some false alarms. Therefore the results reported in the following, even if quite good, have to be considered an underestimate of the possible performance in terms of false alarms.

4.2 JET Databases: ITER Like Wall

In building both databases, only non-intentional disruptions have been retained from the training. Indeed intentional disruptions do not need to be predicted and, being typically different from naturally occurring disruptions, can affect the quality of the adaptive training. Only time slices, whose plasma current exceeds 750 kA, have been considered but no other general selection has been implemented. All the signals have been resampled at 1kH frequency. Since 10 ms is considered the minimum time required on JET to undertake mitigation action, alarms, which are launched 10 ms or less from the beginning of the current quench, are considered tardy. Alarms triggered more than 2.5 s before the beginning of the current quench are considered early, even if this choice is a bit penalising because in various instances, indeed, the predictors have detected an almost disruptive situation but the plasma just managed to survive longer than 2.5 s. Therefore if an alarm had been launched in these cases, since the quality of the plasmas had already been compromised, in general no useful experimental time would have been lost and time for a soft landing would have been available. Therefore keeping these cases in the list of the not properly classified discharges is a conservative choice.

Table I The main training methods adopted for the various versions of the predictors

Case number	Method	Number of Trainings	Noise level in %
1	CART	1	0
2	CART	11	5
3	CART	11	10
4	RF	1	0
5	RF	11	0
6	RF	11	5
7	RF	11	10
8	BAG	1	0
9	BAG	11	0
10	BAG	11	5
11	BAG	11	10

Coming to the database with the ILW wall, the campaigns C29 to C31 have been considered. After proper cleaning and validation of the database, overall 187 disruptive and 1020 non disruptive shots are included. A plot showing the operational space covered by the database is shown in Figure 1 (top row).

The database of the Carbon wall includes the discharges of campaigns C15a, C15b, C16, C1617, C18 and C19 (from shot 65988 to 70749). Overall, 143 disruptive and 2083 non-disruptive shots are included. A plot showing the operational space covered by the database is shown in Figure 1 (bottom row).

4.3 Computational aspects

With the adaptive approach implemented, and described in detail in subsection 4.1, the computational time has been calculated for a typical shot of 20 s, using the input signals of the locked mode amplitude and the internal inductance. On a Dell XPS 13 9350 – 1 Processor i7 6600U 2.5GHz (4 MB L3 Cache) with 8 GB Dual-Channel DDR3 1867MHz, OS Windows 10 Pro 64-bit the computational requirements for the deployment of the proposed tools are of about 2.2 s for the noise-based CART ensemble, about 4.3 s for Bagging and 4.9 s for the Random Forests.

5 Results for disruption prediction in JET with the ITER Like Wall

The various predictors described in Sections 2 and 3 have been applied to the experiments of the campaigns C29-C31. For consistency with past tests using other methods, the amplitude of the locked mode signal and the internal inductance have been used as features. A summary of the tested alternatives is reported in Table I. Each main typology of tree (CART, Bagging and Random Forests) has been trained in the traditional way and at various levels of added noise. For each level of noise, 11 independent realisations of the input data have been generated. The traditional versions of Bagging and Random Forests consist of 40 different trees: in the case of non zero noise, each of the 40 trainings has been performed with an individual realisation of the 11 noised inputs (for a total of 440 independent trees). In all the cases, except the simple CART without noise, the final decision about whether triggering an alarm is reached with the method of majority voting.

An overview of performance, for each alternative training method, is provided in Table II. CART without noise give clearly inferior results. Once the approach of noised based ensemble is implemented, the simple CART trees become very competitive with Bagging and Random Forests. In their turn, the approaches of Bagging and Random Forests are very performing in terms of success rate even if the original signals are provided as inputs without noise. On the other hand, adding noise to the inputs has a significant positive effect in reducing the false alarms also for these methods. Therefore, in general, the approach of providing a series of different inputs signals, with different realisations of the noise, has a quite positive effect on all the methods. The choice of the more appropriate level of noise is not easy and depends on the relative importance is given to success rate versus false alarms. Increasing the noise level from 5% to 10% typically reduces the number of false alarms but also slightly the success rate. It has been checked that increasing the added noise above 10% translates into a serious worsening of performance.

It is good to remember that in Tables 2 and 3 the sum of the performances of disruptive discharges, good, missed, early and tardy, corresponds to 100% of the disruptive shoots, 187 for the ILW and 143 for the CW. The False Alarm column is in percentage of the total 1020 non-disruptive shoots for the ILW and 2083 non-disruptive shoots for the CW.

The training of simple CART trees with different noise realisations allows to easily determine the structure of the typical tree in the set. Obtaining the same information from Bagging is much more difficult and practically impossible in the case of Random Forests.

Table II The traditional performance indicators, used to determine the quality of disruption predictors expressed in percentage, for the ILW campaigns.

Case Number	Succes Rate	Missed	Early	Tardy	False	Mean [ms]	Std [ms]
1	83.87	1.07	10.75	4.30	9.46	323	333
2	94.62	0.54	0.54	4.30	2.36	310	330
3	93.01	1.61	0.54	4.84	1.97	310	330
4	94.08	1.07	1.07	3.76	6.19	323	339
5	94.62	1.07	1.07	3.22	4.32	322	342
6	94.08	1.61	0.54	3.76	2.07	310	328
7	94.08	1.07	0.54	4.30	1.27	297	321
8	94.62	1.07	1.07	3.22	4.42	321	340
9	94.62	1.07	1.07	3.22	4.11	322	341
10	94.08	1.61	0.54	3.76	1.67	314	333
11	93.54	1.61	0.54	4.30	1.57	302	328

Visualizing a tree is of course of great help in terms of interpretability. The most performing tree in the set of 11 CART, trained with different noise realizations, is shown in Figure 2. The key indicators for this tree are: success rate 97.33 % and false alarms 2 %. It is worth noting that this tree is in any case quite representative of the whole family.

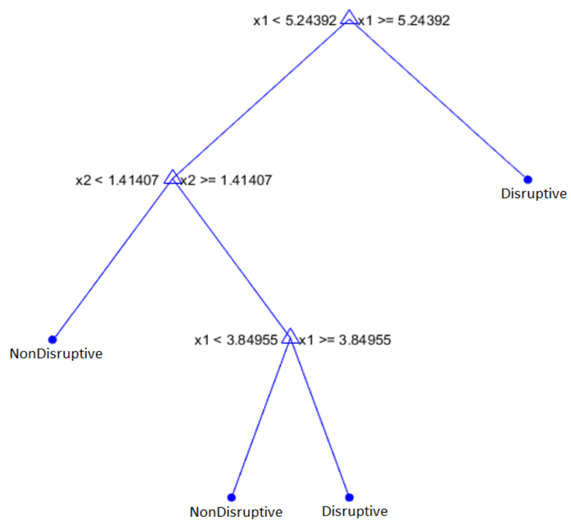

Figure 2. The most performing tree of the 11 CART ones trained with different realisations of the noise of the campaigns with the ILW. In the tree, x_1 indicates the amplitude of the locked mode and x_2 the internal inductance.

6 Results for JET with a Carbon Wall

The same tools, introduced in Sections 2 and 3, have been deployed also to analyse a database of disruptions of JET with the carbon wall. The main objective of this exercise is to confirm the general applicability of the developed techniques, Again, the amplitudes of locked mode and the internal inductance signals have been selected as input features. The adaptive training procedure, described in detail in Section 4.1, has also been followed. For consistency sake, the same cases reported in Table I have been tested also for the campaigns of JET with a Carbon wall.

Table III The traditional performance indicators, used to determine the quality of disruption predictors expressed in percentage, for the carbon wall campaigns.

Case Number	Success Rate	Missed	Early	Tardy	False	Mean [ms]	Std [ms]
1	88.03	0.00	10.56	1.40	17.39	297	371
2	95.07	1.40	1.40	2.11	3.79	265	317
3	95.77	1.40	1.40	1.40	2.83	259	316
4	89.44	0.00	9.15	1.40	8.80	285	367
5	88.03	0.00	10.56	1.40	17.30	296	371
6	94.36	2.11	2.81	0.70	2.92	271	322
7	95.07	1.40	1.40	2.11	1.58	256	317
8	90.84	0.00	7.74	1.40	13.40	281	350
9	88.03	0.00	10.56	1.40	17.30	296	371
10	92.96	2.11	2.81	2.11	3.74	267	320
11	96.48	0.70	1.40	1.40	2.54	271	342

The performance of the proposed tools is summarised in Table III. For this database, the approach of Noise-based Ensembles has a significant improving effect even on the success rates of Bagging and Random Forests and not only on reducing the false alarms. Therefore noise-based ensembling is confirmed to improve performance significantly and to render simple CART trees very competitive. A representative example of such a CART is shown in Figure 3, whose overall performances are about 96.5% of success rate and 1.6 % of false alarms. The structure of this tree is similar to the one obtained for the ILW but a bit more complex, reflecting the larger spectrum of discharges explored in the campaigns with the Carbon Wall.

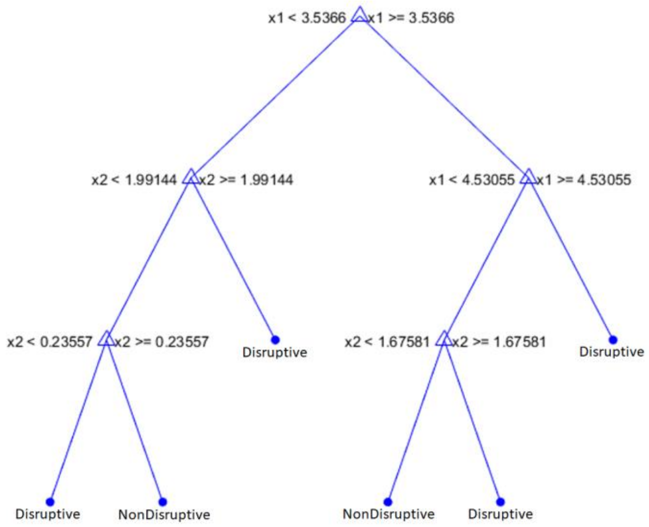

Figure 3. The most performing tree of the 11 CART ones trained with different realisations of the noise for the Carbon Wall. In the tree, x1 indicates the amplitude of the locked mode and x2 the internal inductance.

7 Discussion and Conclusions

The development of RBML classifiers has allowed the implementation of a series of tools for disruption prediction, which provide very competitive performance. The generality of the developed tools has been confirmed by their deployment to predict disruptions in JET for both the Carbon wall and the ITER Like Wall.

The innovative approach of Noise-based Ensembles has proved to be particularly effective. In the case of traditional CART it increases the success rate of about 10 percentage points, bringing it in line with the traditional ensemble methods. If enough computational power is available to apply the approach also to Random Forests and Bagging, the improvement is mainly in a reduction of a couple of percentage points in the false alarms. In the case of the carbon wall, implementing the Noise-based Ensembles improves also the success rate of a couple of percentage point. So, in general, the noise-based approach seems to be a quite effective method to train ensembles of weak learners to reduce both their variance and bias.

The positive results obtained with the application of noise-based ensembles to traditional simple CART tress is not important only from the point of view of performance. The other remarkable advantage is the increase in interpretability, because a performing tree can typically be visualized and a series of specific rules made explicit.

With regard to future developments, it is planned to utilise rule-based classifiers of the CART family also for avoidance. In this perspective, the presented tools could also become important in the selection of the most useful features to obtain early warnings for avoidance.

Aknowledgements

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission. This work was also partially supported by the Spanish Ministry of Economy and Competitiveness under the Projects Nos. ENE2015-64914-C3-1-R and ENE2015-64914-C3-2-R. This work was partially supported by Chilean Ministry of Education under the Project FONDECYT 1161584.

References

[1] J.Wesson "*Tokamaks*" Oxford University Press 2011

[2] R.Wenninger et al "*Power Handling and Plasma Protection Aspects that affect the Design of the DEMO Divertor and First Wall*" submitted for publication in Proceedings of 26th IAEA Fusion Energy Conference

[3] G.Rattà et al Nuclear Fusion, Volume 50, Number 2 January 2010 doi.org/10.1088/0029-5515/50/2/025005

[4] Y.Zhang et al Nuclear Fusion, Volume 51, Number 6 May 2011 doi.org/10.1088/0029-5515/51/6/063039

[5] J. Vega, S. Dormido-Canto, J. M. López, A. Murari, J. M. Ramírez, R. Moreno, M. Ruiz, D. Alves, R. Felton and JET-EFDA Contributors. "Results of the JET real-time disruption predictor in the ITER-like wall campaigns". Fusion Engineering and Design 88 (2013) 1228-1231.

[6] J. Vega, R. Moreno, A. Pereira, S. Dormido-Canto, A. Murari and JET Contributors. "Advanced disruption predictor based on the locked mode signal: application to JET". 1st EPS Conference on Plasma Diagnostics. April 14-17, 2015. Book of abstracts. Frascati, Italy.
[7] J .Vega, A. Murari, S. Dormido-Canto, R. Moreno, A. Pereira, G. A. Rattá and JET Contributors. "Disruption Precursor Detection: Combining the Time and Frequency Domains". Proc. of the 26th Symposium on Fusion Engineering (SOFE 2015). May 31st-June 4th, 2015. Austin (TX), USA
[8] A.Murari et al Nuclear Fusion, Volume 49, Number 5 April 2009 doi.org/10.1088/0029-5515/49/5/055028
[9] A.Murari et al Nuclear Fusion, Volume 48, Number 3 February 2008 doi.org/10.1088/0029-5515/48/3/035010
[10] B.Cannas et al Nuclear Fusion, Volume 53, Number 9 August 2013 doi.org/10.1088/0029-5515/53/9/093023
[11] J. Vega, R. Moreno, A. Pereira, S. Dormido-Canto, A. Murari and JET Contributors. "Advanced disruption predictor based on the locked mode signal: application to JET". Proceedings of Science . ECPD 2015, 028
[12] J. Vega, A. Murari, S. Dormido-Canto, R. Moreno, A. Pereira, A. Acero and JET-EFDA Contributors. "Adaptive high learning rate probabilistic disruption predictors from scratch for the next generation of tokamaks". Nuclear Fusion. 54 (2014) 123001 (17pp).
[13] Leo Breiman, Jerome Friedman, Charles J. Stone, R.A. Olshen "*Classification and regression trees*" Taylor & Francis, 1984