Yoked Learning in Molecular Data Science

Zhixiong Li[a], Yan Xiang[a], Yujing Wen[a], Daniel Reker[a,*]

[a] *Department of Biomedical Engineering, Duke University, Durham, North Carolina 27705, United States*

AUTHOR INFORMATION

* Corresponding Author

E-mail address: daniel.reker@duke.edu

Abstract

Active machine learning is an established and increasingly popular experimental design technique where the machine learning model can request additional data to improve the model's predictive performance. It is generally assumed that this data is optimal for the machine learning model since it relies on the model's predictions or model architecture and therefore cannot be transferred to other models. Inspired by research in pedagogy, we here introduce the concept of yoked machine learning where a second machine learning model learns from the data selected by another model. We found that in 48% of the benchmarked combinations, yoked learning performed similar or better than active learning. We analyze distinct cases in which yoked learning can improve active learning performance. In particular, we prototype Yoked Deep Learning (YoDeL) where a classic machine learning model provides data to a deep neural network, thereby mitigating challenges of active deep learning such as slow refitting time per learning iteration and poor performance on small datasets. In summary, we expect the new concept of yoked (deep) learning to provide a competitive option to boost the performance of active learning and benefit from distinct capabilities of multiple machine learning models during data acquisition, training, and deployment.

Keywords: machine learning, active learning, cheminformatics, drug development

I. Introduction

Active machine learning enables a machine learning (ML) algorithm to request additional data, thereby putting the ML model itself into the driver's seat of experimental design to focus data acquisition on the most useful data for the model [1]. This adaptive data selection can rapidly boost predictive performance and expand the application domain of a ML model. Active learning (AL) has shown great promise in various applications such as image classification [2], speech recognition [3], and molecular data science to hasten drug discovery and development [1, 4]. The improved performance in these studies has been attributed to the ability of AL to specifically employ the ML model architecture to enable the selection of the most useful data for the model at hand [5]. For example, by specifically requesting labels for data with the highest predictive uncertainty, AL focuses resources on data least understood by the model. The definition of a selection function can include predictive uncertainty or expected changes to the model architecture [1]. In either case, the model is actively consulted for the experimental design and thereby the implicit assumption is that access to the specific model architecture is necessary for optimal active learning performance. It could be reasonable to assume that replacing the model's uncertainty with another ML model's uncertainty would lead to a suboptimal performance given the different perspectives of the two models imposed by their respective predictive architectures. To the best of our knowledge, this hypothesis has never been directly tested in ML research.

Meanwhile, pedagogical research of AL in the classroom setting has established the concept of "yoked learning" where a student passively observes the AL process of another student [6]. Such an experimental setup enables education researchers to quantify how much a student benefits from selecting data themselves rather than simply observing data that was deemed valuable by another student. Such experimental protocols have been used as early as 1962 [7] and

continue to inform learning research by quantifying the benefits or challenges of AL compared to learning from observing others [8]. Some of the key studies in this research area have indicated yoked learning provides benefits to the random selection of data but often fails to achieve the same performance as active learning. This suggests that the active selection of novel data by the students themselves is key to the increased efficiency of AL and that the learning process is subjective since the information acquired by one student is unlikely to benefit another student to the same extent.

In the present study, we introduced the concept of "yoked (machine) learning" (YoL). In YoL, a "teacher" model is used to guide data acquisition, while another "student" model is trained using the training set provided by the teacher model (Fig. 1). We systematically investigated the YoL approach by pairing three ML models using three molecular descriptors on 14 benchmarking dataset and compared them with AL and passive learning (PL, random selection). Our study finds that AL outperforms YoL in approximately half of the investigated tasks, indicating the data selected by a model can indeed be most beneficial to the model itself and is less informative for other models. A few outlier cases highlight particularly strong benefits of active learning. However, in the other half of the cases, YoL performed similarly or better than AL, indicating that in certain tasks a surrogate model might provide similar or better performance compared to AL. We investigate these cases and propose future applications of yoked learning. Motivated by our results when pairing classical ML algorithms, we introduce the concept of yoked deep learning (YoDeL) where data selected by a classic ML algorithm is provided to a deep neural network model. Compared to active deep learning (AdeL) where the deep neural network itself is responsible for selecting data, YoDel dramatically accelerates learning by circumventing re-training of complex deep neural network architectures while showing competitive performance.

Taken together, our study and novel experimental setup provides the first direct quantification of the benefits of AL, which enables a more rigorous understanding of active ML performance and will facilitate its future deployment. Furthermore, the yoked learning approach not only enables the investigation of AL but constitutes a novel adaptive ML approach that warrants further study and holds a significant potential to boost AL performance, accelerate learning campaigns, and enable the integration of ML models that are currently not amenable to AL such as certain types of deep neural networks.

II. Materials and Methods

Datasets and descriptors

14 single-task, binary classification datasets were selected from the Therapeutics Data Commons (TDC) [9] and MoleculeNet [10] (Table 1). Data was downloaded as SMILES structures and the molecules were subsequently encoded as Morgan Fingerprints (1024 bits, radius of 2) or MACCS Key Fingerprints using RDKit [11], or with standardized RDKit Descriptors using the DeepChem "featurize" function (version 2.5.0) [12].

Machine learning models

We examined three classical ML models implemented in scikit-learn (version 1.0.2) [13] with default parameters: Random Forest (RF), Logistic Regression (LR), and Naive Bayes classifier for multivariate Bernoulli models (NB). In addition, we test the performance of a deep multilayer perceptron (MLP) based on the implementation in Chemprop [14]. For the MLP we used the default set of parameters (2 hidden layers, 300 nodes per layer, 0% dropout probability, 50 epochs, ReLU activation function [15], cross entropy loss function) since an optimized parameter set (Table S1) did not show improved performance.

Active and yoked learning

For classical ML models, the data was split 50:50 into a pool set and a test set using scaffold-based grouping implemented in TDC [9]. The training set was initialized with two randomly selected data points from the pool set, one "positive" and one "negative". New data is then iteratively selected from the pool set and added to the training set following random selection (passive learning, PL), selecting the datapoint with the most uncertain prediction by the current

model (AL) or selecting the datapoint with the most uncertain prediction by the surrogate "teacher" model (YoL). At every iteration, we calculate the performance of the model on the test set by calculating the Matthews Correlation Coefficient (MCC). The overall learning performance was determined by the area under the learning curve (AULC), i.e., the numerical integral of the AULC as calculated by the sum of the MCC values on the test set from the first to the last iteration of AL. We repeated every experiment 100 times (10 distinct train/test data splits × 10 distinct learning runs with different initial training sets per split). To compare the performance between two learning strategies, we performed a two-sample *t*-test on the 100 AULC values, and we consider two methods to perform significantly different if the *p*-value is smaller than 0.05.

Active and yoked deep learning

For MLP, the data was divided 1:1:1 into a hyperparameter optimization set, pool set and test set. The hyperparameter optimization set was used to optimized the hyperparameters through Bayesian optimization [16] implemented in Chemprop [17], and the optimized hyperparameters are listed in Table S1. The pool-test set was used for learning in the same manner as for the classical model described above. We repeated every experiment 20 times (20 data splits × 1 initial training set).

III. Results and Discussion

Active learning outperforms passive learning.

First, we surveyed the benefits of AL compared to PL on the 14 benchmark datasets using three different models and three different descriptors, totaling 126 test cases (3 models × 3 descriptors x 14 datasets). For each test case, we performed 100 repeats of AL and PL. We defined AL as beneficial if the mean AULC values of the 100 repeats of AL are significantly greater than that of PL. Overall, we found that AL provided significant benefits compared to PL in 85% (107 of 126) of cases (Fig. 2). Notably, when the Naïve Bayes model was trained using standardized RDKit descriptors, AL provided benefits in only 14% (2 out of 14) of the datasets, indicating that this model and descriptor are not a suitable combination to perform AL on our benchmarking datasets. We also observed that some of our AL runs exhibited "turning points" with maximum performance that we described before, indicating a capability of active learning to identify highly informative subsets of training data [18]. Overall, we observed a range of different behaviors per dataset, with 86% of our benchmark datasets having at least one combination of model/descriptor where AL did not provide significant benefits over PL. Conversely, for every dataset and descriptor combination, there was at least one ML model that benefited from AL compared to PL. This distinct behavior of AL for different models across different datasets and descriptors encouraged us to study the effect of YoL on these datasets and see the effect of combining different models with different performances.

Yoked learning reveals a wide range of different behaviors.

We paired all our ML models to run YoL campaigns on all our combinations of datasets and descriptors, leading to 252 different yoked learning campaigns (3 teacher models × 2 student models × 3 descriptors x 14 datasets). First, we compared YoL to PL when using the same student

model to assess whether a teacher model could provide any benefits over random sampling to the student. Using two-sample *t*-tests on the AULC values, we found that YoL performs better than PL in 69% (175 of 252) of the test cases, indicating that a surrogate teacher can provide benefits over random sampling in more than 2/3 of all here investigated benchmark cases. At the same time, the number of datasets where YoL outperformed PL is lower than the 85% beneficial rate of AL, indicating that AL slightly outperforms YoL in terms of the number of benchmarks where it outperforms PL. In terms of average AULC, both AL and YoL significantly outperformed PL ($p < 2\times10^{-27}$, two-sample *t*-test, Fig. 3A) and AL slightly outperformed YoL (average AULC of 163 *vs.* 160, $p < 3\times10^{-5}$, two-sample *t*-test, Fig. 3A). This overall indicates that, on average, AL can outperform YoL and thereby attests to the utility of the model selecting its own data. However, the strong performance of YoL compared to PL shows that data selected by a surrogate model is vastly more beneficial for a model compared to random selection, opening a new avenue for yoked learning research.

To better understand in which cases YoL would perform well, we analyzed the performance on the level of individual benchmarking datasets. We found that poor YoL performance was correlated with poor performance of the teacher model in AL, indicating that teachers that are unable to select informative data for themselves are also unable to provide informative data to a different student model ($p = 1.6 * \times 10^{-12}$, Table 3, Fisher's exact test 1). Another indicator of poor YoL performance is an AL teacher performing worse than the PL student ($p = 2.5\times 10^{-10}$, Table 3, Fisher's exact test 2), indicating that the student is more effective at learning the patterns in the dataset compared to the teacher even if only being provided with random subsets of the data. Similarly, a teacher model that is performing poorly even when training on the complete available training data (MCC < 0.1) is a good indicator that the model will be a

poorly performing teacher to any student model ($p = 3.0 \times 10^{-6}$, Table 3, Fisher's exact test 3). In summary, the teacher's performance appears to be the most important factor in determining whether YoL can succeed. Accordingly, given the trends for model performance we had observed during our AL benchmarks (Fig. 2), Naïve Bayes turned out to be least effective teacher (40/84 YoL performed better than PL) while Random Forest was a more effective teacher (71/84 cases YoL performed better than PL). Conversely, the choice of the student model did not affect YoL success ($p = 0.76$, Table 3, Fisher's exact test 4) nor was the relative performance between student and teacher relevant ($p = 0.27$, Table 3, Fisher's exact test 5). This indicates that YoL is deemed to fail when the AL teacher model does not effectively navigate the chemical space but YoL benefits are largely independent of the employed student model – indicating that a strong teacher model could boost performance of various types of students.

To specifically compare the performance of YoL and AL, we analyzed their performance on the benchmarks were both performed better than PL. We found that the benefits of AL and YoL using the same student model were highly correlated (Fig. 3B, Pearson's r = 0.83). In addition, AL and YoL performed statistically indistinguishable in 53 cases (Fig. 3B). The higher average performance for AL is driven both by a larger number of cases where AL outperforms YoL (75 cases, blue area in pie chart of Fig. 3B inset) and a small number of outlier performances where AL substantially outperformed YoL – indicating cases where a model benefits from its own selected data compared to data selected by a surrogate model. Intriguingly, the four cases where AL most strongly outperformed YoL used Naïve Bayes as a student model – meaning that Naïve Bayes is both the worst teacher and the most effective active learner in these cases (Fig 3B points labeled 1-4).

Based on this finding, we set out to investigate whether there were particularly successful pairings of teacher and student models for YoL (Fig. 3C). The most effective combination was a random forest model teaching a logistic regression model, which led to successful YoL in 86% of all test cases. Conversely, the least effective combination was a Naïve Bayes model teaching a logistic regression model, only leading to success in 45% of the test cases. Interestingly, the teacher-student relationships appeared largely symmetric, with a mean absolute difference in performance (% of benchmark cases leading to successful YoL) of only 5% when swapping student and teacher models. This could be in part tied to the inherent performance of these models on the benchmarking datasets but might also hint at relationships between models that benefit from specific distributions of training data. This could also help explain why Naïve Bayes models are not effectively taught by surrogate models (Fig. 3B).

Finally, we specifically analyzed the cases where YoL outperformed AL, which occurred in a total of 43 benchmarks (orange area in pie chart of Fig. 3B inset). We noted that strong YoL performance is often achieved for model combinations where the performance of the actively learning student and teacher differs across different stages of the learning process (Fig. 3D). For example, in multiple cases, the student was overall better at predicting the test dataset when being provided with a large amount of training data, but the teacher model was able to acquire useful data more rapidly with better performance during the early stages of learning. In such cases, YoL appears to benefit from the data acquisition of the teacher model during the beginning of the learning campaign and subsequently benefits from the more powerful ML architecture of the student to make accurate predictions with more available training data (Fig. 3D). These results hint at possible benefits of using alternative models for active sampling, opening a new avenue of active

ML research that couples multiple ML models to benefit from their distinct advantages at different learning iterations.

Yoked deep learning (YoDeL)

Deep learning models are attracting increasing attention in molecular ML. Inspired by advances in image and text processing through deep learning, the molecular data science community rapidly implemented various deep neural network architectures. Not only do some of these models show competitive performance, unique abilities such as self-learned molecular descriptors, morphing of molecules in latent spaces [19], and *de novo* design of new chemical structures through generative models expands the capabilities of the molecular data science toolbox [20]. However, a major drawback for most deep learning models is their time and resource-consuming training process [21] and their hunger for large datasets, which has made the implementation of active deep learning (ADeL) challenging since it makes re-training after adding a single data instance unfeasible and leads to poor performing models during early learning iterations. Recent studies of ADeL bypassed slow retraining through batch selection, where multiple datapoints are selected based on the same model, but it is well known that this lowers the performance of ADeL due to redundancy in the selected datapoints [22–24]. Since classical ML models can be rapidly re-trained after acquisition of a single datapoint, we tested whether we can use classical ML models as "teachers" to actively select a useful dataset that is subsequently provided to a deep neural network for training. We call this protocol yoked deep learning (YoDeL). As a proof-of-concept, we tested YoDeL performance for a MLP using Morgan fingerprint as the student model since it is considered the "vanilla" base model of deep neural networks.

We first evaluated the performance of ADeL using both default and optimized hyperparameters (Table S1) on all our benchmark datasets (Fig. S1). Compared to the default hyperparameters, the optimized hyperparameters perform better in 4 datasets (3A4, BACE, HERG, PGP), similar in 4 datasets (2C9, 3CL, DILI, HIA), and worse in 6 datasets (2D6, BBBM, Bioavailability, Carcinogens, Clintox, SKIN). This indicates that optimizing the hyperparameters of MLP does not improve their AL performance on our benchmark datasets, and the optimized hyperparameters are commonly creating deeper and wider neural networks (Table S1) which require longer training time at no apparent benefit. Therefore, we used default hyperparameters moving forward.

We next investigated the performance of active learning for the deep MLP model. Overall, ADeL outperformed PDeL in 57% (8 of 14) of the datasets. This number is considerably lower compared to the 85% (107 of 126, Fig. 2) for classic AL outperforming PL, indicating that using MLP's predictions to select data is inferior compared to classical ML models in these benchmarks – potentially due to the need for larger datasets to train complex MLP models. Motivated by this lower performance of ADeL, we analyzed the performance of YoDeL when using RF or LR as teachers to select data for the MLP model. RF yoked MLP outperformed ADeL for three datasets (3CL, BACE, Clintox) and shows no significant differences in the other 11 datasets - suggesting that YoDeL can maintain or even enhance the performance of ADeL. LR YoDeL performs worse than ADeL only in the BBBM dataset and shows no significant differences in the other 13 datasets. RF appears as the better teacher for MLP compared to LR, but both approaches seem to largely lead to competitive results instead of using the MLP directly for querying new data – indicating potential utility for replacing neural network-based uncertainty estimates for active learning through a classic surrogate model for active learning data selection.

Another important advantage of YoDeL is that it effectively circumvents slow retraining of the MLP model at every learning iteration. To quantify this benefit, we record the CPU core hours needed to run the active and yoked learning campaigns (Fig. 4B). We noticed that YoDeL reduces running time by approximately two orders of magnitude compared to AdeL. This reduction was consistent across all datasets and revealed a significant practical advantage of YoDeL compared to ADeL.

Finally, we analyzed the learning curves for the BACE and Clintox datasets to understand why YoDeL outperforms ADeL for these datasets. For the BACE dataset, ADeL outperforms YoDeL but converges early while YoDeL catches up and creates a "turning point" where a subset of the data selected by the RF model outperforms ADeL in the second half and even outperforms an MLP trained on the complete dataset (Fig 4C) [18]. Conversely, for the Clintox dataset, YoDeL gives the MLP an early boost compared to ADeL and continues to maintain higher performance across most of the learning campaign except for the last 20% (Fig 4D). These examples highlight potential different behaviors of YoDeL and how it could benefit a deep MLP model in terms of both running time and learning performance.

IV. Conclusions

We have here introduced and prototyped the concept of yoked machine learning for molecular data sciences. As an extension of the increasingly popular concept of active learning [4], it couples a second "student" model to an actively learning "teacher" model to train this "student" model with the data selected by the "teacher" model. Inspired by research in educational sciences, this experimental setup enables one to quantify benefits of active machine learning by comparing the performance of learning when a model takes an active role in selecting the data compared to when it is provided data that is deemed useful by another model. We evaluated the performance of yoked and active learning which enables us to directly quantify the contributions of active data selection in active learning performance. Specifically, we found that active learning overall performs slightly better than yoked learning and we found several cases in which active learning outperforms yoked learning significantly (Fig. 3B), showcasing the benefits of the active data selection in general and in specific use-cases. Conversely, we were also able to identify many cases where yoked learning performs competitively or even outperforms active data selection and showed that these cases are particularly driven by high performance teachers that provide high quality data to a student, indicating that data selected by a powerful model might be transferable to other machine learning models. In particular, we found that there seems to be symmetric relationships between specific types of machine learning models that indicate cross-compatibility of data selected by these models (Fig. 3C). Finally, our analysis showed that yoked learning can be particularly powerful when this strategy can benefit from the performance of both models, for example by profiting from the rapid early learning of a teacher while also benefiting from the overall better predictive performance of the student (Fig 3D). Such hybrid models could provide powerful yoked learning pipelines that benefit from the individual characteristics of each

employed model. Building on these insights, we here introduced the concept of yoked deep learning (YoDeL) where a classic machine learning model learns actively and feeds the acquired data to a deep neural network. We show that this workflow can lead to competitive learning performance (Fig. 4A) while accelerating the active data acquisition by multiple orders of magnitude (Fig. 4B), thereby circumventing challenges of active deep learning such as slow model re-fitting and poor performance on small datasets. Taken together, we believe that in the future, yoked learning will become a competitive option for active learning-based experimental workflows. By combining multiple machine learning models, this approach can benefit from the advantages of each model, thereby preventing shortcomings of the individual models. Yoked learning holds the potential to accelerate training, improve uncertainty estimations, explain predictions, and boost predictive performance for more efficient and effective active machine learning campaigns.

Availability of Data and Materials

The datasets and the code for yoked learning for classical machine learning models are available at https://github.com/RekerLab/YokedLearning. The code for yoked deep learning is available at https://github.com/Xiangyan93/ActiveLearningBenchmark.

Acknowledgements

We are grateful to the Duke Science & Technology Initiative for funding. All computations were run on the Duke Compute Cluster.

Conflicts of Interest

D. R. acts as a consultant to the pharmaceutical and biotechnology industry. Z.L., Y.X., and D.R. are co-inventors on a provisional patent application describing systems and methods for yoked machine learning.

References

1. Reker D, Schneider G (2015) Active-learning strategies in computer-assisted drug discovery. Drug Discovery Today 20:458–465. https://doi.org/10.1016/j.drudis.2014.12.004

2. Wang K, Zhang D, Li Y, et al (2017) Cost-Effective Active Learning for Deep Image Classification. IEEE Transactions on Circuits and Systems for Video Technology 27:2591–2600. https://doi.org/10.1109/TCSVT.2016.2589879

3. Nassif AB, Shahin I, Attili I, et al (2019) Speech Recognition Using Deep Neural Networks: A Systematic Review. IEEE Access 7:19143–19165. https://doi.org/10.1109/ACCESS.2019.2896880

4. Reker D (2019) Practical considerations for active machine learning in drug discovery. Drug Discovery Today: Technologies 32–33:73–79. https://doi.org/10.1016/j.ddtec.2020.06.001

5. Reker D (2020) Chapter 14:Active Learning for Drug Discovery and Automated Data Curation. In: Artificial Intelligence in Drug Discovery. pp 301–326

6. Markant D, Gureckis T (2010) Category Learning Through Active Sampling. Proceedings of the Annual Meeting of the Cognitive Science Society 32:248–253

7. Huttenlocher J (1962) Effects of manipulation of attributes on efficiency of concept formation. Psychological Reports 10:503–509. https://doi.org/10.2466/PR0.10.2.503-509

8. Gureckis TM, Markant DB (2012) Self-Directed Learning: A Cognitive and Computational Perspective. Perspect Psychol Sci 7:464–481. https://doi.org/10.1177/1745691612454304

9. Huang K, Fu T, Gao W, et al (2021) Therapeutics Data Commons: Machine Learning Datasets and Tasks for Drug Discovery and Development. arXiv:210209548 [cs, q-bio]

10. Wu Z, Ramsundar B, N. Feinberg E, et al (2018) MoleculeNet: a benchmark for molecular machine learning. Chemical Science 9:513–530. https://doi.org/10.1039/C7SC02664A

11. RDKit: Open-source cheminformatics. https://www.rdkit.org

12. Ramsundar B, Eastman P, Walters P, Pande V (2019) Deep Learning for the Life Sciences: Applying Deep Learning to Genomics, Microscopy, Drug Discovery, and More. O'Reilly Media, Inc.

13. Pedregosa F, Varoquaux G, Gramfort A, et al (2011) Scikit-learn: Machine Learning in Python. the Journal of machine Learning research 12:2825–2830

14. (2021) Message Passing Neural Networks for Molecule Property Prediction. https://github.com/chemprop/chemprop. Accessed 30 May 2021

15. Nair V, Hinton GE (2010) Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on International Conference on Machine Learning. Omnipress, Madison, WI, USA, pp 807–814

16. Bergstra J, Yamins D, Cox D (2013) Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. In: Proceedings of the 30th International Conference on Machine Learning. PMLR, pp 115–123

17. Yang K, Swanson K, Jin W, et al (2019) Analyzing learned molecular representations for property prediction. J Chem Inf Model 59:3370–3388. https://doi.org/10.1021/acs.jcim.9b00237

18. Wen Y, Li Z, Xiang Y, Reker D (2023) Improving molecular machine learning through adaptive subsampling with active learning. Digital Discovery 2:1134–1142. https://doi.org/10.1039/D3DD00037K

19. Gilmer J, Schoenholz SS, Riley PF, et al (2017) Neural message passing for quantum chemistry. In: Proceedings of the 34th International Conference on Machine Learning - Volume 70. JMLR.org, Sydney, NSW, Australia, pp 1263–1272

20. Gómez-Bombarelli R, Wei JN, Duvenaud D, et al (2018) Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. ACS Cent Sci 4:268–276. https://doi.org/10.1021/acscentsci.7b00572

21. Shi S, Wang Q, Xu P, Chu X (2016) Benchmarking State-of-the-Art Deep Learning Software Tools. In: 2016 7th International Conference on Cloud Computing and Big Data (CCBD). pp 99–104

22. Zhang Y, Lee AA (2019) Bayesian semi-supervised learning for uncertainty-calibrated prediction of molecular properties and active learning. Chem Sci 10:8154–8163. https://doi.org/10.1039/C9SC00616H

23. Graff DE, Shakhnovich EI, Coley CW (2021) Accelerating high-throughput virtual screening through molecular pool-based active learning. Chem Sci 12:7866–7881. https://doi.org/10.1039/D0SC06805E

24. Soleimany AP, Amini A, Goldman S, et al (2021) Evidential Deep Learning for Guided Molecular Property Prediction and Discovery. ACS Cent Sci 7:1356–1367. https://doi.org/10.1021/acscentsci.1c00546

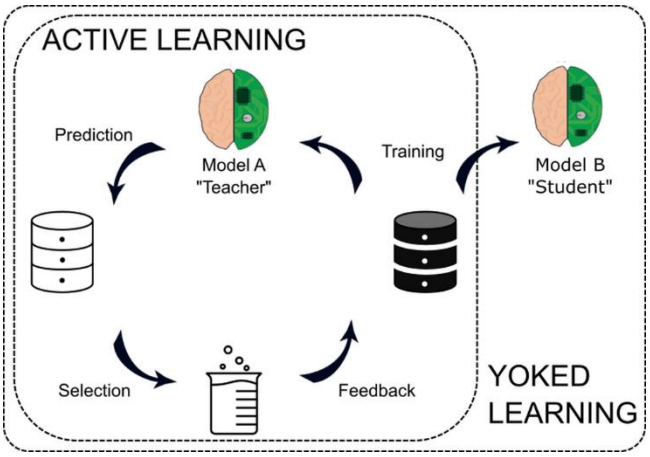

Fig.1 Schematic of yoked machine learning concept

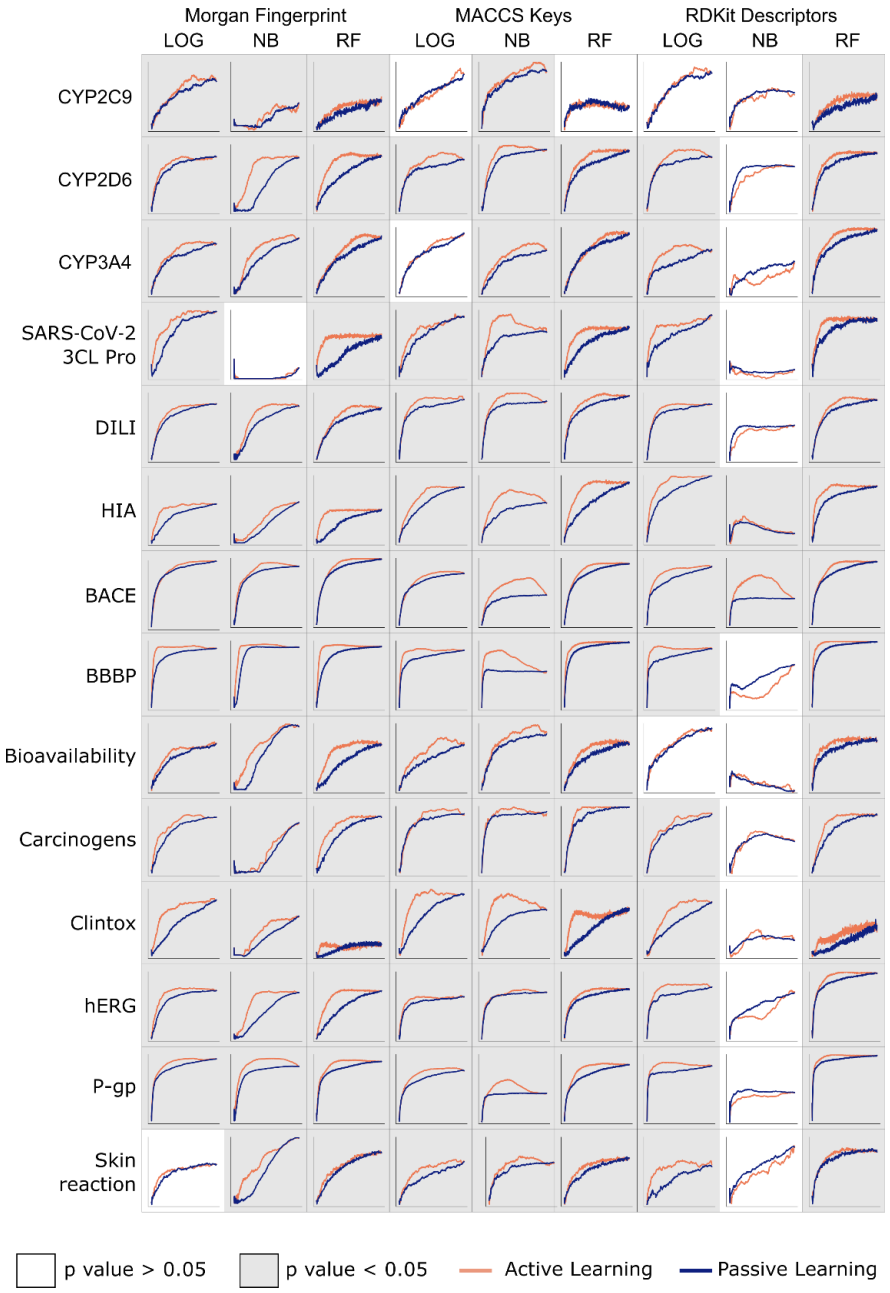

Fig. 2 Learning curves of active learning (AL, red) versus passive learning (PL, blue) on 14 datasets × 3 molecular descriptors × 3 machine learning models. Gray colored blocks indicate that active learning is significantly better than passive learning, while white blocks indicate no significant difference between active learning and passive learning or passive learning outperforming active learning.

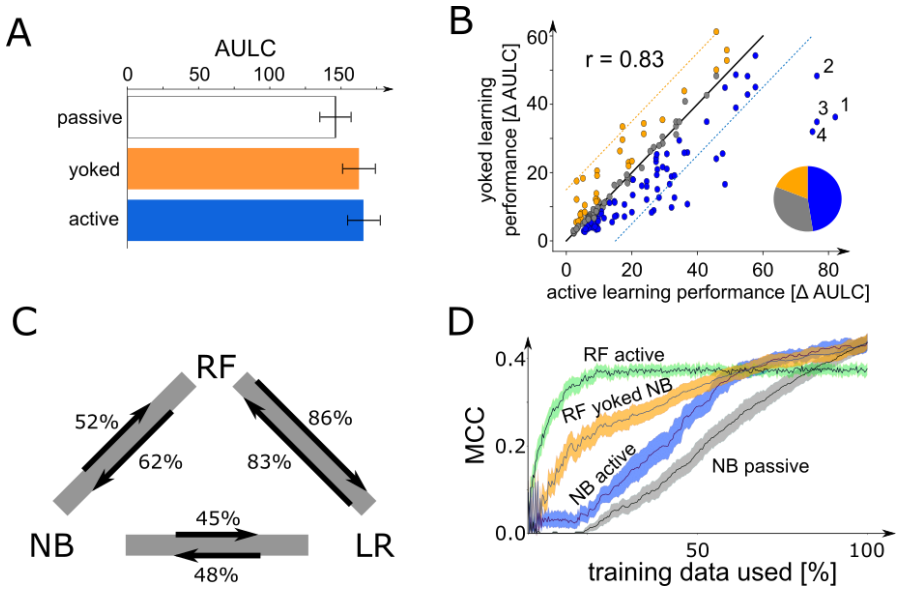

Fig. 3: Yoked Learning Results. (A) Area under learning curve of active learning (AL), yoked learning (YoL), and passive learning (PL) across all benchmarks. AL and YoL significantly outperformed PL ($p < 2\times10^{-27}$, two-sample t-test) and overall performance of AL was higher than YoL ($p < 3\times10^{-5}$, two-sample t-test). **(B)** Correlation plots for improved performance of YoL and AL on the benchmarking studies where both AL and YoL performed better than PL. Every dot corresponds to a combination of teacher model, student model, descriptor, and dataset. Delta AULC is calculated by subtracting PL AULC values from AULC achieved by YoL or AL. Dots are colored depending on whether YoL (orange) or AL (blue) perform significantly better than the other approach, with dots colored in gray if YoL and AL performance is statistically indistinguishable. The pie chart in the inset shows the relative number of benchmarks (dots) where AL and YoL perform better or are statistically indistinguishable. Points labeled as 1-4 highlight cases where AL performs much better than YoL. These are (1) RF teaching NB on BACE using RDKit descriptors, (2) LOG teaching NB on BBBP using MACCS, (3) RF teaching NB on BBBP using MACCS, (4) RF teaching NB on BACE using MACCS. **(C)** The percentage of cases where YoL outperforms PL among all benchmarks pairing a specific teacher and student model show that teacher/student relationships can be symmetric. **(D)** MCC learning curves for yoked machine

learning using circular fingerprints for the human intestinal absorption (HIA) dataset. The student model is naive bayes while the teacher model is random forest.

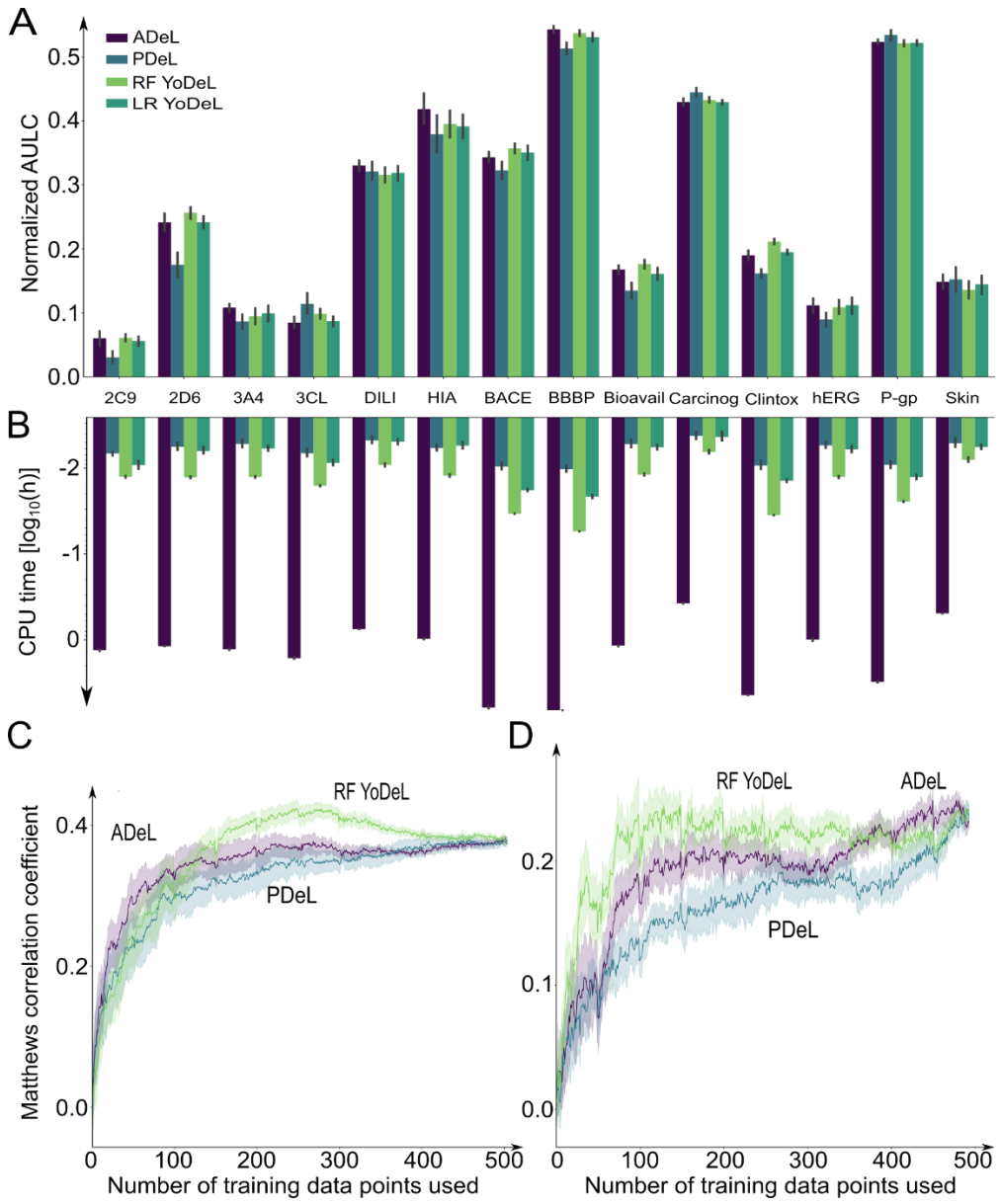

Fig. 4: Yoked Deep Learning Results. (A) Normalized AULC of ADeL, PDeL-MLP, RF YoDeL and LR YoDeL. **(B)** CPU core hours. (**C)** Learning curves of ADeL, PDeL, and RF YoDeL for the BACE dataset. **(D)** Learning curves of ADeL, PDeL, and RF YoDeL for the Clintox dataset.

Table 1 Description and number of datapoints for the datasets used in this study.

Dataset	no. of molecules	description
2C9	666	CYP2C9 Substrate
2D6	664	CYP2D6 Substrate
3A4	667	CYP3A4 Substrate
3CL	879	Activity against SARSCoV2 3CL Protease
DILI	475	Drug induced liver injury
HIA	578	Human intestinal absorption
BACE	1513	Inhibition of human β-secretase 1
BBBP	1975	Ability to penetrate the blood-brain barrier
Bioavail.	640	Oral bioavailability of drugs
Carcinogen.	278	Carcinogenic potential
Clintox	1484	Toxicity observed in clinical trials
hERG	648	Human ether-à-go-go related gene blocker
P-gp	1212	P-glycoprotein inhibition
Skin	404	Skin reaction

Table 2: Contingency tables of Fisher's exact tests used to analyze yoked learning performance.

Fisher's exact test 1 (p-value=1.6×10^{-12})

	P-S < T-S	P-S ≥ T-S
P-T < T-T	168	46
P-T ≥ T-T	7	31

Fisher's exact test 2 (p-value=2.5×10^{-10})

	P-S < T-S	P-S ≥ T-S
P-S < T-T	137	28
P-S ≥ T-T	38	49

Fisher's exact test 3 (p-value=3.0×10^{-6})

	P-S < T-S	P-S ≥ T-S
$MCC_{T,all}$ <0.1	5	17
$MCC_{T,all}$ ≥0.1	170	60

Fisher's exact test 4 (p-value=0.76)

	P-S < T-S	P-S ≥ T-S
S=LR	61	23
S=NB	57	27
S=RF	57	27

Fisher's exact test 5 (p-value=0.27)

	P-S < T-S	P-S ≥ T-S
$MCC_{S,all} < MCC_{T,all}$	92	34
$MCC_{S,all} \geq MCC_{T,all}$	83	43

T: teacher model. S: student model. P: passive. T-T: active learning using teacher model. T-S: yoked learning. P-T: passive learning, teacher model for prediction. P-S: passive learning, student model for prediction. P-S < T-S: the average AULC of P-S is significantly lower than the average AULC of T-S. S =LR/NB/RF: Student model is logistic regression / naïve Bayes / random forest. $MCC_{T/S,all}$: MCC performance on the test set by the teacher / student model trained using all training data.