Review of Scientific Instruments

Outlier classification using autoencoders: Application for fluctuation driven flows in fusion plasmas

Cite as: Rev. Sci. Instrum. **90**, 013505 (2019); https://doi.org/10.1063/1.5049519
Submitted: 23 July 2018 . Accepted: 14 December 2018 . Published Online: 16 January 2019

R. Kube, F. M. Bianchi, D. Brunner, and B. LaBombard

Review of Scientific Instruments ARTICLE scitation.org/journal/rsi

Outlier classification using autoencoders: Application for fluctuation driven flows in fusion plasmas

Cite as: Rev. Sci. Instrum. 90, 013505 (2019); doi: 10.1063/1.5049519
Submitted: 23 July 2018 • Accepted: 14 December 2018 •
Published Online: 16 January 2019

R. Kube,[1,a)] F. M. Bianchi,[1] D. Brunner,[2] and B. LaBombard[3]

AFFILIATIONS

[1]Department of Physics and Technology, UiT The Arctic University of Norway, N-9037 Tromsø, Norway
[2]Commonwealth Fusion Systems, Cambridge, Massachusetts 02139, USA
[3]MIT Plasma Science and Fusion Center, Cambridge, Massachusetts 02139, USA

[a)]**Electronic mail:** ralph.kube@uit.no

ABSTRACT

Understanding the statistics of fluctuation driven flows in the boundary layer of magnetically confined plasmas is desired to accurately model the lifetime of the vacuum vessel components. Mirror Langmuir probes (MLPs) are a novel diagnostic that uniquely allow us to sample the plasma parameters on a time scale shorter than the characteristic time scale of their fluctuations. Sudden large-amplitude fluctuations in the plasma degrade the precision and accuracy of the plasma parameters reported by MLPs for cases in which the probe bias range is of insufficient amplitude. While some data samples can readily be classified as valid and invalid, we find that such a classification may be ambiguous for up to 40% of data sampled for the plasma parameters and bias voltages considered in this study. In this contribution, we employ an autoencoder (AE) to learn a low-dimensional representation of valid data samples. By definition, the coordinates in this space are the features that mostly characterize valid data. Ambiguous data samples are classified in this space using standard classifiers for vectorial data. In this way, we avoid defining complicated threshold rules to identify outliers, which require strong assumptions and introduce biases in the analysis. By removing the outliers that are identified in the latent low-dimensional space of the AE, we find that the average conductive and convective radial heat fluxes are between approximately 5% and 15% lower as when removing outliers identified by threshold values. For contributions to the radial heat flux due to triple correlations, the difference is up to 40%.

I. INTRODUCTION

Tokamaks confine fusion plasmas, a fully ionized hydrogen plasma with a core temperature of approximately 100 000 000 K, using strong, donut-shaped magnetic fields within a vacuum vessel.[1] The outer boundary region of the plasma comprises a region where closed magnetic field lines wind around toroidal surfaces and a region where open magnetic field lines are guided as to intersect material walls, so-called divertor targets, remote from the central plasma column. As plasma streams along the open field lines onto the divertor targets, it cools. These field lines terminate at divertor structures which facilitate the further removal of the plasma. Thereby this region defines an exhaust channel for the plasma. Intermittent, large-amplitude fluctuations of the plasma parameters, such as the density and the temperature, are characteristic for the outboard mid-plane open field line region.[2–6] These fluctuations are foot prints of coherent structures of excess plasma pressure, called blobs, which propagate radially out over through the open field line region onto the vacuum vessel walls at the outboard mid-plane.[7–11] Depending on their amplitude, these fluctuations can potentially erode the vacuum vessel. Impurities released from the wall may furthermore accumulate within the confined plasma column and negatively impact the confinement properties of the plasma. Nowadays tokamaks perform experiments on plasma discharges which last for several seconds. Future fusion reactors

need to operate with long pulses or continuously. In order to model the life time of the plasma facing components for such requirements, a precise and accurate description of this fluctuation driven transport is desired.[12,13]

Langmuir probes are the workhorse used to diagnose this boundary region plasma. They are implemented as electrodes immersed into a plasma. Using electric current and voltage samples recorded by a Langmuir probe, plasma quantities are recovered from the relation[14]

$$I_{pr} = I_{sat}\left[1 - \exp\left(e\frac{V_{pr} - V_f}{T_e}\right)\right]. \quad (1)$$

Here I_{pr} is the collected electric current and V_{pr} is the applied bias voltage. T_e gives the electron temperature of the plasma measured in electronvolt. The floating potential V_f is defined as the electric potential assumed by an electrically isolated object if it were to be immersed into the sampled plasma. The ion saturation current I_{sat} is the maximal current that can be drawn by an electrode, which is limited by ion collection of the electrode.

In order to estimate the particle and heat fluxes driven by the electric drift, the electron density, the temperature, and the local electric field need to be recovered from probe measurements. Commonly, these quantities are recovered from probes by applying a sweeping voltage to the electrode. This allows us to sample several current-voltage measurements (I_{pr}, V_{pr}) during one sweep. From these, I_{sat}, T_e, and V_f are obtained from a fit on Eq. (1). The ion saturation current and the electron temperature can be used to calculate the electron density of the plasma as[14]

$$n_e = 2\frac{I_{sat}}{eA_p\sqrt{k_b T_e/m_i}}. \quad (2)$$

Here e is the elementary charge, A_p is the current collecting area of the electrode, k_b is the Boltzmann constant, and m_i denotes the ion mass. The electric potential in the plasma can be estimated as

$$V_p = V_f + \Lambda T_e, \quad (3)$$

where $\Lambda \approx 2$–3 for scrape-off layer plasmas.[15,16] Potential measurements from poloidally separated electrodes allow us to estimate the poloidal electric field, which drives the radial electric drift.

A characteristic time scale for fluctuations of n_e, T_e, and V_p in the boundary plasma is given by approximately 10 μs.[6,17–24] Sweeping the voltage with a frequency larger than approximately 100 kHz however leads to hysteresis effects in the sampled current-voltage characteristic as the bias voltage polarizes the flux tube in which the plasma is sampled from Refs. 25 and 26. Thus, Langmuir probes used in this manner cannot sample the plasma parameters on a fast enough time scale to resolve the fluctuations of the boundary layer plasma.

The Mirror Langmuir Probe (MLP) biasing technique allows us to sample I_{sat}, T_e, and V_f on a time scale below that of the boundary layer plasma fluctuation.[27,28] The MLP diagnostic consists of three main components. The actual mirror Langmuir probe is an electric circuit outputs a current-voltage (I-V) characteristic with three adjustable parameters I_{sat}, T_e, and V_f, given by Eq. (1). The second main component is a Langmuir electrode immersed in the plasma. Both components are connected to a fast switching biasing waveform, the third main component of the MLP diagnostic. The bias waveform switches between the states (V^+, V^0, V^-) such that the Langmuir electrode draws approximately $\pm I_{sat}$ at the states $V^\pm$ and zero net current when biased to V^0, as shown in Fig. 1 of Ref. 28. The target bias voltage state is updated every 300 ns. Once the bias voltage has settled, the current drawn from the MLP and the Langmuir electrode are sampled. The ion saturation current, the plasma potential, and the electron temperature are recovered by a fit of Eq. (1) to the data samples from the Langmuir electrode.

The main task of the MLP circuit is to set and maintain the optimal range of the bias voltages such that a complete I–V characteristic can be reconstructed from current samples drawn by the Langmuir electrode at the three bias voltage states. In order to account for varying plasma conditions, the MLP dynamically updates the voltage states V^+ and V^- relative to the running average of the electron temperature samples over a 2 ms window such that $\Delta_V < 4\overline{T}_{e,2ms}$ holds. Here, $\Delta_V = V^+ - V^-$ and $\overline{T}_{e,2ms}$ denotes this running average of the electron temperature.

On the other hand, large amplitude fluctuations of the boundary layer plasma have a characteristic time scale of approximately 10 μs. During such transient events, the electron temperature may significantly exceed the running average, $T_e > \overline{T}_{e,2ms}$, such that the adjusted biasing voltage range may be insufficient to guarantee a precise fit on the true I–V characteristic of a hypothetical Langmuir probe. However events such as probe arcing may result in unphysical fit values.

A large body of experimental measurements suggest that the fluctuation statistics of the boundary plasma depend on the global parameters of the plasma discharge, such as line-averaged core plasma density and the magnetic geometry.[18,23,29–32] Since the MLP biasing drive is agnostic to these circumstances, the accuracy and precision of data samples reported by the MLP may vary, depending on the plasma it samples. In order to accurately calculate lower order statistical moments of MLP data time series or distributions such as the probability distribution function or power spectral density, low-accuracy data samples should be discarded. However, if too many samples are discarded, these moments or distribution functions cannot be estimated with high statistical significance, due to the scarcity of the available data points.

One way of pruning MLP data time series is to define valid ranges for the MLP parameters. Within these thresholds, samples are kept and out of bound samples are to be discarded. A sensible boundary, or thresholds, needs to be low enough in order to reject samples with unphysically large fluctuation values. On the other hand, the threshold value needs

to be large enough so that the accepted data points correctly capture the properties of the plasma fluctuations of the interrogated plasma. While measurements with a sufficient or insufficient biasing voltage range are readily identifiable, such a decision is ambiguous for a large fraction of other samples. In practice, it is often the case that several nearby Langmuir electrodes sample the plasma. Given that MLP samples may be quite heterogeneous when operating on a small biasing voltage range, a threshold based method requires domain expertise and inevitably introduces biases.

A. Proposed approach

The approach proposed here adopts simple thresholds to identify all *good* and *bad* measurements as a primer. This identification will be non-exhaustive and will leave a large number of samples unclassified. From this, all uncertainty in the quality of the measurements will be treated with machine learning techniques which exploit statistical properties and regularities in the data. This approach allows us to label unclassified data by making inference, as opposed to labelling them using a complicated set of rules.

Specifically, we present an outlier classification framework based on an autoencoder (AE), a type of neural network that can be used to learn low-dimensional representations of arbitrary datasets. AEs will be trained using only good measurement samples so that they learn how to map them into low dimensional representations. Each dimension of the space induced by the AE mapping corresponds to a combination of features which best characterize the important features of *good* measurements. Those features are identified without making any *a priori* assumption, but are automatically selected by the AE as the ones that are, *on average*, the most informative to describe the training samples. As a consequence, the numerical values of features in training samples will be similar and are mapped into a compact cluster in that low dimensional space.

AEs learn a representation of *good* measurements that are more *powerful*, due to the regularization constraints of the dimensionality reduction, and generalize better the samples. Evaluating similarities among samples represented in this new space is arguably more meaningful and reliable.

Once an AE is trained and the mapping to such a low dimensional space is learned, the unclassified samples will be processed. *Bad* measurements lack the characteristic features of *good* measurements and are expected to map onto vectors with a large distance to the cluster composed of *good* samples.

In order to identify a boundary between the representations of *good* and *bad* measurements, classifiers for vectorial data will be trained in this new space. Unclassified data samples are assigned a label based on which side of the decision boundary they fall.

The rest of this article is structured as follows: Sec. II describes the measurements of plasma fluctuations by MLPs and discusses the structure of valid and invalid data at hand. Section III introduces AEs and describes their application for outlier detection in large datasets. The proposed classification method and its application to MLP data are described in Sec. IV. Section V discusses the performance of the proposed framework, and Sec. VI gives a conclusion.

II. MEASUREMENTS OF PLASMA FLUCTUATIONS

Dedicated experiments with the goal to describe the statistics of fluctuation driven flows in the boundary plasma have been performed in the Alcator C-Mod tokamak.[33–36] In these experiments, the boundary layer of an ohmically heated, lower single-null plasma discharge with a toroidal magnetic field strength of B_T = 5.4 T was interrogated by four MLPs, connected to the electrodes of a Mach probe head, as shown in Fig. 1. The probe head was mounted on a linear servomotor probe drive system[37] and dwelled flush with plasma facing components at the outboard mid-plane location, as shown in Fig. 2. Extraordinarily long data time series of 1 s duration were sampled in stationary plasma discharge conditions with the goal to calculate the fluctuation statistics for the plasma with unprecedented accuracy.

The line-averaged core plasma density of the investigated discharge is $\overline{n}_e/n_G \simeq 0.6$, where n_G denotes the Greenwald density.[30] For such high line-averaged core plasma densities, the average electron temperature in the far scrape-off layer plasma is below 10 eV. For lower $\overline{n}_e/n_G$, the scrape-off layer is commonly warmer.[29] On the other hand, the MLPs register order unity fluctuations of the electron temperature. That is, for such high $\overline{n}_e/n_G$ and accompanying temperatures in the scrape-off layer, the MLP biasing drive operates at the limits of its design.

In order to assess the accuracy of fit parameters reported by the MLPs, they were compared among the four MLPs. Since

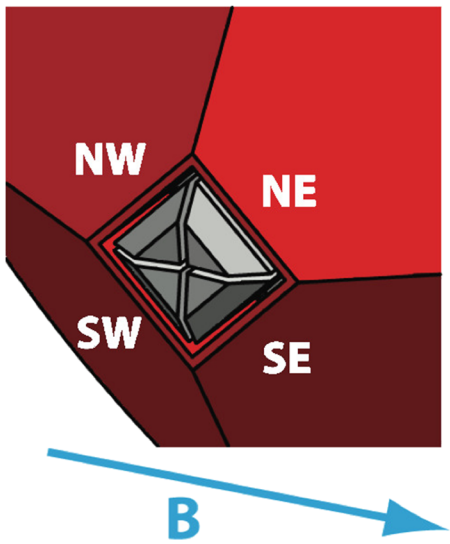

FIG. 1. The Mach probe head with four Langmuir electrodes, labelled "NE," "SE," "SW," and "SE," protruding from its top. The blue arrow denotes the direction of the local magnetic field.

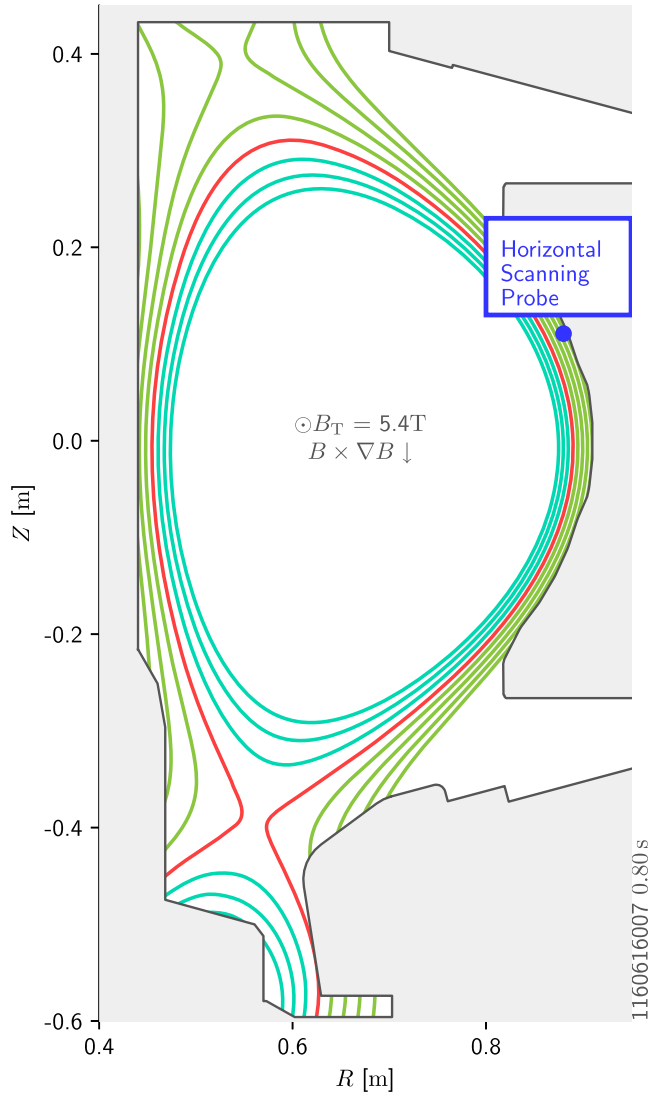

FIG. 2. The poloidal cross section of the Alcator C-Mod tokamak. The blue dot marks the location where the MLPs sample the plasma. Green lines denote the open magnetic field lines, and cyan lines denote the closed magnetic field lines. The red line separates the open field line region from the closed field line region. Material structures are shown in gray.

the electrodes on the probe head are separated by approximately 2 mm, smaller than the characteristic size of the structures in the boundary layer,[11] it is expected that all four MLPs report similar fit parameters. Indeed, I_{sat}, T_e, and V_f fit parameters reported from the four MLPs are of comparable magnitude when the range of the biasing voltage states are large, $\Delta_V > 4T_e$. For the case where $\Delta_V \lesssim 4T_e$, the reported T_e fit parameters may show significant deviations. Operating with small bias voltage ranges, the relative fit error of the electron temperature, σ_{T_e}/T_e, is furthermore on average larger than that for the case $\Delta_V > 4T_e$. The relative error on I_{sat} and T_e reported by the fit routine are correlated with a Pearson sample correlation coefficient of approximately one. The relative error on the floating potential is uncorrelated with the relative error of both I_{sat} and T_e. While both I_{sat} and T_e are positive definite quantities, V_f may assume both positive and negative values. Thus, the relative error on the floating potential, σ_{V_f}/V_f, assumes large absolute values for small absolute values of V_f. This quantity is therefore not suitable to identify poor fits. Poor fits are identified by a large T_e value, a large relative fit error σ_{T_e}/T_e, and a small fit domain Δ_V/T_e.

Figure 3 shows data time series reported by the north-east and south-west MLP. The upper panel shows the electron temperature, the middle panel shows the relative error on T_e, and the lower panel shows the biasing voltage range. A large fraction of the samples feature small to moderate T_e values, together with small error proxies, that is, a relative error $\sigma_{T_e}/T_e \lesssim 0.1$ and a large biasing voltage range. Within these ranges, the fit parameters reported by the different MLPs are similar to one another, indicating that they are both accurate and precise.

Large-amplitude fluctuations of the electron temperature appear intermittently in both time series. While the MLPs register them simultaneously, they report dissimilar T_e values, varying by up to 100%. Large amplitude fluctuations are furthermore associated with a large relative error σ_{T_e}/T_e and a small biasing voltage range. Comparing the appearance of large amplitude peaks sampled by the two MLPs, they may be grouped into several categories. One category are large amplitude peaks recorded by multiple MLPs but with disparate T_e values, for example, at 45.1 ms, at 45.4 ms, or at 45.9 ms. The other category are peaks where the MLPs report similar T_e values, for example, at 45.25 ms or at 46.6 ms. Judging by the fit parameters reported by a single MLP, such peaks should be discarded. However, in the case where multiple MLPs report similar peaks, such samples may be retained. For the data at hand, electrode-averaged values may be used in combination with threshold values to identify samples which should be certainly kept or discarded. But for a majority of the data, such a simple classification may be ambiguous.

This broad range of variations under which large amplitude peaks are observed suggest that it is impractical to

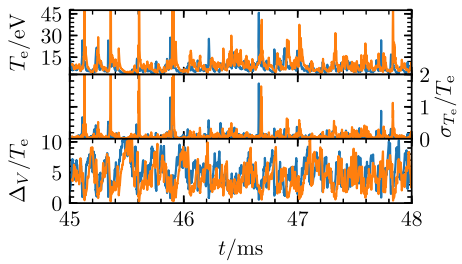

FIG. 3. Time series of the electron temperature (upper panel), the relative error on T_e (middle panel), and the range of the biasing voltages (lower panel), reported by the north-east (blue lines) and south-west (orange lines) MLP. Time series from the latter MLP are delayed by 20 μs for better visibility.

TABLE I. Threshold values used for *a priori* partitioning of the data. The first number gives the threshold for a poor fit, and the second number gives the threshold for a good fit. The lowest row lists the fraction of data labeled as *uncertain* and *bad*.

Quantity	Relaxed	Mid	Strict
T_e/eV	45/50	40/45	35/40
σ_{T_e}/T_e	0.75/1.0	0.5/0.75	0.25/0.5
Δ_V/T_e	2.5/1.5	3.0/2.0	3.5/2.5
Uncertain/*bad*	20.3%/0.1%	30.0%/0.1%	40.2%/0.2%

develop a comprehensive set of rules based on which to accept or reject reported peak amplitudes. In the following, we discuss how statistical inference can be used to derive such rules based from a priming sample of *good* or accepted data.

A. Dataset description and threshold definition

Data time series of T_e, σ_{T_e}/T_e, and Δ_V, sampled by all four MLPs, are combined into a single dataset $\mathcal{X} = \left\{T_{e,p}, \sigma_{T_{e,p}}/T_{e,p}, \Delta_V/T_{e,p} \mid p \in \{\mathrm{NE}, \mathrm{SE}, \mathrm{SW}, \mathrm{NW}\}\right\}$. Each sample is a vector in $\mathbb{R}^{12}$ corresponding to the individual measurements at a given time. We apply a simple threshold mechanism to label only a fraction of the original dataset. In particular, we identify *good* and *bad* samples, $\mathcal{X}^g$ and $\mathcal{X}^b$, while the remaining samples are left unlabelled and referred as *uncertain* $\mathcal{X}^u$.

A fit reported by a single MLP is considered valid if T_e and σ_{T_e}/T_e are below a threshold value and if Δ_V/T_e exceeds a threshold value. If the opposite conditions are true, the fit is considered invalid. If at least two MLPs report a valid fit, the vector is labelled *good* and assigned to $\mathcal{X}^g$. If at least two MLPs report an invalid fit, the vector is labelled *bad* and assigned to $\mathcal{X}^b$.

Table I lists three different sets of threshold values that are used for an *a priori* partitioning of the dataset $\mathcal{X}$. Depending on the threshold values used, the fraction of data points classified as *good*, *uncertain*, and *bad* varies. For example, the category *relaxed* denotes the partitioning that excludes the least amount of data from being categorized. Fits that report electron temperatures of up to 45 eV with a relative error of 0.75 over a range of $\Delta_V/T_e \geq 2.5$ are considered as valid. The fraction of *bad* and *uncertain* samples is listed in bottom row of Table I. Using *relaxed* thresholds, approximately 20% of the data is unclassified, while approximately 40% of the data is labeled *uncertain* when using *strict* thresholds.

In the following, we describe an approach where an AE is facilitated to identify data, which cannot be classified reliably by applying a threshold method.

III. AUTOENCODERS

AEs[38] are a particular class of neural networks, which received increasing interest in recent years.[39–41] AEs can be used to learn unsupervised compressed, or lossy, representations of data, by training the network to map the input in a lower dimensional space through a bottleneck layer and then reconstruct the original input. In this way, the AE learns how to compress inputs, by retaining only the most important information necessary to yield a reconstruction that is as much accurate as possible.[42] Indeed, training AEs by minimizing a reconstruction error corresponds to maximizing the lower bound of the mutual information between the input and the learned representation.[43]

The bottleneck enforces a strong regularization that provides noise filtering, prevents the AE from learning trivial identity mappings (i.e., the identity function), and guarantees robustness to small changes in the inputs.[44] Further regularization can be used to prevent overfitting on the training data and enhance the generalization properties of the representations. The most common regularizations are applying a ℓ_2 norm penalty to the weights learned network and using dropout[45] to randomly drop connections between neurons at each iteration in the training phase. Dropout hinders couplings among neurons and therefore encourages to diversify the behavior of neurons.

In the training phase, an AE learns two functions at the same time. The first one is called *encoder* and provides a mapping from an input domain, $\mathcal{X}$, to a code domain, $\mathcal{Z}$, i.e., the latent representation space. Specifically, an input $\mathbf{x}$ is represented as the output $\mathbf{z}$ of the innermost layer in the AE. The second function, called *decoder*, implements a mapping from $\mathcal{Z}$ back to $\mathcal{X}$. Figure 4 depicts a standard AE architecture with a bottleneck.

The encoding function $\mathrm{E}(\cdot) : \mathcal{X} \to \mathcal{Z}$ and the decoding function $D(\cdot) : \mathcal{Z} \to \mathcal{X}$ of the AE define the following deterministic posteriors:

$$\begin{aligned} \mathbf{z} &= \mathrm{E}(\mathbf{x}) = p(\mathbf{z}|\mathbf{x}; \theta_E), \\ \tilde{\mathbf{x}} &= D(\mathbf{z}) = q(\tilde{\mathbf{x}}|\mathbf{z}; \theta_D), \end{aligned} \tag{4}$$

where θ_E and θ_D are the trainable parameters of the two functions, $\mathbf{x}$ is the original input, $\mathbf{z}$ is the code representation, and $\tilde{\mathbf{x}}$ is the reconstruction of the input. The encoding and decoding functions are usually implemented as two feed-forward

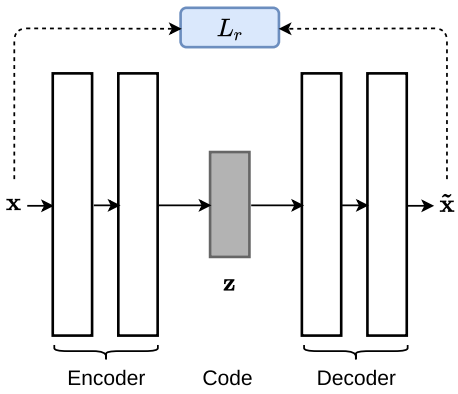

FIG. 4. Schematic representation of the AE architecture with a bottleneck. The encoder generates a low dimensional representation $\mathbf{z}$ of the input $\mathbf{x}$. The AE is trained by minimizing the discrepancy (quantified by the loss L_r) between $\mathbf{x}$ and its reconstruction $\tilde{\mathbf{x}}$ yielded by the decoder.

neural networks, which are constrained to be symmetric. Each network consists of a stack of layers that can be dense, convolutional,[46] or recurrent.[47] Here, we focus only on dense layers that are implemented by an affine transformation followed by a non-linear activation function applied component-wise. Common activation functions are the sigmoid (logistic function, tanh), the maxout,[48] and the rectified linear unit (ReLU).

Each layer contains a different number of processing units (neurons), which affects the capability of approximating a generic function. While a large number of layers and neurons per layer can provide more powerful modeling capabilities, the number of parameters increases with a consequent risk of overfit and a greater demand of computational resources. Therefore, an optimal configuration of the network should account for those contrasting properties.

The configuration of an AE with K layers in the encoder and decoder, respectively, can be suitably expressed as

$$\mathcal{C} = \{e_0, \ldots, e_K, z, d_0, \ldots, d_K\}, \tag{5}$$

where e_i and d_i define the number of neurons in the ith layer of the encoder and the decoder. The size of the innermost layer is denoted by z and defines the dimension of the representation $\mathbf{z}$. As previously stated, we implement a symmetric encoder/decoder architecture by enforcing the following constraint: $e_i = d_{K-i}$.

In order to minimize the discrepancy between the input and its reconstruction, the parameters θ_E and θ_D are adjusted by minimizing the following reconstruction loss through stochastic gradient descent:

$$L = L_r + \lambda L_2 = \mathbb{E}_{\mathbf{x}\sim\mathcal{X}}\left[\|\mathbf{x} - \tilde{\mathbf{x}}\|^2\right] + \lambda\left(\|\theta_E\|^2 + \|\theta_D\|^2\right). \tag{6}$$

The term L_r minimizes the mean squared error between original inputs and their reconstructions, while L_2 penalizes large model weights. The hyperparameter λ controls the latter contribution to the total loss.

Besides the regularization parameter λ and the network configuration $\mathcal{C}$, other hyper-parameters that must be chosen by the user, or optimized by means of a validation procedure, are the following: the probability p_{drop} to drop neural connections during the training, the learning rate η used in stochastic gradient descent, and the type of activation function implementing the non-linearities within each layer of the AE. We refer to the whole set of hyper-parameters as Γ_{ae}.

A. Outlier detection with autoencoders

Outlier detection (also referred to as anomaly detection) is an important area of study in machine learning and is applied to several case-studies where non-nominal samples are scarce, noisy, and not always available during training. The objective of outlier detection procedures is to identify anomalous patterns, the outliers, in data that do not conform to an expected behavior.[49]

Dimensionality reduction procedures, such as Principal Component Analysis (PCA), AEs, and energy based models[50,51] identify a subspace defined by the directions with largest variation among the nominal samples. While PCA can only capture variations that emerge from linear relationships in the data, more sophisticated models such as AEs also account for non-linear relationships. Therefore, AEs can identify a subspace defined by features that better characterize the nominal samples.

Anomaly detection methods based on dimensionality reduction rely on the assumption that anomalous samples do not belong to the subspace, learned during training, that contains nominal data. Indeed, the representations generated for samples of a new, unseen class will arguably fail to retain important characteristics, since the latent low-dimensional space induced by the AE does not span the most relevant features of the anomalous data. As direct consequences, the AE would yield large reconstruction errors for those samples and their low-dimensional representations would be significantly different and more scattered than for samples from the nominal class. This effect can be exploited to obtain an implicit separation between the classes in the code space, which can facilitate the separations of the two classes by a subsequent classifier.

Similar assumptions are reasonable for the MLP dataset at hand. As shown in Fig. 3, a large fraction of the MLP samples feature similar T_e fit values, together with σ_{T_e}/T_e and $\triangle_V$ values which indicate a reliable fit. These samples are considered as inliers and are used to train an AE. Having learned the important characteristics of inlier samples, hitherto unclassified samples will be mapped into the code space of the AE. Samples which do not share the important characteristics of the inlier samples should then be readily identifiable. In the following, we describe a classification framework that exploits this property of the data at hand to identify and separate outliers.

IV. PROPOSED CLASSIFICATION FRAMEWORK AND SELECTION OF MODEL PARAMETERS

The critical components of the proposed classification framework are the AE and the classifier used in the latent code space of the AE to discriminate between *good* and *bad* samples. Beside the trainable parameters, both components depend on a set of hyper-parameters whose tuning may affect the behavior of the whole framework. In the following, we discuss how the choice of a classifier and hyper-parameters for both the AE and the classifier results in different statistics of the inlier T_e data. In Sec. III, we discuss how the choice results in different statistics of the fluctuation driven heat flux. Since there is no ground truth available, that is, the real electron temperature of the plasma is unknown, no quantitative evaluation of the classification framework's performance can be formulated. Instead, the design choices will be guided by the inferred biases of the filtered datasets for any given set of hyperparameters of the classification framework.

As discussed in Sec. III, the AE depends on several hyper-parameters $\mathbf{\Gamma}_{\text{AE}}$. In the following, we discuss the sensitivity of the mapping induced by the AE on them. Figure 5 shows the

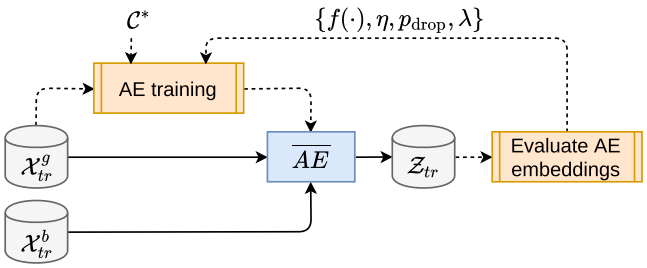

FIG. 5. Pipeline of the procedure to train the AE and select the optimal hyperparameters. $\overline{\text{AE}}$ denotes a trained AE and $\mathcal{C}^*$ denotes an optimal layout of the AE, i.e., the one defined by the hyperparameters configuration that is optimal for the specific task at hand.

pipeline used for this task. For the sake of simplicity, we furthermore consider only the network layout $\mathcal{C}^* = \{12, 2, 12\}$ at this point.

5000 random elements from $\mathcal{X}^g$ are used to train the AE. During training, we observe little sensitivity to the hyperparameters p_{drop}, η, and λ. In the following, we select $p_{\text{drop}} = 10^{-1}$, $\eta = 10^{-2}$, and $\lambda = 10^{-3}$ as hyperparameters.

In feed-forward neural networks, each neuron computes the sum $y = \sum_i x_i w_i + b$, where x_i denotes the input from the previous layer, w_i denotes a weight, and b denotes a bias. The weights and the bias are determined during the training phase. The output of each neuron is $f(y)$, which is called the activation function. The activation functions considered here are

$$\begin{aligned} \text{sigmoid:}\ & f(y) = \frac{1}{1+e^{-y}}, \\ \tanh:\ & f(y) = \frac{1-e^{-2y}}{1+e^{-2y}}, \\ \text{ReLU:}\ & f(y) = \max(0, y), \\ \text{Maxout:}\ & f(y) = \max_{r=1,\ldots,R} (\mathbf{w}_r y + \mathbf{b}_r). \end{aligned} \tag{7}$$

Maxout is a non-parametric activation function which computes the output as the maximum of R different products of the input y with R separate weight vectors $\mathbf{w}_1, \ldots, \mathbf{w}_R$ and biases $\mathbf{b}_1, \ldots, \mathbf{b}_R$.[48] Those weights and biases are trained along with the other parameters of the network. In the experiments, R has been set to 5.

Figure 6 shows 1000 data points of the sets $\mathcal{X}^g$ and $\mathcal{X}^b$ each, mapped into the latent code space of AEs with these activation functions. The resulting sets $\mathcal{Z}^g$ and $\mathcal{Z}^b$ are colored in blue and orange, respectively. A large fraction of the *good* data points are mapped into an ellipsoid-shaped cluster by the tanh activation function, whereas using sigmoids maps them into a hyperbola-shaped cluster. Data from $\mathcal{Z}^g$ however show significant scatter around their respective clusters. *Bad* data are mapped onto band-like structures at the boundary of the image domain of the respective activation functions. Despite it is easy to separate the good and bad data in the embedding space, due to the saturation effect of the sigmoid and tanh activations, all the representations $\mathcal{Z}^b$ are squashed onto a very small manifold that cannot capture the variability of the data. This is of course detrimental since the bad data are the

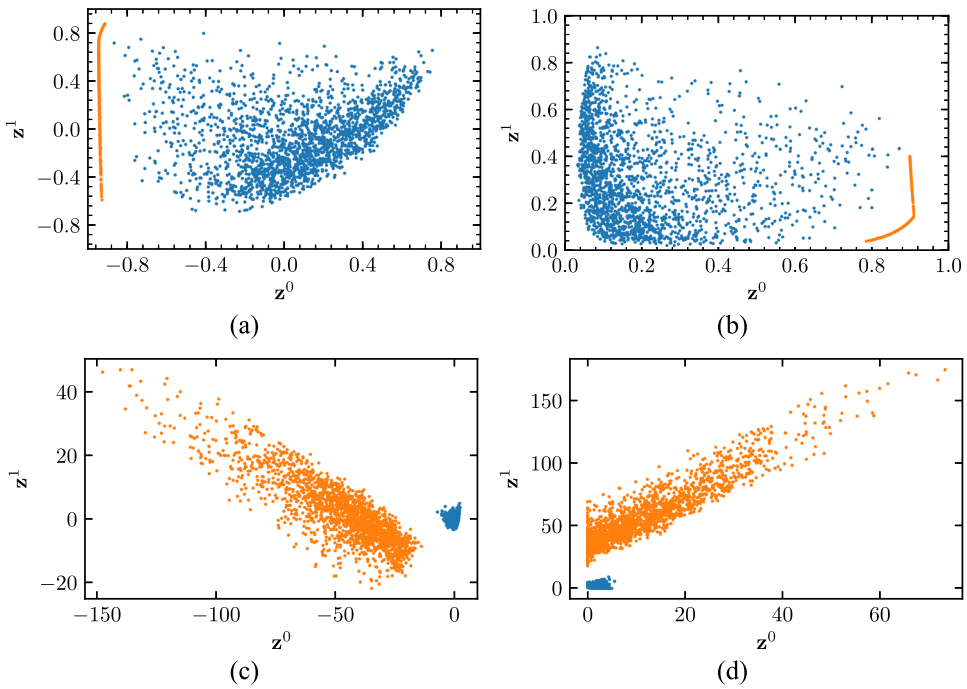

FIG. 6. *Good* (blue dots) and *bad* data (orange dots) in code space of AEs with $\mathcal{C} = \{12, 2, 12\}$ and different activation functions. (a) tanh, (b) sigmoid, (c) maxout, and (d) ReLU.

ones characterized by a greater variability that, however, cannot be modeled well in the current representation. Since the purpose of our analysis is to derive meaningful statistics about the uncertain data, a model that explores in detail only the region where the good data lie and treats equally everything outside this (small and less interesting) space is not useful for our analysis.

When using Maxout or ReLU activation functions instead, the AE maps *good* data points into a narrow cluster and scatters *bad* data points along band-like structures. The image domain of these activation functions has no upper bound, and the separation of the *good* and *bad* data is larger for Maxout and ReLU activation functions than it is for tanh and sigmoids. Most importantly, the good data are mapped in a small and dense region of the embedding space, while representations of the bad data $\mathcal{Z}^b$ are more scattered around. This kind of representation in agreement with our expectation that good data are similar and characterized by a lower variance allows us to explore in detail the space where the bad data are mapped and to draw decision boundaries that allow us to retrieve accurate and meaningful statistics. The ReLU activation function can be considered as a special case of the Maxout function. Since the results obtained by those two activation functions are qualitatively comparable and since Maxout introduces additional parameters, in the following, we only consider AEs using ReLU activation functions.

Codes produced by AEs with different layouts are qualitatively similar to those shown in Fig. 6. For AEs with $z = 3$, the data points usually feature only little variance along one of the three dimensions. That is, they cluster in a similar manner as they do for AEs with $z = 2$. Introducing an additional bottleneck layer in the AE, i.e., choosing $\mathcal{C} = \{12, 5, 2, 5, 12\}$, we observe a similar clustering of the data as is the case for $\mathcal{C} = \{12, 2, 12\}$. Postponing the effect produced by different configurations of $\mathcal{C}$ on the resulting statistics of the inlier T_e data, we continue by discussing the choice of a vector classifier in the code space $\mathcal{Z}$ of the AE.

Once an AE is trained, it defines a mapping from the input domain $\mathcal{X}$ into a unique, latent code space $\mathcal{Z}$. A classifier is trained on $\mathcal{Z}^g$ and $\mathcal{Z}^b$ and subsequently used to assign each $\mathbf{x} \in \mathcal{X}^u$ a label $\ell \in \{good, bad\}$. The set of all labels for the elements of $\mathcal{X}$ is denoted as $\mathcal{L}$. A label $\ell \in \mathcal{L}$ denotes whether a sample will be considered as an inlier or outlier, respectively. Such a classification introduces a bias, but with a validation procedure it is possible to evaluate how well it generalizes to unseen data and select the most suitable model accordingly.

Here we consider three standard classifiers for vectorial data: a support vector machine classifier (SVC), a nearest prototype classifier (PROT), and a so-called least-squares classifier (LSQ). The details of these classifiers and the settings of their hyperparameters in the experiments are discussed in the Appendix.

To train a classifier, data are partitioned into a training and a validation set, $\mathcal{Z}_{\text{tr}}$ and $\mathcal{Z}_{\text{val}}$. These sets contain only labelled samples: $\mathcal{Z}_{\text{tr}} = \{\mathcal{Z}^g_{\text{tr}} \cup \mathcal{Z}^b_{\text{tr}}\}$ and $\mathcal{Z}_{\text{val}} = \{\mathcal{Z}^g_{\text{val}} \cup \mathcal{Z}^b_{\text{val}}\}$. The *good* training and validation datasets contain 1000 random data points each and the *bad* training and validation datasets contain approximately half the *bad* data each. $\mathcal{Z}_{\text{tr}}$ is used to train the classifier, and $\mathcal{Z}_{\text{val}}$ is used to evaluate the generalization capability of the classifier. Figure 7 provides a schematic depiction of the pipeline to train the classifier.

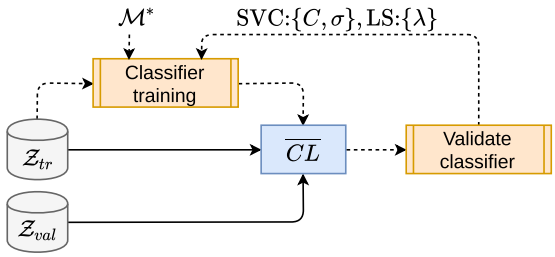

FIG. 7. Pipeline of the classifier training. The classifier $\overline{\text{CL}}$ trained on $\mathcal{Z}_{\text{tr}}$ is validated on $\mathcal{Z}_{val}$ to test its generalization capability and choose the hyperparameters (such as C and σ in the SVM case). The model of the classifier $\mathcal{M} \in \{\text{SVC}, \text{LS}, \text{PROT}\}$ is assumed to be given at this stage.

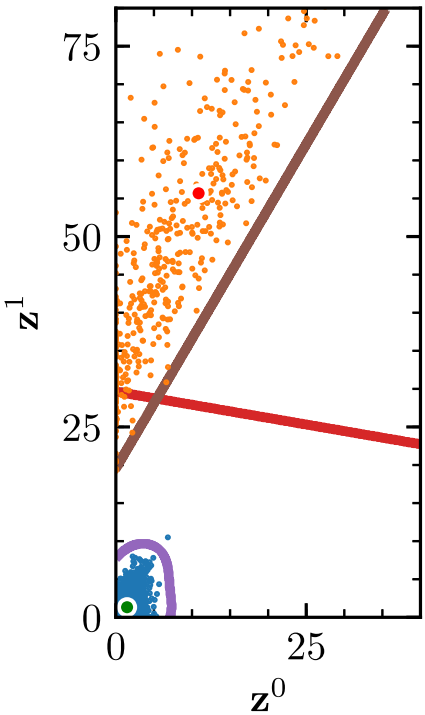

FIG. 8. Decision boundaries for a nearest prototype classifier (red line), a support vector machine classifier (purple line), and a least-square classifier (brown line). The red and green circles denote the class prototypes given by Eq. (A8). The blue dots denote data from $\mathcal{Z}^g_{\text{tr}}$, and the orange dots denote data from $\mathcal{Z}^b_{\text{tr}}$. The green and red circles denote the prototypes given by Eq. (A8).

The generalization capability of the classifier is quantified by the so-called F1 score. It is defined as the harmonic mean of precision and recall, as calculated for the validation data, and assumes a value between zero and one. Precision is defined as the ratio of correctly classified outliers and all correctly classified data points. Recall is defined as the ratio of correctly classified outliers and the number of all data points classified as outliers. An F1 score of zero describes a perfectly inaccurate classifier, and an F1 score of one describes a perfectly accurate classifier.

Figure 8 shows the decision boundaries learned by the three different classifiers as full lines. The training data used to learn the decision boundaries $\mathcal{Z}_{\mathrm{tr}}$ are indicated by the blue and orange dots. The SVC classifier, indicated by the purple line, draws a tight and curved decision boundary around $\mathcal{Z}^g_{\mathrm{tr}}$, and the least-square classifier, the full brown line, draws a tight, linear boundary around $\mathcal{Z}^b_{\mathrm{tr}}$. The decision boundary identified by the nearest prototype classifier, the red line, puts the decision boundary approximately half way between the class prototype. The F1-score of the classifiers are, respectively, given by 1.0, 1.0, and 0.97 for the shown data. This suggests that all three classifiers correctly label unseen data as either *good* or *bad*, that is, all three classifiers generalize equally well to unseen data.

Figure 9 shows an example of the classification process using the nearest prototype classifier. The leftmost panel shows the codes $\mathcal{Z}^g_{\mathrm{tr}}$ in blue dots and the codes $\mathcal{Z}^b_{\mathrm{tr}}$ in orange dots. The codes are clearly linearly separable, and there is a large leeway for placing the decision boundary. A nearest prototype classifier is fitted on $\mathcal{Z}_{\mathrm{tr}}$, and the prototypes μ_g and μ_b, as defined in Eq. (A8), are depicted by a green and red dot, respectively in the left panel. This classifier is subsequently used to assign class labels to the validation data $\mathcal{Z}^g_{\mathrm{val}}$ and $\mathcal{Z}^b_{\mathrm{val}}$, shown in the same color coding in the middle panel. Only few codes are mislabelled by the classifier, and its F1 score is approximately one. The rightmost panel shows the count of *uncertain* data codes $\mathcal{Z}_{\mathrm{u}}$ with assigned class labels.

Returning to the optimal configuration of the AE, we continue by discussing the statistics of all inlier samples $\mathcal{X}'^g = \mathcal{X}^g \cup \{\mathcal{X}^u | \mathcal{L}^u = good\}$ and outlier samples $\mathcal{X}'^b = \mathcal{X}^b \cup \{\mathcal{X}^u | \mathcal{L}^u = bad\}$, as identified by the proposed framework using the nearest prototype classifier. Figures 10(a)–10(d) show the average electron temperature and the relative error on the electron temperature for different *a priori* partitioning and different AE layouts. The numerals in the x-axis labels denote the AE layout $\mathcal{C}$ and staggered plot markers refer to data from the individual MLPs "NE," "SE," "SW," and "NW." The error bars denote the sample standard variation. For the inlier samples, $\overline{\mathrm{T}}_{\mathrm{e}}$ varies between 8 and 10 eV. This average shows little sensitivity to the used AE layout and the partition thresholds. There also appears a systematic difference in T_{e} as reported by the different probe heads. This may be due to shadowing of plasma flows, caused by the protruding probe head geometry. The plasma that is ballooned out at the outboard mid-plane will stream along the magnetic field lines. Following the field lines, it impinges first on the west electrodes. On the other hand, this discrepancy may also be due to a systematic error in the voltage measurements among electrodes due to slightly untuned capacitor bridges in the electronics.

The root-mean-square values of the T_{e} data are negligible for most $\mathcal{X}^g$, except for the $\mathcal{C} = \{12, 2, 12\}$ layout using *relaxed*

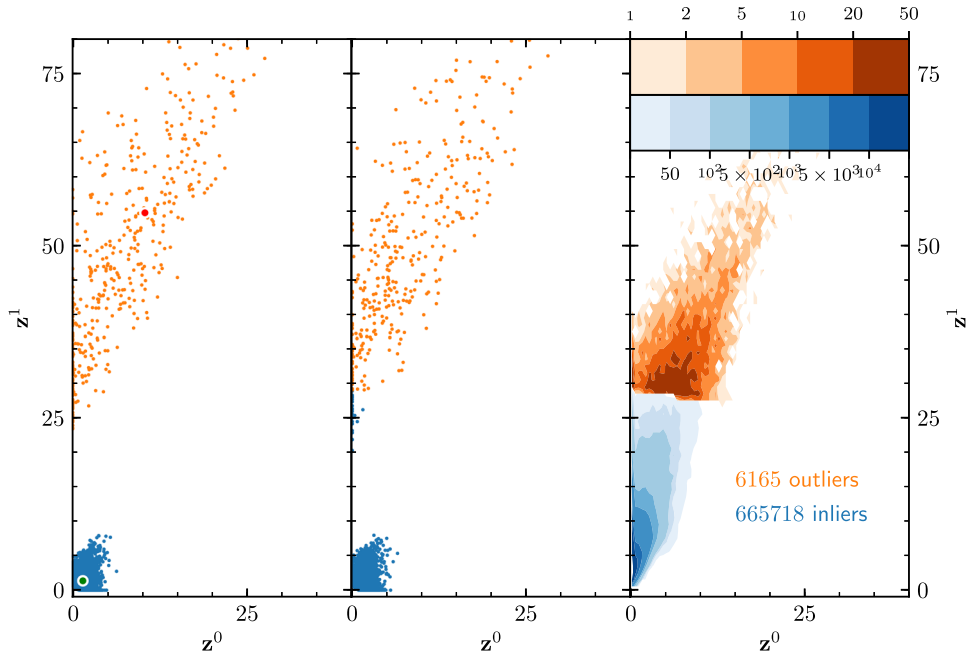

FIG. 9. Representation of the MLP data in code space of an AE with a single hidden layer of size z = 2. Blue denotes the valid data points, and orange denotes the invalid data points. The left and the middle panels, respectively, show training data $\mathcal{Z}_{\mathrm{tr}}$ and validation data $\mathcal{Z}_{\mathrm{val}}$ for the classifier [Eq. (A8)]. The right panel shows the count of data samples classified as either *good* (blue) or *bad* (orange).

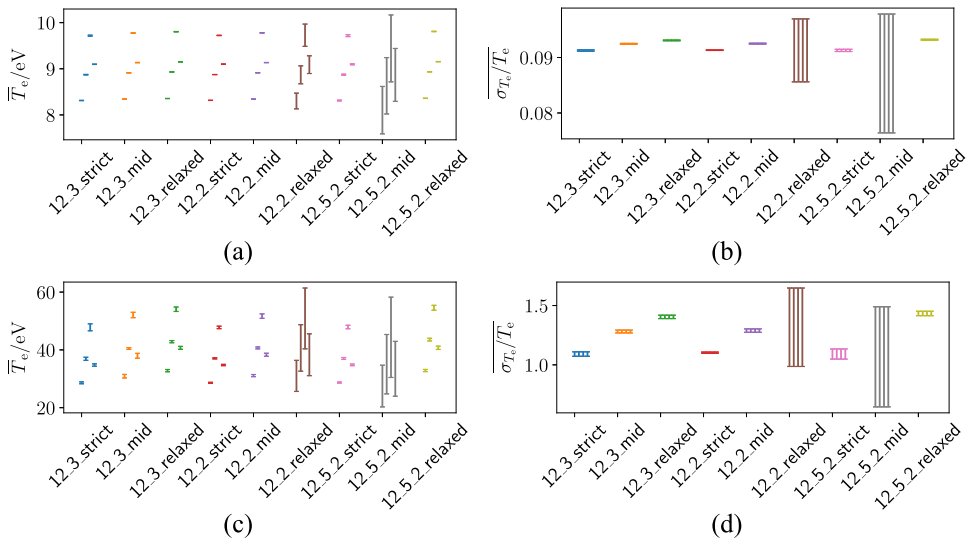

FIG. 10. Average electron temperature and relative error on the electron temperature for the inlier and outlier samples, as identified using different configurations of the AE and partition thresholds. (a) Average electron temperature using $\mathcal{X}'^{g}$, grouped by hyperparameters of the AE and MLP. (b) Average relative fit error on the electron temperature using $\mathcal{X}'^{g}$, grouped by hyperparameters of the AE and MLP. (c) Average electron temperature using $\mathcal{X}'^{b}$, grouped by hyperparameters of the AE and MLP. (d) Average relative fit error on the electron temperature using $\mathcal{X}'^{b}$, grouped by hyperparameters of the AE and MLP.

partition thresholds and the $\mathcal{C} = \{12, 5, 2, 5, 12\}$ layout using *mid* partition thresholds. This effect is due to randomness in the used input data for the AE training. For these cases, significant root mean square values in $\mathcal{X}'^{g}$ are seen. Data points classified as outliers, $\mathcal{X}'^{b}$, show average electron temperatures between approximately 30 and 50 eV. The relative error on these samples is given by approximately one. Again, the standard deviation of these samples is negligible in almost any AE configuration. This analysis suggests that the choice of a specific AE layout does not yield significantly different sample statistics of $\mathcal{X}'^{g}$. Therefore, we opt for the simplest configuration $\mathcal{C} = \{12, 2, 12\}$.

V. PERFORMANCE EVALUATION

In order to evaluate the performance of the proposed classification scheme, we continue by comparing the statistics of the final inlier dataset as well as the lower order statistical moments of the heat flux, as computed from these data. The final inlier datasets are denoted by $\mathcal{X}'^{g}$ and are identified using an SVC classifier, $\mathcal{X}'^{g}_{\text{SVC}}$, a least squares classifier, $\mathcal{X}'^{g}_{\text{lsq}}$, and a nearest prototype classifier, $\mathcal{X}'^{g}_{\text{pro}}$. We evaluate their performance by comparing the resulting statistics to those obtained from the entire dataset $\mathcal{X}$, the data without *a priori* outliers, $\mathcal{X} \setminus \mathcal{X}^{b}$, and the dataset of only *a priori* inliers, $\mathcal{X}^{g}$. Figure 11 illustrates the processing pipeline used to obtain these datasets. For the results presented here, an AE with ReLU activation functions and $\mathcal{C} = \{12, 2, 12\}$ is used.

Figure 12 shows the joint probability distribution function of the electron temperature and the relative error on the electron temperature as computed for these datasets. Here, T_e and σ_{T_e}/T_e denote the average value reported by all four MLPs. The entire dataset $\mathcal{X}$, shown in Fig. 12(a), features many samples with small to medium T_e, associated with small to medium σ_{T_e}/T_e. A non-negligible fraction of the samples however features large T_e values with $\sigma_{T_e}/T_e \gtrsim 1$. Considering only the good data, $\mathcal{X}^{g}$, shown in Fig. 12(b), all samples feature small T_e values and a negligible relative error. The joint probability distribution function (PDF) of the set $\mathcal{X} \setminus \mathcal{X}^{b}$ is similar to that of the set $\mathcal{X}$, but samples with $T_e \gtrsim 40$ eV are almost absent. This is due to the *strict* threshold values applied when removing $\mathcal{X}^{b}$.

Pruning the MLP data using an SVC classifier, $\mathcal{X}'^{g}_{\text{SVC}}$, shown in Fig. 12(d), the joint PDF appears similar in shape to the one for $\mathcal{X}^{g}$ [Fig. 12(b)]. Only samples with $T_e \lesssim 15$ eV, associated with $\sigma_{T_e}/T_e \lesssim 0.3$, are present. Removing the outliers identified by the nearest prototype classifier, $\mathcal{X}'^{g}_{\text{pro}}$, shown in Fig. 12(e), several samples with $T_e \gtrsim 50$ eV are present. However, all samples feature relative errors less than approximately 0.75. Qualitatively, this joint PDF is similar to the joint PDF for $\mathcal{X} \setminus \mathcal{X}^{b}$ [Fig. 12(c)], except that samples with large σ_{T_e}/T_e are missing. Employing a least squares classifier, $\mathcal{X}'^{g}_{\text{lsq}}$, shown in Fig. 12(f), the resulting joint PDF is approximately aligned along an equi-probability contour of the joint PDF for $\mathcal{X}$ [Fig. 12(a)]. There are no samples with $T_e \gtrsim 35$ eV and samples with $\sigma_{T_e}/T_e \gtrsim 1$ are also absent. Notably, samples $T_e \gtrsim 20$ eV with small σ_{T_e}/T_e are absent, while the

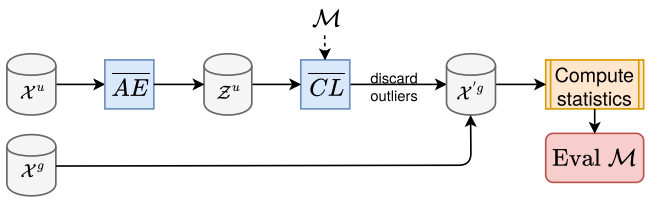

FIG. 11. Uncertain data $\mathcal{X}_u$ are processed by $\overline{AE}$ and $\overline{CL}$, trained as described in Sec. IV. To evaluate the classification model $\mathcal{M}$, we compare statistics on the classification results $\mathcal{X}'^{g}$.

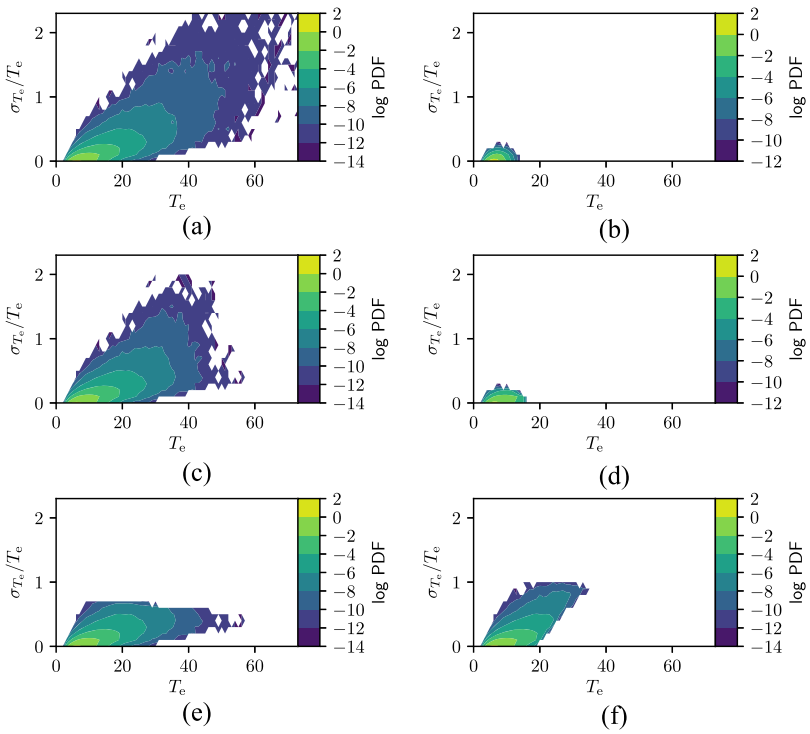

FIG. 12. Joint probability distribution function of the average electron temperature and the average relative error on the electron temperature after outliers have been removed by different methods. (a) All data, $\mathcal{X}$; (b) only *good* data, $\mathcal{X}^g$; (c) no *bad* data, $\mathcal{X} \setminus \mathcal{X}^b$; (d) $\mathcal{X}'^g_{\text{SVC}}$; (e) $\mathcal{X}'^g_{\text{pro}}$; and (f) $\mathcal{X}'^g_{\text{lsq}}$.

dataset still includes samples with $T_e \gtrsim 20$ eV and large values of σ_{T_e}/T_e.

Figure 13 shows the mapping of the labels $\mathcal{L}_{\text{te}}$, as identified by the nearest prototype classifier into the time domain. The black lines and the red dots denote $\mathcal{X}^g$ and $\mathcal{X}^b$, respectively. Blue dots mark samples from $\mathcal{Z}_u$ labelled ℓ = *good*, and orange dots mark samples from $\mathcal{Z}_u$ labelled ℓ = *bad*. The large amplitude fluctuations at 45.1 ms, at 45.9 ms, and at 46.6 ms are mostly classified as *good* data points. Notably, the peak at 46.2 ms is classified as *good*, even though the relative error and the range of the biasing voltage of this MLP are similar to the conditions of the preceding peak at 45.9 ms. This is due to the requirement that at least two MLPs need to report an invalid fit in order for a data point to be rejected.

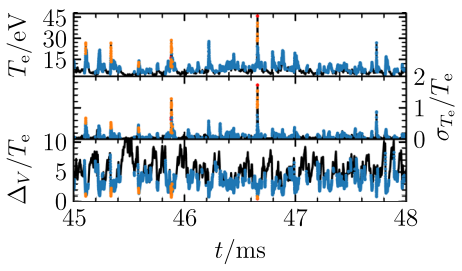

FIG. 13. Data time series of the north-east MLP (cf. blue lines in Fig. 3), overlaid with labels indicating classification of the data. Blue dots denote *good* samples, $\mathcal{X}'^g_{\text{pro}}$, orange dots denote *bad* samples, $\mathcal{X}'^b_{\text{pro}}$, and red crosses denote invalid samples $\mathcal{X}^b$, as classified by the prototype classifier using *strict* thresholds.

A unique capability of mirror Langmuir probes is that they allow us to study the fluctuation statistics of plasma flows driven by the electric drift. The heat flux impinging on plasma facing components is of special interest. It is composed of a conduction driven part, $\widehat{\Gamma}_{\text{T,cond}} = \widetilde{U}\widetilde{T}_e\langle n_e\rangle_{\text{mv}}/n_{\text{e,mrms}}$, a convection driven part $\widehat{\Gamma}_{\text{T,conv}} = \widetilde{U}\widetilde{n}_e\langle T_e\rangle_{\text{mv}}/T_{\text{e,mrms}}$, and contributions from triple correlations $\widehat{\Gamma}_{\text{T,tcor}} = \widetilde{U}\widetilde{n}_e\widetilde{T}_e$. Here $\widetilde{\cdot}$ denotes a quantity re-scaled by subtracting its moving average, $\langle\cdot\rangle_{\text{mv}}$, and by dividing its moving root-mean-square $\cdot_{\text{mrms}}$. In the following, we use a window length of 16 384 elements for these filters.[36]

Figure 14 shows the sample average and standard deviation for the three contributions of the radial heat flux, computed using different datasets and relative to the statistical moments computed ignoring *a priori* outliers $\mathcal{X} \setminus \mathcal{X}^b$. The average heat fluxes and the standard deviations are largest when using the entire dataset $\mathcal{X}$. Using only good data, $\mathcal{X}^g$, yields averages and standard deviations less than 25% of the values calculated using $\mathcal{X} \setminus \mathcal{X}^b$. Notably, for these data, the average radial heat flux due to triple correlations vanishes. Computing the moments using $\mathcal{X}'^g_{\text{SVC}}$, the average conductive and convective heat fluxes are approximately 50% and 60% of the reference values, while the average value of the contributions from triple correlations is approximately 20%.

Removing outlier data as identified by the nearest prototype classifier, $\mathcal{X}'^g_{\text{pro}}$, the average and root-mean-square values

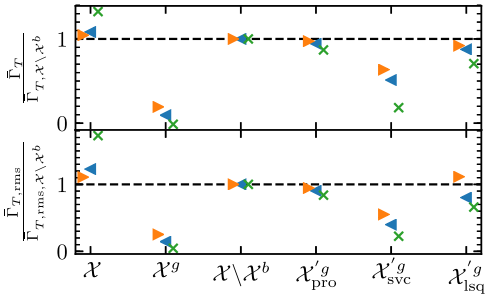

FIG. 14. Radial heat flux due to conduction (triangle left), convection (triangle right), and triple correlations (cross), as computed from the various datasets and relative to the reference values computed using $\mathcal{X}\setminus\mathcal{X}^b$. The upper panel shows the sample average, and the lower panel shows the sample standard deviation.

of the heat fluxes are approximately 85%–95% of the reference value. Finally, using the least squares classifier results in statistical moments of the heat flux comparable to those using the reference case $\mathcal{X}\setminus\mathcal{X}^b$.

The difference in the sample averages and standard deviations of the various heat flux contributions can be related to the shape of the joint PDFs shown in Fig. 12. For this, we note that the heat flux is computed from T_e, n_e, and V_p samples. The relative error on n_e is given by the geometric mean of the relative errors on I_{sat} and T_e. As discussed in Sec. II, σ_{T_e}/T_e and $\sigma_{I_{sat}}/I_{sat}$ are strongly correlated. That is, a larger relative error on σ_{T_e}/T_e implies a large relative error on the electron density.

Comparing the joint PDFs in Fig. 12, it is obvious that employing the different classifiers to remove outliers introduces slightly different biases into the inlier dataset.

VI. CONCLUSION

In conclusion, we propose a framework to classify outlier data in data time series sampled by a group of mirror Langmuir probes in scrape-off layer plasmas. An autoencoder is trained to identify a low-dimensional representation of *good* fit data from this group of probes. In this space, each dimension corresponds to a combination of features which best characterizes the measurements. These are determined by the AE from the training data and without making any *a priori* assumption about the dataset at hand. Outlier data, which do not share the characteristics of *good* data, appear in a separable cluster in the space of the AE. Several classifiers are trained to separate outlier data in this space. With no ground truth available, the performance of the classifiers is evaluated by comparing the lower order statistical moments of the radial electron heat flux.

Using either a least squares or a nearest prototype classifier results in similar statistics of the radial heat flux as obtained when using a threshold classifier to identify outliers. Average contributions of the conductive and convective radial heat flux obtained by these classifiers fall approximately 3% and 14% below the values obtained by applying a threshold. On the other hand, the contribution due to triple correlations falls up to 40% below the value obtained from the thresholding method. These differences result from the different characteristics of the data points which are identified as outliers. While the least squares classifier places the decision boundary close to the outlier data cluster, the nearest prototype classifier places the decision boundary approximately equidistant to both clusters. That is, the least squares classifier gives a more relaxed outlier removal, while the nearest prototype classifier has a lower threshold. The support vector machine (SVM) classifier puts the separating hyperplane in an area characterized by a high concentration of data samples. However, it is well known that when a classifier puts its decision boundary in an area of high density, it achieves low generalization capabilities on unseen data and is more prone to classification error.[52] In our case, an undesired behaviour is evident from the reported statistics in Figs. 12 and 14. This also explains why the SVM classifier results in fluxes that are only marginally larger than the fluxes obtained when only considering good data.

While neither the least squares nor the nearest prototype method can be identified as the correct method to remove outliers from the dataset, this study implies that not employing outlier removal may lead to heat fluxes, over-estimated by a significant amount.

The framework proposed here may also be adapted to other types of sensors than MLPs. The requirements for applying the method described here are as follows: First, any single sensor reports a physical quantity together with an uncertainty of that measurement. Second, any sensor in the group needs to sample roughly the same environment.

ACKNOWLEDGMENTS

This work was supported with financial subvention from the Research Council of Norway under Grant No. 240510/F20 and the U.S. Department of Energy, Office of Science, Office of Fusion Energy Sciences, using User Facility Alcator C-Mod, under Award No. DE-FC02-99ER54512-CMOD. F.M.B. is funded by the Research Council of Norway under FRIPRO Grant No. 239844 "*Next Generation Learning Machines*". R.K. acknowledges the generous hospitality of the MIT Plasma Science and Fusion Center where parts of this work were conducted.

APPENDIX: CLASSIFIERS

Classifiers are algorithms that assign a class label ℓ to new data, on the basis of previously seen training data with known class labels. For the case of two distinct classes, we assign $\ell = \pm 1$ to inliers and outliers, respectively. In the following, we describe how labels ℓ_i for new data $\{x_i\}, x_i \in \mathbb{R}^d$ are retrieved using the classifiers used in this paper.

1. Support vector machine

A Support Vector Machine (SVM) learns a linear classifier in a kernel space[53]

$$\ell_i = g(x_i) = \text{sign}(\phi(w) \cdot \phi(x_i) + b), \tag{A1}$$

induced by a usually non-linear kernel $\phi(w) \cdot \phi(x) = K(w, x)$. A typical choice for K is the radial basis function, defined as

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right),$$

where σ is called the bandwidth of the kernel.

In order to train a SVM, the cost function

$$\phi^*(w) = \arg\min_{\phi(w)} \frac{1}{2}\|\phi(w)\|^2$$

is minimized under the constraint that $y_i(\phi(w) \cdot \phi(x_i) + b) \geq 1$. That is, the training data are taken to be only inlier data with $\ell_i = +1$.

The constraints can be included in the previous quadratic cost by using the Lagrangian multipliers

$$L(\phi(w), b, \alpha) = \frac{1}{2}||\phi(w)||^2 - \sum_i \alpha_i(y_i(\phi(w) \cdot \phi(x_i) + b) - 1). \tag{A2}$$

It follows that the weight vectors become a linear combination of the data points

$$\phi(w) = \sum_i y_i \alpha_i \phi(x_i), \tag{A3}$$

and the classifier can be expressed as

$$\begin{aligned} g(x) &= \text{sign}\left(\sum_i y_i \alpha_i \phi(x_i) \cdot \phi(x) + b\right) \\ &= \text{sign}\left(\sum_i y_i \alpha_i K(x_i, x) + b\right). \end{aligned} \tag{A4}$$

If we substitute (A4) into (A2), we obtain the following dual cost function:

$$\begin{aligned} W(\alpha) &= \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} y_i y_j \alpha_i \alpha_j \phi(x_i) \cdot \phi(x_j) \\ &= \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} y_i y_j \alpha_i \alpha_j K(x_i, x_j), \end{aligned} \tag{A5}$$

and the optimization now reads

$$\begin{aligned} \hat{\alpha} &= \arg\max_{\alpha} W(\alpha) \\ \text{such that} \quad & \alpha_i \geq 0. \end{aligned} \tag{A6}$$

Once the training is complete, new points are classified directly by applying (A4).

The most important hyperparameter in the SVC classifier is the kernel width σ. A commonly used approach is to set σ heuristically according to Silverman's rule, which reads

$$\sigma = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}}, \tag{A7}$$

where $\hat{\sigma}$ is the empirical standard deviation of the samples' features and n is the number of samples in the training data.

2. Prototype classifier

Classification by means of simple a nearest prototype classifier operates as follows. For each class c, a prototype is computed as[54]

$$\mu_c = \frac{1}{|\{x_i\}|}\sum_i x_i. \tag{A8}$$

The class label ℓ of an uncategorized data sample z is assigned as

$$\ell = \arg\min_c \|z - \mu_c\|^2. \tag{A9}$$

This classifier does not depend on any hyperparameter and requires to maintain only the representative of each cluster to classify out-of-sample data. Due to its simplicity, this classifier cannot identify complex decision boundaries to separate samples of different classes.

3. Least squares classifier

A classification function f is learned by minimizing the following quadratic cost:

$$\min \frac{1}{N}\sum_{i=1}^{N}\|\ell_i - f(x_i)\|^2 + \lambda\|f\|^2, \tag{A10}$$

where the first term represents the discrepancy between the output class of the function and the known class of the training data. The second cost term instead encourages smoothness in the target function and is useful to prevent overfitting.

In the analysis presented here, we choose f to be a linear function

$$f(x) = Wx + b$$

whose parameters W and b are optimized according to (A10).

A more flexible choice consists in using a kernel function to define f

$$f(x) = \sum_{i=1}^{N} c_i K(x, x_i).$$

In this case, the objective of the optimization is to find the parameters c_i. This is done by modifying the quadratic loss in (A10), which becomes

$$\min_{\mathbf{c}} \frac{1}{N}\sum_{i=1}^{N}\|y_i - \sum_{j=1}^{N} c_j K(x, x_j)\|^2 + \lambda\|\mathbf{c}^T\mathbf{K}\mathbf{c}\|^2. \tag{A11}$$

The parameter λ has been set to 1 in the experiments.

REFERENCES

[1] J. Wesson, *Tokamaks*, 3rd ed. (Wesson, 2004).

[2] G. Y. Antar, S. I. Krasheninnikov, P. Devynck, R. P. Doerner, E. M. Hollmann, J. A. Boedo, S. C. Luckhardt, and R. W. Conn, "Experimental evidence of intermittent convection in the edge of magnetic confinement devices," Phys. Rev. Lett. **87**, 065001 (2001).

[3] G. Y. Antar, G. Counsell, Y. Yu, B. LaBombard, and P. Devynck, "Universality of intermittent convective transport in the scrape-off layer of magnetically confined devices," Phys. Plasmas **10**, 419–428 (2003).

[4]Y. H. Xu, S. Jachmich, R. R. Weynants, and TEXTOR Team, "On the properties of turbulence intermittency in the boundary of the textor tokamak," Plasma Phys. Controlled Fusion **47**, 1841 (2005).
[5]J. P. Graves, J. Horacek, R. A. Pitts, and K. I. Hopcraft, "Self-similar density turbulence in the TCV tokamak scrape-off layer," Plasma Phys. Controlled Fusion **47**, L1 (2005).
[6]O. E. Garcia, S. M. Fritzner, R. Kube, I. Cziegler, B. LaBombard, and J. L. Terry, "Intermittent fluctuations in the alcator C-MOD scrape-off layer," Phys. Plasmas **20**, 055901 (2013).
[7]J. L. Terry, S. J. Zweben, K. Hallatschek, B. LaBombard, R. J. Maqueda, B. Bai, C. J. Boswell, M. Greenwald, D. Kopon, W. M. Nevins, C. S. Pitcher, B. N. Rogers, D. P. Stotler, and X. Q. Xu, "Observations of the turbulence in the scrape-off-layer of alcator C-MOD and comparisons with simulation," Phys. Plasmas **10**, 1739–1747 (2003).
[8]S. Zweben, R. Maqueda, D. Stotler, A. Keesee, J. Boedo, C. Bush, S. Kaye, B. LeBlanc, J. Lowrance, V. Mastrocola, R. Maingi, N. Nishino, G. Renda, D. Swain, J. Wilgen, and NSTX Team, "High-speed imaging of edge turbulence in NSTX," Nucl. Fusion **44**, 134 (2004).
[9]J. Terry, N. Basse, I. Cziegler, M. Greenwald, O. Grulke, B. LaBombard, S. Zweben, E. Edlund, J. Hughes, L. Lin, Y. Lin, M. Porkolab, M. Sampsell, B. Veto, and S. Wukitch, "Transport phenomena in the edge of alcator C-MOD plasmas," Nucl. Fusion **45**, 1321 (2005).
[10]M. Agostini, J. Terry, P. Scarin, and S. Zweben, "Edge turbulence in different density regimes in alcator C-MOD experiment," Nucl. Fusion **51**, 053020 (2011).
[11]R. Kube, O. Garcia, B. LaBombard, J. Terry, and S. Zweben, "Blob sizes and velocities in the alcator C-MOD scrape-off layer," J. Nucl. Mater. **438**(Suppl.), S505–S508 (2013).
[12]G. Federici, C. Skinner, J. Brooks, J. Coad, C. Grisolia, A. Haasz, A. Hassanein, V. Philipps, C. Pitcher, J. Roth, W. Wampler, and D. Whyte, "Plasma-material interactions in current tokamaks and their implications for next step fusion reactors," Nucl. Fusion **41**, 1967 (2001).
[13]D. Whyte, "On the consequences of neutron induced damage for volumetric fuel retention in plasma facing materials," in *Proceedings of the 18th International Conference on Plasma-Surface Interactions in Controlled Fusion Device* [J. Nucl. Mater. **390–391**, 911–915 (2009)].
[14]I. H. Hutchinson, *Principles of Plasma Diagnostics* (Cambridge University Press, 2002).
[15]P. C. Stangeby, *The Plasma Boundary of Magnetic Fusion Devices* (IOP Publishing, 2000).
[16]V. Rohde, "Langmuir probe measurements in the midplane of ASDEX-upgrade," Contrib. Plasma Phys. **36**, 109–115 (1996).
[17]J. A. Boedo, D. Rudakov, R. Moyer, S. Krasheninnikov, D. Whyte, G. McKee, G. Tynan, M. Schaffer, P. Stangeby, P. West, S. Allen, T. Evans, R. Fonck, E. Hollmann, A. Leonard, A. Mahdavi, G. Porter, M. Tillack, and G. Antar, "Transport by intermittent convection in the boundary of the DIII-D tokamak," Phys. Plasmas **8**, 4826–4833 (2001).
[18]J. A. Boedo, D. L. Rudakov, R. A. Moyer, G. R. McKee, R. J. Colchin, M. J. Schaffer, P. G. Stangeby, W. P. West, S. L. Allen, T. E. Evans, R. J. Fonck, E. M. Hollmann, S. Krasheninnikov, A. W. Leonard, W. Nevins, M. A. Mahdavi, G. D. Porter, G. R. Tynan, D. G. Whyte, and X. Xu, "Transport by intermittency in the boundary of the DIII-D tokamak," Phys. Plasmas **10**, 1670–1677 (2003).
[19]G. S. Kirnev, V. P. Budaev, S. A. Grashin, E. V. Gerasimov, and L. N. Khimchenko, "Intermittent transport in the plasma periphery of the T-10 tokamak," Plasma Phys. Controlled Fusion **46**, 621 (2004).
[20]O. E. Garcia, J. Horacek, R. A. Pitts, A. H. Nielsen, W. Fundamenski, J. P. Graves, V. Naulin, and J. J. Rasmussen, "Interchange turbulence in the TCV scrape-off layer," Plasma Phys. Controlled Fusion **48**, L1 (2006).
[21]J. Horacek, J. Adamek, H. Müller, J. Seidl, A. Nielsen, V. Rohde, F. Mehlmann, C. Ionita, E. Havlíčková, and ASDEX Upgrade Team, "Interpretation of fast measurements of plasma potential, temperature and density in sol of ASDEX upgrade," Nucl. Fusion **50**, 105001 (2010).
[22]O. E. Garcia, R. Kube, A. Theodorsen, J.-G. Bak, S.-H. Hong, H.-S. Kim, KSTAR Project Team, and R. Pitts, "SOL width and intermittent fluctuations in KSTAR," Nucl. Mater. Energy **12**, 36 (2017).
[23]R. Kube, A. Theodorsen, O. E. Garcia, B. LaBombard, and J. L. Terry, "Fluctuation statistics in the scrape-off layer of alcator C-MOD," Plasma Phys. Controlled Fusion **58**, 054001 (2016).
[24]A. Theodorsen, O. E. Garcia, J. Horacek, R. Kube, and R. A. Pitts, "Scrape-off layer turbulence in TCV: Evidence in support of stochastic modelling," Plasma Phys. Controlled Fusion **58**, 044006 (2016).
[25]H. Müller, J. Adamek, J. Horacek, C. Ionita, F. Mehlmann, V. Rohde, R. Schrittwieser, and AU Team, "Towards fast measurement of the electron temperature in the sol of ASDEX upgrade using swept Langmuir probes," Contrib. Plasma Phys. **50**, 847–853 (2010).
[26]P. Verplancke, R. Chodura, J. Noterdaeme, and M. Weinlich, "Characteristics of a Langmuir probe in a magnetic field with high sweep frequencies," Contrib. Plasma Phys. **36**, 145–150 (1996).
[27]B. Labombard and L. Lyons, "Mirror Langmuir probe: A technique for real-time measurement of magnetized plasma conditions using a single Langmuir electrode," Rev. Sci. Instrum. **78**, 073501-1–073501-9 (2007).
[28]B. LaBombard, T. Golfinopoulos, J. L. Terry, D. Brunner, E. Davis, M. Greenwald, and J. W. Hughes, "New insights on boundary plasma turbulence and the quasi-coherent mode in alcator C-MOD using a mirror Langmuir probe," Phys. Plasmas **21**, 056108 (2014).
[29]B. LaBombard, R. L. Boivin, M. Greenwald, J. Hughes, B. Lipschultz, D. Mossessian, C. S. Pitcher, J. L. Terry, S. J. Zweben, and Alcator C-MOD Group (Alcator Group), "Particle transport in the scrape-off layer and its relationship to discharge density limit in alcator C-MOD," Phys. Plasmas **8**, 2107–2117 (2001).
[30]M. Greenwald, "Density limits in toroidal plasmas," Plasma Phys. Controlled Fusion **44**, R27 (2002).
[31]O. E. Garcia, J. Horacek, R. A. Pitts, A. H. Nielsen, W. Fundamenski, V. Naulin, and J. J. Rasmussen, "Fluctuations and transport in the TCV scrape-off layer," Nucl. Fusion **47**, 667 (2007).
[32]D. Carralero, G. Birkenmeier, H. Müller, P. Manz, P. deMarne, S. Müller, F. Reimold, U. Stroth, M. Wischmeier, E. Wolfrum, and TAU Team, "An experimental investigation of the high density transition of the scrape-off layer transport in ASDEX upgrade," Nucl. Fusion **54**, 123005 (2014).
[33]I. H. Hutchinson, R. Boivin, F. Bombarda, P. Bonoli, S. Fairfax, C. Fiore, J. Goetz, S. Golovato, R. Granetz, M. Greenwald, S. Horne, A. Hubbard, J. Irby, B. LaBombard, B. Lipschultz, E. Marmar, G. McCracken, M. Porkolab, J. Rice, J. Snipes, Y. Takase, J. Terry, S. Wolfe, C. Christensen, D. Garnier, M. Graf, T. Hsu, T. Luke, M. May, A. Niemczewski, G. Tinios, J. Schachter, and J. Urbahn, "First results from Alcator-C-MOD*," Phys. Plasmas **1**, 1511–1518 (1994).
[34]M. Greenwald, A. Bader, S. Baek, H. Barnard, W. Beck, W. Bergerson, I. Bespamyatnov, M. Bitter, P. Bonoli, M. Brookman, D. Brower, D. Brunner, W. Burke, J. Candy, M. Chilenski, M. Chung, M. Churchill, I. Cziegler, E. Davis, G. Dekow, L. Delgado-Aparicio, A. Diallo, W. Ding, A. Dominguez, R. Ellis, P. Ennever, D. Ernst, I. Faust, C. Fiore, E. Fitzgerald, T. Fredian, O. Garcia, C. Gao, M. Garrett, T. Golfinopoulos, R. Granetz, R. Groebner, S. Harrison, R. Harvey, Z. Hartwig, K. Hill, J. Hillairet, N. Howard, A. Hubbard, J. Hughes, I. Hutchinson, J. Irby, A. James, A. Kanojia, C. Kasten, J. Kesner, C. Kessel, R. Kube, B. LaBombard, C. Lau, J. Lee, K. Liao, Y. Lin, B. Lipschultz, Y. Ma, E. Marmar, P. McGibbon, O. Meneghini, D. Mikkelsen, D. Miller, R. Mumgaard, R. Murray, R. Ochoukov, G. Olynyk, D. Pace, S. Park, R. Parker, Y. Podpaly, M. Porkolab, M. Preynas, I. Pusztai, M. Reinke, J. Rice, W. Rowan, S. Scott, S. Shiraiwa, J. Sierchio, P. Snyder, B. Sorbom, V. Soukhanovskii, J. Stillerman, L. Sugiyama, C. Sung, D. Terry, J. Terry, C. Theiler, N. Tsujii, R. Vieira, J. Walk, G. Wallace, A. White, D. Whyte, J. Wilson, S. Wolfe, K. Woller, G. Wright, J. Wright, S. Wukitch, G. Wurden, P. Xu, C. Yang, and S. Zweben, "Overview of experimental results and code validation activities at Alcator C-MOD," Nucl. Fusion **53**, 104004 (2013).
[35]M. Greenwald, A. Bader, S. Baek, M. Bakhtiari, H. Barnard, W. Beck, W. Bergerson, I. Bespamyatnov, P. Bonoli, D. Brower, D. Brunner, W. Burke, J. Candy, M. Churchill, I. Cziegler, A. Diallo, A. Dominguez, B. Duval, E. Edlund, P. Ennever, D. Ernst, I. Faust, C. Fiore, T. Fredian, O. Garcia, C. Gao, J. Goetz, T. Golfinopoulos, R. Granetz, O. Grulke, Z. Hartwig, S. Horne, N. Howard, A. Hubbard, J. Hughes, I. Hutchinson, J. Irby, V. Izzo, C. Kessel, B. LaBombard, C. Lau, C. Li, Y. Lin,

B. Lipschultz, A. Loarte, E. Marmar, A. Mazurenko, G. McCracken, R. McDermott, O. Meneghini, D. Mikkelsen, D. Mossessian, R. Mumgaard, J. Myra, E. Nelson-Melby, R. Ochoukov, G. Olynyk, R. Parker, S. Pitcher, Y. Podpaly, M. Porkolab, M. Reinke, J. Rice, W. Rowan, A. Schmidt, S. Scott, S. Shiraiwa, J. Sierchio, N. Smick, J. A. Snipes, P. Snyder, B. Sorbom, J. Stillerman, C. Sung, Y. Takase, V. Tang, J. Terry, D. Terry, C. Theiler, A. Tronchin-James, N. Tsujii, R. Vieira, J. Walk, G. Wallace, A. White, D. Whyte, J. Wilson, S. Wolfe, G. Wright, J. Wright, S. Wukitch, and S. Zweben, "20 years of research on the Alcator C-MOD tokamak," Phys. Plasmas **21**, 110501 (2014).

[36]R. Kube, O. E. Garcia, A. Theodorsen, D. Brunner, A. Q. Kuang, B. LaBombard, and J. L. Terry, "Intermittent electron density and temperature fluctuations and associated fluxes in the Alcator C-MOD scrape-off layer," Plasma Phys. Controlled Fusion **60**, 065002 (2018).

[37]D. Brunner, A. Q. Kuang, B. LaBombard, and W. Burke, "Linear servomotor probe drive system with real-time self-adaptive position control for the Alcator C-MOD tokamak," Rev. Sci. Instrum. **88**, 073501 (2017).

[38]G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," Science **313**, 504 (2006).

[39]D. P. Kingma and M. Welling, "Auto-encoding variational bayes," preprint arXiv:1312.6114 (2013).

[40]A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," preprint arXiv:1511.05644 (2015).

[41]F. M. Bianchi, K. Ø. Mikalsen, and R. Jenssen, "Learning compressed representations of blood samples time series with missing data," in *Proceedings of the 2018 European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning* (ESANN 2018), *Bruges, Belgium*, 25-27 *April* 2018 (UCL/ELEN, 2018).

[42]Y. Bengio, "Learning deep architectures for AI," Found. Trends Mach. Learn. **2**, 1–127 (2009).

[43]P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," J. Mach. Learn. Res. **11**, 3371–3408 (2010).

[44]Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," IEEE Trans. Pattern Anal. Mach. Intell. **35**, 1798–1828 (2013).

[45]N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," J. Mach. Learn. Res. **15**, 1929–1958 (2014).

[46]J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Artificial Neural Networks and Machine Learning–ICANN 2011*, edited by T. Honkela, W. Duch, M. Girolami, and S. Kaski (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011), pp. 52–59.

[47]F. M. Bianchi, E. Maiorino, M. C. Kampffmeyer, A. Rizzi, and R. Jenssen, *Recurrent Neural Networks for Short-Term Load Forecasting: An Overview and Comparative Analysis* (Springer, 2017).

[48]I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio, "Maxout networks," in Proceedings of 30th International Conference on Machine Learning (ICML), 2013.

[49]V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Comput. Surv. **41**, 1–58 (2009).

[50]Z. Shuangfei, C. Yu, L. Weining, and Z. Zhongfei, "Deep structured energy based models for anomaly detection," in Proceedings of the 33rd International Conference on Machine Learning, Proceedings of Machine Learning Research, 2016.

[51]M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in Proceedings of the 2nd Workshop on Machine Learning for Sensory Data Analysis, 2014.

[52]M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," J. Mach. Learn. Res. **7**, 2399 (2006).

[53]V. N. Vapnik, "An overview of statistical learning theory," IEEE Trans. Neural Networks **10**, 988–999 (1999).

[54]J. C. Bezdek and L. I. Kuncheva, "Nearest prototype classifier designs: An experimental study," Int. J. Intell. Syst. **16**, 1445–1473 (2001).