EFDA–JET–PR(13)44

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

Adaptive High Learning Rate Probabilistic Disruption Predictors from Scratch for the NextGeneration of Tokamaks

J. Vega[1], A. Murari[2], S. Dormido-Canto[3], R. Moreno[1], A. Pereira[1], A. Acero[1]
and JET EFDA contributors*

JET-EFDA, Culham Science Centre, OX14 3DB, Abingdon, UK

[1]Asociación EURATOM/CIEMAT para Fusión, Madrid, Spain
[2]Associazione EURATOM/ENEA per la Fusione, Consorzio RFX, 4-35127 Padova, Italy
[3]Departamento Informática y Automática - UNED, Madrid, Spain
** See annex of F. Romanelli et al, "Overview of JET Results",*
(24th IAEA Fusion Energy Conference, San Diego, USA (2012)).

Preprint of Paper to be submitted for publication in
Nuclear Fusion

ABSTRACT

The development of accurate real time disruption predictors is a pre-requisite to any mitigation action. '*Accurate real time prediction*' has to be understood in terms of high success rate, low rates of false alarms as well as an early recognition of an incoming disruption. Only approximate theoretical models of disruptions exist and they do not reliably cope with the disruption issues. Therefore, this article deals with data-driven predictors with a view on ITER and DEMO. A review of existing prediction techniques, from both physics and engineering points of view, is provided. All these methods have to use large training datasets to develop successful predictors. However, ITER or DEMO cannot wait for hundreds of disruptions to have a reliable predictor. So far, the attempts to extrapolate predictors between different tokamaks have not shown satisfactory results. Moreover, it is not clear how valid this approach can be between present devices and ITER/ DEMO, due to the differences in their respective scales and possibly underlying physics. So, this article analyses the requirements to create adaptive predictors from scratch to learn from the data of an individual machine from the beginning of operation. A particular algorithm based on probabilistic classifiers has been developed and it has been applied to the database of the three first ITER-like wall experimental campaigns of JET (1036 non-disruptive and 201 disruptive discharges). The predictions show a success rate of 94%, a false alarm rate of 4.21% and an average warning time of 654 ms. The average probability interval about the reliability and accuracy of all the individual predictions is 0.811 ± 0.189. It should also be mentioned that a very limited number of signals is required by the predictor, an important point particularly at the beginning of the operation of new devices.

1. INTRODUCTION

A disruption is a catastrophic loss of plasma control which not only results in the plasma extinction but also in large power and force loads on the surrounding structures. These loads can produce irreversible damage to the fusion device and, therefore, disruptions can be a big issue in future reactors. As primary goal, the total number of disruptions must be kept as low as possible.

To deal with disruptions, two different concepts are typically used: avoidance and mitigation. The former is closely related to the scenario development and the plasma operational space [1, 2]. The latter involves any possible technique to alleviate the harmful potential effect of disruptions and, even, to initiate a safe plasma shut-down [3, 4, 5, 6, 7]. In other words, avoidance is associated with the aim of free disruption operation whereas mitigation is aimed at the alleviation of the disruption detrimental effects. But it is worth highlighting that a pre-requisite to any mitigation method is to have a reliable real-time disruption predictor during the discharge production.

Ideally, plasma theoretical models can guide operation and provide indications about disruption avoidance and prediction. However, the existing models and simulation tools have performances far from those that are needed [8]. Some related problems are: incomplete models, strong assumptions and unphysical boundary conditions.

Data driven-models have shown to be a practical alternative to physics-driven models. In the case of disruptions, data-driven models allow the estimation of useful relationships among several quantities with the aim of performing an accurate prediction. These relationships are typically obtained by applying pattern recognition techniques, *i.e.* by developing automatic classifiers (pattern recognition and automatic classifier are equivalent terms). Input samples with well-known behaviour (disruptive/non-disruptive) are used to train a system whose objective is to determine a decision function (frontier) to divide the parameter space in two behavioural regions: disruptive and non-disruptive (the terms 'non-disruptive' and 'safe' will be used with the same meaning in this article). The trained system is called the model. Given a sample to classify as either disruptive or non-disruptive, the classification is based on the region of the parameter space where the sample is. It should be noted that the objective of a disruption predictor is to classify the plasma behaviour as disruptive/non-disruptive at regular time intervals while a discharge is in execution. The plasma is expected to show a disruptive behaviour before the disruption itself. The classification of the plasma state as disruptive in advance of the disruption is the reason for using the term 'prediction'. It is important to note that physics-driven models are the preferred methods, first, to promote disruption avoidance strategies and, second, to accurately predict a disruption. Unfortunately as mentioned previously, only partial theoretical models exist and these models do not reliably cope with the disruption issues. However, the lack of feasible disruption theory cannot stop the research on nuclear fusion.

Data-driven models appear as a valid alternative to accurately predict the presence of an incoming disruption as a first step to trigger mitigation actions. At this point, it should be emphasised that disruption mitigation is mandatory for ITER in order to reduce forces, to alleviate heat loads during the thermal quench and to avoid runaway electrons-[9].

Data-driven models use available information from past discharges in order to create predictors. In ITER, a potential predictor could be the extrapolation of data-driven predictors from smaller machines. However, an attempt to accomplish this between JET and ASDEX Upgrade (AUG) showed unsatisfactory results [10]. Training with JET data, the success rate with AUG discharges, considering a prediction time greater than 0.01 s, was 67%. In the other direction, after training with AUG data, the success rate with JET discharges, taking into account a prediction time greater than 0.04 s, was 69%. In this context, prediction time means the interval between the alarm and the thermal quench. Moreover, nowadays, it is not clear how valid the extrapolation between present devices and ITER can be, owing to the differences in their respective scales and possibly underlying physics.

To avoid the problems just mentioned, this article deals with a data-driven approach for a reliable prediction of disruptions based on the use of the experimental data of a single fusion device. This is the typical situation in the prediction of disruptions. For example, the JET real-time Advanced Predictor Of DISruptions (APODIS) was trained/tested with 8169 discharges (7648 safe discharges and 521 unintentional disruptions) [11]. APODIS was working in open loop during the three first

ITER-like wall (ILW) campaigns of JET (years 2011-2012). The results in the recognition of disruptions after 956 discharges can be summarized in a success rate of 98.36% and a false alarm rate of 0.92%. Figure 1 shows the accumulated fraction of detected disruptions with APODIS. The horizontal axis represents the *warning time* or time to disruption from the prediction. The average *warning time* of APODIS during the 2011-2012 campaigns has been 426ms. The vertical line at 30ms shows the minimum time in JET to perform mitigation actions [12] and, therefore, 426ms of average *warning time* is a very good result. It is important to note that the accumulative fraction at 30ms has been 87.50%.

But ITER cannot wait for hundreds of disruptions to develop a successful disruption predictor. ITER will have much higher stored magnetic and thermal energies than present tokamaks. Therefore, only a limited number of disruptions are acceptable to avoid irreversible damage to the plasma facing components, to the first wall and also to the vacuum vessel.

This article looks for disruption predictors valid for ITER and DEMO. The aim is the development of high learning rate predictors *from scratch*. The term '*from scratch*' means 'with absolute lack of previous information about disruptions in a device'. Of course, the past experience of disruptions in previous fusion devices provides the basic information about the plasma signals to be considered as candidates for the predictor development.

The key point of any disruption predictor from scratch is to determine the number of disruptions that are needed to have a reliable predictor. This paper develops a Venn predictor classifier [13] from scratch that starts to make predictions from the first disruption. Venn predictors are probabilistic classifiers that determine the probability of each individual prediction. This implies that they do not make bare predictions, but they provide a probability interval about how accurate and reliable each prediction is. In terms of disruption predictors, this means that every disruptive/non-disruptive prediction is qualified with a probability interval that is used as the confidence level of the prediction.

The predictor has been applied to 1237 JET discharges corresponding to the three first ITER-like wall experimental campaigns. It has been re-trained in an adaptive way (after each missed alarm) following the chronological order of the discharges. The predictor uses a reduced number of tokamak common signals and the results show a success rate of 94%, a false alarm rate of about 4% and an average *warning time* of 654ms. The global average probability of each prediction is 0.811 ± 0.189, which provides a high degree of confidence in the classifier.

This paper is structured in 9 sections. Section 2 introduces the machine learning terminology used in the article. Section 3 reviews recent data-driven predictors for JET and AUG. Section 4 summarizes the results of pioneering predictors from scratch also tested with JET and AUG data respectively and section 5 discusses general requirements to be met for high learning rate predictors from scratch. Section 6 explains the mathematical aspects of the Venn prediction framework (as it is the base of the probabilistic predictor developed in this article) and the specific taxonomy used for the disruption predictor from scratch. Section 7 describes the specific implementation of the predictor and section 8 shows the Venn predictor results with JET campaigns. Finally, section 9 is

devoted to a short discussion of the outcomes of the paper and of the proposals for future work in the line of predictors from scratch.

2. MACHINE LEARNING TERMINOLOGY

This section summarizes the general concepts and terminology that will be used in the article.

In automatic classification problems, a sample is represented by an ordered pair $(\mathbf{x}_i, y_i)$, where $\mathbf{x}_i \in \mathbb{R}^m$ is the feature vector (*i.e.* the set of m features that characterize the sample i) and y_i is its corresponding label ($y_i \in \{C_1, C_2, ..., C_L\}$, where C_j, j = 1,..., L are the existing classes). For disruption prediction the classifiers are binary (L = 2), which corresponds to labels 'disruptive' and 'non-disruptive'. Henceforth, all reference to classifiers will mean binary classifiers unless otherwise noted.

Any automatic classification system requires a training process. In the case of supervised classifiers, the training sample labels are known. The prediction process is made up of two steps: induction and deduction (fig. 2). The first one is an inductive phase by means of which the real learning process is carried out. The training samples, $(\mathbf{x}_i, y_i)$, i = 1, ..., N, are used to obtain a general rule (also called model or decision function). From that moment, the deduction step is used to make predictions. Given any new sample represented by the pair $(\mathbf{x}, y)$ with known feature vector $\mathbf{x}$ but unknown label y, the objective is to estimate the label. The feature vector is used as input to the general rule and the model output is the predicted label. Common methods used for the inductive step in supervised classification problems are artificial neural networks [14], support vector machines (SVM) [15], k-nearest neighbours [16], self-organising maps [17] and generative topographic maps [18] among others.

After finishing the training process, a test dataset with known labels is used to estimate the classifier quality. The feature vectors of the test dataset are used as inputs to the model and the predictions are compared with the real labels. The results obtained in terms of success rate are assumed to be of the same quality for all future feature vectors that will be input to the model. Although this is not necessarily true, this is a standard hypothesis that is based on the fact that the samples are supposed to be independent and identically distributed (*i.i.d.* assumption).

Finally, it should be noted that the selection of the feature vector components is a key aspect for the success of a prediction task. The feature vectors have to provide crucial information about the configuration space of the problem and the set of components can be seen as a '*parameterization*' of such a space. Unfortunately, there are no mathematical theorems to ensure optimal choices and, therefore, the feature selection process is a trial and error procedure that is typically guided by the knowledge of the expert about the problem at hand.

Focusing the attention on disruptions, it is important to emphasise that to deal with disruption prediction, the '*parameterization*' of the configuration space depends not only on the physical quantities but also on their specific representations in both data formats (waveforms, time series data, images and, in general, any abstract form) and signal domains (time/space, frequency/wave

number or both). In this sense, even linear models can be difficult (or impossible) to interpret from a physics point of view. But it should be taken into account that the main objective of a data-driven disruption predictor is to recognise an incoming disruption, although the physics nature of the disruption is not clear. In other words, a data-driven model with these characteristics should be considered an engineering system.

3. REVIEW OF RECENT DATA-DRIVEN PREDICTORS

Disruptions have been of primary interest in medium and large fusion devices. A well-known scenario that causes a disruption is associated to the plasma MHD activity. A rotating island starts to grow, it is slowed down and finally locks and continues growing until the final disruption. The physical mechanism responsible for mode locking is the braking effect of error fields or MHD modes (such as resistive wall modes) that slow down and ultimately stop the plasma rotation [19]. The resulting high amplitude mode degrades the plasma confinement and can subsequently lead to a disruption. The relation between MHD activity and disruptions is well-known since many years [20, 21]. Due to this fact, a common way of disruption prediction is the use of a mode lock signal [22, 23]. At each time instant, the mode lock amplitude is compared to a user-defined threshold. If the amplitude crosses such threshold, an alarm is triggered. This very simple real-time disruption predictor based on a single signal is not reliable enough to manage the detection of a percentage of disruptions close to 100% [11, 12, 24]. Due to this fact, the alternative to the mode lock trigger has been the development of machine learning methods, which can cope with both high dimensional problems and very large databases.

In general, machine learning methods have been used for the development of automatic systems able to recognize the presence of incoming disruptions. Machine learning methods have been developed under two different approaches [25]. The first one is based on forecasting real measurements of signals that are known to be an indicator of incoming disruptions. These ideas were applied to the ADITYA tokamak [26]. However, the main drawback of this implementation was the limited number of disruptions that could be predicted. The second approach consists in the development of supervised binary classification systems to distinguish between disruptive and non-disruptive behaviours. This kind of predictors are based on finding a relation among different signals but without taking into account the physical mechanisms that are responsible for the disruptive event. Recent developments of this kind of predictors can be found in [24, 11, 27, 28, 29, 30] and a brief summary of the most recent ones follows.

In the first reference [24], a special combination of Support Vector Machines (SVM) classifiers and signal information in the frequency domain were used. In particular, 13 signals were considered. The feature vector components are the standard deviation of the power spectrum of the signals after removing the DC component. The reason of this choice resides in the fact that the frequency spectrum can be used as a precursor of disruptions. Empirically, as a disruption is approaching, some plasma signals present more spectral components due to the presence of higher frequencies.

This is translated in terms of a higher standard deviation in the power spectrum. A large dataset extracted from the JET database (shots in the range 42815 to 70722 that correspond to years 1997-2007) was utilised in [24] and all types of disruptions were considered. The overall success rate is 92.73% and the false alarm rate is 6.4%.

The second reference [11] is an improved version of [24] that makes use of the same special combination of SVM classifiers but only seven signals are utilized. The signals are represented in two different domains: time and frequency. This advanced implementation is known as APODIS, (as presented in the introduction), and it is working in the JET real-time data network with very good results. A very important novelty of APODIS, compared to other predictors, is its prediction stability. APODIS was trained with JET data corresponding to the C wall experimental campaigns C19-C22 (years 2007-2008) and this training was carried out with 4070 non-disruptive discharges and 246 unintentional disruptions. All disruptions in the above JET campaigns were taken into account regardless of types (mode lock, neo-classical tearing mode, vertical displacement event, Greenwald limit and so on, [23]) and the part of the discharge where the disruptions are produced (current ramp-up, plateau or ramp-down). The only exception to the selection of specific discharges was to exclude intentional disruptions. Fig. 3 shows the APODIS tests with JET C wall data of campaigns C23-C27 (years 2008-2009) and table 1 presents the results of APODIS during the three first ILW campaigns of JET (years 2011-2012). It should be emphasized that no retraining at all has been performed, but the rates remain stable. It is important to mention again that the *warning time* of APODIS during the JET ILW campaigns has been 426 ms.

Continuing with the above references about predictors, [27] describes an adaptive neural predictor for AUG. The training includes 100 disruptive discharges between years 2002 and 2004. The predictor is tested with discharges of later experimental campaigns (2004-2007). All disruptions that occurred in the chosen experimental campaigns were included except the ones occurring in the ramp-up phase, in the ramp-down phase (if the disruption does not happen in the first 100 ms), those caused by massive gas injection and disruptions following vertical displacement events. Whenever the predictor misses an alarm, the corresponding discharge is added to the training set and the predictor is retrained. The prediction success rate on disruptive pulses exceeds 80% and the false alarm rate is 2.6%.

Finally, reference [28] shows a self-organizing map for AUG which determines the 'novelty' of the input to a Multi-Layer Perceptron predictor module. A retraining process is executed in an incremental way, using data from both novelty detections and wrong predictions. The predictor was trained with discharges in the period 2005-2007 and it was applied to predict the behaviour in discharges between 2007 and 2009. The success rate on disruptive shots is 79.69% and the rate of false alarms is 3.36%.

All the above predictors use as much information as possible from past discharges to recognise in advance an incoming disruption. The main objective is to detect the disruptive event although the recognition mechanism does not rely on physics. However, it is important to mention two

recent works based on data-driven approaches with large amounts of past shots that are founded on empirical physics results.

In the first one, [31], a set of simple predictive criteria, each optimized to predict up to four types of AUG disruptions (vertical instability, edge cooling, impurity accumulation and beta limit) are analysed. Disruptions occurred in the period 2005-2009 were divided into the four types according to the last physical mechanisms leading to them and particular attention was paid to the dominant precursors. Only those disruptions which took place in the flat-top phase or within the first 100 ms of the plasma ramp-down phase (whenever both the plasma current is above 0.7 MA and the plasma elongation is greater than 1.5) were considered. Discriminant analysis techniques were used to select the most significant variables and to derive the discriminant function. A rigorous performance analysis of the four types of disruptions can be found in [31].

The second work, [32], describes the prediction of disruptions based on diagnostic data of the high-β spherical torus NSTX. To this end, the disruptive threshold values of many signals are examined. The paper explains the combination of multiple thresholds to produce a total failure rate of 6.5% when applied to a database of about 2000 disruptions during the plasma current flat top.

4. REVIEW OF PREDICTORS FROM SCRATCH

All the methodologies about disruption prediction that have been reviewed in section 3 absolutely depend on a large database of past discharges to create models able to predict disruptions. However, as it was mentioned in the introduction, tokamaks such as ITER and DEMO can suffer irreversible damage as a consequence of disruptions and, therefore, disruption prediction from scratch will be an essential part of their operation.

Recently, two different works have dealt with the development of adaptive data-driven predictors from scratch to learn from the data of an individual machine. Both predictors are not based on plasma physics behaviours, but on discovering relations between signals to identify a close disruption.

The first work is related to APODIS. During the three first ILW campaigns, APODIS has proven to achieve high success rates, low rates of false alarms, early enough disruption prediction and no 'ageing' effect (fig. 3 and table 1). The word 'ageing' refers to the deterioration of a predictor derived from operating the device in different operational regions from those used for the training [24]. Due to these good results, APODIS has been used to predict from scratch and to estimate the minimum number of disruptions to have a reliable predictor [33]. 1237 JET discharges (1036 non-disruptive and 201 disruptive) corresponding to the JET ILW campaigns have been used in chronological order. The first predictor is created after the first disruption and retrainings are carried out after each missed alarm. The main result is that APODIS reproduces its good prediction capabilities and low rate of false alarms after including in the training process about 40 disruptions [33]. With this set-up, the success rate is 93.5% and the rate of false alarms is 2.3%. However, in spite of this important reduction in the training requirements (to be compared with 246 disruptions to train APODIS in JET), the number of 40 disruptive discharges could be too large for the ITER

needs and higher learning rate classifiers from scratch are necessary.

A second predictor from scratch [34] makes reference to fault detection and isolation (FDI) techniques [35]. In [34], the disruption prediction is formalised as a fault detection problem, where the pulses which are correctly terminated (non-disruptive pulses) are assumed as the normal operation conditions and the disruptions are assumed as status of fault. The normal operation conditions model was built with safe pulses from AUG and the dynamic structure of the data was estimated through the fitting of a multivariate ARX (AutoRegressive with eXogenous inputs) model. The datasets are composed of time series of the radiated fraction of the total input power, the internal inductance and the poloidal beta. The disruption prediction system is based on the analysis of residuals in the multidimensional space of the selected variables. The discrepancy between the outputs provided by the ARX model and the actual measurements is an indication of process fault (disruption). The predictor was applied to AUG data between 2002 and 2009. Results are promising but lower false alarm rates are needed. It should be noted that the methodology is not applied to conditions '*from scratch*' but the technique is susceptible of such a development. However, thinking of ITER, the authors state that, perhaps, the method cannot be applied during the very first pulses of ITER due to the need of a sufficient number of pulses safely landed.

5. GENERAL REQUIREMENTS OF A HIGH LEARNING RATE PREDICTOR FROM SCRATCH

As mentioned previously, disruption predictors can be essential systems in ITER and DEMO. Any disruption predictor in these devices has to be ready to work from the beginning of the operations and has to be able to recognise an incoming disruption regardless of both the root cause and discharge phases. As it was mentioned, nowadays, plasma theoretical models are not completely satisfactory for the prediction and, therefore, data-driven models are the only viable option.

It is important to remark again that a disruption predictor is a pre-requisite for any mitigation system. This fundamental characteristic determines the list of general requirements to be met for any data-driven disruption predictor from scratch (DPFS) that learns from the own experimental data of a fusion device. At least, the following operational requirements should be taken into account: learning from scratch, real-time operation, high success rate, high learning rate, early recognition of disruptions, low rate of false alarms, controlled '*ageing*' effect, predictor simplicity, fast training process and reliable predictions.

R1. *Learning from scratch*: data-driven models require a training set of samples (the greater the better) with well-known labels to distinguish among different classes. However, in ITER and DEMO, there will be a complete absence of previous experimental data and the training has to be accomplished in an adaptive way as the discharges are produced. This lack of previous information is not only limited to new fusion devices but it also happens in existing devices when significant changes are implemented (for example, in JET after the installation of the metallic wall).

R2. *Real-time operation*: *real-time* means a deterministic response to meet the time needs of the fusion device. In general, the induction step of fig. 2 can be very expensive in terms of computational resources (computers and CPU time) and it is carried out on an off-line basis. However, the *real-time operation* requirement is related to the deduction step of fig. 2.

R3. *High success rate*: This is equivalent to demand a low rate of missed alarms. This is a consequence of the negative implications that a missed alarm can have in a device. ITER needs a success rate ≥ 95%.

R4. *High learning rate*: The previous requirement is not enough for a DPFS that has to be used from the beginning of the device operation. The development of predictors with low learning rates is of low practical interest. The requirement R3 has to be achieved in the least possible time and this means that the predictor has to demonstrate a *high learning rate*.

R5. *Early recognition of disruptions*: This requirement is essential in the perspective of mitigation actions. In any disruption mitigation technique, a delay between the predictor alarm and the start of the effective mitigation action (*reaction time*) is inevitable. One of the objectives of a mitigation system is to minimise this time and it is clear that the *reaction time* has to be lesser than the *warning time* (fig. 4).

R6. *Low rate of false alarms*: obviously, it is always possible to design an extremely sensible predictor that triggers an alarm when the minimum sign of disruptive behaviour appears. Such level of sensitivity can result incompatible with standard operation of the devices, because all discharges could end in a premature way rendering the operation very tedious and possibly preventing the achievement of high performance. Therefore, a trade-off is necessary between the disruption risk and the operation interruption in such a way that the false alarms are minimised.
It is important to note that in an on-line implementation of a real-time disruption predictor, the concept of false alarm does not make sense. When a predictor triggers an alarm, the tokamak control system has to activate mitigation actions and it is not possible to know whether or not the alarm was false. In this article, the JET database of all discharges corresponding to the past ILW campaigns C28-C30 are analysed and, therefore, information about false alarms can be obtained.

R7. *Controlled 'ageing' effect*: the term '*ageing*' was defined in section 4. However, the '*ageing*' effect in the context of a high learning rate predictor from scratch means that the predictor has to be an adaptive predictor from the beginning. A continuous re-training strategy is fundamental to maintain an updated system able to properly work at any time.

R8. *Simplicity*: this means to develop the simplest possible classifier to distinguish between disruptive and non-disruptive behaviours at any moment. To this end, the disruption

configuration space will have to be characterized with a reduced number of features compatible with the best possible generalization capability.

R9. *Fast training process*: due to the fact that an adaptive classifier is needed to continuously incorporate new relevant information, the training process should be fast enough (from a computational point of view) to allow inter-shot trainings when necessary.

R10. Reliable predictions: any training process with low number of samples is an issue. For this reason, each individual prediction should be qualified with estimation about its reliability.

In general, the level of suitability of a candidate as predictor from scratch depends on its compliance with the previous requirements. But together with the above operational requirements, certain training specificities have to be considered in the development of general high learning rate predictors from scratch. These specificities are related to the inductive step of fig. 2 and can be seen as constraints to take into account in the training process: chronological order of training data, need of general predictors regardless of disruption types, independency of plasma scenarios and, finally, predictor applicability from plasma breakdown to extinction.

Chronological order of training data: in predictors from scratch, the available information for training purposes is very limited and it depends crucially on the chronological order of the discharges. This means that the experimental program of a new device has to envisage a discharge production plan that takes into account the necessary continuous learning.

Need of general predictors regardless of disruption types: the main objective of the training process is to obtain the most general possible predictor that is independent of disruption root causes but is valid for all of them.

Independency of plasma scenarios: the training set has to incorporate details about any type of discharges thereby avoiding a specialised predictor on specific plasma currents, magnetic fields, densities, temperatures and so on.

Predictor applicability from plasma breakdown to extinction: the successive predictors have to work during the total duration of a discharge and, therefore, the training with past discharges has to cover not only plateau regimes but also ramp-up, ramp-down and plasma transient regimes.

6. VENN PREDICTION FRAMEWORK

This article describes the design and development of a particular disruption predictor from scratch. Its initial objective is to provide a trustworthy disruption predictor in situations with absolute lack of previous information. The aim is to learn in a continuous way and to start the predictions as soon as possible. In this case, the first predictor will include features of both disruptive and non-disruptive discharges. This means that the first predictor can be developed after having at least information about one disruptive discharge and one non-disruptive discharge.

All general requirements (R1-R10) have been taken into account to create the predictor. However,

requirement-R10 (*reliable predictions*) has had a major impact in the search of mathematical approaches to qualify each prediction with a measure of reliability.

Probabilistic predictors are particularly suited to these purposes. However, taking into account that the first predictions will be based on a very reduced number of training examples, it is important to note that just a probability could be misleading. It is necessary to determine a probability interval for each prediction. The probability interval will give a clear idea about the efficiency of the prediction. The smaller the prediction interval the more efficient is the prediction. For example, a classification with a prediction interval [0, 1] is completely uninformative and a prediction interval [0.61, 0.62] is better than a prediction interval [0.6, 0.7].

This section is split in 2 subsections. The first one is devoted to briefly reviewing potential probabilistic classifiers. The second one provides a general description of the probabilistic classifier chosen for the particular DPFS implemented here. This implementation is valid for ITER and DEMO and has been tested with JET discharges after the installation of the metallic wall.

6.1. REVIEW OF POTENTIAL PROBABILISTIC CLASSIFIERS

Several probabilistic classifiers can be proposed. A classifier based on the Bayes decision theory has been considered. In this framework, given a classification task of 2 classes, w_1 and w_2, (disruptive and non-disruptive) and a sample to classify ($\mathbf{x}$, y)with known feature vector $\mathbf{x}$ but unknown label y, the Bayes rule states

$$P(\omega_k \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \omega_k) P(\omega_k)}{p(\mathbf{x})}, k = 1,2$$

This Bayes formula gives the *posterior probability* that the unknown pattern belongs to the respective class w_k, given that the corresponding feature vector takes the value $\mathbf{x}$. The probability distribution function (pdf) p ($\mathbf{x}$| w_k) is the likelihood function of w_k with respect to $\mathbf{x}$. P (w_k) is the *a priori probability* of class w_k and p ($\mathbf{x}$)is the pdf of $\mathbf{x}$ given by

$$p(\mathbf{x}) = \sum_{i=1}^{2} p(\mathbf{x} \mid \omega_i) P(\omega_i)$$

In order to apply the Bayes rule, the likelihood and the prior probability of each class must be known. However, in the particular case of disruption predictors from scratch, the estimation of the likelihood is an issue. The main problem resides in the fact that there are no evidences of parametric forms for the likelihood. In these circumstances, to avoid unjustified assumptions about the form of the pdf, non-parametric approaches are used. Among the non-parametric probability density estimators, the Parzen window method [36] is the most popular. It is based on kernel methods but it needs a minimum number of samples to produce reliable estimations. Due to the fact that the first predictors will not have available enough number of samples, Bayesian classifiers have been discarded.

Other probabilistic classifiers such as SVM binning, SVM isotonic regression or the Platt's

method are known that do not always provide well calibrated outputs (in other words, do not provide a frequentist justification) [37]. Therefore, they have not been considered.

Finally, Venn predictors are machine learning algorithms that provide a probability prediction interval for each prediction and they also provide well calibrated outputs under the *i.i.d.* assumption [13]. The well calibrated probabilistic outputs mean that the accuracy of the Venn predictors is expected to fall within the lower and upper probability intervals and, therefore, the accuracy rates of the predictions do not drop below the minimum probabilities given by the Venn predictors (up to statistical fluctuations). This kind of predictors has been the choice for the DPFS.

6.2. FORMULATION OF A VENN PREDICTOR

Venn predictors belong to the family of conformal predictors [38, 39]. Conformal predictors, unlike other state-of-the art machine learning methods, provide information about their own accuracy and reliability. Conformal predictors do not follow the inductive/deductive steps of fig. 2. Instead of using a batch of old samples (training set) to produce a prediction rule (model) that is applied to new samples, conformal predictions use an alternative framework that makes predictions sequentially, basing each new prediction on all the previous samples. In this way, to make a prediction with a new sample, a shortcut is taken in fig. 2 and the classification process moves from the training samples directly to the prediction (fig. 5). From a mathematical point of view, conformal predictors are always valid. This means that the long-run frequency of errors at confidence level $1 - \varepsilon$ is ε or less. Focusing the attention on the formulation of a Venn predictor [13], the starting point is a training set whose only assumption about the examples is that they satisfy the *i.i.d.* hypothesis. Let $\{z_1,...,z_{n-1}\}$ be the training set, where each $z_i \in Z = X \times Y$ is a pair $(\mathbf{x}_i, y_i)$ consisting of the sample or feature vector $\mathbf{x}_i$ and its class y_i. Given a new feature vector $\mathbf{x}_n$, the objective of a Venn predictor is to estimate the probability of belonging to one class Y_j from a set of J classes ($Y_j \in \{Y_1,..., Y_J\}$).

The Venn prediction framework assigns each one of the possible classifications Y_j to $\mathbf{x}_n$ and divides all examples $\{(\mathbf{x}_1, y_1),..., (\mathbf{x}_{n-1}, y_{n-1}), (\mathbf{x}_1, Y_J)\}$ into a number of categories. To carry out this division, Venn machines use a taxonomy function A_n, $n \in \mathbb{N}$, which classifies the relation between a sample and the set of the other samples:

$$\tau_i = A_n\left(\left\{\left(\mathbf{x}_1, y_1\right),...,\left(\mathbf{x}_{i-1}, y_{i-1}\right),\left(\mathbf{x}_{i+1}, y_{i+1}\right),...,\left(\mathbf{x}_n, y_n\right)\right\},\left(\mathbf{x}_i, y_i\right)\right). \quad (1)$$

Values τ_i are called categories and are taken from a finite set $T = \{\tau_1, \tau_2,,..., \tau_T\}$. Equivalently, a taxonomy function assigns to each sample $(\mathbf{x}_i, y_i)$ its category τ_i, or in other words, it groups all examples into a finite set of categories. This grouping should not depend on the order of examples within a sequence.

For the reader convenience, it is important to mention several types of taxonomies that can be used with multi-class problems in the Venn prediction framework. Five different taxonomies are analyzed in [40], where the Venn predictors are based on neural networks. Lambrou et al. [37]

shows an application of inductive conformal predictors to develop an inductive Venn predictor with a taxonomy derived from a multi-class Support Vector Machine classifier. Nouretdinov et al. [41] describes the logistic taxonomy, which is created from the probabilistic method of logistic regression.

In this article, as it is explained in section 7, the selected taxonomy has been the nearest centroid taxonomy (NCT) [42]. Due to the fact that there are two classes for the disruption predictor ($Y = \{Y_{Disruptive}, Y_{Non\text{-}disruptive}\}$), the number of categories in the NCT Venn taxonomy is also 2 that correspond to the cases '*disruptive*' and '*non-disruptive*'. This means that the NCT sets the category τ_i (eq.1) o.f a sample z_i equal to the label of its nearest centroid (fig. 6). The mathematical formulation of the NCT is

$$\tau_i = A_n\left(\left\{\left(\mathbf{x}_1, y_1\right),...,\left(\mathbf{x}_{i-1}, y_{i-1}\right),\left(\mathbf{x}_{i+1}, y_{i+1}\right),...,\left(\mathbf{x}_n, y_n\right)\right\},\left(\mathbf{x}_i, y_i\right)\right) = Y_j \tag{2}$$

where

$$j = \underset{j=Disruptive, Non-\ disruptive}{\arg\min} \left\|\mathbf{x}_i - \mathbf{C}_j\right\|,$$

$\boldsymbol{C}_j$ are the centroids of the two classes and ||.|| is a distance metric. In this article, the Euclidean distance has been chosen.

Proceeding with the formulation of the Venn predictors, after having chosen a taxonomy with T categories, the classification process of a new sample $\mathbf{x}_n$ with unknown label y_n and J possible labels $(Y_1,..., Y_1\}$ is a (J+1)-step procedure. The first step is to assume that $y_n = Y_1$ and partition the n samples $\{(\mathbf{x}_i, y_1),...,(\mathbf{x}_{n-1}, y_{n-1}), (\mathbf{x}_n, Y_1)\}$ into categories using the taxonomy that was previously chosen. The empirical probability distribution of the labels in the category τ that contains $(\mathbf{x}_n, Y_1)$ will be

$$p^{Y_1}\left(Y_k\right) = \frac{\left|\left\{\left(\mathbf{x}^*, y^*\right) \in \tau : y^* = Y_k\right\}\right|}{|\tau|}, k = 1,..., J.$$

This is a probability distribution for the class of $\mathbf{x}_n$. In other words, after determining the category τ which $\mathbf{x}_n$ is located, the probabilities of each label within the category τ are computed. Therefore, a row vector $(p^{Y1}(Y_1),..., p^{Y1}(Y_J))$ with J components (one per label) is obtained.

The second step is to assume that $y_n = Y_2$ and partition the n samples $\{(\mathbf{x}_i, y_1),...,(\mathbf{x}_{n-1}, y_{n-1}), (\mathbf{x}_n, Y_2)\}$in the same way as in the first step to obtain the row vector $(p^{Y2}(Y_1),..., p^{Y2}(Y_J))$. These steps are carried out J times to obtain all empirical probability distributions.

After having assigned all possible labels to $\mathbf{x}_n$, the above row vectors obtained between steps 1 and J form a square matrix of dimension J:

$$P_j = \left(\begin{array}{ccc} p^{Y_1}(Y_1) & p^{Y_1}(Y_2) & p^{Y_1}(Y_J) \\ p^{Y_2}(Y_1) & p^{Y_2}(Y_2) & p^{Y_2}(Y_J) \\ \hdashline p^{Y_J}(Y_1) & p^{Y_J}(Y_2) & p^{Y_J}(Y_J) \end{array} \right)$$

The last step in the prediction process (step J+1) is to assign a label to the sample $\mathbf{x}_n$ to classify. The Venn predictor outputs the prediction $\hat{y}_n = Y_{Jbest}$, where

$$j_{best} = \arg\max_{k=1,...,J} \overline{p(k)}$$

and $\overline{p(k)}$, $k = 1,..., J$ is the mean of the probabilities obtained for label Y_j, $j = 1,..., J$ among all probability distributions (*i.e.* the mean of every column of matrix P_J). In other words, the label of the column with the maximum mean value is the prediction. The probability interval for the prediction $\hat{y}_n = Y_{Jbest}$ is $[L(Y_{Jbest}), U(Y_{Jbest})]$, where $U(Y_{Jbest})$ and $L(Y_{Jbest})$ are respectively the maximum and minimum probability of the column with the maximum mean value.

7. Implementation details of the disruption predictor from scratch

This section describes the rationale of the DPFS implemented in this article. As stated in section 6, the requirement about reliable predictions (R10) has been the starting point in the search of a predictor from scratch. The use of Venn predictors is justified due to three main facts. Firstly, no additional assumptions to the typical machine learning *i.i.d.* hypothesis about the samples are necessary. Secondly, Venn predictors provide a probability interval for each prediction instead of a single probability. This probability interval can be seen as a probability error bar. Thirdly, the predictions are well calibrated, which means that the accuracy falls between the lower and upper probability interval.

Due to the use of the Venn prediction framework, it is important to clarify the terminology before describing the implementation details. In particular, the term 'training set' has to be detailed. In machine learning, a 'training set' is typically a dataset made up of examples with well-known labels. In the case of non-conformal predictors, the 'training set' is used to obtain a model (fig.2) and from that moment, the training set can be ignored because only the model is necessary to make predictions. However, with Venn predictors, there is not a model and, therefore, the 'training set' cannot be ignored. The 'training set' is always used together with each new sample to classify (fig.5). Due to this reason, in the following, the term 'model' is no longer used. Instead, 'training set' is utilized.

Focusing the attention on the Venn predictor of this paper, it should be noted that the prediction of the disruptive/non-disruptive behaviour of a discharge has to be carried out at regular time intervals during the execution of the discharge. In each time interval, a feature vector is generated with several plasma quantities and this feature vector is used together with the 'training set' to make the prediction. The 'training set' is the dataset that is generated after each re-training. Therefore, it is important to establish that in the case of a DPFS based on a Venn predictor, the term 're-training' means to create a new 'training set'.

The following subsections are devoted to discussing the specific implementation details of the Venn disruption predictor from scratch. Once an explicit selection of a probabilistic predictor has been performed under requirement R10, subsection 7.1 discusses the particular aspects of requirements R1-R9. Subsections 7.2, 7.3 and 7.4 describe the precise choices to optimize the issues related to the real-time response of the DPFS. Finally, subsection 7.5 presents the implementation of the predictor from scratch developed in this article.

7.1 ANALYSIS OF REQUIREMENTS R1-R9

To deal with the issue of learning from scratch (requirement R1), an important consideration about the training examples is needed. In machine learning, it is very difficult in most cases to achieve good results without presenting examples of all the classes. This means that the first training process of the DPFS requires at least one disruptive example and one non-disruptive example. It is important to emphasise that the term 'example' is not necessarily equivalent to 'discharge', *i.e.* one discharge could provide lots of examples and, conversely, one example could summarize data corresponding to several discharges.

Concerning the real-time response of the DPFS (requirement R2), a Venn predictor has been chosen. However, it is well-known that the major issue of any conformal predictor is usually the computation time. To overcome this crucial difficulty, the selection of, first, a reduced number of features, second, the concentration of knowledge about disruptive and non-disruptive behaviours and, third, the choice of a proper Venn prediction taxonomy are essential points. These three aspects are discussed in sections 7.2, 7.3 and 7.4 respectively.

The ageing effect of a classifier from scratch (requirement R7) has to be tackled in the perspective of defining a re-training strategy to maintain an updated system. In the present implementation of the DPFS, the policy has been to only re-train after a missed alarm. This criterion is the closest one to a real on-line operation in which it is not possible to determine whether or not an alarm is true. In this paper, the DPFS is applied to a database of JET discharges and, therefore, the predictor performances are assessed after the analysis of the predictions.

Requirements R8 and R9 are discussed together due to their close relation in the DPFS implementation. First of all, it is important to remember that a Venn predictor does not use the deductive/inductive steps of fig. 2. To make predictions, Venn predictors jointly use all previous samples and the new sample to classify. Consequently, depending on both the total number of samples and the feature vectors dimensionality, the Venn predictor framework can be computationally very expensive and, therefore, the computational needs can be an issue under real-time requisites. This implies to handle the least possible number of samples with a reduced dimensionality. All aspects related to this are treated in sections 7.2, 7.3 and 7.4.

The requirements R3-R6 are final objectives and they cannot be used to impose "a priori" implementation constraints. Results about the success rate (R3), number of disruptions to have a reliable classifier (R4), average warning time (R5) and false alarm rate (R6) can only be assessed

after analysing a particular implementation of a DPFS. If the predictor evaluation is not good enough in terms of requirements R3-R6, the specific choices performed in requirements R1, R2, R7-R9 will have to be reviewed.

7.2 USE OF A REDUCED NUMBER OF FEATURES IN THE DPFS

To satisfy the requirement of simplicity (R8), the predictor architecture consists of just one single classifier instead of using a multi-layer architecture as APODIS. This has been decided to avoid the extra computation of several classifiers plus their combination into a single output.

The starting point has been to select a reduced number of signals to identify disruptive and non-disruptive behaviours. Due to the characteristic of '*starting from scratch*', only those quantities that are considered essential diagnostics from the start of a fusion device have been considered. In particular, the same signals that are used by APODIS in JET (table 2) have been chosen.

After this decision, it is necessary to take advantage of some knowledge derived from the use of APODIS. APODIS has shown that mean values (in the time domain) and standard deviations (in the frequency domain) corresponding to temporal windows of 32 ms provide a powerful parameterization of the disruption configuration space. Therefore, the new predictor will also use 32 ms long temporal windows, (with sampling periods of 1 ms for all the signals), as basic time segments inside which the mean values (time domain) and standard deviations after removing the DC component (frequency domain) are computed to be used as features. So, the discharge behaviour (disruptive/non-disruptive) will be evaluated every 32 ms from the plasma start to the plasma extinction with the same APODIS additional condition that the plasma current is above 750 kA.
However, in spite of this low dimensional parameterization of the disruption configuration space (14 features), the real-time computation requirements (the smaller amount of features the better) motivate an even further reduction of dimensionality. The objective is to find out which subset of the 14 features can be used. Therefore, it is necessary to perform a combinatorial analysis to determine the minimum number of features that provide good enough prediction results. All possible combinations with a number of features between 2 and 7 have been tested. This means that 9893 different predictors have to be analysed:

$$\sum_{n=2}^{7} C_{14,n} = 9893, \quad \text{where} \quad C_{m,n} = \frac{m!}{n!(m-n)!}$$

7.3. SELECTION OF TRAINING FEATURE VECTORS TO CONDENSE THE KNOWLEDGE ABOUT DISRUPTIVE/NON-DISRUPTIVE BEHAVIOURS

The previous subsection has determined that the potential disruption predictors will have a number of features between 2 and 7. In fact, this number of features will be computed every 32 ms to predict the plasma behaviour. However, it is necessary to define the feature vectors that form the initial and successive 'training sets' after each re-training (as explained at the beginning of section 7, it should be reminded that 're-train' means the creation of a new 'training set').

In the discussion of requirement R1, it was established the fact that both disruptive and non-disruptive samples are necessary in the training sets. But it is important to realize that, in the normal operating scenario of a tokamak, a high unbalance between disruptive and non-disruptive discharges is an intrinsic property. For example, over the last years of JET operation with C plasma facing components, the unintentional disruption rate was 3.4% and with the ILW campaigns the unintentional disruption rate is slightly above 10% [43]. But in terms of machine learning, the unbalance is not only related to the number of disruptive and non-disruptive discharges. The discharges are evaluated on a periodic basis (for example, tens of ms) and, therefore, a non-disruptive discharge in JET can contain thousands of non-disruptive samples. On the contrary, only the last samples of a disruptive discharge (let's say 1 s before the disruption) can be considered disruptive. This unbalance between disruptive and non-disruptive samples was analysed in [33] for predictors from scratch and the conclusion was to use balanced training sets. Therefore, the DPFS will be using a balanced approach of disruptive and non-disruptive examples in the training sets.
Given a number of features between 2 and 7, the first training set will be made up of just 1 disruptive feature vector and 1 non-disruptive feature vector. Fig. 7 shows the set of feature vectors 32 ms long corresponding to the first disruptive discharge. They have been ordered backwards from the disruption time t_D. The right column gives the time intervals of the temporal windows. The feature vector that characterizes the first disruptive feature vector in the first 'training set' corresponds to the time window previous to the disruption (gray background in fig. 7). This selection is based on the fact that all signals show the most clear morphological patterns of an incoming disruption just in the time segment closest to the disruption.

The selection of the first non-disruptive feature vector for the first 'training set' is not as straightforward as the disruptive one. It is clear that there are much more non-disruptive discharges than disruptive ones (and, of course, much more non-disruptive feature vectors than disruptive ones). The objective is to use a feature vector able to condense (in some sense) the non-disruptive character just into a single vector. A random choice of a non-disruptive feature vector does not guarantee the selection of a feature vector that represents an '*average behaviour*' of non-disruptive behaviours. The '*average behaviour*' feature vector has been defined as the mean value of the components of all previous feature vectors of non-disruptive discharges (fig. 8).

After defining the first 'training set' once the first disruption has occurred, it is possible to make predictions. According to the retraining strategy, this 'training set' can be used until the first missed alarm. When this happens, new knowledge has to be incorporated to the predictor. Taking into account the objective of using a balanced system, the new training set will include, in addition to the previous training feature vectors, knowledge about both the missed alarm and the non-disruptive discharges from the first 'training set'. The information to include relatively to the missed alarm is the last feature vector previous to the missed disruption (see fig. 7). With regard to the new feature vector corresponding to non-disruptive behaviours, an '*average behaviour*' feature vector, whose components are the mean value of the components of all non-disruptive discharges from the

preceding training up to the last safe discharge previous to the missed disruption, is added. This criterion ensures balanced training sets between disruptive and non-disruptive samples.

However, it is necessary to note that this balance can be altered under a specific condition. If two or more consecutive discharges are disruptive and the predictor fails in the recognition, there are not safe discharges between the contiguous disruptive ones and, therefore, only the feature vectors of each disruptive discharge previous to the disruptions will be included.

7.4 VENN PREDICTOR TAXONOMY

The next aspect to consider is the choice of the Venn taxonomy for the DPFS. The disruptive/non-disruptive behaviours ideally should be condensed in two regions of the parameter space that are well separated between them. This condition looks for achieving enough generalization capability with the predictor, independently of the number of feature vectors in the training set. To concentrate as much as possible the information of both behaviours in just two points, the Venn predictor uses the nearest centroid taxonomy that was described in section 6.2.

7.5 DPFS ALGORITHM

In relation with the DPFS, it is necessary to emphasize the importance of making predictions as soon as possible, *i.e.* with the minimum number of examples. Taking into account the decision of using balanced 'training sets', ideally, the first predictor should be put into operation after the first disruption (typically, when a tokamak starts the operation, a number of k_1 non-disruptive discharges are produced before the first disruptive one). These k_1+1 discharges will be used to generate the initial training set. The Venn predictor is used for the first time with the discharge k_1+2. During this discharge, feature vectors to assess the plasma behaviour are calculated every 32 ms and each feature vector, together with the initial 'training set', is used to make predictions. The initial 'training set', with both disruptive and non-disruptive discharges, remains valid until it misses an alarm. At this moment, it is necessary to re-train the predictor, which means to generate a new 'training set'. Algorithm 1 represents in pseudo-code the DPFS working scheme from the beginning of the tokamak operation.

Algorithm 1: Pseudo-code of the DPFS predictor. The Venn prediction output depends on both the 'training set' and the 'feature vector' to classify at each time instant. When a new 'training set' is set-up, it includes all the previous samples plus knowledge about the last missed alarm and the safe discharges after the previous training.

Start of tokamak operation
K1 non-disruptive discharges
First disruptive discharge
First 'training set'

Prediction loop

Start of discharge

While Ip > 750 kA and every 32 ms

Form ‘feature vector’

prediction = Venn predictor output

- ‘training set’
- ‘feature vector’

If prediction == ‘disruptive’

break

End of **If prediction**

End of **While Ip**

End of discharge

If prediction == ‘non-disruptive’ & it is false

Missed alarm

New ‘training set’

- Previous training set
- Knowledge about
  - the missed alarm
  - safe discharges from the last training

End of **If prediction**

End of **Prediction loop**

Evaluation of success rate and false alarm rate

8. RESULTS

Algotithm 1 has been implemented and it has been applied to a database of 1237 JET discharges (1036 safe discharges and 201 unintentional disruptions) corresponding to the three first JET experimental campaigns after the installation of the metallic wall. The first predictor is obtained after the first disruption and, from that moment, all discharges are analysed in chronological order. Each discharge is analysed by simulating a real-time data processing. Feature vectors are created every 32 ms and they are classified as disruptive or non-disruptive. After a missed alarm, a new ‘training set’ is created to incorporate new knowledge as previously explained.

Table 3 shows the 14 features used with the Venn predictors. All possible combinations between 2 and 7 features have been tested (9893 predictors have been developed). Given a specific combination of features, the algorithm 1 is executed for the whole dataset of discharges. The first predictor is generated after the first disruption and it is used with all posterior discharges, subject to re-train after every missed alarm. The statistics presented in this section correspond to the evaluation of the predictors after the analysis of the 1237 discharges. In other words, the success and false alarm rates are the cumulative results of algorithm 1 after 1237 discharges. It is important to note that the rates

are expressed in terms of discharges and not in terms of the number of feature vectors classified. This means that one discharge is classified as safe when all its feature vectors are classified as non-disruptive. Also, one discharge is recognized as disruptive when just 1 feature vector is classified as disruptive. Finally, a single feature vector classified as disruptive in a safe discharge is enough to identify the event as a false alarm.

The first result to emphasise is the achievement of the same success rate, 94%, for almost all predictors. The lower bound of false alarms is 4.21% and the average prediction probability is high (the minimum value of the error bar is 0.604 and the maximum one is always 1). This high prediction probability gives enough confidence about the reliability of the predictors in spite of the low number of training samples that have been used. The use of probabilistic classifiers has been contemplated just to validate the prediction results as long as the prediction probabilities are high enough.

According to the table 5, the best candidates for predictions are the ones with false alarm rates of 4.21%. Among the five candidates, the predictors with lower number of features are preferred (simplicity requirement R8). This reduces the possibilities to only two options. The final selection between them is favourable to the predictor with the smaller probability error bar (the predictor with features 2, 3, 4 and 5) because it means a more accurate prediction.

Figure 9 corresponds to the results of the predictor with features 2, 3, 4 and 5. Fig. 9a shows the evolution of the success rate with the number of disruptions. It is important to observe the fast learning rate achieved in the learning process, as a success rate of 90.91% is reached with a training set made up of only 2 disruptive examples and 2 non-disruptive examples. The squares in the figure show the missed alarms. Fig. 9b represents the evolution of the false alarm rate. The false alarm rate tends to slightly increase with the number of disruptions. Last but not least, fig. 9c gives the evolution of the prediction probability as the system learns. The prediction probability increases and the error bars diminish (which means more accurate predictions) with the number of disruptions.

The comparison of these results with the APODIS version from scratch described in [33] is favourable to the Venn predictors. The Venn predictors do not need tens of disruptions to achieve stable and good enough success and false alarm rates as in the case of APODIS from scratch [23] (fig. 10). One important issue of the APODIS from scratch version is the need of about 40 disruptions to provide good performances. This problem is avoided with the Venn predictors, as it can be seen by comparing the success and false alarm rates of figs. 9a, 9b and 10.

From table 5 it is clear that both the stored diamagnetic energy time derivative and the total input power do not play any role in the prediction from scratch. The most important signal is the locked mode, which is present in both time and frequency domains in practically all feature combinations. The interpretation is clear taking into account the known relation between disruptions and MHD. The density signal also becomes important in both domains as the number of features increase.

The plasma current contributes but only with the information in the frequency domain. Again, the interpretation is straightforward. The plasma current amplitude cannot determine the presence of an incoming disruption but its frequency spectrum (through its standard deviation after removing

the DC component) is a good precursor of a disruption. The use of the power spectrum as disruption precursor was discussed in section 3.

Conversely to the plasma current, the plasma internal inductance to predict from scratch is only significant in the time domain. Finally, it is important to note that the radiated power contribution to the predictions mainly comes from the temporal domain.

Moreover, the warning times obtained for the different predictors should also be carefully evaluated. Table 6 provides the average warning time of the predictors shown in table 5 together with the accumulative fraction of detected disruptions at a time instant 30 ms previous to the disruptions. The comparison of these warning times with the ones obtained by APODIS during the same experimental campaigns, (see section 1), has to be interpreted in the sense that the predictors from scratch are much more sensible to the disruptive behaviour. This also explains the larger value of the false alarm rate. On the other hand, it should be noted that the cumulative fraction of detected disruptions 30 ms before the disruption is lesser than in the case of APODIS.

Figure 11 shows the warning times obtained with the Venn predictor with features 2, 3, 4 and 5. The vertical line at 30 ms shows the minimum time in JET to perform mitigation actions. This figure should be compared to fig. 1 that shows the results with APODIS in JET.

9. DISCUSSION

This article has shown the viability of developing high learning rate disruption predictors from scratch. The prediction probabilities are high enough to guarantee the reliability of the predictors. After applying the algorithm 1 to 1237 JET discharges (1036 safe and 201 disruptive) and using only four features from three different signals (plasma current, mode lock and plasma internal inductance), the results of predicting from scratch give a success rate of 94%, a false alarm rate of 4.21% and a warning time of 654 ms. Table 7 summarizes the generation of training sets after the missed alarms. It is important to note that the first predictor (grey background in the table) is generated after the first disruption.

The results obtained with the Venn predictor from scratch are absolutely comparable to the ones obtained with APODIS. However, there exists an important difference. APODIS was trained with 246 disruptive discharges and 4070 non-disruptive discharges. Taking into account that 3 feature vectors per disruptive discharge were used with label 'Disruptive', a number of 738 disruptive samples were necessary to train APODIS. In addition, about 500 feature vectors per safe discharge were used with label 'Non-disruptive', which gives 2035000 safe samples in total. For a fair comparison with the present Venn predictors, the number of safe and disruptive training samples, 2035000 safe and 738 respectively, has to be compared with the samples used in the adaptive Venn predictor. The first Venn predictor uses just 1 disruptive feature vector and 1 non-disruptive feature vector and a very limited number of feature vectors is included during the posterior re-trainings after the missed alarms. According to table 7, only 12 retrainings were needed and the final 'training set' after 12 missed alarms consists of just 13 disruptive feature vectors and 13 non-disruptive feature vectors.

The good results obtained with so few training feature vectors have to be interpreted as a combination of several factors. Firstly, the feature selection (mean values and standard deviation of the power spectrum after removing the DC component) is good enough to describe disruptive and non-disruptive behaviours. This fact is not new and it has been used from the first versions of APODIS. Secondly, the particular choice of feature vectors should be mentioned. The feature vector previous to a missed alarm as disruptive example, together with the selection of an average vector to summarize a non-disruptive behaviour, allows condensing the disruptive/non-disruptive information in a convenient way. Thirdly, the use of a nearest centroid taxonomy has provided a high generalization capability to distinguish between both types of plasma behaviours.

It should also be mentioned that the union of the Venn predictors with APODIS from scratch is a valid possibility to predict without previous information. Venn predictors can be the first predictors up to about 50 disruptions. Fig. 9b shows a false alarm rate of 2% for these predictors. From this moment, APODIS from scratch can be used because with 50 disruptions there are enough information to provide high success rates and low false alarm rates (fig. 10).

The Venn predictors described in this article have been the first adaptive predictors that have been trained from scratch and with very good results in terms of success rate, false alarm rate and warning time. The several predictors developed with different number of features satisfy all operational requirements discussed in section 5.

The prediction from scratch is an open research line in nuclear fusion in the perspective of ITER/DEMO and important efforts should be devoted to them. As a first suggestion in the line of this article, it should be mentioned the need of improving the predictions to reduce the false alarm rates. As a second point, it is important to note that ITER is going to have different time-scales to JET and, therefore, an analysis about the influence of sampling times in the warning times is necessary and, perhaps, it could show a significant effect. Finally, another very important aspect is the validation of the approach with different isotopic compositions of the main plasma. Indeed, it must be considered that ITER operation will include different fuels mixtures, from H and He to full D-T, and the performance of adaptive predictors in these varying conditions should be carefully assessed.

ACKNOWLEDGMENTS

This work was partially funded by the Spanish Ministry of Economy and Competitiveness under the Projects No ENE2012-38970-C04-01 and ENE2012-38970-C04-03. This work, supported by the European Communities under the contract of Association between EURATOM/CIEMAT, was carried out within the framework of the European Fusion Development Agreement. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

REFERENCES

[1]. B. Esposito, G. Granucci, S. Nowak, J.R. Martin-Solis, L. Gabellieri, E. Lazzaro, P. Smeulders, M. Maraschek, G. Pautasso, J. Stober, W. Treutterer, L. Urso, F. Volpe, H. Zohm, FTU, ECRH,

and ASDEX Upgrade Teams. Nuclear Fusion **49** (2009) 065014 (10 pp).

[2]. T. C. Hender, J. C. Wesley, J. Bialek, A. Bondeson, A.H. Boozer, R. J. Buttery, A. Garofalo, T. P. Goodman, R. S. Granetz, Y. Gribov, O. Gruber, M. Gryaznevich, G. Giruzzi, S. Günter, N. Hayashi, P. Helander, C.C. Hegna, D.F. Howell, D.A. Humphreys, G. T. A. Huysmans, A. W. Hyatt, A. Isayama, S. C. Jardin, Y. Kawano, A. Kellman, C. Kessel, H.R. Koslowski, R. J. La Haye, E. Lazzaro, Y. Q. Liu, V. Lukash, J. Manickam, S. Medvedev, V. Mertens, S.V. Mirnov, Y. Nakamura, G. Navratil, M. Okabayashi, T. Ozeki, R. Paccagnella, G. Pautasso, F. Porcelli, V.D. Pustovitov, V. Riccardo, M. Sato, O. Sauter, M.J. Schaffer, M. Shimada, P. Sonato, E.J. Strait, M. Sugihara, M. Takechi, A.D. Turnbull, E. Westerhof, D.G. Whyte, R. Yoshino, H. Zohm and the ITPA MHD, Disruption and Magnetic Control Topical Group. Nuclear Fusion **47** (2007) S128-S202.

[3]. C. Reux, J. Bucalossi, F. Saint-Laurent, C. Gil, P. Moreau and P. Maget. Nuclear Fusion 50 (2010) 095006 (9pp).

[4]. N. Commaux, L.R. Baylor, T.C. Jernigan, E. M. Hollmann, P.B. Parks, D.A. Humphreys, J. C. Wesley and J.H. Yu. Nuclear Fusion **50** (2010) 112001 (4pp).

[5]. M. Bakhtiari, G. Olynyk, R. Granetz, D. G. Whyte, M. L. Reinke, K. Zhurovich and V. Izzo. Nuclear Fusion **51** (2011) 063007 (9pp).

[6]. N. Commaux, L.R. Baylor, S.K. Combs, N.W. Eidietis, T.E. Evans, C.R. Foust, E. M. Hollmann, D.A. Humphreys, V.A. Izzo, A.N. James, T.C. Jernigan, S.J. Meitner, P.B. Parks, J.C.Wesley and J.H. Yu. Nuclear Fusion **51** (2011) 103001 (9pp).

[7]. M. Lehnen, A. Alonso, G. Arnoux, N. Baumgarten, S.A. Bozhenkov, S. Brezinsek, M. Brix, T. Eich, S.N. Gerasimov, A. Huber, S. Jachmich, U. Kruezi, P. D. Morgan, V. V. Plyusnin, C. Reux, V. Riccardo, G. Sergienko, M. F. Stamp and JET EFDA contributors. Nuclear Fusion **51** (2011) 123010 (12pp).

[8]. A.H. Boozer. Physics of Plasmas. **19**, 058101 (2012).

[9]. M. Lehnen, A. Alonso, G. Arnoux, N. Baumgarten, S.A. Bozhenkov, S. Brezinsek, M. Brix, T. Eich, S.N. Gerasimov, A. Huber, S. Jachmich, U. Kruezi, P.D. Morgan, V.V. Plyusnin, C. Reux, V. Riccardo, G. Sergienko, M.F. Stamp and JET EFDA contributors. Nuclear Fusion **51** (2011) 123010 (12pp).

[10]. C. G. Windsor, G. Pautasso, C. Tichmann, R.J. Buttery, T. C. Hender, JET-EFDA Contributors and the ASDEX Upgrade Team. Nuclear Fusion **45** (2005) 337–350.

[11]. 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.

[12]. P. C. de Vries, M. F. Johnson, I. Segui and JET-EFDA Contributors. Nuclear Fusion 49 (2009) 055011 (12 pp).

[13]. V. Vovk, A. Gammerman, G. Shafer. "Algorithmic learning in a random world". Springer (2005).

[14]. C. M. Bishop. “Neural Networks for Pattern Recognition”. Oxford University Press. (2004).
[15]. V. Cherkassky, F. Mulier. “Learning from data“. Second edition. John Wiley & Sons, Inc. (2007).
[16]. R. O. Duda, P. E. Hart, D. G. Stork. ”Pattern classification”. Second edition. Wiley-Interscience. (2001).
[17]. T. Kohonen. “The self-organizing map”. Neurocomputing, **21**:1-6 (1998).
[18]. C. M. Bishop, M. Svensén, and C. K. I. Williams. “The generative topographic mapping”. Neural Computation, **10**:215-234 (1998).
[19]. R. J. Buttery, A. H. Boozer, Y. Q. Liu, J.-K. Park, N. M. Ferraro, V. Amoskov, Y. Gribov, R. J. La Haye, E. Lamzin, J. E. Menard, M. J. Schaffer, E. J. Strait, and the DIII-D Team. Physics of Plasmas **19**, 056111 (2012).
[20]. J.A. Wesson, R.D. Gill, M. Hugon, F.C. Schuller, J.A. Snipes, D.J. Ward, D.V. Bartlett, D.J. Campbell, P.A. Duperrex, A.W. Edwards, R.S. Granetz, N.A.O. Gottardi, T.C. Hender, E. Lazzaro, P.J. Lomas, N. Lopes Cardozo, K.F. Mast, M.F.F. Nave, N.A. Salmon, P. Smeulders, P.R. Thomas, B.J.D. Tubbing, M.F. Turner, A. Weller. Nuclear Fusion **29** (4) 1989, 641-666.
[21]. M.F.F. Nave, A. Wesson. Nuclear Fusion **30** (12) 1990, 2575-2583.
[22]. B. Cannas, A. Fanni, P. Sonato, M. K. Zedda and JET-EFDA Contributors. Nuclear Fusion **47** (2007) 1559-1569.
[23]. P C. de Vries, M.F. Johnson, B. Alper, P. Buratti, T.C. Hender, H.R. Koslowski, V. Riccardo and JET-EFDA Contributors. Nuclear Fusion **51** (2011) 053018 (12 pp).
[24]. G.A. Rattá, J. Vega, A. Murari, G. Vagliasindi, M.F. Johnson, P.C. de Vries and JET-EFDA Contributors. “An Advanced Disruption Predictor for JET tested in a simulated Real Time Environment” Nuclear Fusion **50** (2010) 025005 (10pp).
[25]. B. Cannas, A. Fanni, E. Marongiu, P. Sonato. Nuclear Fusion 44 (2004) 68-76.
[26]. A. Sengupta and P. Ranjan. Forecasting disruptions in the ADITYA tokamak using neural networks Nuclear Fusion **40** (2000) 1993–2008.
[27]. B. Cannas, A. Fanni, G. Pautasso, G. Sias and P. Sonato. Nuclear Fusion **50** (2010) 075004 (12pp).
[28]. B. Cannas, A. Fanni, G. Pautasso, G. Sias and the ASDEX Upgrade Team. Fusion Engineering and Design **86** (2011) 1039–1044.
[29]. A. Murari, G. Vagliasindi, P. Arena, L. Fortuna, O. Barana, M. Johnson and JET-EFDA Contributors. “Prototype of an adaptive disruption predictor for JET based on fuzzy logic and regression trees”. Nuclear Fusion **48** (2008) 035010 (10pp).
[30]. G. Vagliasindi, A. Murari, P. Arena, L. Fortuna, M. Johnson, D. Howell and JET-EFDA Contributors. “A Disruption Predictor Based on Fuzzy Logic Applied to JET Database”. IEEE Transactions on Plasma Science. **36** (1) 2008 (253-262).
[31]. Y. Zhang, G. Pautasso, O. Kardaun, G. Tardini, X.D. Zhang and the ASDEX Upgrade Team. Nucl. Fusion **51** (2011) 063039 (12pp).
[32]. S.P. Gerhardt, D.S. Darrow, R.E. Bell, B.P. LeBlanc, J.E. Menard, D. Mueller, A.L. Roquemore, S.A. Sabbagh and H. Yuh. Nuclear Fusion **53** (2013) 063021 (19pp).

[33] . S. Dormido-Canto, J. Vega, J.M. Ramírez, A. Murari, R. Moreno, J.M. López, A. Pereira and JET-EFDA Contributors. Nuclear Fusion.**53** (2013) 113001 (8pp).

[34]. R. Aledda, B. Cannas, A. Fanni, G. Sias, G. Pautasso and ASDEX Upgrade Team. "Multivariate statistical models for disruption prediction at ASDEX Upgrade". Fusion Engineering and Design **88** (2013) 1297-1301.

[35]. R.J. Patton, P. Frank and R. Clark (1989). "Fault Diagnosis in Dynamic Systems. Theory and applications". Control Engineering Series, Prentice Hall.

[36]. R. O. Duda, P. E. Hart, D. G. Stork. "Pattern classification". Second edition. Wiley-Interscience. (2001).

[37]. A. Lambrou, H. Papadopoulos, I. Nouretdinov, A. Gammerman. "Reliable Probability Estimates Based on Support Vector Machines for Large Multiclass Datasets". AIAI 2012 Workshops, IFIP AICT 382, pp. 182–191, 2012.

[38]. V. Vovk, A. Gammerman, C. Saunders. "Machine learning applications of algorithmic randomness". Proceedings of the Sixteenth International Conference on Machine Learning. (1999) 444-453. San Francisco, CA. Morgan Kaufmann.

[39]. C. Saunders, A. Gammerman, V. Vovk. "Transduction with confidence and credibility". Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence. **2** (1999) 722-726. Morgan Kaufmann.

[40]. H. Papadopoulos. "Reliable Probabilistic Classification with Neural Networks". Neurocomputing **107** (2013) 59-68.

[41]. I. Nouretdinov, D. Devetyarov, B. Burford, S. Camuzeaux, A. Gentry-Maharaj, A. Tiss et al. "Multiprobabilistic Venn Predictors with Logistic Regression". AIAI 2012 Workshops, IFIP AICT 382, pp. 224-233. Springer, 2012.

[42]. M. Dashevskiy, Z. Luo. "Reliable Probabilistic Classification and Its Application to Internet Traffic". In: Huang, D.-S., Wunsch II, D.C., Levine, D.S., Jo, K.-H. (eds.) ICIC 2008. LNCS, vol. 5226, pp. 380–388. Springer, Heidelberg (2008).

[43]. P.C. de Vries, G. Arnoux, A. Huber, J. Flanagan, M. Lehnen, V. Riccardo, C. Reux, S. Jachmich, C. Lowry, G. Calabro, D. Frigione, M. Tsalas, N. Hartmann, S. Brezinsek, M. Clever, D. Douai, M. Groth, T.C. Hender, E. Hodille, E. Joffrin, U. Kruezi, G. F. Matthews, J. Morris, R. Neu, V. Philipps, G. Sergienko, M. Sertoli and JET EFDA contributors. "The impact of the ITER-like wall at JET on disruptions". Plasma Physics and Controlled Fusion **54** (2012) 124032 (9pp).

ILW campaigns (years 2011-2012)	%
True positives (disruption success rate)	98.36
False positives (missed alarm rate)	1.64
True negatives (non-disruptive rate)	99.08
False negatives (false alarm rate)	0.92

Table 1: Results of APODIS with the three first ILW campaigns without any retraining from the data corresponding to the C wall campaigns C19-C22.

Signal name	Acronym	Units
Plasma current	Ip	A
Mode locked amplitude	ML	T
Plasma internal inductance	LI	
Plasma density	Ne	m^{-3}
Stored diamagnetic energy time derivative	dW/dt	W
Radiated power	Pout	W
Total input power	Pin	W

Table 2: List of signals to characterize the disruptive/non-disruptive status of JET plasmas.

Feature id.	Definition
1	mean(Ip)
2	std(|fft(Ip)|)
3	mean(ML)
4	std(|fft(ML)|)
5	mean(LI)
6	std(|fft(LI)|)
7	mean(Ne)
8	std(|fft(Ne)|)
9	mean(dW/dt)
10	std(|fft(dW/dt)|)
11	mean(Pout)
12	std(|fft(Pout)|)
13	mean(Pin)
14	std(|fft(Pin)|)

Table 3: Feature identification. Acronyms are related to table 2. mean(.) represents the mean value during the time window of 32 ms. std(|fft(.)|) signifies the standard deviation of the Fourier spectrum during the time window of 32 ms (the DC component has been removed).

nfeat	$C_{14,nfeat}$	GSR (%)	GFAR (%)
2	91	96.68±2.18	70.30±30.68
3	364	96.79±2.13	67.82±31.37
4	1001	96.37±1.97	58.19±30.68
5	2002	96.34±1.76	54.94±27.81
6	3003	96.38±1.56	52.97±24.67
7	3432	96.47±1.39	51.53±21.51

Table 4 shows the average success and false alarm rates corresponding to the different predictors developed with different number of features. Globally, the success rates are above 96% but the false alarm rates are too high. This rate decreases with the number of features but not sufficiently. According to these results, the selection of a specific predictor should not to be based on the success rate (quite high in all cases) but on a combination of high success rate and false alarm rate (the smaller the better). Table 5 provides the best two predictors for each number of features (different cell backgrounds allow the distinction on the number of features).

Feature id.														SR (%)	FA (%)	AVP
1	2	3	4	5	6	7	8	9	10	11	12	13	14			
		x	x											94.00	4.70	0.813±0.187
	x		x											92.50	5.09	0.831±0.169
	x	x	x											94.00	4.31	0.809±0.191
		x	x							x				94.00	4.70	0.813±0.187
	x	x	x	x										94.00	4.21	0.811±0.189
	x	x	x							x				94.00	4.21	0.810±0.190
	x	x	x	x						x				94.00	4.21	0.811±0.189
	x	x	x			x	x							94.00	4.21	0.803±0.197
	x	x	x			x	x			x				94.00	4.21	0.803±0.197
	x	x	x	x						x	x			94.00	4.31	0.810±0.190
	x	x	x	x		x	x			x				94.00	4.31	0.802±0.198
	x	x	x			x	x			x	x			94.00	4.31	0.802±0.198

Table 5: Success rate (SR), false alarm rate (FA) and average prediction probability (AVP) for several combination of feature vectors. The numbers to identify features correspond to the ones of table 3.

Features	AWT (ms)	AF30 (%)
3, 4	654.564	83.0
2, 4	732.281	79.0
2, 3, 4	641.798	83.0
3, 4, 11	654.564	83.0
2, 3, 4, 5	654.394	83.0
2, 3, 4, 11	653.884	83.0
2, 3, 4, 5, 11	654.394	83.0
2, 3, 4, 7, 8	672.947	83.5
2, 3, 4, 7, 8, 11	672.947	83.5
2, 3, 4, 5, 11, 12	707,330	83.0
2, 3, 4, 5, 7, 8, 11	673.628	83.5
2, 3, 4, 7, 8, 11, 12	725.883	83.5

Table 6: The column 'Features' identifies the features from table 3. AWT is the average warning time of the different predictors and AF30 is the accumulative fraction of detected disruptions 30 ms before the disruption.

#disruption
1
2
13
25
27
40
85
93
101
122
128
161
183

Table 7: The column shows the number of disruption, in chronological order, with missed alarms. The row corresponding to disruption number 1 is not a missed alarm. It allows the generation of the first classifier to start to make predictions.

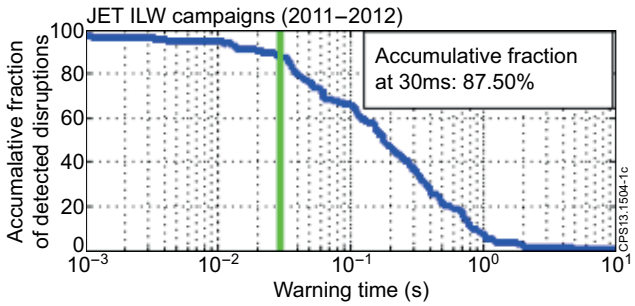

Figure 1: Accumulated fraction of detected disruptions with APODIS during the ILW campaigns of JET in the period 2011-2012.

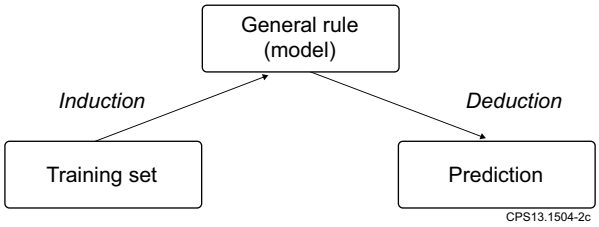

Figure 2: A classifier has to be understood as consisting of two steps: induction and deduction. Both of them are indissolubly linked together in the notion of classical classifiers.

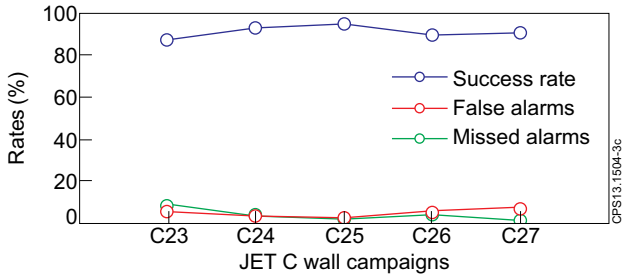

Figure 3: Tests of the APODIS predictor that show the stability of the success, false alarms and missed alarms rates during several campaigns after the training (campaigns C19-C22). Campaign C27 has been the last campaign with the C wall.

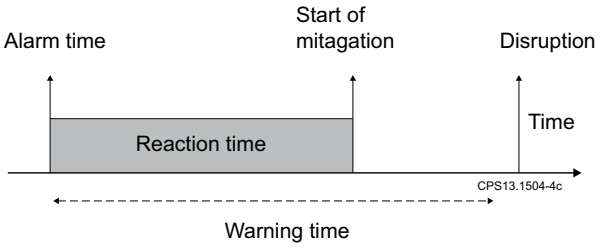

Figure 4: If the reaction time is larger than the warning time, the mitigation action is late.

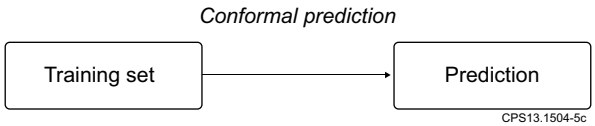

Figure 5: Conformal predictors do not follow the induction/deduction steps. All training samples are used with each new prediction.

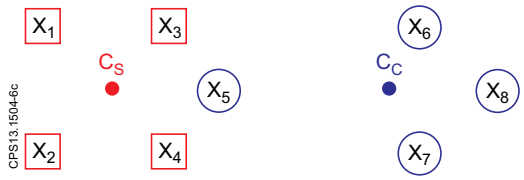

Figure 6: The four left-most samples of this example have label 'square' and the four right-most ones have label 'circle'. $\mathbf{C}_s$ and $\mathbf{C}_c$ are respectively the centroids of the classes 'square' and 'circle'. With the nearest centroid taxonomy, the number of categories is equal to the number of labels. Therefore, in the example, the category of the respective samples derived from the NCT taxonomy (eq.(2)) are $\tau_1 = \square$, $\tau_2 = \square$, $\tau_3 = \square$, $\tau_4 = \square$, $\tau_5 = \square$, $\tau_6 = \bigcirc$, $\tau_7 = \bigcirc$, $\tau_8 = \bigcirc$.

...	...	...	...	
Feat 1	Feat 2	...	Feat n	(t_D−96, t_D−65)
Feat 1	Feat 2	...	Feat n	(t_D−64, t_D−33)
Feat 1	Feat 2	...	Feat n	(t_D−32, t_D−1)
D I S R U P T I O N (t_D)				

CPS13.1504-7c

Figure 7: Each row corresponds to a feature vector during the discharge. The components (in a number between 2 and 7) have been calculated in the time interval of the last column (times in ms and the disruption takes place at time t_D). The feature vector that represents a disruptive behaviour in the 'training set' is the feature vector previous to the disruption (in grey background).

Discharge 1

...	...	...	...	
Feat 1	Feat 2	...	Feat n	32ms
Feat 1	Feat 2	...	Feat n	32ms
...	...	...	...	

. . .

Discharge K_1

...	...	...	...	
Feat 1	Feat 2	...	Feat n	32ms
Feat 1	Feat 2	...	Feat n	32ms
...	...	...	...	

mean(col 1)	mean(col 2)	...	mean(col n)

CPS13.1504-8c

Figure 8: The example that describes a safe behaviour in the first training set is a row vector whose components are the mean values of the different features between the first non-disruptive discharge and the last one previous to the first disruption.

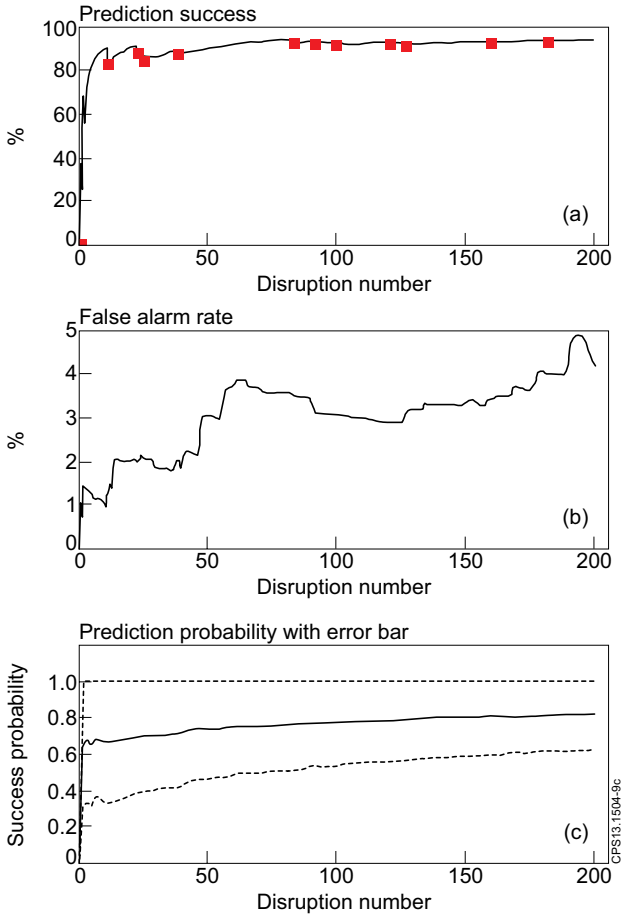

Figure 9: Results of the Venn predictor with features 2, 3, 4 and 5: evolution of the success rate, false alarm rate and prediction probability with the number of disruptions.

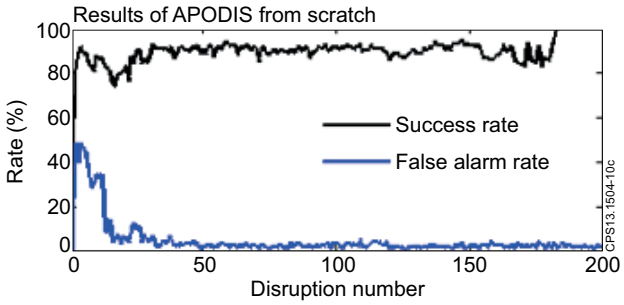

Figure 10: Results of the APODIS version from scratch developed in [33]. It should be emphasized the different scales in the Y axis between this figure and fig. 9b.

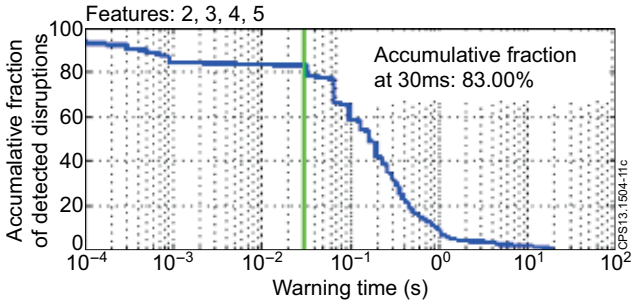

Figure 11. Warning times from the Venn predictor with features 2, 3, 4 and 5.