The Journal of Chemical Physics ARTICLE scitation.org/journal/jcp

Aggressively optimizing validation statistics can degrade interpretability of data-driven materials models

Cite as: J. Chem. Phys. 155, 054105 (2021); doi: 10.1063/5.0050885
Submitted: 19 March 2021 • Accepted: 8 July 2021 •
Published Online: 3 August 2021

Katherine Lei,[1] Howie Joress,[2] Nils Persson,[2] Jason R. Hattrick-Simpers,[2] and Brian DeCost[2,a)]

AFFILIATIONS

[1] Montgomery Blair High School, 57 University Blvd E., Silver Spring, Maryland 20901, USA
[2] Material Measurement Laboratory, National Institute of Standards and Technology, 100 Bureau Drive, Gaithersburg, Maryland 20899, USA

Note: This paper is part of the JCP Special Topic on Computational Materials Discovery.
[a)] **Author to whom correspondence should be addressed:** brian.decost@nist.gov

ABSTRACT

One of the key factors in enabling trust in artificial intelligence within the materials science community is the interpretability (or explainability) of the underlying models used. By understanding what features were used to generate predictions, scientists are then able to critically evaluate the credibility of the predictions and gain new insights. Here, we demonstrate that ignoring hyperparameters viewed as less impactful to the overall model performance can deprecate model explainability. Specifically, we demonstrate that random forest models trained using unconstrained maximum depths, in accordance with accepted best practices, often can report a randomly generated feature as being one of the most important features in generated predictions for classifying an alloy as being a high entropy alloy. We demonstrate that this is the case for impurity, permutation, and Shapley importance rankings, and the latter two showed no strong structure in terms of optimal hyperparameters. Furthermore, we demonstrate that, for the case of impurity importance rankings, only optimizing the validation accuracy, as is also considered standard in the random forest community, yields models that prefer the random feature in generating their predictions. We show that by adopting a Pareto optimization strategy to model performance that balances validation statistics with the differences between the training and validation statistics, one obtains models that reject random features and thus balance model predictive power and explainability.

Published by AIP Publishing. https://doi.org/10.1063/5.0050885

I. INTRODUCTION

The use of artificial intelligence (AI) or machine learning (ML) in the physical sciences has exploded over the past ten years.[1–4] From those efforts, a number of truly remarkable discoveries in materials science have been made, including new phase change materials,[5] amorphous alloys,[6] and novel drugs.[7] Materials science presents an interesting use case for AI/ML techniques where datasets are small and expensive, contain underlying physical explanations, and are heterogeneously distributed in the search space of interest. As a result, AI practitioners increasingly look to the materials science community for high quality, statistically significant data sparsely distributed in feature space to develop and train new types of AI models. Meanwhile, the materials science community prioritizes gaining new insights from the AI models through understanding how the featurization of their materials is translated into predictions.

In the absence of true scientific-AI models that incorporate physical models within the AI frameworks, the current state of the art is to use interpretable[8] or explainable[9] AI to assess the scientific basis of model predictions. The rationale behind this is clear: given a dataset, its featurization, and a trained model, trust in the predictions of the model can be improved if the model reports on the relationship between the features and the model outcome. Existing interpretable methods tend to involve sacrificing model flexibility for increased inductive bias, while existing "explainability" methods require *post-hoc* interrogation of "black box" models. An interpretable model might provide the predictions as a function of feature dependence (e.g., log-linear properties such as activation

energies). An AI model used for predicting the tendency of a new multi-component alloy to form a solid solution would be expected to rely on the difference in atomic radii and the similarity of crystal structures, electronegativity, and valence of the elements as the most important features.[10] Trusting the predictions of interpretable or explainable models can come at the peril of the reaffirmation of existing scientific biases or intuitions surrounding expected causal laws, which may result in ignoring other plausible explanations for the relationships between features and prediction.

There are a variety of techniques researchers use to explain the behavior of black box models. In the case of a model such as Random Forest (RF),[11] explainability is often pursued through feature importance ranking (FIR[12,13]). This provides a window into which input features of a dataset were deemed important by the model; a variety of methods are typically used, including impurity, permutation, and Shapley importance rankings. In addition, partial dependence plots[14,15] attempt to show univariate predictive trends after accounting for the effects of all other input features. Similarly, a variety of visualization techniques offer subjective insight into the image features that deep image recognition neural networks are optimized to respond to[16]. These methods tend to provide a subjective explanatory value, often requiring interactive exploration for the research to "connect the dots" of understanding.

Here, we investigate the sensitivity of FIR with respect to hyperparameters for a model to predict high entropy alloy (HEA) or metallic glass (MG) formation. We will demonstrate that although a broad range of hyperparameters yield RF models with acceptable validation statistics, many models will prioritize a spurious feature filled with random numbers over more physically meaningful features. We will further demonstrate that this is the case for three of the most commonly used feature ranking techniques for RF. We use a hyperparameter study to demonstrate that by simultaneously seeking to maximize the validation statistics and minimize the difference between the validation and training statistics, an acceptably accurate model that deprioritizes the random feature can be generated only for the impurity importance ranking. Along the way, we will also show that the standard practice of not restricting the maximum tree depth yields models that do not minimize the difference in validation and training statistics.

II. METHODS

The composition and phase of the HEA alloys used as the training set were obtained from the multi-principal element alloys (MPEA) dataset available from Citrine.[17,18] The dataset has been discussed in detail elsewhere, but briefly it contains 1545 entries for different high entropy alloys and includes descriptors such as processing, types of phases observed (called microstructure in the original dataset), and physical properties such as density and Young's modulus, among other properties. Here, the dataset was converted to a set of descriptors relating the chemical composition to a set of multilabel encoding of the experimentally observed crystal structure: BCC, FCC, HCP, and intermetallic. In this context, the multilabel annotation for each structure consists of a binary indicator for the presence of each phase so that models can be generated for each column separately. We also created a label HEA, which is false for any alloy that contains more than one structure in its multilabel annotation, even within the same structure family (e.g., BCC + B2, BCC + intermetallic, or BCC + BCC). The dataset spans a broad range of alloy chemistries including traditional Cantor alloys and refractory alloys. In this study, we focused on the largest subset of the MPEA dataset, entries with a processing flag of "cast," which contained 599 entries including both Cantor and refractory high entropy alloys. Data without microstructure metadata were dropped from the training set. As a note, of the 599 entries, only three single-phase alloys were not single-phase FCC or BCC alloys, i.e., they were fully ordered intermetallics. The MG dataset considered was pulled from the metallic glass (MG) dataset generated by Ward *et al.*[12] This dataset contains 5792 unique alloys, which have been labeled as glass, crystalline, or mixed phase. A logical filter was used to define a binary classifier describing whether a given alloy is single phase glass forming or not. No additional conditioning was necessary for this dataset.

The compositions of the HEAs and MGs were featurized using the standard Magpie feature set,[12,19] except that the component-wise space group number features were excluded. The Magpie features were each normalized to a standard distribution prior to training and evaluating models using the StandardScalar package from scikit-learn.[20] The interested reader should look to the recent paper by Wang *et al.* for a primer on best practices for materials science model building.[21] Feature standardization is not necessary for generating good RF models but may help the accuracy of feature ranking. Train/validation splits were created using chemical similarity of the alloys to prevent the information leakage resulting in a "leave k alloy systems out" cross validation (LKSO-CV) scheme. For the purposes of this study, an additional random feature was added to the feature set by randomly sampling values for each entry from the standard normal distribution. A similar random feature injection study was performed by Zahrt *et al.* where the focus was on the critical evaluation of combinatorial datasets and ensuring models were trained on linkages between descriptors and the desired response variable, not on noise.[22] A unique random feature column was sampled for each cross-validation iteration.

Random Forest (RF) classification from scikit-learn was used for all models used in this study. The details of our base model can be found in the accompanying code but are described here briefly. To critically evaluate the explainability imparted by different feature ranking tools for RF, we conducted a gridded hyperparameter study of the HEA and MG models. Of the many hyperparameters in the RF model and training algorithm, we chose to study the following three hyperparameters: the minimum number of training samples allowed per leaf node in each decision tree, the maximum number of features considered during a split, and the maximum depth of the trees within the forest. Notably, textbooks and best practices generally support an unconstrained maximum tree depth.[23] Recommended values for the maximum features per split and minimum number of samples per leaf are generally the square root of the number of features and 1, respectively. RF models are widely considered to be relatively insensitive to these hyperparameter values. We have found previously (and will demonstrate below) that there exist a broad range of values that produce roughly equivalent models in terms of their predictive power (as measured by validation set accuracy). We performed a grid-based hyperparameter study, performing LKCSO-CV for each hyperparameter combination in the grid. For the HEA dataset, LKCSO-CV was performed 50×, but for the larger MG dataset, it was only performed 25× for each set of

hyperparameters. We varied both the maximum number of features and the minimum number of samples per leaf, which could be 1, 2, 5, or 10. For this Magpie featurization, the baseline setting for the maximum number of features is 10 [using the sqrt(n_features) heuristic]. The maximum depth of the trees was allowed to be 1, 2, 3, 4, 5, 10, or 15. This analysis was performed for impurity, permutation, and Shapley importance rankings described below.

For each version of the model, we used the 25 or 50 LKCSO-CV iterations to quantify the 95% confidence intervals (CFIs) for model accuracy and feature importance ranking with respect to the stochasticity of RF training. We use an 80/20 ratio for these random train/validation splits so that each LKCSO-CV iteration is trained on 80% and validated on 20% of the unique chemical systems in the full dataset. Here, we explicitly do not fix the random state seed so that each of the 50 models has a different initialization, and thus, we avoid only exploring models that have fortuitous test/train splits and initializations. Feature importance rankings, Area Under the Receiver Operating Curve (AUC) values, and precision and recall were obtained for all models across the 50 training cycles and used to calculate the CV-averaged importance of each feature. We report 95% confidence intervals (CFIs) for these metrics, computed across 50 LKCSO-CV iterations.

There are three commonly used ways to quantify feature importance for random forest. The first is the Gini or impurity importance, which is the default implementation in scikit-learn. The default implementation of impurity importance is calculated on the training dataset and calculates importance based on an average of how high up in the decision tree each feature was. Such importance values are generated on the training set and may not necessarily represent the importance values for the validation set. They are also sensitive to the cardinality of features, e.g., features with more unique entries are more likely to be deemed important. The latter issue is likely to not be a great issue with the current dataset described using the Magpie feature set as roughly a third of the features are compositional averages of elemental properties and will therefore provide many unique entries. The second method of quantifying feature importance is the permutation importance, which was developed by the bioinformatics community[24] and was run using the inspection module in scikit-learn.[11] Permutation importance rankings are calculated by randomly permuting individual features and evaluating the performance of the model on the validation data, with features more negatively impacting model performance upon permutation being ranked as more important. Permutation importance rankings are more computationally intensive than impurity importance rankings; therefore, the number of training cycles for the former was reduced to 25 to reduce the total computation time to ~3 days. The final feature ranking technique used was to calculate the Shapley values that compare models trained on all possible subsets of input features to obtain feature importance rankings.[25] The generalization SHAP (SHapley Additive explanation)[26] values extend these rankings to provide relative contributions to the model predictions for each input feature. In addition to providing insight into the relevance of input features, Shapley methods can also be used to assess the quality and influence of training examples and to potentially identify outliers.[27,28] To obtain per-feature Shapley importance rankings, we compute the instance-wise mean absolute value of the Shapley values for each instance in the validation set. Because we use feature standardization, these aggregate Shapley importance values are directly comparable and can be used to generate feature importance rankings in the same way as with impurity importance ranking.

III. RESULTS

Figures 1(a)–1(c) present the results of all of the impurity, permutation, and Shapley importance vs hyperparameter studies for the HEA model; each marker corresponds to a single CV iteration. Figures 1(d)–1(f) present the results of the same hyperparameter study for the MG dataset. The plots show the interplay between AUC train, AUC validation, and the importance ranking of the random feature for the three models considered. For the smaller HEA [Figs. 1(b) and 1(c)] dataset, both the Shapley and permutation importance rankings show no clear structure in terms of the random feature importance ranking. This means that neither an overfitting model (e.g., high AUC_{train} and low AUC_{val}) nor a severely underfitting model (e.g., low AUC_{train}) is statistically more likely to rank the random feature as being important than a well-trained model. A figure of the statistical distribution of the random feature importance ranking can be found in the supplementary material, Fig. 1. Any combination of AUC_{train} and AUC_{val} scores is likely to rank the random feature as being either very important or unimportant to the model's predictions. Results from the Shapley and permutation importance rankings from the MG model [Figs. 1(e) and 1(f)], where an order of magnitude increase in the number of data points reduces the influence of bad CV splits, reflect the same overall trend.

Conversely, the impurity importance rankings for the HEA model [Fig. 1(a)] and the MG model [Fig. 1(d)] have clear structures. In each dataset, both the underfitting and overfitting models show a statistical preference for a higher ranking of the random feature. The smaller HEA dataset feature rankings do exhibit some instability, likely due to bad CV splits; however, in the case of the larger MG dataset, the trend is clearer. For both datasets, there is a sweet spot at intermediate AUC_{train} for which the random feature is shown to be less important on average (rank >80 out of 106 total features). For the current study, neither permutation importance nor Shapley importance rankings showed a sufficient structure during hyperparameter tuning to be used in model tuning. In the ensuing discussion, we focus on only the impurity feature importance ranking.

Figures 2(a) and 2(c) show the variation of the AUC_{val} and AUC_{train} for the MG and HEA averaged models as a function of the hyperparameters. The dashed line in the figures represent models for which the $AUC_{val} = AUC_{train}$, and for all models below this line, the training statistics are better than the validation statistics, as expected. The color map indicates the average impurity ranking of the random feature for each hyperparameter tuning iteration. A clear trend across both models is that as the training AUC approaches unity (e.g., a perfect fit of the training data), the rank of the random feature increases until it becomes the most important feature. To be clear, for both models, we observe a global maximum of AUC_val in the hyperparameter region near the default hyperparameter values, but this is where the highest feature importance ranking of the random sentinel feature is located. This suggests that the CV procedure may be somewhat overestimating the true generalization performance of these models, leading to some degree of overfitting (consistent with informing predictive decisions based on a spurious feature).

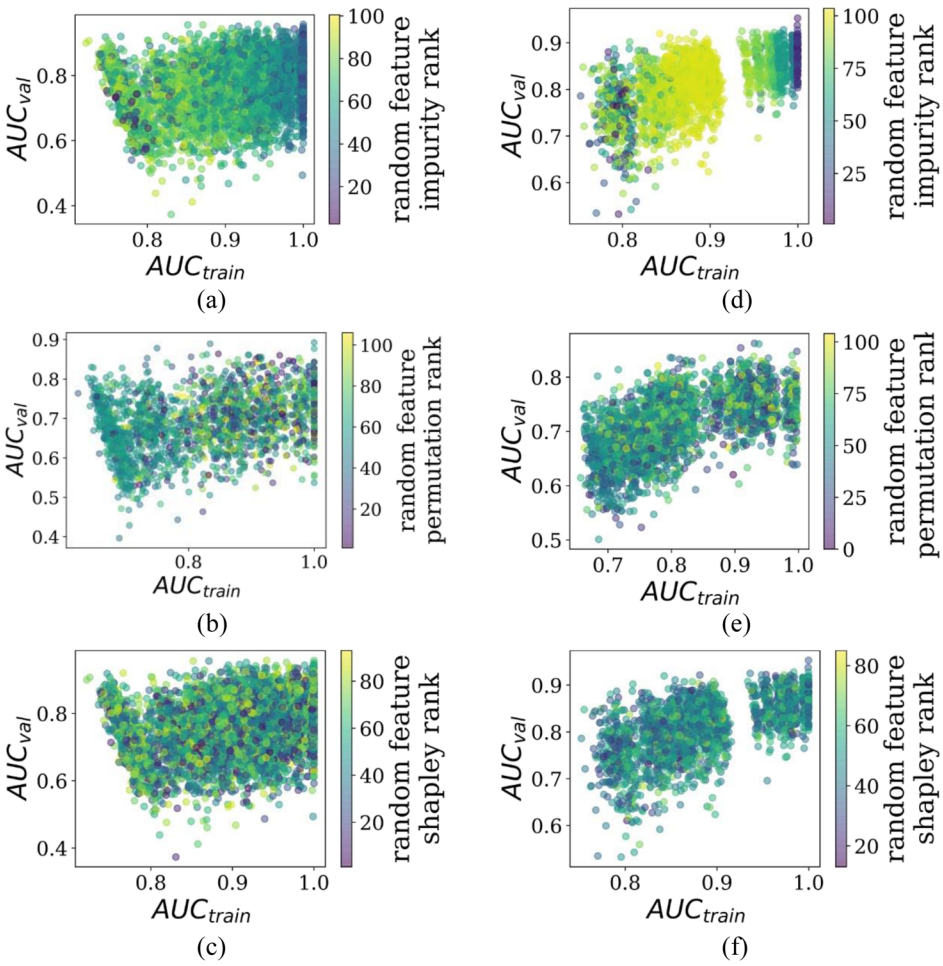

FIG. 1. Summary plots of hyperparameter studies across all replicates. The correlation of random feature rank using impurity importance is shown in (a) for HEA and in (d) for MG. The lack of clear correlation of the random feature rank using permutation and Shapley importance is shown in (b) and (c) for HEA and (e) and (f) for MG.

Figures 2(b) and 2(d) show that there is a broad range of models with comparable AUC_{val} with varying levels of discrepancy between the training and validation AUC ($AUC_{train}-AUC_{val}$). Among models with high AUC_{val}, there is a clear trend of increasing random feature importance as the train/val AUC gap increases. Figures 2(c) and 2(e) present the same information as Figs. 2(b) and 2(d) but plot the random feature importance on the x-axis and indicate $AUC_{train}-AUC_{val}$ with a color map. These figures show more clearly that models with high AUC_{val} can give vastly different importance rankings for the random feature.

The regions that show the highest AUC_{train} and AUC_{val}, the largest difference in training and validation AUCs, and the highest random feature importance contain hyperparameters that are closest to the defaults for RF. While a common strategy for hyperparameter selection is to simply maximize validation set performance, a large discrepancy between training and validation performance is considered a sign of overfitting. This heuristic is most commonly discussed in the context of early stopping regularization while training deep neural networks. Early stopping regularization is implemented by monitoring both training and validation performance in order to terminate the iterative, gradient base training optimization when the validation plateaus or begins to degrade.[29] In contrast, a common sentiment within the materials community is that the training AUC for a RF model will almost always be close to unity and that the best hyperparameters are those that correspond to the global maximum of the validation accuracy or F1 score. It is not clear from this work that basing model selection solely on the validation AUC accuracy or F1 score leads to a less generalizable model. However, from Fig. 2, simply maximizing the validation AUC leads to model explanations that one should not trust, since the random feature becomes prominent in the importance ranking. Interestingly, for both models, increasing the depth of the trees increases the impurity random feature importance.

Figure 3 plots a histogram of the random feature rank for the seven highest val_AUC models using the impurity and Shapley importance rankings. The Shapley importance of the random feature is roughly uniformly distributed across all possible importance values, with a minimum importance rank of third. Conversely, the distribution of the random feature impurity importance ranking is roughly normal with a mean value of 39.5. The tails of the

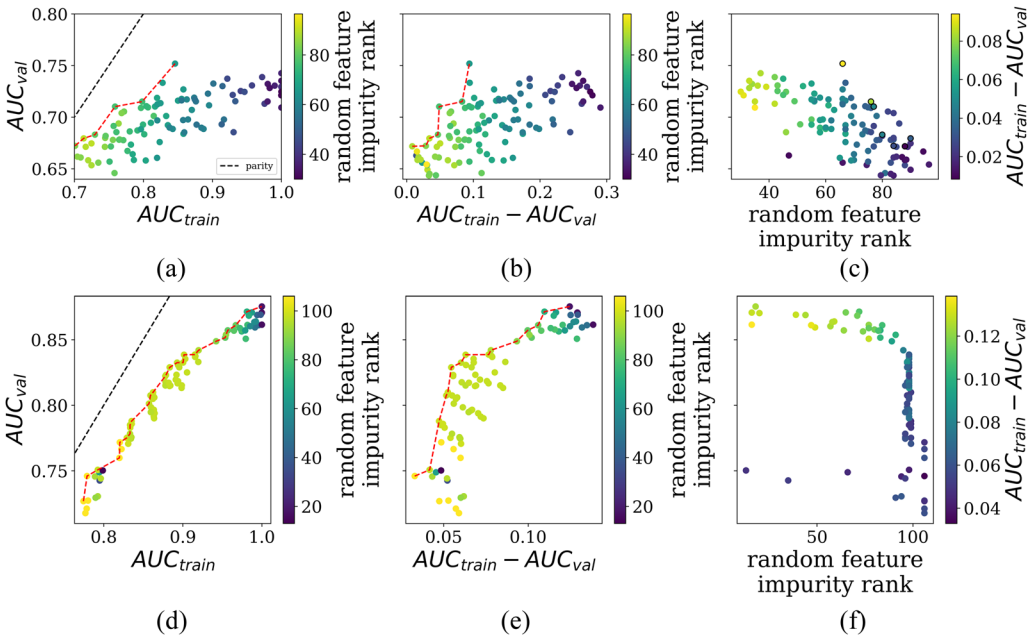

FIG. 2. Plots of the role of hyperparameter values on training statistics for the HEA model (a)–(c) and the MG model (d)–(f). Note that the y-axis for each set of plots is identical (AUC_{val}), but that the x-axis is different in each plot. The comparison of AUC_{train}, AUC_{val}, and random feature rank (color bar) is shown in (a)–(d). The comparison of AUC_{val}, (AUC_{train}–AUC_{val}), and the random feature rank (color bar) is shown in (b) and (e). Another comparison of AUC_{val}, the random feature rank, and (AUC_{train}–AUC_{val}) (color bar) is shown in (c) and (f).

distribution however are long, and the random feature can be found among the top eight most important features in a number of fivefold CV tests.

The red dashed lines in Figs. 2(a) and 2(b) highlight the Pareto optimal hyperparameters that simultaneously maximize the validation AUC and minimize the difference between the validation and training AUC. This Pareto optimal surface represents the trade-off in validation set performance and the baseline trustability of a FIR: selecting a hyperparameter sitting near the edge of the apparent high validation AUC plateau tends to deprioritize the random feature without substantially reducing generalization performance.

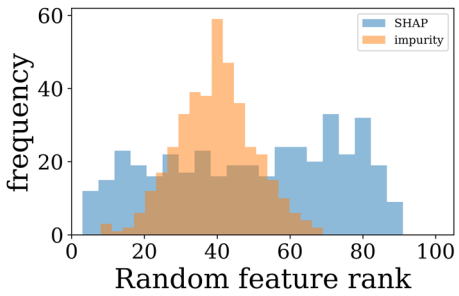

FIG. 3. Histogram of the random feature importance ranking over the seven models (350 fivefold CV iterations) with the highest Val_AUC. The random feature importance ranking forms a roughly uniform distribution for all ranks 4–90. The importance ranking using the impurity method has a near normal distribution with the random feature sometimes being ranked in the top eight features.

The model in Fig. 2(c) shows a bend in the validation AUC vs random feature importance near the Pareto optimal hyperparameters. Models near this bend contain the subset of hyperparameters that generate models with maximal predictive power and on average preserve their explainability.

We would like to point out that there are other methods for assessing the contributions of individual input features and training samples on the overall quality of a model. It is also known that feature importance ranking can be susceptible to distortions due to interactions between highly correlated input features—a common situation in many materials descriptor sets, e.g., Ref. 30. Common feature selection algorithms such as LASSO and least angle regression can mitigate this by retaining only informative features.[31,32] The FeaLect method attempts to select more relevant features than these methods by performing least angle regression on bootstrap samples of the training data, aggregating relevance scores for each feature across bootstrap samples, and performing automatic threshold selection on the feature relevance distribution to identify relevant features.[32,33] Unfortunately, there is not currently a stable release of FeaLect in the current sklearn or Waikato Environment for Knowledge Analysis (WEKA) toolboxes and nor is it a part of the standard Anaconda[18] download package. Meanwhile, LARS and LASSO are excellent for feature selection in linear regression but are not always suitable for solving materials science problems, where there may be a preference for nonlinear models.

Here, we have provided a rational path toward preserving model explainability while maximizing the predictive performance of RF models. A matminer feature set was used to describe an

open HEA dataset, RF models for the ability of alloys to form HEA or MG samples were generated, and their explainability through various feature importance rankings was explored. We show that focusing only on maximizing the validation AUC has the negative consequence of degrading trust in the ranking of feature importance. Specifically, we showed that using the default RF parameters resulted in a random feature being identified as the up to the fifth most important feature for the HEA model, depending upon the random seed used. By shifting the emphasis of hyperparameter optimization from a single parameter optimization to one that balances validation AUC and the differences in the training and validation AUC, we are able to suppress the importance of spurious random features, thus retaining an explainable model. Practically speaking, we suggest sorting the results of the hyperparameter study by (AUC_{train}–AUC_{val}), AUC_{Val}, and the ranking of the random feature, in that order. The user should then down select to select the highest capacity model in that subset of models.

Although the models explicitly investigated here were constrained to RF, there are a few important takeaways from this study that may generalize to the AI for materials community. First, while it is helpful to use the literature and online AI community forums to get tips on which hyperparameters are normally less important for a given AI model, proper model optimization requires rigorous variation of parameters that may be considered unimportant. Second, rather than using a single objective (i.e., validation AUC or F1 score) for model-hyper parameter tuning, best practices to preserve explainability would be to look for the subset of hyperparameters that balance training and validation accuracy and minimize the importance of spurious features. Finally, the ultimate selection of the set of hyperparameters to choose can be constrained via human intervention, e.g., selecting models that prioritize features known to mediate a property; however, doing so does carry with it the risk of biasing future models.

SUPPLEMENTARY MATERIAL

The code and data used to generate these data are provided in the supplementary material.

DATA AVAILABILITY

The data that support the findings of this study are available within the article and its supplementary material.

REFERENCES

[1]R. K. Vasudevan *et al.*, "Materials science in the artificial intelligence age: High-throughput library generation, machine learning, and a pathway from correlations to the underpinning physics," MRS Commun. **9**, 821 (2019).
[2]D. P. Tabor *et al.*, "Accelerating the discovery of materials for clean energy in the era of smart automation," Nat. Rev. Mater. **3**, 5–20 (2018).
[3]A. Aspuru-Guzik and K. A. Persson, Materials Acceleration Platform, Mission Innovation: Innovation Challenge No. 6, 2018.
[4]L. Zhang, D.-Y. Lin, H. Wang, R. Car, and E. Weinan, "Active learning of uniformly accurate interatomic potentials for materials simulation," Phys. Rev. Mater. **3**, 023804 (2019).
[5]A. G. Kusne *et al.*, "On-the-fly closed-loop materials discovery via Bayesian active learning," Nat. Commun. **11**, 5966 (2020).
[6]F. Ren *et al.*, "Accelerated discovery of metallic glasses through iteration of machine learning and high-throughput experiments," Sci. Adv. **4**, eaaq1566 (2018).
[7]Z. Xu, S. Wang, F. Zhu, and J. Huang, "Seq2seq fingerprint: An unsupervised deep molecular embedding for drug discovery," in *ACM-BCB 2017—Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics* (Association for Computing Machinery, Inc., 2017), Vol. 17, pp. 285–294.
[8]C. Molnar (2019). "Interpretable machine learning," Github. https://github.com/christophM/interpretable-ml-book.
[9]P. Raccuglia *et al.*, "Machine-learning-assisted materials discovery using failed experiments," Nature **533**, 73–76 (2016).
[10]W. Hume-Rothery, R. W. Smallman, and C. W. Haworth, *The Structure of Metals and Alloys* (Metals & Metallurgy Trust, 1969).
[11]L. Breiman, "Random forests," Int. J. Mach. Learn. Cybern. **45**, 5–32 (2001).
[12]L. Ward, A. Agrawal, A. Choudhary, and C. Wolverton, "A general-purpose machine learning framework for predicting properties of inorganic materials," npj Comput. Mater. **2**, 16028 (2016).
[13]O. Isayev *et al.*, "Universal fragment descriptors for predicting properties of inorganic crystals," Nat. Commun. **8**, 15679 (2017).
[14]J. Friedman, "Greedy function approximation: A gradient boosting machine," Ann. Stat. **29**, 1189–1232 (2001).
[15]A. Goldstein, A. Kapelner, J. Bleich, and E. Pitkin, "Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation," J. Comput. Graph Stat. **24**, 44–65 (2015).
[16]C. Olah *et al.*, *The Building Blocks of Interpretability* (Distill, 2018).
[17]C. K. H. Borg *et al.*, "Expanded dataset of mechanical properties and observed phases of multi-principal element alloys," Sci. Data **7**, 430 (2020).
[18]Certain commercial equipment, instruments, or materials (or suppliers or software, etc.) are identified in this paper to foster understanding. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.
[19]L. Ward *et al.*, "Matminer: An open source toolkit for materials data mining," Comput. Mater. Sci. **152**, 60–69 (2018).
[20]F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res. **12**, 2825–2830 (2011).
[21]A. Y.-T. Wang *et al.*, "Machine learning for materials scientists: An introductory guide toward best practices," Chem. Mater. **32**, 4954–4965 (2020).
[22]A. F. Zahrt, J. J. Henle, and S. E. Denmark, "Cautionary guidelines for machine learning studies with combinatorial datasets," ACS Comb. Sci. **22**, 586–591 (2020).
[23]T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning* (Springer, 2009), Vol. 2.
[24]C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: Illustrations, sources and a solution," BMC Bioinf. **8**, 25 (2007).
[25]L. S. Shapley, "A value for n-person games," Contrib. Theor. Game **2**, 307–317 (1953).
[26]S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, edited by I. Guyon *et al.* (Curran Associates, Inc., 2017), Vol. 30, pp. 4765–4774.
[27]A. Ghorbani and J. Zou, "Data Shapley: Equitable valuation of data for machine learning," in *Proceedings of the 36th International Conference on Machine Learning*, edited by K. Chaudhuri and R. Salakhutdinov (PMLR, 2019), Vol. 97, pp. 2242–2251.
[28]R. Jia *et al.*, "Towards efficient data valuation based on the Shapley value," in *Proceedings of Machine Learning Research*, edited by K. Chaudhuri and M. Sugiyama (PMLR, 2019), Vol. 89 pp. 1167–1176.
[29]I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).

[30]A. Mangal and E. A. Holm, "Applied machine learning to predict stress hotspots I: Face centered cubic materials," Int. J. Plast. **111**, 122–134 (2018).
[31]R. Tibshirani, "Regression shrinkage and selection via the Lasso," J. R. Stat. Soc., Ser. B **58**, 267–288 (1996).
[32]B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," Ann. Stat. **32**, 407–499 (2004).
[33]H. Zare, G. Haffari, A. Gupta, and R. R. Brinkman, "Scoring relevancy of features based on combinatorial analysis of Lasso with application to lymphoma diagnosis," BMC Genomics **14**, S14 (2013).