Understanding disruptions in tokamaks with interpretable machine learning

Andrew D. Maris

Plasma Science and Fusion Center, MIT

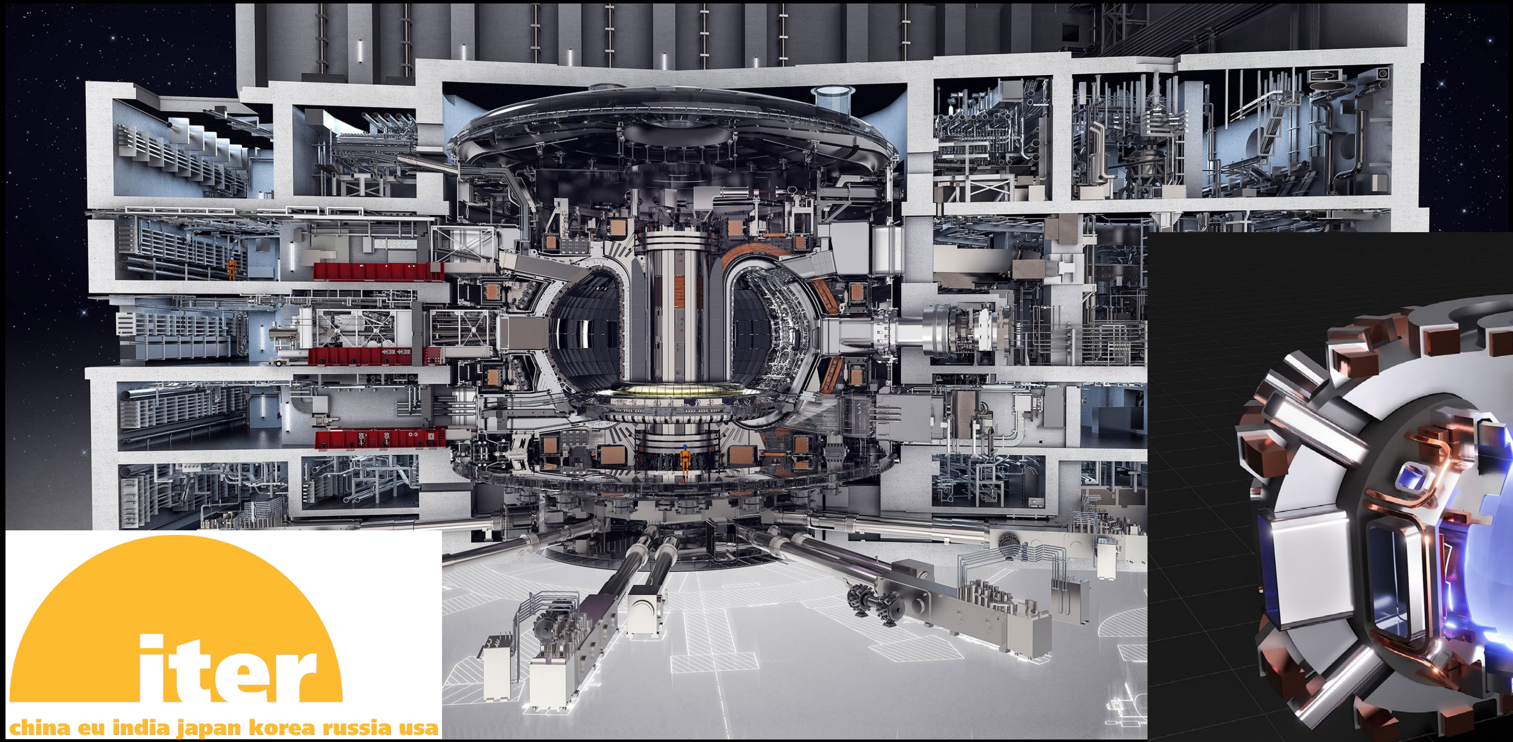
8/16

Outline

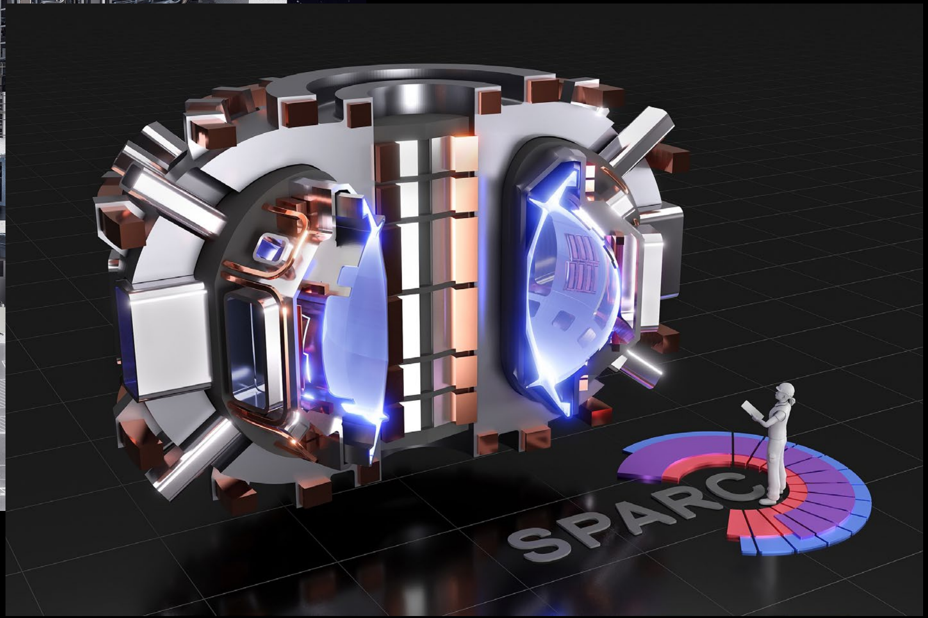
- A primer on tokamak disruptions
- Disruption prediction with machine learning
- My research: creating a physically interpretable disruptivity boundary using linear SVMs

A primer on tokamak disruptions

The tokamak is one of the leading candidates for a near-term fusion energy



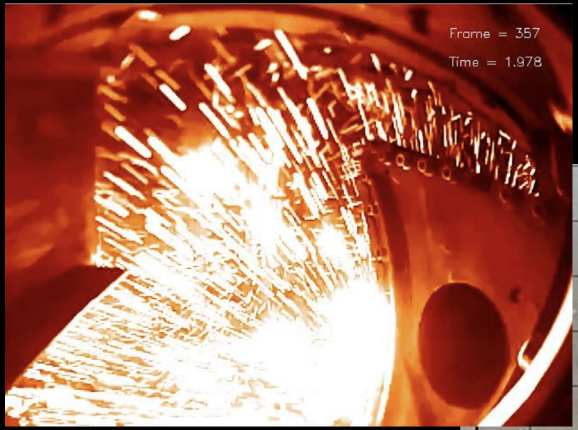
Schematic of ITER Tokamak and Plant System (adapted from Oak Ridge National Laboratory)



Rendering of SPARC (Commonwealth Fusion Systems and MIT)

Disruptions are a "grand challenge" for the tokamak fusion energy path

- A *disruption* is a sudden loss of plasma confinement
- Disruptions can cause major damage in a tokamak via
  - High heat fluxes on PFCs
  - Currents in tokamak wall (**J** x **B** forces)
  - Runaway electron beam



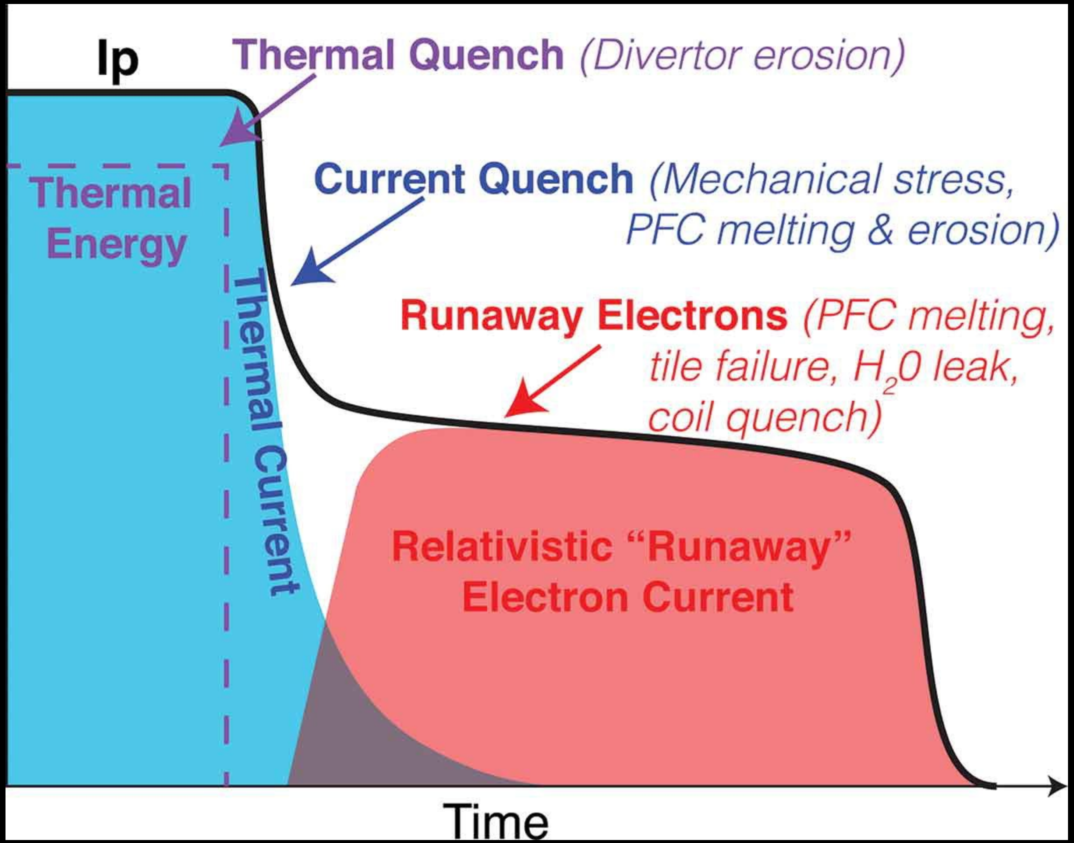
View from visible camera of disruption on Alcator C-Mod. Courtesy R.A. Tinguely

Melting caused by a runaway electron beam on JET (https://www.iter.org/newsline/-/2234)

Disruptions in next-generation devices could be catastrophic without intervention

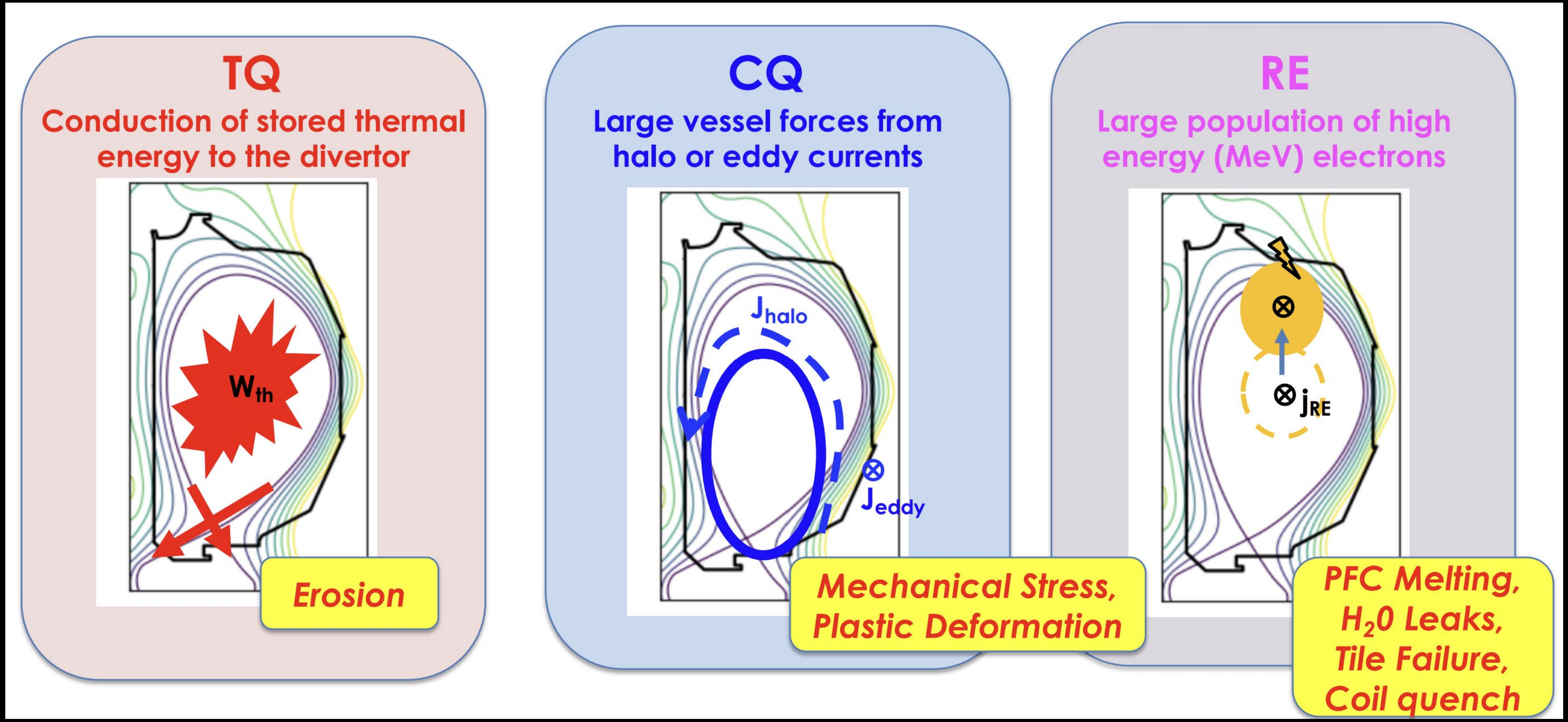
A disruption generally has three stages

1. Thermal quench (TQ): plasma deposits nearly all thermal energy into the wall
2. Current quench (CQ): plasma current begins to sharply decline
3. Runaway electrons (RE): beam of high energy electrons forms, as much as 2/3 original plasma current



Stages of a disruption listed with associated damage, (N. W. Eidietis NF 2021)

All three stages pose significant danger to the machine



Stages of a disruption listed with associated damage, (N. W. Eidietis IAEA-PPPL Theory and Simulation of Disruptions Workshop 2021)

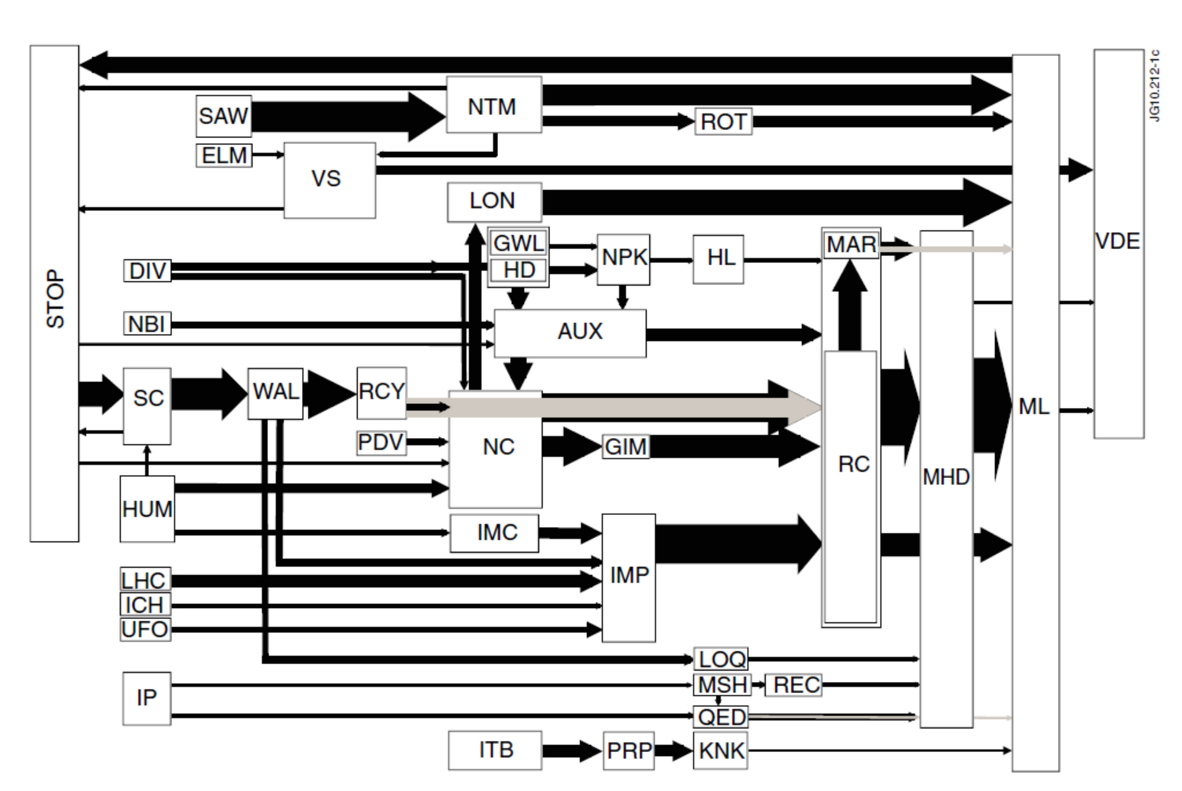
We can limit the effect of disruptions via scenario optimization, avoidance, mitigation, and resilience

Strategy	Definition	Analogy to driving a car
Scenario optimization	Planning/crafting stable plasma configurations	Choosing to drive on safe, familiar roads
Avoidance	Identifying and eliminating the disruption precursors	Changing lanes to avoid debris on the road
Mitigation	Bring the discharge to “soft” landing by launching a large amount of mass into the plasma	Slamming on the breaks
Resilience	Designing the machine to withstand disruption effects	Choosing a car built with sturdy components

Disruption prediction with machine learning

Disruptions are challenging to predict, especially in real time

- Disruptions can be caused by complex chains of events that can include a range of precursors such as:
  - MHD events
  - Hardware failures
  - "UFOs"
- Disruption precursors can evolve faster than first-principles simulations
  - Only physics event-based models or ML can keep up with plasma



Disruption event chain at JET (De Vries et al. NF 2011)

Machine learning (ML) allows us to build data-driven models from empirical results

Old-school automation (idealized)

1. Identify task
2. Find domain expert who completely understands a task
3. Get expert to encode knowledge into computer
4. Debug
5. Success!

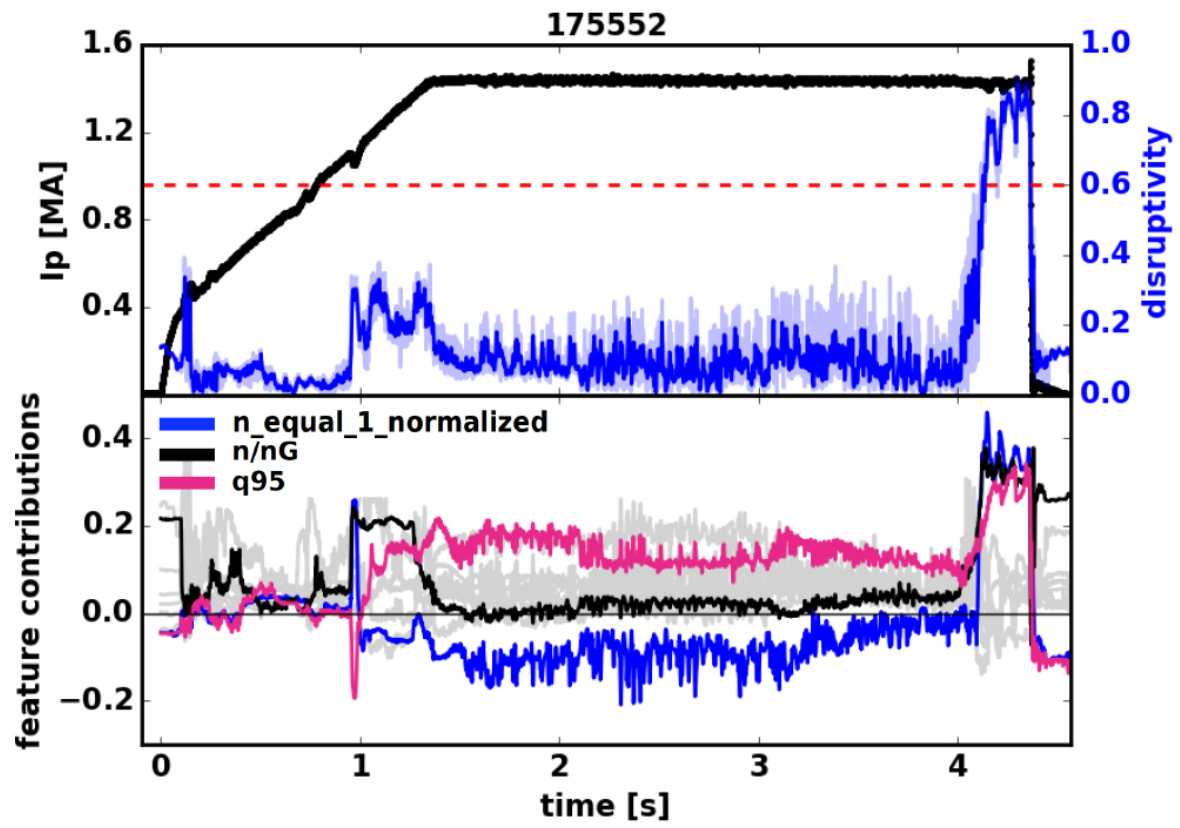
Machine learning (idealized)

1. Identify task
2. Find empirical data related to task
3. Use domain expert to pre-process data
4. Tune machine learning model
5. Success!

ML can be a powerful tool for studying disruptions

- Pre-trained ML models can execute computations almost instantaneously → enable use in real-time control settings
- Three main applications of ML for disruptions
  - Train on database of simulations to approximate simulation codes in real-time (surrogate model)
  - Event identification for database creation
  - **Train on experimental data to predict onset of disruption**

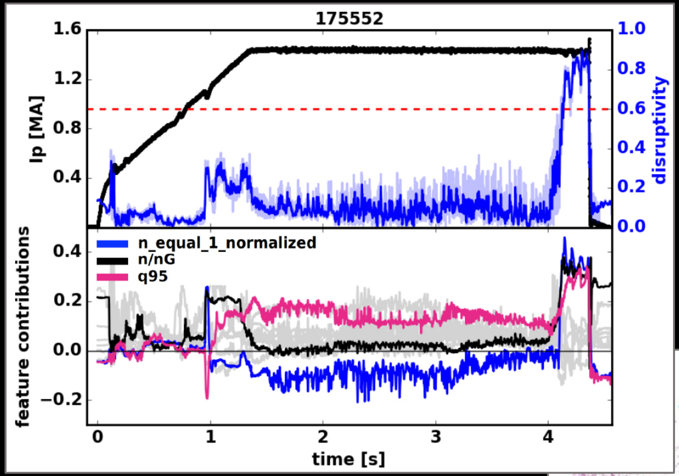
Interpretable ML can be a powerful tool for disruption avoidance and mitigation



Example of interpretable ML disruption predictor, Rea et al, NF 2019

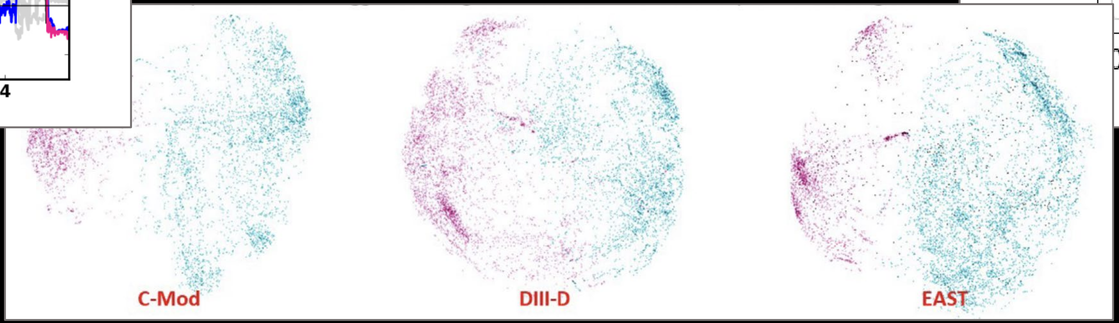
Our group is involved in a range of ML-based disruption research

Interpretable disruption prediction with Random Forests
Cristina Rea – Research Scientist



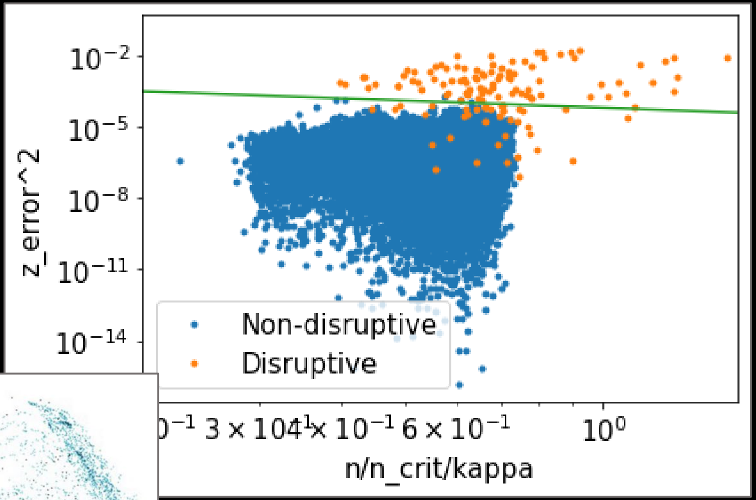
Example of interpretable ML disruption predictor (Rea et al, NF 2019)

Scenario adaptive disruption prediction
Jinxiang Zhu – PhD Candidate



PCA clustering plots showing separation between high (purple) and low (cyan) performance discharges, (J. X. Zhu, IAEA-PPPL Theory and Simulation of Disruptions Workshop 2021)

Creating an interpretable physical VDE boundary using linear SVM
Andrew D. Maris – PhD Student



Summary

- Disruptions are a grand challenge for the tokamak fusion energy path
  - There are three phases of a disruption, each with their own threats
    1. Thermal quench
    2. Current quench
    3. Runaway electron
- Disruption prediction enables avoidance and mitigation, thereby reducing the threat of disruptions
- Machine learning can assist in the study of disruptions
  - Interpretable ML opens the door to data-driven avoidance and mitigation
  - ML is an exciting frontier of disruption studies!

Thank you!

A special thanks to my advisors Cristina Rea and Bob Granetz, whose presentation was the basis of this talk

One more thing! Do you want to help shape the future of fusion energy in the US?

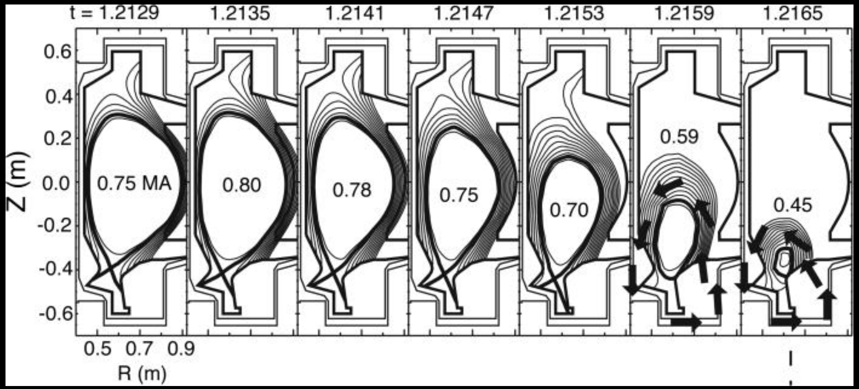
We are forming a graduate student organization to move the needle on fusion energy through public and legislative engagement. That could mean writing for public audiences and even visiting your congressperson in DC! Our goal is to make this low time commitment with big impact.

Join the Discord (I will drop in the chat) and join the email list (https://forms.gle/qVjdpRH3PEopc64RA)!

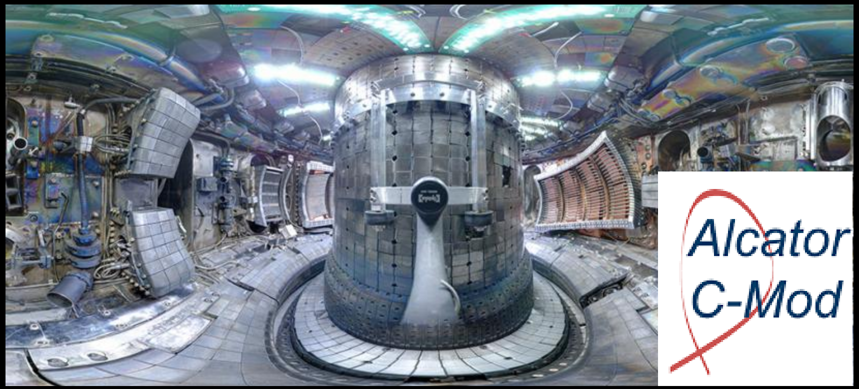
Creating a physically interpretable disruptivity boundary using linear SVMs

We plan to identify "hot" Vertical Displacement Events (VDEs)

- Why VDEs?
  - VDEs are relatively well understood, enabling us to validate the disruptivity boundary with known physics
  - Succeeding with VDEs will build confidence for more exotic disruption types
- We will narrow focus to C-Mod for preliminary study
- Potential limitations of this approach
  - Functional form of boundary may be complicated – large search space
  - Does not take time history into account

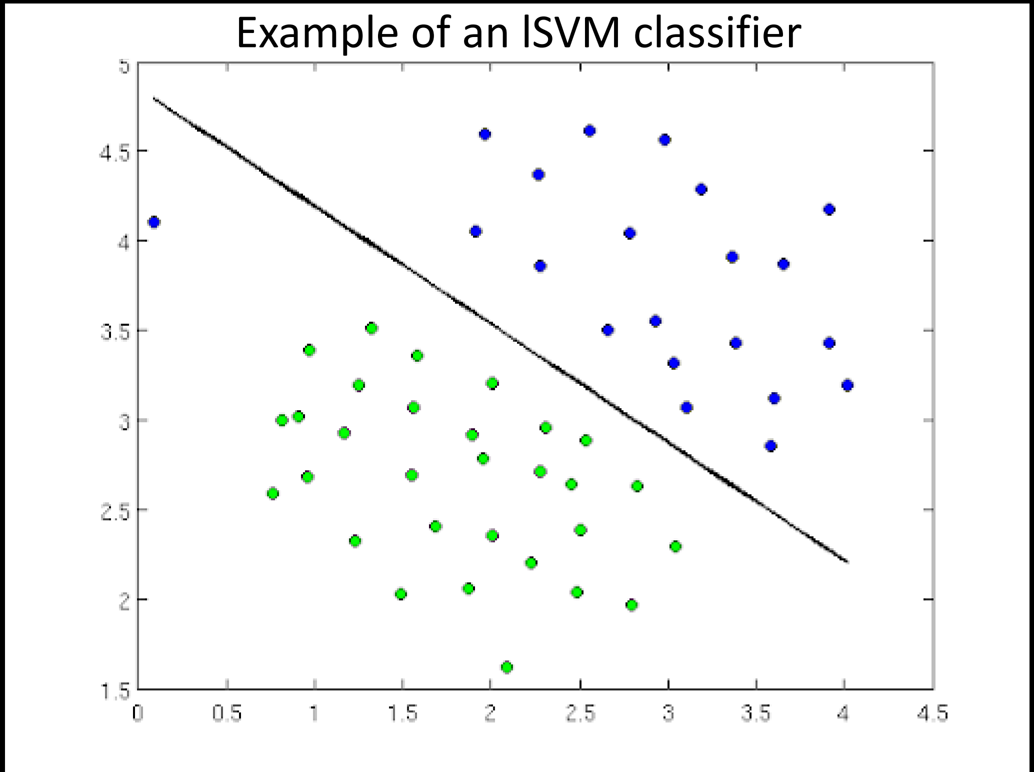


ITER Physics Expert Group on Disruptions, Plasma Control, and MHD and ITER Physics Basis Editors, NF (1999)



Interior of Alcator C-Mod

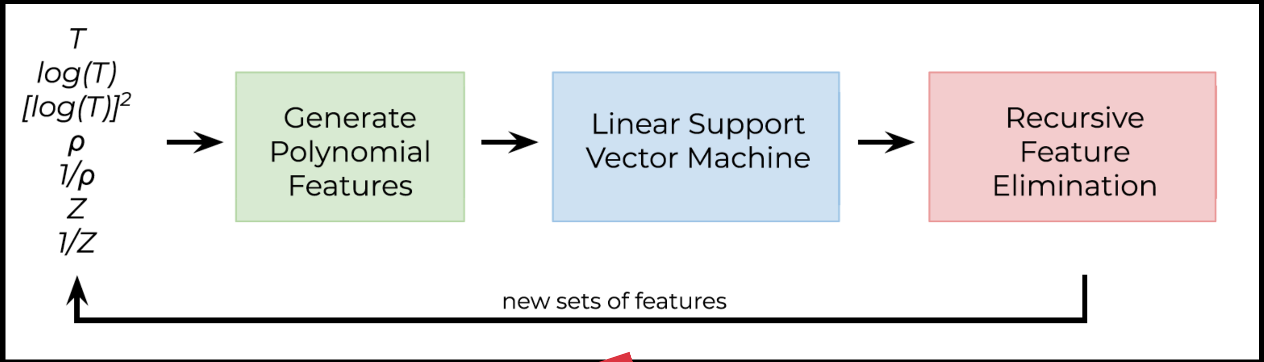
Linear support vector machines (lSVMs) create a linear boundary between classes of data



(Machine Learning, Stanford Open Classroom)

lSVMs & recursive feature updating (RFU) have been used to create nonlinear symbolic decision boundaries

Workflow



Learned decision boundary

$$\xi = \log^2(\mathrm{T/eV})\frac{(\rho + 10)/(\mathrm{g/cm^3})}{\mathrm{Z}} > 2.0,$$

$\approx$

$$\zeta = \frac{\langle Z \rangle}{Z} > 0.35.$$

Physical decision boundary

Machine Learning Discovery of Computational Model Efficacy Boundaries. Murillo (2020), PRL

We investigated whether we can build a better VDE disruptivity boundary using lSVMs

- Traditional vertical stability metric: $n/n_{crit} = 1$
- Two new approaches for finding VDE boundary
  - Use lSVM to build **polynomial** disruptivity boundary (described in next slide)
  - Use lSVM to build **power law** disruptivity boundary
- Features/signals:
  - κ
  - l_i
  - lower gap
  - n/n_{crit}
  - $Ip_{error,\ normalized}$
  - $z_{error}^{\ 2}$
  - $v_z^{\ 2}$

Main findings

- ML approach develops better vertical disruptivity boundary than n/n_crit
- Both polynomial and power law tend to converge to a boundary in z_error that is modified by n/n_crit
  - z_error is most predictive single feature
- Elongation does not seem to be an important parameter → manually removing it at the beginning of the ML pipeline has no effect on final result

ISVM workflow

- CV: Hyperparameter and feature selection
  - Find optimal hyperparameter C for all features
  - Find optimal combination of features using 5-fold group CV
  - If polynomial approach
    - Create inverse features (ex. $1/l_i$)
    - Create 2^{nd}, 3^{rd}, or 4^{th} order combinations of all features (including division)
    - Find optimal combination of these higher-order features using 5-fold group CV
  - Find optimal hyperparameter C for final set of features
  - *Caveats: results can be sensitive to initial C, scoring method, and different train/test splits*
    - Usually use f1 score for power law and f2 score for polynomial
- Test
  - A positive detection for a particular shot occurs when two sequential time steps are predicted to be disruptive (except for the polynomial approach, where only one time step is needed. This is because the polynomial approach naturally is robust to false positives)

Preliminary results shows this approach works far better than n/n_crit VDE boundary

Baseline: n/n_crit

n/n_crit = 1.03

- FPR: 71.3%
- TPR (% of VDE shots "caught")

t until disrupt	TPR
20ms	68%
15ms	69%
10ms	70%
0ms	73%

Power law

|n_over_ncrit|^0.68 |z_error|^0.59 = 0.032

- FPR: 47.1%
- TPR "

t until disrupt	TPR
20ms	80%
15ms	88%
10ms	91%
0ms	100%

Preliminary results show it works well, but more analysis needed

Power law

|n_over_ncrit|^0.68 |z_error|^0.59 = 0.032

FPR: 47.1%

TPR:

t until disrupt	TPR
20ms	80%
15ms	88%
10ms	91%
0ms	100%

Polynomial

n/n_crit * z_error^2 = 5.2*10^-5

FPR: 5.2%

TPR:

t until disrupt	TPR
20ms	48%
15ms	64%
10ms	79%
0ms	96%