Deep or Not Deep: Supervised Learning Approaches to Modeling the Pedestal Density

Adam Kit[1]**, Aaro Järvinen**[1, 2]**, Lorenzo Frassinetti**[3]**, Sven Wiesen**[4]**, and JET Contributors**[5]

[1]**University of Helsinki, FI-00014 Helsinki, Finland; adamkit@helsinki.fi**
[2]**VTT Technical Research Centre of Finland, FI-02044 VTT, Finland**
[3]**KTH Royal Institute of Technology, SE-11428 Stockholm, Sweden**
[4]**Forschungszentrum Jülich GmbH, Institut für Energie- und Klimaforschung–Plasmaphysik, DE-52425 Jülich, Germany**
[5]**See the author list of 'Overview of JET results for optimising ITER operation' by J. Mailloux et al. to be published in Nuclear Fusion Special issue: Overview and Summary Papers from the 28th Fusion Energy Conference (Nice, France, 10-15 May 2021)**

ABSTRACT

Pedestal is the key to conventional high performance plasma scenarios in tokamaks. However, high fidelity simulations of pedestal plasmas are extremely challenging due to the multiple physical processes and scales that are encompassed by tokamak pedestals. The leading paradigm for predicting the pedestal top pressure is encompassed by EPED-like models [1, 2, 3, 4, 5]. However, EPED does not predict the pedestal top density, $n_{e,ped}$, but requires it as an input. EUROPED [4] employs simplified models, such as log-linear regression, to constrain $n_{e,ped}$ with tokamak machine control parameters in an EPED-like model. However, these simplified models for $n_{e,ped}$ often show disagreements with experimental observations and do not use all of the available numerical and categorical machine control information. In this work it is observed that using the same input parameters, decision tree ensembles and deep learning models improve the predictive quality of $n_{e,ped}$ by about 23% relative to that obtained with log-linear scaling laws, measured by root mean square error. Including all of the available tokamak machine control parameters, both numerical and categorical, leads to further improvement of about 13%. Finally, predictive quality was tested when including global normalized plasma pressure and effective charge state as inputs, as these parameters are known to impact pedestals. Surprisingly, these parameters lead to only a few percent further improvement of the predictive quality.

Keywords: machine learning, fusion, pedestal

INTRODUCTION

One of the primary challenges faced by future fusion reactors is to integrate high performance fusion plasmas with tolerable conditions at the reactor components [6, 7]. The conventional approach toward a high performance fusion core in tokamak plasmas is to operate the plasma in a high confinement mode (H-mode), such that a transport barrier and pedestal are formed at the edge of the plasma [8]. The tension of core-edge integration balances at the pedestal, with mutually competing requirements of high pedestal pressure for fusion performance and mitigation of plasma heat fluxes in the scrape-off layer to avoid component overheating. Therefore, predictive capability for the pedestal region of the tokamak plasma is essential for predicting performance and core-edge integration in future fusion reactors.

First principles simulations for pedestal plasmas are extremely challenging, and the ongoing effort of the scientific community has led to models such as EPED [1, 2, 3], EUROPED [9, 4], and iPED [5] that predict the pedestal height and width based on certain simplifying assumptions. The EPED-like models encompass the leading paradigm for predicting pedestal top pressures assuming that the pedestal gradient is turbulence limited by the kinetic ballooning mode and the pedestal top pressure is magnetohydrodynamic stability limited by the peeling-ballooning mode [1, 2, 3]. However, EPED

cannot be considered a fully predictive model as, in addition to machine control parameters, EPED applies information about the confined plasma state, such as total normalized plasma plasma pressure relative to the magnetic pressure, β_N, and pedestal top density, $n_{e,ped}$, as inputs. EUROPED [4] can work exactly as EPED or can additionally employ simple core transport models to constrain β_N and models for $n_{e,ped}$. With these additional models, EUROPED is able to predict pedestal plasmas with an EPED-like model using only engineering control parameters. However, both models for $n_{e,ped}$ that are employed by EUROPED, (i) the log-linear regression given by Urano [10] based on experimental data and (ii) the neutral penetration model, show disagreements with experimental observations for the JET database, especially for $n_{e,ped}$ exceeding 8×10^{19} m^{-3} [4].

Supervised machine learning (ML) offers potential for regression beyond the standard log-linear approach to predict $n_{e,ped}$ used by EUROPED. In this work, various ML models are applied for predicting $n_{e,ped}$ using the JET pedestal database described in [11], and the performance of these models is compared to the log-linear scaling laws for $n_{e,ped}$ produced by Urano [10] and Frassinetti [11]. A related study by Gillgren et al. [12], investigated the application of shallow artificial neural networks (ANNs) with promising results for pedestal electron density and temperature predictions to provide boundary conditions for core transport simulations.

With this work we aim to answer the following questions:

1. ***What is the hierarchy of input parameters in terms of providing information to improve the pedestal density prediction quality?*** The current state-of-the-art log-linear scaling laws from Urano [10] and Frassinetti [11] use reduced subsets of all available machine control parameters as inputs. These subsets were chosen as the leading order input parameters using domain knowledge. Machine learning algorithms can utilize all possible input data, even if there are strong cross-correlations among the inputs. Therefore, it is important to compare the predictive performance of the ML algorithms when the input space is expanded to include (i) all the available numerical machine control parameters, (ii) categorical machine control parameters, such as divertor configuration, and (iii) global plasma information through normalized plasma pressure, β_N, and effective charge, Z_{eff}, that are known to impact the performance of the pedestal.

2. ***Can advanced machine learning methods like tree-based ensembles and deep learning improve prediction quality for pedestal density predictions?*** ML can encompass a wide variety of approaches, ranging from traditional linear regression to more complex deep learning networks or decision tree ensemble methods. It is not always *a priori* obvious which of these approaches is best suited for the given application. Additionally, with increased complexity, the time for training and application (inference) increases. Even though it would intuitively seem that more complex models would increase the risk of overfitting by having the capacity to memorize samples, over-parameterized deep learning networks tend to still maintain generalization capabilities [13]. This work aims to document the advantages and disadvantages of these approaches when developing a regression based predictor for $n_{e,ped}$.

JET PEDESTAL DATABASE

The data used in this work is obtained from the EUROfusion JET pedestal database [11]. A full analysis of the database can be found in [11]. The data is in tabular form with rows (entries) of shots and time windows that have columns (features) corresponding to time averaged machine control parameters and measurements of various plasma parameters.

The subset of the JET pedestal database used in this analysis includes techniques to control edge localized modes, such as kicks [14] or resonant magnetic perturbations (RMPs), [15], as well as plasmas with impurity seeding and pellets [16, 17]. We can feed this information to machine learning models in a numerically encoded format, such that shots with kicks are given a column value 1 and shots without a column value 0. This approach extends to an arbitrary number of categories as the list can be extended to as many numbers as are needed to describe the category. While the included dataset contains over 1500 shots with the JET ITER-like tungsten beryllium wall (JET-ILW) [18], there are also 422 shots with the previous carbon wall (JET-C). While a dedicated wall material flag is not given to the algorithms, the information about JET-C is mostly coded in the categorical variable containing the charge number of the impurity species in the plasma, Z_{imp}^{atomic}. However, there are 21 shots with JET-ILW and $Z_{imp}^{atomic} = 6$, due to seeding of methane. Future studies will investigate the impact of including a dedicated wall material flag.

Furthermore, a dedicated impurity injection rate value is not given in the input set, which is expected to reduce predictive performance as the database contains plasmas at varying impurity injection rates. The role of the impurity injection rate will be addressed in future studies. Even though impurity injection rate is not included explicitly, this information is partially coded into Z_{eff} and, therefore, accessible to model experiments containing Z_{eff} information as input. The major radius, R, is notably missing from the list of main engineering parameters found in Table 1. For single machine analysis, R does not vary in a meaningful way, and through the Shafranov shift of the magnetic configuration [19], encodes data about the pedestal pressure.

Table 1. Relevant parameter domains of JET pedestal database used in this analysis. Individual parameters are detailed in [11]. Features with †, ‡ correspond to those used in the Urano and Frassinetti subsets of input space respectively. While for triangularity, δ, Urano and Frassinetti used the average value, here separate values are used for the upper and lower triangularities. The divertor configuration categorical variable represents the specific divertor plate of the strike points. A plasma with a 'V/H' divertor configuration has an inner strike point on the vertical target and outer strike point on the horizontal target, and is detailed in Figure 5 of [11].

Feature (Units)	Domain [Min, Max]
Main Engineering	
I_P (MA) †, ‡	[0.813, 4.48]
B_T (T) †	[0.9616, 3.685]
a (m)	[0.8278, 0.9749]
P_{NBI} (MW) †	[0, 32.1367]
P_{ICRH} (MW)	[0, 7.963]
P_{TOT} (MW) ‡	[3.4021, 38.2205]
V_P (m^3)	[58.301, 82.1878]
q_{95} (-)	[2.425, 6.095]
δ_{lower} (-) †, ‡	[0.2318, 0.4915]
δ_{upper} (-) †, ‡	[0.0438, 0.5896]
κ (-)	[1.587, 1.819]
Γ (10^{22} e/s) †, ‡	[0, 15.502]
Categorical	
Divertor	[C/C,V/H,V/C,V/V,C/V,C/H]
$Z_{\text{imp}}^{\text{atomic}}$	[0, 2, 6, 7, 8, 10, 18, 36]
Kicks	[0, 1]
RMPs	[0, 1]
Pellets	[0, 1]
Global	
β_N^{MHD} (-)	[0.574, 3.625]
Z_{eff} (-)	[1, 3.746]
Target	
$n_{\text{e,ped}}$ (10^{19} m^{-3})	[1.4192, 11.737]

MACHINE LEARNING FOR TABULAR DATASETS

Decision tree ensembles and deep learning methods can capture many nuances of tabular datasets. A decision tree ensembles can exploit multiple decision trees fit on unique subsets of the datasets while deep learning models make use of specialized layers designed for tabular data. A brief description of each model used and why they are chosen in this analysis is given in Section 0.1. In contrast to decision tree ensembles and deep learning methods, log-linear regression seeks to find a single equation to parameterize the entirety of a dataset. Deep learning has had particular success with data like images, text and audio [20] but has yet to see the same success with structured data in tabular form where decision trees still dominate [21].

However, tree-based learning approaches do not make use of differentiable gradients during their construction, meaning they cannot be used in component pipelines combining different models and individual modules. Here, deep learning has the potential to create end-to-end pipelines for problems,

where some of the inputs could come from tabular data and others from raw diagnostic measurements or images, and a full deep learning model could be trained in one computational graph. This, as well as the potential for improved performance, has led to many proposed deep learning solutions specific for tabular data. In this work, we compare the state-of-the-art deep learning architectures designed for tabular data with popular decision tree ensemble techniques.

Unlike log-linear regression, other ML models can efficiently utilize categorical variables. In the case of pedestal databases, these variables can be specific experimental setups of a shot, e.g., the divertor configuration, which can take on one of 6 values written in Table 1. The deep networks analyzed in this paper make use of varying approaches to extract meaningful relationships within the categorical variables. These approaches include embedding or attention layers [22]. Some decision tree based methods, such as XGBoost [23] or CATBoost [24], are constructed such that they can natively produce a meaningful response to categorical inputs.

MACHINE LEARNING EXPERIMENTS

0.1 Comparing Models

We investigate which tree-based ensemble and deep learning methods have advantages over the log-linear regression used for the problem of regression of $n_{e,ped}$ using the JET pedestal database. We focus on the relative performance of different models. Model-agnostic deep learning methodologies such as data augmentation, learning rate warmup or decay [25], or pretraining [26], are not used, except in TabNet [27], which employs these in the recommended settings. While these practices have the potential to improve performance, our goal is to evaluate the intrinsic model.

The Urano and Frassinetti log-linear regression models serve as a performance baseline. We then compare that performance with the following decision tree models, which have been chosen for their generally good performance on a wide variety of datasets as seen in [21, 28, 29]:

1. **Random Forests (RF)** [30] An ensemble of decision trees, where each tree is fitted in parallel using a unique subset of the dataset (a process known as bagging) and the total forest prediction is the averaged prediction between all decision trees in the ensemble. As opposed to a log-linear regression tool trying to fit all points in the dataset at once, the RF ensemble leverages the fact that each tree fits a small subset of the total dataset, which could lead to a better generalization of the whole dataset.

2. **Extreme Randomized Trees (ERT)** [31] An implementation of RFs that leads to more diversified trees. This is achieved by changing the algorithm of how a node is split in a decision tree, where in the RF a split is made based on an information criterion, and in ERT's this split is made randomly.

3. **XGBoost** [23] A decision tree ensemble that can build problem specific trees to aid in the ensemble prediction. Instead of fitting decision trees in parallel like RF's and ERT's, gradient boosted trees (GBDTs) conducts tree fitting sequentially: decision trees are fit iterative (one at at time) with each new tree fit by applying the gradient of the error in the previous. In XGBoost, the error application is Newton-Raphson method in function space while a generic GBDT uses gradient descent. This could enable the model to build problem specific trees, e.g., trees dedicated to high $n_{e,ped}$ regression.

4. **CATBoost** [24] A GBDT implementation that leads to more balanced trees than XGBoost. In each level of the decision tree, the node split combination that leads to the best prediction is selected and used for all the nodes on that level. This could enable the model to reduce overfitting in comparison to XGBoost, while still utilizing the benefits of gradient boosting.

To compare against the tree-based ensembles, we investigate the following deep learning architectures designed specifically for tabular data:

1. **Multi-layer Perceptron (MLP)** The simplest ANN to establish the basic $n_{e,ped}$ regression capabilities of deep learning.

2. **Feed-Forward Nets with Category Embedding (FFCAT)** A simple feed forward network with additional layer types that are specifically designed for categorical input parameters. With this model we can leverage the MLP capabilities while adding functionality to the model in regards to using categorial variables.

3. **TabNet** [27] Once fit, this model selects which features to be used in prediction on a per-instance basis (attention layers). This could be useful when predicting $n_{e,ped}$, as the scaling seems to be different for low and high $n_{e,ped}$ ranges, as observed in the struggle of the log-linear regression in fitting both.

4. **AutoInt** [32] This model can learn correlations between parameters by mapping both numerical and categorical inputs into a low-dimensional space. It also utilizes attention layers like TabNet.

5. **NODE** [33] An ensemble of decision trees that are fitted instead using the gradient descent methods of ANN's. Like CATBoost, this model uses symmetric splitting of each tree, but once again in a differential form. This model leverages the benefits of decision tree ensemble methods, but has the ability to be paired with other ANN models in the case of a multi-modal pipeline.

0.2 Input Space

We investigate whether the inclusion of both categorical and more numerical input parameters leads to an increase in the performance of ML models in comparison to the inputs defined in traditional log-linear scalings. Additionally, we investigate whether the inclusion of global parameters, β_N, Z_{eff}, also aid in prediction. A fully predictive model would be able to function without these global parameters, thus it is important to determine if regressive models can function without them. All possible input parameters and their corresponding sub domains are listed in Table 1. Using these input parameters, we create 6 unique sets of possible inputs to a model, which are listed in Table 2. For input spaces U and F, we use the input parameters from the established scaling laws from Urano and Frassinetti, respectively.

Table 2. List of parameters included in each input parameter space from the categories in Table 1. The parameter sets are named by which variables they include: **E**: Main Engineering, **C**: Categorical, **B**: β_N, **Z**: Z_{eff}. **U** and **F** refer to the Urano and Frassinetti subspaces respectively.

Parameter Sets	U	F	E	EC	ECB	ECBZ	EB	EBZ
Main Engineering	†	‡	*	*	*	*	*	*
Categorical				*	*	*		
β_N					*	*	*	*
Z_{eff}						*		*

0.3 Experiment Details

For each input space, the testing of each model is done in the following systematic approach.

Preprocessing. The standard scalar transformation from the Scikit-learn library [34] was applied to the numerical features as well as the target variable, $n_{e,ped}$. This transforms each feature to a domain with mean of zero and standard deviation of 1, which reduces the impact of the relative magnitude of each feature. The categorical features are transformed into integer representations, e.g., binary features like kicks are transformed into a 1 or 0 for a shot with and without kicks, respectively.

The dataset is split into training-validation-test subsets, so that all model types use the same set of training data. Since the target variable, $n_{e,ped}$, is continuous, a pseudo stratified binning procedure is implemented such that the training-validation-test splits are even in their representation of $n_{e,ped}$. The resulting splits were examined using the Kolmogorov-Smirnov test from which the training, validation, and test sets all shared a 'pvalue' score above 0.99, which is to say each set is likely to come from the same distribution.

Hyperparameter Tuning. It was seen that the tuning of hyperparameters played little role in improving the performance metrics of each model, therefore we abstained from hyperparameter optimization in the evaluation of models and utilized the baseline model hyperparameters suggested in each model's respective papers. Often, ML models trained on large datasets benefit from large scale Bayesian optimization routines, but due to the relatively small size of the dataset this was left out.

Evaluation and Metrics. We run 15 experiments with different random seeds when splitting the dataset into training and validation sets, and report the Root-Mean-Squared-Error (RMSE) on the unseen test set.

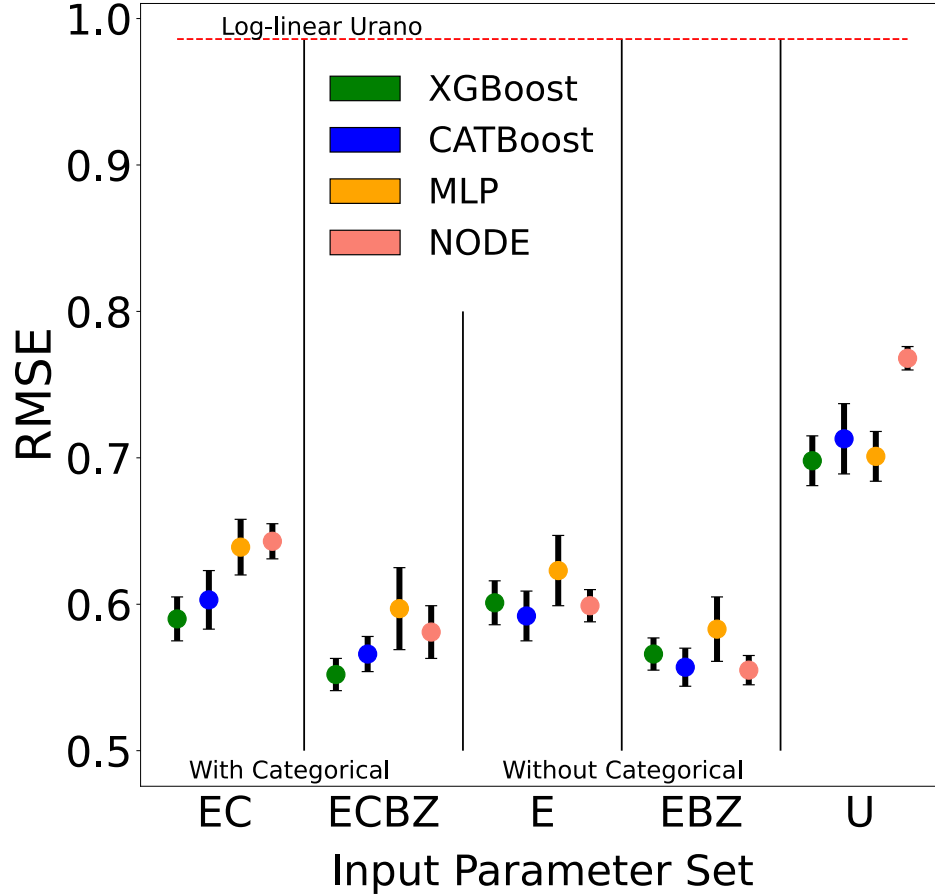## RESULTS

Model Comparison

Figure 1. Results of the RMSE (on the order of 10^{19}) of model prediction on the test set for various parameter sets and models compared to the performance of the log-linear regression used by EUROPED (dashed red line with RMSE 0.968). The average RMSE for each model is taken on the test set for 15 random splits of training/validation data. The standard deviation for the performances is visualized by the error bars. The parameter sets chosen show the differences in performance based on including/excluding categorical input parameters as well as the reduced set of input parameters used in the log-linear regression models. The models chosen are the top two performers for deep learning and decision tree ensemble groups. See Table 4 for a full table of results and standard deviations for all models and input spaces.

Model Performance

Decision tree ensembles and deep learning based regression models outperform the log-linear scaling laws by about 27% and 20% respectively when fit using the Urano input space (Fig 1 Table 4). This observation clearly shows the utility of these regression tools in building experimental models for physical observables that encompass highly non-linear dependencies in the plasmas. Decision tree based models perform systematically better than deep learning models with XGBoost performing the best. Of the deep learning architectures, NODE achieves the lowest RMSE and has very similar performance as the decision tree techniques when input spaces do not contain categorical features. This comes as no surprise, as

NODE is essentially a neural network wrapper for GBDTs. The XGBoost shows a very good regression performance throughout the $n_{\text{e,ped}}$ range, including the high density points, for which the log-linear scaling laws systematically underpredict the $n_{\text{e,ped}}$ values (Fig. 2).

Takeaway: If sufficient experimental data is available to build an interpolating regression model for $n_{\text{e,ped}}$, as is the case for many present-day tokamaks, the results indicate that using GBDT-type regression approaches is likely to outperform the conventional log-linear regression approach.
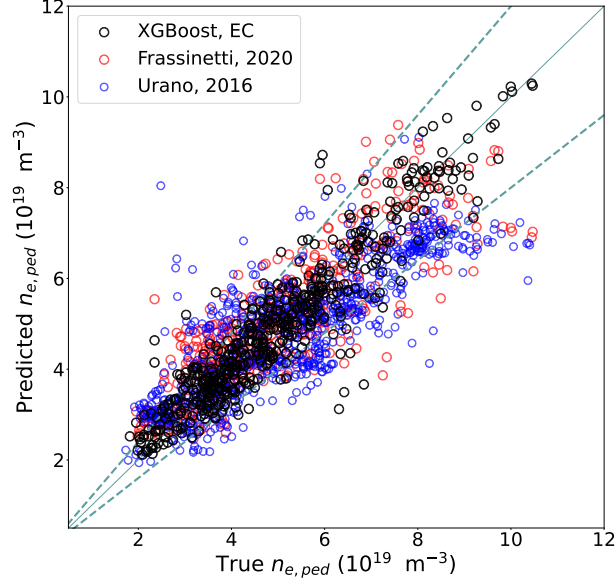
Figure 2. Predictions of $n_{\text{e,ped}}$ from the log-linear scaling law from Frassinetti [11] and Urano [10] (red and blue) compared to XGBoost model trained on parameter space **EC** (black). The log-linear scaling law shows systematic underprediction with experimental observations for $n_{\text{e,ped}} \geq 8 \times 10^{19}$ (m^{-3}), which does not occur for XGboost. The solid line marks the perfect match and the dashed lines represent 20% error

Input Spaces

Increasing input parameters to all available engineering control parameters the prediction quality is further improved by about 13% for both decision tree ensembles and deep learning models with respect to models fit using the Urano input space. This suggests that additional information in the form of shaping relations a, κ, q_{95}, V_P are important to $n_{\text{e,ped}}$ predictions.

When categorical variables are introduced, the change in prediction quality is negligible for decision tree ensembles and NODE, but leads to an average degradation by 3% for other deep learning models. The lack of improvement suggests the additional categorical variables have less impact on $n_{\text{e,ped}}$ than the numerical variables, which is surprising as divertor configuration and impurity seeding, for example, should impact $n_{\text{e,ped}}$ prediction. The decrease in quality of predictions from deep learning models may come from the categorical variables competing for relevance in the model's attention layers, which are normally utilized with more leading order numerical parameters.

Including global plasma quantities β_N and Z_{eff} leads to a further 5% increase in prediction quality for both decision tree ensembles and deep learning models. An improvement in prediction quality is expected, since β_N plays a role in pedestal performance and encodes density information via the global plasma pressure. However, the small improvement relative to only using machine parameters as inputs suggests that most of the relevant information about $n_{\text{e,ped}}$ is contained in the machine parameters, and would be a small sacrifice in order to maintain a fully predictive model.

Takeaway: We suggest to increase the input space to all available numerical machine parameters for a given surrogate model. The inclusion of β_N and Z_{eff} lead to small improvements but renders the model not fully predictive.

Table 3. For selected models, the following statistics are collected: the training time (Training) and the amount of predictions per second when performing inference for each model (Throughput). These experiments use input space **EC** from Table 2, which has 17 total features. Notation: ↓ lower is better, ↑ higher is better. Models with GPU compatibility are trained on one or more Nvidia V100s and others used 40 Intel Xeon processors. Inference timings were done using the Intel processors to eliminate differences in final performance.

Model	Random Forest	XGBoost	MLP	TabNet	NODE
Training (s) ↓	<1	5	10	465	792
Throughput (pred/s) ↑	50	850	7400	230	150

Computational Efficiency

We observed that the decision tree based techniques required 99% less time to train than the more complex deep learning methods (4). All models are capable of producing more than 50 predictions per second and thus could be fast viable components of a surrogate model pipeline.

Takeaway: We continue to suggest GBDT methods like XGBoost or CATBoost if a simple surrogate model is desired. When an end-to-end deep learning model is suitable, e.g., model pipelines consisting of other gradient based ANNs, we suggest NODE.

	Input Spaces							
Model	EC	ECB	ECBZ	E	EB	EBZ	U	F
	Decision Trees							
RF	0.642 ±0.012	0.634 ±0.012	0.624 ±0.015	0.66 ±0.012	0.637 ±0.012	0.621 ±0.013	0.733 ±0.017	0.732 ±0.019
ERT	0.62 ±0.018	0.606 ±0.018	0.58 ±0.018	0.601 ±0.015	0.592 ±0.015	0.568 ±0.015	**0.682** ±0.025	**0.674** ±0.021
XGBoost	**0.59** ±0.015	**0.574** ±0.017	**0.552** ±0.011	0.601 ±0.015	0.589 ±0.014	0.566 ±0.011	0.698 ±0.017	0.706 ±0.015
CATBoost	0.603 ±0.02	0.598 ±0.018	0.566 ±0.012	**0.592** ±0.015	**0.584** ±0.013	0.557 ±0.013	0.713 ±0.024	0.703±0.017
	Deep Learning							
MLP	0.639 ±0.019	0.60 ±0.02	0.597 ±0.028	0.623 ±0.024	0.633 ±0.02	0.583 ±0.022	0.701 ±0.017	0.760±0.012
FFCAT	0.672 ±0.018	0.663 ±0.021	0.657 ±0.026	0.655 ±0.022	0.624 ±0.015	0.623 ±0.021	0.756 ±0.014	0.763 ±0.017
TABNET	0.829 ±0.041	0.849 ±0.038	0.808 ±0.043	0.83 ±0.04	0.84 ±0.047	0.775 ±0.037	0.845 ±0.022	0.855 ±0.05
AUTOINT	0.723 ±0.027	0.71 ±0.021	0.722 ±0.021	0.696 ±0.021	0.68 ±0.024	0.69 ±0.023	0.816 ±0.026	0.832 ±0.032
NODE	0.643 ±0.012	0.621 ±0.014	0.581 ±0.018	0.599 ±0.011	0.586 ±0.012	**0.555**±0.01	0.768 ±0.008	0.769 ±0.007

Table 4. Results of experiments, where **bold** numbers represent the best performance (root-mean-squared-error and standard deviation on the order of 10^{19}) for each input space, and **green** denotes the best performance overall. The log-linear scaling laws for Urano and Frassinetti obtained a RMSE of 0.986 and 0.942 respectively.

CONCLUSION

In this work, decision tree ensembles and deep learning algorithms were explored for building regression models for $n_{\text{e,ped}}$ at JET. Based on the model experiments conducted on the EUROfusion JET pedestal database [11], decision tree ensembles and deep learning methods are viable alternatives for surrogate modeling of $n_{\text{e,ped}}$ using the JET pedestal database. Both decision tree ensembles and deep learning architectures outperform log-linear regression by 23% when applying the same set of input parameters as used in log-linear regressions by Frassinetti [11] and Urano [10]. Extending the input space to more available machine control parameters, some of which are strongly cross-correlated, we achieve an additional 13% improvement. Including global parameters, such as β_N and Z_{eff}, to the input space, improves prediction quality further by 5% but also renders the regression model not fully predictive. Unexpectedly, the categorical parameters, such as divertor configuration or atomic number of seeded impurity, did not seem to impact the prediction quality significantly, even though divertor configuration is know to impact pedestal density in tokamaks [17]. The detailed investigations for the reasons for this observation are a subject for future studies.

This work can serve as a basis for further developments of using decision tree based or deep learning models as alternatives for $n_{\text{e,ped}}$ regression in pedestal models, such as EUROPED.

ACKNOWLEDGMENTS

REFERENCES

[1] P. B. Snyder, R. J. Groebner, A. W. Leonard, T. H. Osborne, and H. R. Wilson. Development and validation of a predictive model for the pedestal height. *Physics of Plasmas*, 16(5):056118, 2009. doi: 10.1063/1.3122146. URL https://doi.org/10.1063/1.3122146.

[2] P.B. Snyder et al. A first-principles predictive model of the pedestal height and width: development, testing and ITER optimization with the EPED model. *Nuclear Fusion*, 51(10):103016, 2011. doi: 10.1088/0029-5515/51/10/103016. URL https://doi.org/10.1088/0029-5515/51/10/103016.

[3] P. B. Snyder, T. H. Osborne, K. H. Burrell, R. J. Groebner, A. W. Leonard, R. Nazikian, D. M. Orlov, O. Schmitz, M. R. Wade, and H. R. Wilson. The EPED pedestal model and edge localized mode-suppressed regimes: Studies of quiescent H-mode and development of a model for edge localized mode suppression via resonant magnetic perturbations. *Physics of Plasmas*, 19(5):056115, 2012. doi: 10.1063/1.3699623. URL https://doi.org/10.1063/1.3699623.

[4] S. Saarelma, L. Frassinetti, et al. Self-consistent pedestal prediction for JET-ILW in preparation of the DT campaign. *Physics of Plasmas*, 26(7):072501, 2019. doi: 10.1063/1.5096870. URL https://doi.org/10.1063/1.5096870.

[5] M G Dunne, L Frassinetti, M N A Beurskens, M Cavedon, S Fietz, R Fischer, L Giannone, G T A Huijsmans, B Kurzan, F Laggner, P J McCarthy, R M McDermott, G Tardini, E Viezzer, M Willens-dorfer, E Wolfrum, and and. Global performance enhancements via pedestal optimisation on ASDEX upgrade. *Plasma Physics and Controlled Fusion*, 59(2):025010, dec 2016. doi: 10.1088/1361-6587/59/2/025010. URL https://doi.org/10.1088/1361-6587/59/2/025010.

[6] K. Ikeda. Progress in the ITER physics basis. *Nuclear Fusion*, 47(6), jun 2007. doi: 10.1088/0029-5515/47/6/e01. URL https://doi.org/10.1088/0029-5515/47/6/e01.

[7] H. Zohm, F. Träuble, W. Biel, E. Fable, R. Kemp, H. Lux, M. Siccinio, and R. Wenninger. A stepladder approach to a tokamak fusion power plant. *Nuclear Fusion*, 57(8):086002, jun 2017. doi: 10.1088/1741-4326/aa739e. URL https://doi.org/10.1088/1741-4326/aa739e.

[8] H Zohm. Edge localized modes (ELMs). *Plasma Physics and Controlled Fusion*, 38(2):105–128, feb 1996. doi: 10.1088/0741-3335/38/2/001. URL https://doi.org/10.1088/0741-3335/38/2/001.

[9] S Saarelma, C D Challis, L Garzotti, L Frassinetti, C F Maggi, M Romanelli, and C Stokes and. Integrated modelling of h-mode pedestal and confinement in JET-ILW. *Plasma Physics and Controlled Fusion*, 60(1):014042, nov 2017. doi: 10.1088/1361-6587/aa8d45. URL https://doi.org/10.1088/1361-6587/aa8d45.

[10] J. H. Urano, C. F. Maggi, and E. Joffrin. Characterization of electron density based on operational parameters in JET H-mode plasmas with C and ILW. In *Proceedings of the 43rd EPS Conference on Plasma Physics*, page O4.121, Leuven, Belgium, 4–8 Jul 2016.

[11] L. Frassinetti, S. Saarelma, G. Verdoolaege, M. Groth, J.C. Hillesheim, P. Bilkova, P. Bohm, M. Dunne, R. Fridström, E. Giovannozzi, F. Imbeaux, B. Labit, E. de la Luna, C. Maggi, M. Owsiak, and R. Scannell. Pedestal structure, stability and scalings in JET-ILW: the EUROfusion JET-ILW pedestal database. *Nuclear Fusion*, 61(1):016001, nov 2020. doi: 10.1088/1741-4326/abb79e. URL https://doi.org/10.1088/1741-4326/abb79e.

[12] A. Gillgren, E. Fransson, D. Yadykin, L. Frassinetti, P. Strand, and JET contributors. Enabling adaptive pedestals in predictive transport simulations using neural networks. Manuscript submitted for publication, 2021.

[13] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3):107–115, feb 2021. ISSN 0001-0782. doi: 10.1145/3446776. URL `https://doi.org/10.1145/3446776`.

[14] E. de la Luna, I.T. Chapman, F. Rimini, P.J. Lomas, G. Saibene, F. Koechl, R. Sartori, S. Saarelma, R. Albanese, J. Flanagan, F. Maviglia, V. Parail, A.C.C. Sips, and E.R. Solano and. Understanding the physics of ELM pacing via vertical kicks in JET in view of ITER. *Nuclear Fusion*, 56(2): 026001, dec 2015. doi: 10.1088/0029-5515/56/2/026001. URL `https://doi.org/10.1088/0029-5515/56/2/026001`.

[15] Todd Evans, Richard Moyer, Kieth Burrell, Max Fenstermacher, Ilon Joseph, Anthony Leonard, Thomas Osborn, Gary Porter, Michael Schaffer, Phillib Snyder, Raul Thomas, Jonathan Watkins, and William West. Edge stability and transport control with resonant magnetic perturbations in collisionless tokamak plasmas. *Nature Physics*, 2006. doi: https://doi.org/10.1038/nphys312.

[16] P. T. Lang, K. Büchl, M. Kaufmann, R. S. Lang, V. Mertens, H. W. Müller, and J. Neuhauser. High-efficiency plasma refuelling by pellet injection from the magnetic high-field side into asdex upgrade. *Phys. Rev. Lett.*, 79:1487–1490, Aug 1997. doi: 10.1103/PhysRevLett.79.1487. URL `https://link.aps.org/doi/10.1103/PhysRevLett.79.1487`.

[17] C Giroud, S Jachmich, P Jacquet, A Järvinen, E Lerche, F Rimini, L Aho-Mantila, N Aiba, I Balboa, P Belo, C Angioni, M Beurskens, S Brezinsek, F J Casson, I Coffey, G Cunningham, E Delabie, S Devaux, P Drewelow, L Frassinetti, A Figueiredo, A Huber, J Hillesheim, L Garzotti, M Goniche, M Groth, Hyun-Tae Kim, M Leyland, P Lomas, G Maddison, S Marsen, G Matthews, A Meigs, S Menmuir, T Puetterich, G van Rooij, S Saarelma, M Stamp, H Urano, and A Webster and. Progress at JET in integrating ITER-relevant core and edge plasmas within the constraints of an ITER-like wall. *Plasma Physics and Controlled Fusion*, 57(3):035004, feb 2015. doi: 10.1088/0741-3335/57/3/035004. URL `https://doi.org/10.1088/0741-3335/57/3/035004`.

[18] G F Matthews, M Beurskens, S Brezinsek, M Groth, E Joffrin, A Loving, M Kear, M-L Mayoral, R Neu, P Prior, V Riccardo, F Rimini, M Rubel, G Sips, E Villedieu, P de Vries, and M L Watkins and. JET ITER-like wall—overview and experimental programme. *Physica Scripta*, T145:014001, dec 2011. doi: 10.1088/0031-8949/2011/t145/014001. URL `https://doi.org/10.1088/0031-8949/2011/t145/014001`.

[19] T.I. Gutkin and S.N. Lozovsky. Equilibrium of a toroidal pinch in a static magnetic field and in a high-frequency large-amplitude field. *Nuclear Fusion*, 14(4):475–481, sep 1974. doi: 10.1088/0029-5515/14/4/001. URL `https://doi.org/10.1088/0029-5515/14/4/001`.

[20] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.

[21] Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. *CoRR*, abs/2106.03253, 2021. URL `https://arxiv.org/abs/2106.03253`.

[22] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 3111–3119, Red Hook, NY, USA, 2013. Curran Associates Inc.

[23] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4232-2. doi: 10.1145/2939672.2939785. URL `http://doi.acm.org/10.1145/2939672.2939785`.

[24] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features, 2019.

[25] Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: training imagenet in 1 hour. *CoRR*, abs/1706.02677, 2017. URL `http://arxiv.org/abs/1706.02677`.

[26] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? *J. Mach. Learn. Res.*, 11: 625–660, March 2010. ISSN 1532-4435.

[27] Sercan Ömer Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. *CoRR*, abs/1908.07442, 2019. URL `http://arxiv.org/abs/1908.07442`.

[28] L. Katzir, G. Elidan, and Ran El-Yaniv. Net-dnf: Effective deep modeling of tabular data. In *ICLR*, 2021.

[29] Yu. V. Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. *ArXiv*, abs/2106.11959, 2021.

[30] Leo Breiman. Random forests. *Machine Learning*, 45, 2001. doi: 10.1023/A:1010933404324.

[31] Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine Learning*, 63, 2006. doi: 10.1007/s10994-006-6226-1.

[32] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. Autoint: Automatic feature interaction learning via self-attentive neural networks. *CoRR*, abs/1810.11921, 2018. URL `http://arxiv.org/abs/1810.11921`.

[33] Sergei Popov, Stanislav Morozov, and Artem Babenko. Neural oblivious decision ensembles for deep learning on tabular data. *CoRR*, abs/1909.06312, 2019. URL `http://arxiv.org/abs/1909.06312`.

[34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.