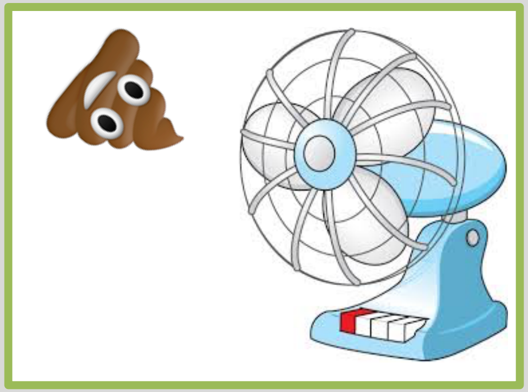
Disruption Predictor Feature Developer

(When you need to know before plasma hits the wall)



Matthew Parsons, Bill Tang, Eliot Feibush

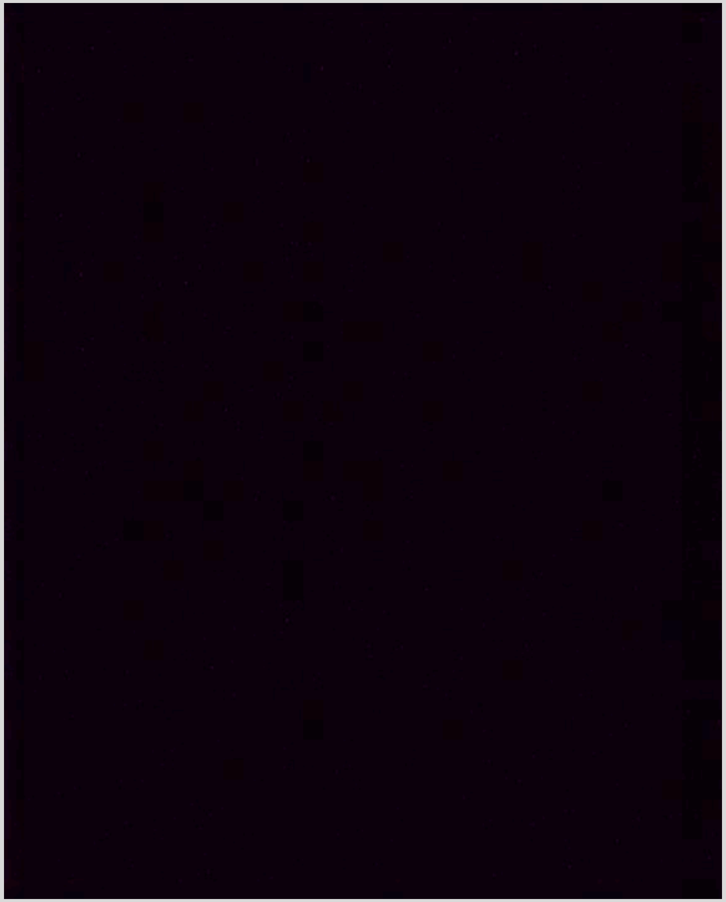
Outline

- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines
- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?
- Future Pursuits

Outline

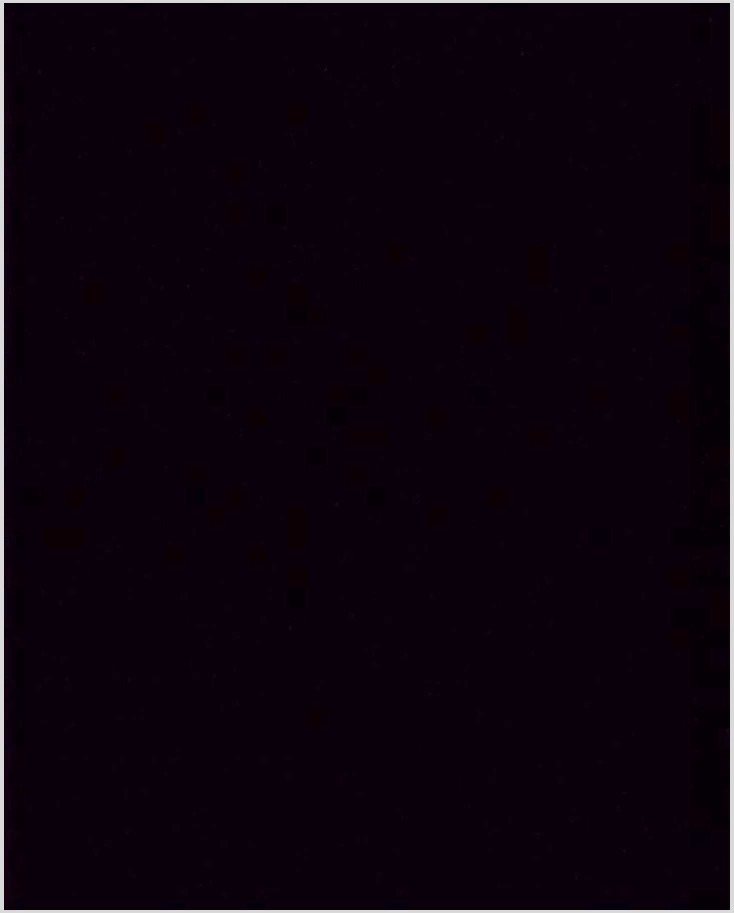
- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines
- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?
- Future Pursuits

JET Disruptions



82503

KL1-O8WA Visible Camera

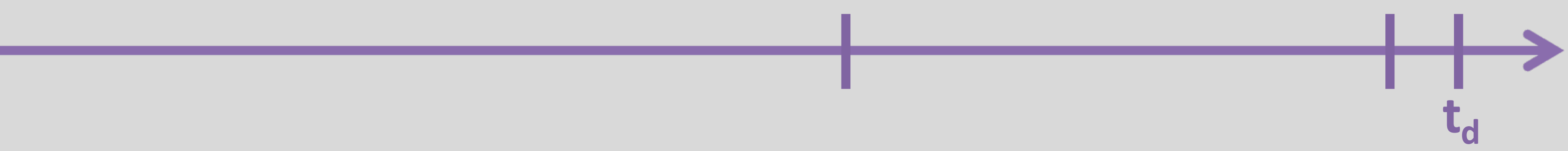


82502

M.S. Parsons, 23 May 2016
NSTX-U Monday Physics Meeting

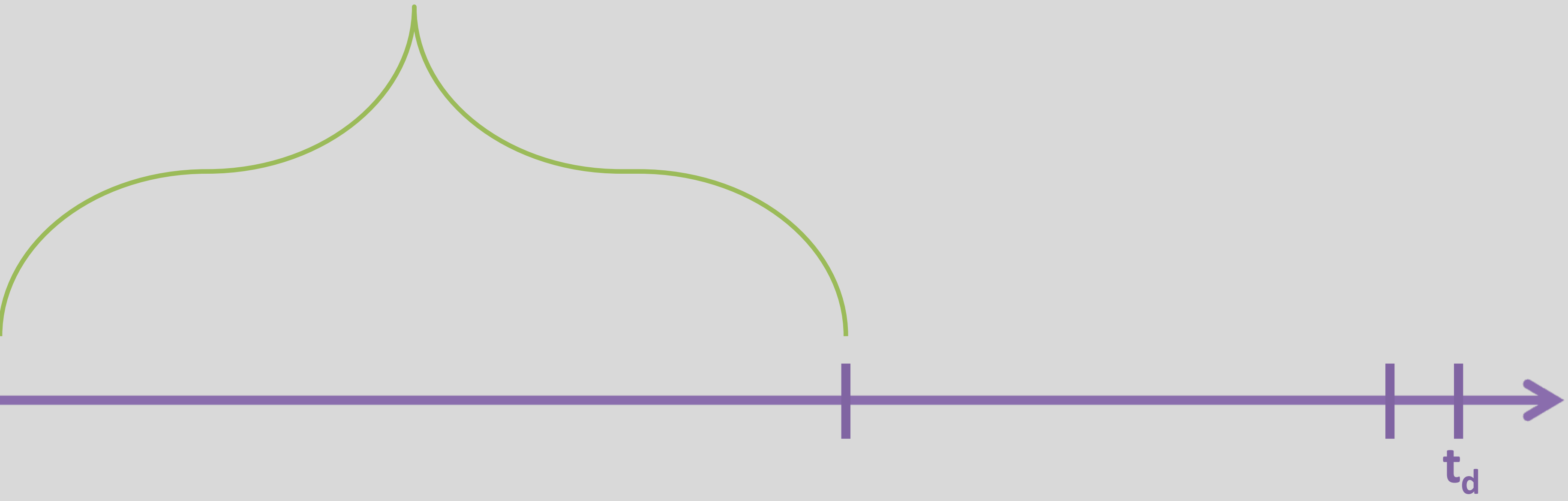
Disruption Prediction Objective

Timeline of Discharge



Disruption Prediction Objective

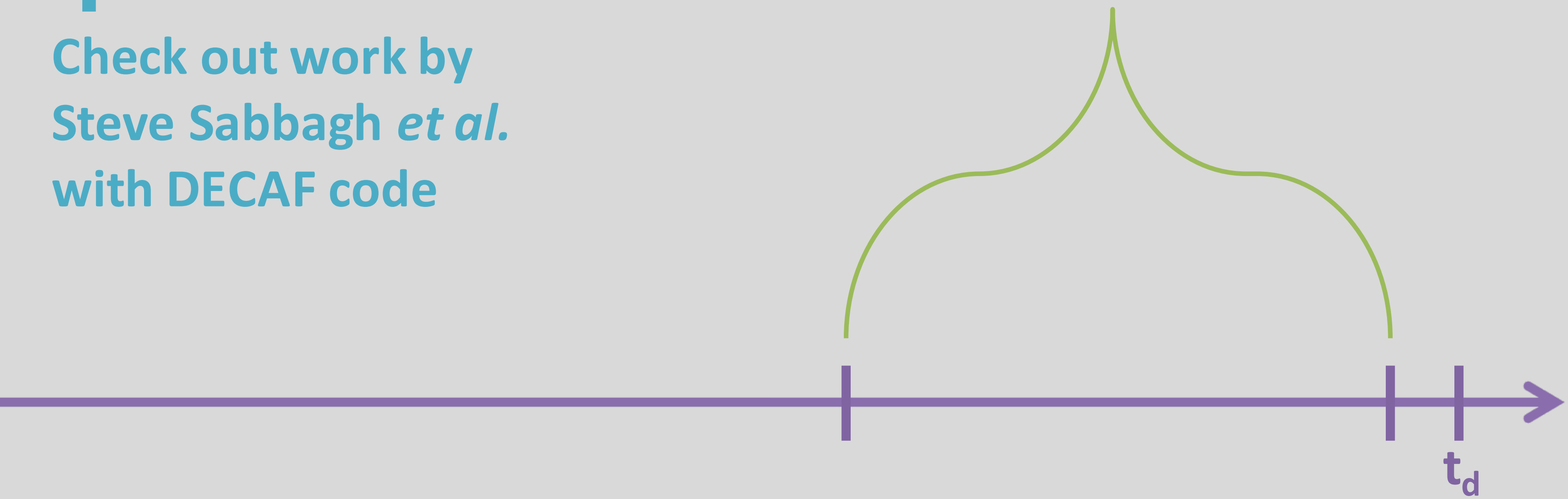
"Steady-State" regime
No clear characteristics of disruptive behavior
Use standard controls



Disruption Prediction Objective

"Disruption Avoidance" regime
Chain of disruption precursors leading to disruption
Use controls to steer discharge away from disruption

Check out work by
Steve Sabbagh *et al.*
with DECAF code



Disruption Prediction Objective

"Disruption Mitigation" regime

Scope of present work → Identify imminent disruption

Radiate energy away to mitigate damage

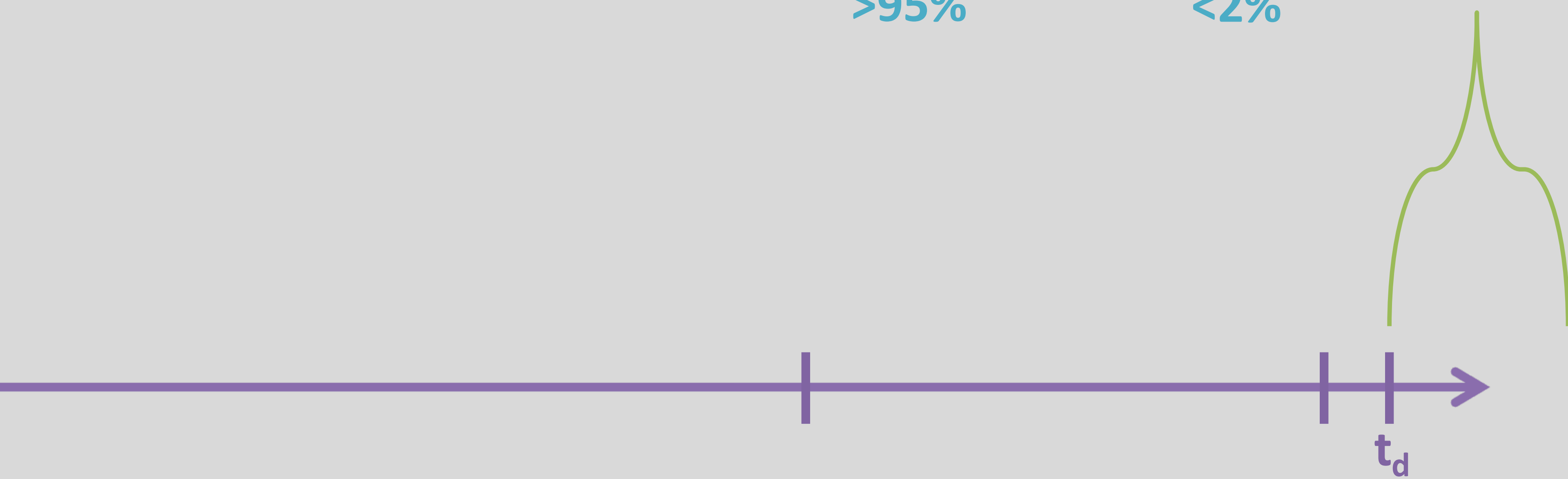
Disruption Prediction Objective

"Buy a new wall for ITER" regime

Cannot have many (any) unmitigated disruptions at high I_p

Need to identify disruptions with high success / low false alarm rates

>95% **<2%**



How can we predict disruptions?

- No comprehensive analytic models for disruptions
  - Need something else to try!
- Tons of data to analyze (~petabyte worldwide)
  - Begs for advanced statistical techniques to be applied!

Machine Learning

- Automated pattern recognition
  - Algorithms build a model based on sample inputs
  - Model is used to interpret new data
- Data-driven analysis is successful in other fields
  - Why not in plasma physics?
- Previous work shows promise for disruption prediction
  - Many opportunities to improve!

Machine Learning Objective

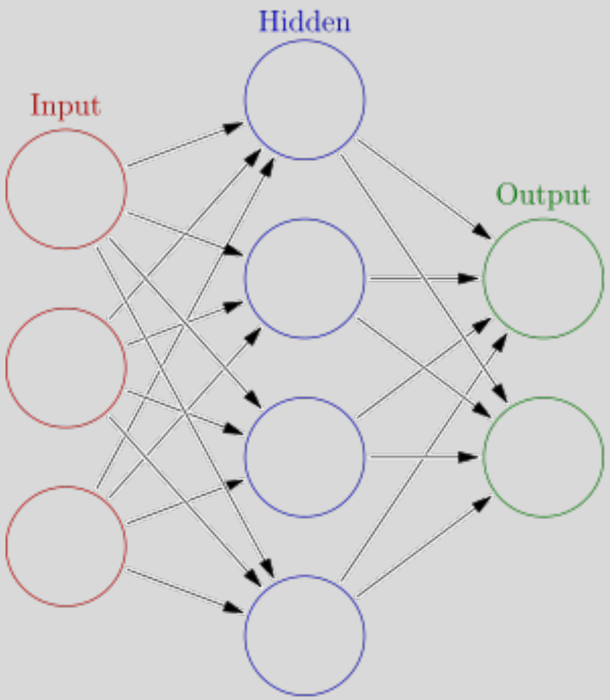
- Binary classification problem
  - Is the plasma Disruptive or Nondisruptive?
  - Is there an imminent disruption to mitigate?

Classification: Learning by Example

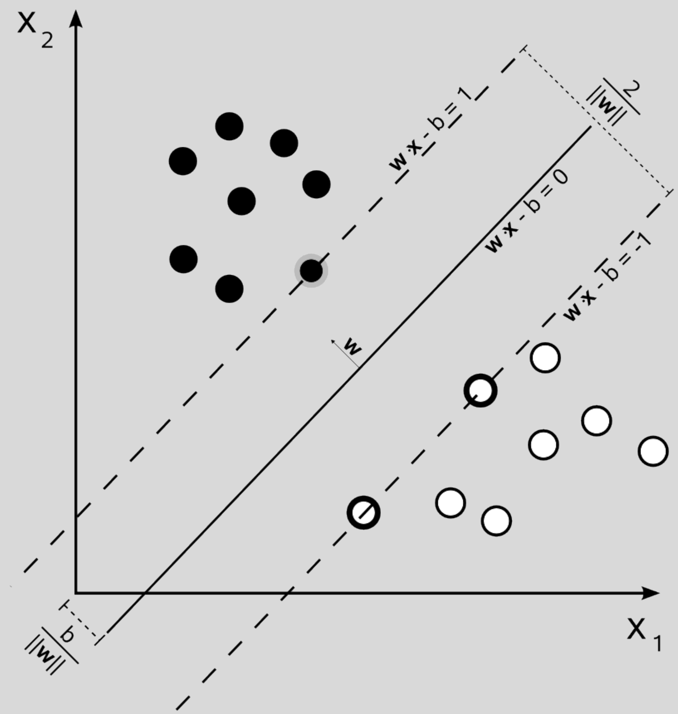
- Machine learning classification is Supervised
  - Define the classes (disruptive, nondisruptive)
  - Use a training set to generate a model
- Contrast to clustering techniques (unsupervised)

Classification: Techniques

Artificial Neural Networks



Support Vector Machines



Outline

- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines

- Disruptor Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?

- Future Pursuits

Classification: Disruption Features

- Neural nets and novelty detection with 0D signals

Input Signals	Neural Network 1	Neural Network 2	Neural Network 3
Mode lock		✓	✓
Plasma Density	✓	✓	✓
Total Input Power	✓	✓	
Total Radiated Power	✓	✓	
Stored Energy Derivative	✓	✓	
Plasma Internal Inductance ℓ_i	✓	✓	✓
Safety Factor q_{95}	✓	✓	✓
Poloidal beta β_p	✓	✓	✓
Radiated/Input Power ratio			✓

Table 5.1: Input variables of the three different network

Classification: Disruption Features

- Neural nets
  - Added I_p to signal set
  - B. Cannas *et al. Nuclear Fusion*, **44** (2004)
- SVM for disruption classification
  - Density limit, H-L transition, Mode lock, Internal transport barrier
  - B. Cannas *et al. Nuclear Fusion*, **46** (2006)
- Neural net predictor / SVM for novelty detection
  - B. Cannas *et al*. *Fusion Engineering and Design*, **82** (2007)
  - B. Cannas *et al. Nuclear Fusion*, **47** (2007)
- Fuzzy logic predictor / regression trees for signal selection
  - A. Murari *et al*. *Nuclear Fusion*, **48** (2008)

Classification: Disruption Features

- SVM for disruption prediction
  - Compared signal means and std(FFT)
  - G.A. Rattá *et al. Review of Scientific Instruments,* **79** (2008)
- Two-tiered SVM's for disruption prediction
  - Used std(FFT) instead of mean
  - G.A. Rattá *et al. Nuclear Fusion,* **50** (2010)
- Genetic algorithms to reduce set signal
  - Used std(FFT) *and* mean
  - G.A. Rattá *et al. Fusion Engineering and Design,* **87** (2012)

Classification: Disruption Features

- APODIS, current set of signals established (JET)

Table 1
List of signals to characterize the disruptive/non-disruptive status of JET plasmas.

Signal name	Units
Plasma current	A
Mode lock amplitude	T
Total input power	W
Plasma internal inductance	
Plasma density	m^{-3}
Stored diamagnetic energy time derivative	W
Radiated power	W

Classification: Feature Vectors

Reduce data set to form feature vectors {$\mathbf{x}_i$, y_i}

$$\mathbf{x} \in \Re^d$$

d = 14

7 signals* x 2 representations

$$y \in \{+1, -1\}$$

{disruptive, nondisruptive}

Signals:
(1) Plasma current [A]
(2) Mode lock amplitude [T]
(3) Plasma density [m^{-2}]
(4) Radiated power [W]
(5) Total input power [W]
(6) d/dt Stored Diamagnetic Energy [W]
(7) Plasma Internal Inductance

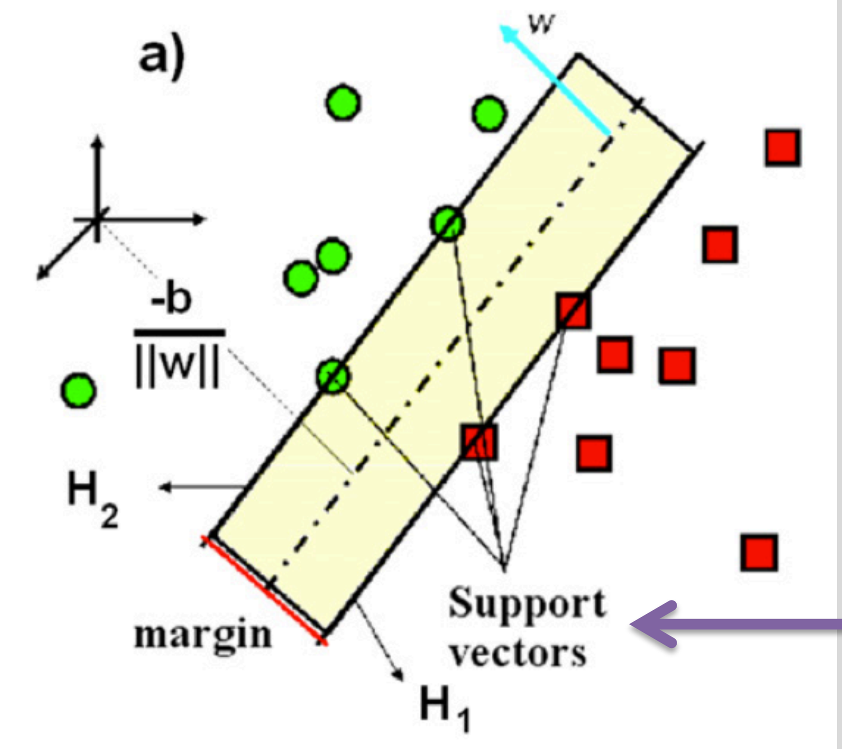
Representations (set of 32 samples at 1 kHz):
(1) Mean
(2) Standard deviation of positive FFT spectrum (excluding first component)

*Each signal normalized to [0,1] over entire data set

Outline

- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines
- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?
- Future Pursuits

Support Vector Machine Paradigm

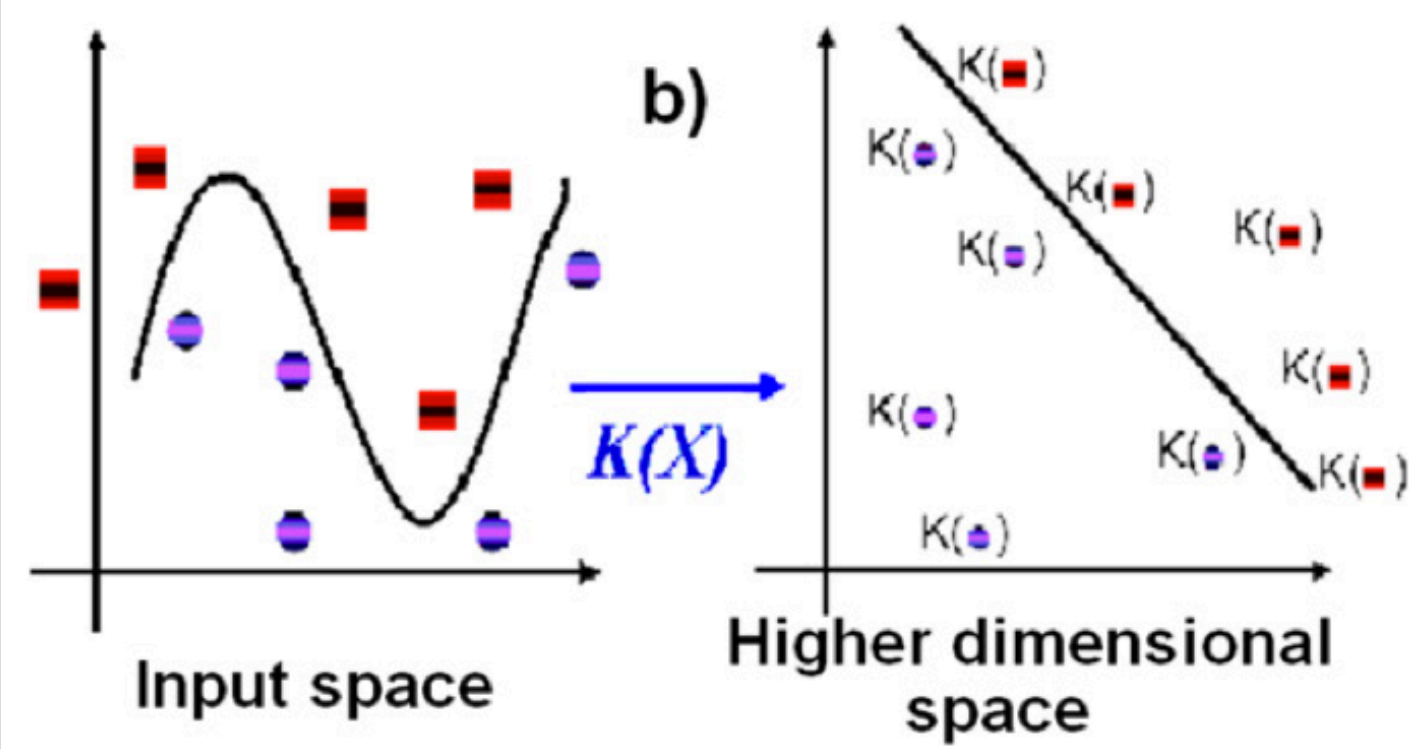


Support Vectors form boundary

Find the hyperplane that separates “disruptive” and “nondisruptive” states with widest possible margin 2/w

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b = 0$$

SVM Nonlinear Reality



Use a function to map the data to a higher dimensional space where it can actually be separated

$$f(\mathbf{x}) = \mathbf{w} \cdot \varphi(\mathbf{x}) + b = 0$$

SVM Decision Function

After solving the optimization problem, classify new data using:

Lagrange Multipliers

Support Vectors

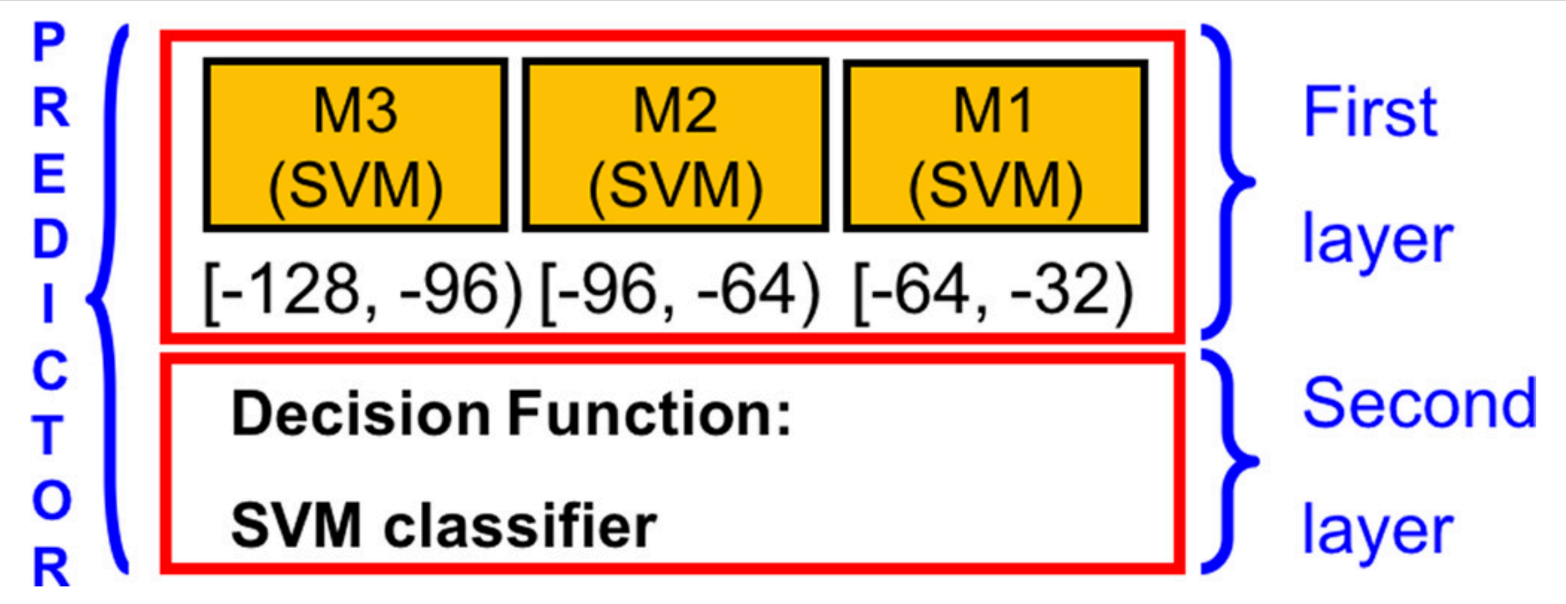
$$f_D(\mathbf{x}) = \sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x})$$

Kernel Function:
Gaussian, Linear

Assess accuracy in terms of:

- Correct predictions
- Missed alarms
- False alarms

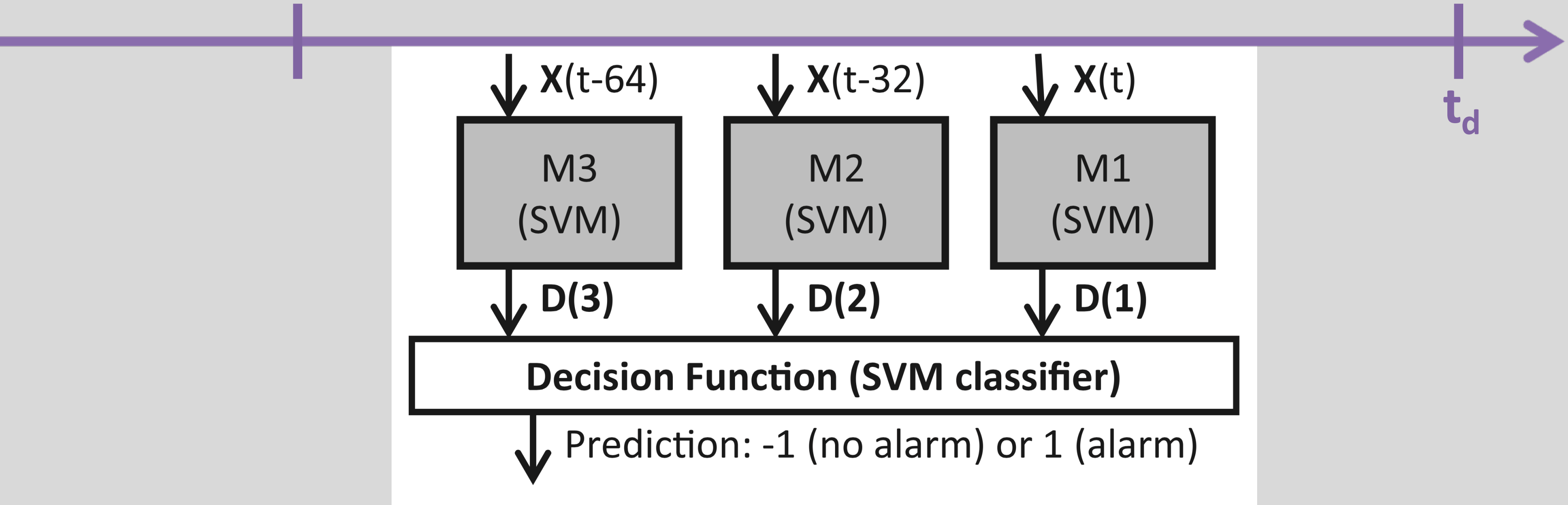
Two-tiered SVM



1st Tier: Three models trained separately, Gaussian kernel

2nd Tier: Trained on combined Tier 1 outputs, linear kernel

Two-tiered SVM



1st Tier: Three models trained separately, Gaussian kernel

2nd Tier: Trained on combined Tier 1 outputs, linear kernel

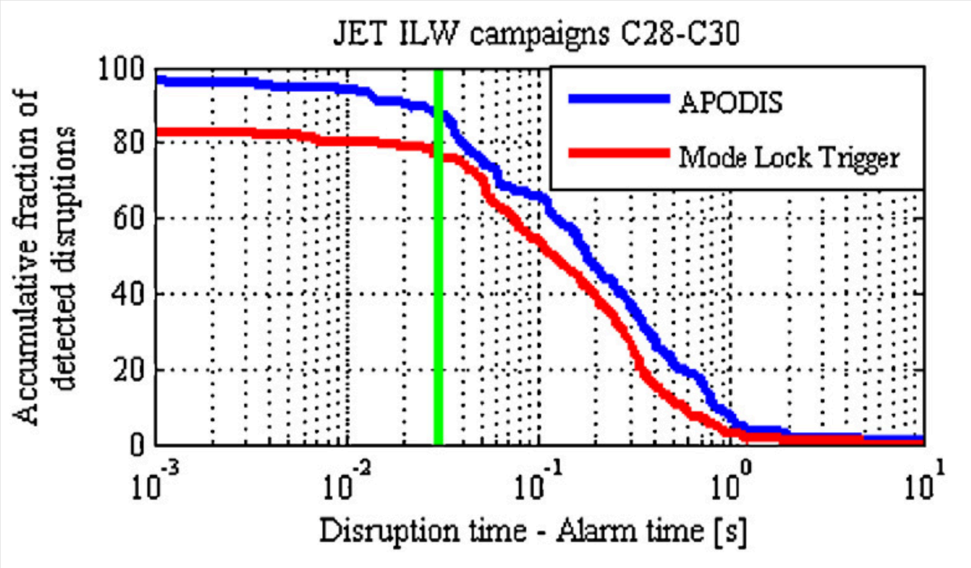
Advanced Predictor of Disruptions

- APODIS is implemented on JET for real-time analysis
  - Model trained on carbon-wall data, still works with Be wall
- Typically used in a passive mode
  - Alarms don’t trigger responses
- Operators can choose to use it as a Disruption Mitigation System trigger
- Occasionally used as a fast camera trigger

APODIS Results (2013)

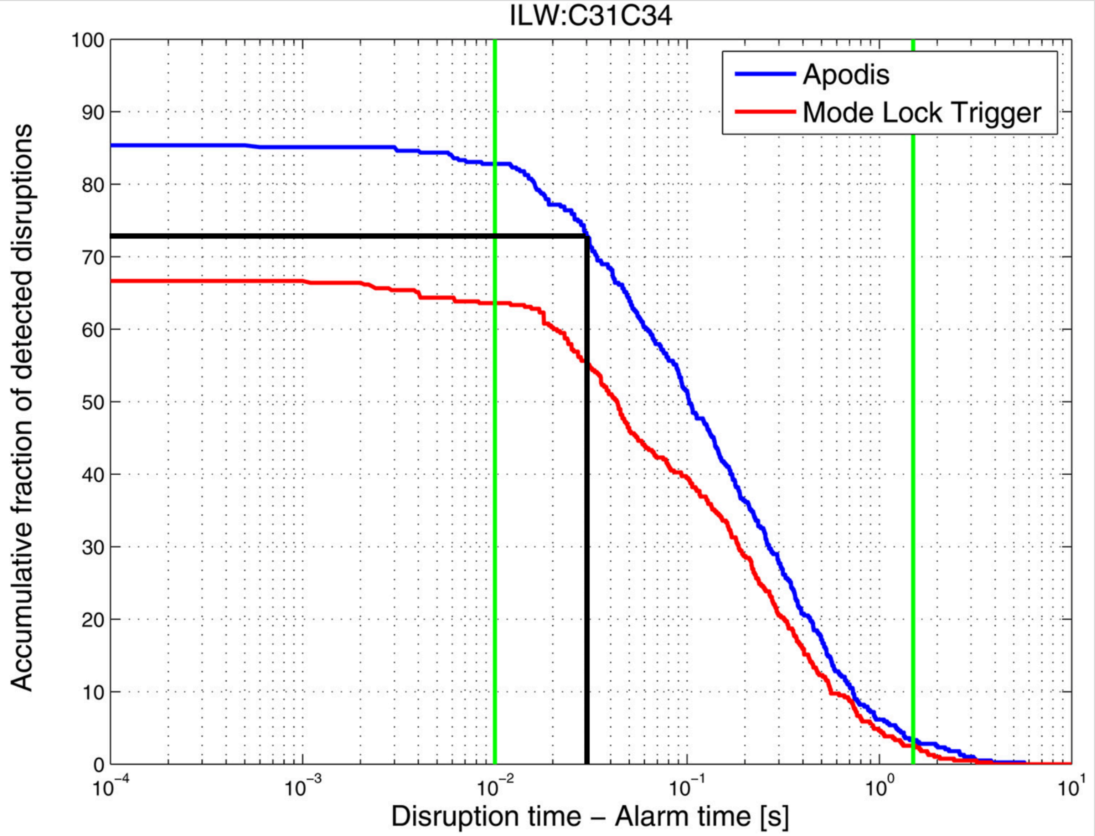
CFC: 93.42% of disruptions identified (213/228)
5.11% of shots give false alarms (183/3578)

ILW: 98.36% of disruptions identified (300/305)
0.92% of shots give false alarms (6/651)



APODIS Results (2016)

ILW: 85.38% of disruptions identified (333/390)
2.46% of shots give false alarms (26/1059)



Outline

- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines
- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?
- Future Pursuits

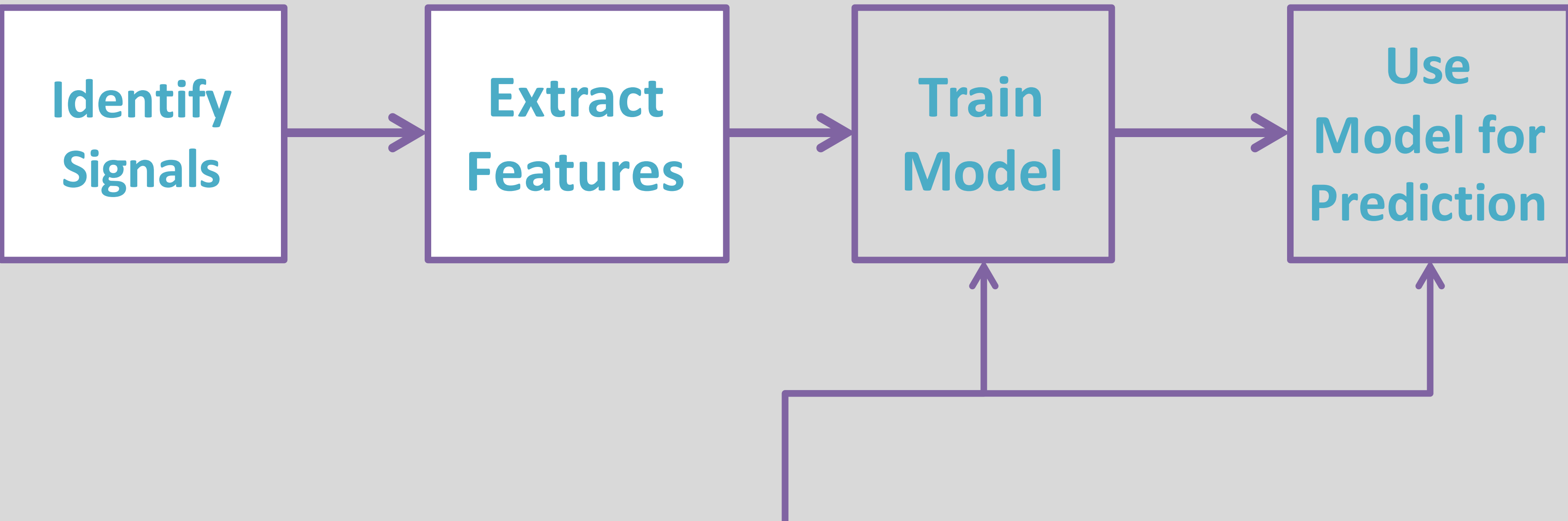
Opportunities and Project Scope

- Develop new features to improve disruption prediction
  - Do multi-dimensional signals give improved performance?
  - If so, use as inputs to threshold tests or other predictors
- Compare different machines to determine predictor portability
  - Find machine-independent normalizations
  - NSTX-U is right down the hall
- Learn about disruption dynamics
  - Similarities to other phenomena? (e.g. L-H transition)
  - Gain ability to identify disruption precursors (e.g. NTMs)
- Possibility of using SVM as a backbone for disruption prediction
  - Train SVM based on outputs of multiple predictors
  - Use SVM in parallel with other predictors

Classification Workflow

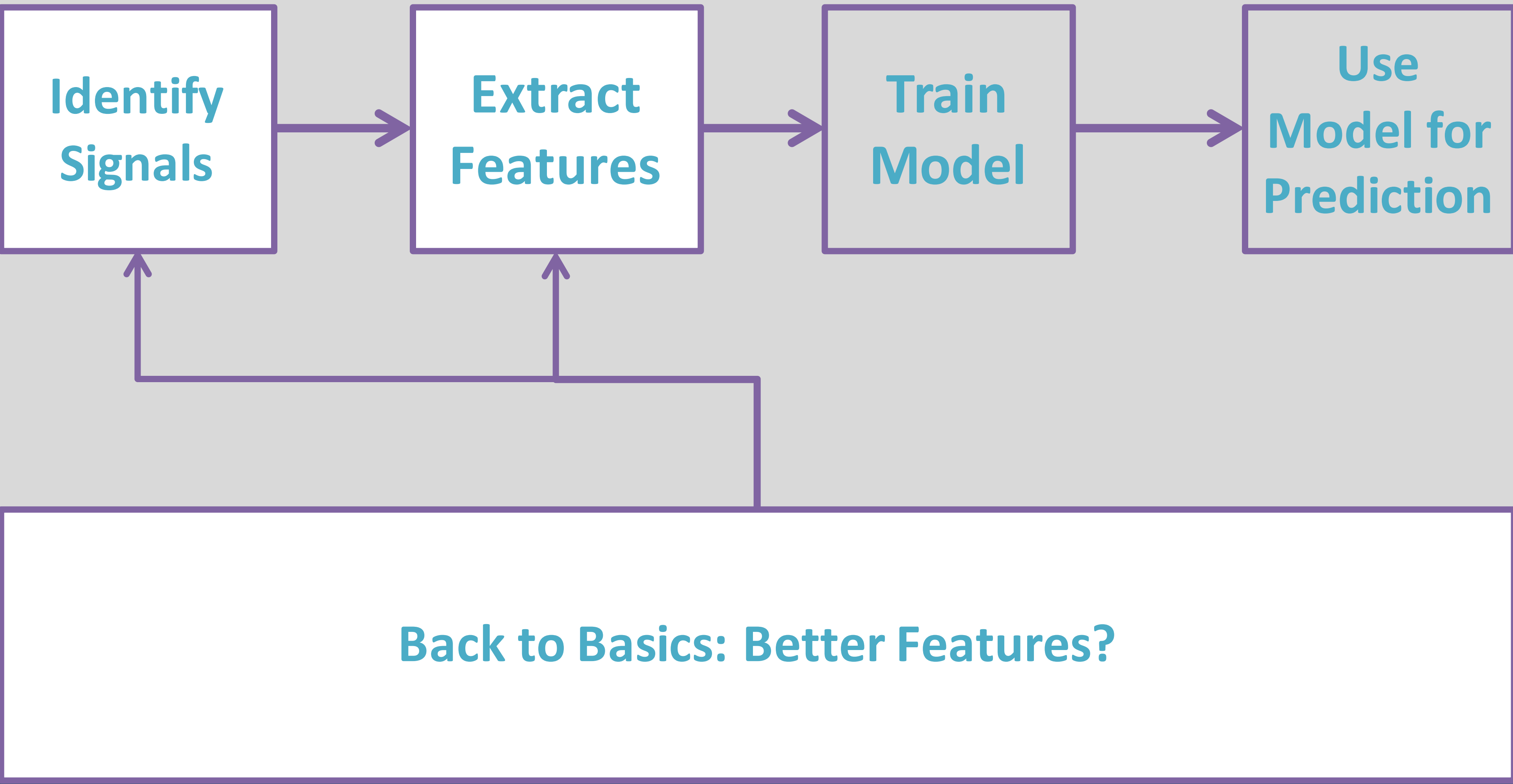
Identify Signals → Extract Features → Train Model → Use Model for Prediction

PPPL Workflow

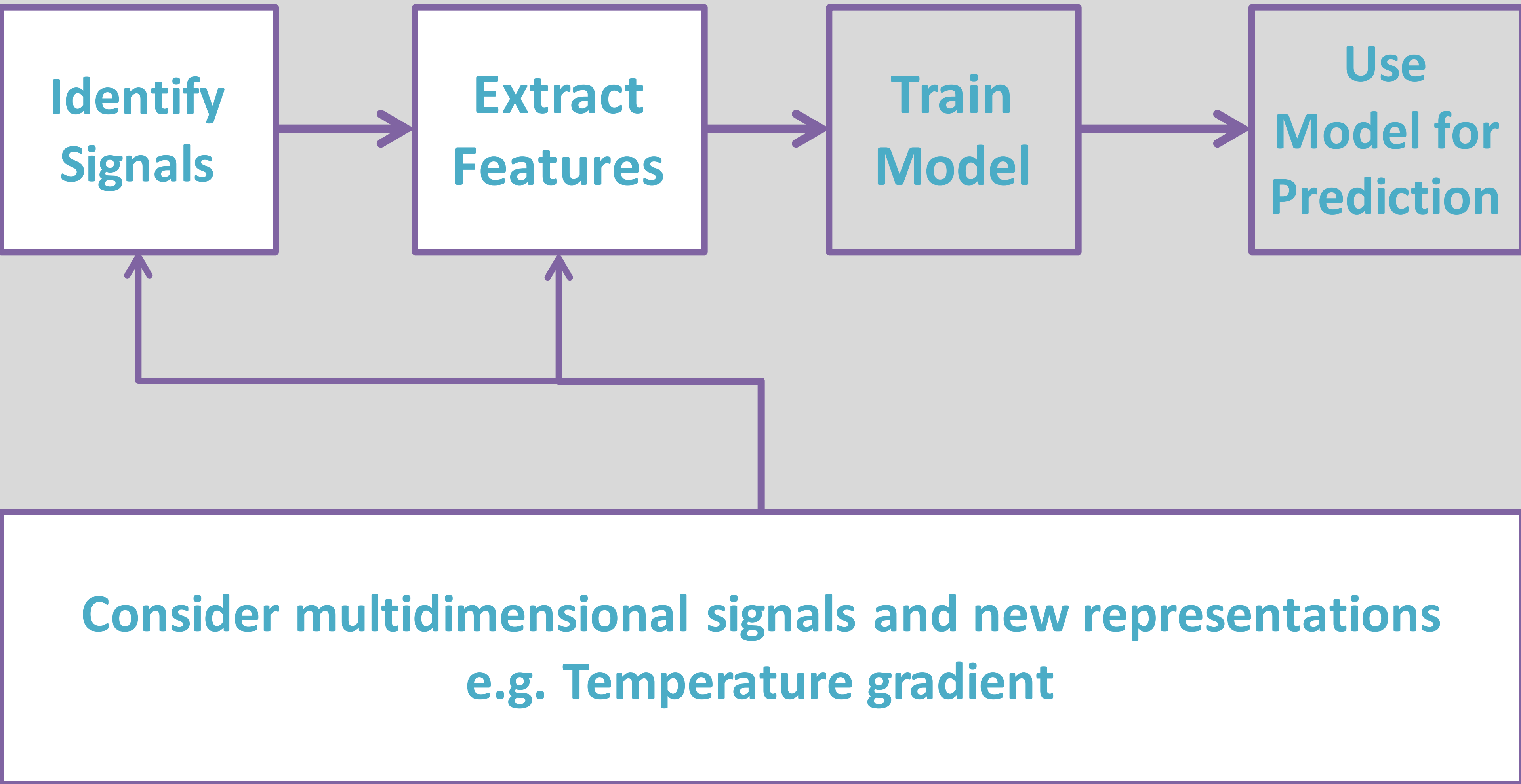


SVMs show a lot of promise,
Offline analysis is good for developing new techniques

PPPL Workflow

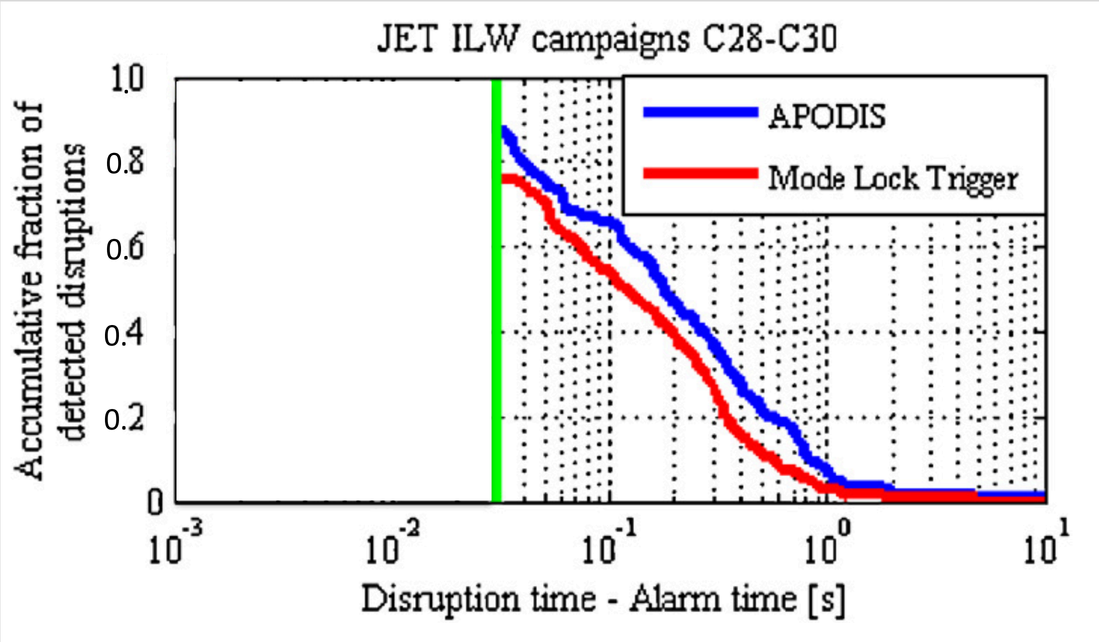


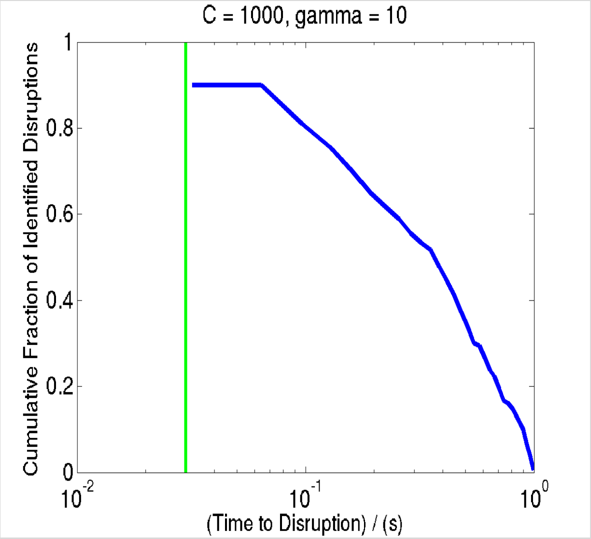
PPPL Workflow



Disruption Predictor Feature Developer

- Began code development in Spring 2015
- **Successfully benchmarked against published APODIS result**
- APODIS and DPFD both predict ~90% of disruptions 30 ms early





Outline

- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines
- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?
- Future Pursuits

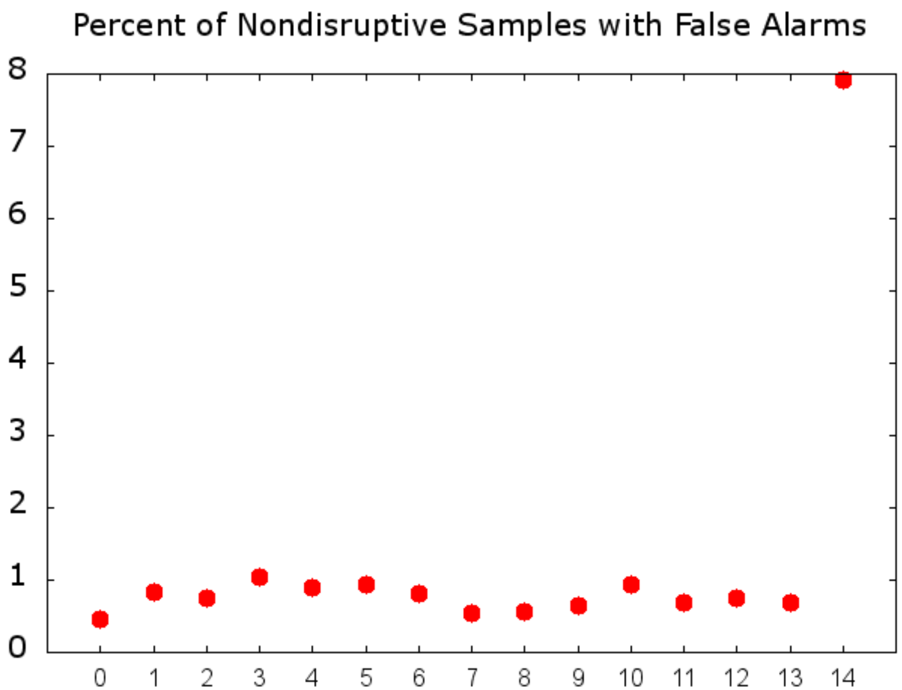
Analysis of Multidimensional Data

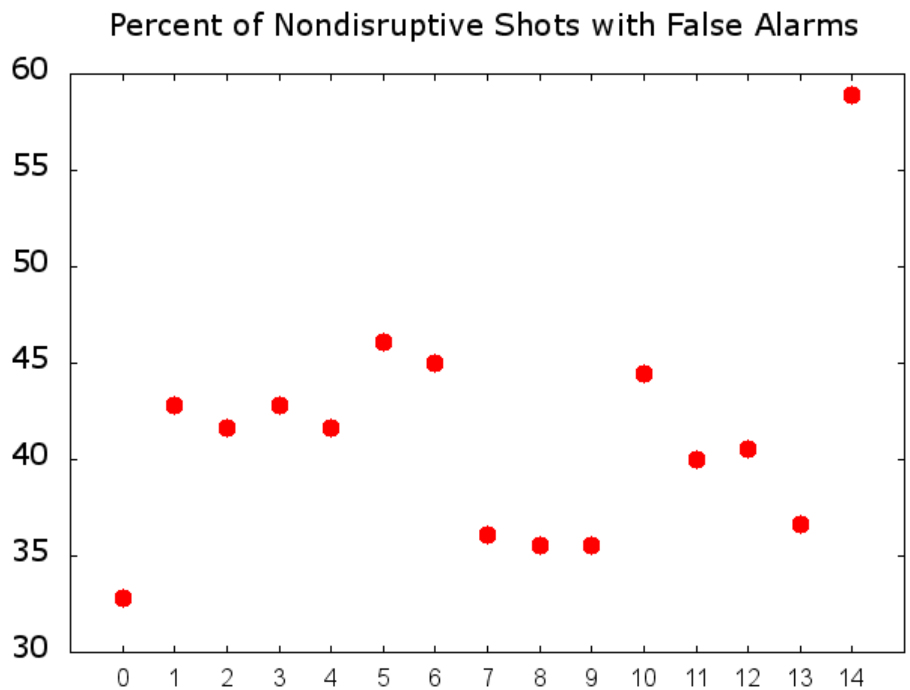
- Electron temperature from JET ECE diagnostic
  - 96 channels measuring different frequencies
  - Some overlap in radial coverage between channels

- Working directly with T_e profiles (not raw data)
  - Select radial domain to use prediction
    - ~98% of shots had data spanning R = [2.9, 3.5] m
      - B_0 = 2.96 m, a ~ 1.25 m (no edge coverage)
  - Interpolate onto grid with resolution 5cm and 10ms
    - T_e at 13 radial positions (every 5cm from 2.9m to 3.5m)

Analysis of Multidimensional Data

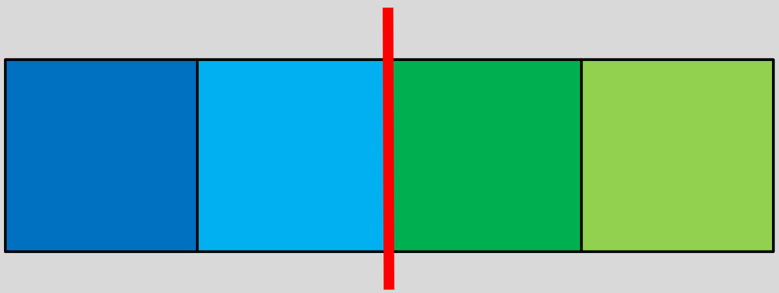
- Using ECE profile (13 pos) causes false alarm rate to spike





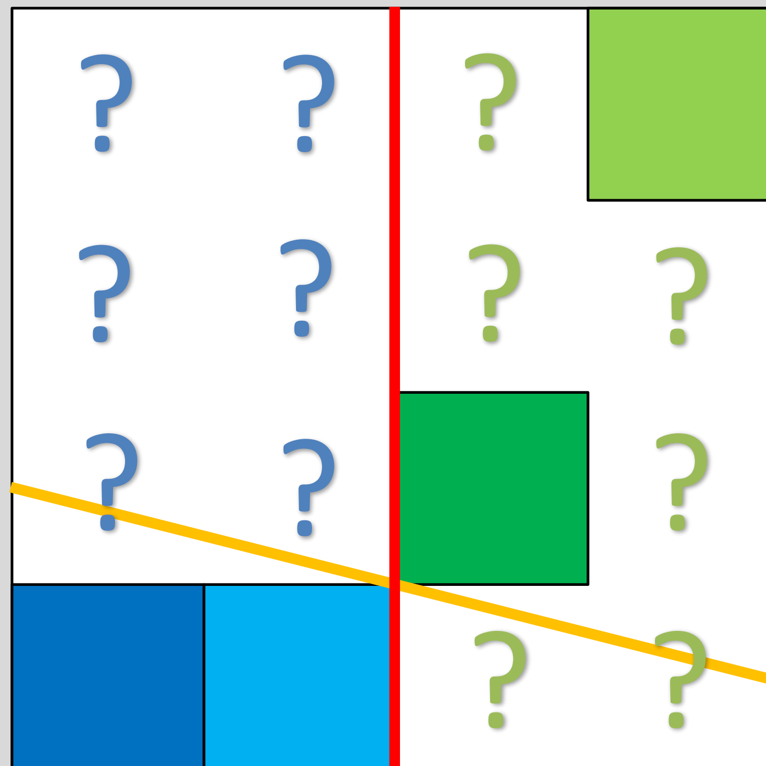
Analysis of Multidimensional Data

- Curse of Dimensionality
  - Sparse data (14d –> 40d) means decreased statistical significance



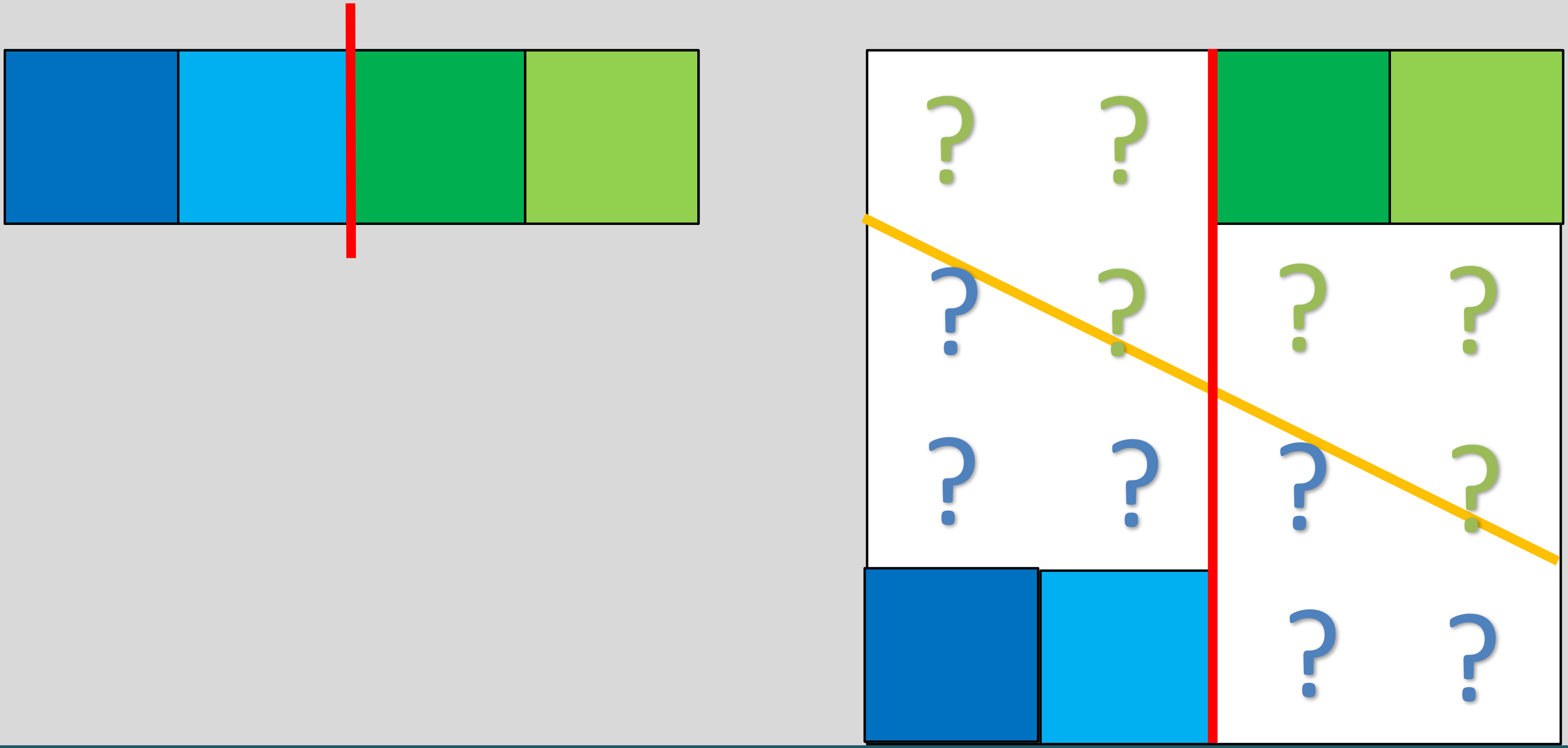
Points cover spanned 1D space

Same points cover smaller portion of spanned 2D space



Analysis of Multidimensional Data

- But more dimensions *could* be helpful, if chosen carefully



Analysis of Multidimensional Data

- To Do:
  - Parameter scan to see if a better model exists
  - Look at different representations of temperature profile
    - Temperature gradient at different radial locations
  - Consider other multidimensional signals (e.g. density)

Outline

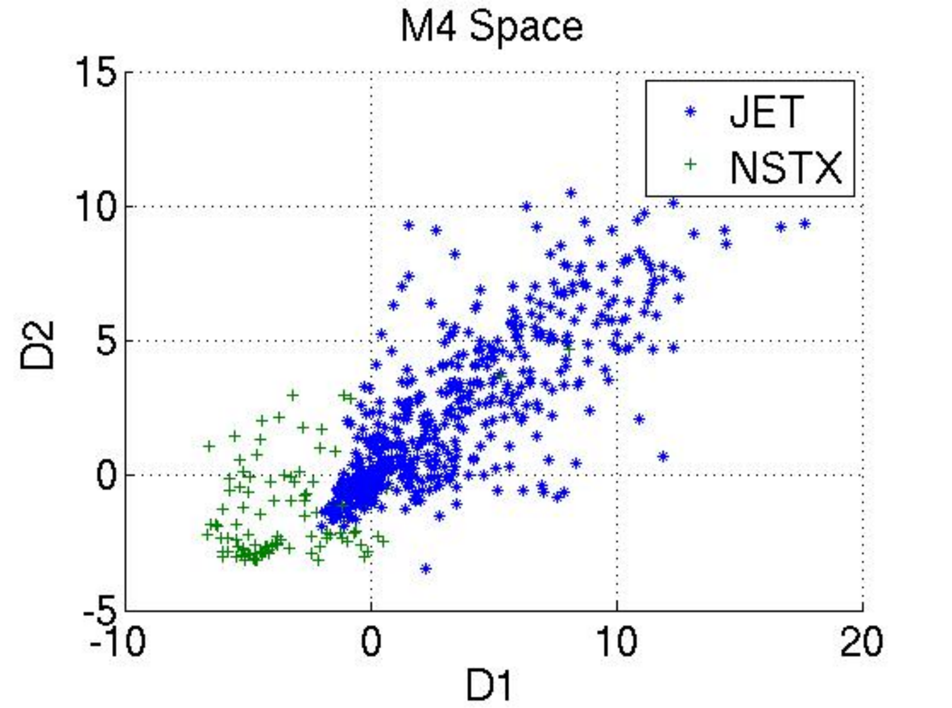
- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines
- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?
- Future Pursuits

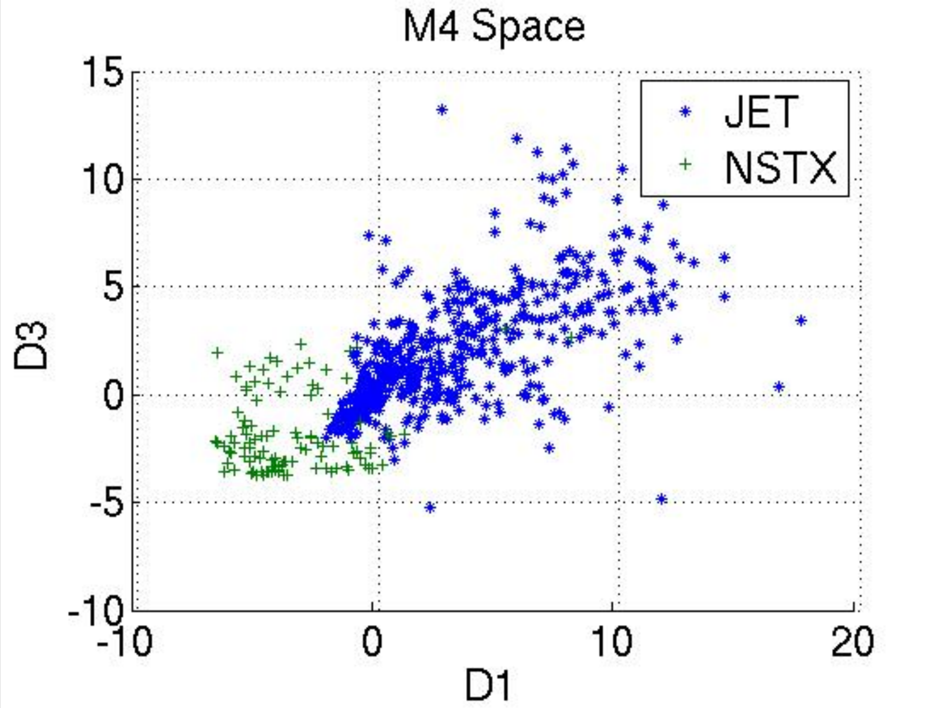
Cross-Machine Analysis

- Starting with old NSTX data
  - Train model with JET (CFC) data
  - Test on NSTX data
    - Use JET normalizations
- Tested 37 NSTX disruptions, only 10 detected
  - All with between 64 and 96 ms of warning

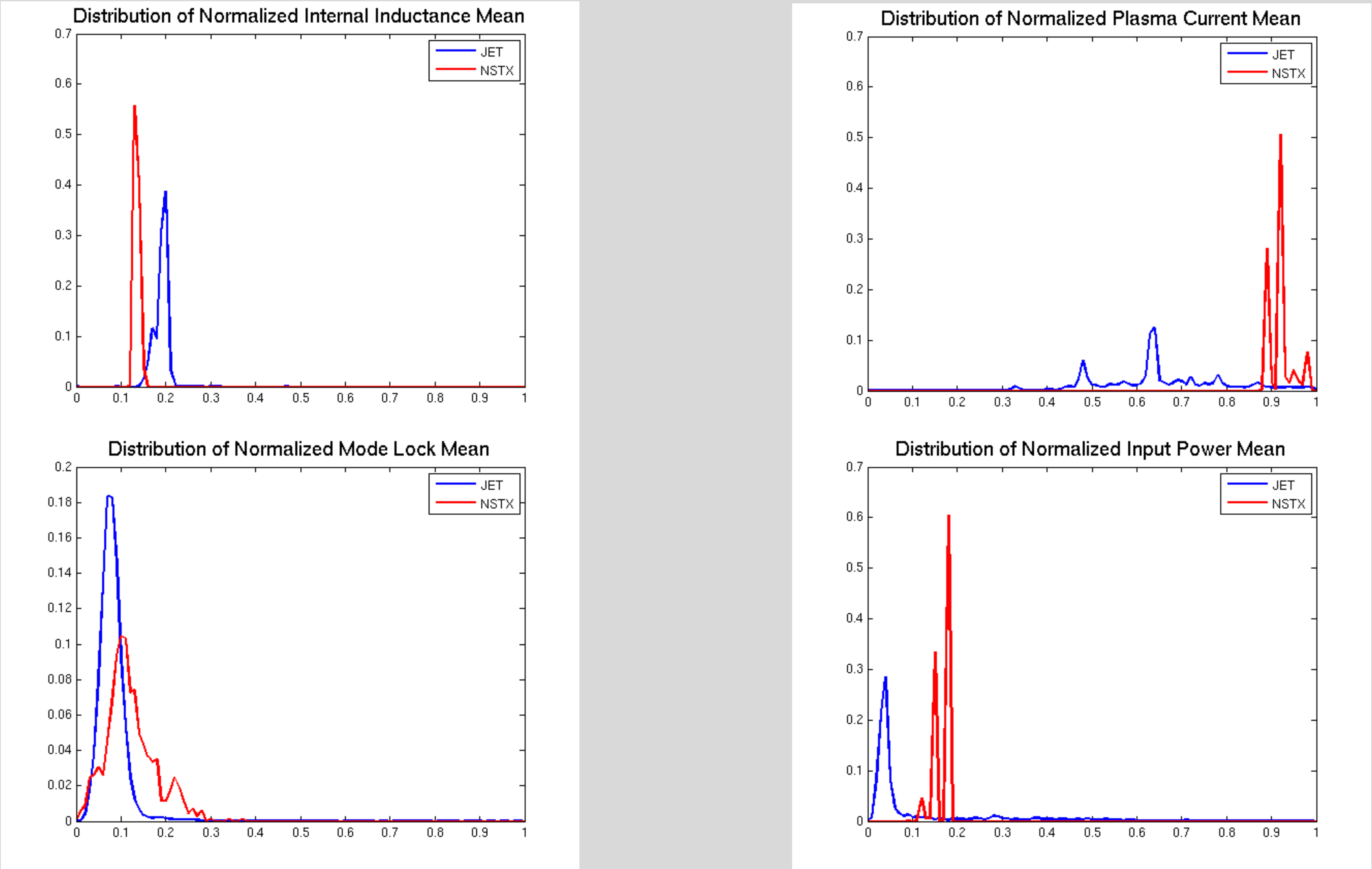
Cross-Machine Analysis

- NSTX is clearly in a different regime

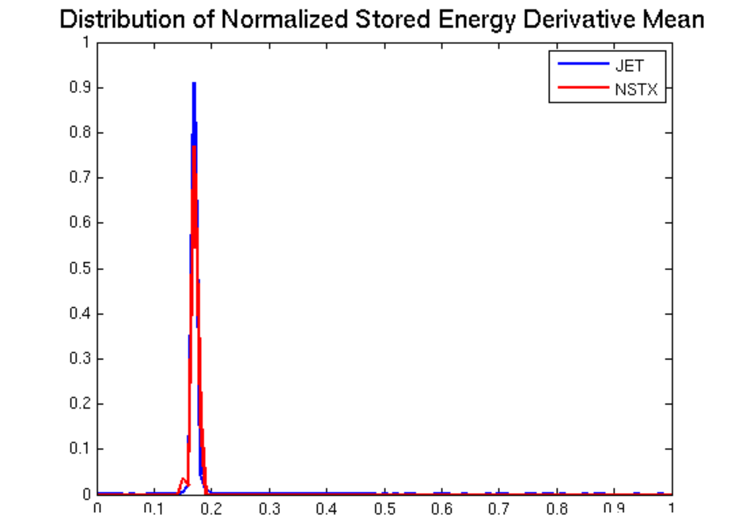


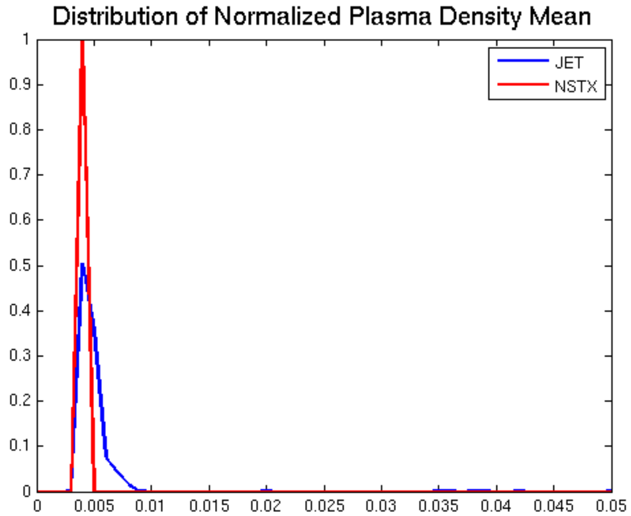


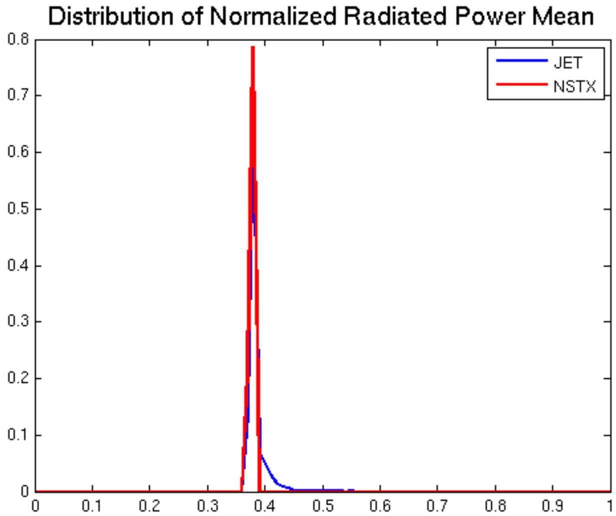
Cross-Machine Analysis



Cross-Machine Analysis







Cross-Machine Analysis

- To Do:
  - Test DIII-D data
    - Different combinations of data for testing/training
  - Consider dimensionless parameters
    - Examine effects on JET data set
    - Examine effects on portability to other machines

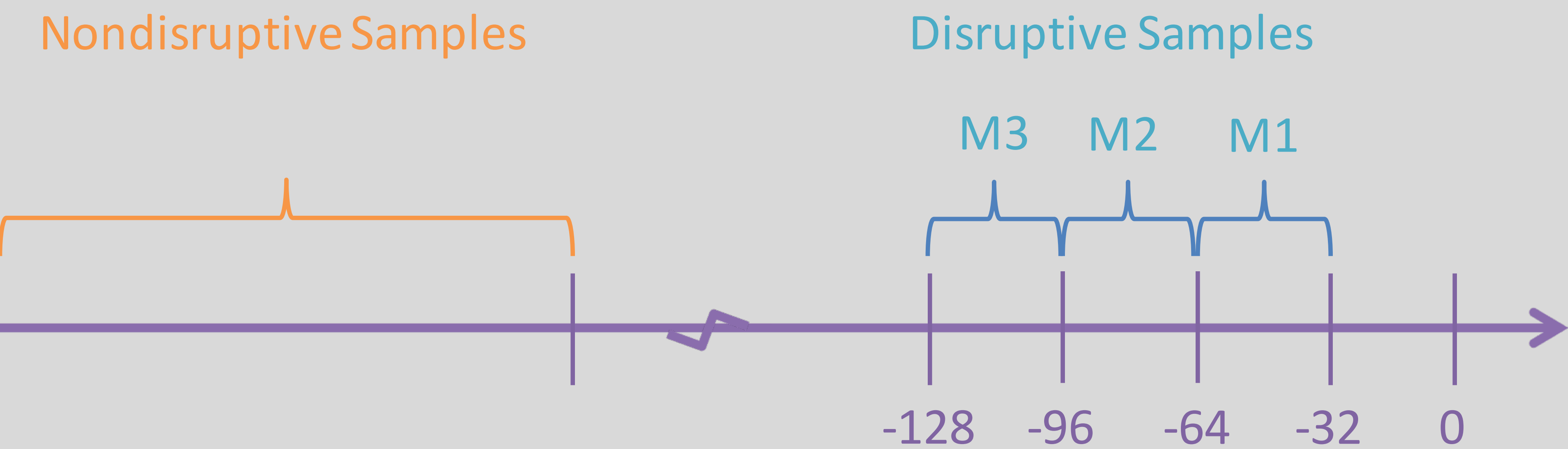
Outline

- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines
- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?
- Future Pursuits

Opportunities and Project Scope

- Develop new features to improve disruption prediction
  - Do multi-dimensional signals give improved performance?
  - If so, use as inputs to threshold tests or other predictors
- Compare different machines to determine predictor portability
  - Find machine-independent normalizations
  - NSTX-U is right down the hall
- Learn about disruption dynamics
  - Similarities to other phenomena? (e.g. L-H transition)
  - Gain ability to identify disruption precursors (e.g. NTMs)
- Possibility of using SVM as a backbone for disruption prediction
  - Train SVM based on outputs of multiple predictors
  - Use SVM in parallel with other predictors

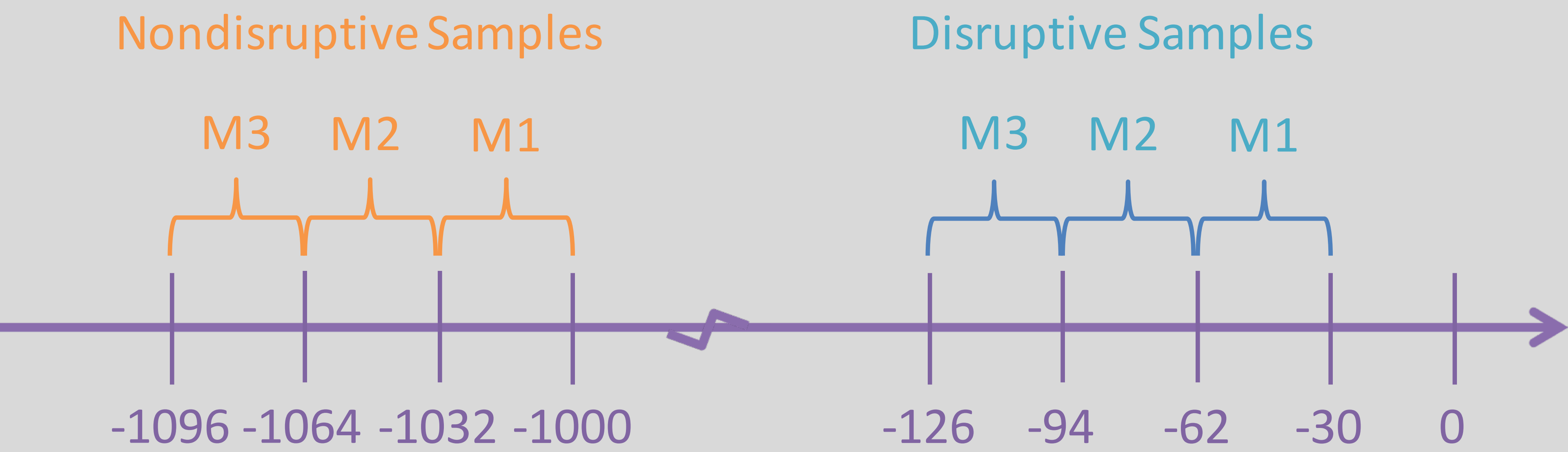
APODIS Training



AND every time interval of every nondisruptive shot

Times given as t - t_d in milliseconds,
where t_d is the beginning of the current quench

DPFD Training

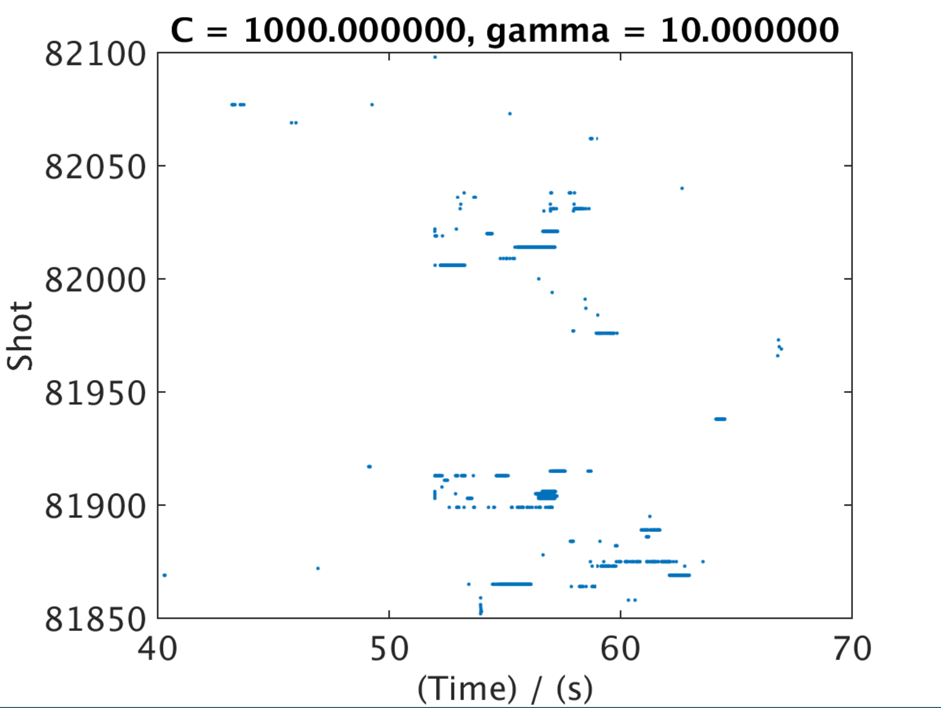


Times given as t - t_d in milliseconds,
where t_d is the beginning of the current quench

Sensitive Predictions

- APODIS trained on 738 dis. and 2,035,000 nondis. samples
- DPFD trained on 894 dis. and 894 nondis. samples

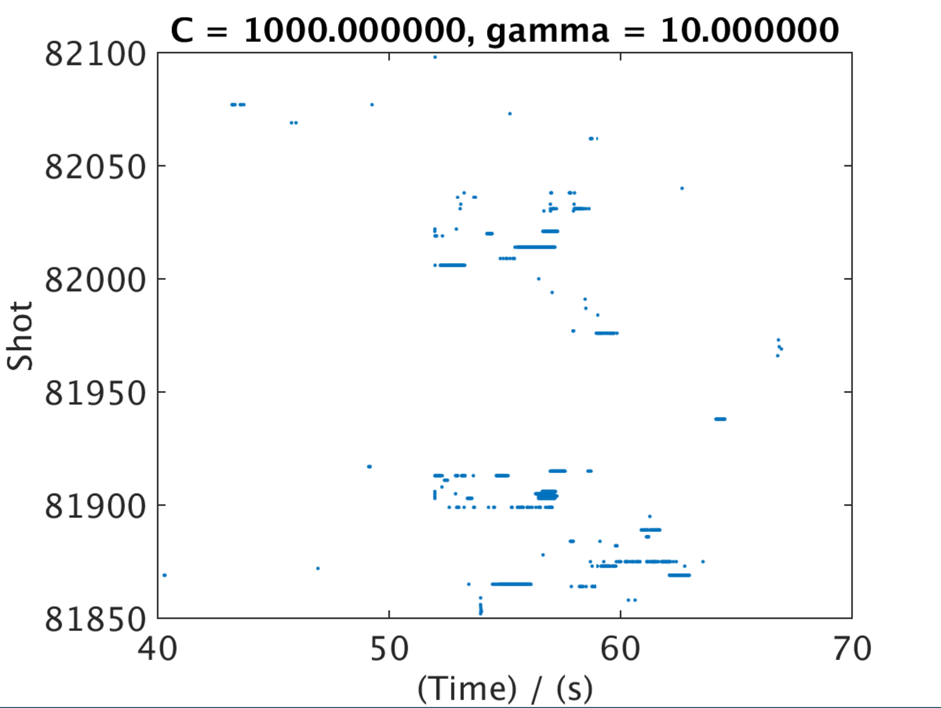
DPFD False alarms, Nondisruptive shots



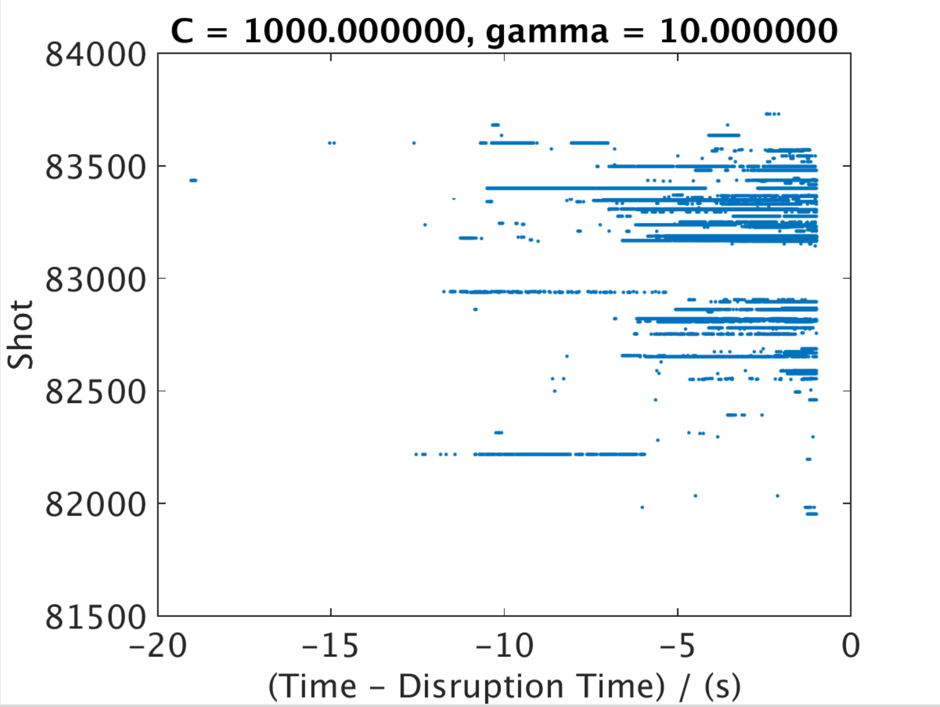
Hidden Physics?

- Something intrinsically different about disruptive shots?
  - Artifact or actual fact?

DPFD False alarms, Nondisruptive shots



DPFD Early alarms, Disruptive shots



Hidden Physics?

- To Do:
  - Evaluate whether early alarms correspond to real, physical characteristics
  - Hypothesis: Instabilities during ramp-up never settle down, plasma never actually reaches steady-state

Outline

- Disruption Prediction by Classification
  - Objective
  - Signal Features
  - Support Vector Machines

- Disruption Predictor Feature Developer
  - Project baseline
  - Analysis of multidimensional data
  - Cross-machine analysis
  - Hidden physics?

- Future Pursuits

Timeline for Future Work

- Now – Two DOE Computational Sciences Graduate Fellowship students arrive to spend the summer working on this project

- August – M. Parsons moves to France to work with ITER's Stability and Control Section on disruption prediction

- October – New Joint Research Target (not on disruptions)


WE WANT
YOU
TO TRY DPFD
www.says-it.com/unclesam/

Opportunities for Collaboration

- Testing your ideas for new disruption features to include
- Analysis of SVM outputs as a state-space for controls
- Identifying requirements to implement a predictor on NSTX-U

DPFD Starter Kit

- DPFD exists as a coherent suite of Matlab scripts that perform all parts of the feature development workflow
  - Data extraction from MDS+
  - Signal normalization
  - Filters to identify portions of shots with usable signals
  - Feature extraction
  - Feature vector setup
  - Training/testing with SVM

- **The DPFD Package is available with documentation for any interested user**

Thank You!

Extras

Classification: Disruption Features

- Genetic algorithms to reduce set signal (JET)
  – Used std(FFT) *and* mean

Table 2
List of employed signals.

Signal name	Abbreviation	Units
1. Plasma current	PC	A
2. Poloidal beta	PB	
3. Poloidal beta time derivative	PB_d	s^{-1}
4. Mode lock amplitude	ML	T
5. Safety factor at 95% of minor radius	Q95	
6. Safety factor at 95% of minor radius time derivative	Q95_d	s^{-1}
7. Total input power	PTOT	W
8. Plasma internal inductance	LI	
9. Plasma internal inductance time derivative	LI_d	s^{-1}
10. Plasma vertical centroid position	PVP	m
11. Plasma density	n	m^{-3}
12. Stored diamagnetic energy time derivative	Wdia	W
13. Net power (total input power *minus* total radiated power)	Pnet	W

SVM Method

Task: Find the hyperplane $f(\mathbf{x}) = \mathbf{w} \cdot \varphi(\mathbf{x}) + b = 0$

Method: Solve dual problem

$$\text{maximize: } \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K_{ij}$$

Subject to Lagrange multiplier constraints:

$$\sum_i y_i \alpha_i = 0 \qquad 0 \leq \alpha_i$$

SVM Method (Soft-Margin)

Task: Find the hyperplane

$$f(\mathbf{x}) = \mathbf{w} \cdot \varphi(\mathbf{x}) + b = 0$$

Method: Solve dual problem

$$\text{maximize: } \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K_{ij}$$

Subject to Lagrange multiplier constraints:

$$\sum_i y_i \alpha_i = 0$$

$$\boxed{0 \leq \alpha_i \leq \frac{C}{N}}$$

Allows points inside margin, but penalizes them

C – box regularization parameter, *N* – # points

SVM Kernel

Task: Find the hyperplane

$$f(\mathbf{x}) = \mathbf{w} \cdot \varphi(\mathbf{x}) + b = 0$$

Method: Solve dual problem

$$\text{maximize:} \quad \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \boxed{K_{ij}}$$

Kernel:

$$\boxed{K_{ij}} = K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j)$$

But how do we get the mapping function?!

SVM Kernel Trick

Kernel:

$$K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) = \boxed{\varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j)}$$

Must satisfy symmetry, linearity and positive-definiteness of an <u>inner product space</u>

Trick: Use these properties to define a kernel function to avoid ever having to specify the mapping function!

$$\text{Linear: } K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j$$

$$\text{RBF: } K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma|\mathbf{x}_i - \mathbf{x}_j|^2)$$

Multi-Machine Analysis

- NSTX signals used:
  - Engineering / ip1
  - Operations / rwmef_plas_n1_amp_br
  - Passivespec / bolom_totpwr
  - Activespec / ts_ld
  - Efit02 / li
  - Efit02 / wpdot
  - Nbi / nb_p_inj