EFDA–JET–CP(14)04/01

A. Acero, J. Vega, S. Dormido-Canto, M. Guinaldo, A. Murari
and JET EFDA contributors

Assessment of Probabilistic Venn Machines as Real-Time Disruption Predictors from Scratch: Application to JET with a View on ITER

Assessment of Probabilistic Venn Machines as Real-Time Disruption Predictors from Scratch: Application to JET with a View on ITER

A. Acero[1], J. Vega[2], S. Dormido-Canto[3], M. Guinaldo[3], A. Murari[4]
and JET EFDA contributors*

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

[1]CIEMAT, Computer Centre, Madrid, Spain
[2]CIEMAT, Laboratorio Nacional de Fusión, Madrid, Spain
[3]UNED, Departamento Informática y Automática, Madrid, Spain
[4]Associazione EURATOM-ENEA sulla Fusione, Consorzio RFX Padova, Italy
** 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 Proceedings of the
19th REAL-TIME Conference, Nara, Japan
26th May 2014 – 30th May 2014

ABSTRACT

Due to the good off-line results (in terms of high learning rate, high success rate, low false alarm rate and high prediction probability) provided by Venn machines as adaptive disruption predictors in JET, this article assesses their use under real-time requirements. Venn machines predictions can be expensive from a computational point of view but they can be used in JET in a deterministic way. The JET characteristic time to take mitigation actions for disruptions is about 30ms. This article shows that Venn predictions take computation times of 1ms for JET conditions. The influence of both the dimensionality of feature vectors and the number of training examples to make predictions are analyzed. Also a short discussion about the potential applicability to ITER is presented.

1. INTRODUCTION

A disruption is a catastrophic loss of plasma control in tokamaks that can produce irreversible damage to a fusion device. Therefore, disruptions can be a big issue in future reactors such as ITER or DEMO. The concepts of avoidance and mitigation are closely related to disruptions. The former is associated with the objective of free disruption operation. The latter is aimed at the alleviation of the disruption detrimental effects when a disruption is unavoidable. 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.

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 [1]. Some related problems are: incomplete models, strong assumptions and unphysical boundary conditions. Due to the lack of reliable physics models, data-driven models are a practical alternative. They are based on discovering relations between signals to identify an incoming disruption. These data-driven predictors follow the temporal evolution of the discharges and make predictions on periodic basis (ms or tens of ms).

To train a data-driven disruption predictor, the more discharges the better. Training datasets can use hundreds of disruptions, like for instance the Advanced Predictor of DISruptions (APODIS) on JET. However, next generation of tokamaks cannot wait for hundreds of disruptions to obtain a predictor. Recent analyses have shown that it is possible to develop adaptive disruption predictors from scratch [2]. Dormido-Canto et al. uses APODIS in [2] to develop an adaptive disruption predictor from scratch. The main result shows that about 40 disruptions are needed in the training set to have good prediction capabilities: success rate of 93.5% and false alarm rate of 2.3%.

Trying to speed up the learning process and to reduce the number of required discharges, a new proposal of predictor from scratch was put forward in [3]. It is based on Venn predictors [4] (or Venn machines) that provide with each prediction not only a probability but also a probability error bar. The results in [3] show that reliable disruption predictors can be obtained in an adaptive way starting with predictors that have just 1 disruptive example and 1 non-disruptive example. To include more knowledge about the disruptive/non-disruptive parameter space, the number of training examples is increased after each missed alarm. This adaptive approach based on Venn predictors allows achieving a success rate and a false alarm rate of 94% and 4% respectively. These results have been obtained after the off-line analysis in chronological order of 1237 JET discharges (1036 non-disruptive and 201 disruptive) corresponding to the three first ILW campaigns (years 2011–2012). The average probability interval about the reliability and accuracy of all the individual predictions is 0.811 ± 0.189.

Venn predictors are a particular case of conformal predictors [4]. Conformal predictors always provide, together with the prediction, a measure about how reliable the prediction is. This reliability measure establishes the level of confidence in the prediction. In the case of the Venn predictors, the confidence in the prediction is provided by both the probability and the corresponding error bar.

The price to pay for having reliable predictions is computation time. In general, an important drawback of conformal predictors is the computation time needed to make predictions. Therefore, the application of this kind of predictors under real-time conditions can be an issue. Due to the good results obtained in JET (in terms of success rate, false alarm rate and probabilities), this article evaluates the possibility of using Venn predictors as disruption predictors from scratch under real-time requirements. Tests have been performed with the JET database. At present, the time cycle to identify a disruptive behaviour in JET is 32 ms. This interval is about the minimum needed in JET to perform mitigation actions after the recognition of a forthcoming disruption. Therefore, this time is the upper bound for the real-time data processing (feature extraction from the plasma quantities plus the prediction algorithm) to implement a Venn disruption predictor from scratch in JET.

The good results of the adaptive Venn predictor with the JET database makes it a good candidate as predictor from scratch for ITER. With this type of predictor, there is no need to wait for tens of disruptions to produce a reliable prediction. The only point to assess, as mentioned previously, is to ensure that the computation time to make predictions fits the cycle time to put into operation mitigation actions in a fusion device.

Section 2 gives a summary of the Venn predictor theory and exhaustive bibliography is presented. Section 3 shows the real-time needs to develop the adaptive Venn predictors in JET. Section 4presents the computation time results and, finally, section V is a short discussion about the potential needs for ITER.

2. VENN PREDICTORS

In terms of machine learning theory, disruption predictors are considered binary classification problems. Typically, a binary classifier is implemented in a two-step procedure (fig. 1). The first one is an inductive process, in which a general model (typically, a mathematical function) is created from a dataset of training examples. Generally speaking, the mathematical function is obtained by solving an optimization problem. The second step is a deductive process that uses the model created in the first step to make predictions. It is important to note that the training set is no longer utilized for the predictions. Instead, the model is used as a decision function to predict the labels of the new examples. Usually, the training process can be very expensive in terms of computational resources but the deductive step can require times below 1ms.

Conformal predictors (and the Venn predictors in particular) do no follow the inductive/deductive schema previously mentioned (Fig.1). The predictions are made in a transductive way, which means that no general model is created and the whole training dataset is necessary to make each new prediction. Conformal predictors assume each possible label for the example to classify. The classification with the maximum confidence is taken as prediction. Therefore, a binary classification problem implies the computation of two different confidence measures per example to classify, a fact which can involve a significant computational load.

Focusing the attention on binary Venn predictors, let's consider a training dataset consisting of objects ($\mathbf{x}_i \in \mathbb{R}'''$), labels ($y, \in \{0,1\}$) and pairs: ($\mathbf{x}_1$, y_1),..., ($\mathbf{x}_{,,}$, $y_{,,}$). As it has been described in the preceding paragraph, to predict a label $y_{\mathrm{n}+1}$ for a new object $\mathbf{x}_{\mathrm{n}+1} = \mathbf{x}_{new}$, it is necessary to check the different hypothesis $y_{\mathrm{n}+1} = y$ each time including the pair ($\mathbf{x}_1$, y_1),..., ($\mathbf{x}_{new}$, y) into the dataset. From a mathematical point of view [4], the idea of Venn predictors is based on a taxonomy function A_n, $n \in \mathbb{N}$, which classifies the relation between an example and the set of other examples:

$$\begin{aligned}\tau_i &= A_{n+1}((\mathbf{x}_i, y_i),\\ &\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{i-1}, y_{i-1}), (\mathbf{x}_{i+1}, y_{i+1}), \ldots, (\mathbf{x}_{n+1}, y_{n+1})\}).\end{aligned}$$

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

It is out of the scope of the article to go into an in depth explanation of Venn predictors. Reference [4] contains all the theory about them. Anyway, 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 [5], where the Venn predictors are based on neural networks. Lambrou et al. [6] 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. [7] describes the logistic taxonomy, which is created from the probabilistic method of logistic regression. Finally, this article uses the nearest centroid taxonomy, whose foundations can be found in [8].

3. REAL-TIME NEEDS OF ADAPTIVE VENN PREDICTORS FOR JET

To analyze JET real-time requirements for the adaptive Venn predictors described in [3], it is important to review the characteristics of the real-time prediction system. Taking into account the objective of '*learning from scratch*', only those quantities that are considered essential diagnostics from the start of a fusion device have been considered (Table I).

The quantities of table I are used to form feature vectors on a periodic basis (32ms). These feature vectors ($\mathbf{x} \in \mathbb{R}^m$) have m components that allow the characterization of the plasma behavior (disruptive/non-disruptive) every 32ms. It is well-known that the disruptive/non-disruptive character of the JET plasmas at any time instant is well defined through two different signal representations [9]: time domain and frequency domain. During the discharge, each signal is digitized at 1kSample/s and is processed in temporal windows 32ms long. The processing consists of computing, with the 32 samples per window, the mean value (temporal domain) and the standard deviation of the power spectrum (frequency domain) after removing the DC component. These two values per signal are the components of the feature vectors. Therefore, a feature vector is generated every 32ms and is used as input to the Venn predictor to recognize the plasma behavior.

According to the previous paragraph, the use of 7 plasma quantities (Table I) and two different representations per signal means that the maximum dimension of the feature vectors is 14. Reference [3] shows the results of Venn predictors developed with 14 features taken N_f at a time, where $N_f = 2, \ldots, 7$. The computations were performed in Matlab and the results are shown in Table 2.

The real-time implementation of the Venn predictor has to ensure that predictions must be performed every 32ms in less than 32ms. The total computation time for the prediction will be the sum of three different times: the time to compute the mean values, the time to compute the standard deviation of the signal power spectrum and the time required for the Venn predictor itself. Due to the fact that the Venn predictors assume each possible label for the example to classify (in this case, disruptive and non-disruptive), always two classifications are necessary within each 32ms temporal window. But the transductive character of the Venn predictors has to be taken into account. As mentioned previously, this means that the whole training dataset is necessary to make each new prediction. Therefore, two different classification problems have to be solved in less than 32ms and the number of training samples can be an issue in this respect (the more samples the longer the computation time).

Next section evaluates the real-time computation times of the Venn predictors developed in [3] with a nearest centroid taxonomy as a function of the feature vector dimensionality and the number of training examples.

4. COMPUTATION TIME ASSESSMENT

All simulations to determine the computation times have been performed in a PC with an Intel Xeon CPU E5-1603 3 @ 2.80 GHz processor and 6 Gbytes of RAM memory. The PC operating system is CentOS release 6.3 (Final). The codes have been written in C language and the compiler used is gcc 4.4.6.

It should be reminded that the predictions are performed with 32 samples per signal and the computations are carried out every 32ms. As a first step, the computation time to form the corresponding feature vector has to be considered. This time includes the computation time of both mean values and standard deviation of the power spectrum. Due to the impossibility of evaluating these times on individual cases, average values on millions of cases are shown. The average computation times corresponding to the mean values of 32 samples is 0.122±0.001us. In the same way, the average values to compute the standard deviation of the power spectrum of 32 samples is 4.339±0.016 us.

Next section evaluates the real-time computation times of the Venn predictors developed in [3] with a nearest centroid taxonomy as a function of the feature vector dimensionality and the number of training examples.

As mentioned, the Venn prediction computation times depend on both the feature vector dimensionality and the number of training examples to make the predictions. With regard to the feature vector dimensionality, it should be noted that as reported in Table II, a maximum number of 7 features has been considered. Owing to the fact that the most time consuming computation is the estimation of the standard deviation of the power spectrum, this time will be used as an upper bound of the computation time to form a feature vector. Therefore, this maximum computation time (t_F) follows the equation

$$t_F \leq 4.339 \cdot N_f \text{us}, N_f = 2,..., 7$$

where N_f is the number of features (i.e. the feature vector dimensionality).

Figure 2 shows the computation times of Venn predictions as a function of the number of features. The plot represents JET results with a number of features per vector between 2 and 7. The different curves (from top to bottom) correspond to different training examples (2000, 1750, 1500, …, 500, 250).

Figure 3 shows the same results of fig.2 but in a (training examples, computation time) representation. In this plot, the curves from top to bottom are the number of features (7, 6, …, 2).

According to these simulations, the maximum computation time in JET, to generate the feature vector and to perform the prediction (which signifies two probability estimations about disruptive and non-disruptive behaviors), is less than 1ms. This time has to be compared with the characteristic time in JET to take mitigation actions that is about 30ms. It is important to emphasize that this results corresponds to use a maximum number of 7 features to describe the JET parameter space and a maximum number of 2000 training examples. Both situations are suitable for JET.

5. DISCUSSION

The Venn disruption predictor developed in [3] can be used under real-time requirements in JET because it provides a high learning rate, high success rate, low false alarm rate, high prediction probability and low computational cost. Therefore, Venn predictors are potential candidates as disruption predictors from scratch for ITER. However, future work has to deal with specific ITER factors that can limit the prediction determinism in a running discharge. Firstly, only seven plasma quantities are enough for a proper description of the JET disruptive/non-disruptive parameter space. In ITER a larger number of plasma quantities can be necessary and, depending on the number, this can have a big impact on the dimensionality of the feature vectors. Secondly, reference [3] shows that the disruptive/non-disruptive characteristics of JET plasmas can be condensed in few feature vectors. This means that the number of training samples to make predictions does not need to be large. However, this situation is unknown for ITER and a more exhaustive analysis about the influence of the number of training examples on the computation time has to be tackled. Thirdly, it should be noted that the prediction codes can be parallelized. The most immediate analysis about this is to execute in parallel the two probability estimations corresponding to the assumptions of each possible label for the example to classify every 32ms.

ACKNOWLEDGEMENTS

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

[2]. S. Dormido-Canto, J. Vega, J.M. Ramírez, A. Murari, R. Moreno, J.M. López et al. “Development of an efficient real-time disruption predictor from scratch on JET and implications for ITER”. Nuclear Fusion. **53** (2013) 113001 (8pp).

[3]. J. Vega, A. Murari, S. Dormido-Canto, R. Moreno, A. Pereira, A. Acero. “Adaptive high learning rate probabilistic disruption predictors from scratch for the next generation of tokamaks”. Nuclear Fusion. Submitted for publication.

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

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

[6]. 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.

[7]. 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.

[8]. 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).

[9]. J. Vega, S. Dormido-Canto, J.M. López, A. Murari, J.M. Ramírez, R. Moreno et al. “Results of the JET real-time disruption predictor in the ITER-like wall campaigns”, Fusion Engineering and Design **88** (2013) 1228-1231.

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 1: List of signals to characterize the disruptive/Non-disruptive status of the JET Plasmas.

Feature id.								SR(%)	FA (%)	AVP
2	3	4	5	7	8	11	12			
	x	x						94.0	4.7	0.81±0.19
x		x						92.5	5.1	0.83±0.17
x	x	x						94.0	4.3	0.81±0.19
	x	x				x		94.0	4.7	0.81±0.19
x	x	x	x					94.0	4.2	0.81±0.19
x	x	x				x		94.0	4.2	0.81±0.19
x	x	x	x			x		94.0	4.2	0.81±0.19
x	x	x		x	x			94.0	4.2	0.80±0.20
x	x	x		x	x	x		94.0	4.2	0.80±0.20
x	x	x	x			x	x	94.0	4.3	0.81±0.10
x	x	x	x	x	x	x		94.0	4.3	0.80±0.20
x	x	x		x	x	x	x	94.0	4.3	0.80±0.20

Table 2: 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.

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

Table 3: Feature identification. Acronyms are related to Table 1. Mean(.) Represents the mean value during the time window of 32ms. STD(|FFT(.)|) Signifies the standard deviation of the fourier spectrum during the timewindow of 32ms (The DC component has been removed).

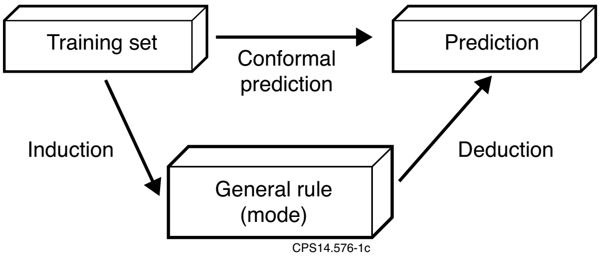

Figure 1: A traditional classifier has to be understood as consisting of two steps: induction and deduction. However, conformal predictors do not follow the induction/deduction steps. All training samples are used with each new prediction and no general rule is needed.

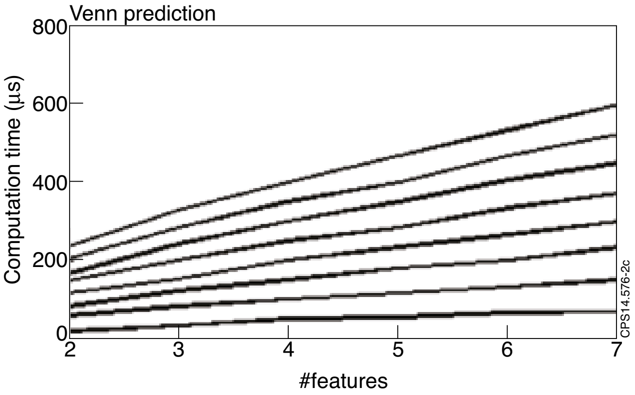

Figure 2: Computation times for the Venn predictors as a function of the feature vector dimension.

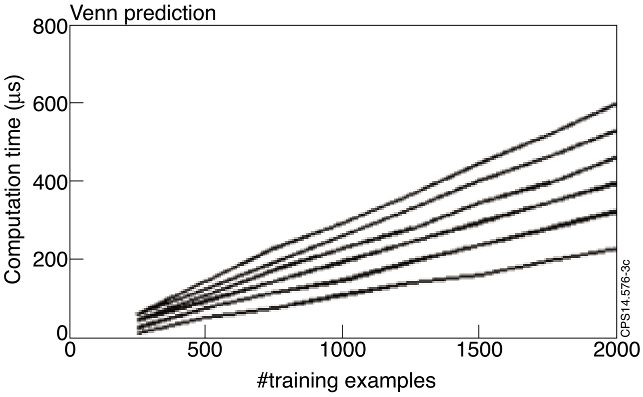

Figure 3: Computation times for the Venn predictors as a function of the number of training examples.