# Active Meta-Learning for Predicting and Selecting Perovskite Crystallization Experiments

Venkateswaran Shekar,[1] Gareth Nicholas,[1] Mansoor Ani Najeeb,[2] Margaret Zeile,[2] Vincent Yu,[1] Xiaorong Wang,[1] Dylan Slack,[1] Zhi Li,[3] Philip W. Nega,[3] Emory M. Chan,[3] Alexander J. Norquist,[2] Joshua Schrier,[4] and Sorelle A. Friedler[1]

[1)] *Department of Computer Science, Haverford College, 370 Lancaster Avenue, Haverford, Pennsylvania, 19041, USA*

[2)] *Department of Chemistry, Haverford College, 370 Lancaster Avenue, Haverford, Pennsylvania, 19041, USA*

[3)] *The Molecular Foundry, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, California, 94720, USA*

[4)] *Department of Chemistry, Fordham University, 441 E. Fordham Road, The Bronx, New York, 10458, USA*

(*Electronic mail: sorelle@cs.haverford.edu)

(*Electronic mail: jschrier@fordham.edu)

(Dated: 26 October 2021)

Autonomous experimentation systems use algorithms and data from prior experiments to select and perform new experiments in order to meet a specified objective. In most experimental chemistry situations there is a limited set of prior historical data available, and acquiring new data may be expensive and time consuming, which places constraints on machine learning methods. Active learning methods prioritize new experiment selection by using machine learning model uncertainty and predicted outcomes. Meta-learning methods attempt to construct models that can learn quickly with a limited set of data for a new task. In this paper, we applied the model-agnostic meta-learning (MAML) model and Probabilistic LATent model for Incorporating Priors and Uncertainty in few-Shot learning (PLATIPUS) approach, which extends MAML to active learning, to the problem of halide perovskite growth by inverse temperature crystallization. Using a dataset of 1870 reactions conducted using 19 different organoammoniumn lead iodide systems, we determined the optimal strategies for incorporating historical data into active and meta-learning models. We then evaluated the best three algorithms (PLATIPUS, and active-learning k-Nearest Neighbor and Decision Tree algorithms) with four new chemical systems in experimental laboratory tests. With a fixed budget of 20 experiments, PLATIPUS makes superior predictions of reaction outcome compared to other active-learning algorithms and a random baseline.

# I. INTRODUCTION

Materials discovery can be accelerated by combining simulations, machine-learning, and automation[1,2]. Autonomous experimentation systems, in which algorithms specify an iterative sequence of new experiments based on incoming results without human intervention, have been the subject of recent reviews[3–6]. Autonomous experimentation systems have been demonstrated for a variety of materials optimizations problems, including carbon nanotube growth[7,8], additive manufacturing[9], colloidal nanoparticle syntheses[10–12], thin-film devices[13], photocatalysts synthesis and characterization[14], alloy phase mapping[15], and optimization of battery electrolyte compositions[16].

Metal halide perovskites are a promising class of materials for next-generation photovoltaic and optoelectronic devices[17]. The ability to incorporate different organic cations results in a vast, chemically diverse space to explore[18]. The relatively mild, solution-based syntheses for these materials make them amenable to high-throughput automated experimentation.[19] Some examples include, antisolvent precipitation of polycrystals[20,21], antisolvent vapor diffusion[22,23], perovskite thin films[13,24–26] and production of nanocrystals under batch[27,28] and flow[10] conditions. We previously described our Robot Accelerated Perovskite Investigation and Discovery (RAPID) system for performing high-throughput inverse temperature crystallization (ITC) growth of halide perovskites[29]. RAPID has collected data on $14,838$ reactions (and counting), spanning 56 organic cations and 3 solvents, a subset of which are used in this study. RAPID has been used to assess and demonstrate data-driven approaches to experimental tasks, including model fusion strategies for automating quality control of high-throughput data[30], and statistical analyses of uncontrolled variations in lab conditions to identify the role of humidity in reaction outcomes[31]. ML models trained on 96 randomly selected experiments within a chemical system can interpolatively predict subsequent outcomes in that system[29]. By augmenting the dataset to include molecular and solution physicochemical features, extrapolative prediction of reaction outcomes for new chemical systems (i.e., when the protonated organic amine is changed) has $\sim 40\%$ precision on average, but with large variations[32]. While better than random experiment selection ($\sim 25\%$ precision), this suggests the need for improved algorithms. As any set of descriptors may not capture all interactions specific to a particular molecular species, this suggests the need for better algorithms that can learn the specific attributes of a chemical system from a limited set of new experiments.

Experiment selection algorithms, such as active learning algorithms, are a central part of autonomous experimentation systems, and have been summarized in several recent reviews[3,21]. *Ac-*

*tive learning* (AL) methods have ML algorithms iteratively request new data-points during training. Requested data are prioritized by specifying a policy that balances exploration (reducing model uncertainty) and exploitation (requesting new points with a high value according to the existing model). This transforms the model training into a sequential learning process, in which each new experimental datum is incorporated into the model, and this improved model is used to request the next experiment. Active learning has been widely adopted in molecular simulations and the construction of ML models on computational data. Notable examples include determination of phase diagrams[33], parameterization of ML force fields[34], design of organometallic complexes[35], and computational searches for $CO_2$ electrocatalytic alloys[36]. Notable demonstrations of active learning in the laboratory setting include determining the reaction conditions for polyoxometalate crystallization[37,38], antisolvent vapor diffusion syntheses of halide perovskites[22,23], electrocatalytic alloys for oxygen evolution reactions[39], alloy phase mapping[15], neutron scattering determinations of magnetic properties[40], determination of material property curves[41], and battery electrolyte optimization[16]. Active learning is typically framed in the context of parameterizing a single model applicable to the entire problem domain. Our previous work suggests that it may be more effective to consider each chemical system as comprising a distinct problem domain with its own ML model. The naïve strategy of performing an active learning parameterization *ab initio* for each system would not make use of the valuable information contained in previous experimental data.

*Transfer learning* uses information from one problem (i.e., chemical system) to solve a different, but related problem. The premise is that the model will have already learned fundamental representations and the general structure of the task. Therefore, by starting with a model pretrained on the previous system, a smaller amount of data on the new system is needed to fine tune those previously learned characteristics. Applications in computational chemistry include parameterization of ML forcefields[34], *in silico* drug discovery[42], and efficient metadynamics sampling in protein molecular dynamics simulations[43]. Applications to chemical experimentation are more limited, but examples include Tandem mass spec proteomics (with a task transfer from unmodified to post-translationally modified proteins)[44], defect identification in silicon CMOS devices (with a task transfer between transistor gate geometries)[45], and band gap and catalytic activation energy prediction (with transfer between DFT prediction results and experimental values)[46].

*Meta-learning* is a form of transfer learning in which ML models are constructed to minimize the loss functions and are evaluated on their ability to "learn how to learn" when presented with data in a new domain (or “task”)[47]. In practice, this results in an initial model that is parameterized to describe a generic case, but more importantly, focuses data acquisition during the sequential learning phase such that it rapidly converges for the system at hand. Applications of meta-learning in chemistry have largely focused on *in silico* drug design tasks, namely determination of quantitative structure activity relationships (QSAR)[48,49], identification of potential drug-drug interactions[50], and ligand optimization[51]. Other applications of meta learning in chemistry include peptide[52] and RNA design[53], soot density recognition in combustion[54], and gas adsorption in nanoporous materials[55].

In this paper, we apply model-agnostic meta-learning (MAML)[56] to the problem of metal halide perovskite crystal growth. We consider each chemical system as a new task, and determine the viability of this approach for few-shot meta-learning suitable for laboratory experimentation. As performing experiments is costly and time-consuming, we use active learning to best iteratively improve the per-task (per-amine) models. To do this, we apply a MAML variant that allows the determination of prediction probabilities of each outcome, specifically the Probabilistic LATent model for Incorporating Priors and Uncertainty in few-Shot learning (PLATIPUS)[57]. We describe computational studies using historical data to assess the benefits of an active meta-learning approach relative to "mere" active learning approaches and develop an appropriate training and validation procedure for applying these methods to laboratory tasks. In addition to evaluating the results on time-held out data, we also performed a laboratory experimental task on previously unseen systems. Statistical analysis of the laboratory results demonstrates that the PLATIPUS active meta-learning technique is more successful in predicting new experiments that traditional active learning methods or random experiment selection.

## II. METHODS

### A. Theory: MAML and PLATIPUS

The goal of meta-learning is to train a model that can quickly adapt to new tasks using only a few datapoints and iterations. MAML formulates this problem in a model-agnostic way by adding a gradient-based learning rule (in addition to whatever other loss function is present) that prefers model parameters that are sensitive to changes in the task. By doing so, small changes in the parameters produce large improvements on the loss function of any task drawn from a distribution of possible example tasks when altered in the direction of the gradient of that loss.[56] Consider a model $f_\theta$, with parameters $\theta$. When applied to a new task, $T_i$, the model's parameters should be updated from $\theta$ to $\theta_i'$. This can be formalized by considering the update in terms of a gradient descent on task $T_i$,

$$\theta_i' = \theta - \alpha \nabla_\theta L_{T_i}(f_\theta) \tag{1}$$

where $\alpha$ is a step size, and $L$ is the user-specified loss function evaluated on task $T_i$ using model $f_\theta$. We can see from this why this is a model-agnostic approach—it is applicable to any model, $f_\theta$ for which we can compute gradients of any loss function $L$. Model parameters are trained by optimizing for the performance of $f_{\theta_i'}$ with respect to $\theta$ across tasks sampled from the distribution of possible tasks, $p(T)$, by optimizing a meta-objective,

$$\min_\theta \sum_{T_i \sim p(T)} L_{T_i}(f_{\theta - \alpha \nabla_\theta L_{T_i}(f_\theta)}) \tag{2}$$

Minimizing this meta-objective also requires a gradient, and so we note that this requires the gradient of a gradient to update $\theta_i$. In practice, this meta-optimization is also solved by stochastic gradient descent.

The MAML model can quickly adapt to a new task by training on a handful of samples from that task, but lacks the ability to provide uncertainty for predicted samples. Even with the best possible prior, MAML cannot determine whether there is enough information in the small set of samples to resolve the new task with high certainty. PLATIPUS is one such method that can propose multiple solutions to an ambiguous few-shot problem. Evaluating this uncertainty, we can perform active learning by providing the models with labels to samples with lowest uncertainty.

PLATIPUS extends MAML to model a distribution over prior model parameters.[57] The PLATIPUS algorithm initializes a distribution over model parameters $\Theta$. The distribution is generated using average model parameters $\mu_\theta$, variance of model parameters $\gamma_\theta^2$, learned diagonal covariance $v_q$ and two learning rate vectors $\gamma_p$ and $\gamma_q$. The algorithm assumes the distribution of model parameters to be a normal distribution.

$$\Theta := \{\mu_\theta, \gamma_\theta^2, v_q, \gamma_p, \gamma_q\} \tag{3}$$

The key difference between MAML and PLATIPUS is instead of a single initialization point $\theta$, the model learns the distributions over priors $q$ and $p$ which are the meta-training ($D_{T_i}^{train}$) and meta-testing ($D_{T_i}^{test}$) data of a task $T_i$. A Kullback-Leibler (KL) divergence loss-term is added to the MAML meta-objective that measures the distance between the distributions $q(\theta|D^{test}_{T_i})$ and $p(\theta|D^{train}_{T_i})$. Thus the algorithm updates $\Theta$ such that both the KL divergence and the loss of the updated models are minimized on the testing data. The meta objective in this case is:

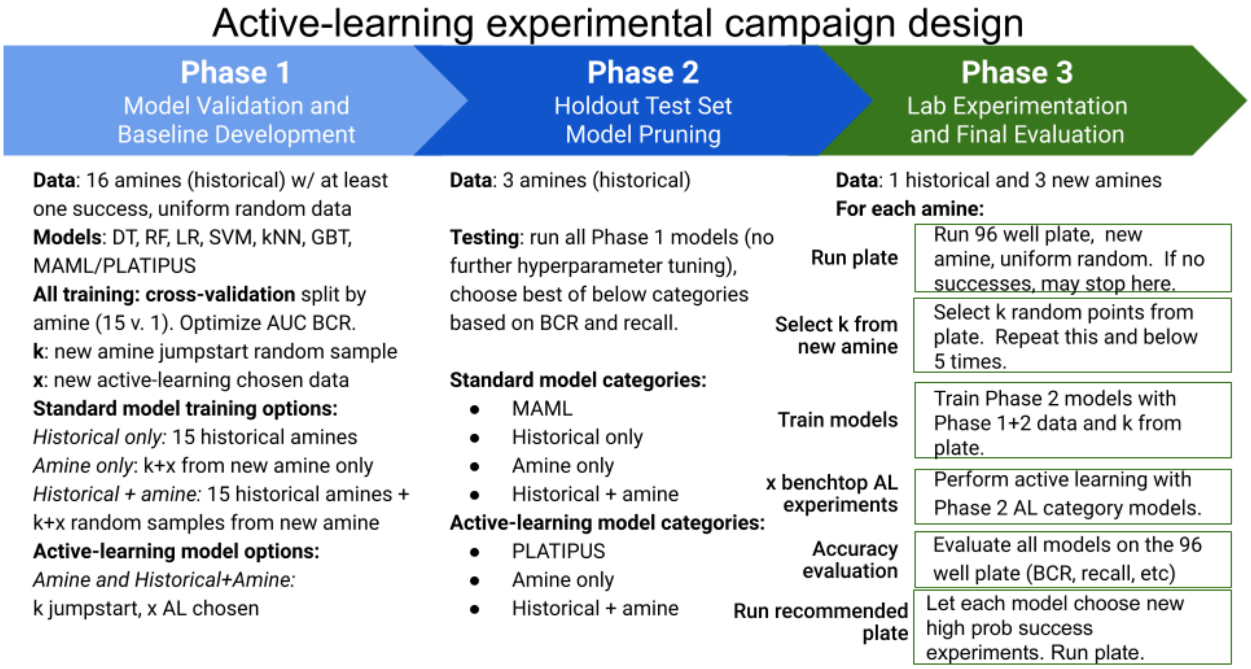
$$\min_\theta \sum_{T_i\sim(T)} L_{T_i}(f_{\theta-\alpha\nabla_\theta L_{T_i}(f_\theta)}) + D_{KL}(q(\theta|D^{test}_{T_i})||p(\theta|D^{train}_{T_i})) \tag{4}$$

In the testing phase, the algorithm samples several sets of model parameters from the distribution $\Theta$. Next, it performs gradient descent on all sets of parameters to obtain multiple task specific models. These trained models can be used to make predictions where the uncertainty is the difference in the predicted probabilities between sampled models.

In this study, both the MAML and PLATIPUS models use neural networks with three hidden layers. The hidden nodes, training rates and other hyper-parameters are presented in Table S-2 and Table S-3 in the supplementary material.

## B. Overview of the Training and Evaluation Process

Figure 1 depicts an overview of the numerical and laboratory experimental campaign designed for this study. The campaign is split into three phases: (1) validation, (2) hold-out testing and



Active-learning experimental campaign design

**Phase 1**
Model Validation and Baseline Development

**Data**: 16 amines (historical) w/ at least one success, uniform random data
**Models**: DT, RF, LR, SVM, kNN, GBT, MAML/PLATIPUS
**All training: cross-validation** split by amine (15 v. 1). Optimize AUC BCR.
**k**: new amine jumpstart random sample
**x**: new active-learning chosen data
**Standard model training options:**
*Historical only:* 15 historical amines
*Amine only*: k+x from new amine only
*Historical + amine:* 15 historical amines + k+x random samples from new amine
**Active-learning model options:**
*Amine and Historical+Amine:*
k jumpstart, x AL chosen

**Phase 2**
Holdout Test Set Model Pruning

**Data**: 3 amines (historical)

**Testing**: run all Phase 1 models (no further hyperparameter tuning), choose best of below categories based on BCR and recall.

**Standard model categories:**
- MAML
- Historical only
- Amine only
- Historical + amine

**Active-learning model categories:**
- PLATIPUS
- Amine only
- Historical + amine

**Phase 3**
Lab Experimentation and Final Evaluation

**Data**: 1 historical and 3 new amines
**For each amine:**

**Run plate**: Run 96 well plate, new amine, uniform random. If no successes, may stop here.
**Select k from new amine**: Select k random points from plate. Repeat this and below 5 times.
**Train models**: Train Phase 2 models with Phase 1+2 data and k from plate.
**x benchtop AL experiments**: Perform active learning with Phase 2 AL category models.
**Accuracy evaluation**: Evaluate all models on the 96 well plate (BCR, recall, etc)
**Run recommended plate**: Let each model choose new high prob success experiments. Run plate.

FIG. 1. Experimental Campaign Overview

(3) in-lab testing. During the validation phase, we evaluated a number of models and performed baseline development with existing data. To establish a baseline, we considered standard machine learning models: *k*-nearest neighbors (KNN), decision trees (DT), support vector machines (SVM), logistic regression (LR), gradient boosted trees (GBT), and random forest (RF) models, along with the MAML[56] and PLATIPUS[57] meta-models. Throughout, our goal was to ensure a fair comparison across models, so that models always had access to the same data. This first phase focused on training and validation on historical data from perovskite experiments across 16 amines, so as to determine the best set of hyperparameters across a wide variety of models and training strategies. In the hold-out testing phase, the top performing models—with their hyperparameters fixed—were tested on three (3) held-out amines in our dataset. The goal of this second phase was to identify the best models to advance to the final laboratory testing phase. The goal of the third phase was to evaluate the model performance, in an actual laboratory setting, on a new system with the ability to request experiments. Each model received the same initial set of random starting data, requested its own desired experiments, and then attempted to predict successful outcomes which were then validated. This process was repeated twice for each of four (4) previously unknown chemical systems (constituting a time-separated laboratory test ). Below we describe the datasets, models, the training and validation conducted in each phase, and laboratory experimentation methodology.

## C. Datasets

The data used to train, validate, and test the developed models consists of the experimental reagents, experimental conditions, derived chemical descriptors, and observed outcomes of metal halide perovskite experiments. The data was obtained using the Robot-Accelerated Perovskite Investigation and Discovery (RAPID) system discussed in our previous work.[29] Each data item describes an inverse-temperature crystallization (ITC) synthesis comprised of includes two inorganic components (lead and iodide) and an organoammonium cation (which for brevity we refer to as, the amine), reaction conditions, and outcomes. Reaction conditions include the temperature and reagent concentrations—we considered only experiments conducted at a nominal 105 °C, and only those where the concentrations were chosen uniformly over the achievable convex hull of possible compositions,[58] and for which at least one successful outcome was observed. Of the 20 amines satisfying this criteria in the historical data, 16 amines (and all experimental data using those amines) were selected randomly for cross-validation experiments, three amines were selected randomly for hold-out testing, and one amine (Dimethylammonium iodide) was held out to be used as part of the phase three laboratory test experiments. In addition, we acquired a uniformly-sampled baseline for three additional amines for which we had no previous data, in order to demonstrate the resulting models on a true time-separated hold-out set. Table I summarizes the amines included in each phase of the study, and the number of experiments from the historical dataset.

Each amine is used to separate the data into *tasks* for the developed meta-learning models. The ESCALATE software was used to append stoichiometric and physicochemical descriptors from the raw record of reaction conditions and amine structure.[32] In total, each experiment is described by 50 features: 28 molecular descriptors (number of atoms, rotatable bond counts, etc.), 7 reaction conditions (temperature, concentration, etc.), and 15 stoichiometric descriptors. The full list of included features can be found in Table S-1. Numerical features in the dataset were scaled to unit variance for training models. The mean and standard deviation of the training data were used to scale the training set, samples from the unseen amine, and the pool of potential experiments used for active learning. The 44 numerical features in the dataset were scaled using this process while the 6 remaining binary features were not.

Experimental outcomes are scored into four classes: (1) no solid observed in the solution; (2) fine powder observed; (3) small crystals observed; (4) large crystals observed ($>$ 0.1 mm). In this study, the outcomes are represented as binary values for the machine learning classification task, with large (class 4) crystals considered as successful (denoted as a classification outcome of 1) and all other classes are considered as failed experiments (classification outcome of 0)[29,59,60]. In general, the success rates of any given amine (shown in the final column of Table I) are very low; a randomly chosen experiment is more likely to fail than succeed for all but one amine. Of the 23 included amines, 7 have success rates less than 10% and another 11 have success rates between 10% and 20%. Given this large class-imbalance, models will be evaluated based on their balanced classification rate, as discussed in the next section. A machine readable copy of the dataset is available at https://github.com/darkreactions/platipus/.

TABLE I: Per-amine data statistics for all experimental phases.

| **Amine chemical name** | **Number of samples** | **Number of successes** | **Fraction of success** |
|---|---|---|---|
| Cyclohexylmethylamine | 96 | 60 | 0.62 |
| Phenylamine | 96 | 29 | 0.30 |
| t-Butylamine | 96 | 19 | 0.20 |
| 4-Fluoro-benzylamine | 96 | 18 | 0.19 |
| N,N-Dimethylpropane-1,3-diamine | 96 | 16 | 0.17 |
| Methylamine | 32 | 4 | 0.13 |
| Morpholine | 96 | 11 | 0.12 |
| 4-Fluoro-phenylamine | 71 | 8 | 0.11 |
| Cyclohexylamine | 96 | 9 | 0.10 |
| n-Hexylamine | 96 | 8 | 0.09 |
| Piperidine | 156 | 11 | 0.07 |
| Propane-1,3-diamine | 81 | 5 | 0.07 |
| N,N-Diethylpropane-1,3-diamine | 96 | 4 | 0.05 |
| N,N-Diethylethane-1,2-diamine | 96 | 1 | 0.02 |
| Ethylamine | 81 | 1 | 0.01 |
| Butane-1,4-diamine | 96 | 1 | 0.01 |
| **Hold-out testing data** | | | |
| iso-Butylamine | 35 | 10 | 0.29 |
| n-Butylamine | 42 | 5 | 0.12 |
| 4-Trifluoromethyl-phenylamine | 72 | 8 | 0.11 |
| **Time-separated hold-out live lab experimentation data** | | | |
| 4-Hydroxyphenethylamine | 96 | 45 | 0.47 |
| 4-Chlorophenethylamine | 96 | 27 | 0.28 |
| 4-Chlorophenylamine | 96 | 19 | 0.20 |
| Dimethylamine | 95 | 15 | 0.16 |

### D. Phase 1: Model Training

The training and model evaluation phase considers many possible training paradigms and differently composed training datasets. In addition to classic one-shot models, we will develop and compare to meta-models and to active learning versions of both one-shot and meta-models. The goal of these training procedures is to evaluate all models against reasonable benchmarks, making sure that the classical one-shot models have the same access to (and advantage from) provided training data as active meta-models.

Data associated with a specific amine will serve as the meta-learning task separation of the data; one-shot models will mimic this assessment by creating training and testing splits so that all experiments with a specific amine are in either training or testing, but not both. Meta-learning models are given initial *jump start* data from a new task (an “unseen" amine absent in the training data), comprised of $k = 10$ uniformly randomly sampled reaction data for the new amine. We chose $k$ to be relatively small, as it represents the initial data that needs to be collected either in an automated or non-automated experiment. To create reasonable benchmarks, one-shot models are also given access to the same initial $k = 10$ jump start reactions for the new amine. Active learning models request $x = 10$ additional samples from the new amine in an iterative fashion; again, this low value of $x$ was chosen so as to be feasible for non-automated experimentation. Non-active learned models will also have access to $x = 10$ additional experimental data points chosen uniformly at random from the new amine. During phase 1 and 2 (model validation and hold-out testing), the iterative experiment requests are chosen from the pool of archived experimental data. During phase 3, they are selected from a much larger *stateset* comprised of a grid of $\sim 20,0000$ possible compositions achievable with the stock solutions used. To summarize, each model has access to at most 20 samples from the new amine during its training process.

To determine how best to incorporate historical data, different combinations of historical training data and per-amine data were used during model training, as summarized in Table II. One-shot models were trained in the following ways: (i) *Historical only* using only historical data and no data from the unseen amine; (ii) *Amine only* using only $k + x = 20$ data sampled uniformly at random from the unseen amine; (iii) *Amine only with success* using only $k + x = 20$ data sampled uniformly from the historical data, but where this must contain at least one successful experiment; (iv) *Historical* + *amine* using all available training data, i.e., all historical data in addition to and 20 points sampled randomly from the unseen amine. Additionally, active learning models were

TABLE II. Training data sets considered in this study.

| Training strategy | Historical data | 20 randomly sampled points | | $k = 10$ uniform random $x = 10$ actively sampled | |
|---|---|---|---|---|---|
| | | with or w/o success | with at least 1 success | with or w/o success | with at least 1 success |
| Historical only | ✓ | | | | |
| Amine only | | ✓ | | | |
| Amine only w/ success | | | ✓ | | |
| Historical + Amine | ✓ | ✓ | | | |
| Amine only AL | | | | ✓ | |
| Amine only w/ success AL | | | | | ✓ |
| Historical + Amine AL | ✓ | | | ✓ | |

trained in the following ways: (v) *Amine only active learning* using amine only data as above, where $k = 10$ are given as initial training data and $x = 10$ are queried iteratively via active learning; (vi) *Historical + amine active learning* using all previously available data, where $k = 10$ data points from the new amine are added to the initial training set and $x = 10$ are queried to refine the model via active learning.

The models trained under the above options are $k$-nearest neighbor (KNN), random forest (RF), decision trees (DT), logistic regression (LR), support vector machine (SVM), and gradient boosted tree (GBT) models. MAML was trained under the historical and amine option and PLATIPUS was trained under the historical + amine active learning option as summarized Table III. In the amine only strategy, the training set contains 20 samples from the held out amine. Due to the unbalanced nature of the outcomes, all 20 random samples may be failures. The SVM, GBT and LR models require at least one sample from each class in its training data, thus the SVM, GBT, and LR models can only be examined via the “amine only with success" training option (and not the “amine only" version).

TABLE III. Models considered in this study and data used to train them

| Model | Historical | Amine | Amine only | Historical + | Amine only | Amine only | Historical + |
|---|---|---|---|---|---|---|---|
| | only | only | with success | Amine | AL | w/ success AL | Amine AL |
| KNN | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| RF | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| DT | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| LR | ✓ | | ✓ | ✓ | | ✓ | ✓ |
| SVM | ✓ | | ✓ | ✓ | | ✓ | ✓ |
| GBT | ✓ | | ✓ | ✓ | | ✓ | ✓ |
| MAML | | | | ✓ | | | |
| PLATIPUS | | | | | | | ✓ |

All models in this study use maximum uncertainty sampling to request active learning queries. Uncertainty sampling is defined as

$$U(X) = 1 - P(\hat{X}|X) \tag{5}$$

where $P(\hat{X}|X)$ is the model's estimated probability of the most likely prediction $\hat{X}$ of instance $X$. For each active learning step, the instance $X$ with the largest value of $U(X)$ is selected as the next experiment to be queried.

### E. Phase 1: Model Validation and Baseline development

To mimic the meta-learning per-task evaluation process for all models, cross validation is performed so that experiments with the same amine are grouped together. Models are trained on data associated with 15 amines and 1 amine is used for validation, i.e., a 16-fold leave-one-amine-out cross validation. We take 5 different draws of per-amine samples (either $k = 10$ for active learning models or $k + x = 20$ samples for one-shot models) to test the models under different starting conditions. Next, all active learning models request the scores of $x = 10$ more experiments sequentially from the remaining (historical data) samples for that amine. Active learning models update their uncertainty values before requesting a new sample. Thus each model trains on a total of 20 samples from the held out amine. Models are evaluated by testing on all held out amine samples in each fold. Accuracy statistics (recall, precision, accuracy, and balanced classification rate) are calculated by considering the mean per-amine accuracy statistic over the 5 random draws and taking the mean over all single amine cross-validation folds. Given the large class imbalance in the dataset (see Table I), balanced classification rate (BCR) is the primary performance metric, defined as

$$BCR = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right) \tag{6}$$

where TP, TN, FP, FN are the number of true positive, true negative, false positive, and false negative classification outcomes, respectively. For non-active learning models the BCR is used directly, while for active learning models BCR is calculated at each step of the active learning process and the area under the BCR curve (BCR AUC) is used to measure the rate of improvement. All model hyperparameters and model architecture choices listed in Table S-2 are evaluated based on the BCR or area under the BCR curve (for non-active learning or active learning models, respectively).

## F. Phase 2: Hold out Testing

The hyperparameters that provide the best results in the model validation phase for each model type and training strategy are fixed for use in hold-out testing; those finalized hyperparameters are shown in Table S-3. During this phase, models are trained with all 16 amines used in the validation phase. Trained models are then tested on 3 held out amines as indicated in Table I. Similar to the model validation phase, we take 5 different draws of per-amine samples to test the models under different starting conditions. Models that perform well in this phase under each training strategy are selected to be used in the final evaluation. Phase 2 verifies the phase 1 training and testing process, and serves as the qualifying round that determines which models advance to laboratory experimentation.

## G. Phase 3: Experimental Method

The experimental procedure for the high-throughput inverse temperature crystallization (ITC) synthesis of metal halide perovskite single crystals is described in our previous work[29]. In brief, a Hamilton Microlab NIMBUS automatic liquid handler robot pipettes four different types of stock solutions into glass vials on a 96-well microplate. These stock solutions consist of: (a) lead (II) iodide and the selected organoammonium iodide in solvent, (b) organoammonium iodide in solvent, (c) neat solvent (dimethylformamide, DMF, was used for all reactions described in this work), and (d) neat formic acid. The liquid handling robot dispenses the reagent stock solutions into pre-heated (70 °C) glass vials placed in a 96 well microplate. The plate is vortexed for 35 minutes to ensure the proper mixing of stock solutions. The robot then heats the microplates (to a nominal setting of 105 °C, which we measured as 95 °C by IR thermometry) without vortexing for 150 minutes to allow for crystal growth. The reaction outcome is scored by visual inspection into the four outcome classes described previously. The raw data file, contained in the supplementary material, includes a description of the stock solution concentrations used for each experiment, as well as details of the pipetting instructions and final compositions of each reaction.

### H. Phase 3: Testing Model Performance in the Laboratory

For each of the four amines used in the laboratory testing phase, we acquired 96 experiments sampling the concentrations uniformly in the achievable composition space (dimethylammonium iodide used existing data; the other three were acquired newly for this study). Next, two draws of $x = 10$ experiments were selected using uniform random sampling from this pool, and used to jump start the models. Models requested $k = 10$ additional experiments sequentially from within the stateset of possible achievable compositions for the amine. Because only one experiment is requested by each model at a time, the requested experiments were dispensed by manual pipetting, but otherwise follow the same experimental process described above. At the conclusion of the experiment, the results were returned to the models. Each ITC experiment requires approximately 4 hours to complete, allowing for 2 active learning rounds per amine per day. At the conclusion of the $x + k = 20$ data points, each fully trained model selected the top 9 experiments with the highest probability of yielding a large single crystal (class 4) outcome and these experiments were conducted in the laboratory using the liquid handler robot.

## III. RESULTS AND DISCUSSION

### A. Phase 1: Model Validation

The goal of this first phase was to determine the best set of hyperparameters across a wide variety of models and training strategies; the results are Figure 2. We first define three (non-active, non-meta) baselines training schemes and their results. The *historical only* training strategy

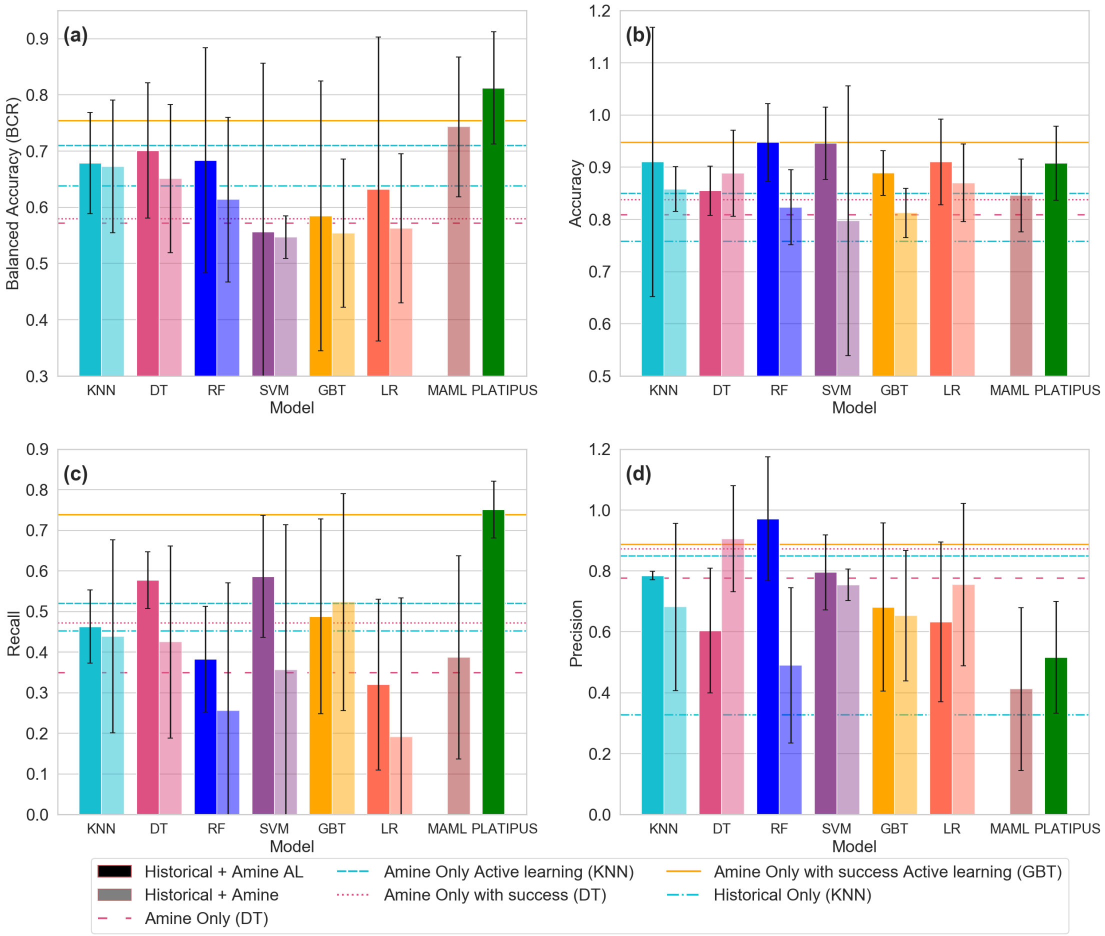


FIG. 2. Cross-validation results showing the (a) balanced classification rate (BCR), (b) accuracy, (c) recall, and (d) precision for each training strategy averaged over all folds and with 5 draws for each fold. *Historical + amine* active learning and *historical + amine* training strategy results are shown for all models. Error bars indicate the standard deviation of the average accuracy statistic across all amines (16 folds). Lines indicate the model with the highest average BCR for its corresponding strategy. The large standard deviation for all models indicates large variability across amines.

provides a baseline for how well models can predict the outcomes of reactions for new amines that have not previously been seen during training. Such models do not perform well; the best performing model using this historical only training strategy is *k*-nearest neighbors (indicated as the black dot-dashed line in Figure 2), with an average BCR value of 0.64. The second baseline, *amine only* models, only have access to limited data on the new unseen amine. The best performing models trained in this manner are worse than the best historical only model (specifically, decision tree with an average BCR of 0.57, indicated by the red dashed line in Figure 2). As noted in the

Methods section, SVM, GBT and LR require the training data to contain at least one successful sample, and the imbalances seen in these experiments often precludes this. To establish a baseline for these methods, the *amine only with success* training scheme ensures that the training data includes at least one success. This slightly improves the results over the *amine only* version, but the best performing model is still poor; the DT model achieves a BCR of 0.58, indicated by the dotted red line in Figure 2. Although all of these baselines are better than random guessing (which would have a BCR value of 0.5), there is room for improvement.

First, we assess the value of meta-learning on historical data and compare it to simply adding historical data into the model. In general, all models trained using the *historical* + *amine* data perform better than those trained using only the historical data or only the new amine data (lighter-shaded bars, in Figure 2). This indicates the value of combining both types of information when exploring a new chemical system. However, the performance is still generally poor; the best performing model among standard (non-meta-learning) models is KNN, with a BCR of 0.67 In contrast, the MAML method, using the historical data for meta-optimization, and then training on the small set of new amine data, yields better performance, with a BCR of 0.74, indicating the value of meta-learning.

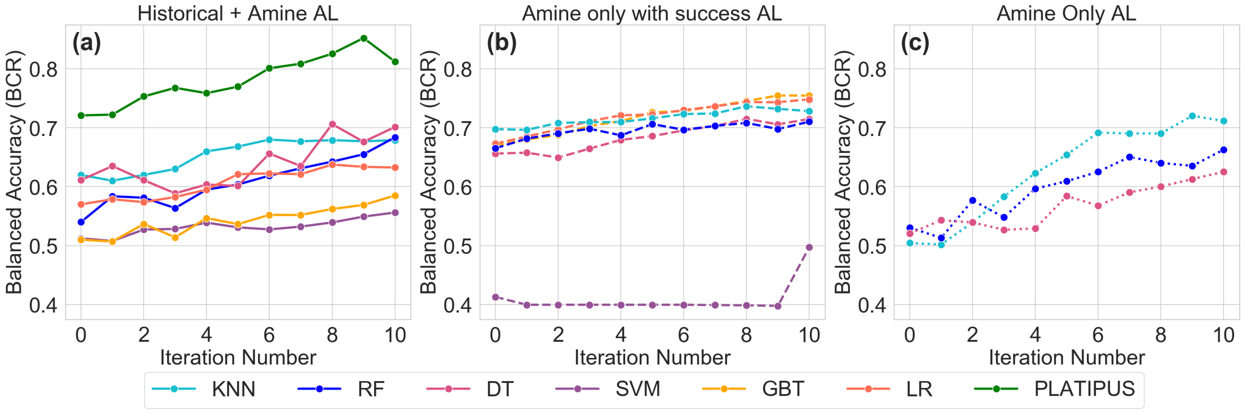


FIG. 3. Cross-validation results for the active learning models showing the number of active learning queries versus the average balanced classification rate (BCR) over 5 draws for each fold, averaged over all folds. Accuracy, recall, and precision metrics are shown in Figure S-2 . Shown models are the best per training category after a hyperparameter search. Solid lines in (a) represent the *historical+amine* training strategy and dashed lines in (b) are *amine only with successes* and dotted lines in the (c) are *amine only with random* selection.

Applying an *active learning* scheme to the *amine only* or *historical* + *amine* training results in

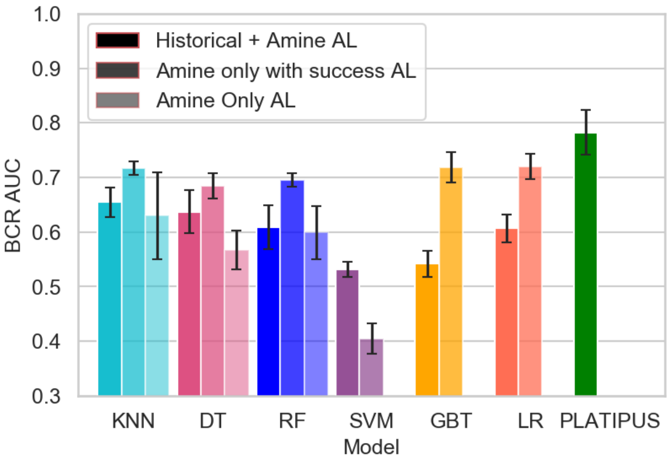


FIG. 4. Cross-validation BCR AUC for the active learning strategies. PLATIPUS has the highest BCR AUC among all models followed by KNN and DT using the historical+amine training strategy. KNN with amine only has the highest BCR AUC among the *amine only* models, but it is lower than KNN with the *historical+amine* strategy. SVM, GBT, and LR models could not be evaluated under the amine only strategy since they require both outcome classes in their training data.

a large improvement over the non-actively learned equivalent model. Because one-shot models are given access to an equivalent number ($x = 10$) of randomly sampled data points, the appropriate comparison to the actively learned models is at the $x = 10$ point on the right of the BCR plots in Figure 3 graphs. The KNN, decision tree, and random forest models learned with all available data (*historical* + *amine*) all perform similarly well after 10 active learning queries, with BCR between 0.67 and 0.7, and perform better than the SVM, GBT, or LR equivalents. The *amine only active learning* versions of KNN, DT, and RF similarly perform better than the non-active learning versions (with BCRs between 0.64 and 0.71), as does the *amine only with success active learning* for which GBT performs best with a BCR of 0.74. Thus, the active learning process improves the model more than random sampling, but the models' performance is still not particularly strong.

The PLATIPUS results (indicated by green in Figures 2-4) demonstrate the value of *active meta-learning*. Even with only the jump start data for the specific amine (the $x = 0$ point on the graphs in Figure 3), PLATIPUS already has higher BCR than the other actively learned models achieve after an additional $x = 10$ queries. After 10 active learning queries, the PLATIPUS model has a BCR of 0.81, outperforming all other models considered. However, as the standard deviation across amines is relatively large (0.1), even for the best performing models, it is important to evaluate these models on held out amines in the next phase.

## B. Phase 2: Hold-out Testing

To evaluate the models on fully unseen data, the models were next tested on three amines held out from the previous dataset (see Table I). Models were trained on the entire set of 16 amines from the validation experimentation phase using the previously determined optimal hyperparameters (see Table S-3).

Overall, the models perform better on the hold-out test set than in the cross-validation evaluation. This is likely due to the random choice of amines in the hold-out test set and not any general improvement in the models. Thus, we will focus here mostly on the relative performance of different model types and training strategies.

Models trained solely on the *historical only* and with no data from the unseen amine again perform relatively poorly, though in some cases as well or better than other options for a specific model type, as shown in Figure 5. Interestingly, the average BCR for the decision tree models is much higher for the *historical only* training strategy ($0.78 \pm 0.13$) shown as a pink dash-dot line, than for the *historical+amine* strategy ($0.68 \pm 0.12$), shown as the light pink bar. However the standard deviation of the BCR across the three held-out amines is quite large. The models trained using *amine only* perform similarly to or better than the models trained using only *historical only* data. This is different from the pattern seen during validation, and indicates a lack of generalization of the standard models to these held-out amines. The models trained using the *amine only with success* strategy are not consistently better or worse than those trained using *amine only* data. Both training strategies have large standard deviations in the BCR across amines, and the variability across amine data may be more important to a model's ability to predict successfully than the initial sampling choice. The *amine only* and *amine only with success* strategies perform reasonably well without historical data, which suggests that these are reasonable training strategies in the absence of a large historical dataset.

The standard models trained using all available data (*historical* + *amine*) again do not clearly dominate any of the other strategies using standard models. MAML similarly performs on par with, but not clearly dominating, the best of the other models, with an average BCR of 0.79. The impact of active learning on the standard models is also inconsistent across model and training strategy, with some models and strategies increasing performance under active learning and some

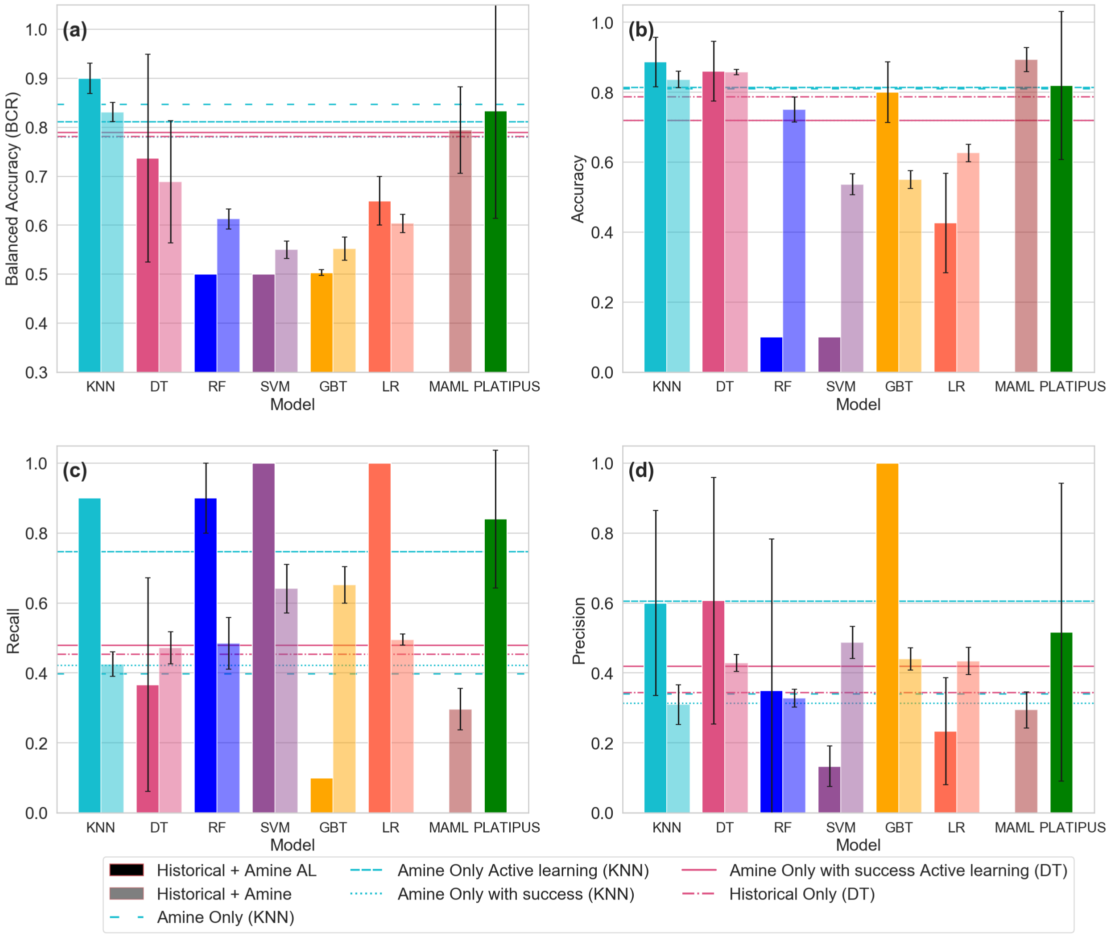


FIG. 5. Hold-out testing results showing active and non-active learning model accuracy statistics ((a) BCR, (b) accuracy, (c) recall, and (d) precision) averaged over 3 held-out amines, each with initial data chosen from 5 random draws. Bars represent the *Historical+Amine* and *Historical+Amine AL* strategies. Error bars shown indicate the standard deviation of the accuracy statistic over the three held-out amines. The best performing models for other training strategies are indicated using horizontal lines. SVM, GBT, and LR models can only be trained with at least one successful reaction, so these models do not have results for all training strategies.

decreasing. KNN performs the best with the *historical+amine* strategy and *active learning*, but some weaker models (RF, SVM, and GBT) under this training strategy have a BCR of about 0.5, which is the performance of a random model. Analyzing the active learning queries made by the KNN model revealed that the model naïvely requests the first point in the list of remaining experiments in the pool. Since this pool contains points that are uniformly sampled throughout the statespace, any point selected will improve model performance.

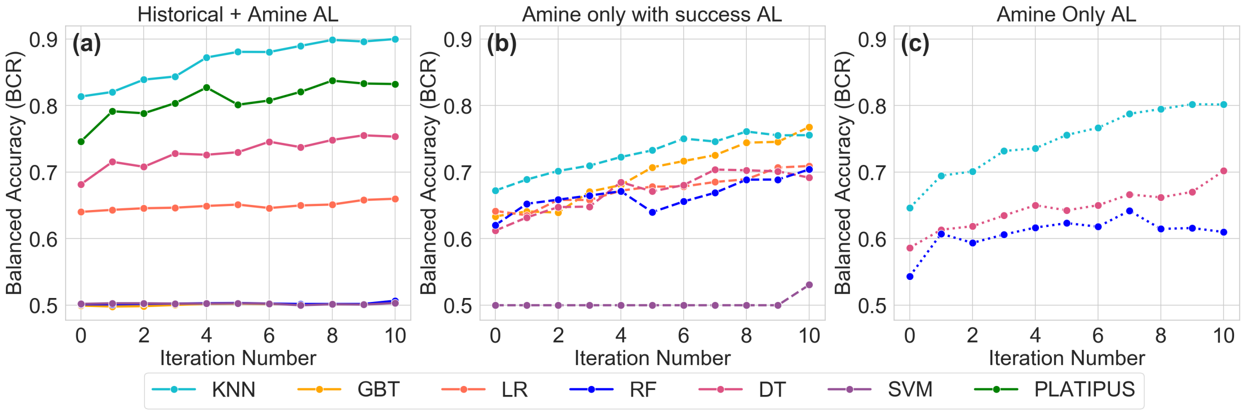


FIG. 6. Hold-out testing results for the active learning models showing the number of active learning queries versus the average balanced classification rate (BCR) over 5 draws for each amine, averaged over all amines. Accuracy, recall, and precision metrics are shown in Figure S-7 in the Supplementary Materials. Solid lines in (a) represent the *historical+amine* training strategy. Dotted lines in (b) represent *amine only* and dashed lines in (c) represent *amine only with success* training strategy. Models are evaluated based on the BCR AUC (see Figure 7) and KNN is the best performing model.

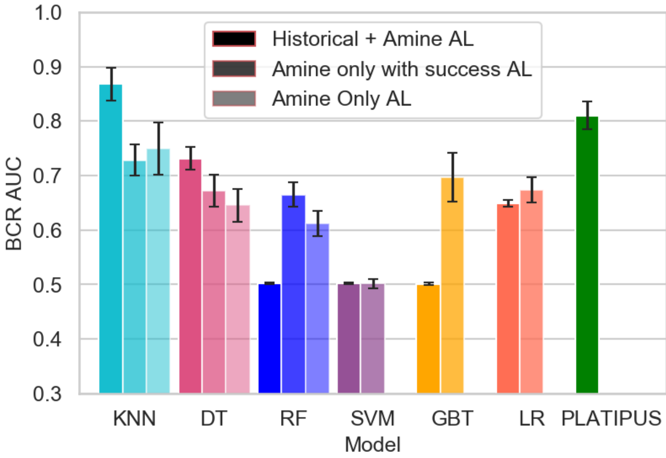


FIG. 7. Hold-out testing BCR AUC for the active learning strategies. KNN has the highest BCR AUC among all models followed by PLATIPUS and DT using the *historical+amine* training strategy.

While the validation evaluation showed PLATIPUS to be the best of the evaluated models and training strategies, the performance improvement is less clear on the hold-out testing set. PLATIPUS seems to perform similarly to the best standard models (KNN under varying strategies), with a large standard deviation in BCR across held-out amines.

In order to test the developed models in a real world scenario, the next phase of experimentation involves in-lab live active learning, including laboratory testing of chosen queries. Real world experiments require materials and labor, so we will limit our attention to the best active learning models. Based on the success of the KNN and DT models across both validation and hold-out testing, as well as the general success of the *historical+amine* active learning training strategy, the in-lab experimentation will include KNN and DT *historical+amine* active learning models as well as PLATIPUS.

Given the constraints of performing live lab experiments on previously unused amines, it is also impractical to continue using the *amine only with success* strategy; the goal in the live lab experiments is to use a limited and fixed experimental budget, which is incompatible with sampling until a success is found. Furthermore, the validation and hold-out testing results indicate that this strategy does not yield significant benefits in model performance. Thus, keeping the same hyperparameters for each model, we move forward to the next phase with an examination of the KNN, DT and PLATIPUS models using *historical+amine* in-lab active learning, and compare against a baseline of standard models using different training strategies.

### C. Phase 3: Laboratory evaluation

As discussed in the methods section, each model was trained on the historical data of 19 amines used in model validation and hold out testing phases, provided with the same $k = 10$ jump start data and allowed to request its own $x = 10$ additional experiments from the state space of possible compositions, and then evaluated on its ability to identify 9 successful reaction conditions for 4 new amines. The entire process is repeated twice for each amine, using different randomly selected jump-starts, to assess the dependence of model performance on initial conditions.

Does active learning improve model quality relative to random experiment sampling in a new task? Figure 8 compares the performance of all baseline standard and meta models, training on the same data available to the in-lab active learning models (using the same historical and jump start data on the amine), and testing using the high-throughput baseline data. These best results for the non-active learning models tend to use *historical* + *amine* data (except for KNN, which does better with *amine only* training), and MAML has the best overall performance. Therefore, when active learning is impractical, we recommend using MAML when historical data is available, and KNN when it is not. However, using active learning improves performance. Notably, the PLATIPUS model dominates all other models with an average BCR of 0.81. The DT model with active learning does better than DT with other strategies, but KNN with active learning does worse than KNN with *historical* + *amine* and *amine only* strategies. Similar to the behaviour seen in the hold-out testing phase, KNN naïvely selects points from the stateset in the order it is presented to the model for lab evaluation, which limits the information it gains from the active experiment requests.

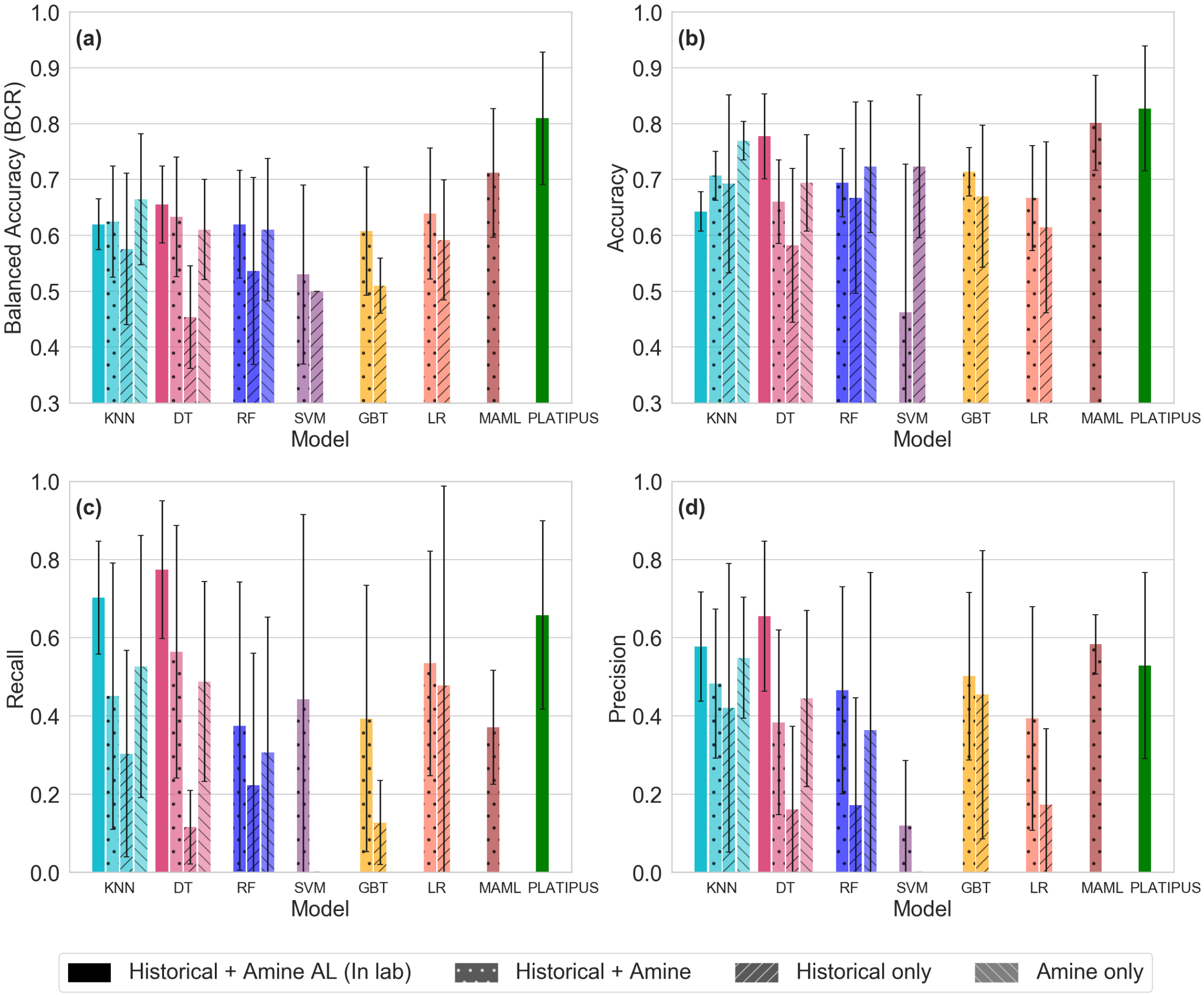


FIG. 8. Final phase testing results showing active and non-active learning model accuracy statistics ((a) BCR, (b) accuracy, (c) recall, and (d) precision) averaged over 4 amines, each with initial data chosen from 2 random draws. Solid bars represent models trained using $x = 10$ benchtop experiments (*Historical*+*Amine AL* (In lab)), dotted bars represent models trained using *historical* + *amine* strategy, bars with down-left and down-right stripes are trained using *historical only* and *amine only* strategies, respectively.

How do the active learning models improve with each experimental request? Figure 9a shows the BCR values at each step of the active learning process averaged over both draws over four

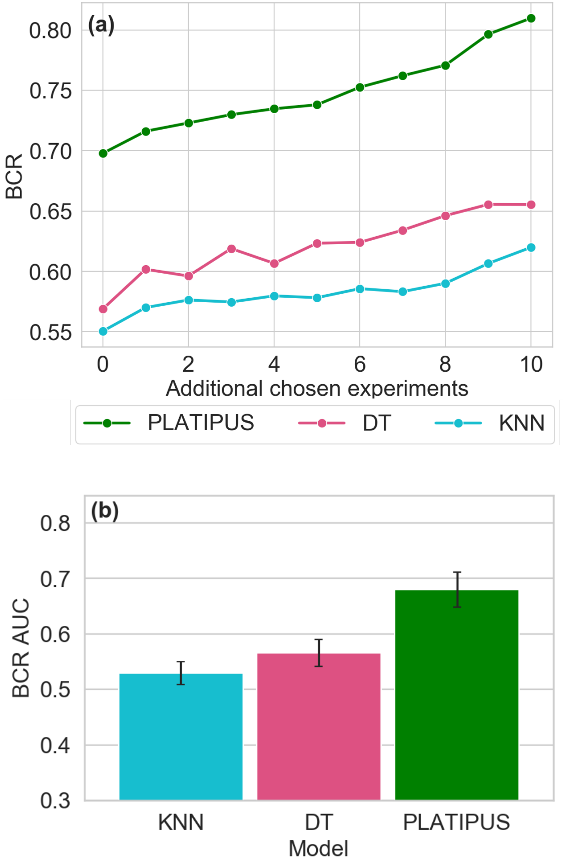


FIG. 9. Laboratory evaluation training results for the active learning models averaged over two draws over four amines. (a) active learning BCR and (b) average BCR AUC. Error bars indicate standard deviation across the 8 trials.

amines and Figure 9b shows the BCR AUC of each model. In addition to starting with an initially higher BCR, the experiment selections by PLATIPUS increase the BCR more than active learning on the DT and KNN models. This demonstrates the value of PLATIPUS for exploration.

How well can trained active learning models predict new experiments? Table IV summarizes the actual experimental outcomes observed for 9 reaction that each model predicted to be successful; this allows us to assess how well each model can be used to exploit the information it has learned during the active learning process. Large variations are observed between the different jump-start draws, which reflects the dependence on initial data. However, in every case, PLATIPUS makes more successful predictions than DT- and KNN-based active learning models. This is evidence that regardless of the initial conditions, PLATIPUS makes better use of its experimental requests than these other models to learn the relationship between composition and reaction outcome.

TABLE IV. Summary of the laboratory evaluation results on predicting 9 experimental outcomes.

| Model | | PLATIPUS | | Decision Tree | | KNN | |
|---|---|---|---|---|---|---|---|
| Amine | Draw | Fraction Success | Number of Successes | Fraction Success | Number of Successes | Fraction Success | Number of Successes |
| Dimethylamine | 1 | 0.78 | 7 | 0.67 | 6 | 0.00 | 0 |
| | 2 | 0.22 | 2 | 0.22 | 2 | 0.22 | 2 |
| 4-Chlorophenethylamine | 1 | 0.55 | 5 | 0.11 | 1 | 0.00 | 0 |
| | 2 | 0.89 | 8 | 0.22 | 2 | 0.00 | 0 |
| 4-Hydroxyphenethylamine | 1 | 0.44 | 4 | 0.22 | 2 | 0.00 | 0 |
| | 2 | 0.44 | 4 | 0.11 | 1 | 0.00 | 0 |
| 4-Chlorophenylamine | 1 | 0.78 | 7 | 0.11 | 1 | 0.22 | 2 |
| | 2 | 0.44 | 4 | 0.22 | 2 | 0.33 | 3 |

We quantify the prediction quality using a simple statistical approach. Suppose that each model is an oracle that makes correct predictions with probability $p$, i.e., each experiment is a Bernoulli trial. The number of successes $m$ that occur in a batch of $n$ experiments is the binomial distribution. Given an observation of $m$ successes, we wish to determine the probability density function (PDF) of $p$ consistent with this outcome. This is merely the PDF of the binomial distribution times the appropriate normalization factor for $n$ trials,

$$f(p) = (n+1)(1-p)^{n-m} p^{m} \binom{n}{m}. \tag{7}$$

Eq (7) can be used to assess each model's predictions quality (peaks at higher $p$) and uncertainty (width of the peak). To focus on each model's general performance, we combine the two draws together. Figure 10 plots the estimated PDF of $p$ for each model for each amine; in each case there are $n = 18$ experiments and $m$ is the sum of successes reported in Table IV for the two draws. We also compare this to the random baseline results for each amine (black line), using the data from Table I. The random baseline distribution is narrower because of the larger number of random baseline samples; the statistical treatment allows us to account for the uncertainty associated with different numbers of experimental trials in a consistent way. As depicted in Figure 10, the PDF for PLATIPUS (red) is higher or comparable to that of the other reference methods. This indicates that PLATIPUS has a better maximum likelihood ($p$ that maximizes the PDF) of making successful reaction predictions than the other approaches. However, one might also ask how the uncertainty in our estimate might change this evaluation.

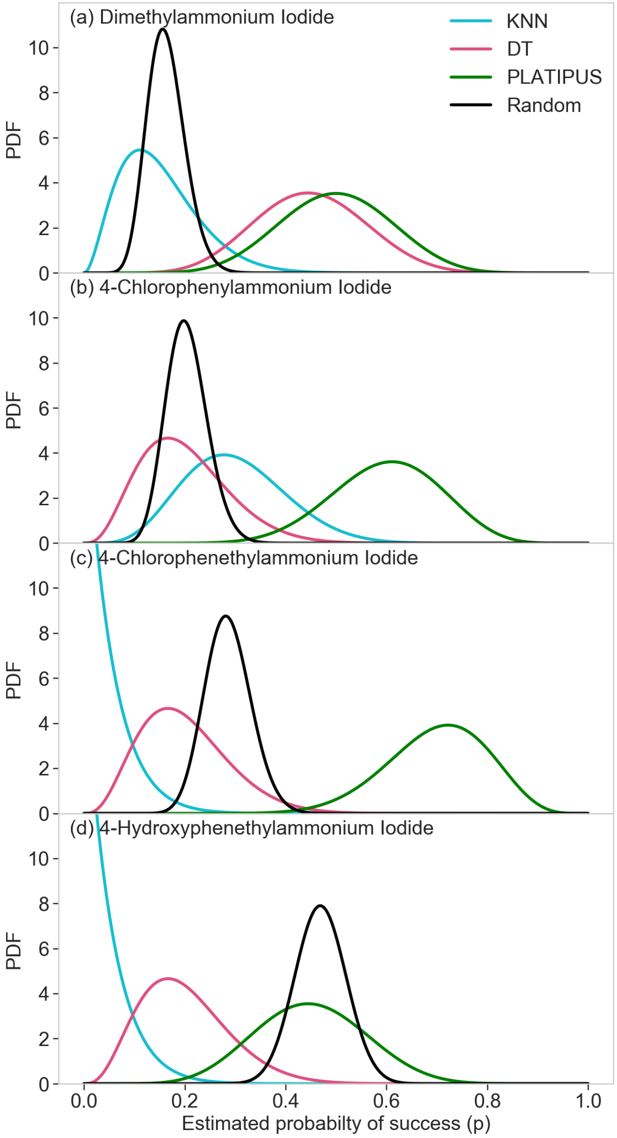


FIG. 10. Probability density function (PDF), Eq. (7) of estimated model success probability, $p$, for the KNN, DT, and PLATIPUS active learning models and random baseline data. The distribution of $p$ for PLATIPUS is better than, or comparable to all other contenders.

A useful way to approach decision making in uncertain environments is to think in terms of bets.[61]. Consider a wager placed on one of two different models, with PDFs described by $f_A(p)$ and $f_B(p)$. The integral of the joint PDF (which in this case is simply the product of the two independent PDFs, $f_A(p_A)f_B(p_B)$) for $p_A > p_B$,

$$g_{A>B} = \int_0^1 \mathrm{d}p_A \int_0^{p_A} \mathrm{d}p_B \, f_A(p_A) f_B(p_B), \tag{8}$$

indicates how often a bet on $A$ is better than a bet on $B$. An illustrative example is shown in Figure 11, using the example of dimethylammonium iodide. In each inset, the PDF of each individual model (taken from Figure 10a) is shown in the margins, and the joint PDF is depicted as a contour plot. The region below the dotted bisectrix line is where $p_A > p_B$. The integral $g_{A>B}$ is larger when more of the joint PDF sits below this bisectrix. For example, the joint PDF of the PLATIPUS and random sampling schemes (Figure 11a) is mostly below the bisectrix, indicating that the PLATIPUS model ("A") is typically more successful than the random model ("B"), as $g_{A>B}$ is closer to 1. In contrast, the difference between the PLATIPUS and DT PDFs is not as pronounced, and as a result, the joint PDF is more symmetrical about the bisectrix (Figure 11b). As a result, $g_{A>B}$ will be closer to 0.5, indicating that these are equally good bets.

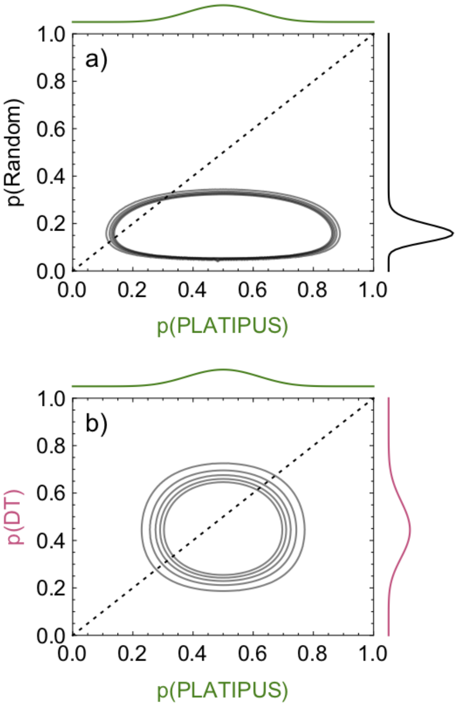


FIG. 11. Illustrative examples of comparing joint probability density functions (PDF) to determine which model is a better choice, for dimethylammonium iodide reactions. The PDF for each model is shown as the colored lines on the axes, and the joint PDF is depicted as a contour plot. The dotted diagonal line indicates the bisectrix. (a) Comparison of PLATIPUS and random choice. The contour sits below the bisectrix indicating PLATIPUS as the better bet (b) Comparison of PLATIPUS and the active decision tree model. The contour is more symmetrical around bisectrix indicating equally good bets.

Eq 8 can be evaluated analytically, resulting in a rational fraction for each value of $g_{A>B}$ (see

TABLE V. Estimation of which model is more likely to be successful, by integration of Equation (8).

| Amine | $g_{P>KNN}$ | $g_{P>DT}$ | $g_{P>Rand}$ | $g_{DT>Rand}$ |
|---|---|---|---|---|
| Dimethylamine | 0.994 | 0.627 | 0.999 | 0.996 |
| 4-Chlorophenylamine | 0.976 | 0.997 | 1.00 | 0.451 |
| 4-Chlorophenethylamine | 1.00 | 1.00 | 1.00 | 0.188 |
| 4-Hydroxyphenethylamine | 0.999 | 0.961 | 0.433 | 0.009 |

Supplementary Material). As the results are somewhat unwieldy, Table V shows the decimal truncation, with a comparison of PLATIPUS (P) to the KNN and DT active learning methods and against the random baseline; As noted above, $g_{A>B}$ values closest to 1 indicate that it is almost certain that model $A$ will have a superior outcome, and values of 0.5 indicate that each model has an equal likelihood of winning. In all cases, PLATIPUS is a better choice than the other active learning models, and in most cases should outperform every other strategy $> 96\%$ of the time. There are two exceptions: For dimethylammonium iodide, PLATIPUS outperforms DT only $62\%$ of the time. Nonetheless, PLATIPUS remains a better choice, even though this advantage is smaller than usual. For 4-hydroxyphenethylammonium iodide, despite outperforming the other active learning methods, PLATIPUS is less likely to succeed than random choice. However, this amine has an anomalously high success rate of 47%, compared to 16-28% for other amines (Table I). In other words, adopting a smart strategy offers few advantages when dumb luck has a high chance of success. For reference, the last column in Table V shows a similar comparison of the DT model against random experimentation. In only one case does the active DT model outperform the random baseline. This further highlights the strength of PLATIPUS. In summary, PLATIPUS is comparable or better to any other strategy for all amines considered, indicating that is a robust strategy to adopt when attempting new experiment campaigns.

## IV. CONCLUSION

Experimental chemistry datasets are typically small, which makes efficient data use imperative. Similarly, acquiring new experimental data is typically orders of magnitude more expensive and time consuming than computational work, so methods that reduce the need to acquire new data are valuable. Chemical reaction systems are complicated, and while there are often broad trends between different systems, each chemical system has its own unique peculiarities. By performing an extensive series of computational experiments using historical data, we have demonstrated that the MAML meta-learning method uses historical data to get more explanatory value from a subsequent fixed, limited set of data for a new chemical system. Additionally, we have demonstrated that the PLATIPUS active meta-learning method gives additional improvements in model quality when it is possible to acquire additional data. The PLATIPUS active meta-learning approach learns better models than active learning alone, on both historical data and in-laboratory testing. The demonstrated advantage of PLATIPUS for halide perovskite synthesis in the laboratory indicates its robustness to noise in a real world setting. More broadly, the training and evaluation strategies we describe are generally applicable to other types of chemical and material synthesis problems that can be described in terms of distinct, but related *tasks*. Tasks such as replacing one chemical ingredient with another, are examples of Wittgenstein's notion of family resemblance (*Famlienähnlichkeit*), in the sense that there is only a "complicated network of similarities overlapping and criss-crossing" rather than any specific features common to all tasks.[62] Meta-learning approaches, such as MAML used here, allow us to incorporate the peculiar details of the new task, while still making use of the general structure of a historical dataset of related tasks. Adding active-learning iterations using PLATIPUS increases the value of limited experiments, and thus is generally applicable to the various autonomous experimentation systems discussed in the introduction.

## ACKNOWLEDGEMENTS

This study is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001118C0036. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA. Work at the Molecular Foundry was supported by the Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. JS acknowledges the Henry Dreyfus Teacher-Scholar Award (TH-14-010) and resources of the MERCURY consortium (http://mercuryconsortium.org/) under NSF Grant No. CNS-2018427.

## AUTHOR DECLARATIONS

The authors have no conflicts to disclose.

## DATA AVAILABILITY

All data and code needed to reproduce the results of this study are available at https://github.com/darkreactions/platipus

## REFERENCES

[1] R. K. Vasudevan, K. Choudhary, A. Mehta, R. Smith, G. Kusne, F. Tavazza, L. Vlcek, M. Ziatdinov, S. V. Kalinin, and J. Hattrick-Simpers, MRS Commun. **9**, 821 (2019).

[2] J. E. Saal, A. O. Oliynyk, and B. Meredig, Annu. Rev. Mater. Res. **50**, 49 (2020).

[3] E. Stach, B. DeCost, A. G. Kusne, J. Hattrick-Simpers, K. A. Brown, K. G. Reyes, J. Schrier, S. Billinge, T. Buonassisi, I. Foster, C. P. Gomes, J. M. Gregoire, A. Mehta, J. Montoya, E. Olivetti, C. Park, E. Rotenberg, S. K. Saikin, S. Smullin, V. Stanev, and B. Maruyama, Matter (2021), 10.1016/j.matt.2021.06.036.

[4] M. M. Flores-Leonar, L. M. Mejía-Mendoza, A. Aguilar-Granda, B. Sanchez-Lengeling, H. Tribukait, C. Amador-Bedolla, and A. Aspuru-Guzik, Curr. Opin. Green Sustain. Chem. **25**, 100370 (2020).

[5] F. Häse, L. M. Roch, and A. Aspuru-Guzik, Trends Chem. **1**, 282 (2019).

[6] N. J. Szymanski, Y. Zeng, H. Huo, C. J. Bartel, H. Kim, and G. Ceder, Mater. Horiz. **8**, 2169 (2021).

[7] P. Nikolaev, D. Hooper, F. Webber, R. Rao, K. Decker, M. Krein, J. Poleski, R. Barto, and B. Maruyama, npj Comput. Mater. **2** (2016), 10.1038/npjcompumats.2016.31.

[8] J. Chang, P. Nikolaev, J. Carpena-Núñez, R. Rao, K. Decker, A. E. Islam, J. Kim, M. A. Pitt, J. I. Myung, and B. Maruyama, Sci. Rep. **10** (2020), 10.1038/s41598-020-64397-3.

[9] A. E. Gongora, B. Xu, W. Perry, C. Okoye, P. Riley, K. G. Reyes, E. F. Morgan, and K. A. Brown, Sci. Adv. **6**, eaaz1708 (2020).

[10] R. W. Epps, M. S. Bowen, A. A. Volk, K. Abdel-Latif, S. Han, K. G. Reyes, A. Amassian, and M. Abolhasani, Adv. Mater. **32**, 2001626 (2020).

[11] A. A. Volk and M. Abolhasani, Trends Chem. **3**, 519 (2021).

[12] A. Y. Fong, L. Pellouchoud, M. Davidson, R. C. Walroth, C. Church, E. Tcareva, L. Wu, K. Peterson, B. Meredig, and C. J. Tassone, J. Chem. Phys. **154**, 224201 (2021).

[13] B. P. MacLeod, F. G. L. Parlane, T. D. Morrissey, F. Häse, L. M. Roch, K. E. Dettelbach, R. Moreira, L. P. E. Yunker, M. B. Rooney, J. R. Deeth, V. Lai, G. J. Ng, H. Situ, R. H. Zhang, M. S. Elliott, T. H. Haley, D. J. Dvorak, A. Aspuru-Guzik, J. E. Hein, and C. P. Berlinguette, Sci. Adv. **6**, eaaz8867 (2020).

[14] B. Burger, P. M. Maffettone, V. V. Gusev, C. M. Aitchison, Y. Bai, X. Wang, X. Li, B. M. Alston, B. Li, R. Clowes, N. Rankin, B. Harris, R. S. Sprick, and A. I. Cooper, Nature **583**, 237 (2020).

[15] A. G. Kusne, H. Yu, C. Wu, H. Zhang, J. Hattrick-Simpers, B. DeCost, S. Sarker, C. Oses, C. Toher, S. Curtarolo, A. V. Davydov, R. Agarwal, L. A. Bendersky, M. Li, A. Mehta, and I. Takeuchi, Nat. Commun. **11** (2020), 10.1038/s41467-020-19597-w.

[16] A. Dave, J. Mitchell, K. Kandasamy, H. Wang, S. Burke, B. Paria, B. Póczos, J. Whitacre, and V. Viswanathan, Cell Rep. Phys. Sci. **1**, 100264 (2020).

[17] A. K. Jena, A. Kulkarni, and T. Miyasaka, Chem. Rev. **119**, 3036 (2019).

[18] M. D. Smith, E. J. Crace, A. Jaffe, and H. I. Karunadasa, Annu. Rev. Mater. Res. **48**, 111 (2018).

[19] M. Ahmadi, M. Ziatdinov, Y. Zhou, E. A. Lass, and S. Kalinin, Joule (2021), 10.1016/j.joule.2021.10.001.

[20] K. Higgins, S. M. Valleti, M. Ziatdinov, S. V. Kalinin, and M. Ahmadi, ACS Energy Lett. **5**, 3426 (2020).

[21] S. Chen, Y. Hou, H. Chen, X. Tang, S. Langner, N. Li, T. Stubhan, I. Levchuk, E. Gu, A. Osvet, and C. J. Brabec, Adv. Energy Mater. **8**, 1701543 (2017).

[22] J. Kirman, A. Johnston, D. A. Kuntz, M. Askerka, Y. Gao, P. Todorović, D. Ma, G. G. Privé, and E. H. Sargent, Matter **2**, 938 (2020).

[23] Z. Li, P. Nega, M. Najeeb, C. Dun, M. Zeller, J. Urban, W. Saidi, J. Schrier, A. Norquist, E. Chan, and et al., ChemRxiv (2021), 10.33774/chemrxiv-2021-w2c7b.

[24] Y. Zhao, J. Zhang, Z. Xu, S. Sun, S. Langner, N. T. P. Hartono, T. Heumueller, Y. Hou, J. Elia, N. Li, G. J. Matt, X. Du, W. Meng, A. Osvet, K. Zhang, T. Stubhan, Y. Feng, J. Hauch, E. H. Sargent, T. Buonassisi, and C. J. Brabec, Nat. Commun. **12** (2021), 10.1038/s41467-021-22472-x.

[25] S. Sun, A. Tiihonen, F. Oviedo, Z. Liu, J. Thapa, Y. Zhao, N. T. P. Hartono, A. Goyal, T. Heumueller, C. Batali, A. Encinas, J. J. Yoo, R. Li, Z. Ren, I. M. Peters, C. J. Brabec, M. G. Bawendi, V. Stevanovic, J. Fisher, and T. Buonassisi, Matter **4**, 1305 (2021).

[26] F. Akhundova, L. Lüer, A. Osvet, J. Hauch, I. M. Peters, K. Forberich, N. Li, and C. Brabec, Appl. Phys. Lett. **118**, 243903 (2021).

[27] J. Li, J. Li, R. Liu, Y. Tu, Y. Li, J. Cheng, T. He, and X. Zhu, Nat. Commun. **11** (2020), 10.1038/s41467-020-15728-5.

[28] J. C. Dahl, X. Wang, X. Huang, E. M. Chan, and A. P. Alivisatos, J. Am. Chem. Soc. **142**, 11915 (2020).

[29] Z. Li, M. A. Najeeb, L. Alves, A. Z. Sherman, V. Shekar, P. C. Parrilla, I. M. Pendleton, W. Wang, P. W. Nega, M. Zeller, J. Schrier, A. J. Norquist, and E. M. Chan, Chem. Mater. **32**, 5650 (2020).

[30] Y. Tang, Z. Li, M. A. N. Nellikkal, H. Eramian, E. M. Chan, A. J. Norquist, D. F. Hsu, and J. Schrier, J. Chem. Inf. Model. **61**, 1593 (2021).

[31] P. W. Nega, Z. Li, V. Ghosh, J. Thapa, S. Sun, N. T. P. Hartono, M. A. N. Nellikkal, A. J. Norquist, T. Buonassisi, E. M. Chan, and J. Schrier, Appl. Phys. Lett. **119**, 041903 (2021).

[32] I. M. Pendleton, M. K. Caucci, M. Tynes, A. Dharna, M. A. N. Nellikkal, Z. Li, E. M. Chan, A. J. Norquist, and J. Schrier, J. Phys. Chem. C **124**, 13982 (2020).

[33] C. Dai and S. C. Glotzer, J. Phys. Chem. B **124**, 1275 (2020).

[34] J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev, and A. E. Roitberg, J. Chem. Phys. **148**, 241733 (2018).

[35] C. Duan, F. Liu, A. Nandy, and H. J. Kulik, J. Phys. Chem. Lett. **12**, 4628 (2021).

[36] Z. Zhou, X. Li, and R. N. Zare, ACS Cent. Sci. **3**, 1337 (2017).

[37] V. Duros, J. Grizou, W. Xuan, Z. Hosni, D.-L. Long, H. N. Miras, and L. Cronin, Angew. Chem. Int. Edit. **129**, 10955 (2017).

[38] V. Duros, J. Grizou, A. Sharma, S. H. M. Mehr, A. Bubliauskas, P. Frei, H. N. Miras, and L. Cronin, J. Chem. Inf. Model. **59**, 2664 (2019).

[39] B. Rohr, H. S. Stein, D. Guevarra, Y. Wang, J. A. Haber, M. Aykol, S. K. Suram, and J. M. Gregoire, Chem. Sci. **11**, 2696 (2020).

[40] A. McDannald, M. Frontzek, A. T. Savici, M. Doucet, E. E. Rodriguez, K. Meuse, J. Opsahl-Ong, D. Samarov, I. Takeuchi, A. G. Kusne, and W. Ratcliff, "On-the-fly autonomous control of neutron diffraction via physics-informed bayesian active learning," (2021), arXiv:2108.08918 [cond-mat.mtrl-sci].

[41] Y. Tian, D. Xue, R. Yuan, Y. Zhou, X. Ding, J. Sun, and T. Lookman, Phys. Rev. Mater. **5** (2021), 10.1103/physrevmaterials.5.013802.

[42] C. Cai, S. Wang, Y. Xu, W. Zhang, K. Tang, Q. Ouyang, L. Lai, and J. Pei, J. Med. Chem. **63**, 8683 (2020).

[43] M. M. Sultan and V. S. Pande, J. Phys. Chem. B **122**, 5291 (2017).

[44] W.-F. Zeng, X.-X. Zhou, W.-J. Zhou, H. Chi, J. Zhan, and S.-M. He, Anal. Chem. **91**, 9724 (2019).

[45] J. Pan, K. L. Low, J. Ghosh, S. Jayavelu, M. M. Ferdaus, S. Y. Lim, E. Zamburg, Y. Li, B. Tang, X. Wang, J. F. Leong, S. Ramasamy, T. Buonassisi, C.-K. Tham, and A. V.-Y. Thean, ACS Appl. Nano Mater. **4**, 6903 (2021).

[46] M. L. Hutchinson, E. Antono, B. M. Gibbons, S. Paradiso, J. Ling, and B. Meredig, "Overcoming data scarcity with transfer learning," (2017), arXiv:1711.05099 [cs.LG].

[47] J. Vanschoren, in *Automated Machine Learning* (Springer International Publishing, 2019) pp. 35–61.

[48] I. Olier, N. Sadawi, G. R. Bickerton, J. Vanschoren, C. Grosan, L. Soldatova, and R. D. King, Mach. Learn **107**, 285 (2017).

[49] C. Q. Nguyen, C. Kreatsoulas, and K. M. Branson, ChemRxiv (2020), 10.26434/chemrxiv.11981622.v1.

[50] S. Deepika and T. Geetha, J. Biomed. Inform. **84**, 136 (2018).

[51] J. Wang, S. Zheng, J. Chen, and Y. Yang, J. Chem. Inf. Model. **61**, 1627 (2021).

[52] R. Barrett and A. D. White, J. Chem. Inf. Model. **61**, 95 (2020).

[53] F. Runge, D. Stoll, S. Falkner, and F. Hutter, arXiv preprint arXiv:1812.11951 (2018).

[54] K. Gu, Y. Zhang, and J. Qiao, IEEE Trans. Ind. Informat **17**, 2261 (2021).

[55] Y. Sun, R. F. DeJaco, Z. Li, D. Tang, S. Glante, D. S. Sholl, C. M. Colina, R. Q. Snurr, M. Thommes, M. Hartmann, and J. I. Siepmann, Sci. Adv. **7**, eabg3983 (2021).

[56] C. Finn, P. Abbeel, and S. Levine, in *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (JMLR. org, 2017) pp. 1126–1135.

[57] C. Finn, K. Xu, and S. Levine, in *Advances in Neural Information Processing Systems* (2018) pp. 9516–9527.

[58] J. Schrier, J. Chem. Educ. **98**, 1659 (2021).

[59] P. Raccuglia, K. C. Elbert, P. D. Adler, C. Falk, M. B. Wenny, A. Mollo, M. Zeller, S. A. Friedler, J. Schrier, and A. J. Norquist, Nature **533**, 73 (2016).

[60] X. Jia, A. Lynch, Y. Huang, M. Danielson, I. Lang'at, A. Milder, A. E. Ruby, H. Wang, S. A. Friedler, A. J. Norquist, *et al.*, Nature **573**, 251 (2019).

[61] A. Duke, *Thinking in bets: Making smarter decisions when you don't have all the facts* (Portfolio, 2019).

[62] L. Wittgenstein, *Philosophical investigations* (John Wiley & Sons, 2010) Chap. §65-71.

[63] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, in *2010 20th international conference on pattern recognition* (IEEE, 2010) pp. 3121–3124.

[64] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, Nature **559**, 547 (2018).

[65] V. Dragone, V. Sans, A. B. Henson, J. M. Granda, and L. Cronin, Nat. Commun. **8**, 1 (2017).

[66] T. Lookman, P. V. Balachandran, D. Xue, and R. Yuan, npj Comput. Mater. **5**, 1 (2019).

[67] C. Q. Nguyen, C. Kreatsoulas, and K. M. Branson, arXiv preprint arXiv:2003.05996 (2020).

[68] I. Olier, N. Sadawi, G. R. Bickerton, J. Vanschoren, C. Grosan, L. Soldatova, and R. D. King, Mach. Learn **107**, 285 (2018).

[69] I. M. Pendleton, G. Cattabriga, Z. Li, M. A. Najeeb, S. A. Friedler, A. J. Norquist, E. M. Chan, and J. Schrier, MRS Commun. **9**, 846 (2019).

[70] G. Pilania, P. V. Balachandran, C. Kim, and T. Lookman, Front. Mater. **3**, 19 (2016).

[71] D. Reker, E. A. Hoyt, G. J. Bernardes, and T. Rodrigues, Cell Rep. Phys. Sci. **1**, 100247 (2020).

[72] J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev, and A. E. Roitberg, J. Chem. Phys. **148**, 241733 (2018).

[73] C. Suh, C. Fare, J. A. Warren, and E. O. Pyzer-Knapp, Annu. Rev. Mater. Res. **50**, 1 (2020).

[74] S. Sun, N. T. Hartono, Z. D. Ren, F. Oviedo, A. M. Buscemi, M. Layurova, D. X. Chen, T. Ogunfunmi, J. Thapa, S. Ramasamy, *et al.*, Joule **3**, 1437 (2019).

[75] Z. Yang, J. Shu, Y. Liang, D. Meng, and Z. Xu, arXiv preprint arXiv:2009.00792 (2020).

[76] B. Yılmaz and R. Yıldırım, Nano Energy **1**, 105546 (2020).

[77] L. Zhang, M. He, and S. Shao, Nano Energy **1**, 105380 (2020).

[78] M. Zhong, K. Tran, Y. Min, C. Wang, Z. Wang, C.-T. Dinh, P. D. Luna, Z. Yu, A. S. Rasouli, P. Brodersen, S. Sun, O. Voznyy, C.-S. Tan, M. Askerka, F. Che, M. Liu, A. Seifitokaldani, Y. Pang, S.-C. Lo, A. Ip, Z. Ulissi, and E. H. Sargent, Nature **581**, 178 (2020).