A Generalized Entropy-Based Framework for Modeling Performance of Machine Learning Algorithms

Raj Kishor Bisht[1*], Ayushi Bisht [2], Tushar Joshi[2], and Janvi Sirohi[2]

[1] School of Computing, Graphic Era Hill University, Dehradun, India
[2] Department of Mathematics, Graphic Era Hill University, Dehradun, India
*bishtrk@gmail.com

Abstract. The performance of classification algorithms depends on several factors such as training–test split ratio, number of features, class balance, and hyperparameters etc. These factors introduce uncertainty in the behavior and performance of machine learning (ML) algorithms. Hence, the performance of an algorithm is influenced by multiple sources of uncertainty. In this study, we propose a generalized entropy performance model that expresses the performance of an algorithm in terms of various entropy measures corresponding to these influential factors. First, we define distinct entropies representing data distribution, prediction diversity, confidence, and prediction error. Then, we develop a generalized regression model to predict key performance metrics (accuracy, precision, recall, and F-score) of four classification algorithms (KNN, SVM, Random Forest and Logistic regression) on the basis of these entropy measures. Four different datasets are considered for evaluation. Experimental results demonstrate that the proposed model effectively estimates different performance metric using these entropic measures.

Keywords: Entropy, Accuracy, Precision, Recall, F-score, Classification.

1 Introduction

Uncertainty is defined as the degree or variability that exists in a model's prediction, it reflects that how unpredictable or inconsistent a model's outputs are, these inconsistencies are either from incomplete model knowledge which is called epistemic uncertainty or from inherent randomness in data which is called aleatoric uncertainty. Even in advanced machine learning models and large datasets, these entropy-based uncertainties remain unavoidable. In recent years, this aspect gained significant attention, especially in safety-critical domains such as healthcare, autonomous systems, and scientific research, where understanding how confident a model is, is very important.

Traditional performance metrics like accuracy, precision, recall and F1 score indicate how well a model performs but these metrics fail to explain why the model performs that way. These metrics focus on results but will not explain the model's internal confidence and variability. To address this gap, we used Entropy based uncertainty quantification which provides a unified way to interpret both data level and model level uncertainties.

We present a Generalized Entropy Performance Model (GEPM), a comprehensive framework that connects multiple entropy components to the traditional performance metrics. The model works by identifying five key sources of entropy – from how skewed the dataset is, to how tentatively the classifier makes decisions, to where those decisions go wrong. The GEPM framework considers all relevant entropy measures for a particular metric, that is, it treats different performance metrics in different manners. For example, precision will be affected by different types of uncertainty than recall.

To validate the GEPM framework, experiments are conducted using four widely used classifiers K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR), and we apply these models on both balanced and imbalanced datasets. Also training and testing datasets are changed based on a set of information for a particular model. Regression analysis is performed to estimate the contribution of each entropy component.

The organization of the paper is as follows: Section II deals with the related research work in the direction of uncertainty of ML algorithms, In Section III, first we define different entropy measures, the GEPM is defined. Section IV deals with the outcomes of the experimental work and Section V deals with the conclusion of the study.

2 Literature Review

Uncertainty in machine learning is a point of major focus, as it directly affects the reliability of a model's prediction. The authors in their work [1] provided a structured review of different research work on uncertainty in ML. The authors [2,3,4] in their articles provided a detail introduction of uncertainty and the methods of measuring uncertainty. The author [5] discussed aleatoric and epistemic entropies in ML and methods of calculating these entropies. In case of healthcare, uncertainty in ML may have serious consequences, thus it is necessary to find the sources of uncertainty so that unsafe predictions can be avoided. The authors [6] discussed the sources of prediction uncertainty and method of implementing these metrics in applications. Sluijterman et al. [7] discussed evaluation of uncertainty in regression. In the work [8], a new nearest neighbor-based uncertainty measure is proposed that considers the ratio of labels in neighbor in addition to distance. This estimate is further applied to natural language processing tasks. Performance of ML algorithms depends on may factors like training-test split, sampling scheme etc, thus these factors also introduce uncertainty in prediction. The authors [9], in their work made a detail analysis of role of training -test split, random state in prediction accuracy of different classification algorithms.

Uncertainty in ML algorithms may be developed through the inherent data properties; thus, a proper method is needed to address it. In the work [10], the authors investigated and characterized the properties of ML models when the distribution of training and test data is different. The authors [11] in their work discussed uncertainty prediction using variogram model deriving necessary formulas for KNN method for forest inventories related data. Uncertainty in prediction may be there due to the variability in a data which is not considered in modelling ML process, thus model uncertainty may be something different measurement uncertainty. The authors [12] in their work differentiated these two kinds of uncertainties. Generally uncertainty is related to accuracy of an ML algorithm and minimization of uncertainty increases the accuracy in prediction. The authors [13] discussed different methods, functions used for measuring uncertainty in ML algorithms. In the work [14], the authors summarized three different approaches to evaluate uncertainty. Different ML algorithms have different output as per the nature of the algorithm; for example, the output of classification algorithms is generally a categorical variable and in case of regression it is a numerical variable. Thus, prediction output uncertainty may also depend on the nature or data type of the output variable. In the work [15], the authors discussed the uncertainty in the prediction of nominal variables.

In the work [16], the author discussed uncertainty in ML and different methods of reducing uncertainty. Uncertainty in prediction outcome also suggest the fairness of an algorithm. In the work [17], the authors defined fairness measure in terms of uncertainties. Different methods of measuring in uncertainty in deep learning algorithms are discussed in [18, 19]. AI-based models are nowadays quite popular in different fields. Uncertainty in their prediction has a great concern and its severity depends on the problem under interest. The authors [20] compared different approaches for measuring uncertainty in AI-based models of material science. A review of different approaches measuring uncertainty in ML is conducted in [21], similarly review of uncertainty in deep learning is conducted in [22].

3 Generalized Entropy Performance Model

We define a generalized model that can predict the performance metric M (Accuracy, Precision, Recall, F-score and AUC) in terms of entropy-based uncertainty measures derived from both data and model behavior. First, we define five different data driven entropy measures.

3.1 Data Entropy

Data entropy is defined as a measurement of uncertainty inherent in the dataset. It shows how balanced or imbalanced a given data set is. Let $C = \{c_1, c_2, \dots, c_k\}$ be the set of class labels, and $P(c_i)$ be the proportion of samples in class c_i, then we define data entropy (H_D) as follows:

$$H_D = -\sum_{i=1}^{k} P(c_i)\, log\, P(c_i) \tag{1}$$

$H_D = 0$, when all samples belong to one class (max imbalance). $H_D = \log k$, when all classes are equally likely (perfect balance). For normalization we divide data entropy by $\log k$.

$H_D = 0$, when all samples belong to one class (max imbalance). $H_D = \log k$, when all classes are equally likely (perfect balance). For normalization we divide data entropy by $\log k$.

3.2 Prediction Entropy

It represents uncertainty in model predictions, that is, how confident the classifier is overall. Let there are N observations and k classes. For each sample x_j let the model output class probabilities $P_j = \{p_{j1},\ p_{j2}, \ldots, p_{jk}\}$, then prediction entropy (H_{pred}) is defined as follows:

$$H_{pred} = \frac{1}{N}\sum_{j=1}^{N}\left(-\sum_{i=1}^{k} p_{ji}\, log\, p_{ji}\right) \tag{2}$$

Prediction entropy captures the average probabilistic confusion in predictions. $H_{pred} = 0$ shows that the model is very confident (sharp predictions) while $H_{pred} = \log k$, shows that the model is uncertain (uniform probabilities). For normalization we divide data entropy by $\log k$.

3.3 Local Entropy

It is algorithm dependent entropy. It captures local data uncertainty, that is, overlapping in defining classes using particular algorithm. For example, how mixed the neighbors of each point in KNN etc. For each sample x_j, and class i, let the local class probability is denoted by $P_j(c_i)$, then local entropy is defined as

$$H_{local} = \frac{1}{N}\sum_{j=1}^{N}\left(-\sum_{i=1}^{k} P_j(c_i), \log P_j(c_i)\ \right) \tag{3}$$

For models without local structure (e.g., SVM, LR), we can estimate local entropy using distances in feature space or probabilistic calibration neighborhoods.

3.4 Confidence Entropy

It represents spread of confidence scores for the predicted class. For each sample x_j, let the model's confidence score in the predicted class is $s_j = \max_i p_{ij}$. Since s_j is a continuous value in the range [0,1], we divide the confidence range [0,1] into m bins $b_1 = \left[0, \frac{1}{m}\right]$, $b_2 = \left[\frac{1}{m}, \frac{2}{m}\right], \ldots, b_m = \left[\frac{m-1}{m}, 1\right]$. Let f_i be the number of confidence values fall in each bin. We define the relative frequency $q(b_i) = \frac{f_i}{\sum_{i=1}^{m} f_i}$ and the confidence entropy (H_{conf}) is defined as

$$H_{conf} = -\sum_{i=1}^{m} q(b_i) \log q(b_i) \tag{4}$$

A low value of H_{conf} shows very certain or overconfident while a high value indicates that the model is inconsistently confident. For normalization we divide data entropy by $\log m$.

3.5 Error Entropy

In Error Entropy, we measure the uncertainty in a model's error pattern. We define a binary random variable e_j for each sample x_j as

$$e_j = \begin{cases} 1 & if \quad x_j\ is\ misclassified \\ 0 & if \quad x_j\ is\ correctly\ classified \end{cases}$$

Suppose we have N total samples in the test set. Let $N_{mc} =$ number of misclassified samples and $N_c =$ number of correctly classified samples, then the empirical probabilities are $p_{e(1)} = \frac{N_{mc}}{N}$ and $p_{e(0)} = \frac{N_c}{N}$. Then the error entropy (H_{error}) is defined as

$$H_{error} = -\sum_{i\in\{0,1\}} p_{e(i)} \log p_{e(i)} \tag{5}$$

We propose the Generalized Entropy–Performance Model (GEPM) as:

$$M = \alpha_0 + \sum_{i=1}^{5} \alpha_i a_i (1 - H_i) + \epsilon \tag{6}$$

where, $\alpha_i \in [0{,}1]$ represent metric-specific coefficients, ϵ is random noise or model residual, the term $(1 - H_i)$ reflects that higher uncertainty typically reduces performance. , $a_i = 1\ or\ 0$ depends on active component nature of a particular entropy on a given performance measure as Each metric (Accuracy, Precision, Recall, F1, AUC) has its own active entropy components. Table 1 shows the active component vector a_i for different performance metrics.

Table 1. Active components in different performance measures and their vectors.

Metric	Active Components	Component Vector
Accuracy	H_D , H_{pred} , H_{error}	[1,1,0,0,1]
Precision	H_D , H_{pred} , H_{conf}	[1,1,0,1,0]
Recall	H_D , H_{pred}, H_{local}	[1,1,1,0,0]
F1	H_D , H_{pred}, H_{local} , H_{conf}	[1,1,1,1,0]
AUC	H_D , H_{pred} , H_{conf}, H_{error}	[1,1,0,1,1]

For example, to model accuracy, the proposed model will be defined as

$$M_{accuracy} = \alpha_0 + \alpha_1(1 - H_D) + \alpha_2\left(1 - H_{pred}\right) + \alpha_3(1 - H_{error}) + \epsilon \tag{7}$$

Similarly, other performance measures can be defined using the component vector given in Table 1.

4 Experimental results

We have considered four different datasets (Diabetes [22], Wine [23], Spam [24], Titanic [25]) available at Kaggle for the evaluation of our proposed model. Performances of four different classifications algorithms K-nearest neighbor (KNN), Support vector classification (SVC), Random Forest (RF), and Logistic Regression (LR) are evaluated on these datasets and entropies are calculated, then for each performance measure and for each dataset, multiple regression is applied to model performance on the basis of different entropies.

For example, Table 2 shows the values of 5 random data items of different entropies, accuracy and predicted accuracy by the regression model for 'Spam dataset' and Random forest algorithm. Table 3 shows the values of 5 random data items of different entropies, accuracy and predicted accuracy by the regression model for 'Wine dataset' and SVC.

Table 2. Different entropies and accuracy for spam dataset using random forest algorithm

H_D	H_{pred}	H_{error}	**Accuracy**	**Accuracy(P)**
0.967	0.198	0.958	0.874	0.867
0.967	0.199	0.989	0.866	0.867
0.967	0.224	0.978	0.871	0.875
0.967	0.263	0.970	0.867	0.886
0.967	0.215	0.988	0.881	0.872

Table 3. Different entropies and accuracy for wine dataset using SVC algorithm

H_D	H_{pred}	H_{error}	**Accuracy**	**Accuracy(P)**
1.568	0.290	0.989	0.407	0.449
1.566	1.055	0.994	0.389	0.410
1.567	0.293	0.991	0.400	0.448
1.565	0.325	0.994	0.389	0.446
1.566	0.349	0.992	0.397	0.445

Table 4. Different entropies and accuracy for diabetes dataset using KNN algorithm

H_D	H_{pred}	H_{error}	**Accuracy**	**Accuracy(P)**
0.934	0.000	1.000	0.690	0.703
0.933	0.000	0.997	0.708	0.694
0.933	0.000	0.994	0.703	0.698
0.932	0.000	0.997	0.732	0.691
0.933	0.000	0.997	0.706	0.695

Table 4 shows the values of 5 random data items of different entropies, accuracy and predicted accuracy by the regression model for ‘Diabetes dataset’ and KNN. Table 5 shows the values of 4 random data items of different entropies, accuracy and predicted accuracy by the regression model for ‘Diabetes dataset’ and KNN.

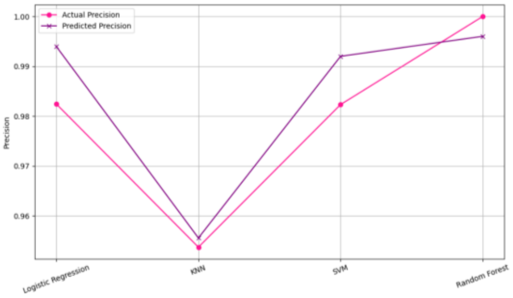

Fig. 1. Actual precision and predicted precision for wine dataset

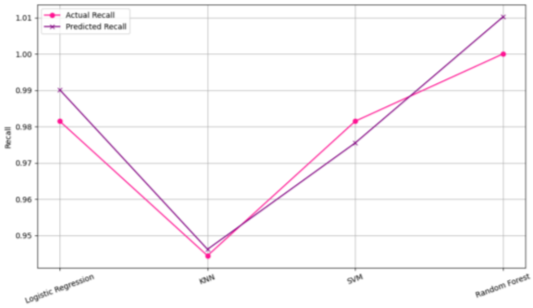

Fig. 2. Actual recall and predicted recall for wine dataset

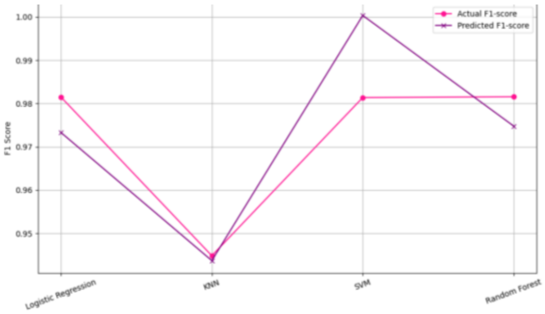

Fig. 3. Actual F-Score and predicted F-Score for wine dataset

Fig. 1, Fig. 2, Fig. 3 and Fig. 4 show the graphical representation of comparative performance scores of different algorithms and predicted performance for ‘Wine dataset’ with respect to the performance measures: ‘precision’, ‘recall’, ‘F-score’ and accuracy, respectively.

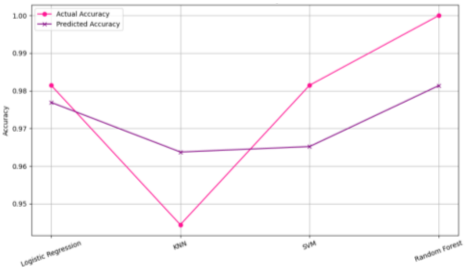

Fig. 4. Actual accuracy and predicted accuracy for wine dataset

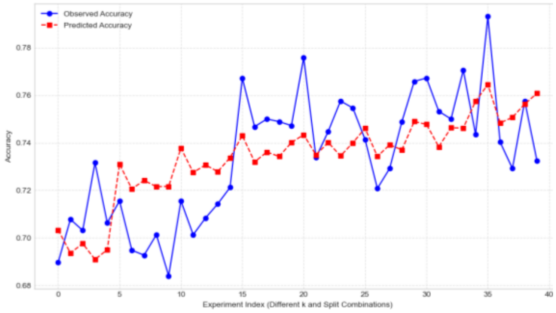

Fig. 5. Observed and predicted accuracy for diabetes dataset using KNN

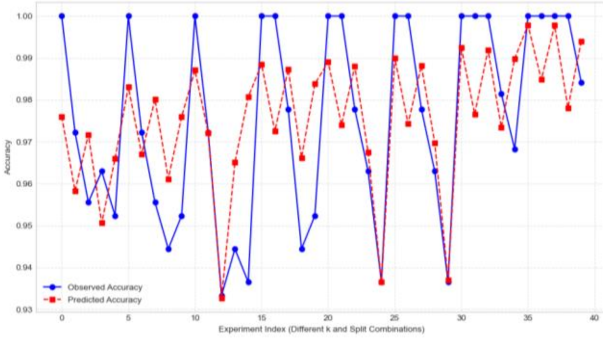

Fig. 6. Observed and predicted accuracy for wine dataset using KNN

Fig. 5, Fig. 6, Fig. 7 and Fig. 8 shows the observed and predicted accuracies of KNN algorithm for Diabetes, Wine, Spam, and Titanic datasets respectively. Different values of K and different training-test splits are considered.

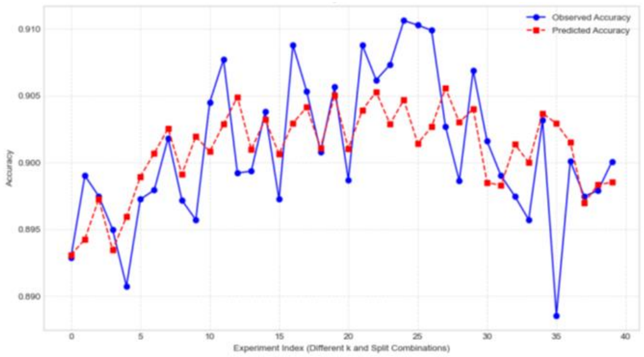

Fig. 7. Observed and predicted accuracy for spam dataset using KNN

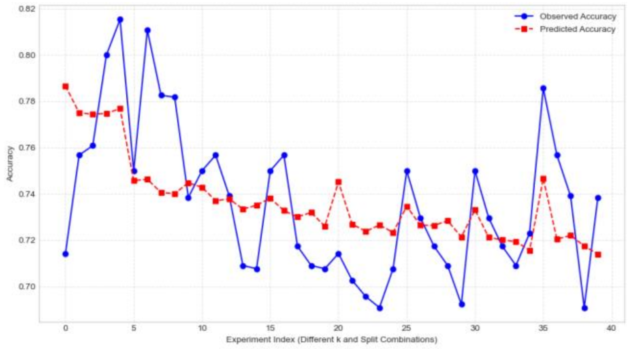

Fig. 8. Observed and predicted accuracy for Titanic dataset using KNN

We have calculated root mean square error (RMSE) to check the performance of regression models. RMSE for different algorithms and datasets is shown in Table 5. From Table 5, we observe that RMSE values are too low indicating a good fit of regression model. Thus, we can model different performance measures in terms of various uncertainty measures in the form of entropies.

Table 5. RMSE values for different datasets and algorithms

Dataset	**Classification Algorithm**			
	KNN	**RF**	**SVC**	**LR**
Diabetes	0.019	0.021	0.024	0.005
Wine	0.017	0.019	0.078	0.009
Spam	0.004	0.005	0.029	0.001
Titanic	0.027	0.033	0.027	0.004

5 Conclusion

Performance of an algorithm depends on many factors. The present work is an effort to model performance measures of an algorithm in terms of uncertainty introduced by different inherent or algorithmic factors. First, we defined such uncertainty measures and then a regression model is used to predict the performance on the basis of different entropy measures. Experiment is conducted on different datasets and using different algorithms. We found that the regression model is working well and hence performance of an algorithm can be assessed through different entropies. The proposed work is limited to classification algorithms, it can be extended to other machine learning algorithms. Further, individual modelling of a performance measure can be conducted for in-depth understanding of an algorithm.

References

1. Fakour, F., Mosleh, A., Ramezani, R.: A structured review of literature on uncertainty in machine learning and deep learning. arXiv (2024). https://doi.org/10.48550/arXiv.2406.00332
2. Payong, A.: Introduction to uncertainty in machine learning models: Concepts and methods – Part 1. Paperspace Blog (2023). https://blog.paperspace.com
3. Payong, A.: A comprehensive introduction to uncertainty in machine learning. iMerit Blog (2023). https://blog.imerit.net
4. Brownlee, J.: A gentle introduction to uncertainty in machine learning. Machine Learning Mastery (2019). https://machinelearningmastery.com
5. Hüllermeier, E., Waegeman, W.: Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. Machine Learning 110, 457–506 (2021). https://doi.org/10.1007/s10994-021-05946-3
6. Chua, M., Kim, D., Choi, J., et al.: Tackling prediction uncertainty in machine learning for healthcare. Nature Biomedical Engineering 7, 711–718 (2023). https://doi.org/10.1038/s41551-022-00988-x
7. Sluijterman, L., Cator, E., Heskes, T.: How to evaluate uncertainty estimates in machine learning for regression? Neural Networks 173, 106203 (2024). https://doi.org/10.1016/j.neunet.2024.106203
8. Hashimoto, W., Kamigaito, H., Watanabe, T.: Efficient nearest neighbour-based uncertainty estimation for natural language processing tasks. In: Findings of the Association for Computational Linguistics: NAACL 2025, pp. 4350–4366. Association for Computational Linguistics (2025).
9. Bisht, R.K., Bisht, I.P.: Investigation of the role of test size, random state, and dataset in the accuracy of classification algorithms. In: Sharma, H. et al. (eds.) Communication and Intelligent Systems. Springer, Singapore (2024). https://doi.org/10.1007/978-981-99-2100-3_55
10. Incorvaia, G., Hond, D., Asgari, H.: Uncertainty quantification of machine learning model performance via anomaly-based dataset dissimilarity measures. Electronics 13(5), 939 (2024). https://doi.org/10.3390/electronics13050939

11. Kim, H.-J., Tomppo, E.: Model-based prediction error uncertainty estimation for k-NN method. Remote Sensing of Environment 104, 257–263 (2006). https://doi.org/10.1016/j.rse.2006.04.009
12. Huber, M., Müller, T., Schmitt, R.: Differentiation between model error and measurement uncertainty in machine learning modeling for measurement processes. Measurement: Sensors 38, 101608 (2025). https://doi.org/10.1016/j.measen.2024.101608
13. Shirmohammadi, S., Amiri, M.H., Al Osman, H.: Uncertainty as a predictor of classification accuracy in machine learning-assisted measurements [Measurement Methodology]. IEEE Instrumentation & Measurement Magazine 27(7), 37–45 (2024). https://doi.org/10.1109/MIM.2024.10700740
14. Forbes, A.B.: Approaches to evaluating measurement uncertainty. International Journal of Metrology and Quality Engineering 3(1), 1–9 (2012).
15. Bilson, S., Cox, M., Pustogvar, A., Thompson, A.: A metrological framework for uncertainty evaluation in machine learning classification models. arXiv preprint (2025). https://doi.org/10.48550/arXiv.2504.03359
16. Ramdani, F.: Machine learning, measuring uncertainty, and forecasting. In: Data Science: Foundations and Hands-on Experience, pp. 1–20. Springer, Singapore (2025). https://doi.org/10.1007/978-981-96-4683-8_6
17. Kuzucu, S., Cheong, J., Gunes, H., Kalkan, S.: Uncertainty as a fairness measure. Journal of Artificial Intelligence Research 81, 307–335 (2024). https://doi.org/10.1613/jair.1.16041
18. Alqarafi, A., Batool, H., Abbas, T., Janjua, J.I., Ramay, S.A., Ahmed, M.: Estimating uncertainty in deep learning methods and applications. In: Proceedings of the 2024 International Conference on Computer and Applications (ICCA), pp. 1–6. IEEE, Cairo (2024). https://doi.org/10.1109/ICCA62237.2024.10928030
19. Salim, S., Jayasudha, J.S.: A literature survey on estimating uncertainty in deep learning models: Ensuring safety in intelligent systems. In: Proceedings of ICCSC 2023, pp. 1–5 (2023). https://doi.org/10.1109/ICCSC56913.2023.10143025
20. Tavazza, F., DeCost, B., Choudhary, K.: Uncertainty prediction for machine learning models of material properties. ACS Omega 6(47), 32431–32440 (2021). https://doi.org/10.1021/acsomega.1c03752
21. Tyralis, H., Papacharalampous, G.: A review of predictive uncertainty estimation with machine learning. Artificial Intelligence Review 57, 94 (2024). https://doi.org/10.1007/s10462-023-10698-8
22. Mena, J., Pujol, O., Vitrià, J.: A survey on uncertainty estimation in deep learning classification systems from a Bayesian perspective. ACM Computing Surveys (CSUR) 54(9), Article 193, 1–35 (2021). https://doi.org/10.1145/3477140