Confinement mode analysis of fusion plasmas on Alcator C-Mod utilizing machine learning methods

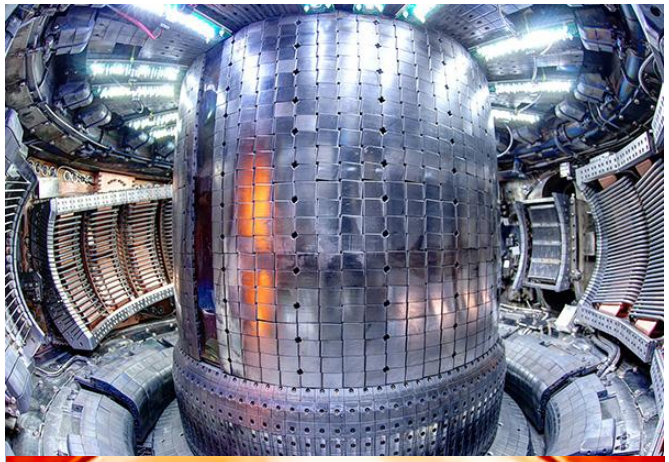
A. Mathews, J.W. Hughes, S.M. Wolfe, A.E. Hubbard, R.S. Granetz, C. Rea, D. Brunner, T. Golfinopoulos, and the Alcator C-Mod Team

Massachusetts Institute of Technology
Plasma Science and Fusion Center

August 14, 2018

Alcator C-Mod

- Compact, high-magnetic field, diverted tokamak
- $R_0 = 0.68$m, $a_0 = 0.22$m
- Primary auxiliary heating source via minority heating (ICRF)
- Record for plasma pressure in magnetically confined fusion



SPARC

- Soonest/Smallest Private-Funded Affordable Robust Compact
- Collaboration between MIT and Commonwealth Fusion Systems (CFS)
- Goal: $Q > 1$ and about 100 MW of heat for 10-second pulses
- HTS magnets will be key

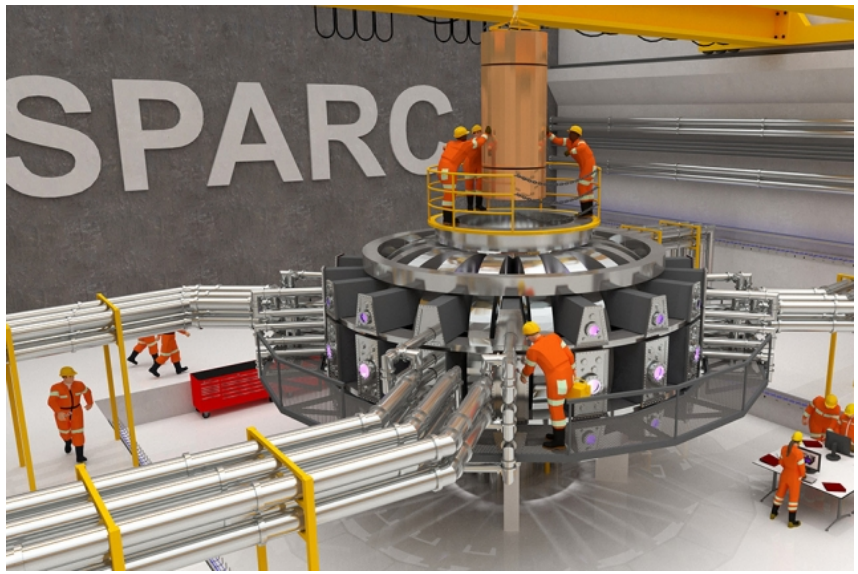


Figure 1: Visualization by Ken Filar, PSFC research affiliate

Confinement regimes

$$\frac{dS}{dt} = -P_{loss} + P_{source}$$
$$= -\frac{S}{\tau} + P_{source}$$

- $\tau_E \rightarrow$ energy confinement time
- $\tau_P \rightarrow$ particle confinement time
- L-mode:
  $\tau_{E,L} = 0.048(I/MA)^{0.85} B_t^{0.2} R^{1.2} a^{0.3}$
  $(e/10^{20})^{0.1} \kappa^{0.5} (P_{tot}/MW)^{-0.5} (m_i/m_p)^{0.5} s$
- H-mode: $\tau_{E,H} = H \cdot \tau_{E,L}$ where $H \sim 2$ and larger τ_P
- I-mode: L-mode's τ_P with H-mode's τ_E

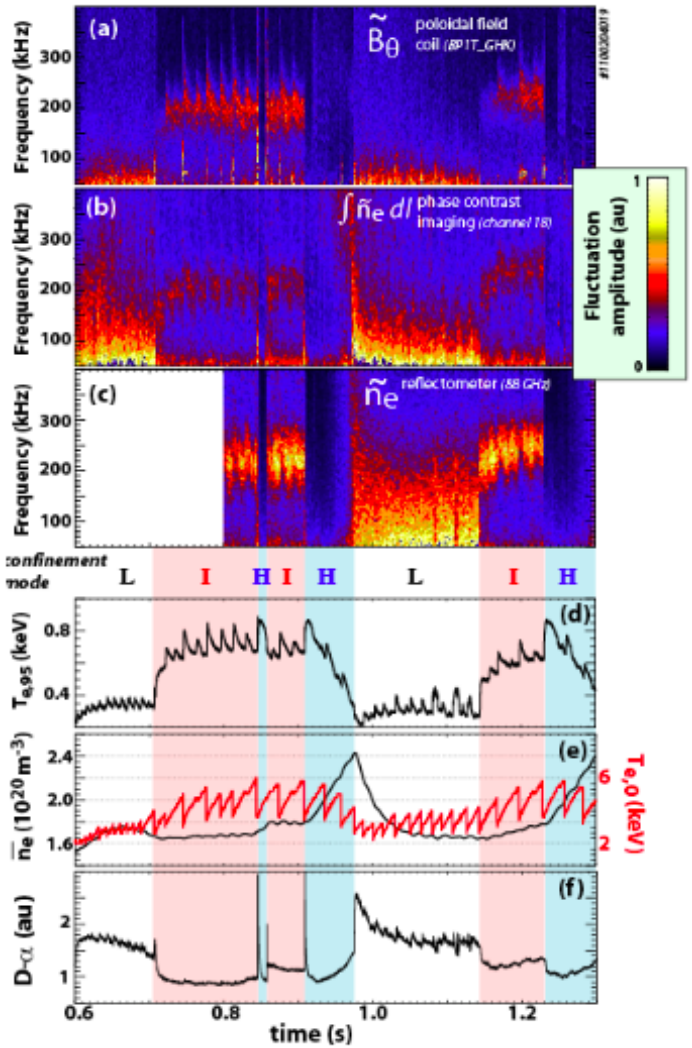


Figure 2: D.G. Whyte *et al* 2010 *Nucl. Fusion* **50** 105005

Confinement regimes

$$\frac{dS}{dt} = -P_{loss} + P_{source}$$
$$= -\frac{S}{\tau} + P_{source}$$

- $\tau_E \rightarrow$ energy confinement time
- $\tau_P \rightarrow$ particle confinement time
- L-mode:
  $\tau_{E,L} = 0.048(I/MA)^{0.85} B_t^{0.2} R^{1.2} a^{0.3}$
  $(_e/10^{20})^{0.1} \kappa^{0.5} (P_{tot}/MW)^{-0.5} (m_i/m_p)^{0.5}$
- H-mode: $\tau_{E,H} = H \cdot \tau_{E,L}$ where $H \sim 2$ and larger τ_P
- I-mode: L-mode's τ_P with H-mode's τ_E

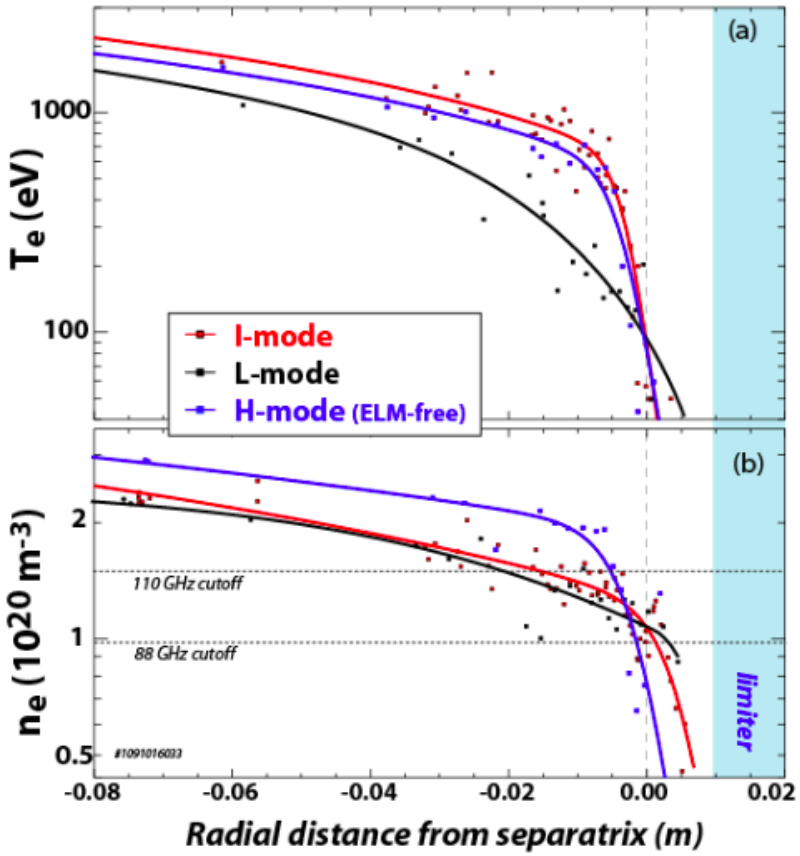


Figure 3: D.G. Whyte *et al* 2010 *Nucl. Fusion* **50** 105005

Exploring machine learning methods

- Further understanding of confinement regime boundaries
- Control tokamak performance in current and future devices
- Large-scale comparative analysis via instant mode identification

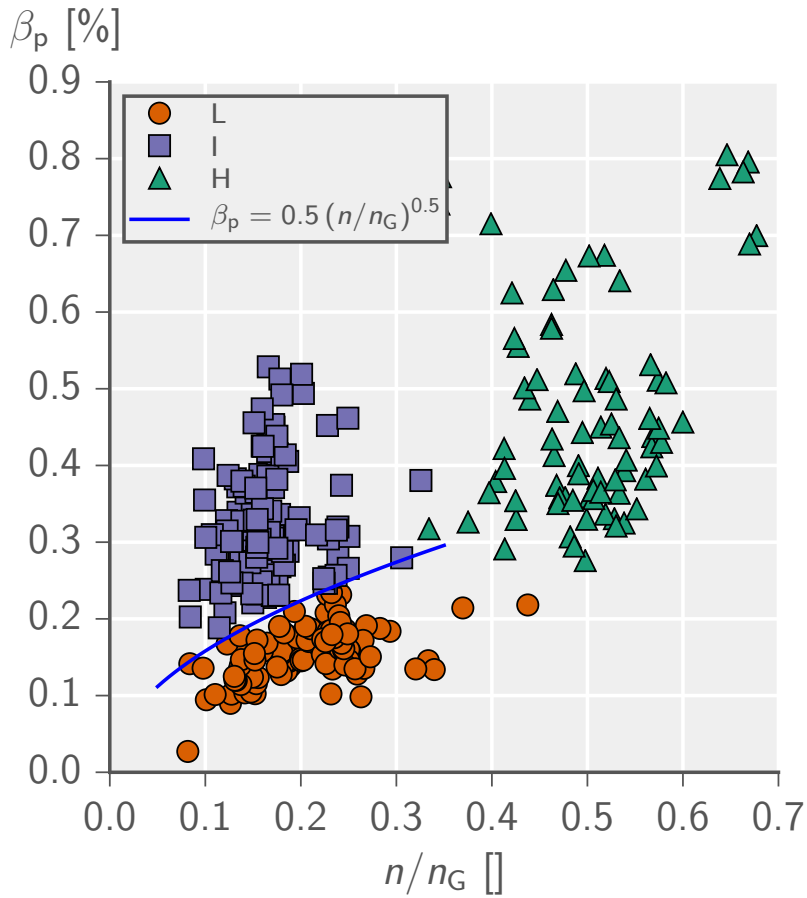


Figure 4: Plot by Dan Brunner

Analyzing L-, H-, and I-modes

- Database based on Alcator C-Mod includes shots dating back to 1995
- Classifications derived from multiple sources (almost all I-modes based on original identification by A. Hubbard or D. Whyte)
- Over 200 distinct shots consisting of approximately 400 L-, 200 H-, and 100 I-mode periods
- Quantity such as energy confinement time and/or pedestal height could add quantitative performance measure and automate database growth
- Both multi-class classification and regression tasks involved

Supervised learning techniques

Classification

- Gaussian naïve Bayes, Logistic Regression, Multilayer Perceptron, and Random Forest

Regression

- Elastic Net, k-Nearest Neighbours, Multilayer Perceptron, and Random Forest

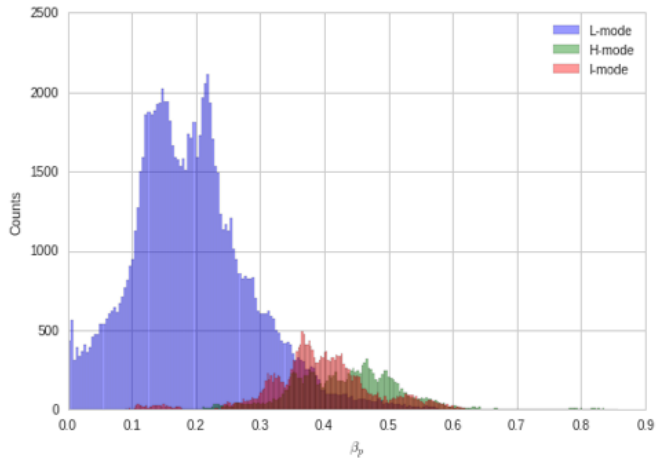
Features

- $\beta_p, \bar{n}, l_i, P_{tot}$
- Currently 0-D data
- Profiles and time-dependency to be considered

Supervised learning techniques: classification

Gaussian naïve Bayes

- Conditional probability model to classify an input feature vector $\mathbf{x} = (x_1, \ldots, x_n)$ into class C_k
- posterior $= \dfrac{\text{prior} \times \text{likelihood}}{\text{evidence}} \Leftrightarrow p(C_k \mid \mathbf{x}) = \dfrac{p(C_k)\, p(\mathbf{x}|C_k)}{p(\mathbf{x})}$
- Gaussian: $p(x_i|C_k) = \dfrac{1}{2\pi\sigma_k^2} \exp[-\dfrac{(x_i-\mu_k)^2}{2\sigma_k^2}]$
- naïve: $p(C_k \mid x_1, \ldots, x_n) \propto p(C_k)p(x_1 \mid C_k)p(x_2 \mid C_k) \cdots p(x_n \mid C_k)$
- Decision rule: $\hat{y} = \underset{k \in \{1,\ldots,K\}}{\text{argmax}}\ p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$



Supervised learning techniques: classification

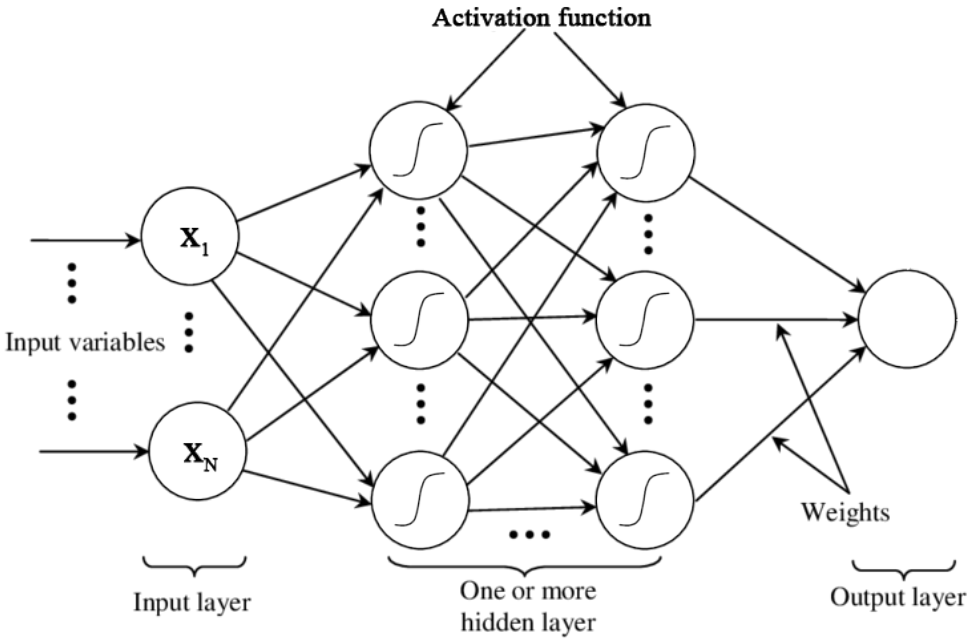
Logistic Regression

- Binary outcome y_n equals 0 or 1 (true distribution)
- $p_n = \frac{1}{1+e^{-(\beta_0+\beta \mathbf{x}_n)}}$ (estimated distribution)
- Cost: $-\frac{1}{N}\sum_{n=1}^{N}\left[y_n \log p_n + (1-y_n)\log(1-p_n)\right]$ with L2 regularization
- Known as logistic loss or cross-entropy
- Odds: $\frac{p(\mathbf{x})}{1-p(\mathbf{x})} = e^{\beta_0+\beta \mathbf{x}}$
- Odds ratio: $\frac{\text{odds}(x_i+1)}{\text{odds}(x_i)} = \frac{e^{\beta_i(x_i+1)}}{e^{\beta_i x_i}} = e^{\beta_i}$
- Physical meaning: odds multiply by e^{β_i} for every 1-unit increase in x_i

Supervised learning techniques: classification

Mutlilayer Perceptron

- Feedforward artificial neural network
- Fully connected with $100 \times 100 \times 100$ hidden layers
- Universal function approximator (Cybenko's theorem)
- Rectified linear unit (ReLU) activation function: $f(z_i) = \max(0, z_i)$
- Optimizes logistic loss function via stochastic gradient descent
- Number of neurons in output layer equals number of classes



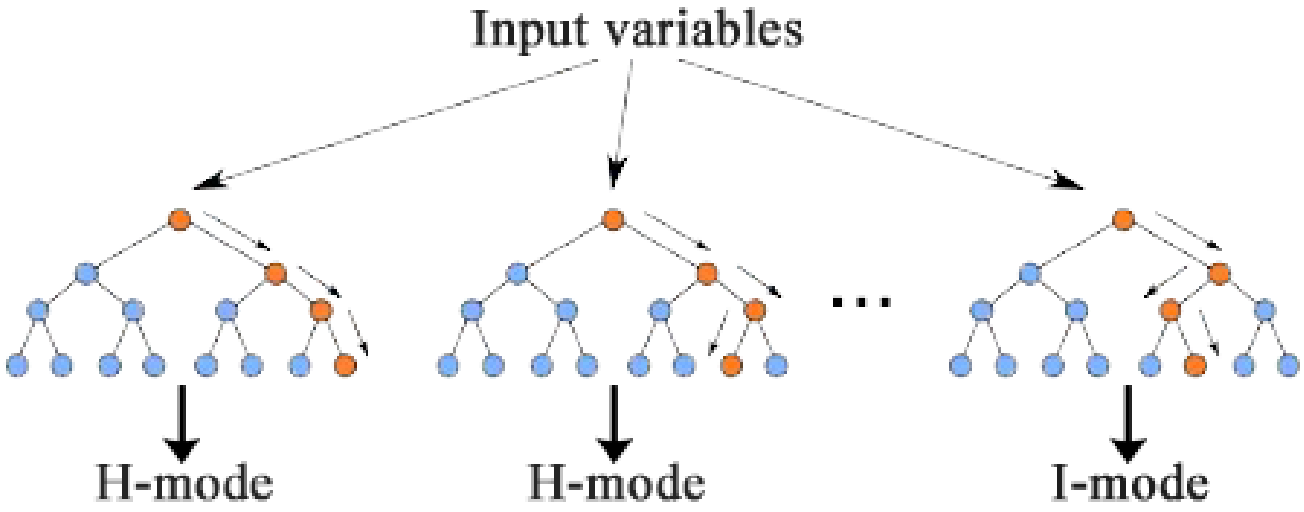
Supervised learning techniques: classification

Random Forest

- Ensemble algorithm using subset of features to divide samples
- Minimize an impurity measure each split, e.g. Gini impurity:

$$I_G(p) = \sum_{i=1}^{K} p_i(1 - p_i)$$

- 100 fully grown decision trees (i.e. all leaves pure)
- Reduces overfitting via bootstrap aggregation and pruning



Feature selection

1. β_p (0.396 ± 0.059)
2. P_{tot} (0.242 ± 0.062)
3. $\bar{n}$ (0.223 ± 0.028)
4. l_i (0.139 ± 0.022)

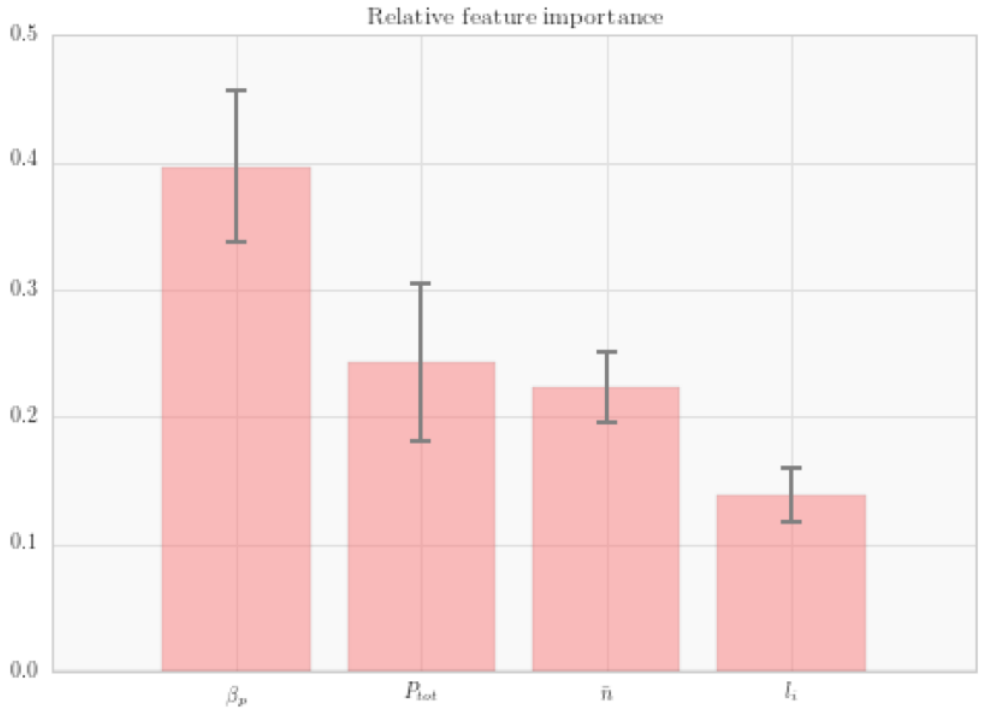


Figure 5: Relative feature importance based on mean decrease impurity via random forest

Accuracy metrics

$$\text{TPR} = \frac{\text{TP}}{\text{TP+FN}} \text{ (sensitivity or recall)}$$

$$\text{TNR} = \frac{\text{TN}}{\text{TN+FP}} \text{ (specificity)}$$

$$\text{PPV} = \frac{\text{TP}}{\text{TP+FP}} \text{ (precision)}$$

$$\text{NPV} = \frac{\text{TN}}{\text{TN+FN}}$$

$$\text{ACC} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \text{ (accuracy)}$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP+FP})(\text{TP+FN})(\text{TN+FP})(\text{TN+FN})}}$$

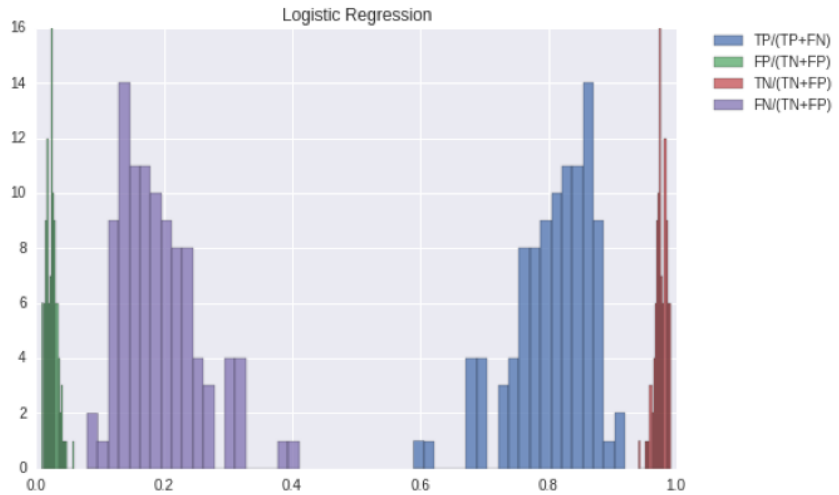


Figure 6: Results averaged over 100 cycles (binary case example)

Actual Class			
0	TN	FP	
1	FN	TP	
	0	1	

Predicted Class

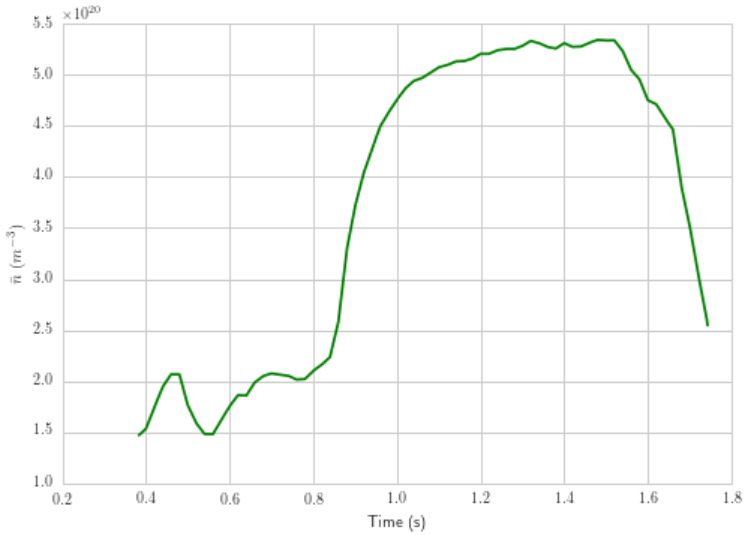
Multi-class classification: L vs H vs I (validation)

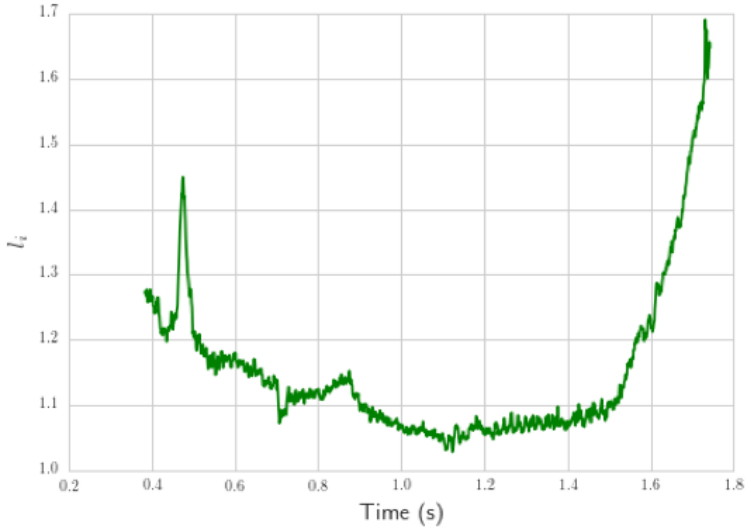
		GNB	LR	MLP	RF
L-mode	PPV	0.957 ± 0.004	0.945 ± 0.003	0.985 ± 0.003	0.989 ± 0.001
	TPR	0.919 ± 0.005	0.958 ± 0.003	0.985 ± 0.003	0.991 ± 0.001
	TNR	0.862 ± 0.012	0.815 ± 0.013	0.953 ± 0.011	0.964 ± 0.003
	NPV	0.763 ± 0.013	0.854 ± 0.009	0.953 ± 0.009	0.972 ± 0.002
	AUC	0.955 ± 0.004	0.971 ± 0.002	0.997 ± 0.000	0.998 ± 0.000
	MCC	0.750 ± 0.015	0.786 ± 0.012	0.938 ± 0.006	0.958 ± 0.003
H-mode	PPV	0.774 ± 0.015	0.830 ± 0.012	0.924 ± 0.015	0.954 ± 0.004
	TPR	0.785 ± 0.017	0.754 ± 0.019	0.922 ± 0.018	0.937 ± 0.006
	TNR	0.965 ± 0.003	0.977 ± 0.002	0.989 ± 0.003	0.994 ± 0.001
	NPV	0.967 ± 0.002	0.964 ± 0.002	0.989 ± 0.002	0.991 ± 0.001
	AUC	0.955 ± 0.005	0.966 ± 0.003	0.997 ± 0.001	0.998 ± 0.000
	MCC	0.746 ± 0.015	0.762 ± 0.015	0.912 ± 0.009	0.938 ± 0.005
I-mode	PPV	0.712 ± 0.020	0.817 ± 0.014	0.972 ± 0.008	0.984 ± 0.002
	TPR	0.889 ± 0.020	0.825 ± 0.018	0.975 ± 0.008	0.987 ± 0.002
	TNR	0.952 ± 0.005	0.975 ± 0.002	0.996 ± 0.001	0.998 ± 0.000
	NPV	0.985 ± 0.003	0.977 ± 0.002	0.997 ± 0.001	0.998 ± 0.000
	AUC	0.973 ± 0.005	0.984 ± 0.002	0.999 ± 0.000	0.999 ± 0.000
	MCC	0.765 ± 0.021	0.797 ± 0.016	0.970 ± 0.004	0.984 ± 0.002
	ACC	0.899 ± 0.006	0.917 ± 0.004	0.976 ± 0.002	0.984 ± 0.001

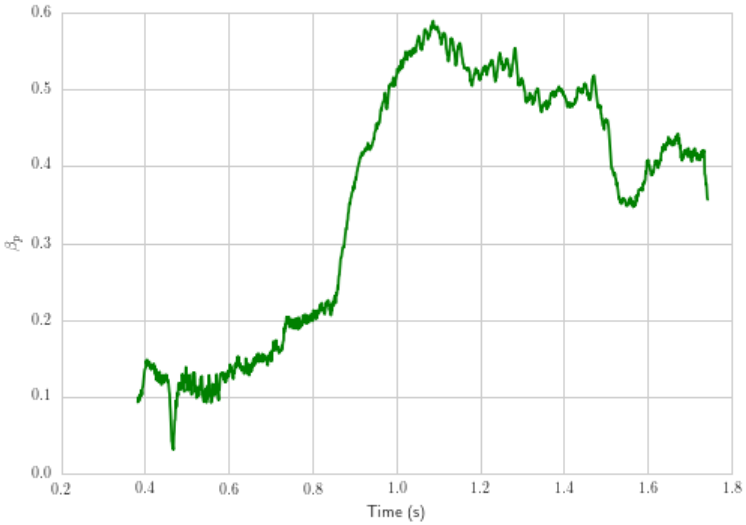
Multi-class classification: L vs H vs I (test)

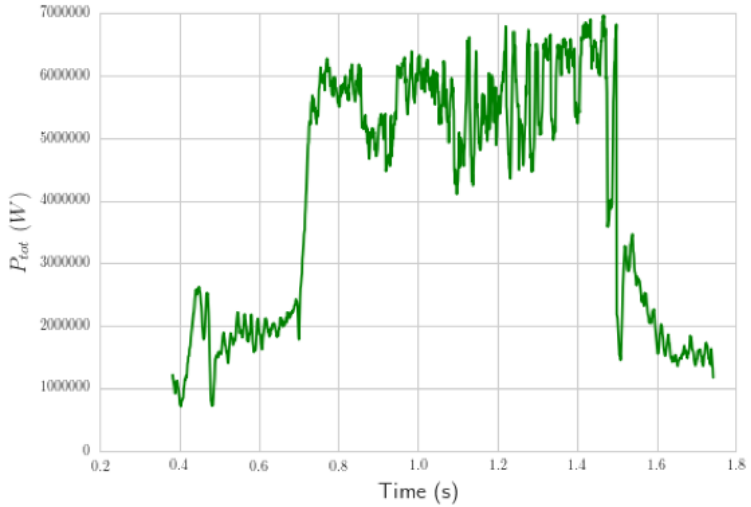
		GNB	LR	MLP	RF
L-mode	PPV	0.956 ± 0.014	0.945 ± 0.013	0.941 ± 0.017	0.943 ± 0.016
	TPR	0.920 ± 0.021	0.956 ± 0.013	0.944 ± 0.015	0.952 ± 0.012
	TNR	0.857 ± 0.051	0.812 ± 0.048	0.794 ± 0.060	0.803 ± 0.059
	NPV	0.762 ± 0.052	0.846 ± 0.044	0.806 ± 0.049	0.831 ± 0.041
	AUC	0.947 ± 0.025	0.971 ± 0.009	0.954 ± 0.021	0.960 ± 0.018
	MCC	0.747 ± 0.045	0.779 ± 0.041	0.742 ± 0.048	0.764 ± 0.046
H-mode	PPV	0.772 ± 0.052	0.837 ± 0.053	0.734 ± 0.064	0.774 ± 0.057
	TPR	0.773 ± 0.065	0.747 ± 0.066	0.884 ± 0.033	0.764 ± 0.066
	TNR	0.965 ± 0.010	0.978 ± 0.008	0.958 ± 0.012	0.966 ± 0.010
	NPV	0.965 ± 0.010	0.962 ± 0.010	0.961 ± 0.010	0.964 ± 0.010
	AUC	0.955 ± 0.016	0.966 ± 0.012	0.952 ± 0.016	0.951 ± 0.021
	MCC	0.737 ± 0.052	0.761 ± 0.052	0.777 ± 0.045	0.733 ± 0.051
I-mode	PPV	0.699 ± 0.086	0.784 ± 0.080	0.792 ± 0.087	0.806 ± 0.076
	TPR	0.882 ± 0.069	0.815 ± 0.076	0.753 ± 0.098	0.764 ± 0.087
	TNR	0.951 ± 0.018	0.972 ± 0.011	0.975 ± 0.012	0.977 ± 0.009
	NPV	0.984 ± 0.010	0.977 ± 0.009	0.968 ± 0.015	0.970 ± 0.014
	AUC	0.953 ± 0.042	0.981 ± 0.008	0.957 ± 0.047	0.969 ± 0.025
	MCC	0.753 ± 0.058	0.773 ± 0.066	0.742 ± 0.074	0.757 ± 0.071
	ACC	0.898 ± 0.016	0.915 ± 0.014	0.898 ± 0.018	0.908 ± 0.016

Shot 1160930033

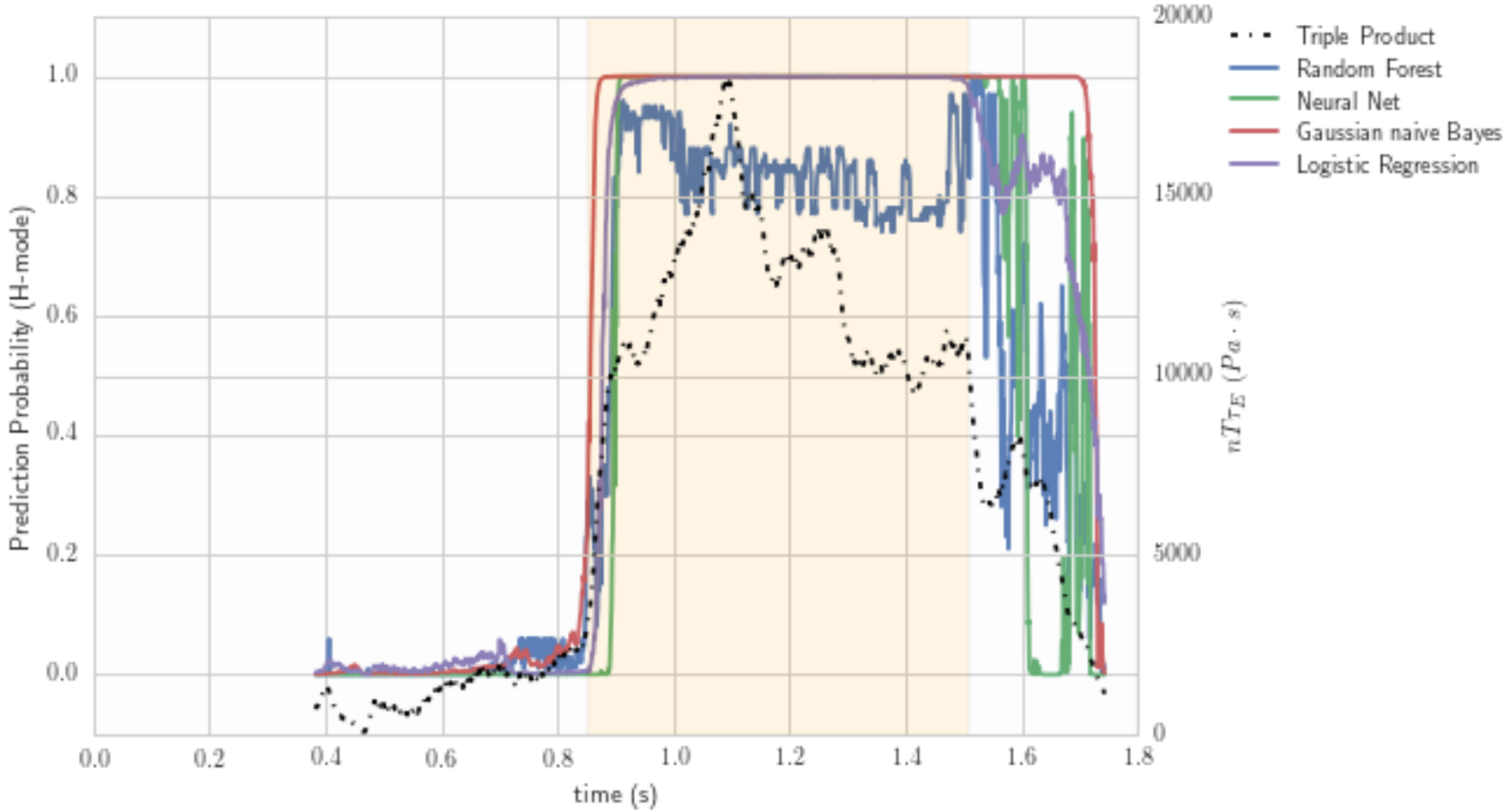






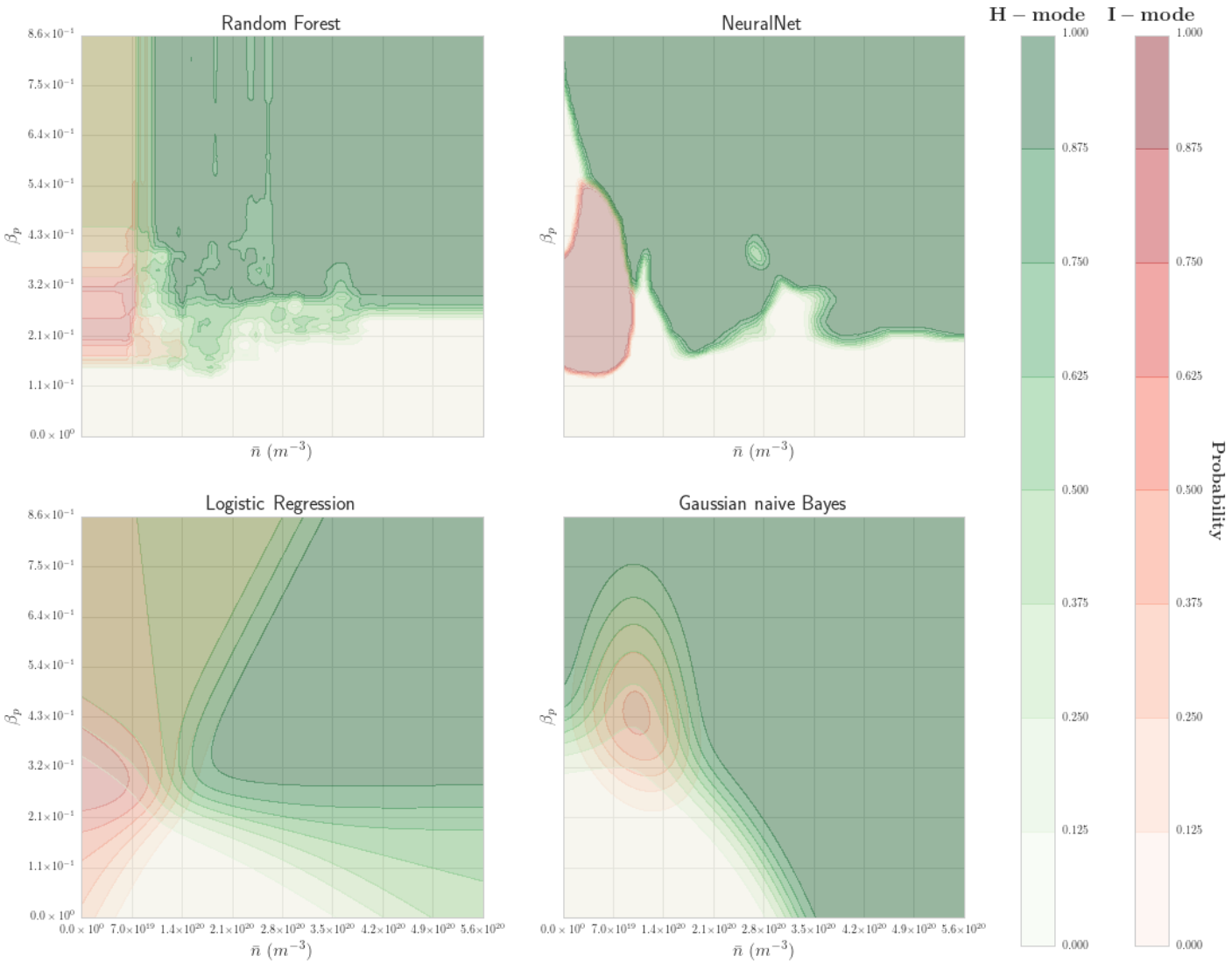


Shot 1160930033



Control

$$P_{tot} = 2.5{\times}10^{6}W,\ l_i = 1.1{\times}10^{0}$$



Summary

Results:

- Development of extensive database involving variety of shots in L-, H-, and I-mode phases
- Capability to instantly identify shots on Alcator C-Mod
- Opens pathway to further investigate relationships such as energy confinement time scaling laws in different regimes
- Can explore parameter space not necessarily tested in past experiments to predict probable outcome and assess feature importance

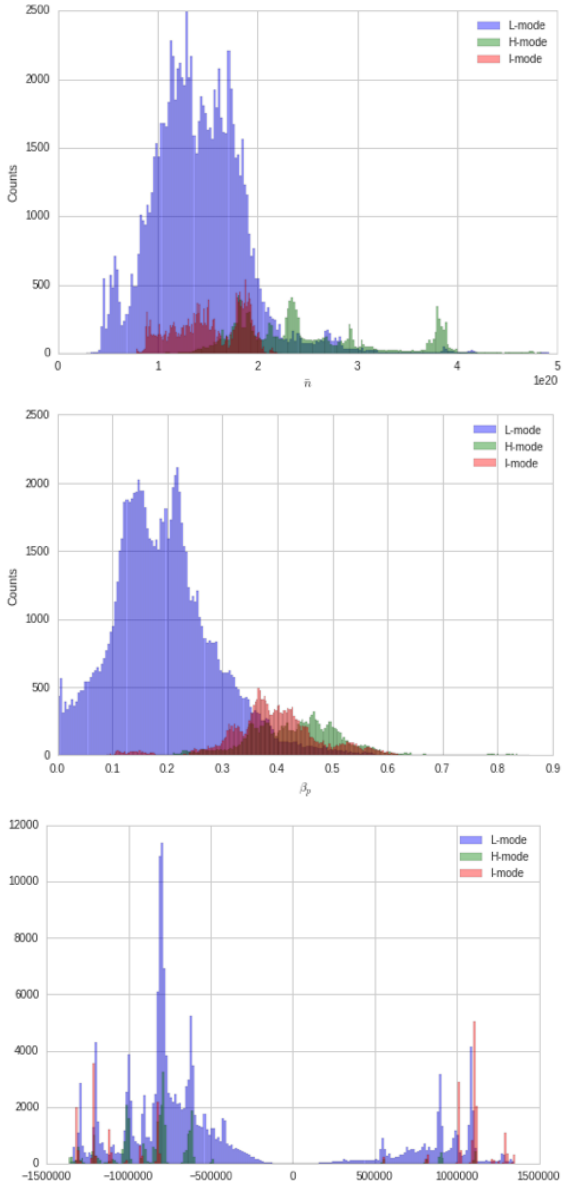
Summary

To do:

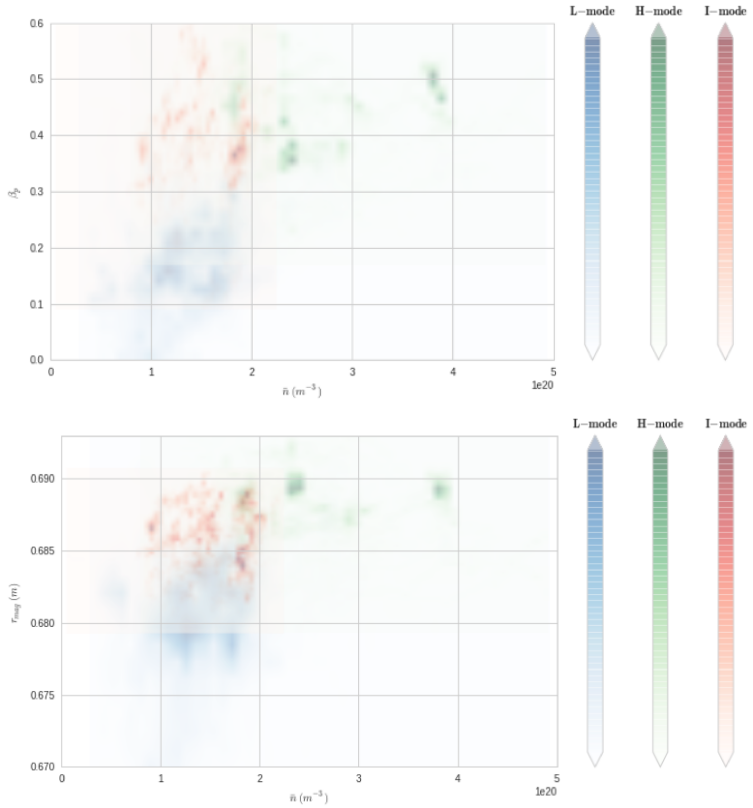
- Development of regression technique for energy confinement time possibly coupled with confinement identification
- Incorporate relevant and readily available spatial and time-dependent quantities for predictive purposes
- Sensitivity analysis and consideration of weaknesses in developing control for real-time application to optimize fusion power output
- Cross-machine implementation and validation (possibly combining disruption predictor)

Extra Slides

Extra: Feature selection



Multiple ways to partition data without a necessarily unique solution



Extra: Feature selection

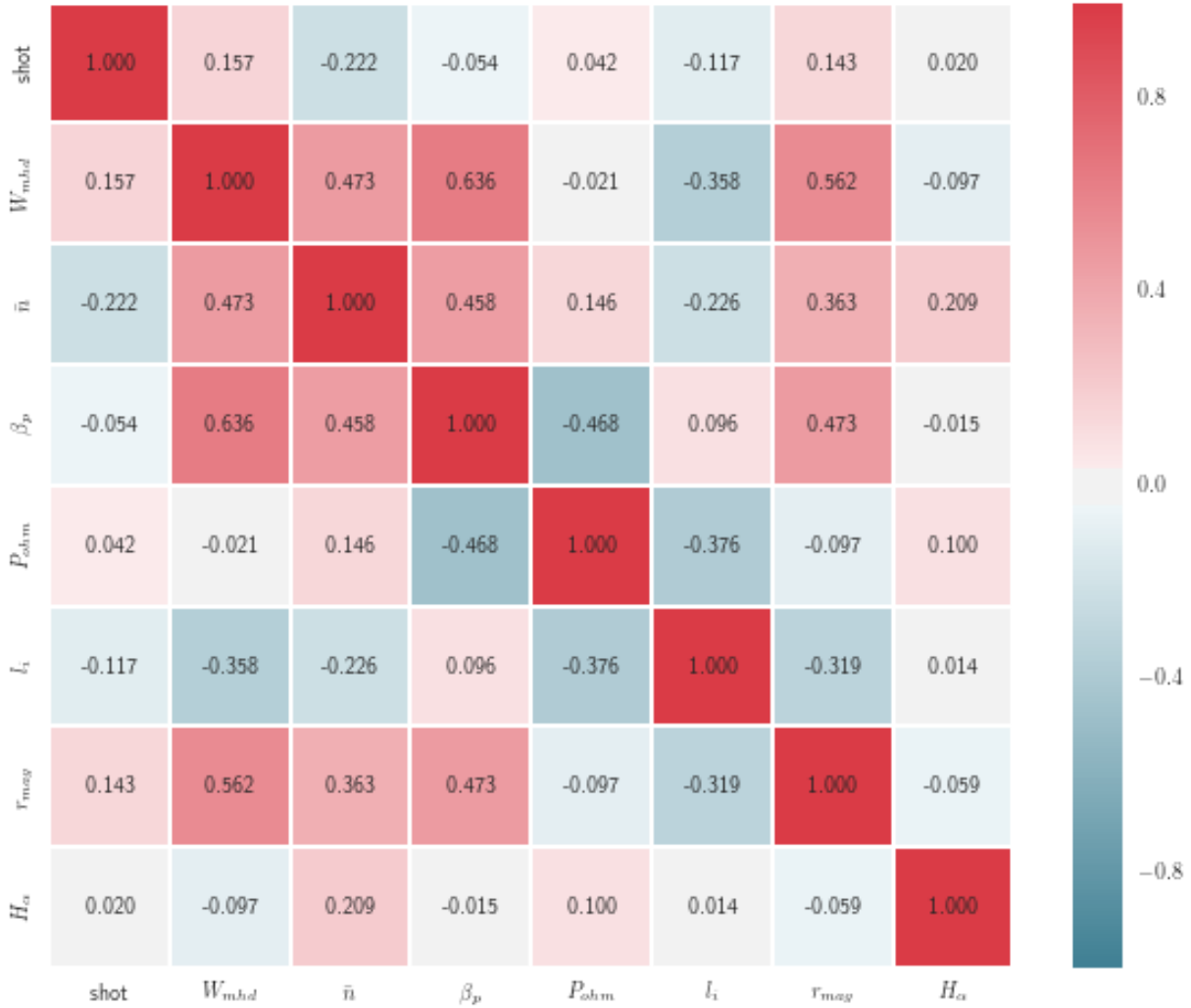


Figure 7: Correlation matrix

Extra: Features

$$l_i = \frac{\langle B_\theta^2 \rangle_P}{B_\theta^2(a)} = \frac{2\pi \int_0^a B_\theta^2(\rho)\rho d\rho}{\pi a^2 B_\theta^2(a)}$$ (for circular cross section plasmas)

Using Ampère's Law ($2\pi a B_\theta(a) = \mu_0 I$), one obtains

$$l_i = \frac{L_i}{2\pi R_0}\frac{4\pi}{\mu_0} = \frac{2L_i}{\mu_0 R_0}$$ where $\frac{1}{2}L_i I^2 = \int_P \frac{B^2}{2\mu_0} dV$

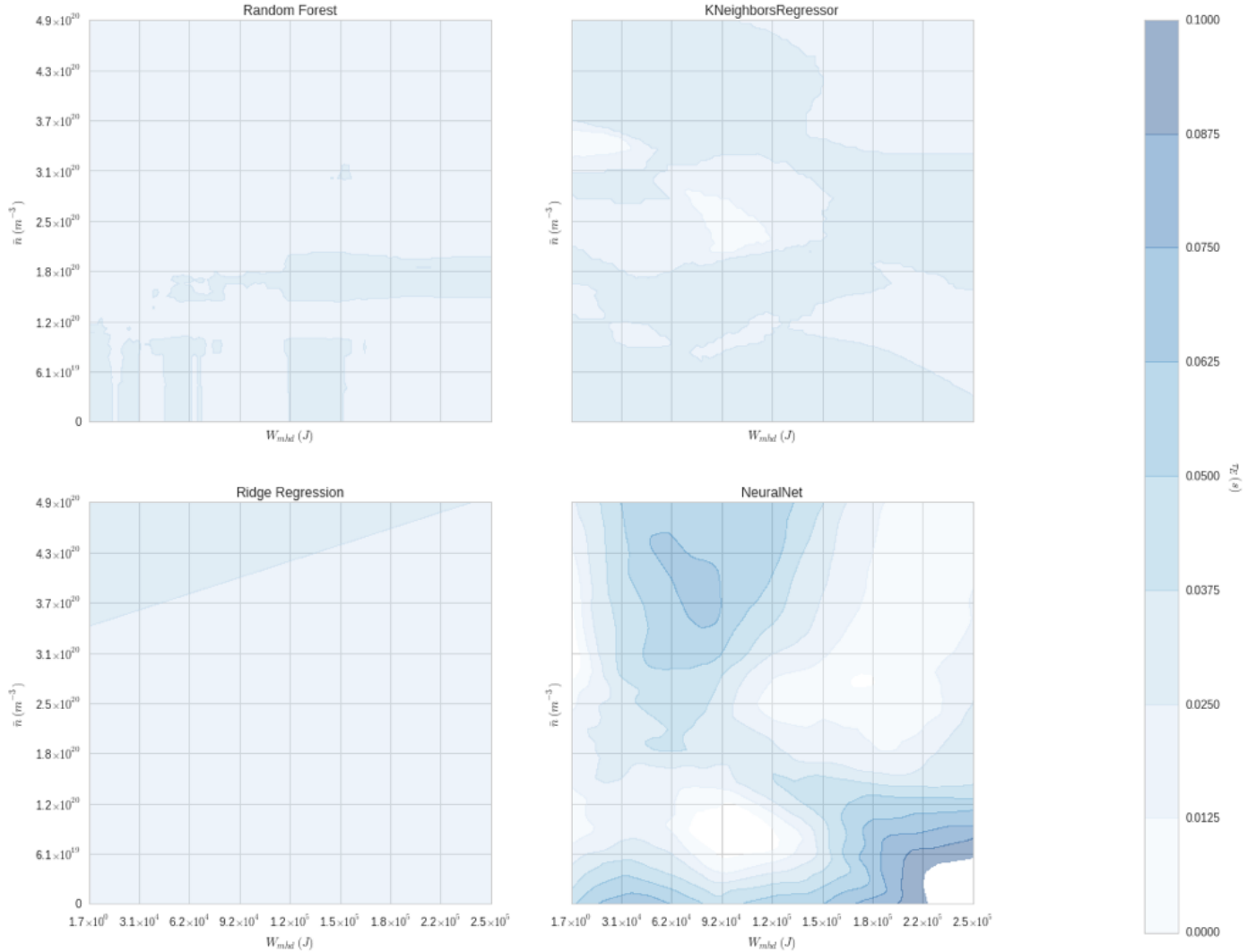
(internal inductance is a volume integral only over the plasma)

$$P_{tot} = P_{RF} + P_{ohm}$$

$$\beta_p = \frac{\langle p \rangle}{B_\theta^2/2\mu_0}$$

Extra: Control

$\beta_p = 3.5 \times 10^{-1}$, $P_{ohm} = 1.7 \times 10^{6}\, W$, $l_i = 1.2 \times 10^{0}$, $r_{mag} = 6.8 \times 10^{-1}\, m$, $H_\alpha = 1.9 \times 10^{0}\, \frac{W}{m^2\, sr}$



A binary choice model assumes a latent variable U_n, the utility (or net benefit) that person n obtains from taking an action (as opposed to not taking the action). The utility the person obtains from taking the action depends on the characteristics of the person, some of which are observed by the researcher and some are not: $U_n = \boldsymbol{\beta} \cdot \mathbf{s_n} + \varepsilon_n$ where $\boldsymbol{\beta}$ is a set of regression coefficients and $\mathbf{s_n}$ is a set of independent variables (also known as "features") describing person n, which may be either discrete dummy variables or regular continuous variables. ε_n is a random variable specifying "noise" or "error" in the prediction, assumed to be distributed according to some distribution. Normally, if there is a mean or variance parameter in the distribution, it cannot be identified, so the parameters are set to convenient values — by convention usually mean 0, variance 1.

The person takes the action, $y_n = 1$, if $U_n > 0$. The unobserved term, ϵ_n, is assumed to have a logistic distribution.

The specification is written succinctly as:

$$U_n = \boldsymbol{\beta} \cdot \mathbf{s_n} + \epsilon_n$$

$$Y_n = \begin{cases} 1, & \text{if } U_n > 0 \\ 0, & \text{if } U_n \leqslant 0 \end{cases} \quad (\epsilon \sim \text{logistic, normal, etc.})$$

Written slightly differently:

$$U_n = \boldsymbol{\beta} \cdot \mathbf{s_n} - e_n$$

$$Y_n = \begin{cases} 1, & \text{if } U_n > 0 \\ 0, & \text{if } U_n \leqslant 0 \end{cases} \quad (e \sim \text{logistic, normal, etc.})$$

Here we have made the substitution $e_n = -\epsilon$. This changes a random variable into a slightly different one, defined over a negated domain. As it happens, the error distributions we usually consider (e.g. logistic distribution, normal distribution, Student's t-distribution, etc.) are symmetric about 0, and hence the distribution over e_n is identical to the distribution over ϵ_n

Denote the cumulative distribution function (CDF) of e as F_e, and the quantile function (inverse CDF) of e as F_e^{-1}. Note that

$$\Pr(Y_n = 1) = \Pr(U_n > 0) \tag{1}$$

$$= \Pr(\boldsymbol{\beta} \cdot \mathbf{s_n} - e_n > 0) \tag{2}$$

$$= \Pr(-e_n > -\boldsymbol{\beta} \cdot \mathbf{s_n}) \tag{3}$$

$$= \Pr(e_n \leqslant \boldsymbol{\beta} \cdot \mathbf{s_n}) \tag{4}$$

$$= F_e(\boldsymbol{\beta} \cdot \mathbf{s_n}) \tag{5}$$

Since Y_n is a Bernoulli trial, where $\mathbb{E}[Y_n] = \Pr(Y_n = 1)$, we have $\mathbb{E}[Y_n] = F_e(\boldsymbol{\beta} \cdot \mathbf{s_n})$, or, equivalently, $F_e^{-1}(\mathbb{E}[Y_n]) = \boldsymbol{\beta} \cdot \mathbf{s_n}$.

Note that this is exactly equivalent to the binomial regression model expressed in the formalism of the generalized linear model.

If $e_n \sim \mathcal{N}(0, 1)$, i.e. distributed as a standard normal distribution, then $\Phi^{-1}(\mathbb{E}[Y_n]) = \boldsymbol{\beta} \cdot \mathbf{s_n}$, which is exactly a probit model.

If $e_n \sim \text{Logistic}(0, 1)$, i.e. distributed as a standard logistic distribution with mean 0 and scale parameter 1, then the corresponding quantile function is the logit function, and $\text{logit}(\mathbb{E}[Y_n]) = \boldsymbol{\beta} \cdot \mathbf{s_n}$, which is exactly a logit model.

Note: $\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = \log(p) - \log(1-p) = -\log\left(\frac{1}{p} - 1\right)$.

The probability density function (pdf) of the logistic distribution is given by:

$$f(x;\mu,s) = \frac{e^{-\frac{x-\mu}{s}}}{s\left(1+e^{-\frac{x-\mu}{s}}\right)^2} = \frac{1}{s\left(e^{\frac{x-\mu}{2s}}+e^{-\frac{x-\mu}{2s}}\right)^2} = \frac{1}{4s}\operatorname{sech}^2\left(\frac{x-\mu}{2s}\right) \quad (6)$$

The logistic distribution receives its name from its cumulative distribution function (cdf), which is an instance of the family of logistic functions. The cumulative distribution function of the logistic distribution is also a scaled version of the hyperbolic tangent:

$$F(x;\mu,s) = \frac{1}{1+e^{-\frac{x-\mu}{s}}} = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{x-\mu}{2s}\right) \quad (7)$$

In this equation, x is the random variable, μ is the mean, and s is a scale parameter proportional to the standard deviation ($\sigma^2 = \frac{\pi^2 s^2}{3}$). The inverse cumulative distribution function (quantile function) of the logistic distribution is a generalization of the logit function. Its derivative is called the quantile density function. They are defined as follows:

$$Q(p;\mu,s) = \mu + s\,\ln\left(\frac{p}{1-p}\right); Q'(p;s) = \frac{s}{p(1-p)}$$

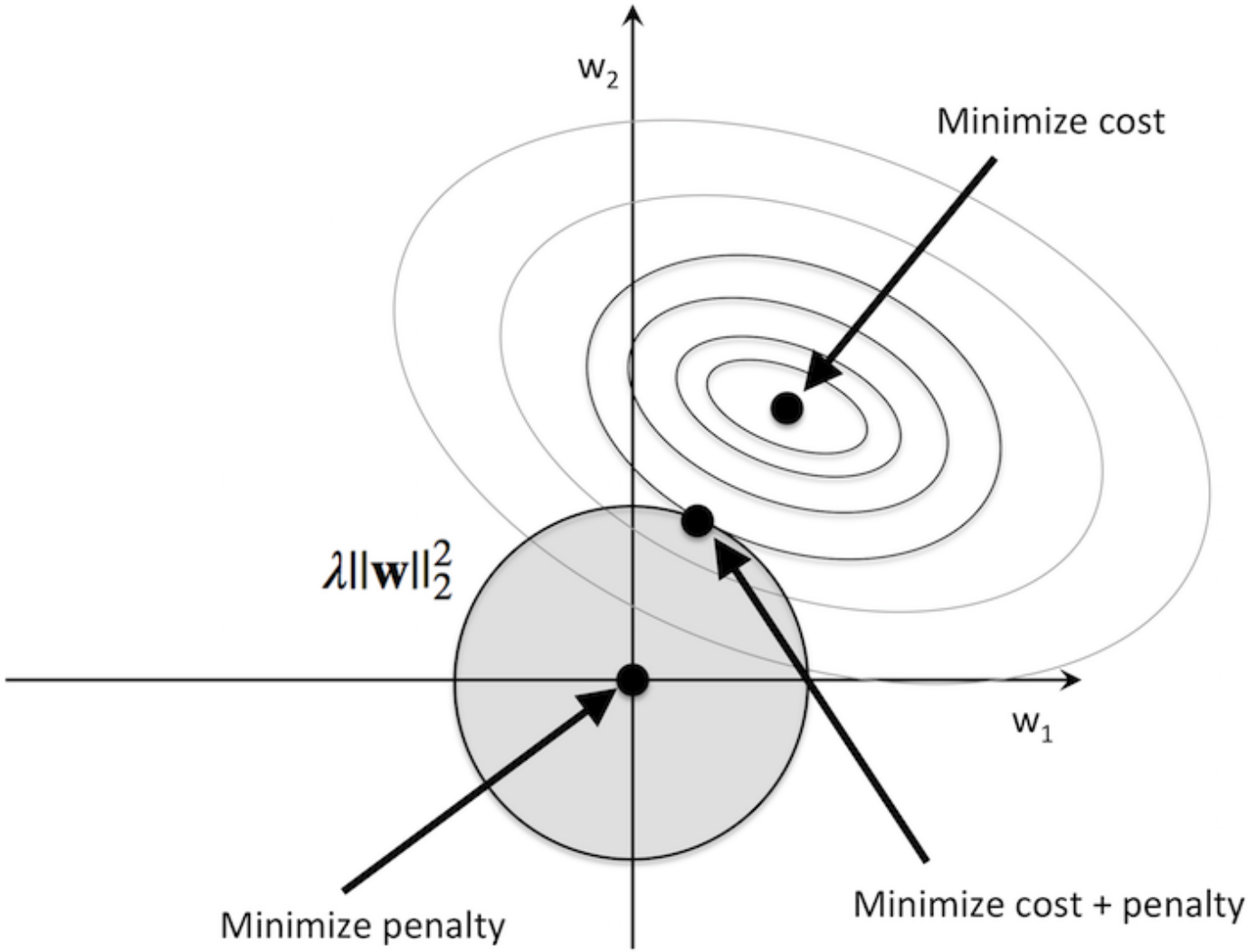
In information theory, Kraft's inequality establishes that any directly decodable coding scheme for coding a message to identify one value x_i out of a set of possibilities X can be seen as representing an implicit probability distribution $q(x_i) = 2^{-l_i}$ over X, where l_i is the length of the code for x_i in bits. Therefore, cross entropy can be interpreted as the expected message-length per datum when a wrong distribution Q is assumed while the data actually follows a distribution P. That is why the expectation is taken over the probability distribution P and not Q.

$$H(p, q) = \mathrm{E}_p[l_i] = \mathrm{E}_p\left[\log \frac{1}{q(x_i)}\right]$$

$$H(p, q) = \sum_{x_i} p(x_i) \log \frac{1}{q(x_i)}$$

$$H(p, q) = -\sum_x p(x) \log q(x).$$

In L1 regularization, some weights approach 0 → sparser solutions compared to L2 which is a quadratic regularizer (below):



$\sum_i (y_i - \hat{y}_i)^2$ is indeed convex in $\hat{y}_i$. But if $\hat{y}_i = f(x_i; \theta)$ it may not be convex in θ, which is the situation with most non-linear models, and we actually care about convexity in θ because that's what we're optimizing the cost function over. Consider a network with 1 hidden layer of N units and a linear output layer: our cost function is

$$g(\alpha, W) = \sum_i (y_i - \alpha_i \sigma(W x_i))^2,$$

where $x_i \in \mathbb{R}^p$ and $W \in \mathbb{R}^{N \times p}$ (and I'm omitting bias terms for simplicity). This is not necessarily convex when viewed as a function of (α, W) (depending on σ: if a linear activation function is used then this still can be convex). And the deeper our network gets, the less convex things are.