Research Article

www.molinf.com

molecular informatics

doi.org/10.1002/minf.202200043

# Using Active Learning to Develop Machine Learning Models for Reaction Yield Prediction

Simon Viet Johansson[+],[a, b] Hampus Gummesson Svensson[+],*[a, b] Esben Bjerrum,[a] Alexander Schliep,[b] Morteza Haghir Chehreghani,[b] Christian Tyrchan,[c] and Ola Engkvist[a, b]

**Abstract:** Computer aided synthesis planning, suggesting synthetic routes for molecules of interest, is a rapidly growing field. The machine learning methods used are often dependent on access to large datasets for training, but finite experimental budgets limit how much data can be obtained from experiments. This suggests the use of schemes for data collection such as active learning, which identifies the data points of highest impact for model accuracy, and which has been used in recent studies with success. However, little has been done to explore the robustness of the methods predicting reaction yield when used together with active learning to reduce the amount of experimental data needed for training. This study aims to investigate the influence of machine learning algorithms and the number of initial data points on reaction yield prediction for two public high-throughput experimentation datasets. Our results show that active learning based on output margin reached a pre-defined AUROC faster than random sampling on both datasets. Analysis of feature importance of the trained machine learning models suggests active learning had a larger influence on the model accuracy when only a few features were important for the model prediction.

**Keywords:** Active Learning · Reaction Yield Prediction · Bayesian Matrix Factorization · Random Forest · Neural Networks

## Introduction

The recent advances in computer aided synthesis planning (CASP)[1–6] have made it a promising tool for finding and assessing plausible chemical routes. Accurately finding and assessing chemical routes can help to reduce the time required to find novel drugs and materials.[7] One goal within CASP is to increase the likelihood of finding efficient routes to produce a target compound. This uses forward prediction: predicting reaction outcomes in a forward synthesis.[8–10] Current research in forward prediction utilizes machine learning (ML) to build models to more rapidly evaluate a suggested reaction, i.e., predicting whether the reaction is plausible or not. Accurate ML methods are usually data hungry, which can become a problem when the data is not always consistent, if at all available in a machine-readable format.[11]

High-throughput experimentation (HTE) has emerged as a time- and material efficient technique for producing large amounts of chemical reaction data.[12,13] HTE is thus a suitable approach to generate combinatorial datasets for CASP,[14] which requires large training sets. Recent developments in HTE, and automation in general, have enabled platforms that can conduct and analyse thousands of experiments per day as demonstrated for batch chemistry[15,16] as well as for flow chemistry.[17] However, it is not feasible to use HTE to investigate all necessary permutations of reaction variables in a typical reaction.[18] Therefore, it is important to identify the most informative data points, e.g., finding the smallest subset of data points that provides the most information about a reaction to a given machine learning model.

Active learning (AL) is a subfield of machine learning exploring different strategies of finding the most informative data points.[19] The aim is to determine which subset of data points that maximises the learning and performance of a machine learning model. During training, the learning model then autonomously queries those points to be labelled and uses these labels to incrementally improve.[19] One approach to active learning, called pool-based sampling,[20] assumes a small pool of labelled data $L$ and a large pool of unlabelled data $U$.[21] Labels of unlabelled data are obtained by querying the labels of data points of $U$

[a] *S. Viet Johansson,[+] H. Gummesson Svensson,[+] E. Bjerrum, O. Engkvist*
*Molecular AI, Discovery Sciences, R&D, AstraZeneca, SE-431 83, Mölndal, Sweden*
*E-mail: hamsven@chalmers.se*
[b] *S. Viet Johansson,[+] H. Gummesson Svensson,[+] A. Schliep, M. Haghir Chehreghani, O. Engkvist*
*Department of Computer Science and Engineering, Chalmers University of Technology and University of Gothenburg, SE-412 96, Göteborg, Sweden*
[c] *C. Tyrchan*
*Medicinal Chemistry, Research and Early Development, Respiratory and Immunology (R&I), BioPharmaceuticals R&D, AstraZeneca, SE-431 83, Mölndal, Sweden*
[+] *Contributed equally.*

Supporting information for this article is available on the WWW under https://doi.org/10.1002/minf.202200043

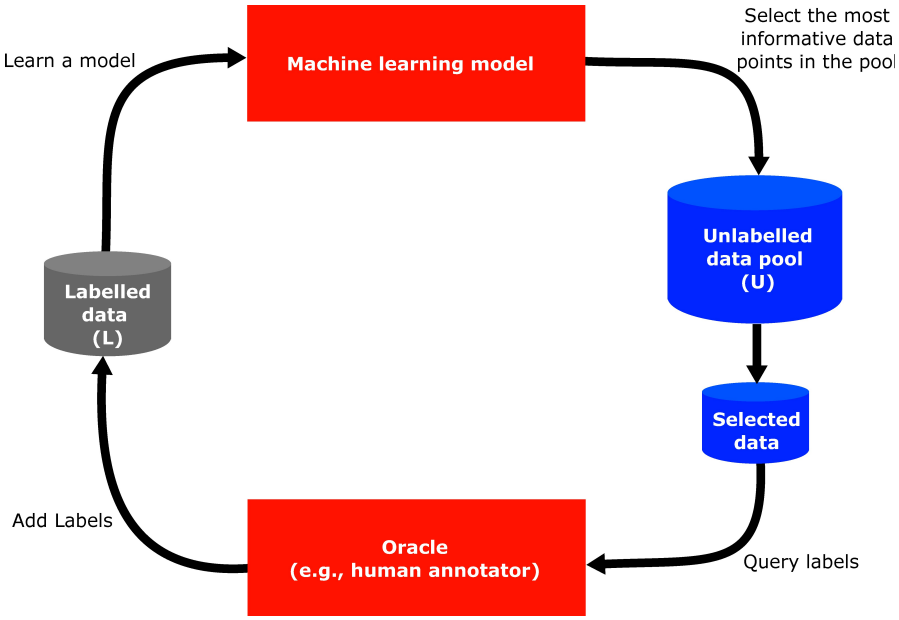


**Figure 1.** The pool-based active learning cycle.

from an oracle, e.g., previously known results or conducted experiments. These data points are then added to the pool of labelled data *L* and used to improve the model, as illustrated in Figure 1. The goal in this case is to find the optimal model with the smallest possible pool of labelled data since each query of labels is associated with a cost, such as the cost of conducting an experiment. The problem is then to determine the data points that are most informative and whose labels therefore should be queried from the oracle. The data points that should be included in each query are determined by an acquisition function that estimates the informativeness of each data point. One popular type of active learning strategies is uncertainty sampling where the most uncertain data points, i.e., the data points the learning model knows the least about, are assumed to be most informative. A pre-determined acquisition function then determines the most uncertain batch of unlabelled data points whose labels should be queried from the oracle. Thus, data points that the model is already confident about are avoided.

In computational chemistry, active learning has been applied for several applications including drug design.[22–24] For HTE, recent studies have applied active learning to select data points for neural network models.[25] These had positive impact on reducing the number of experiments needed to generate a training set for predicting the reaction yield.[26] However, active learning still struggles to show a significant performance gain compared to randomly selecting data points to query, so called random sampling, when only a few data points have been labelled.[26] It is also possible that the performance of active learning differs depending on settings such as initial size and machine learning algorithm used.

In this work, we have explored different settings of active learning in HTE to better understand when and if active learning can help a machine learning model reach a pre-defined level of accuracy for reaction yield prediction faster than random sampling. We investigated this in a forward prediction setting where reactions are predicted to be either successful or unsuccessful depending on the reaction yield. The rationale for using a binary classification model is that in discovery chemistry a reaction only needs to provide sufficient yield.[27] The aim is to develop a model that covers the whole design space of building blocks and reaction conditions. This is different from process chemistry where the objective is to find the reaction conditions that maximizes the yield. In particular, we have investigated uncertainty-based active learning with the Margin strategy on different machine learning models and different numbers of initial (labelled) data points for modelling of reaction data.

## Methods

We explored different settings for active learning for predicting the reaction yield of two combinatorially generated datasets. The model used for prediction and initial size were varied. The goal is to develop a machine learning model with a pre-defined accuracy for binary classification of the reaction yield with the labels "successful" reaction and "unsuccessful" reaction.

  

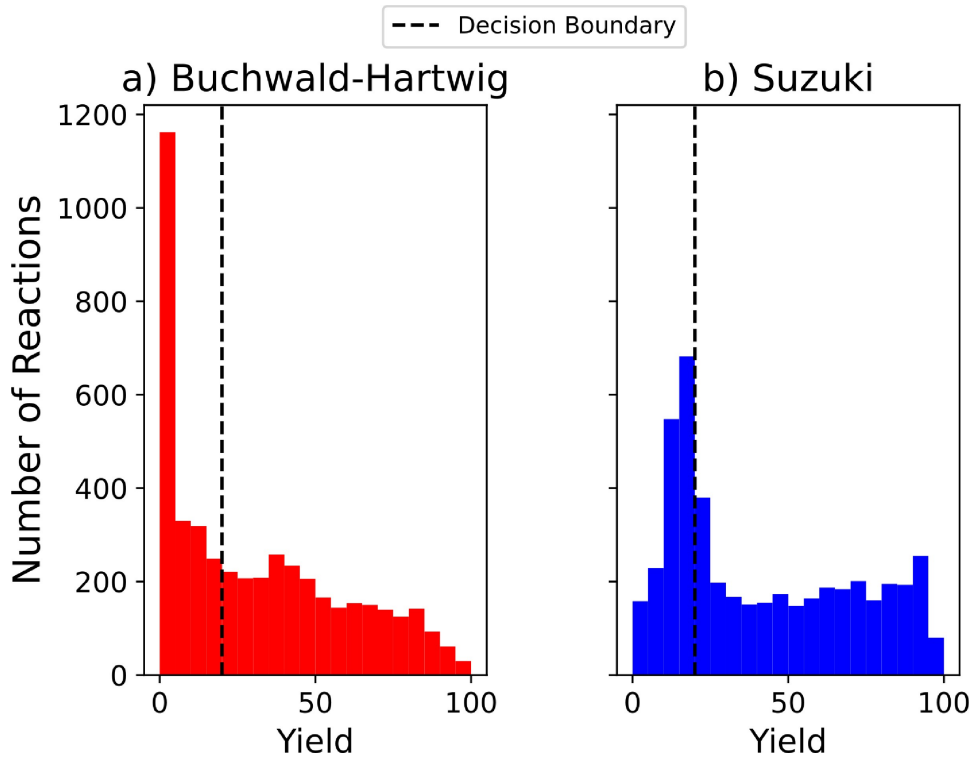


**Figure 2.** The distributions of reaction yield of the 4068 reactions in the Buchwald-Hartwig and Suzuki datasets.

## Datasets

Previous studies from HTE provide fully combinatorial datasets that can be used to benchmark active learning strategies. Thus, this is a retrospective study where all true labels are known beforehand. This changes the question to an interpolation problem with the goal to reduce the number of experiments. We assume that only the true labels of the initial training and queried data points are known at each active learning cycle. We explored a Buchwald-Hartwig reaction dataset[16] and a Suzuki reaction dataset.[17] The Suzuki dataset consists of 5769 Suzuki-Miyaura couplings with five varied reaction variables, namely, reactant 1, reactant 2, ligand, base and solvent. We discarded the fourth choice of reactant 2 to obtain a fully combinatorial dataset which consisted of 4608 data points. The Buchwald-Hartwig dataset consists of 4608 cross-couplings of aryl halides with four reaction variables, aryl halide, additive, ligand and base.

One-hot encodings were used to represent the different combinations of reaction variables described above in the datasets. Both datasets consisted of the reaction yield of every combination of reaction variables. The distribution of reaction yields for all 4608 reactions in each dataset are displayed in Figure 2. This is a study from the discovery chemistry perspective, rather than process optimization.[27] As such, we are not interested in necessarily finding the optimal conditions w.r.t reaction yield. Instead, the important result is whether or not a reaction is possible with "sufficient" yield. Therefore, a hard threshold of 20% yield was used to determine the label of each data point, i.e., one combination of reaction variables. A reaction with a yield above this threshold was labelled as a "successful" reaction, encoded as class 1. In the same way, a reaction with a yield below 20% was labelled as an "unsuccessful" reaction, encoded as class 0. The proportion of "successful" reactions in the Buchwald-Hartwig and Suzuki dataset are 54% and 65%, respectively. In order to evaluate the performance of the machine learning models on these datasets, each dataset was randomly split into a training and evaluation set consisting of 80% and 20%, respectively, of the whole dataset. In combinatorial library design, we are interested in reducing the needed data rather than predicting particular, hold-out, features. For this reason, we do not follow the experimental design used by Ahneman et al to show model generalizability.[16] We evaluated the performance on the evaluation set in terms of the area under the receiver operating characteristic curve (AUROC) with the goal of reaching the largest AUROC with the least number of labelled training data points from each dataset. To be robust against fluctuations in the predictions, we used a moving average to determine if an experiment reached the pre-defined desired AUROC. A run was determined to have reached the (desired) AUROC after $k$ queries if:

  

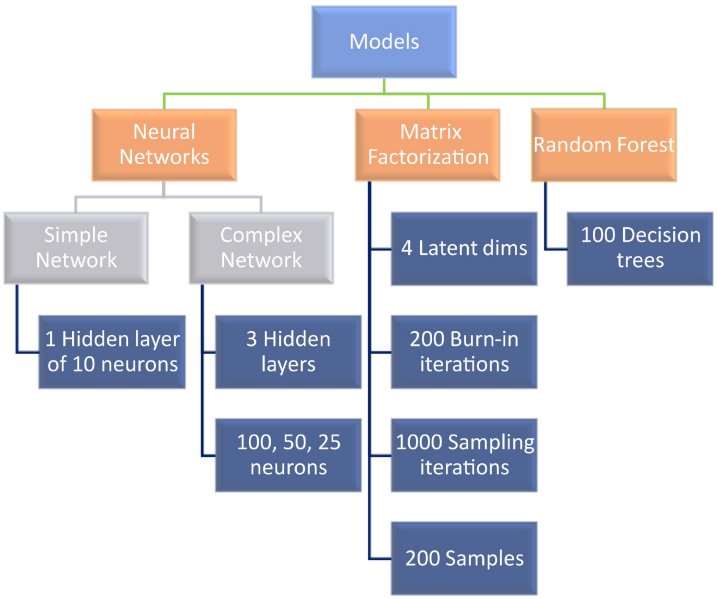


**Figure 3.** Schematic illustration of the different machine learning models that were investigated.

$$\frac{\sum_{i=k-n}^{k+n} \mathrm{AUROC}_i}{2n+1} \geq \mathrm{AUROC}_{\mathrm{target}},$$

where $\mathrm{AUROC}_i$ is the AUROC after the $i$-th query, $\mathrm{AUROC}_{\mathrm{target}}$ is the desired AUROC and $n$ is the number of values of the moving average considered before and after the $k$-th query. In this study, $n = 3$. Zero-padding was used to obtain a moving average for all number of queries.

### *Initial Pool of Labels*

In order to investigate how the initial size affects the performance of active learning, we evaluated three different initial sizes of labelled training data for active learning, namely, 10, 100 and 1000 labelled (training) data points. Five sets were randomly selected for each initial size, which gives 15 different initial pools per dataset in total.

### Models

For both datasets, we investigated four different model types: (1) simple neural network; (2) complex neural network; (3) Bayesian matrix factorization model;[28] (4) random forest classifier[29,30] from scikit-learn.[31] The models are visualised in Figure 3. We trained the models using cumulative learning, i.e., each model was retrained after every query. Cumulative learning usually obtains better results in active learning compared to incremental learning.[32]

### *Neural Networks*

The neural networks were implemented using PyTorch 1.3.1[33] together with the PyTorch wrapper PyTorch Lightning 1.0.8.[34] The experiments were conducted using Nvidia K80 GPUs with driver version 450.80.02, CUDA 11.0. The simple neural network had one hidden layer with 10 neurons; while the complex neural network had three hidden layers with 100, 50 and 25 neurons, respectively. The initially designed architectures displayed reasonable performances and no further attempts to optimize the architectures were tried. The output layers consisted of two neurons (one for each class) and the input layer corresponded to the one-hot encoding. The complex neural network used dropout with probability 0.5 while the simple neural network used no dropout. Moreover, the networks used Leaky ReLU[35] as activation functions for the hidden layers and softmax as the activation function of the output. Optimization of the parameters was performed using AdamW[36] with a learning rate of 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.99$, $\varepsilon = 10^{-8}$, a weight decay of 0.01 and a mini-batch size of 8. The networks were trained for 50 epochs after each query of active learning. For each initial pool (of labels) of a specific initial size, the same initial weights were used in every first epoch of training. Since training was cumulative, this means that the same initial weights were used to retrain the neural networks after each query.

### *Bayesian Matrix Factorization*

The matrix factorization method Macau[37] in the Smurff 0.16.0[28] framework was used for Bayesian matrix factoriza-

  

tion. The model used four latent dimensions and 1200 training iterations using a probit noise model, of which 200 where burn-in samples and 1000 used for Bayesian sampling. We sampled every 5th iteration, which yielded 200 predictions. The probability assigned to a label was the frequency that the label had been estimated to be the most likely label in the 200 predictions.

### *Random Forest*

The random forest classifier in Scikit-learn 0.24.2 was used.[31] The random forest model consisted of 100 decision trees. The model used the Gini impurity criterion for assessing the quality of the node splits. On initialisation of each iteration, the individual trees were given bootstrapped datasets.

### Active Learning

As seen in Figure 1, when the machine learning model had been trained on the labelled data, a query strategy was used to determine the most informative data point of the unlabelled data pool. In this work, we have investigated an uncertainty-based active learning strategy called *Margin*.[19] Margin queries data points based on the output margin

$$x^* = \underset{x}{\operatorname{argmin}}[P_\theta(\hat{y}_1|\, x) - \; P_\theta(\hat{y}_2|\, x)],$$

where $P_\theta(y|x)$ is the probability, of an arbitrary classification algorithm, that the true label of a data point $x$ is label $y$, and $\hat{y}_1$ and $\hat{y}_2$ are the most and second most probable labels, respectively, according to the classification algorithm. If we have binary labels and model them as Bernoulli random variables, ranking unlabelled data points based on maximal variance will provide the same order as using Margin. Hence, for binary labels, Margin will try to reduce the variance of the predictions of the unlabelled data. Moreover, better performance of Margin compared to other strategies, such as querying based on the maximum entropy and random sampling, has been observed in prior work.[32,38] Margin was compared to random sampling, where every unlabelled data point is assumed to be equally informative. Hence, when using random sampling, data points were labelled at random, rather than using the trained machine learning model to determine the most informative ones to label. Furthermore, for both Margin and random sampling, we evaluated the active learning setting where a batch of one (1) data point is queried (and subsequently labelled) for each active learning cycle. That is, at every cycle, the label, i.e., the true reaction yield, of one (1) combination of reaction variables was queried and subsequently added to the training data. Consequently, this single new label and previously known labels were used to retrain the machine learning model.We investigated how the different combinations of model, query strategy (either random sampling or Margin), and initial size affect the required number of labelled data points to achieve specific pre-defined levels of AUROC. The pre-defined levels of AUROC were 0.800, 0.850, 0.900, 0.950 and 0.975. For each combination of model, query strategy and initial size, we investigated the performance until all training data points had been labelled. Moreover, each combination was run five times to investigate the stochastic behaviours of the models and query strategies.

## Results

In this section, we present our comparison of different settings of active learning: initial size, machine learning model and dataset. The objective was to develop a model for reaction yield prediction, with specific performance requirements. We compared the number of data points that had to be labelled to reach the pre-defined levels of AUROC on the evaluation set: 0.800, 0.850, 0.900, 0.950 and 0.975. Random sampling was used as a baseline to compare the effects these settings have on active learning. During each active learning cycle, using either Margin or random sampling, one (1) data point was queried. To avoid any potential bias in the initial data, each combination of settings was investigated with five different initial sets. Computations for each initial set were repeated five times, for a total of 25 runs. We show these results in Figures 4 and 5 for the Buchwald-Hartwig reaction data, and Figures 6 and 7 for the Suzuki reaction data. To show the variation within each setting, Figures 4 and 6 make use of boxplots to display the outliers, the minimum (excluding outliers), the maximum (excluding outliers), the sample median, the first quartile and the third quartile. Furthermore, Tables 1 and 2 highlight the difference between the average number of labels that was queried within each setting to reach the pre-defined levels of AUROC.

### Buchwald-Hartwig Reaction Data

Figures 4(a)–(c) show boxplots that illustrate the required number of labelled data points to reach an AUROC of 0.800, 0.850 and 0.900, respectively, for the Buchwald-Hartwig reaction data when starting with either 10 or 100 labelled data points. These boxplots were extracted from the computed AUROCs of each active learning cycle, which are displayed in Figure S1. Note that, the labelled data points include the initial points (labels). Starting with 1000 labelled data points reached the target AUROC scores using only the initial data points and are, therefore, omitted in these figures. Hence, when utilizing 1000 initial points, no additional data were needed to obtain an AUROC of 0.800, 0.850 and 0.900. When utilizing Margin with the complex

  

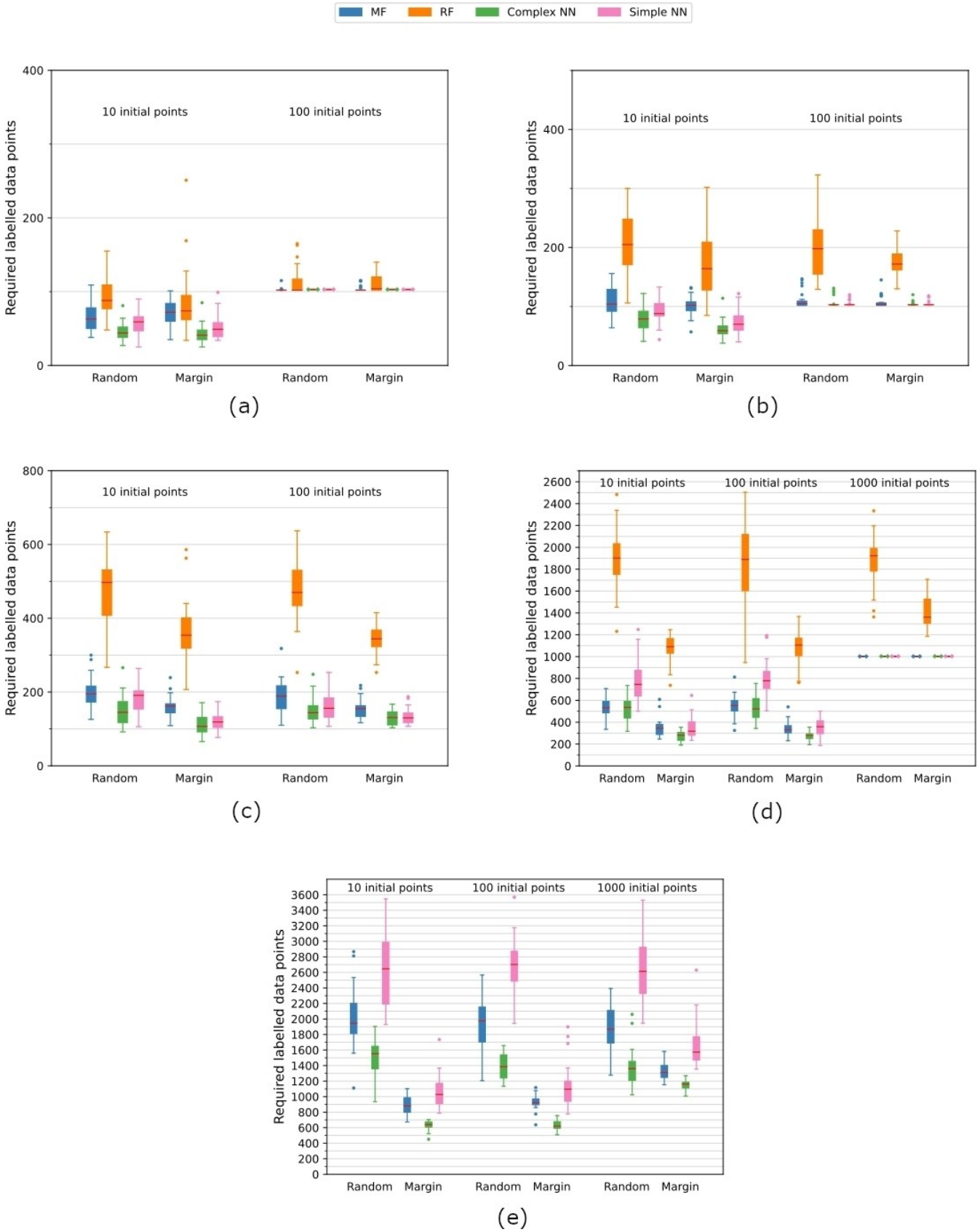


**Figure 4.** Boxplots showing the number of required labelled data points for the Buchwald-Hartwig reaction to achieve an AUROC of (a) 0.800, (b) 0.850, (c) 0.900, (d) 0.950 and (e) 0.975. When using an initial size of 1000, all models reached the AUROC of 0.800, 0.850 and 0.900 by using only the initial labels and, therefore, these models are not shown.

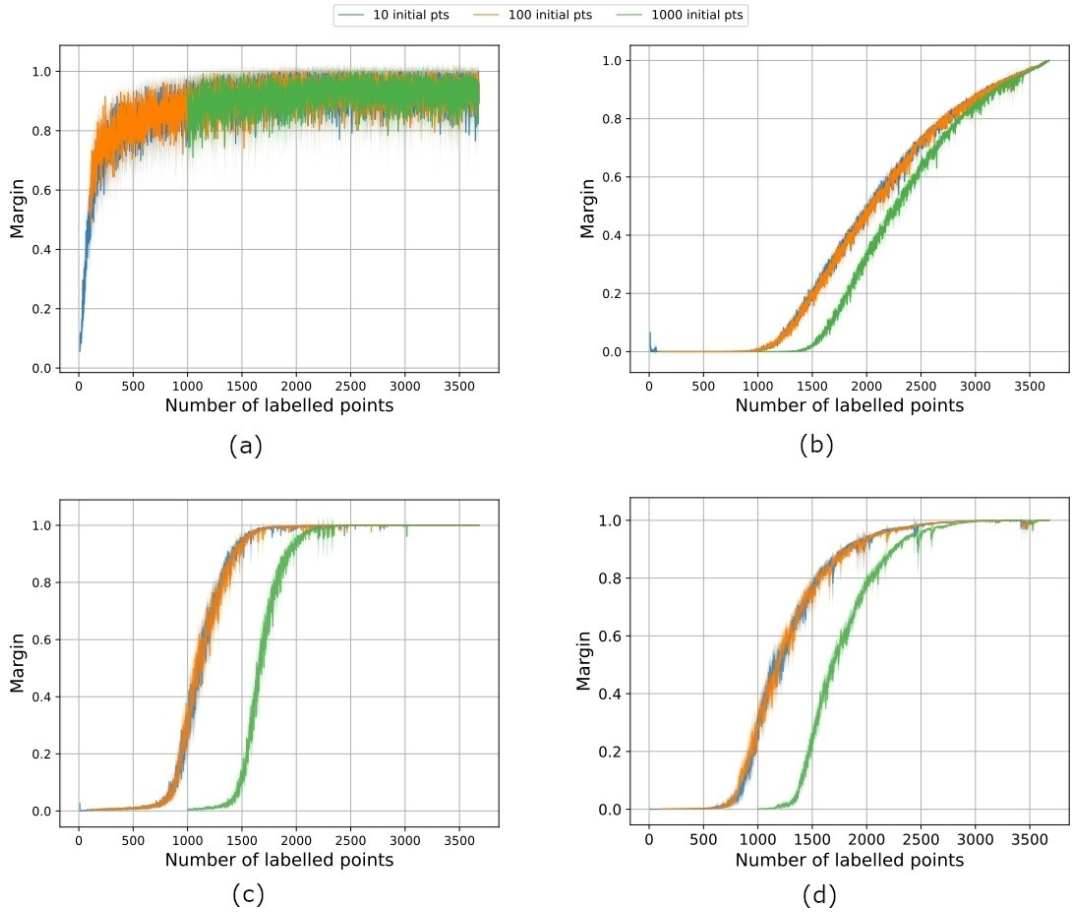


**Figure 5.** Output margins of queried data points, averaged over the 25 runs, for (a) matrix factorization, (b) random forest, (c) complex neural network and (d) simple neural network when starting with 10 labels and using Margin as acquisition function on the Buchwald-Hartwig reaction data. Displays the 95% approximate confidence intervals of the averages over the 25 runs.

and simple neural network on the Buchwald-Hartwig data, starting with 10 labelled data points seems to require a lower number of labelled data points to achieve an AUROC of 0.800 or 0.850, compared to starting with 100 labelled data points. Utilizing matrix factorization and random forest starting with either 10 or 100 labelled data points show no substantial difference of required data points to label to reach an AUROC of 0.850. No substantial difference is displayed when utilizing random sampling compared to Margin to obtain an AUROC of 0.800, 0.850 and 0.900.

Figures 4(d)–(e) display boxplots of the required number of labelled data points to reach an AUROC of 0.950 and 0.975, respectively, when starting with either 10, 100 or 1000 labelled data points. Both figures show a lower number of required labelled data points when using Margin, compared to using random sampling. The greatest difference is observed when we want to obtain an AUROC of 0.975. Random forest was not able to achieve this AUROC. When utilizing Margin, matrix factorization and the simple neural network seems to show a similar performance, compared to random sampling where they show more distinguishable performances. In particular, this trend is visible when using 10 or 100 initial points. Also, we can see that the variation seems to be greater when using random sampling.

In Figures 4(a)–(e) we see that using Margin together with the complex network consistently requires a lower number of labelled data points, while random forest consistently requires a greater number. In general, the difference seems become greater for every greater AUROC that we want to achieve.

Table 1 shows the differences of the number of queried labels required between random sampling and Margin averaged over all 25 runs of each setting on the Buchwald-Hartwig dataset. Note that, the queried labels exclude the initial points (labels). A positive difference means that Margin requires less labels to reach the target AUROC, while a negative difference means that random sampling requires less labels to be queried. We see that to obtain an AUROC of 0.800, the differences are both negative and positive, meaning that Margin and random sampling seems show similar performances. On the other hand, when we want to

  

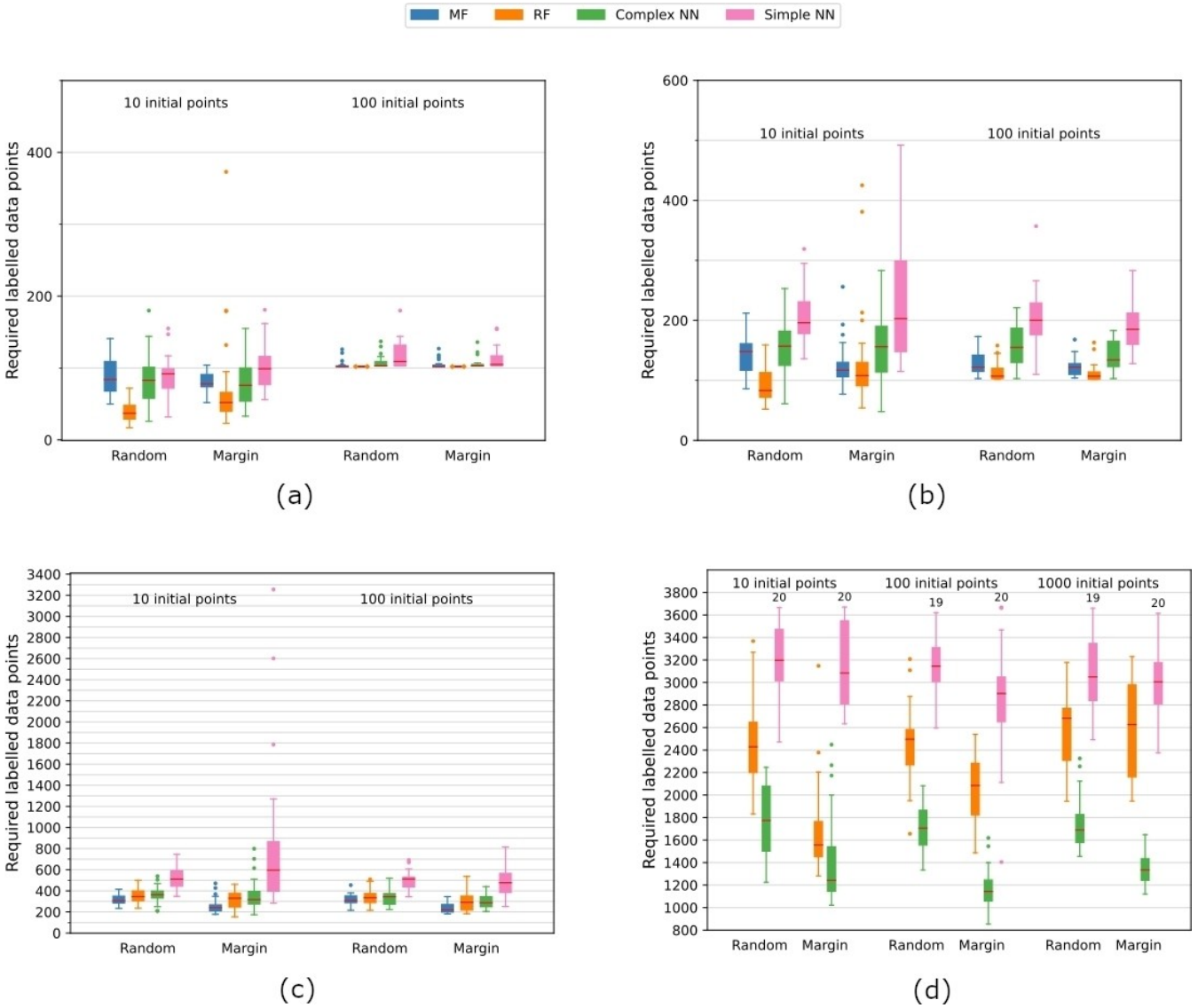


**Figure 6.** Boxplots showing the number of required labelled data points of the Suzuki reaction dataset to achieve an AUROC of (a) 0.800, (b) 0.850, (c) 0.900 and (d) 0.950. When using an initial size of 1000, all models reached an AUROC of 0.800, 0.850 and 0.900 by using only the initial labels and, therefore, these setting are not displayed. No setting reached an AUROC of 0.975.

obtain a better AUROC, in particular 0.900 or higher, we only see positive differences and the differences increases as the target AUROC is increased. Hence, we see that Margin requires fewer number of labels, compared to random sampling, as we increase the target AUROC.

Figure 5 displays output margins of the queried points of all models averaged over all 25 runs using 10 initial points and Margin as acquisition function. For matrix factorization we see no significant difference between starting with either 10, 100 or 1000 points. For the random forest model, simple neural network and complex neural network, there is a substantial difference when starting with 1000 points compared to starting with either 10 or 100 points. For all number of starting points, the output margins are around 0 in the beginning and then increases as the number of labels increases, until they converge to 1. This means starting with 1000 points gives lower output margin for the same number of labels, compared to using 10 or 100 initial points. Moreover, we found that the complex neural network needs the fewest number of labels for the output margins to converge to 1, in particular when using 10 or 100 initial points.

## Suzuki Reaction Data

Figures 6(a)–(d) display boxplots showing the required number of labelled data points to obtain an AUROC of

  

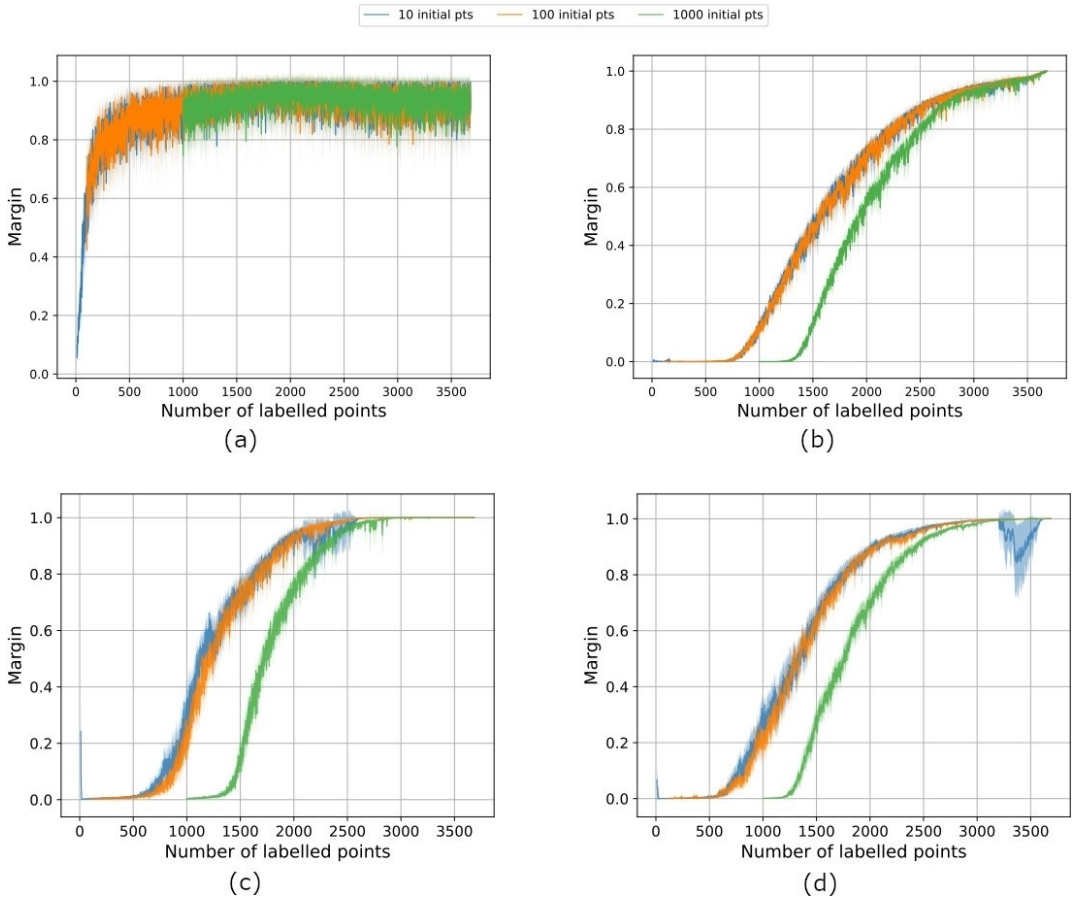


**Figure 7.** Output margins of queried data points, averaged over the 25 runs, for (a) matrix factorization, (b) random forest, (c) complex neural network and (d) simple neural network when starting with 10 labels and using Margin as acquisition function on the Suzuki reaction data. Displays the 95% approximate confidence intervals of the averages over the 25 runs.

0.800, 0.850, 0.900 and 0.950, respectively, for the different settings on the Suzuki reaction data. These boxplots were extracted from the computed AUROCs of each active learning cycle, which are displayed in Figure S2. Note that, the labelled data points include the initial points (labels). None of the settings obtained an AUROC of 0.975 and, therefore, this target AUROC is omitted for the Suzuki reaction data. For settings where not all the 25 runs reached the pre-defined level of AUROC, the numbers above the boxplots denote the number of runs that successfully reached the pre-defined level. The settings starting with 1000 labelled data points reached the three lowest levels of pre-defined AUROC by only using the initial data points and, therefore, these are omitted in Figures 6(a)–(c).

As seen in Figure 6(d), the complex neural network, simple neural network and random forest are able to obtain an AUROC of 0.950. However, none of the runs of matrix factorization reaches this AUROC. When we want to achieve an AUROC of 0.950, there is a substantial difference between the complex neural network and random forest; in particular when starting with either 10 or 100 labelled data points. Also, it should be noted that matrix factorization is able to achieve the highest possible pre-defined level of AUROC for the Buchwald-Hartwig reaction data, while random forest is not.

Figures 6(a)–(d) show that to achieve an AUROC of either 0.800 or 0.850, random forest using random sampling consistently requires the smallest amount of (labelled) data. For an AUROC of 0.900, matrix factorization utilizing Margin seems to be the best choice; while the complex neural network using Margin is the best choice when we want to obtain an AUROC of 0.950.

Table 2 shows the differences between the average number of queried labels needed to reach the different target AUROCs between random sampling and Margin on the Suzuki reaction data. The averages are over all 25 runs of each setting and the 95% approximate confidence intervals of the differences are displayed. A positive difference means that, on average, Margin needed fewer queried labels to obtain a specific target AUROC, while a negative difference means that random sampling needed to query fewer labels. Note that, the number of queried labels does not include the initial points. No model reached an AUROC of 0.975 and, therefore, this target is omitted from the table.

  

**Table 1.** Differences of the average number of queried labels required to reach the target AUROC using random sampling and Margin on the Buchwald-Hartwig dataset. The averages are over all 25 runs of the corresponding settings and the 95% approximate confidence intervals of the differences are displayed. For targets not reached, cells are left blank.

| Target AUROC | 10 initial pts | | | | 100 initial pts | | | | 1000 initial pts | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MF | RF | Complex NN | Simple NN | MF | RF | Complex NN | Simple NN | MF | RF | Complex NN | Simple NN |
| 0.800 | −7.5 ± 9.9 | 7.8 ± 20.8 | 3.8 ± 6.8 | 7.0 ± 8.7 | −1.4 ± 2.0 | 2.0 ± 9.2 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.850 | 7.7 ± 11.7 | 27.4 ± 31.4 | 17.0 ± 10.1 | 20.9 ± 11.9 | 3.1 ± 6.9 | 22.7 ± 21.2 | 2.4 ± 3.7 | 0.3 ± 2.7 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.900 | 38.1 ± 20.0 | 109.8 ± 52.8 | 35.6 ± 20.0 | 56.7 ± 18.4 | 32.8 ± 21.0 | 134.9 ± 38.0 | 17.9 ± 15.2 | 27.0 ± 17.3 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.950 | 181.8 ± 48.4 | 827.0 ± 119.2 | 249.6 ± 47.6 | 434.6 ± 85.3 | 211.0 ± 47.3 | 801.7 ± 152.6 | 252.4 ± 47.7 | 452.6 ± 68.4 | 0.0 | 461.3 ± 111.3 | 0.0 | 0.0 |
| 0.975 | 1123.4 ± 156.0 | | 852.6 ± 99.9 | 1566.6 ± 203.6 | 991.2 ± 148.7 | | 762.8 ± 74.4 | 1553.5 ± 170.5 | 548.6 ± 666.5 | | 244.8 ± 102.3 | 971.4 ± 197.2 |

We see that for almost all settings there is no significant difference between random sampling and Margin when the target is to obtain an AUROC of either 0.800, 0.850 or 0.900 with the exception that matrix factorization requires a smaller number of (labelled) data points to achieve an AUROC of 0.900. For a target AUROC of 0.950, there are positive differences when using either random forest or complex neural network with 10 or 100 initial points. For 1000 initial points, we only see a positive difference for the complex neural network when wanting to reach this target AUROC.

Figure 7 displays outputs margins of the queried point in each iteration of the active learning cycle averaged over all 25 runs on the Suzuki dataset. We see similar trends as in Figure 5 for the Buchwald-Hartwig dataset. However, the simple and complex neural network on the Suzuki data show a slower convergence to an output margin of 1 compared to the output margins on the Buchwald-Hartwig data. Conversely, compared to the results on the Buchwald-Hartwig data, random forest shows larger output margins for the same number of labelled points.

## Discussions

The results suggest a relation between how well the machine learning models had learned the observed reaction data and how well the Margin acquisition function performed. As indicated on both datasets, the more accurate we require the models to be, the better performance is observed when querying with Margin. This is consistent with earlier studies[26] showing good results using active learning after a certain number of labelled data points. Moreover, there is no substantial difference in performance when starting with 10 or 100 labelled points, which indicates that Margin is robust to potential bias in the starting data. A dataset of 10 or 100 initial points is relatively small compared to starting with 1000 data points. However, it is not advantageous to utilize Margin with 1000 initial data points since this combination consistently increases the required number of labelled data points to achieve a specific AUROC. Hence, there seems to be no evident advantage to use a large initial size, which is consistent with the fact that utilizing Margin is more advantageous when we want to obtain a greater AUROC.

Comparing the two different datasets, model development on the Buchwald-Hartwig requires a smaller number of labelled data points to reach an accuracy. This indicates that for the Suzuki dataset is it more difficult to build an accurate reaction yield prediction model using the studied machine learning algorithms. Active learning can help to increase the accuracy with less labelled data, even when the dataset is more difficult to learn. However, it is also apparent that the gain of active learning is more evident for a dataset that is easier to learn.

**Table 2.** Differences of the average number of queried labels required to reach the target AUROC using random sampling and Margin on the Suzuki dataset. The averages are over all 25 runs of the corresponding settings and the 95% approximate confidence intervals of the differences are displayed. For targets not reached, cells are left blank.

| Target AUROC | 10 initial pts | | | | 100 initial pts | | | | 1000 initial pts | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MF | RF | Complex NN | Simple NN | MF | RF | Complex NN | Simple NN | MF | RF | Complex NN | Simple NN |
| 0.800 | 8.8 ± 11.5 | −34.3 ± 30.1 | 4.2 ± 19.0 | −13.4 ± 16.8 | −0.6 ± 3.4 | 0.0 | 1.9 ± 4.7 | 5.0 ± 9.6 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.850 | 18.8 ± 19.7 | −40.0 ± 37.3 | 1.8 ± 27.5 | −23.6 ± 45.0 | 6.5 ± 10.2 | 2.4 ± 8.8 | 16.8 ± 17.6 | 13.7 ± 26.5 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.900 | 58.6 ± 34.8 | 40.0 ± 43.5 | −4.2 ± 68.7 | −317.6 ± 285.2 | 76.9 ± 28.2 | 49 ± 44.9 | 38.1 ± 40.5 | 21.8 ± 59.1 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.950 | | 803.6 ± 2245.0 | 384.4 ± 202.4 | 2.8 ± 222.1 | | 400.2 ± 186.4 | 547.5 ± 113.6 | 300.5 ± 264.2 | | −7.8 ± 206.9 | 395.2 ± 107.3 | 49.4 ± 209.8 |

### On Hyperparameter Tuning, Generalizability and Choice of Features

The focus of this study is on elucidating the impact of query strategy on the convergence of ML training. Consequently, we explored this over a wide range of ML methods using identical hyperparameters for the different query strategies to not confound results with performance differences depending on distinct hyperparameter choices.

Furthermore, since the HTE design typically spans a combinatorial space, it contains inherent correlations when varying one dimension if the other dimensions are fixed. These correlations are captured in one-hot encoding, even without providing chemical information, which greatly reduces the number of parameters that we need to provide. This is faster than structural featurization but is not optimal for out-of-box predictions as new data with unseen conditions is not fully supported in all dimensions by previous data. On some results where the most similar reaction conditions yield consistent results, it might be possible on a case-by-case basis to generate useful heuristics, but a general reactivity model based on one-hot encoding should not be attempted and is not the goal of this work.

### Feature Importance Analysis

Here we investigate the underlying reasons of why there is a difference in both model performance and impact of learning strategy between the datasets. We do this by analyzing the feature importance across both datasets. We have computed the permutation importance of each feature (reaction condition) for the random forest and matrix factorization model, and the Integrated Gradient for the complex and simple neural network. For the random forest and matrix factorization models, the permutation importance was analysed for all number of labelled data points when using an initial size of 10 labels and utilizing Margin to query labels. For the matrix factorization model, the permutation importance averaged over all 25 runs on both the Buchwald-Hartwig and Suzuki reaction data are shown in Figures 8 (a) and 9 (a), respectively. The corresponding values for the random forest model are shown in Figures 8 (b) and 9 (b), respectively. By inspecting the computed label frequencies of both datasets in Table S1, this analysis shows that a high frequency was directly related to a lower importance to the model predictions in the Buchwald-Hartwig reactions, and we see similar trends for the Suzuki counterparts. However, our analysis was not able to explain this relationship.

To do the same analysis for the complex and simple neural network, we computed the Integrated Gradients (IGs),[39] with the zero vector as baseline, for each reaction of the Buchwald-Hartwig and Suzuki datasets. Similar to the random forest and matrix factorization case, this was done

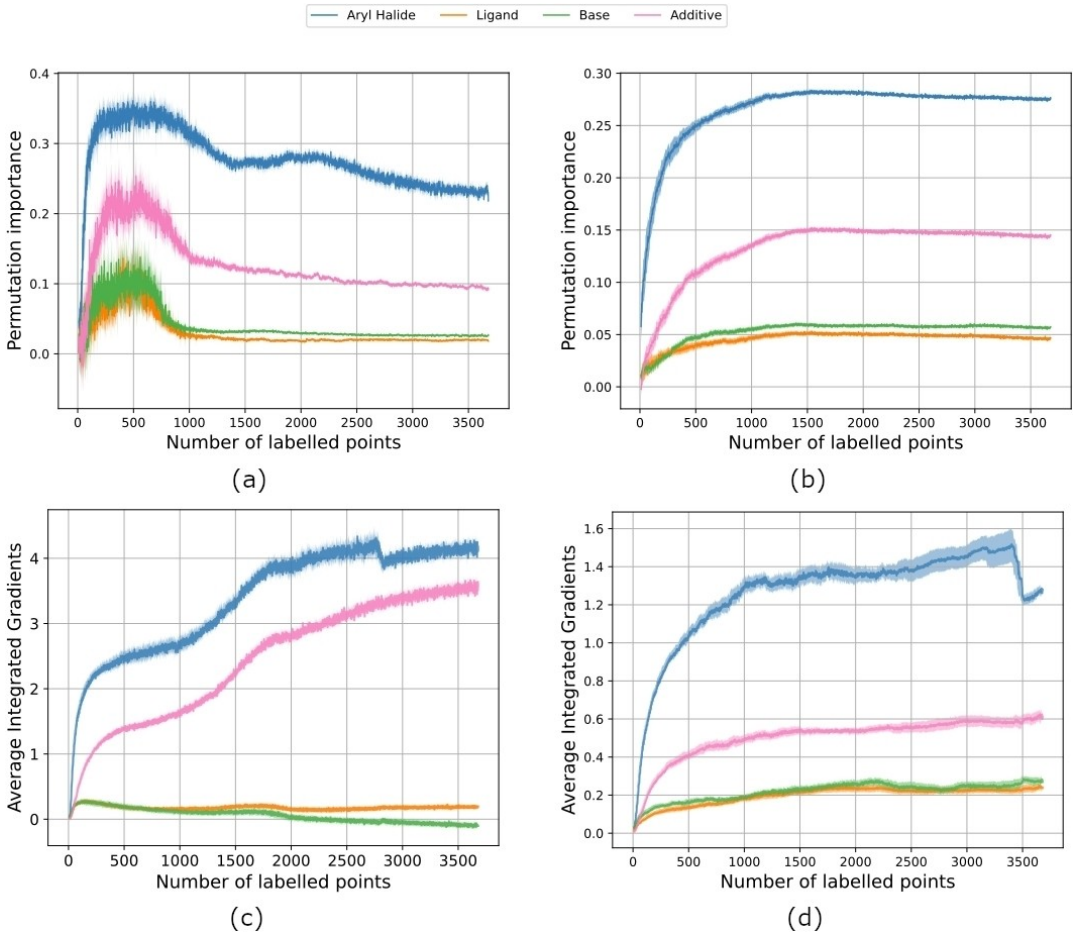


**Figure 8.** Permutation importance of (a) matrix factorization and (b) random forest, and Average Integrated Gradients of (c) complex and (d) simple neural network on the Buchwald-Hartwig reaction data. These were computed for the setting of 10 initial labels and using Margin acquisition function. All show the average over all 25 runs and the corresponding 95 % approximate confidence interval.

for all number of labelled data points for the setting using an initial size of 10 labels and utilizing Margin to query labels. The IGs for each dataset were averaged over all reactions to obtain a single value for each feature, which we call Average Integrated Gradients. This means that for each run we obtain an Average IGs value. For the complex neural network, the Average IGs averaged over all 25 runs on the Buchwald-Hartwig and Suzuki data are shown in Figures 8 (c) and 9 (c), respectively. The corresponding values for the simple neural network are shown in Figures 8 (d) and 9 (d), respectively. This provides a way to compare the relative importance of the features. For the Buchwald-Hartwig data, we see that the aryl halide and additive are the most important features which is consistent with the permutation importance obtained by the random forest and matrix factorization model. For the Suzuki data, reactant 1 and Ligand seem to be the most important features for the complex neural network when inspecting the Average IGs, while the reactant 1 and reactant 2 seem to be the most important features for the simple neural network. Hence, the order of feature importance of the simple neural network is consistent with ones of the random forest and matrix factorization model, while the complex neural network displays a different order on the Suzuki data. Moreover, compared to the Buchwald-Hartwig data, the absolute importance is not as substantial for the most important features.

This allows us to conclude that for all models, two features in the Buchwald-Hartwig dataset accounted for most of the prediction outcome with reasonably high AUROC, which is a good indication that the dataset was easy to learn for all models. On the other hand, for the Suzuki more features seem to play an important role in the prediction outcome and no features are as important as for the Buchwald-Hartwig data. Since each feature individually plays a larger role and a higher dimensional space of knowledge is needed to make the decision, this indicates that the Suzuki reaction dataset is more complex to learn using the observed models.

  

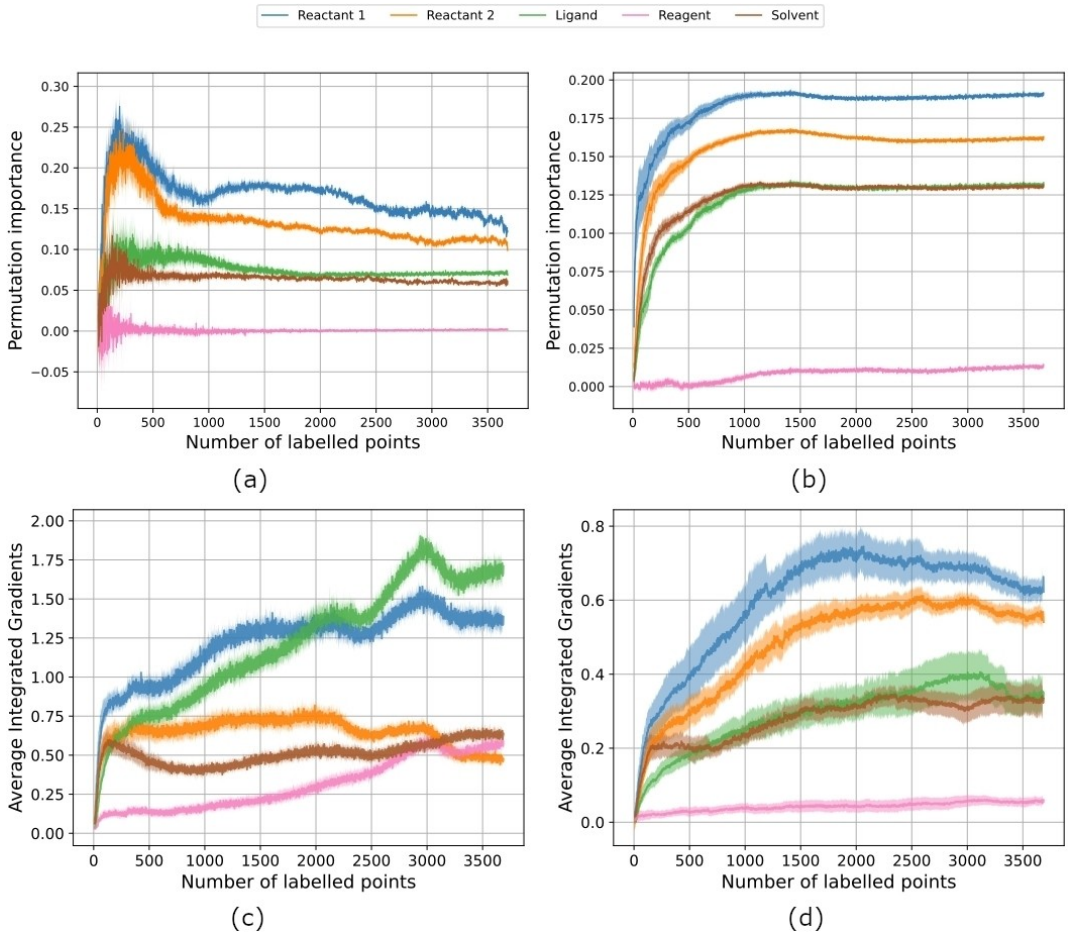


**Figure 9.** Permutation importance of (a) matrix factorization and (b) random forest, and Average Integrated Gradients of (c) complex and (d) simple neural network on the Suzuki reaction data. These were computed for the setting of 10 initial labels and using Margin acquisition function. All show the average over all 25 runs and the corresponding 95% approximate confidence interval.

It is of note that our feature importance does not match that of Chuang & Keiser, performed on the Buchwald-Hartwig reaction dataset.[40] We attribute this to our setup of binary classification with a_specific threshold, compared to minimizing the RMSE of regression. That is, as the variations of the binding affinity caused by a feature (e.g., additive) could all be within the same binary label, a feature can have a high importance with respect to the regression while not having an effect on the classification.

### Applicability of Results

We want to emphasize that the applicability of the study is meant for combinatorial library design, which is used to limit the number of different combinations of conditions one need to acquire for one study. Here, our results show that classification models reach a high accuracy already at a couple of hundred data points (~10% of the total data) with low marginal gain for conducting the rest of the experiments. Furthermore, in the best-case scenario, the Margin acquisition function reduced the number of needed data points by several hundreds to up to a thousand, considerably reducing the experimental budget to reach an 'acceptable' model.

Besides showing that there might be a relation between the models learning the observed space and how well that Margin performed, the results of this study imply that there is no drawback to implementing an active learning strategy starting at a small number of labelled data points. We observed that the Margin acquisition function always performed at least as well compared to a random scheme. As both learning schemes operate within the same space of data, fully trained models should converge towards the same AUROC, at which point active learning becomes irrelevant. The benchmarking used in this study focused on minimizing the experimental cost by looking at which point a desired AUROC was met and found an observable advantage of active learning. It is important to note however, that these experiments were computed with a

  

query consisting of a batch of one data point. The experimental designs of full batches of HTE experiments, as used in the Buchwald-Hartwig dataset, are typically of sizes 96, 384 or 1536. Thus, the results of this study might not be directly translatable to batch processes. However, if analysis time can be reduced to the same time-scale as the conducted experiments, the results could be of interest in flow-process design as one experiment is added at a time. While HTE setups in flow processes are less common they do exist, such as in the generation of the Suzuki dataset.

## Conclusions

We have explored active learning on two different high-throughput experimentation datasets. The aim was to investigate if active learning can be used to efficiently learn to predict reaction yields with a certain AUROC. We focused on comparing the uncertainty-based active learning strategy Margin to random sampling, and investigated matrix factorization, random forest, simple and complex neural networks. Moreover, in our study we explored initial sizes of 10, 100 and 1000 data points.

We have observed that active learning, in particular the Margin strategy, can arrive at a model of pre-defined AUROC with a smaller number of labelled data points. In fact, the higher the model AUROC we want to achieve, the more evident is the gain in AUROC that can be obtained using active learning. This study also shows, that for binary classification of reaction activity, the models achieve an AUROC of more than 0.9 by using 10% of the experimental data. After this point, the marginal gain severely decays. Depending on how high the accuracy requirements are, this study implies that acceptable performance might be reached by just a fraction of the data. Moreover, we have observed that the reduction in number of required labelled data points with active learning differs between different datasets. This could be due to different complexity of the datasets. Our feature importance analysis suggests that the models reach a higher AUROC when only few features were important.

To conclude, using active learning to create optimal training sets to build machine learning models for reaction yield prediction is an efficient way to reduce the experimental efforts needed.

## Funding

SJ, HGS, AS and MHC are supported by the Wallenberg Artificial Intelligence, Autonomous Systems, and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation, Sweden.

## Acknowledgements

The authors want to thank Carl Dersell for scientific discussion and input. The authors further thank Samuel Espley for his efforts in widening our perspective for HTE and the challenges for integrating an active learning feedback loop into a HTE batch workflow. Also, the authors want to thank Dr. Amol Thakkar for constructive criticism of the manuscript and many scientific discussions. Finally, the authors thank the Molecular AI team at AstraZeneca for numerous discussions and feedback.

## Conflict of Interest

None declared.

## Data Availability Statement

The source code used to run the experiments is available through a GitHub repository at https://github.com/ham-pusgs/AL-for-reaction-yield-prediction.

## References

[1] S. Johansson, A. Thakkar, T. Kogej, E. Bjerrum, S. Genheden, T. Bastys, C. Kannas, A. Schliep, H. Chen, O. Engkvist, *Drug Discovery Today: Technol.* **2019**, *32–33*, 65–72.
[2] O. Engkvist, P.-O. Norrby, N. Selmi, Y.-h. Lam, Z. Peng, E. C. Sherer, W. Amberg, T. Erhard, L. A. Smyth, *Drug Discovery Today* **2018**, *23*, 1203–1218.
[3] C. W. Coley, W. H. Green, K. F. Jensen, *Acc. Chem. Res.* **2018**, *51*, 1281–1289.
[4] A. Thakkar, S. Johansson, K. Jorner, D. Buttar, J.-L. Reymond, O. Engkvist, *React. Chem. Eng.* **2021**, *6*, 27–51.
[5] C. W. Coley, N. S. Eyke, K. F. Jensen, *Angew. Chem. Int. Ed.* **2020**, *59, 22858–22893*.
[6] K. F. Jensen, C. W. Coley, N. S. Eyke, *Angew. Chem. Int. Ed.* **2020**, *59, 23414–23436*.
[7] J. Kirman, A. Johnston, D. A. Kuntz, M. Askerka, Y. Gao, P. Todorović, D. Ma, G. G. Privé, E. H. Sargent, *Matter* **2020**, *2*, 938–947.
[8] D. Probst, P. Schwaller, J.-L. Reymond, *Digital Discovery* **2022**, *1*, 91–97.
[9] P. Schwaller, A. C. Vaucher, T. Laino, J.-L. Reymond, *Mach. Learn.: Sci. Technol.* **2021**, *2*, 015016.
[10] A. Sato, T. Miyao, K. Funatsu, *Mol. Inf.* **2022**, *41*, 2100156.
[11] A. Thakkar, T. Kogej, J.-L. Reymond, O. Engkvist, E. J. Bjerrum, *Chem. Sci.* **2020**, *11*, 154–168.
[12] E. S. Isbrandt, R. J. Sullivan, S. G. Newman, *Angew. Chem. Int. Ed.* **2019**, *58*, 7180–7191; *Angew. Chem.* **2019**, *131*, 7254–7267.
[13] B. Mahjour, Y. Shen, T. Cernak, *Acc. Chem. Res.* **2021**, *54*, 2337–2346.
[14] A. L. Haywood, J. Redshaw, M. W. D. Hanson-Heine, A. Taylor, A. Brown, A. M. Mason, T. Gärtner, J. D. Hirst, *J. Chem. Inf. Model.* **2021**, *62 (9)*, 2077–2092.

  

[15] S. Lin, S. Dikler, W. D. Blincoe, R. D. Ferguson, R. P. Sheridan, Z. Peng, D. V. Conway, K. Zawatzky, H. Wang, T. Cernak, I. W. Davies, D. A. DiRocco, H. Sheng, C. J. Welch, S. D. Dreher, *Science* **2018**, *361*, eaar6236 .
[16] D. T. Ahneman, J. G. Estrada, S. Lin, S. D. Dreher, A. G. Doyle, *Science* **2018**, *360*, 186–190.
[17] D. Perera, J. W. Tucker, S. Brahmbhatt, C. J. Helal, A. Chong, W. Farrell, P. Richardson, N. W. Sach, *Science* **2018**, *359*, 429 LP-434.
[18] P. M. Murray, S. N. G. Tyler, J. D. Moseley, *Org. Process Res. Dev.* **2013**, *17*, 40–46.
[19] L. G. Valiant, *Commun. ACM* **1984**, *27*, 1134–1142.
[20] D. D. Lewis, W. A. Gale, in *SIGIR'94*, Springer London, London, **1994**, pp. 3–12.
[21] B. Settles in *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Vol. 6 (Eds.: R. J. Brachman, W. W. Cohen, T. G. Dietterich), Morgan and Claypool, Williston, **2012**, pp. 1–114.
[22] M. K. K. Warmuth, G. Rätsch, M. Mathieson, J. Liao, C. Lemmen, in *Advances in Neural Information Processing Systems*, *Vol. 14* (Eds.: T. Dietterich, S. Becker, Z. Ghahramani), MIT Press, **2002**.
[23] A. Mehrjou, A. Soleymani, A. Jesson, P. Notin, Y. Gal, S. Bauer, P. Schwab, in *10th International Conference on Learning Representations*, **2021**.
[24] D. Reker, G. Schneider, *Drug Discovery Today* **2015**, *20*, 458–465.
[25] D. E. Graff, E. I. Shakhnovich, C. W. J. C. s Coley, *Chem. Sci.* **2021**, *12*, 7866–7881.
[26] N. S. Eyke, W. H. Green, K. F. Jensen, *React. Chem. Eng.* **2020**, *5*, 1963–1972.
[27] N. Yasuda, (Ed.: N. Yasuda), Wiley-VCH Verlag GmbH & Co. KGaA, **2010**, pp. I–XV.
[28] T. V. Aa, I. Chakroun, T. J. Ashby, J. Simm, A. Arany, Y. Moreau, T. L. Van, J. F. G. Dzib, J. Wegner, V. Chupakhin, H. Ceulemans, R. Wuyts, W. Verachtert, **2019**, arXiv preprint arXiv:1904.02514v3 [cs.LG].
[29] T. K. Ho, in *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*, IEEE Computer Society, USA, **1995**, pp. 278–278.
[30] L. Breiman, *Machine Learning* **2001**, *45*, 5–32.
[31] 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, É. Duchesnay, *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.
[32] J. D. Bossér, E. Sörstadius, M. H. Chehreghani, in *2021 IEEE International Conference on Big Data (Big Data)*, **2021**, pp. 5053–5062.
[33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, in *Advances in Neural Information Processing Systems*, *Vol. 32* (Eds.: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett), Curran Associates, Inc., **2019**.
[34] W. Falcon, J. Borovec, A. Wälchli, N. Eggert, J. Schock, J. Jordan, N. Skafte, Ir1dXd, V. Bereznyuk, E. Harris, T. Murrell, P. Yu, S. Præsius, T. Addair, J. Zhong, D. Lipin, S. Uchida, S. Bapat, H. Schröter, B. Dayma, A. Karnachev, A. Kulkarni, S. Komatsu, Martin.B, J.-B. Schiratti, H. Mary, D. Byrne, C. Eyzaguirre, cinjon, A. Bakhtin, *Zenodo* **2020**.
[35] A. L. Maas, A. Y. Hannun, A. Y. Ng, in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, **2013**.
[36] I. Loshchilov, F. Hutter, in *7th International Conference on Learning Representations*, **2019**.
[37] J. Simm, A. Arany, P. Zakeri, T. Haber, J. K. Wegner, V. Chupakhin, H. Ceulemans, Y. Moreau, in *2017 IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP)*, **2017**, pp. 1–6.
[38] C. Körner, S. Wrobel, in *ECML*, **2006**, pp. 687–694.
[39] M. Sundararajan, A. Taly, Q. Yan, in *Proceedings of the 34th International Conference on Machine Learning, Vol. 70* (Eds.: P. Doina, T. Yee Whye), PMLR, Proceedings of Machine Learning Research, **2017**, pp. 3319–3328.
[40] K. V. Chuang, M. J. Keiser, *Science* **2018**, *362*, eaat8603.

Received: February 21, 2022
Accepted: June 22, 2022
Published online on July 14, 2022

  