Towards Applications of Deep Learning Techniques to Establish Surrogate Models for the Power Exhaust in Tokamaks

M. Brenzke[1], S. Wiesen[1], M. Bernert[2], D.P. Coster[2], U. von Toussaint[2], EUROfusion MST1 Team* and the ASDEX Upgrade Team†

[1] *Forschungszentrum Jülich, Institut für Energie- und Klimaforschung - Plasmaphysik, Partner of the Trilateral Euregio Cluster (TEC), 52425 Jülich, Germany*

[2] *Max Planck Institute for Plasma Physics, 85748 Garching, Germany*

For future fusion devices, such as ITER and DEMO, significant power dissipation in the plasma edge will be required in order not to severely damage the divertor [1]. In light of this it is desirable to model the power flux to the divertor targets before a discharge in order to get an estimate of the expected thermal load at the target plates. However, analytical models of the power exhaust of tokamaks only exist for attached conditions in which dissipation is too small so that the divertor targets of future fusion devices would suffer significant damage. For modeling the physically more complex regime of plasma detachment there exist coupled fluid-kinetic codes, such as SOLPS-ITER. The disadvantage of such codes is that an adequate simulation of a discharge and therefore the prediction of the expected thermal loads causes significant computational overhead and is very time consuming. Furthermore, due to the complex nature of the plasma interactions that have to be taken into account in detachment, some processes are only approximated in these codes. In this work we present first steps towards data-driven modeling of thermal loads at the divertor targets using machine learning approaches to extract a model from experimental data and utilize that model for prediction. Extracting a model from experimental data has the advantage that predictions can be obtained significantly faster than with a full simulation once the model is set up. Moreover, this approach might also enable us to implicitly model the complex physics of detachment further.

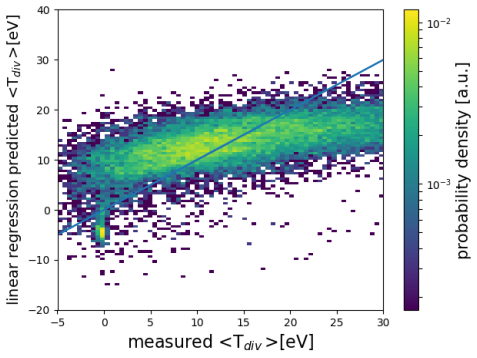
In this first approach we analyzed experimental data from $\sim$3600 discharges performed in ASDEX Upgrade, each carried out in a lower single null configuration. For each discharge we focused on stable operational phases which we defined via the plasma current. For every discharge we analyzed time intervals of 0.2 s and if the plasma current did not vary by more than 10% of the mean value over this time window we considered this to be a stable phase. All signals we considered were then averaged over this time interval to obtain one data point per 0.2 s interval. After obtaining the data points we shuffled them in order not to select data points from only a few different discharges. The input parameters to our model are the plasma current (I_p), the toroidal magnetic field (B_t), total applied heating power (P_{tot}), line integrated electron

*See the author list of B. Labit et al. 2019 Nucl. Fusion accepted (https://doi.org/10.1088/1741-4326/ab2211)

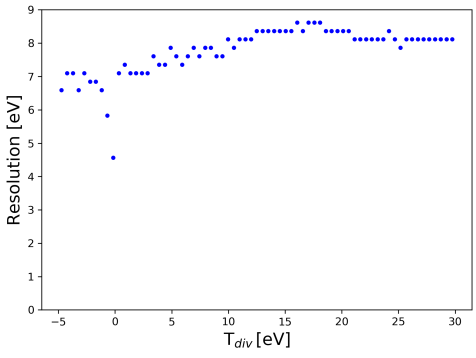
†See the author list of H. Meyer et al. 2019 Nucl. Fusion accepted (https://doi.org/10.1088/1741-4326/ab18b8)

density in the core ($n_{e,core}$), triangularity (δ_{upper} and δ_{lower}), elongation (κ) and the total gas puffs of each of the species hydrogen, deuterium, helium, neon and nitrogen ($\Gamma_{[H,D,He,Ne,N]}$). The output parameter is T_{div} which is, in a certain range, a real time estimate of the electron temperature in front of the outer divertor target [2]. After obtaining the time averaged values of all relevant signals (<I_p>, <B_t>, ... <T_{div}>) we removed all data points that yielded a negative time averaged density (either in the core or in the edge), a total applied heating power below 0 MW or above 20 MW, a positive value for the magnetic field strength or a fraction of total radiated power over total heating power greater than 2. We introduced these constraints on the data in order to remove erroneous measurements and because the data base did not contain enough data points with a positive magnetic field. Moreover, we limited the range of <T_{div}> to [-5 eV, 30 eV]. After applying these cuts ∼78500 data points remained for our analysis. For each of the following models we normalized the input quantities by subtracting the mean value of the data used for fitting/training the model and dividing by the standard deviation of the training set.

As a baseline model we tested a least squares multiple linear regression. For this model we used 70% of the data (∼55000 data points) for fitting and held out the remaining 30% (∼23500 data points) for the evaluation of the model's performance after the fit. Figure 1a shows the performance achieved on the test set with this very simple approach. Depicted is the probability density of the predicted value of <T_{div}> obtained from the model versus the measured value extracted from the data. The blue line indicates a 1:1 correlation which would be desirable. To



(a) *Probability density of predicted <T_{div}> from multiple linear regression vs. measured values acquired from experimental data; blue line indicates 1:1 correlation*

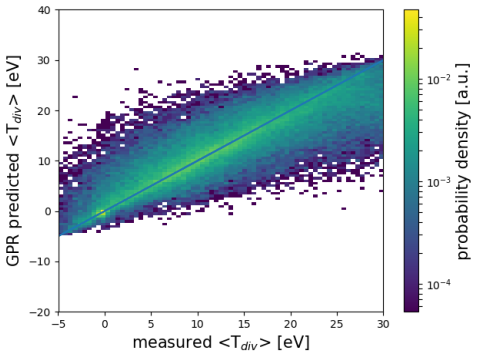


(b) *Resolution of the multiple linear regression model; smaller values indicate a more certain prediction*

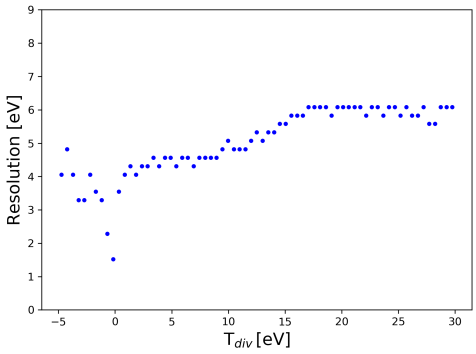
characterize the model's performance in one value we determined the median of the distribution of relative differences between model and measurement given by $\frac{|<T_{div,measured}>-<T_{div,model}>|}{|<T_{div,measured}>|}$. The uncertainties on this value arise from the central 68^{th} percentile of the distribution. With a median relative difference of $0.348^{+1.302}_{-0.238}$ the model's predictions are off by ∼35% on average. As a further criterion for model performance we determined the model's resolution as measure of

the prediction's scatter around their expectation analogously to [3]. Figure 1b shows the model's resolution as a function of $<T_{div}>$, note that a smaller value indicates a more certain prediction and therefore a better performance. With a resolution of around 7 eV to 8 eV the model predictions scatter quite strongly around their expectation.

For a more sophisticated approach we also tested Gaussian Process Regression (GPR) [4] to obtain predictions for $<T_{div}>$. We used a rational quadratic kernel with an additional noise term. Since the computations for GPR scale as $\sim n_{fit}^3$ we limited the number of data points used for fitting the model to 7% of the data resulting in ∼5500 data points for the fit. We used the rest of the data set for testing the model. Figures 2a and 2b show the performance of the GPR approach.



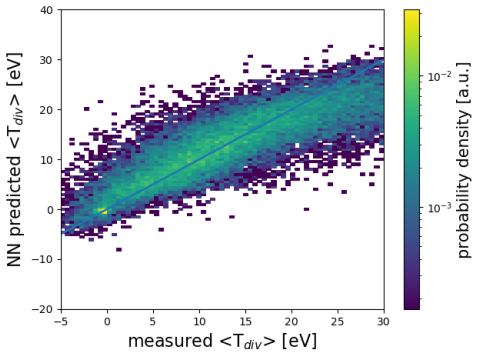
(a) *Probability density of predicted $<T_{div}>$ from Gaussian Process Regression vs. measured values acquired from experimental data; blue line indicates 1:1 correlation*



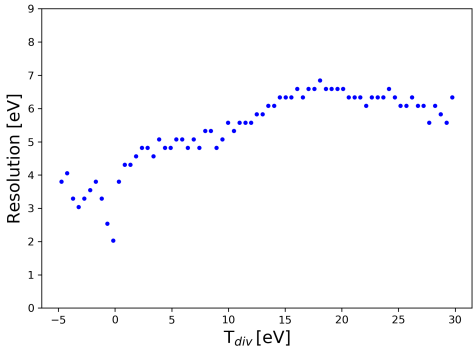
(b) *Resolution of the Gaussian Process Regression model; smaller values indicate a more certain prediction*

With a median relative difference of $0.182^{+0.454}_{-0.139}$ the model shows a better performance than the simple linear approach. However, within the uncertainties on the median relative differences the two models are comparable. In terms of resolution the model obtained from the GPR tends to perform better than the linear model but still on a comparable level.

As a third approach we set up a fully connected Neural Network (see e.g. [5]). In this first step we configured a Neural Network (NN) with three hidden layers with 100 neurons each and one output neuron. We trained the network on 70% of the data for 2000 epochs. Of this data we used 30% for validation during the training process. The rest of the data set was used for testing the model. We used a standard mean squared error loss function with an L1 regularizer term and the Adagrad [6] optimizer. We applied ELU activation functions [7] after each hidden layer and a linear activation in the output.



(a) *Probability density of predicted <T_{div}> from Neural Network vs. measured values acquired from experimental data; blue line indicates 1:1 correlation*



(b) *Resolution of the Neural Network model; smaller values indicate a more certain prediction*

Figures 3a and 3b show the performance of the NN. With a value of $0.231^{+0.541}_{-0.164}$ the median relative difference is slightly worse than that of the GPR model but still comparable to both other models within the given uncertainties. The resolution shows that the model is still on a comparable level to both other models.

To summarize, we set up an automated data extraction that gives temporal averages of general plasma properties from which we tried to predict a proxy variable for the power exhaust. We then tested different machine learning approaches to model the dependency of power exhaust on these input quantities by treating every time average as independent data. All the models tested still show a large scatter around the 1:1 correlation between prediction and the corresponding expected values extracted from the experiment. However, taking into account the complex physics and the simplicity of our approach achieving a tendency towards a 1:1 correlation is a first, albeit small, success. Although the GPR and the NN approach tend to fit the desired 1:1 correlation better than the simple linear model, both GPR and NN yield performances comparable to the linear model. Thus, in order to obtain a significant gain in performance, we might have to reevaluate the general approach of using this very simplified time averaged data set. A next step could be to include the temporal information by using time series as input to a Neural Network enabling us to also take transients into account.

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 and 2019-2020 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

[1] A. Kallenbach et al. 2019 *Nuclear Materials and Energy* **18** 166-174

[2] A. Kallenbach et al. 2010 *Plasma Phys. Control. Fusion* **52** 055002

[3] M. G. Aartsen et al. (IceCube Collaboration) 2014 *Energy Reconstruction Methods in the IceCube Neutrino Telescope* Journal of Instrumentation 9 P03009 doi: 10.1088/1748-0221/9/03/P03009

[4] C. E. Rasmussen and C. K. I. Williams 2006 *Gaussian Processes for Machine Learning* MIT Press ISBN: 026218253

[5] Michael A. Nielsen 2015 *Neural Networks and Deep Learning* Determination Press http://neuralnetworksanddeeplearning.com/

[6] J. Duchi, E. Hazan and Y. Singer 2011 *Adaptive Subgradient Methods for Online Learning and Stochastic Optimization* Journal of Machine Learning Research 12 2121-2159

[7] D.-A. Clevert, Th. Unterthiner and S. Hochreiter 2015 *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)* CoRR abs/1511.07289 http://arxiv.org/abs/1511.07289