IAEA nuclear fusion

PAPER • OPEN ACCESS

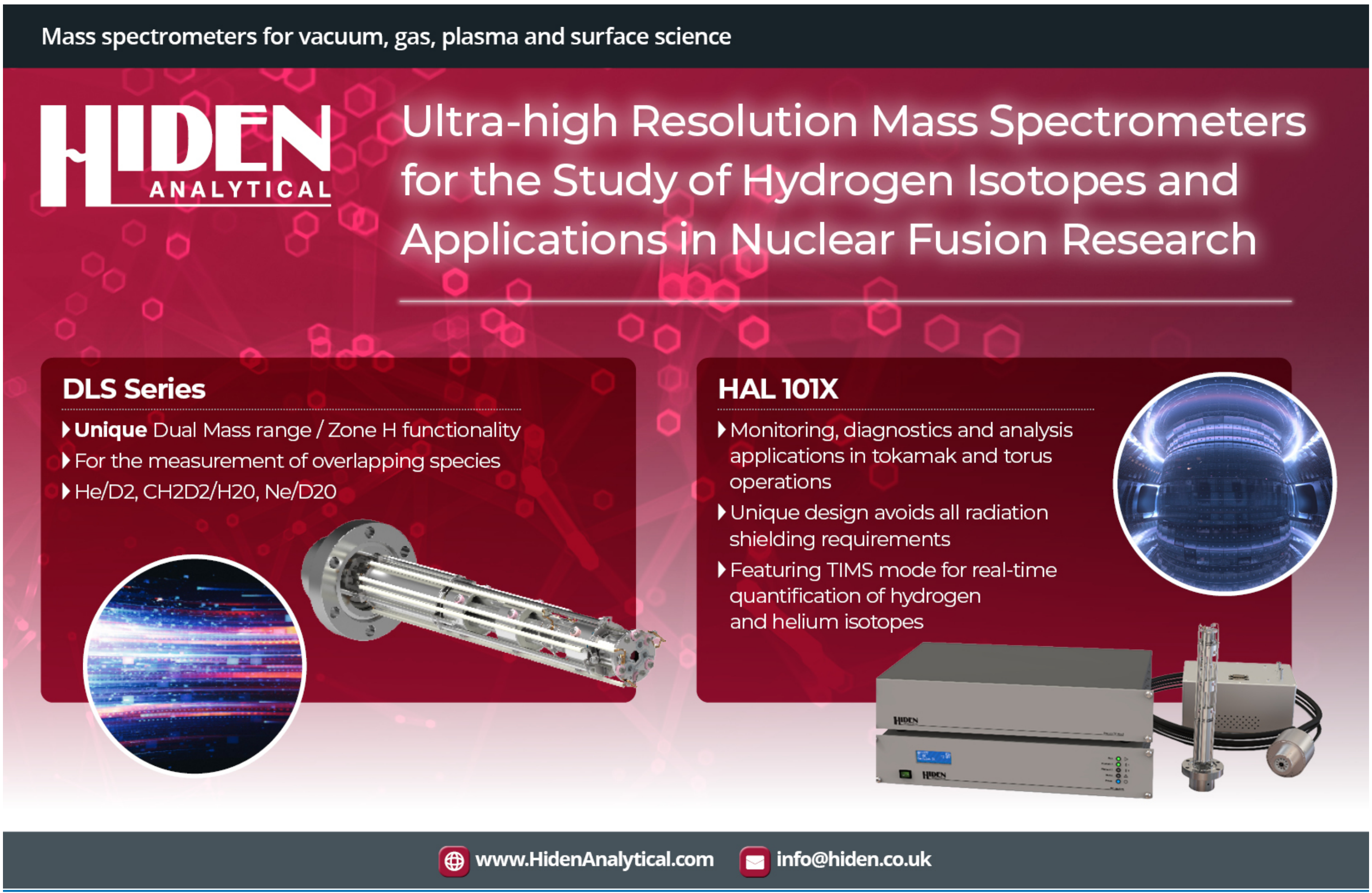
Efficient training sets for surrogate models of tokamak turbulence with Active Deep Ensembles

To cite this article: L. Zanisi *et al* 2024 *Nucl. Fusion* **64** 036022

View the article online for updates and enhancements.

You may also like

- Flux-driven integrated modelling of main ion pressure and trace tungsten transport in ASDEX Upgrade
O. Linder, J. Citrin, G.M.D. Hogeweij et al.
- Core turbulent transport in tokamak plasmas: bridging theory and experiment with QuaLiKiz
C Bourdelle, J Citrin, B Baiocchi et al.
- Multiple-isotope pellet cycles captured by turbulent transport modelling in the JET tokamak
M. Marin, J. Citrin, L. Garzotti et al.



Mass spectrometers for vacuum, gas, plasma and surface science

HIDEN ANALYTICAL

Ultra-high Resolution Mass Spectrometers for the Study of Hydrogen Isotopes and Applications in Nuclear Fusion Research

DLS Series
- **Unique** Dual Mass range / Zone H functionality
- For the measurement of overlapping species
- He/D2, CH2D2/H20, Ne/D20

HAL 101X
- Monitoring, diagnostics and analysis applications in tokamak and torus operations
- Unique design avoids all radiation shielding requirements
- Featuring TIMS mode for real-time quantification of hydrogen and helium isotopes

www.HidenAnalytical.com info@hiden.co.uk

This content was downloaded from IP address 194.81.223.66 on 18/11/2025 at 11:16

OPEN ACCESS
IOP Publishing | International Atomic Energy Agency
Nuclear Fusion
Nucl. Fusion **64** (2024) 036022 (22pp)
https://doi.org/10.1088/1741-4326/ad240d

Efficient training sets for surrogate models of tokamak turbulence with Active Deep Ensembles

L. Zanisi[1,*], A. Ho[2,3], J. Barr[4,5], T. Madula[4,5], J. Citrin[2,3], S. Pamela[1], J. Buchanan[1], F.J. Casson[1], V. Gopakumar[1,5] and JET Contributors[a]

[1] United Kingdom Atomic Energy Authority, Culham Centre for Fusion Energy, Abingdon, United Kingdom of Great Britain and Northern Ireland
[2] DIFFER—Dutch Institute for Fundamental Energy Research, Eindhoven, Netherlands
[3] Science and Technology of Nuclear Fusion Group, Eindhoven University of Technology, Eindhoven, Netherlands
[4] UCL's Centre for Doctoral Training in Data Intensive Sciences, University College London, London, United Kingdom of Great Britain and Northern Ireland
[5] University College London, London, United Kingdom of Great Britain and Northern Ireland

E-mail: lorenzo.zanisi@ukaea.uk

Received 12 October 2023, revised 22 December 2023
Accepted for publication 30 January 2024
Published 13 February 2024

Abstract

Model-based plasma scenario development lies at the heart of the design and operation of future fusion powerplants. Including turbulent transport in integrated models is essential for delivering a successful roadmap towards operation of ITER and the design of DEMO-class devices. Given the highly iterative nature of integrated models, fast machine-learning-based surrogates of turbulent transport are fundamental to fulfil the pressing need for faster simulations opening up pulse design, optimization, and flight simulator applications. A significant bottleneck is the generation of suitably large training datasets covering a large volume in parameter space, which can be prohibitively expensive to obtain for higher fidelity codes. In this work, we propose ADEPT (Active Deep Ensembles for Plasma Turbulence), a physics-informed, two-stage Active Learning strategy to ease this challenge. Active Learning queries a given model by means of an acquisition function that identifies regions where additional data would improve the surrogate model. We provide a benchmark study using available data from the literature for the QuaLiKiz quasilinear transport model. We demonstrate quantitatively that the physics-informed nature of the proposed workflow reduces the need to perform simulations in stable regions of the parameter space, resulting in significantly improved data efficiency compared to non-physics informed approaches which consider a regression problem over the whole domain. We show an up to a factor of 20 reduction in training dataset size needed to achieve the same performance as random sampling. We then validate the surrogates on multichannel integrated modelling of ITG-dominated JET scenarios and demonstrate that they recover the performance of QuaLiKiz to better than 10%. This matches the performance obtained in previous work, but with two orders of magnitude fewer training data points.

[a] See Mailloux *et al* 2022 (https://doi.org/10.1088/1741-4326/ac47b4) for JET Contributors.
[*] Author to whom any correspondence should be addressed.

Original Content from this work may be used under the terms of the Creative Commons Attribution 4.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

1741-4326/24/036022+22$33.00 Printed in the UK

© 2024 Crown copyright, UKAEA Culham

Keywords: integrated model, tokamak, surrogate model, active learning

(Some figures may appear in colour only in the online journal)

1. Introduction

Turbulent transport is the dominant transport mechanism in tokamak plasmas. Understanding and predicting it is essential to achieving fusion power [1, 2]. Transport models constitute a fundamental tool towards the delivery of ITER, DEMO-class reactors and beyond. However, the computational cost associated to integrating transport models in highly iterative applications such as multichannel integrated models (e.g. [3–7]) requires the delivery of fast and accurate surrogates, particularly for many-query applications such as simulation uncertainty quantification, scenario optimization and controller design. Feed-forward neural network (NN) surrogate models of the quasi-linear gyrokinetic model QuaLiKiz [8–10] and the gyrofluid model TGLF [11–13], have shown a factor 10^4 prediction speedup thus enabling real-time capable profile prediction [14], discharge optimisation studies [15] and integrated core-pedestal transport models [13, 16] at a fraction of the computational cost.

Due to the cost associated to retrieving large simulation databases to be adopted as training sets for these NNs, previous works have focused on spanning a small volume in the input space. This restricts some of the current applications to small dimensionality and narrow range in parameter space [17, 18], or medium dimensionality [10], also sometimes based on experiments [13, 19]. At the same time, in [20], where linear GKW [21] simulations were used to derive semi-empirical saturation rules based on JT60 discharges, an increase in data availability was indicated as a major contributor to the success of the derived reduced-order model in integrated models. In recent work on developing bespoke gyrokinetic surrogates for ITER [22], increased data efficiency was also identified as a top priority for the development of surrogate models of higher fidelity gyrokinetic codes.

Big data is not always necessarily informative. Indeed, current datasets obtained from experimental parameter spaces are severely oversampled. For example, [23] devised a clustering algorithm for the dataset presented in [19], demonstrating that a performing surrogate can be trained on a carefully selected subsample of the full dataset, with up to a factor of 10 reduction in training set size. Thus, the amount of information needed to obtain an actionable surrogate is contained in a significantly smaller subset of current gyrokinetic databases. While extremely useful to uncover the oversampling problem typical of current approaches, the work by [23] was performed *a-posteriori*, once the costly training set had already been generated.

Moreover, by the nature of the critical threshold characteristic of tokamak turbulence, not all plasma states result in unstable modes. In previous work [10, 19], the consistency of the surrogate with the critical threshold behaviour was enforced by means of a physics-based loss function that encouraged a controlled extrapolation to negative values where the true output fluxes were null. Negative predictions would then be clipped to zero at inference time. Although effective, this strategy lacked in efficiency as it resulted in a large fraction of the computational budget being spent to obtain stable modes (roughly 40% in [19] across all the electrostatic modes resolved). Instead, we hypothesise that the boundary manifold between stable and unstable inputs may be learned more efficiently using a separate surrogate model. This idea first appeared in our previous work [24], and it was developed concurrently by Hornsby *et al* [25] for data-efficient surrogates of micro-tearing modes.

This study proposes to build NN surrogate models of gyrokinetic turbulence by leveraging Active Learning (AL, [26]) methods. Active Learning is a sequential sampling strategy that queries an expensive black box function (in our case a gyrokinetic model) by means of an acquisition function that identifies regions where additional data would improve the NN performance. Contrary to Bayesian Optimisation approaches, which aim to perform sequential optimisation with only a few function evaluations (see for example [17, 27–29] for applications relevant to Fusion), Active Learning enables learning of the function to be approximated over the entire parameter space.

Here we develop ADEPT (Active Deep Ensembles for Plasma Turbulence), a two-stage AL framework where a surrogate of the critical gradient threshold in the form of a classifier determines whether a given input will result in growing modes, and a regressor predicts the output turbulent transport fluxes. We focus on an acquisition function that queries inputs for which the output uncertainty of the NN is highest, thus maximising informativeness [30]. Deep Ensembles [31], which provide state-of-the-art uncertainty quantification capabilities for NNs, are adopted as the surrogate model.

We provide a demonstration of the ADEPT pipeline using an existing large database of QuaLiKiz simulations obtained from JET inputs [19]. For this proof-of-concept work we focus on ITG turbulence only. As the input–output mappings are already available in the dataset, we can easily test the performance of the two-stage workflow. Explicit integration of gyrokinetic models in ADEPT will follow in upcoming work.

The paper outline is as follows. We describe the dataset in section 2, we introduce the ADEPT methodology in section 3.1 and we outline the integrated modelling framework in section 4. In section 5 we give the first main result of the paper. We demonstrate that even only the inclusion of the classifier stage and the adoption of more powerful deep learning models such as Deep Ensembles results in actionable performance for turbulent transport surrogates with around 200 000 simulations for 15 input dimensions, that is a two order of magnitude reduction from the original dataset. The physics-informed nature of the proposed sampling strategy only queries a minority of the inputs in the stable regions, which are instead dominant in the original dataset, thus enabling the surrogate to focus on accurate modelling of non-zero transport fluxes. Sequentially building the training dataset via AL results in a further large reduction in training sample size. In section 6 we validate ADEPT on a representative parameter scan and on integrated modelling of ITG-dominated JET scenarios. We find that ADEPT and previous work [19] agree with JINTRAC runs that adopt the original QuaLiKiz model to better than 10%, albeit ADEPT was trained with two orders of magnitude less data compared to the surrogates in [19]. Finally, in section 7 we discuss the results obtained, identify remaining issues and propose potential solutions to be explored in future work.

2. Data

We use the existing JET-Exp-15D dataset devised in [19, 32]. The dataset contains the input–output mappings of the QuaLiKiz [8, 9] quasilinear model. The inputs are based on 2135 JET experimental discharges including a variety of plasma scenarios, augmented taking into account measurement uncertainties for the parameters that turbulence is most sensitive to. The dataset generation took approximately 350 kCPUh.

The input space is 15-dimensional and it is detailed in full in [19]. A full list of inputs and outputs, and how they are computed, is available in their tables 2 and 3; this information is summarised here briefly for convenience. The inputs include: the species charge number, the species mass number, the fractional species density, the logarithmic electron density gradient, the ion and electron temperature gradients, the rotation Mach number, the rotation gradient, the radial coordinate, the tokamak aspect ratio, the safety factor, the magnetic shear, the pressure gradient (via α_{MHD}) the collisionality and the ExB shearing rate. The output encompasses the multiple channels of transport of ITG, ETG and TEM turbulence obtained from QuaLiKiz. The raw dataset produced from all the available inputs was subjected to consistency checks to either enforce physical consistency within the data (i.e. ambipolar particle fluxes, consistency between the predicted fluxes and the fluxes calculated from combining diffusive and convective terms computed separately) or discard abnormally large heat fluxes and abnormally small particle fluxes. This filtering process resulted in approximately 30% of the data being discarded, see table 6 of [19] for more details.

In the remainder of this work the focus will be on ITG turbulence, for which only less than 25% of inputs in the JET-Exp-15D dataset develop turbulent transport. The transport fluxes considered (in GyroBohm units, see table 3 of [19]) are the heat flux of ions ($q_{\mathrm{i,ITG}}$) the heat flux of electrons ($q_{\mathrm{e,ITG}}$) the momentum flux of ions ($\Pi_{\mathrm{i,ITG}}$) the particle flux of electrons ($\Gamma_{\mathrm{e,ITG}}$) and the particle flux of ions ($\Gamma_{\mathrm{i,ITG}}$).

3. Data-efficient surrogate models

3.1. Active learning

3.1.1. Basics. Active Learning (AL, e.g. [26] for a review) is a sampling strategy that aims at reducing the amount of training data needed to obtain a performing surrogate. An AL system comprises three components: a learner, an oracle and a query strategy. The learner is a ML method, such as a NN or Gaussian Process [33], that improves its performance as more data is collected from the oracle according to the query strategy. The decision on which learner to use depends on the nature of the problem: Gaussian Processes are more suitable in the low-data regime, while NNs are more effective in the big-data limit. The oracle is a costly data acquisition system that provides the training data for the learner; the oracle might be, for example, a simulator (which is the case this paper is focused on) or a human annotator (such as in the GalaxyZoo project, [34]). The main focus of the AL literature is on defining efficient query strategies [26].

AL can be applied in a pool setting and a streaming setting. In the first case, the query strategy acts on a pre-existing pool of unlabelled data (that is, for which only inputs are available but outputs are unavailable) and the distribution of the input space is fixed, while in the second setting a decision on which data to focus the labelling effort is made on a source of streaming data, potentially from a non-stationary distribution. Although digital twinning applications involving building surrogate models off streaming data from fusion devices may benefit from AL, the aim of this paper is to prove the simpler pool setting.

3.1.2. Maximum informativeness and uncertainty sampling. The goal of AL is to obtain a machine learning predictive model by identifying training points that are more efficient than random selection. Space-filling methods, such as Latin Hypercube Sampling (LHS, [35]), have been shown to improve upon random selection, however space-filling algorithms sample the input space just once, and therefore do not account for potential redundancy in the information provided by different inputs. A more efficient query strategy consists in maximising the informativeness of the training sample as a whole. The simultaneous placement of N points to obtain optimal coverage of the parameter space of interest is, unfortunately, computationally intractable [36]. Popular alternatives, which include sequential acquisition strategies that account for changes in the model induced by the newly collected training data, are still more advantageous than the fixed design space offered by space-filling algorithms.

The sequential strategy proposed in [30] queries the inputs for which the surrogate model's predictive uncertainty is largest,

$$x_{\text{query}} = \underset{x \in \mathcal{U}}{\arg\max}\, \sigma^2\left(x; D_{\text{train},t}\right), \tag{1}$$

$$D_{\text{train},t+1} = x_{\text{query}} \cup D_{\text{train},t} \tag{2}$$

where $\sigma(x; D_{\text{train},t})$ is the output uncertainty of a learner trained on a dataset $D_{\text{train},t}$, t is the current iteration and $\mathcal{U}$ is a pool of inputs (e.g. the plasma states) for which the outputs (e.g. the turbulent fluxes) are not available. As indicated in the expression above, the dataset at the next iteration is enriched with data obtained from the query. The uncertainty is the standard deviation of a regression model.

Here, we adopt Batch Mode AL (e.g. [37]), which consists in performing the acquisition for the M inputs that rank highest in the model uncertainty. Batch Mode AL is more suitable for NNs, as retraining a NN with just one new sample is impractical.

The literature on AL strategies is vast (see [26, 38] for two excellent reviews). In the following, we will adopt the acquisition function in equation (1) for the following reasons. First, it is good practice to develop surrogate models that offer uncertainty estimates on their predictions, especially in view of incorporating surrogates of gyrokinetic models, the topic of this paper, into integrated suites to enable uncertainty quantification studies. Moreover, the implementation of uncertainty sampling by exploiting surrogate models with such capabilities is trivial and, as we will show, it performs well in practice. Furthermore, while conceptually very simple, uncertainty-driven AL is widely used with great success in other fields, such as, for example, drug discovery [39].

As a final note, it is worth pointing out that AL tends to induce a shift between the distribution of the unlabelled pool $\mathcal{U}$ and the that of the training set over time, as only the most informative points are selected for labelling [30, 40]. It is therefore crucial to ensure that the NN uncertainties are well-calibrated also out of distribution. A discussion on this matter is carried out in section 3.3.

3.2. *Physics-informed active learning for gyrokinetic models with ADEPT*

Linear gyrokinetic turbulence exhibits a critical gradient behaviour, whereby growing modes and the resulting turbulent transport are triggered only above a certain threshold in the driving gradients that depends on the plasma conditions[6]. This creates a further complication for surrogate models, as the fluxes predicted need to be exactly zero in the stable region to avoid the presence of spurious transport that would alter the predictions of integrated models. In previous work [10] showed that the sharp transition between the stable and unstable region is smoothed out in naive approaches where a single regressor surrogate model is trained on the entire space. The solution proposed in [10] was to identify the critical gradient threshold by encouraging a NN to predict negative values whenever the true flux was null. These were then clipped to zero for use in the integrated model. For positive fluxes, instead, the NN would be trained using a standard Mean Squared Error loss function. van de Plassche *et al* [10] showed that a NN surrogate that does not account for the critical gradient behaviour of gyrokinetic turbulence leads to oversmoothing around the critical gradient and therefore overpredicts transport. The physics-informed training adopted in [10, 19] elegantly enables the NN to perform both classification tasks (i.e. whether an input results in growing modes) and regression tasks (to predict the turbulent fluxes). While effective, [10]'s method results in the computational budget spent to obtain the training set to be overly focused on points well within the stable region of the input space.

Below we propose ADEPT (Active Deep Ensembles for Plasma Turbulence)[7], a two-stage, physics-informed active learning strategy that delivers a significant reduction in the volume of data required to train performing surrogates. Contrary to previous work, we assign the classification and regression tasks to two separate NNs. This setup preserves the physics-informed nature of the framework proposed by [10], but it splits the burden of identifying the critical gradients and regressing to turbulent fluxes between two highly specialised NNs. Given a data pool $\mathcal{U}$ of inputs, a NN classifier and a NN regressor are pretrained on a small (20 000 points) random sample of data for which the input–output mapping is available. This pretraining allows to capture a general initial representation of the space. Hereafter, for each iteration, the networks and the labelled dataset are updated following the strategy shown in figure 1:

- The classifier is tasked with screening a sample of candidate points in the data pool $\mathcal{U}$. This is the physics-informed stage of the workflow. The entire pool may be screened, but this may slow down the acquisition process in the case of large data pools, such as that of the JET-Exp-15D dataset. Therefore the classifier is used to evaluate 300 000 inputs randomly sampled from the pool;
- The acquisition function in equation (1) uses the regressor's uncertainty (in our case, the epistemic uncertainty in equation (8)) to select for acquisition a batch of inputs of size `acquisition_batch` candidates. Extending the acquisition strategy to account for the uncertainty of the classifier will be the subject of future work;
- The outputs for the input candidates selected are queried from the model of choice (QuaLiKiz in the case of this paper);
- The newly available input–output mappings are appended to the training data;
- Both the regressor and the classifier NNs are trained again.

The above loop iterates until either the computational budget is consumed or the surrogates achieve the target performance.

As gyrokinetic turbulence involves multichannel transport, it is necessary to maximise the information gain for all the

[6] The reader is reminded that in nonlinear models, instead, the identity between the linear stability threshold and the onset of transport is broken by the Dimits shift [41].

[7] We adopt the `PyTorch` library.

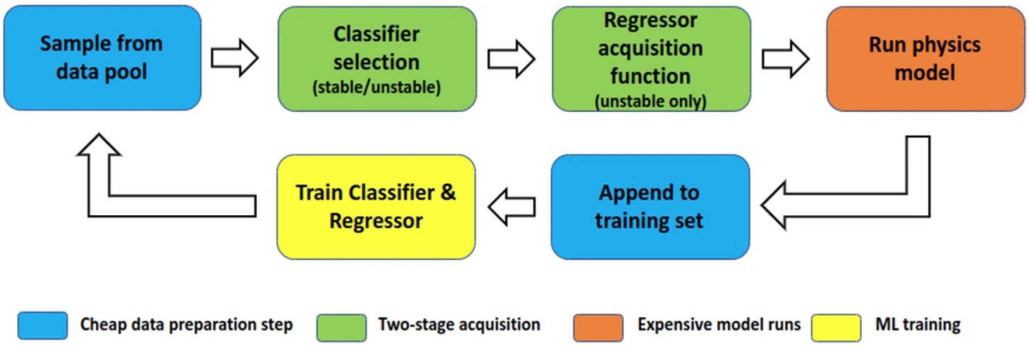

Sample from data pool
Classifier selection (stable/unstable)
Regressor acquisition function (unstable only)
Run physics model
Append to training set
Train Classifier & Regressor
Cheap data preparation step
Two-stage acquisition
Expensive model runs
ML training

Figure 1. Schematic diagram of the two-stage physics-informed AL workflow used in this work. Given a data pool for which only inputs are available, a classifier evaluates the likelihood of a given input in the pool resulting in unstable modes. The acquisition function is evaluated on the unstable inputs, and a batch of the most uncertain ones are selected to be run through the gyrokinetic model. The newly obtained input–output mappings are used to train both NNs. This strategy is repeated until the computational budget has been exhausted or the performance of the surrogates is deemed actionable.

fluxes, and therefore the acquisition function in equation (1) becomes

$$x_{\text{query}} = \underset{x\in\mathcal{U}}{\arg\max} \sum_{k=1}^{N_{\text{fluxes}}} \sigma_k^2\left(x; D_{\text{train},t}\right). \tag{3}$$

3.3. Surrogate uncertainty via deep ensembles and its uses within integrated modelling

It has long been established that NN models give overconfident predictions that are factually wrong (e.g. [42]). Equipping NNs with a notion of uncertainty in their own predictions has since become a mainstream line of research producing a rich literature [43]. The calibration of NN uncertainties are currently debated in the community [44], with new frequentist methods on the rise ([45] and references therein). In particular, although NN uncertainties generally increase moving away from the training distribution across all uncertainty estimation methods, the issue of their calibration remains a point of concern.

Lakshminarayanan *et al* [31] proposed to train NNs using *proper scoring rules* [46] to obtain calibrated uncertainties. Given a probabilistic NN approximation $p_\theta(y|x)$ [8] of the true distribution $q(y|x)$, a scoring rule $\mathcal{S}\left(p_\theta,(x,y)\right)$ assigns to a learned supervised model a score based on the quality of the probabilistic prediction p_θ and, consequently, the quality of the predictive uncertainty. A scoring rule can be formalised as global metric by integrating over the full probability space,

$$S\left(p_\theta, q\right) = \int q\left(y|x\right) \mathcal{S}\left(p_\theta,(x,y)\right) dxdy. \tag{4}$$

A scoring rule is *strictly proper* if $S(p_\theta, q) \leqslant S(q,q)$, that is, the learned approximation p_θ is best only in the case where it perfectly reproduces q, that is when $p_\theta = q$. In other words, the approximation p_θ can never provide better probabilistic predictions than q.

A NN trained with a proper scoring rule is encouraged to provide better calibrated uncertainties compared to one trained on MSE. The standard MSE loss routinely used to train NNs (see equation (9)) is not a strictly proper scoring rule [47], and therefore cannot provide calibrated uncertainties out of the box, as it only considers the mean prediction of a probabilistic model. Moreover, all probabilistic models p_θ featuring the same mean will be scored in the same way, although the disagreement between the higher moments of p_θ and q can be arbitrarily high.

Instead, [31] showed that the log-likelihood of the data under a learned NN, $\log p_\theta(y|x)$, is always a strictly proper scoring rule and it provides calibrated uncertainties also in practice (see their figure 7). Ensembles of NNs trained with a proper scoring rule are termed *Deep Ensembles*. If the ensemble is treated as a uniformly weighted mixture of M models, then the proper scoring rule for the ensemble is

$$NLL = -\frac{1}{M}\sum_{k=1}^{M} \log p_{\theta_k}\left(y|x\right) \tag{5}$$

where we have taken the negative of the log-likelihood as an objective to minimise.

For classification, the usual binary cross-entropy loss is also a proper scoring rule, and therefore deep ensembles and regular NN ensembles coincide. For regression problems, the Gaussian negative log-likelihood below is a proper scoring rule,

$$-\log p_\theta\left(y|x\right) = \frac{\log \sigma_\theta^2\left(x\right)}{2} + \frac{\left(y-\mu_\theta\left(x\right)\right)^2}{2\sigma_\theta^2\left(x\right)} + const. \tag{6}$$

A NN with two output neurons trained with the objective above will explicitly learn the mean $\mu_\theta(x)$ and variance $\sigma_\theta^2(x)$, where the suffix indicates that these quantities are parametrised by the same NN with parameters θ. With this expression,

[8] Where θ are the parameters of the NN.

the NN is encouraged to learn that, in order to have a low variance to minimise the first term of equation (6), the predictions μ_θ need to be very accurate to keep the second term small.

The mean μ_E and variance σ_E of the deep ensemble as a whole can be computed under the assumption of a uniformly weighted mixture of M members:

$$\mu_E = \frac{1}{M}\sum_{k=1}^{M}\mu_{\theta_k} \tag{7}$$

$$\sigma_E^2 = \underbrace{\frac{1}{M}\sum_{k=1}^{M}\sigma_{\theta_k}^2}_{\text{aleatoric}} + \underbrace{\left(\frac{1}{M}\sum_{k=1}^{M}\mu_{\theta_k}^2\right) - \mu_E^2}_{\text{epistemic}}. \tag{8}$$

On the other hand, [19] used an NN ensembling slightly different approach compared to Deep Ensemble to obtain a notion of uncertainty. The approach consisted in training a committee of ten NNs with identical architecture but different random initialisation, and the mean and variance of the predictions were then used for downstream applications. The NNs in [19] were trained to minimise the Mean Squared Error (MSE) between each NN prediction, $\hat{y}$ and the target, y_{true},

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}_i - y_{\text{true},i}\right)^2. \tag{9}$$

Note that the expression in equation (6) allows for heteroskedasticity in the variance estimate (i.e. the variance can vary based on each individual input, and this is explicitly modelled). It is important to realise that, without this feature, the expressions in equations (6) and (9) would coincide (up to a constant) after identifying $\mu_\theta \equiv \hat{y}$. Although this may seem only a subtle difference between Deep Ensembles and regular NN committees, the objective in equation (9) does not explicitly capture NN uncertainty. Therefore, the uncertainties obtained by considering the standard deviation of the ensemble outputs are not guaranteed to be valid. Instead, training Deep Ensembles involves the optimisation of the negative log likelihood, which improves MSE with the constraint of fitting sensible uncertainty estimates. Hence, Deep Ensembles strike a balance between uncertainty quantification capabilities and predictive power, which are both equally important in downstream applications.

The variance of a Deep Ensemble regressor (equation (8)) is composed of two contributions. The first one is the average variance between all members. The second one is the variance of the means of the ensemble, as shown in the last two terms on the right hand side of equation (8). The uncertainty of the deep ensembles (equation (8)) is sometimes interpreted as the sum of the epistemic uncertainty (i.e. the uncertainty of the model) and the aleatoric uncertainty (i.e. the irreducible noise in the data), e.g. [48]. The epistemic uncertainty is the natural choice to use in the acquisition function, as we seek to improve the inherent accuracy of the model regardless of data noise [49]. Conversely, the total uncertainty should be used to assess how much trust should be placed in the surrogate predictions for downstream applications such as integrated models.

Uncertainty quantification capabilities are also a natural feature of the classifier NN. The confidence of the classifier can be defined as its output probability of a point being unstable. Probabilities close to a value of 0.5 inform downstream applications that performing a run of the original QuaLiKiz model is recommended. Entropy [50], which measures the disagreement between the members of the ensemble, may also be used as an information-theoretical measure of uncertainty:

$$\mathcal{H}(x) = -\sum_{i=1}^{M} p_i(x)\log p_i(x), \tag{10}$$

where $p_i(x)$ is the output probability of the ith ensemble member. Both probability and entropy will be shown as measures of uncertainty for the classifier for a few parameter scans in section 6.1.

3.4. Details of the training procedure

We borrow from [10] the idea of fitting NN surrogate models to the 'leading flux' of a given turbulence type and the flux ratio between the leading flux and the secondary fluxes. This methodology was devised to ensure the same critical gradient behaviour for all fluxes of a given turbulent mode. While this is not strictly necessary in our case, as the classifier takes care of identifying the critical gradients, we opted for this option to minimise changes in the JINTRAC integration.

We train a suite of deep ensembles, each regressing to one turbulent flux, and one deep ensemble classifier for the stability boundary. We adopt 5 ensembles per model, each consisting of 8 layers with 512 parameters each and `ReLU` activation functions. At each Active Learning cycle, each model is trained for 200 epochs with 100 epochs of patience, a weight decay of $\lambda = 10^{-4}$ and a `batch_size` of 512. Each NN in every ensemble initialised with a different random seed, that results in different trajectories in weights space during training that converge to different local minima, and that will therefore result in different predictions. All the members of each ensemble are trained in parallel on a single A100 GPU using the `joblib` library[9].

For the regressor we adopt the NLL loss and for the classifier the binary crossentropy loss. Each `acquisition_batch` consists of 1024 training samples, doubling every 30 acquisitions due to the costs associated with retraining NNs on large datasets.

Extensive hyperparameter tuning was not performed in this study. Optimizing hyperparameters should ideally occur during each iteration of AL. However, even in AL research this is rarely done due to its high computational cost. Performing hyperparameter tuning at every iteration can be prohibitively time-consuming, especially as the training dataset grows. However, the extra computational cost incurred is expected to be small compared to the data acquisition for expensive high fidelity codes (e.g. [21, 51]).

[9] The implementation of https://github.com/TorchEnsemble-Community/Ensemble-Pytorch is followed.

Table 1. Summary table of most pertinent JINTRAC settings of the base case simulation.

	JET#73342	JET#92436
Description	H-mode	H-mode
Simulation type	Stationary	Stationary
# of grid points	51	101
Plasma time	20.75–22.75 s	10.0–12.0 s
Sim. boundary (ρ_{tor})	0.85	0.85
Equilibrium	Fixed	ESCO
Neoclassical transport model	NCLASS	NCLASS
Neutral particle model	None	None
NBI source model	Fixed	Fixed
ICRH source model	Fixed	Fixed
Impurity species	C	Be, Ni, W
Impurity transport model	None	SANCO
QuaLiKiz region	0.15–0.85	0.15–0.85
QuaLiKiz rot. option	2	2
Part. trans. option[a]	4	4
Momentum profile	Fixed	Predicted

[a] The particle transport options are only applicable when using the QLKNN model. Further details about the different options available within QLKNN are given in [19].

4. Integrated modelling

The JINTRAC integrated modelling suite [4] was chosen for this study both due its history of integration with QLKNN [10, 19] and its relevance for ITER scenarios [52]. The JINTRAC integrated model test cases and settings used in this study were taken from those used to validate QLKNN-jetexp [19], specifically selecting:

- the H-mode carbon wall discharge (JET#73342) [9];
- and the H-mode beryllium wall discharge (JET#92436) [32].

A summary of the major JINTRAC settings used for these test cases are provided in table 1. In the following sections we will compare the results of adopting either ADEPT, QLKNN-jetexp and the original QuaLiKiz within JINTRAC. Specifically, the critical gradient threshold in ADEPT will be estimated using the trained classifier, while it is estimated according to the methodology summarised in section 3.2 for the QLKNN-jetexp surrogates.

As outlined in section 4.2 of [19], the ion transport coefficients for the JET-Exp-15D dataset were derived only for a pure deuterium plasma, and therefore some assumptions need to be made to model impurity transport. While the differences in heat transport among different ion species can usually be neglected, this is not typically true for particle fluxes [53].

A first condition to allow treatment of the particle transport coefficients of impurities stems from the ambipolarity constraint,

$$\mathbf{\Gamma}_e = \sum_i \mathbf{\Gamma}_i Z_i. \tag{11}$$

However, a second condition needs to be specified for equation (11) to admit a unique solution. In this work, we follow [19] and assume a proportionality between the electron and the impurities particle fluxes:

$$\mathbf{\Gamma}_i = \mathbf{\Gamma}_e \frac{n_i}{n_e}. \tag{12}$$

ADEPT generates surrogate models that inherently include a measure of uncertainty. This characteristic can be leveraged in integrated modeling to evaluate the level of confidence that should be placed in the surrogate model's predictions, and perform a QuaLiKiz run whenever the uncertainty of surrogate is not considered acceptable. An in-depth study of the impact of the precise acceptance threshold on the integrated modelling results is outside the scope of this work, but it is highly recommended for future investigation. In particular, in this study the average predictions of the surrogates are used regardless of the surrogate uncertainty.

5. Results: data-efficient training sets with active learning

In this section, the surrogate performance resulting from the ADEPT strategy is presented. It should be noted that no QuaLiKiz simulations have been performed in this work, as the QuaLiKiz outputs are already available in the JET-Exp-15D dataset.

5.1. Benchmarking data efficiency

An important benchmark of performance of any Active Learning strategy, including ADEPT, is given by random sampling, which does not require a potentially expensive iterative strategy nor setup costs such as building a bespoke code base. In this section, Active Learning is shown to be more efficient than random sampling as it produces surrogate models that performs better while using fewer training data. Further considerations on the computational costs of ADEPT versus random sampling can be found in section 5.2.

Figure 2 shows the performance of ADEPT on ITG turbulence compared to random selection as a function of number of training samples collected. The metrics used to assess the surrogate performance are described in detail in appendix B. For both ADEPT and random selection the same NN architectures and training hyperparameters were adopted. It can be seen that ADEPT provides up a factor of 20 data reduction compared to random sampling. As shown in section 5.3, an important contribution to this success is the inclusion of the classifier stage, which allows for a more data-efficient learning of the manifold where unstable turbulent fluxes develop. In particular, the x axis of all subfigures in figure 2 refers to the total amount of labels collected, including both stable and unstable points, for both ADEPT and random sampling. However, for random sampling, that would imply that, of all the points in the training set, only a fraction is actually used for training the regressors. For ITG turbulence in the JET-Exp-15D data this is around 25%. ADEPT instead makes use of all the points in the training set. The contribution of the classifier in making ADEPT

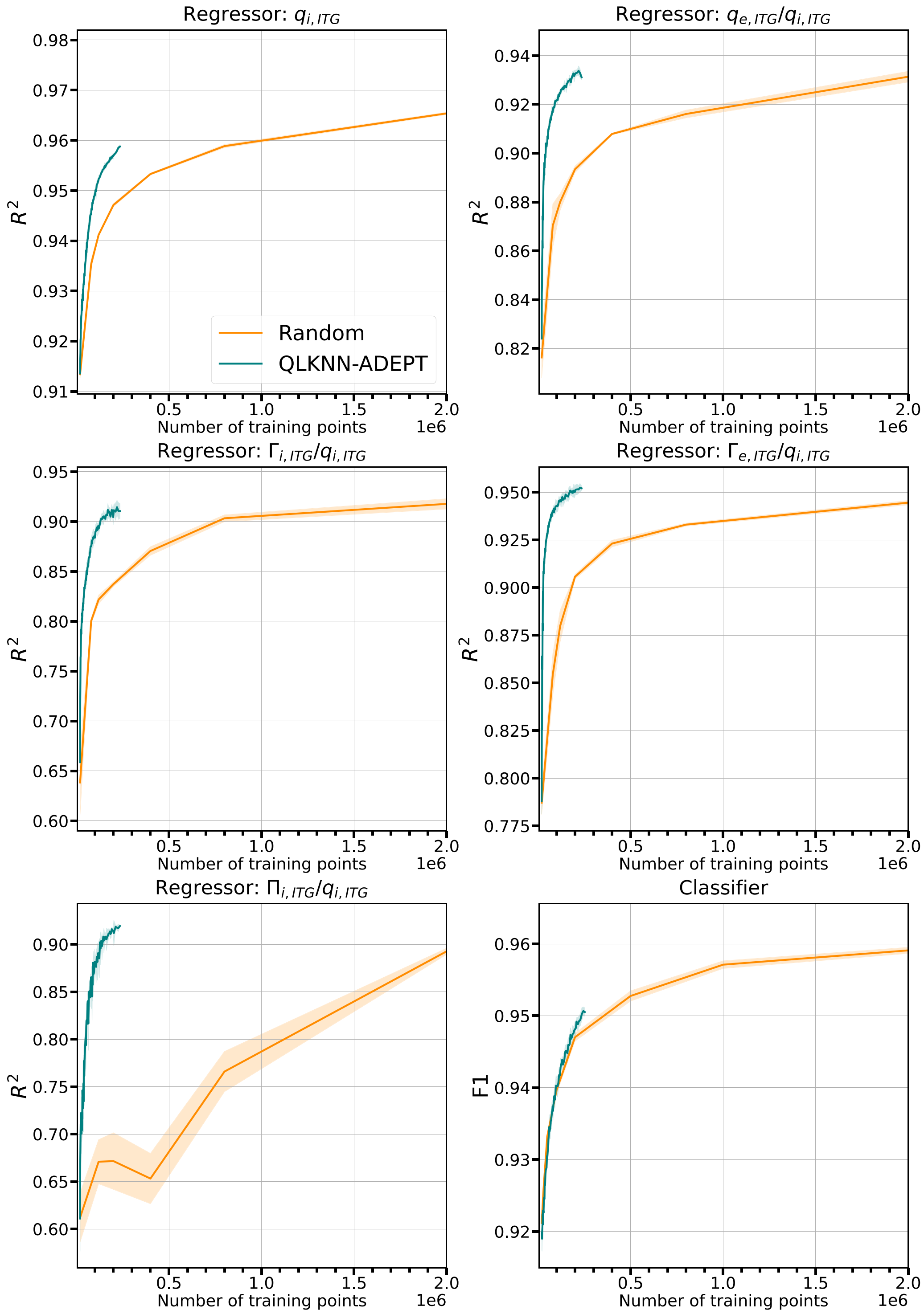


Figure 2. ADEPT vs random sampling performance. We use the R^2 for the deep ensembles regressing to the electron heat flux and the flux ratios involving the ion heat flux and the ion momentum flux. The $F1$ score for the stability boundary classifier is also shown. The shaded areas represent the standard deviation of 5 runs with different random seed. Active Learning improves data efficiency by at least factor of 2 (and up to a factor of 20) compared random sampling. ADEPT acquisitions were run until exhaustion of the computational budget (36 h).

more data efficient can be quantified by comparing ADEPT to random sampling only in the unstable region. This experiment is carried out in appendix D1, where it is shown that ADEPT is more data efficient than random sampling even in this limiting case for most fluxes apart from, notably, the leading flux $q_{\mathrm{i,ITG}}$. Overall, the classifier stage is estimated as resulting in a factor ∼3–5 reduction, while the uncertainty acquisition alone in a factor ∼0–5.

A performance comparison on a test set between ADEPT and the surrogates presented in [19], which were trained using approximately 20 000 000 data points, is given in tables 2 and 3. Although the surrogates in [19] do not explicitly employ

Table 2. The performance of ADEPT trained with up to 200 000 samples compared to the NNs presented in [19] (where 20 000 000 samples were used), in terms of R^2 score for the fluxes (see appendix B for a description of the performance metrics).

	$q_{\mathrm{i,ITG}}$	$q_{\mathrm{e,ITG}}/q_{\mathrm{i,ITG}}$	$\Gamma_{\mathrm{i,ITG}}/q_{\mathrm{i,ITG}}$	$\Gamma_{\mathrm{e,ITG}}/q_{\mathrm{i,ITG}}$	$\Pi_{\mathrm{i,ITG}}/q_{\mathrm{i,ITG}}$
ADEPT	0.9585	0.9366	0.9174	0.9554	0.9108
[19]	0.9518	0.9505	0.6987	0.5268	0.9140

Table 3. The performance of ADEPT trained with up to 200 000 samples compared to the NNs presented in [19] (where 20 000 000 samples were used), for the classifier (see appendix B for a description of the performance metrics).

	F1	Recall	Precision
ADEPT	0.9504	0.9412	0.9602
[19]	0.7791	0.9880	0.6431

a separate classifier NN to model the critical gradient, they achieve a comparable effect by zeroing out all fluxes when the average predicted leading flux from their NN ensemble is predicted to be negative. Therefore, we can evaluate the F1 performance of these surrogates, as they effectively exhibit classifier-like behaviour in this context.

It can be seen that the performance of ADEPT in terms of the F1 score is comparable or superior to that of the NNs in [19], albeit with two orders of magnitude fewer data. In particular, a high classifier performance in the case of ADEPT is crucial in ensuring that sampling does not occur deep in the stable regions. As a demonstration, we have computed that the contribution of stable inputs to the training set of the classifier is 75% in the random sampling case (and, indeed, the case of [19]), while this drops to around 20% in the case of ADEPT. The surrogates of [19], instead, feature a poor Precision, showing a high number of non-zero flux predictions in the stable region. The latter surrogates, however, achieve a better Recall, albeit with two orders of magnitude more training data points. ADEPT would need more data to reach the same kind of performance. As further discussed in section 7, the classifier is not currently included in the acquisition function explicitly, which instead will be crucial to improve its data efficiency compared to random sampling. This feature will be explored in future work.

The classifier metrics in ADEPT depend on the threshold of the output probability chosen to determine whether the input is unstable. The results in table 3 have been derived using a probability threshold $p_{thresh} = 0.5$. A comparison with QLKNN-jetexp where p_{thresh} is varied is shown in appendix C where a qualitatively similar behaviour to table 3 holds irrespectively of the precise threshold chosen.

5.2. Computational efficiency considerations

As shown in the previous section, ADEPT is up to a factor of 10 (or more) more data-efficient than random sampling. However, the speedup obtained in terms of data efficiency should also be considered in the context of retraining a suite of Deep Ensembles multiple times, which can be expensive. Specifically, if the total cost of data acquisition is C_{a}, and the total cost of training is C_{train}, then the most economical data acquisition system will be the one that minimises $C_{\mathrm{tot}} = C_{\mathrm{train}} + C_{\mathrm{a}}$. Some considerations on this can be found in this section.

As 2M points is the largest number of training data considered in this work, the performance of six Deep Ensembles (five for the fluxes and one classifier) trained using random sampling on 2M data points will be taken as the benchmark. The generation of 2M points with QuaLiKiz takes around $C_{\mathrm{a}} \sim 20\,\mathrm{kCPUh}$ (inferring from table 1 of [19]). Training the six surrogate models once on 2M data points sampled at random takes $C_{\mathrm{train}} \sim 8\,\mathrm{h}$ on a single A100 GPU card. In total, then, the cost is $C_{\mathrm{tot}} \sim 20\,\mathrm{kCPUh} + 8\,\mathrm{GPUh}$.

On the contrary, the Active Learning training pipeline with ADEPT took $C_{\mathrm{train}} = 36\,\mathrm{h}$ on a single A100 GPU. Active Learning collected up to 200 000 samples in that timeframe, resulting in 10% of the compute cost to acquire the QuaLiKiz labels compared to the random sampling case above, at little or no loss in performance, for a total of $C_{\mathrm{a}} \sim 2\,\mathrm{kCPUh}$. In total, then, the cost is $C_{\mathrm{tot}} \sim 2\,\mathrm{kCPUh} + 36\,\mathrm{GPUh}$.

Thus, the total cost of training the surrogate multiple times is dwarfed by the cost of the data acquisition, which is two orders of magnitude higher. Admittedly, multiple QuaLiKiz runs can be parallelised, while ADEPT is an iterative strategy in nature. To mitigate this, in the future each ensemble could be trained on separate GPUs, while in this work all ensembles were trained in series on the same card. QuaLiKiz is a relatively lightweight model, and the efficiency gains are likely to be much higher for higher fidelity codes, for which the execution time is orders of magnitude higher than QuaLiKiz.

5.3. The effect of abandoning the physics-informed approach

The importance of ensuring that the critical gradient of turbulent transport is preserved by surrogates was discussed already in [10]. The behaviour of one 'naive' regressor surrogate model that predicted all output fluxes, including in the stable region, and without the clipping strategy for negative leading fluxes proposed in [10], was shown to oversmooth the critical gradient behaviour and produced unphysical results within integrated modelling.

In this section we further demonstrate the two following points: (i) providing an estimate for the critical gradient (i.e. utilising a physics-informed approach) results in increased data efficiency within ADEPT compared to naive surrogates and (ii) as a consequence of (i) the seemingly good integrated performance of the naive approach actually results in poor performance in the unstable region compared to ADEPT.

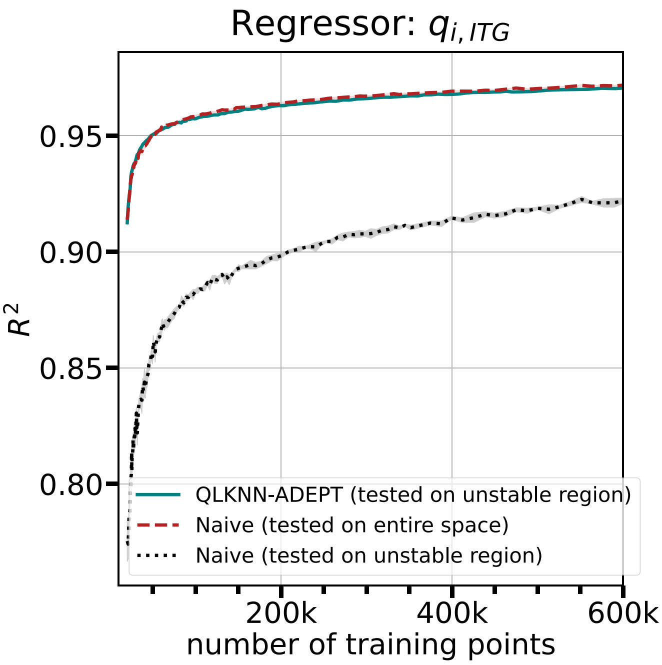


Figure 3. The performance of the regressor in the physics-informed two-stage ADEPT workflow (teal) versus a naive approach where both zero and non-zero fluxes are fit by the same regressor NN (red). It can be seen that, for a fixed amount of training data, when tested on the unstable region only (grey line), the naive approach achieves a much poorer performance than the physics-informed approach.

To this end, we performed an experiment where Active Learning was run in a naive fashion, where one regressor was trained on both stable and unstable regions, using only the regressor uncertainty to drive the acquisition. No classifier was used for this experiment. The test set that is natural to use for this method is drawn from the entire space (red line in figure 3) and, at face value, the performance of the naive methodology seems actionable. It is however instructive to inspect the performance solely on the unstable region. Figure 3 demonstrates that the data efficiency of the naive method degrades significantly when specifically tested on unstable inputs. A crucial observation that justifies the observed behaviour is that the JET-Exp-15D dataset used in this work contains a significant proportion of stable inputs, accounting for over 75% of the data available for ITG turbulence. Thus, the representation learned by the naive approach is not capable of accurately capturing the mapping for both stable and unstable regions.

On the contrary, the classifier stage of ADEPT helps prevent querying points inside the stable region and instead allows the regressor to focus on the unstable region, thus resulting in improved data efficiency. In line with [10], our findings show that integrated performance metrics must be handled with care when informing suitability for downstream applications.

5.4. Training dynamics dependence on number of fluxes

The acquisition function in equation (3) fully accounts for the multichannel nature of gyrokinetic turbulent transport. It is instructive to inspect the training dynamics induced by the training samples collected iteratively by the acquisition function when using a different number of fluxes.

In figure 4 we show the test performance of the two-stage ADEPT pipeline when only the leading flux, $q_{\mathrm{i,ITG}}$, is used. We compare the results to the case where all five fluxes are considered. While the performance on predicting the fluxes in the unstable region is greatly improved compared to the multichannel case, it can be seen that the classifier performance degrades significantly, performing even worse than random sampling. A benchmark of the role of the ADEPT classifier compared to a case where sampling is performed at random in the unstable region only is shown in appendix D, where it is demonstrated that training ADEPT on one flux only is more advantageous, while training ADEPT on five fluxes results in worse performance than random. A possible explanation for the behaviours observed is that in the multichannel case the contribution of the uncertainties from the different fluxes conspire to query a batch that carries high information for the classifier, but not for the regressor surrogate of $q_{\mathrm{i,ITG}}$.

Ultimately, the patterns evident in figure 4 are driven by the acquisition function, which relies solely on the uncertainty of the regressors. We believe that this behaviour can be controlled by developing an alternative acquisition function that explicitly takes into account classifier uncertainty.

6. Results: validation

Bearing in mind that a large-scale evaluation study including uncertainty quantification is outside the scope of the present paper, in this section we validate ADEPT on parameter scans (section 6.1) and JINTRAC modelling of selected JET discharges (section 6.2) as introduced in section 4. We use surrogates that were trained on a final dataset of 200 000 input–output pairs collected using the ADEPT strategy and compare their performance to the work of [19] (QLKNN-jetexp in the following), which were trained using approximately 20 000 000 input–output pairs.

6.1. Parameter scans

In this section, we validate the ADEPT surrogates on parameter scans obtained by running the original QuaLiKiz model. For each output flux, we fix 14 of the 15 input dimensions of the dataset to their median value and we perform a scan in the remaining dimension. Figure 5 shows the parameter scans for R/L_{T_i}. Other similar figures can be found in appendix E. There is good agreement between the true QuaLiKiz model compared to the NN predictions and their uncertainty on the turbulent fluxes. As expected, the NN uncertainty is larger further away from the bulk of the training distribution for all fluxes. The only exception is $q_{\mathrm{i,ITG}}$, where both surrogate models are clearly overconfident for $q_{\mathrm{i,ITG}} > 100$ GB, which is outside of the training data regime. The observed behaviour for this particular flux serves as a counterexample to the claims made in [54], who observed that, for a limited number of highly curated datasets typically used for benchmarking NN

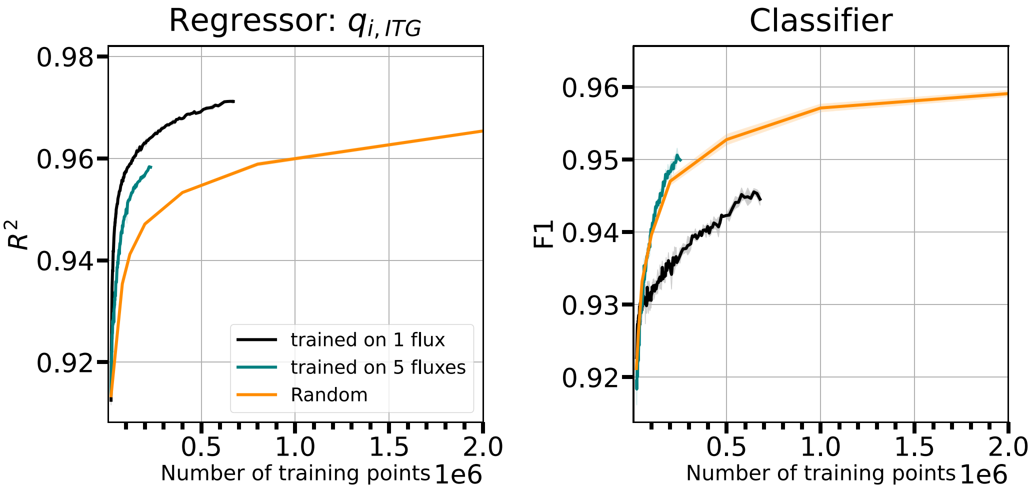


Figure 4. The performance of the ADEPT pipeline for $q_{\mathrm{i,ITG}}$ when the surrogates are trained either $q_{\mathrm{i,ITG}}$ only (black lines) or on all the five fluxes considered (teal lines, reproduced again from figure 2 for convenience). The behaviour of both the regressor and classifier depend strongly on the number of fluxes used. ADEPT acquisitions for both the black and teal lines were run until exhaustion of the computational budget (36 h).

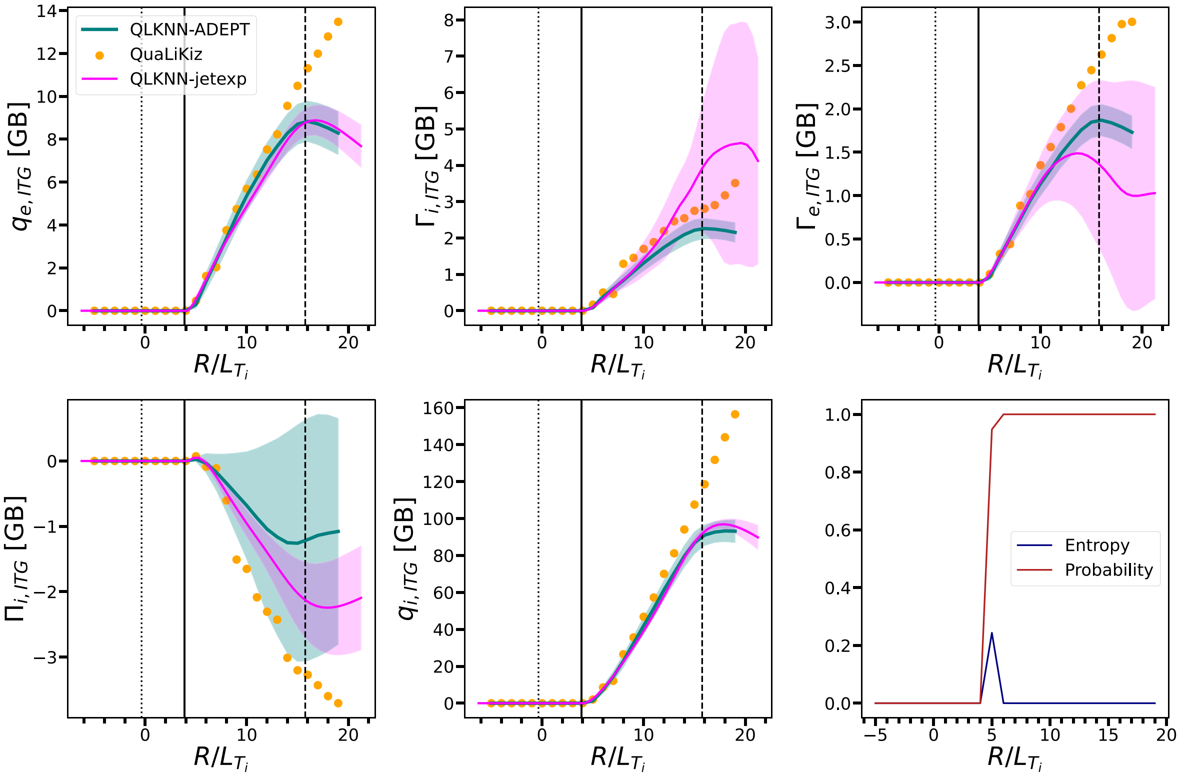
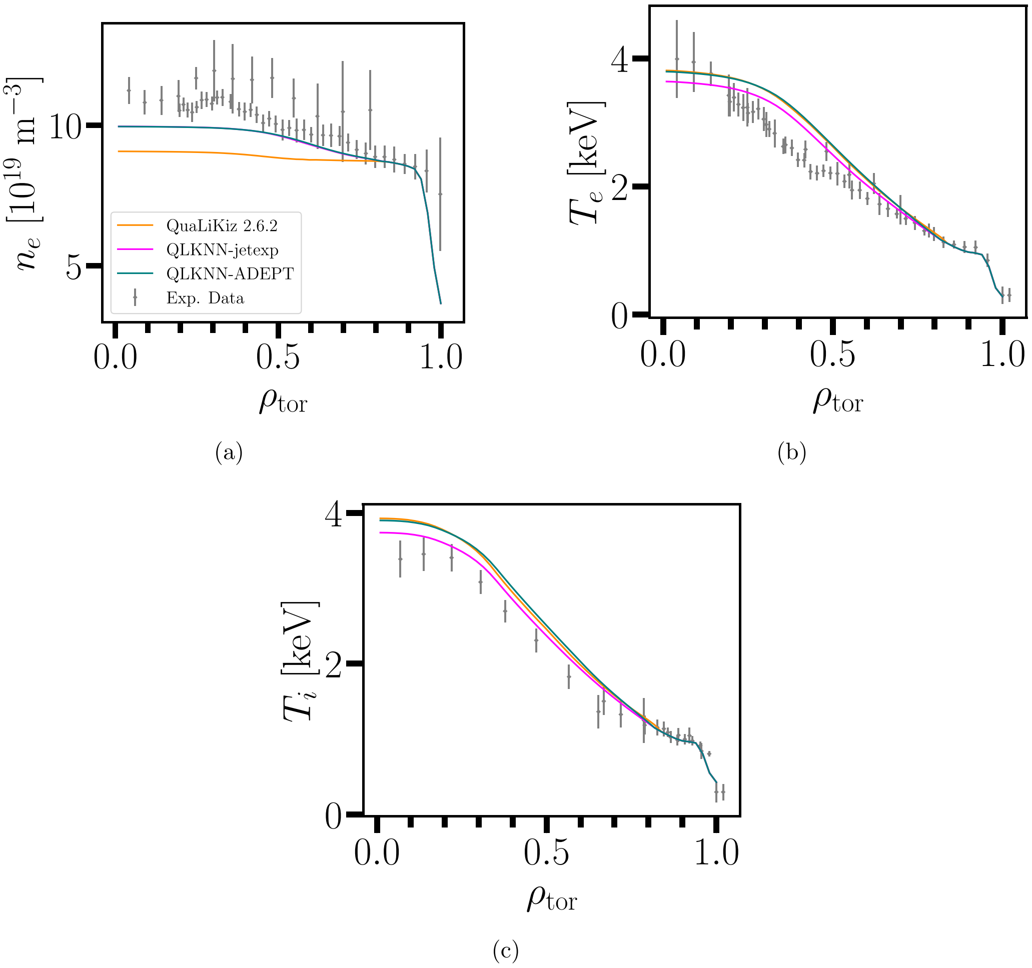


Figure 5. Parameter scans in R/L_{T_i} for the five ITG fluxes used in this work. The QuaLiKiz runs are shown in orange, while the predictions of the surrogates are shown in teal for ADEPT and magenta for QLKNN-jetexp. The shaded areas indicate the 1σ confidence levels. Dotted, solid and dashed lines indicate the 2.5%, 50% and 97.5% of the distribution in R/L_{T_i}. The bottom right panel shows the uncertainty for the Deep Ensemble classifier in terms of probability of an input being unstable and entropy of the ensemble. Note that the uncertainty estimates provided by the committees in [19] and by ADEPT differ significantly. See main text for discussion.

performance, Deep Ensemble uncertainties are more reliable than other methods for out-of-distribution samples.

Although qualitatively QLKNN-jetexp and ADEPT reproduce the QuaLiKiz trends, there are important differences in how the two approaches perform around the critical gradient. In particular, QLKNN-jetexp tends to provide a smoother behaviour while the two-stage nature of ADEPT results in sometimes too sharp discontinuities (see figure E1). However, in some instances (see, e.g. figures E2 and E4) QLKNN-jetexp oversmooths the trends around the critical gradient, albeit it does so out of the training distribution. It is also important to note that the classifier uncertainty for ADEPT peaks around



(a)

(b)

(c)

Figure 6. Comparison of the steady-state profiles from the simulation of JET#73342.

the critical gradient, which is highly desirable as it provides a way to refine the critical gradient estimation—this is a new feature that was not present in QLKNN-jetexp.

6.2. Validation on ITG-dominated JET discharges

Figures 6 and 7 show the steady state profiles obtained by adopting ADEPT, the original QLKNN-jetexp surrogates of [19] and the original QuaLiKiz model. The experimental data is shown here for reference, however the purpose of this test is to verify whether the surrogate models are able to reproduce the behaviour of QuaLiKiz within JINTRAC. Table 4 provides the profile-averaged relative RMS (RRMS) for these JINTRAC runs using their respective networks, with the QLKNN-jetexp reference given inside the square brackets. The RRMS is computed as:

$$\text{RRMS} = \sqrt{\frac{1}{N}\sum_i \frac{\left(Y_{\text{NN,i}} - Y_{\text{QLK,i}}\right)^2}{Y^2_{\text{QLK,i}}}} \tag{13}$$

where the sum is over the number of radial points, and $Y_{\text{NN,i}}$ and $Y_{\text{QLK,i}}$ indicate the profiles computed using the NN prediction and QLK respectively. The RRMS computation is restricted to the region between the inner core and the pedestal, $0.15 \leqslant \rho_{\text{tor}} \leqslant 0.85$, as QuaLiKiz predicts zero flux in the inner core and the pedestal is not evolved in the JINTRAC runs.

Both the surrogate models considered achieve a match with QuaLiKiz that is better than 10%. The experiments carried out in this section suggest that both ADEPT and QLKNN-jetexp models can effectively replace the original transport model as a drop-in replacement for obtaining steady-state profiles, albeit the training dataset acquired by ADEPT was two orders of magnitude smaller than for QLKNN-jetexp.

As a caveat in the comparison shown here, it should be noted that the pre-processing done on the QuaLiKiz database removed points according to the physics-motivated sanity filters (see [19]). While the convergence criteria used to the generate the dataset are identical to those used in the integrated model, this filtering process is not explicitly applied to QuaLiKiz inside JINTRAC. The exact impact this filtering would have on the distinct plasma states represented in this simulation has not been characterized. However, considering that 30% of the JET-Exp-15D dataset is discarded due to the filtering, this is likely to be an important source of discrepancy

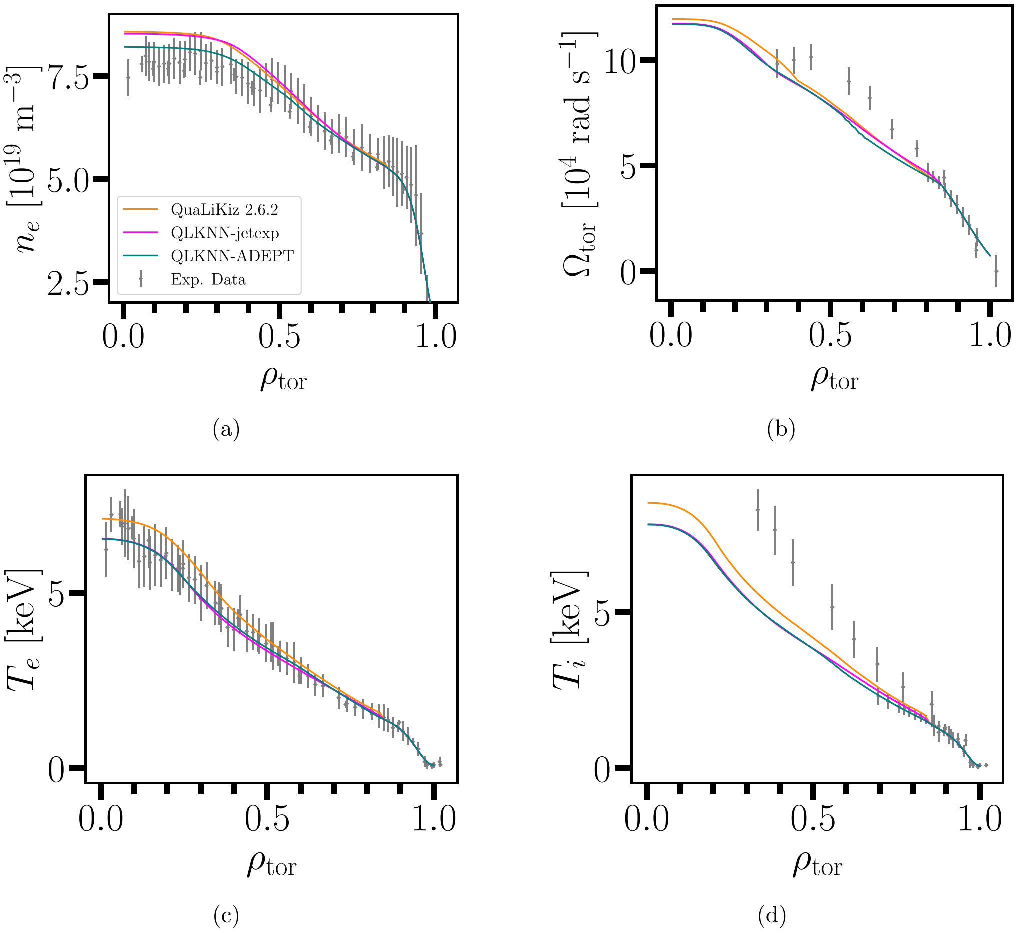


Figure 7. Comparison of the steady-state profiles from the simulation of JET#92436.

Table 4. Summary table of the JINTRAC predicted profile RRMS within the QuaLiKiz evaluation region. The values are given for the QLKNN-ADEPT simulation, with the reference QLKNN-jetexp simulation provided within the square brackets.

	JET#73342	JET#92436
T_e RRMS	1.1% [4.1%]	8.0% [8.8%]
T_i RRMS	1.7% [3.3%]	9.1% [7.3%]
n_e RRMS	7.7% [7.7%]	2.7% [0.7%]
$\Omega_{\rm tor}$ RRMS	— [—]	4.0% [3.0%]

between the JINTRAC runs that use the NNs and QuaLiKiz without filtering at runtime.

7. Summary, conclusions and future work

We presented ADEPT (Active Deep Ensembles for Plasma Turbulence), a two-stage physics-informed Active Learning framework for data-efficient surrogate models of gyrokinetic turbulence. ADEPT consists of a classifier NN that learns the boundary manifold between regions that are stable and unstable under linear gyrokinetic turbulence, thus simultaneously providing a model for the critical gradient and limiting the search space of a second surrogate regressing to the turbulent transport fluxes. Using an existing large dataset of QuaLiKiz simulations based on experimental JET discharges [19, 32], we have demonstrated a sizeable reduction in the size of the training dataset needed to obtain surrogates with integrated performance metrics comparable to surrogates trained by sampling at random. The reduction factor can be up to a factor of 10 or more, and it is due to both the adoption of Active Learning and the physics-informed nature of ADEPT enforced by the classifier, which alone results in increased data efficiency, as only a minority of the QuaLiKiz runs would have been performed deep into the stable regions of the parameter space, thus limiting the need to run costly and uninformative simulations. Compared to previous work, ADEPT delivers similar or superior performance albeit using two order of magnitude fewer data points, and a proportionally lower compute time. We also showed agreement with QuaLiKiz and previous surrogates in parameter scans and in integrated modelling applications to two very different JET discharges. The classifier stage of ADEPT may be relevant for any other model where restricting the parameter space to a certain region with desirable properties is useful, such as the case of building surrogate models of codes modelling magnetohydrodynamic instabilities [55], or identifying regions with unrealistically large GB fluxes in transport models.

While our results are extremely encouraging, the data volume required to obtain a performing surrogate valid over the sizeable but not extreme parameter space of JET still required of the order of hundreds of thousands simulations, even with Active Learning in a physics-informed setting. Much more efficient strategies should be employed to deliver actionable surrogates of higher fidelity models with high dimensionality and over wide parameter spaces. For instance, acquisition functions that do not employ the surrogate uncertainty should also be considered (e.g. [37, 56, 57]). Moreover, as noted in section 5.4, the acquisition function adopted in this work uniquely depends on the sum of the uncertainties of the NNs regressing to the turbulent fluxes. As a result, it is found that the performance of the classifier for a fixed amount of training data does depend on the number of fluxes that contribute to the acquisition function; explicitly accounting for classifier uncertainty in the acquisition function may alleviate or resolve this issue. Furthermore, while ADEPT is equipped with uncertainty quantification capabilities, it is clear from some of the T_i parameter scans in figure 5 that sometimes the uncertainty does not increase significantly outside of the training distribution, while it is sensible within it. The putative role of estimating uncertainties in surrogate models is to inform downstream tasks, such as issuing a call to the original transport model where the uncertainty is very high; however this role is challenging to fulfil if the uncertainties are poorly calibrated out of distribution. Unfortunately, quantifying the uncertainty of NNs away from the training data is a longstanding challenge [43], and benchmarking different NN uncertainty quantification strategies constitutes an important future direction of research for the Fusion surrogate modelling community. Alternatively, should current NN uncertainty quantification methods be found unreliable to perform out-of-distribution detection, other methods, such as likelihood-based generative models [58, 59] or energy-based models [60], may be adopted. A related matter is how to best treat and propagate the uncertainties in the surrogate models within integrated modelling, as naive Monte Carlo approaches would be computationally intractable. While the scope of the present work is to provide a proof of concept that a two-stage Active Learning strategy such as ADEPT is suitable for downstream integrated modelling tasks, large-scale validation of surrogate models within integrated models on a wide array experimental plasma discharges is highly recommended.

Acknowledgments

This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No. 101052200 EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them. The authors thank the reviewers for helpful and constructive comments that helped improve the manuscript. The authors thank A. Kit for comments that helped improve the manuscript.

Appendix A. Supervised machine learning with neural networks (NNs)

NNs are non-linear, parametric machine learning models based on single units called *neurons*. A collection of neurons is a *layer*, and a collection of layers connected to each other defines the *architecture* of the NN. At each layer, each neuron constructs a learnable linear combination of the outputs from the previous layer using a weight matrix W and a bias $\mathbf{b}$, the parameters of which are indicated collectively as θ. A non-linear *activation function* f is then applied,

$$\begin{aligned}\mathbf{z}_j &= W_{ij}\mathbf{a}_i + \mathbf{b}_j \\ \mathbf{a}_j &= f(\mathbf{z}_j)\,,\end{aligned} \tag{A.1}$$

where $\mathbf{a}_j$, W_{ij} and $\mathbf{b}_j$ are the output, weight matrix and bias of the current layer, while $\mathbf{a}_i$ is the output of the previous layer. The first layer ingests the data $\mathbf{x}$, so that $\mathbf{a}_0 \equiv \mathbf{x}$. If the ith and jth layers contain M and N neurons respectively, then W_{ij} will be an MxN matrix. As the layers of a NN may have different number of neurons, subsequent layers are linked by weight matrices W with suitable dimensions.

In this work, we are interested in using NNs for supervised learning. In supervised learning, a machine learning algorithm is trained on a dataset for which both x_{train} and y_{train} are known. In probabilistic terms, the algorithm learns the distribution $p(y|x)$ of labels y given an input x. During training, the discrepancy between the output of the NN $\hat{y}|\mathbf{x}_{\mathrm{train}}$ and the true output y_{train} is quantified by means of a loss functions, and this information is used to adjust the weights and biases of the NN.

Supervised learning includes both *regression* and in *classification* tasks. For regression, the labels $y \in \mathbb{R}$ are real numbers which can assume any value in the real domain. Instead, in a classification task the labels are discrete.

Appendix B. NN integrated performance metrics

We evaluate the surrogates in terms of the R^2 score for the regressors and the F1 score for the classifier, defined as follows:

$$\mathrm{R}^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} \tag{B.1}$$

$$\mathrm{F1} = 2\text{ x }\frac{\text{Precision x Recall}}{\text{Precision} + \text{Recall}} \tag{B.2}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{B.3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{B.4}$$

where $\bar{y}$ is the mean of the target flux in the test dataset, and TP, FP and FN are the true positives, false positive and false negatives. For all metrics, higher values indicate better quality of the surrogates, with a maximum value of 1.

The F1 score is more suitable than Accuracy to evaluate performance on imbalanced datasets like ours, where only 25% of inputs are unstable, and it is in general recommended for AL workflows as the relative proportion of positive and negative labels (i.e. unstable and stable regions in our case) is unknown *a priori*. To be more precise, Accuracy provides a measure of correct predictions across both positive and negative labels. In situations where datasets are heavily imbalanced, the model might focus solely on performing well with the majority class, resulting in seemingly good or even almost perfect performance, while the minority class is never predicted correctly. Thus, Accuracy alone does not shed light on the occurrences of false positives and false negatives, which are equally important to capture. Recall specifically addresses false negatives, indicating how often the model erroneously classifies something as negative when it is actually positive. Conversely, precision deals with false positives, revealing how frequently the model incorrectly labels something as positive when it is truly negative. Therefore, in regions of instability, having a high recall is beneficial as it maximizes the detection of truly unstable points while minimizing false negatives. Conversely, in stable regions a high precision is valuable because it reduces the occurrence of spurious fluxes (see also appendix G of [10]). The F1 score is the geometric average of Precision and Recall, thus capturing an overall performance across both regions while accounting for imbalanced data.

Appendix C. Further comparison between the critical gradient estimation in QLKNN-jetexp and QLKNN-ADEPT

The probability threshold p_{thresh} above which an input is considered unstable is an important hyperparameter in the QLKNN-ADEPT classifier. The clipping strategy for QLKNN-jetexp can be interpreted as performing the action of a classifier, but it does not intrinsically entail a probability of the flux being positive, which is instead what the ADEPT method provides. However, a comparison between QLKNN-ADEPT and QLKNN-jetexp in terms of a probability threshold can still be produced, as shown below.

As QLKNN-jetexp is an ensemble method, it is in principle possible to treat the outputs of QLKNN-jetexp as implementing a probability threshold. Given an ensemble of N neural networks, if M of those predict a positive flux and M-1 a negative flux, define $p_{\text{thresh}} = M/N$ as the probability threshold above which the prediction is taken to positive and below which the flux should be clipped to zero. Note that this is different than the probability currently used in ADEPT, which is given by $p_{\text{thresh}} = \frac{1}{N}\sum_i p_{\text{thresh},i}$, where $p_i = \text{softmax}(NN_i)$.

Based on the probability threshold thus derived for QLKNN-jetexp, a comparison plot with ADEPT has been produced for the classifier metrics. As can be seen in figure C1, the NNs trained with the ADEPT strategy with up to 200 000 data points always results in better precision but worse recall. The F1 score, representing a balance between the two metrics, is always better for ADEPT.

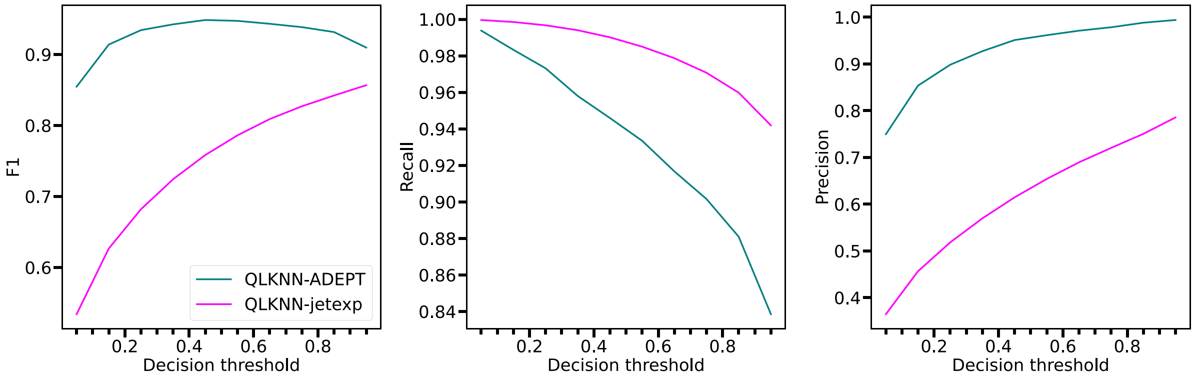


Figure C1. Comparison between the classifier metrics in QLKNN-ADEPT and QLKNN-jetexp as a function of the probability decision threshold p_{thresh} above which an input is considered unstable.

Appendix D. ADEPT vs random sampling only in the unstable region

In sections 5.1 and 5.3 it was shown that a portion of the gains of ADEPT compared to random sampling can be ascribed to the classifier, which prevents uninformative simulations to be run deep into the stable region. This section provides an experiment that helps gauge the role of the classifier. To this end, in principle, both Active Learning and random sampling could be performed in the unstable region only. However, while informative, this comparison is not reflective of production scenarios. Moreover, note that sampling (either with Active Learning or at random) amongst the unstable points is equivalent to applying a perfect classifier before performing random sampling. In ADEPT, the classifier is not perfect and therefore some of the data points collected by the current acquisition function are stable. This puts the ADEPT regressors at a disadvantage compared to random sampling from the unstable region only. It is thus instructive to evaluate the whole ADEPT pipeline, trained on the entire space, against randomly sampling only in the unstable region, which provides both a useful benchmark of how the ADEPT production scenario performs compared to a highly idealised scenario, as well as a lower limit to the performance of Active Learning in the unstable region only.

Such comparison is shown in figure D1. To further clarify, in this figure the x axis corresponds to only unstable points for random sampling, and a mix of stable and unstable points for ADEPT. Although the ADEPT regressors are theoretically at a disadvantage in terms of data efficiency, as outlined above, they still outperform random sampling for most fluxes. The only exceptions are the ion heat flux, for which Active Learning produces slightly worse results than random, and the ion particle flux, for which the gains from obtaining more training data past 200 000 points would seem negligible if the current trajectory was extrapolated. For the remaining fluxes the gain is up to a factor of 4–5. Considering the total gain of ADEPT of up to a factor of 10–20, it can be concluded that the classifier stage alone is responsible for a factor ∼3–5 gain in data efficiency.

For completeness, a comparison between ADEPT trained on one or five fluxes and random sampling in the unstable region has also been performed and it is shown in figure D2, where it is shown that training on one flux is more data efficient than random sampling, contrary to the case of five fluxes.

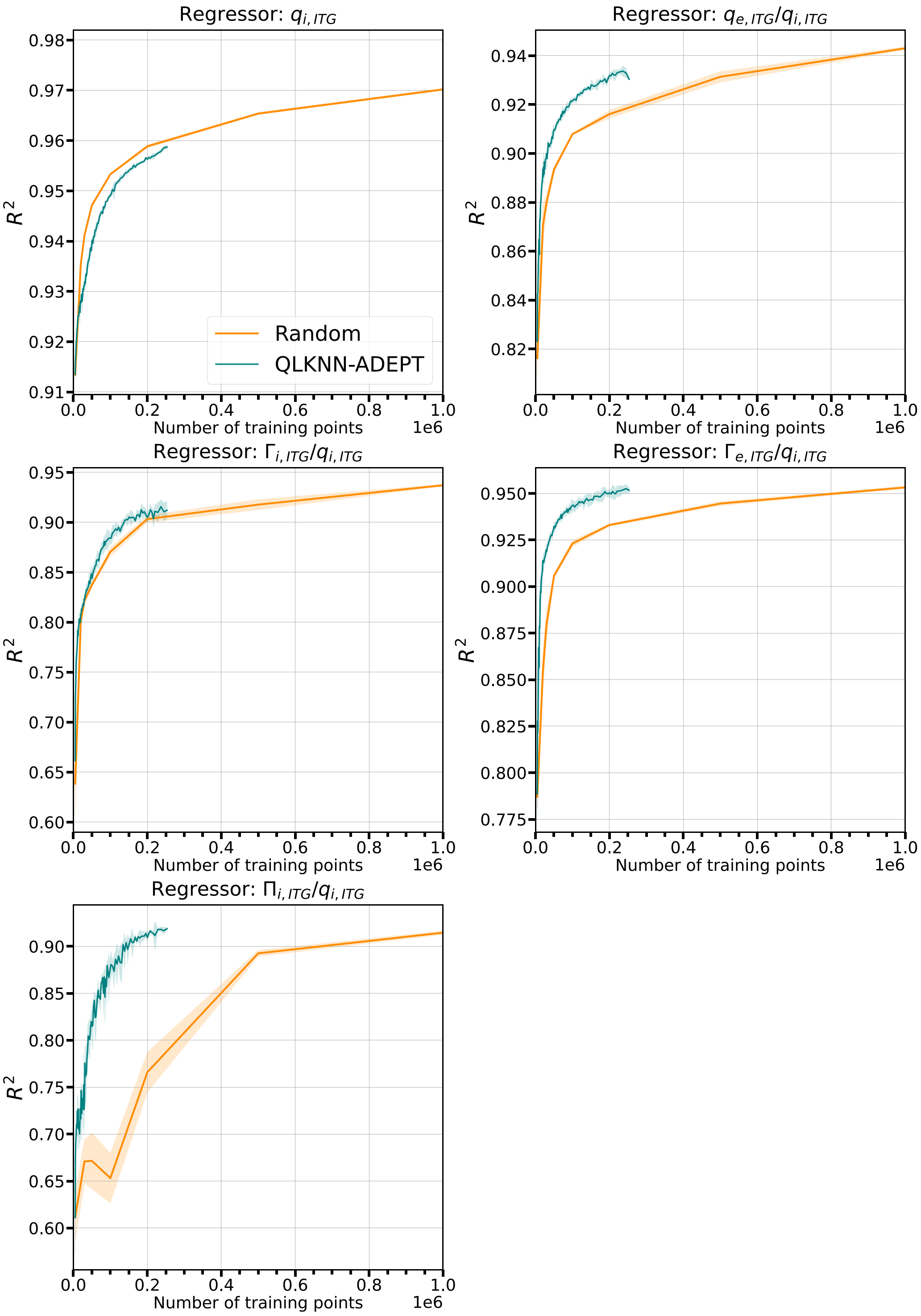


Figure D1. Same as figure 2, but random sampling is performed only in the unstable region (and therefore the panel for the classifier is omitted in this case). Despite the theoretical disadvantage (see main text for details), ADEPT is still more data efficient than random sampling in most cases.

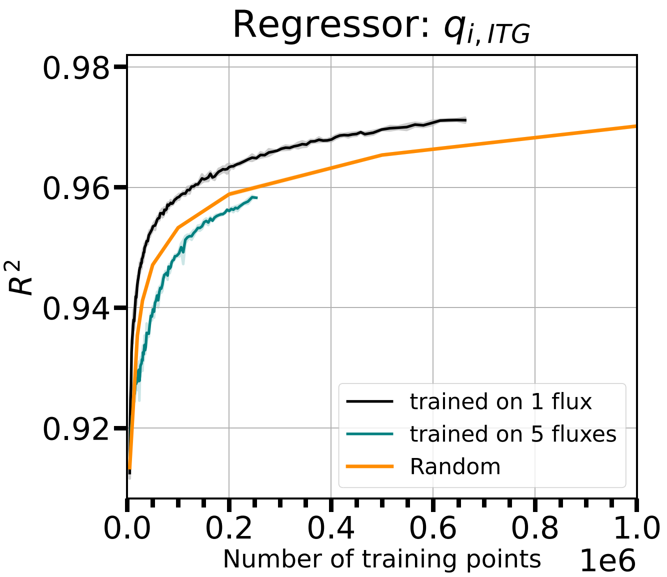


Figure D2. Same as figure 4, but random sampling is performed only in the unstable region (and therefore the panel for the classifier is omitted in this case). ADEPT is less data efficient than random sampling when trained on five fluxes (see also figure D1), but the opposite is true when trained on one flux.

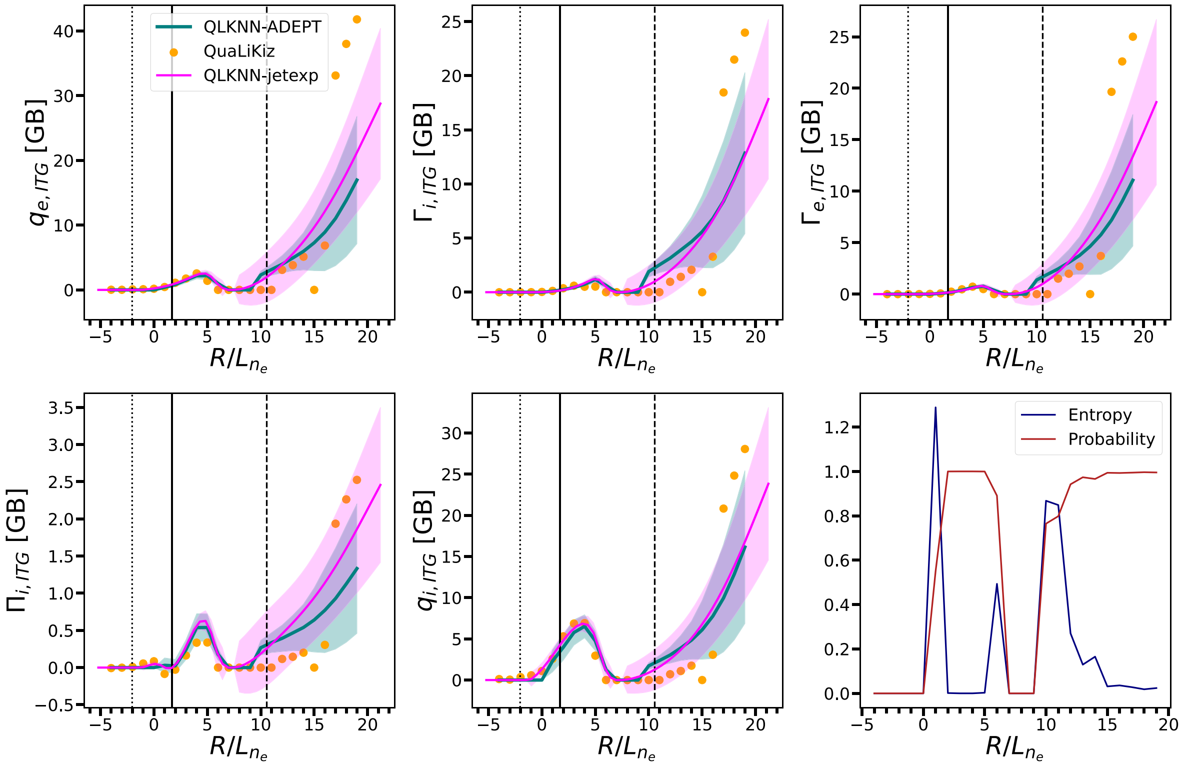


Figure E1. Parameter scans in R/L_{n_e} for the five ITG fluxes used in this work.

Appendix E. Further parameter scan validation

Further parameter scans performed using ADEPT are shown in figures E1–E4. In all the figures, the QuaLiKiz runs are shown in green, while the predictions of the surrogates are shown in red and the dashed areas indicate the 1σ confidence levels. Dotted, solid and dashed lines indicate the 2.5%, 50% and 97.5% of the distribution of the parameter being scanned. The bottom right panel shows the uncertainty for the Deep Ensemble classifier in terms of probability of an input being unstable and entropy of the ensemble.

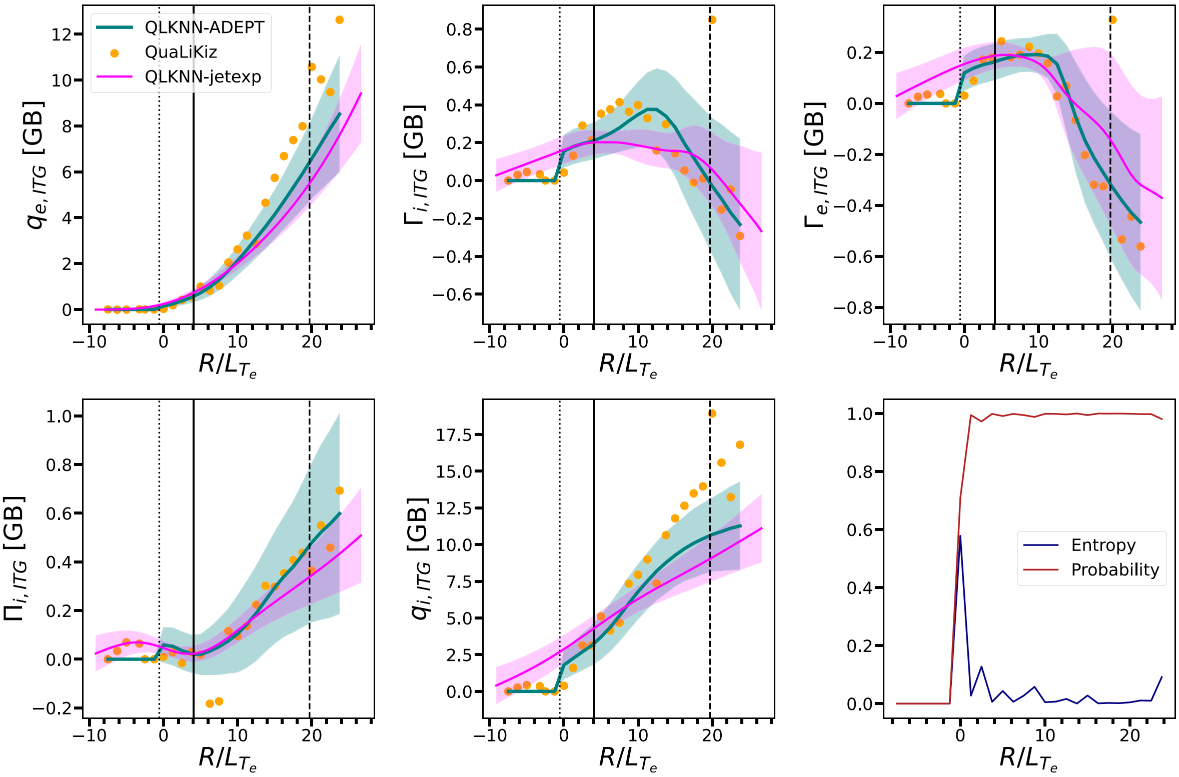


Figure E2. Parameter scans in R/L_{T_e} for the five ITG fluxes used in this work.

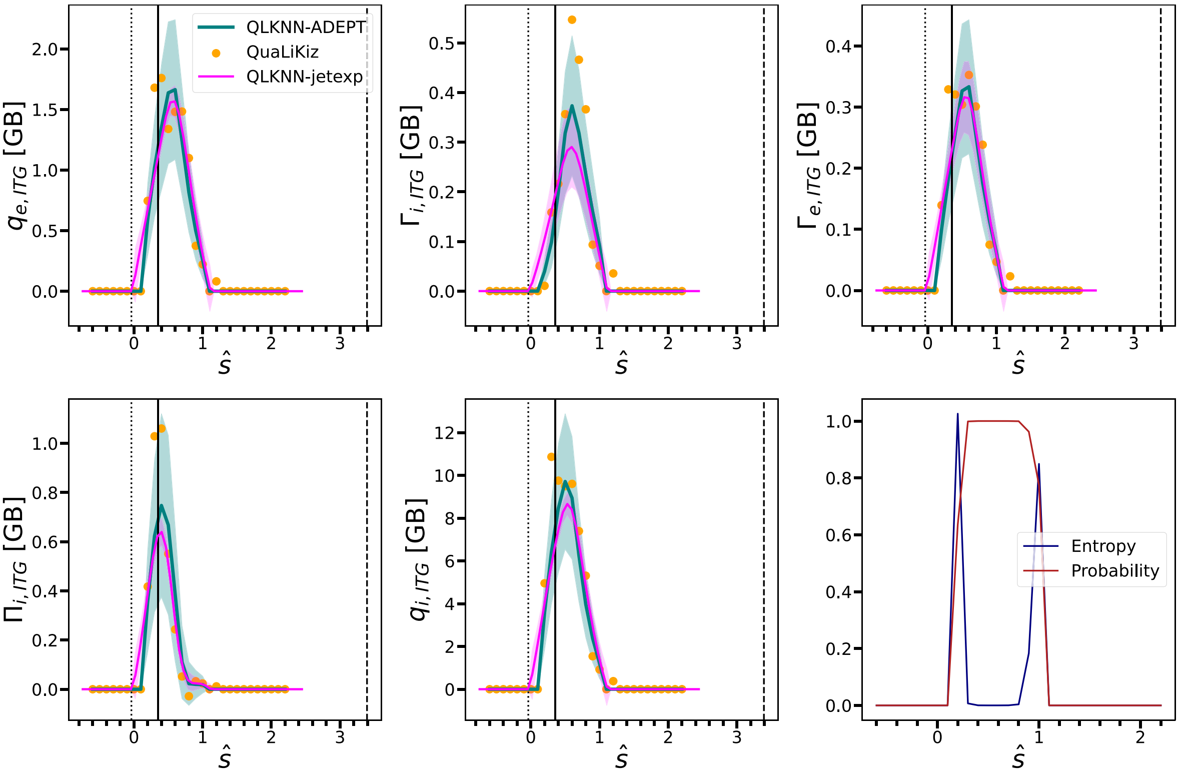


Figure E3. Parameter scans in $\hat{s}$ for the five ITG fluxes used in this work.

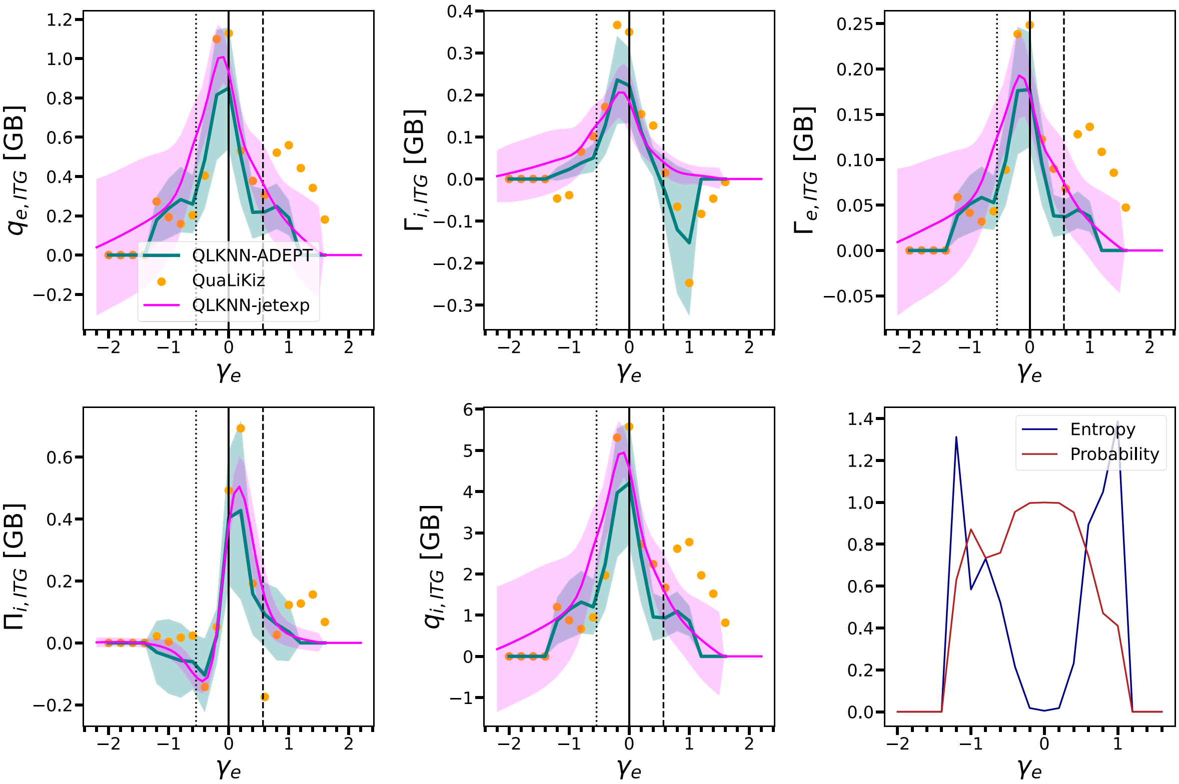


Figure E4. Parameter scans in γ_E for the five ITG fluxes used in this work.

ORCID iDs

A. Ho https://orcid.org/0000-0001-5107-3531
J. Citrin https://orcid.org/0000-0001-8007-5501
F.J. Casson https://orcid.org/0000-0001-5371-5876
V. Gopakumar https://orcid.org/0000-0003-0904-3448

References

[1] Callen J.D. 1992 Transport processes in magnetically confined plasmas *Phys. Fluids* B **4** 2142–54

[2] Hinton F.L. and Hazeltine R.D. 1976 Theory of plasma transport in toroidal confinement systems *Rev. Mod. Phys.* **48** 239–308

[3] Artaud J.F. *et al* 2010 The CRONOS suite of codes for integrated tokamak modelling *Nucl. Fusion* **50** 043001

[4] Romanelli M. *et al* 2014 JINTRAC: a system of codes for integrated simulation of tokamak scenarios *Plasma Fusion Res.* **9** 3403023

[5] Maria Poli F. 2018 Integrated tokamak modeling: when physics informs engineering and research planning *Phys. Plasmas* **25** 055602

[6] Felici F., Sauter O., Coda S., Duval B.P., Goodman T.P., Moret J.-M. and Paley J.I. (TCV Team) 2011 Real-time physics-model-based simulation of the current density profile in tokamak plasmas *Nucl. Fusion* **51** 083052

[7] Pereverzev G.V. and Yushmanov P.N. 2002 Astra—automated system for transport analysis *IPP-Report* IPP5/98 (Max Planck Institut fur Plasmaphysik)

[8] Bourdelle C., Citrin J., Baiocchi B., Casati A., Cottier P., Garbet X. and Imbeaux F. (JET Contributors) 2015 Core turbulent transport in tokamak plasmas: bridging theory and experiment with QuaLiKiz *Plasma Phys. Control. Fusion* **58** 014036

[9] Citrin J. *et al* (JET Contributors) 2017 Tractable flux-driven temperature, density and rotation profile evolution with the quasilinear gyrokinetic transport model QuaLiKiz *Plasma Phys. Control. Fusion* **59** 124005

[10] van de Plassche K.L., Citrin J., Bourdelle C., Camenen Y., Casson F.J., Dagnelie V.I., Felici F., Ho A. and Van Mulders S. (JET Contributors) 2020 Fast modeling of turbulent transport in fusion plasmas using neural networks *Phys. Plasmas* **27** 022310

[11] Staebler G.M., Kinsey J.E. and Waltz R.E. 2007 A theory-based transport model with comprehensive physics *Phys. Plasmas* **14** 055909

[12] Staebler G.M. and Kinsey J.E. 2010 Electron collisions in the trapped gyro-Landau fluid transport model *Phys. Plasmas* **17** 122309

[13] Meneghini O. *et al* 2017 Self-consistent core-pedestal transport simulations with neural network accelerated models *Nucl. Fusion* **57** 086034

[14] Felici F., Citrin J., Teplukhina A.A., Redondo J., Bourdelle C., Imbeaux F. and Sauter O. (JET Contributors and The EUROfusion MST1 Team) 2018 Real-time-capable prediction of temperature and density profiles in a tokamak using RAPTOR and a first-principle-based transport model *Nucl. Fusion* **58** 096006

[15] Van Mulders S., Felici F., Sauter O., Citrin J., Ho A., Marin M. and van de Plassche K.L. 2021 Rapid optimization of stationary tokamak plasmas in RAPTOR: demonstration for the ITER hybrid scenario with neural network surrogate transport model QLKNN *Nucl. Fusion* **61** 086019

[16] Meneghini O. *et al* 2020 Neural-network accelerated coupled core-pedestal simulations with self-consistent transport of impurities and compatible with ITER IMAS *Nucl. Fusion* **61** 026006

[17] Rodriguez-Fernandez P., Howard N.T. and Candy J. 2022 Nonlinear gyrokinetic predictions of SPARC burning plasma profiles enabled by surrogate modeling *Nucl. Fusion* **62** 076036

[18] Farcaş I.-G., Merlo G. and Jenko F. 2022 A general framework for quantifying uncertainty at scale *Commun. Eng.* **1** 43

[19] Ho A., Citrin J., Bourdelle C., Camenen Y., Casson F.J., van de Plassche K.L. and Weisen H. 2021 Neural network surrogate of QuaLiKiz using JET experimental data to populate training space *Phys. Plasmas* **28** 032305

[20] Narita E., Honda M., Nakata M., Yoshida M. and Hayashi N. 2021 Quasilinear turbulent particle and heat transport modelling with a neural-network-based approach founded on gyrokinetic calculations and experimental data *Nucl. Fusion* **61** 116041

[21] Peeters A.G., Camenen Y., Casson F.J., Hornsby W.A., Snodin A.P., Strintzi D. and Szepesi G. 2009 The nonlinear gyro-kinetic flux tube code GKW *Comput. Phys. Commun.* **180** 2650–72

[22] Citrin J., Trochim P., Goerler T., Pfau D., van de Plassche K.L. and Jenko F. 2023 Fast transport simulations with higher-fidelity surrogate models for ITER *Phys. Plasmas* **30** 062501

[23] Kremers B.J.J., Citrin J., Ho A. and van de Plassche K.L. 2023 Two-step clustering for data reduction combining DBSCAN and *k*-means clustering *Contrib. Plasma Phys.* **63** e202200177

[24] Barr J., Madula T., Zanisi L., Ho A., Citrin J. and Gopakumar V. (JET Contributors) 2022 An active learning pipeline for surrogate models of gyrokinetic turbulence *48th EPS Conf. on Plasma Physics 27 June–1 July 2022* (*Europhysics Conf. Abstracts*) (European Physical Society (EPS))

[25] Hornsby W. *et al* 2023 Gaussian process regression models for the properties of micro-tearing modes in spherical tokamak (arXiv:2309.09785 [physics.plasm-ph])

[26] Aggarwal C.C., Kong X., Gu Q., Han J. and Yu P.S. 2014 *Active Learning: A Survey* (CRC Press) pp 571–605

[27] Järvinen A.E., Fülöp T., Hirvijoki E., Hoppe M., Kit A. and Åström J. 2022 Bayesian approach for validation of runaway electron simulations *J. Plasma Phys.* **88** 905880612

[28] Škvára V., Šmídl V. and Urban J. 2018 Robust sparse linear regression for tokamak plasma boundary estimation using variational Bayes *J. Phys.: Conf. Ser.* **1047** 012015

[29] Chung Y., Char I., Neiswanger W., Kandasamy K., Oakleigh Nelson A., Boyer M.D., Kolemen E. and Schneider J. 2020 Offline contextual Bayesian optimization for nuclear fusion (arXiv:2001.01793 [cs.LG])

[30] MacKay D.J.C. 1992 Information-based objective functions for active data selection *Neural Comput.* **4** 590–604

[31] Lakshminarayanan B., Pritzel A. and Blundell C. 2017 Simple and scalable predictive uncertainty estimation using deep ensembles (arXiv:1612.01474 [stat.ML])

[32] Ho A., Citrin J., Auriemma F., Bourdelle C., Casson F.J., Kim H.-T., Manas P., Szepesi G. and Weisen H. (JET Contributors) 2019 Application of Gaussian process regression to plasma turbulent transport model validation via integrated modelling *Nucl. Fusion* **59** 056007

[33] Rasmussen C.E. 2004 Gaussian processes in machine learning *Advanced Lectures on Machine Learning* (Springer) pp 63–71

[34] Walmsley M. *et al* 2020 Galaxy Zoo: probabilistic morphology through Bayesian CNNs and active learning *Mon. Not. R. Astron. Soc.* **491** 1554–74

[35] McKay M.D., Beckman R.J. and Conover W.J. 1979 A comparison of three methods for selecting values of input variables in the analysis of output from a computer code *Technometrics* **21** 239

[36] Ko C.-W., Lee J. and Queyranne M. 1995 An exact algorithm for maximum entropy sampling *Oper. Res.* **43** 684–91

[37] Holzmüller D., Zaverkin V., Kästner J. and Steinwart I. 2022 A framework and benchmark for deep batch active learning for regression (arXiv:2203.09410 [stat.ML])

[38] Ren P., Xiao Y., Chang X., Huang P.-Y., Li Z., Gupta B.B., Chen X. and Wang X. 2021 A survey of deep active learning *ACM Comput. Surv.* **54** 180

[39] Soleimany A.P., Amini A., Goldman S., Rus D., Bhatia S.N. and Coley C.W. 2021 Evidential deep learning for guided molecular property prediction and discovery *ACS Cent. Sci.* **7** 1356–67

[40] Dasgupta S. and Hsu D. 2008 Hierarchical sampling for active learning *Proc. 25th Int. Conf. on Machine Learning (ICML'08)* (Association for Computing Machinery) pp 208–15

[41] Dimits A.M. *et al* 2000 Comparisons and physics basis of tokamak transport models and turbulence simulations *Phys. Plasmas* **7** 969–83

[42] Nguyen A., Yosinski J. and Clune J. 2015 Deep neural networks are easily fooled: high confidence predictions for unrecognizable images (arXiv:1412.1897 [cs.CV])

[43] Gawlikowski J. *et al* 2022 A survey of uncertainty in deep neural networks (arXiv:2107.03342 [cs.LG])

[44] Guo C., Pleiss G., Sun Y. and Weinberger K.Q. 2017 On calibration of modern neural networks *Proc. 34th Int. Conf. on Machine Learning* (*Proc. Machine Learning Research* vol 70) ed D. Precup and Y.W. Teh (PMLR) pp 1321–30

[45] Angelopoulos A.N. and Bates S. 2022 A gentle introduction to conformal prediction and distribution-free uncertainty quantification (arXiv:2107.07511 [cs.LG])

[46] Gneiting T. and Raftery A.E. 2007 Strictly proper scoring rules, prediction and estimation *J. Am. Stat. Assoc.* **102** 359–78

[47] Bröcker J. and Smith L.A. 2007 Scoring probabilistic forecasts: the importance of being proper *Weather Forecast.* **22** 382–8

[48] Gustafsson F.K., Danelljan M. and Schön T.B. 2020 Evaluating scalable Bayesian deep learning methods for robust computer vision (arXiv:1906.01620 [cs.LG])

[49] Yudin Y., Coster D., von Toussaint U. and Jenko F. 2023 Epistemic and aleatoric uncertainty quantification and surrogate modelling in high-performance multiscale plasma physics simulations *ICCS 2023* (Springer) pp 572–86

[50] Shannon C.E. 1948 A mathematical theory of communication *Bell Syst. Tech. J.* **27** 379–423

[51] Jenko F., Dorland W., Kotschenreuther M. and Rogers B.N. 2000 Electron temperature gradient driven turbulence *Phys. Plasmas* **7** 1904–10

[52] Militello Asp E. *et al* 2022 JINTRAC integrated simulations of ITER scenarios including fuelling and divertor power flux control for H, He and DT plasmas *Nucl. Fusion* **62** 126033

[53] Marin M., Citrin J., Bourdelle C., Camenen Y., Casson F.J., Ho A., Koechl F. and Maslov M. (JET Contributors) 2020 First-principles-based multiple-isotope particle transport modelling at JET *Nucl. Fusion* **60** 046007

[54] Ovadia Y., Fertig E., Ren J., Nado Z., Sculley D., Nowozin S., Dillon J.V., Lakshminarayanan B. and Snoek J. 2019 Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift (arXiv:1906.02530 [stat.ML])

[55] Mikhailovskii A.B. 1998 Generalized MHD for numerical stability analysis of high-performance plasmas in tokamaks *Plasma Phys. Control. Fusion* **40** 1907–20

[56] Ash J.T., Zhang C., Krishnamurthy A., Langford J. and Agarwal A. 2020 Deep batch active learning by diverse, uncertain gradient lower bounds (arXiv:1906.03671 [cs.LG])

[57] Sener O. and Savarese S. 2018 Active learning for convolutional neural networks: a core-set approach (arXiv:1708.00489 [stat.ML])

[58] Bishop C.M. 1994 Novelty detection and neural network validation *IEE Proc. Vis. Image Signal Process.* **141** 217

[59] Zanisi L. *et al* 2021 A deep learning approach to test the small-scale galaxy morphology and its relationship with star formation activity in hydrodynamical simulations *Mon. Not. R. Astron. Soc.* **501** 4359–82

[60] Liu W., Wang X., Owens J.D. and Li Y. 2021 Energy-based out-of-distribution detection (arXiv:2010.03759 [cs.LG])