A Comparative Study of Machine Learning Algorithms on Datasets of Varying Sizes

Xiaoting Huang[1], Xuelian Xi[2], Siqi Wang[3], Zahra Sadeghi[4], Asif Samir[5], Stan Matwin[6]

[1] Department of Electrical and Computer Engineering, Dalhousie University, Halifax, Nova Scotia
[2] Department of Computer Engineering, Dalhousie University, Halifax, Nova Scotia

Abstract. This research investigates the nuanced performance dynamics of machine learning algorithms across datasets of varying sizes. Employing three distinct datasets characterized by different sample sizes, we implement four machine learning algorithms—Logistic Regression, K Nearest Neighbors (KNN), Naïve Bayes, and Random Forest. This paper delineates the application methodologies, hyperparameter tuning procedures, and the resultant performance outcomes. Our findings reveal that Logistic Regression and Naïve Bayes exhibit relatively minor sensitivity to dataset size fluctuations. In stark contrast, KNN is markedly influenced by dataset size, showcasing significant performance variations. Notably, Random Forest demonstrates consistently superior performance across the four machine learning models considered, particularly excelling with the largest datasets. The implications of these observations are discussed, contributing to a comprehensive understanding of the interplay between machine learning algorithms and dataset characteristics. The effectiveness of a machine learning algorithm does not rely solely on expanding the dataset size. Our research indicates that, once a certain threshold is reached, merely increasing the dataset size does not proportionally improve performance[1].

Keywords: Supervised Learning, Data size, Random Forest, Logistic regression, KNN, Naïve Bayes

1 Introduction

The field of machine learning is at the forefront of modern computing advances, providing transformative solutions for a wide range of fields. The core of its capabilities lies in the ability to adapt different algorithms and apply them to different datasets. This study titled "A Comparative Analysis of Machine Learning Algorithm Across Diverse Datasets" embarks on a journey of discovery to understand the complexity and performance of various machine learning algorithms when applied to different datasets.

[1] https://github.com/XiaotingH/ML-model-analysis-across-diverse-datasets

In the ever-expanding field of data, choosing the right machine learning algorithm becomes crucial. Several factors, including the dataset size, complexity, and the nature of the features involved, often influence this decision. Recognizing this, our research embarks on a thorough comparison of four well-known machine learning algorithms: Random Forest (RF), Naïve Bayes (NB), logistic regression (LR), and K-nearest neighbors (KNN). Each of these algorithms has a unique approach to learning from data for three different datasets: adult income, airline satisfaction, and credit approval.

Each dataset presents a unique set of challenges and characteristics. For example, the Adult Income dataset reflects variables that influence income levels, while the Airline Satisfaction dataset provides insights into customer satisfaction parameters in the airline industry. On the other hand, the credit approval dataset contains factors that influence credit card approval decisions. This diversity of datasets allows for a robust analysis of how dataset characteristics affect the performance and applicability of each algorithm.

The primary objective of this study is twofold: to evaluate the efficacy of these algorithms across diverse datasets and to establish a framework for guiding the selection of the most appropriate algorithm based on the dataset's characteristics. By examining the interactions between algorithms and datasets, this study aims to reveal the decisions that shape the future of machine learning applications.

Through this comparative analysis, we aspire to contribute to a broader understanding of machine learning algorithms, fostering more informed algorithmic choices and application approaches in different real-world scenarios.

2 Literature Review

In recent studies, researchers persist in exploring and comparing multiple machine learning models across various scenarios and datasets. Some of the study such as [1] focused on evaluating the performance of various models on a small dataset. Khanam, J.J. et al. [1] applied seven different ML algorithms on diabetes prediction, including Decision Tree (DT), KNN, Random Forest (RF), Naïve Bayes (NB), Adaptive Boosting, Logistic Regression, SVM to predict and evaluate their performance. The dataset used is Pima Indian diabetes datasets (PIDD) with 9 attributes and 768 observations. Most of the models are intuitively provide over 70% accuracy. Whereas the models built with Logistic regression and SVM deliver better performance, the accuracy achieve 78.78% and 77.71% respectively. However, Dris, A.B.et al. [2]. focus more on investigating the impact of model performance on small datasets with different number of features and instances. Among the three employed machine learning algorithms, Decision Tree, SVM and Naïve Bayes, SVM delivered overall the best classification accuracy and Naïve Bayes has worse performance.

Concurrently, scholars are directed towards the investigation of large datasets, given their inherent susceptibility to overfitting. Addressing the primary concern associated with large datasets, the exploration of diverse machine learning models to identify those that effectively mitigate the impact of overfitting emerges as a pivotal

topic within the research field. Catal, C.et al. [3] used five public NASA datasets from PROMISE repository to construct and evaluate software fault prediction model on the dimensions of dataset size and features selection. According to the study, Random Forest was concluded to deliver the higher prediction accuracy for large dataset, and Naïve Bayes provides best performance on the small dataset.

In a comparable study, Althnian, et al. seek to explore how the performance of commonly used supervised machine learning models is influenced by the size of the dataset [4]. Their investigation involves the utilization of two substantial datasets and three subsets obtained by reducing the size of the original datasets. According to the experiment result, AB and NB emerged as the most robust models for limited size of dataset, whereas Decision Tree exhibited the poorest performance. Additionally, the results imply the overall model performance is influenced more by the distribution of dataset rather than its size. Machine learning techniques have found extensive applications in public health research and the analysis of biomedical data. They are employed for diverse purposes, such as predicting disease outcomes and assessing the severity of diseases through clinical diagnosis. In reference to skin infection analysis and forecasting, one study evaluates the performance of leading machine learning classifier models [13], while another research focuses on the classification of breast cancer using similar techniques [17]. Additionally, machine learning models play a crucial role in predicting virus generation based on symptoms and test reports. Research in this domain, exemplified by [14] for Hepatitis viruses and [18] for COVID-19, underscores the significant threat these viruses pose to human life. Consequently, it is imperative to dedicate efforts to developing efficient and high-precision models in the medical field. Such advancements can contribute significantly to improving social healthcare services and enhancing the overall cure rate of diseases.

In this study, three different dataset sizes related to classification problems from various domains are chosen. Four frequently employed supervised machine learning models are utilized to assess their performance.

3 Method and Experiments

3.1 Datasets

In the context of this project, three datasets have been chosen from the UC Irvine repository and Kaggle website. Among them, two are sizable datasets — dataset 2[2] and dataset 3[3] — each encompassing over a thousand instances with varying numbers of features. Specifically, dataset 3 is focused on predicting passengers' satisfaction based on their inflight habits and personal information, while dataset 2 is geared towards predicting income ranges, distinguishing between those

2 Barry Becker, and Ronny Kohavi. “Adult Dataset.” UCI Machine Learning Repository, https://archive.ics.uci.edu/dataset/2/adult.

3 T. J. Klein. “Airline Passenger Satisfaction Dataset.” Kaggle, https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction/data

above and below five thousand. On the other hand, dataset 1[4] is comparatively smaller, consisting of approximately 700 instances and a similar number of features. This dataset pertains to the prediction of credit approval decisions.

Table 1. Datasets information.

	Dataset Name	**Number of features**	**Number of Instances**
Dataset1	Credit Approval	16	690
Dataset2	Adult	14	32561
Dataset3	Airlines	25	103904

3.2 Data preprocessing

Normalization

The datasets demonstrate a noticeable presence of skewed and imbalanced features to varying extents. Specifically, for strongly skewed features like "captain loss" and "captain gain," within the income prediction dataset, Min-Max scaling is the preferred standardization method due to its robustness to outliers. Conversely, for features with a more bell-shaped distribution, Z-score scaling is under consideration.

Feature encoding

In the preprocessing phase, two primary encoding techniques are employed, which are label encoding and ordinal encoding. Label encoding is selected in this task work due to its simplicity and memory usage reduction. However, when wording with features exhibiting a meaningful order, such as the in-flight class and airline passengers' satisfaction prediction dataset or the education level in an income prediction dataset, it is preferrable to apply ordinal encoding.

Feature selection

In the phase of data preprocessing, meticulous scrutiny was applied to the features, leading to the removal of those exhibiting high correlations. For instance, in Dataset 3, a heatmap was generated to visually assess correlations among features. Subsequently, a detailed comparison and data analysis were conducted to identify and eliminate redundant features based on the observed correlations.

4 J. R. Quinlan. "Credit Approval Dataset." UCI Machine Learning Repository, https://archive.ics.uci.edu/dataset/27/credit+approval.

3.2 Model Construction

Logistic Regression Algorithm

Logistic Regression, a predictive analytics and classification algorithm, is optimally suited for scenarios where the response variable is categorical. This algorithm is particularly adept at binary classification tasks, yielding dichotomous outcomes like Yes/No, Positive/Negative, or 1/0.

In our research, we applied the initial Logistic Regression model to three distinct datasets. To refine the model's accuracy, we engaged in hyperparameter tuning using GridSearchCV, a method that performs an exhaustive search over a defined parameter grid to identify the combination that delivers the best cross-validation score. The hyperparameters we adjusted included 'C' (regularization strength), with values ranging from 0.001 to 10; 'penalty' (norm used in penalization), with options of 'l1', 'l2', and 'elasticnet'; 'solver' (optimization algorithm), with choices like 'newton-cg', 'lbfgs', 'liblinear', 'sag', and 'saga'; and 'tol' (tolerance for stopping criteria), with levels set at 1e-4, 1e-3, and 1e-2.

The optimization process yielded distinct optimal parameters for each dataset. For the adult dataset, the best parameters were {'C': 10, 'penalty': 'l2', 'solver': 'liblinear', 'tol': 0.0001}. In the case of the credit dataset, the optimal parameters were {'C': 1, 'penalty': 'l1', 'solver': 'liblinear', 'tol': 0.001}. For the airline dataset, the best parameters identified were {'C': 10, 'penalty': 'l1', 'solver': 'liblinear', 'tol': 0.001}. These results highlight the critical role of precise hyperparameter tuning in enhancing the performance of Logistic Regression models across different datasets.

Naïve Bayes Algorithm

Naive Bayes is a versatile and efficient probabilistic supervised machine learning algorithm widely employed for classification tasks. The fundamental concept behind this algorithm involves calculating the probability of instances belonging to a specific class using Bayes' theorem. It learns the prior probability and likelihood of each feature occurring in each class, subsequently making predictions based on these probability calculations. One of the most significant advantages of Naive Bayes is its lack of hyperparameters, coupled with computational efficiency, making it a favorable choice for handling large datasets. However, it is important to note that the "naive" assumption, which assumes variables are conditionally independent and follow specific distributions, can be limiting. This limitation becomes apparent when the assumption is violated, particularly in real-world scenarios where ideal conditions may not always apply.

In the context of this study, three Naive Bayes models were constructed for different datasets, each assuming features follow Gaussian, Multinomial, and Bernoulli distributions, respectively. The datasets encompass a variety of predictor types, including continuous, discrete, and Boolean features, each corresponding to a well-suited distribution. Consequently, the key to constructing effective Naive Bayes models lies in determining the most suitable distribution type for each dataset scenario.

K-Nearest Neighbors Algorithm

K-Nearest Neighbors (KNN) is a resilient and intuitive supervised machine learning algorithm that offers a practical and uncomplicated method for handling both classification and regression tasks effectively. As the name suggests, KNN employs a neighborhood-centric learning strategy, making predictions based on the characteristics of nearby neighbors. The parameter “k”, representing the number of neighbors taken into consideration, plays a crucial role in shaping the predictive outcome.

In this task, a KNN model was constructed, utilizing the Euclidean distance as the metric to locate the surrounding k neighbors. To determine the optimal number of neighbors for the model, an exhaustive search ranging from one to fifty with a step of one was conducted. Subsequently, stratified k-fold cross-validation was applied to the dataset, and the cross-validation scores were obtained. The optimal number of neighbors for this study was determined by selecting the parameter "k" that achieved the highest cross-validation score. It's noteworthy that in this investigation, the optimal K nearest neighbors were chosen from a range spanning 1 to 50. In the experiment, the optimum parameters (k values) were determined as 4, 32, and 6 for datasets 1, 2, and 3, respectively.

Random Forest (RF)

Random Forest is a broadly utilized and accepted ensemble learning approach employed for tasks involving both classification and regression in the domain of supervised learning. As a member of the decision tree-based methods family, it is widely acknowledged for its exceptional accuracy, robustness, and efficacy when dealing with large datasets that encompass numerous features. Ensemble learning means that it combines the predictions from multiple models to make a final prediction. In the case of Random Forest, these models are decision trees. Random Forest uses a technique called bagging, where multiple decision trees are trained on different subsets of the training data. Each subset is created by randomly sampling with replacement (bootstrap sampling) from the original dataset. For classification tasks, Random Forest typically uses a majority voting scheme, where the class predicted by most trees is selected as the final prediction. For regression tasks, it averages the predictions of individual trees.

For this classification task, Random Forest was selected as one of the comparative learning algorithms. The scikit-learn library was utilized for classification using RF. Afterward, we performed a contrast between Random Search and Grid Search to pinpoint various hyperparameters. These parameters encompassed the forest's number of trees, the minimum samples needed to split an internal node, the minimum samples needed at a leaf node, and the number of features considered for the optimal split, also known as the maximum tree depth. Finally, the determined optimal hyperparameters with the best performance on validation set, were applied to fit the test dataset. The optimal hyperparameters, which exhibited the best performance in both random and grid searches across all three datasets, are presented in Table 5. Corresponding performance metrics evaluation are detailed in Table 2, Table 3 and Table 4.

Table 2. Random search versus grid search in dataset1.

Dataset1: Credit	Accuracy	Precision (MacroAverage)	Recall (MacroAverage)	F1-Score (MacroAverage)
Random Search	**0.8913**	0.89	0.89	0.89
Grid Search	0.8696	0.87	0.86	0.87
		"Precision (WeightedAverage)"	**"Recall (WeightedAverage)"**	**"F1-score (WeightedAverage)"**
Random Search	**0.8913**	0.89	0.89	0.89
Grid Search	0.8696	0.87	0.87	0.87

Table 3. Random search versus grid search in dataset2.

Dataset2: Adult	Accuracy	Precision (MacroAverage)	Recall (MacroAverage)	F1-Score (MacroAverage)
Random Search	**0.8655**	0.84	0.77	0.8
Grid Search	0.8661	0.84	0.77	0.8
		"Precision (WeightedAverage)"	**"Recall (WeightedAverage)"**	**"F1-score (WeightedAverage)"**
Random Search	**0.8655**	0.86	0.87	0.86
Grid Search	0.8661	0.86	0.87	0.86

Table 4. Random search versus grid search in dataset3.

Dataset1: Airline	Accuracy	Precision (MacroAverage)	Recall (MacroAverage)	F1-Score (MacroAverage)
Random Search	**0.9623**	0.96	0.96	0.96
Grid Search	0.9620	0.96	0.96	0.96
		"Precision (WeightedAverage)"	**"Recall (WeightedAverage)"**	**"F1-score (WeightedAverage)"**
Random Search	**0.9623**	0.96	0.96	0.96
Grid Search	0.9620	0.96	0.96	0.96

Table 5. Random Forest hyperparameter selection for all datasets.

		Min_sample_ leaf	Min_sample_ split	n_estimaters	Max_depth	Accuracy
Dataset1	Random Search	1	2	100	None	**0.8913**
	Grid Search	2	5	100	None	0.8696
Dataset2	Random Search	4	5	100	None	**0.8655**
	Grid Search	2	2	100	None	0.8611
Dataset3	Random Search	1	5	100	None	**0.9623**
	Grid Search	1	2	150	None	0.9620

4 Results

Table 6 through Table 11 showcase the assessed performance of LR, KNN, NB, and RF algorithms using performance metrics of accuracy, precision, recall, and F1-score. Notably, LR exhibited consistent performance across the three datasets, with its best performance observed in the credit approval prediction dataset, the smallest among the three with multiple multiclass features. This suggests that LR is less influenced by dataset size and is more affected by the types of features and their quantity. In contrast, Random Forest demonstrated the highest overall accuracy among the four models, particularly excelling in predicting airline passengers' satisfaction. This dataset, characterized by multiclass features and a mix of continuous values, showcased RF's superior performance. Naïve Bayes (NB) yielded varying accuracy levels across three distinct types, strongly influenced by feature distribution and types. Similar to RF, NB performed exceptionally well in predicting airline passengers' satisfaction. Conversely, KNN exhibited the most fluctuating accuracy, reaching over 90 percent in the airline dataset but only delivering 73 percent accuracy in the credit dataset. This indicates that KNN is the most sensitive model among the four, responding differently to variations in dataset characteristics.

Simultaneously, due to the presence of imbalances and skewness in the selected datasets, both macro-average, from equation (1) to equation (3), and weighted-average evaluation metrics, from equation (4) to (6), are computed. This approach offers insights into the imbalance within the datasets. Upon comparing the two averaging methods, it is observed that only the adult income prediction dataset exhibits a significant difference between macro and weighted average evaluation metrics. This discrepancy underscores that the adult income prediction dataset is the most imbalanced dataset among the three presented datasets.

Table 6. Performance evaluation of LR, KNN, NB, RF, dataset Credit based on macro average.

Dataset1: Credit	Accuracy	Precision (MacroAverage)	Recall (MacroAverage)	F1-Score (MacroAverage)
Logistic Regression	0.85	0.86	0.86	0.85
KNN	0.73	0.74	0.70	0.70
Naive Bayes - GaussianNB	0.82	0.83	0.80	0.80
Naive Bayes - BernoulliNB	0.83	0.85	0.81	0.82
Naive Bayes - MultinomialNB	0.72	0.72	0.72	0.72
Random Forest	**0.90**	0.90	0.90	0.90

Table 7. Performance evaluation of LR, KNN, NB, RF, dataset Adult based on macro average.

Dataset2: Adult	Accuracy	Precision (MacroAverage)	Recall (MacroAverage)	F1-Score (MacroAverage)
Logistic Regression	0.84	0.80	0.74	0.76
KNN	0.84	0.79	0.74	0.76
Naive Bayes - GaussianNB	0.72	0.70	0.77	0.69
Naive Bayes - BernoulliNB	0.79	0.72	0.76	0.74
Naive Bayes - MultinomialNB	0.79	0.71	0.59	0.60
Random Forest	**0.87**	0.84	0.77	0.80

Table 8. Performance evaluation of LR, KNN, NB, RF, dataset Airline based on macro average.

Dataset3: Airline	Accuracy	Precision (MacroAverage)	Recall (MacroAverage)	F1-Score (MacroAverage)
Logistic Regression	0.87	0.87	0.87	0.87
KNN	0.93	0.94	0.93	0.93
Naive Bayes - GaussianNB	0.86	0.86	0.86	0.86
Naive Bayes - BernoulliNB	0.78	0.77	0.78	0.77
Naive Bayes - MultinomialNB	0.82	0.82	0.81	0.81
Random Forest	**0.96**	0.96	0.96	0.96

Table 9. Performance evaluation of LR, KNN, NB, RF, dataset Credit based on weighted average.

Dataset1: Credit	Accuracy	Precision (WeightedAverage)	Recall (WeightedAverage)	F1-Score (WeightedAverage)
Logistic Regression	0.85	0.86	0.85	0.85
KNN	0.73	0.74	0.73	0.72
Naive Bayes - GaussianNB	0.82	0.82	0.82	0.81
Naive Bayes - BernoulliNB	0.83	0.84	0.83	0.83
Naive Bayes - MultinomialNB	0.72	0.72	0.72	0.72
Random Forest	**0.90**	0.90	0.90	0.90

Table 10. Performance evaluation of LR, KNN, NB, RF, dataset Adult, based on weighted average.

Dataset2: Adult	Accuracy	Precision (WeightedAverage)	Recall (WeightedAverage)	F1-Score (WeightedAverage)
Logistic Regression	0.84	0.84	0.84	0.84
KNN	0.84	0.83	0.84	0.83
Naive Bayes - GaussianNB	0.72	0.83	0.72	0.74
Naive Bayes - BernoulliNB	0.79	0.82	0.79	0.80
Naive Bayes - MultinomialNB	0.79	0.76	0.79	0.75
Random Forest	**0.86**	0.86	0.86	0.85

Table 11. Performance evaluation of LR, KNN, NB, RF, dataset Airline, weighted average.

Dataset3: Airline	Accuracy	Precision (WeightedAverage)	Recall (WeightedAverage)	F1-Score (WeightedAverage)
Logistic Regression	0.87	0.87	0.87	0.87
KNN	0.93	0.93	0.93	0.93
Naive Bayes - GaussianNB	0.86	0.86	0.86	0.86
Naive Bayes - BernoulliNB	0.78	0.78	0.78	0.78
Naive Bayes - MultinomialNB	0.82	0.82	0.82	0.82
Random Forest	**0.96**	0.96	0.96	0.96

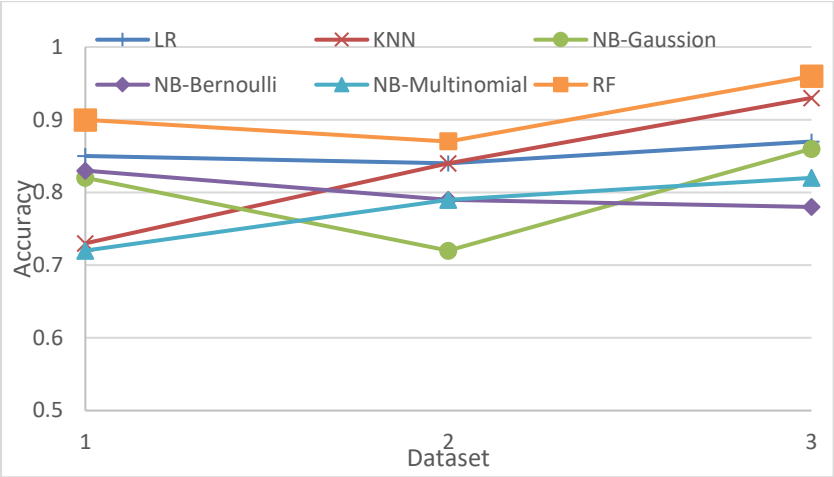

Fig. 1. ML models performance comparison over all three datasets.

5 Discussion

According to the results above, we delved into two primary dimensions concerning the interplay among machine learning methodologies (LR, KNN, NB, and RF) and dataset sizes. One is the performance stability analysis of each specific learning algorithm on varying datasets sizes. And the other is the comparison analysis of different algorithms' performance on large or small datasets.

From Figure 1, we can conclude the performance stability of different learning algorithms on each dataset. We find that LR is the least sensitive algorithm to datasets change, while the performance of KNN fluctuates a lot on varied sizes of datasets. Accuracy values of LR for the three different experiment datasets are all above 0.84, indicating a satisfactory result. In observational studies employing logistic regression for analysis, it is advised by [5] to consider a sample size which will contain a minimum of 500 samples. This recommendation aims to ensure that the derived statistics effectively capture the parameters within the targeted dataset. The smallest dataset we have is 690 samples. Therefore, the correct minimum size of dataset lays the foundation for LR effective prediction. Meanwhile, the performance of LR is stable, only differing within 3% on accuracy across three the datasets. This finding is aligned with [6], in which Researchers tried to find the effect of dataset size and interactions on prediction performance of logistic regression and deep learning models and found that regardless of the interaction order, increasing dataset size had negligible effects on the performance of ML models, especially the logistic regression model. On the flip side, the performance of KNN exhibits discernible fluctuations with variations in dataset size. The outcomes suggest a relatively positive

correlation, indicating that as the dataset size increases, KNN tends to deliver improved performance. Similar to the findings of Olsson et al. in their research [7], the accuracy of the K-nearest neighbor (KNN) algorithm is dependent on the selection of the 'k' parameter, particularly when working with limited available data. Furthermore, it should be noted that the optimal 'k' value tends to increase as the size of the training data grows.

Similar observations hold true for Naïve Bayes, demonstrating robust and consistent performance across the three selected datasets. In our experiments, we developed three Naïve Bayes model: Gaussian NB, Bernoulli NB, and Multinomial NB. The performance of these models are intricately linked to the type of features, effectively capturing the prevalent feature distribution across the entire dataset. The model with the highest accuracy is chosen to represent the ultimate evaluation performance of Naïve Bayes. Examining the performance details presented in the result section, it becomes apparent that the size of the dataset may not be a critical factor influencing Naïve Bayes performance. Instead, the performance is more reliant on the attributes of the features. For instance, in dataset 1, the performance of the constructed Multinomial NB model is slightly inferior to the other two models. This implies that the variable distribution in dataset 1 aligns more closely with a Gaussian or Bernoulli distribution. In such cases, if the NB model is constructed based on the assumption that features follow a multinomial distribution, the performance will be notably degraded.

In addition, Random Forest consistently achieved remarkable and satisfactory performance with an accuracy value above 0.87. Notably, Random Forest demonstrates its best performance on the largest dataset, dataset 3, with accuracy surging to 0.96. It further proves the capability of Random Forest to effectively handle large and complex datasets, leveraging its ability to learn intricate details as the dataset size increases. Moreover, Random Forest exhibits commendable robustness in dealing with outliers and displays a high level of generalization across different types of features. Nevertheless, one notable drawback of Random Forest is its high complexity. The procedure of hyperparameters tunning to construct an optimal model that best fits the dataset can be time-consuming, rendering the computational cost relatively high.

In general, all learning algorithms, except KNN, exhibit satisfactory performance in the smallest dataset (Dataset 1). In the case of the medium-sized dataset (Dataset 2), there is no notable difference in the performance of any algorithm. However, when dealing with the largest dataset (Dataset 3), stability is upheld by Logistic Regression (LR), while the performance of all other algorithms displays improvement. Notably, significant enhancements are observed, particularly for KNN and Random Forest (RF).

6 Conclusion

Our comprehensive study offers pivotal insights into the performance dynamics of various machine learning algorithms across datasets of differing sizes. We meticulously evaluated four widely-adopted algorithms—Logistic Regression, K-Nearest Neighbors, Naïve Bayes, and Random Forest—across three distinct datasets.

According to the experimental findings, it was observed that both logistic regression and naive Bayes classifiers demonstrated consistent performance across the three chosen datasets. This implies that variations in dataset size might not be the predominant factor influencing their performance. Instead, factors such as correlations between variables and class imbalance could exert a more substantial influence, contingent on their inherent properties and assumptions.

The KNN algorithm exhibited significant variability in performance contingent on dataset size. Our findings underscore that KNN's efficacy is largely dependent on the dataset's volume, with larger datasets generally enhancing its predictive accuracy. This suggests that KNN may be more suited for scenarios where sufficient data is available, and its parameter 'k' can be optimally tuned.

Random Forest emerged as a notably powerful algorithm, especially with larger datasets. Its ensemble approach, combining multiple decision trees, allows it to excel in complex scenarios involving substantial datasets. This characteristic makes RF an ideal candidate for applications where large-scale data is prevalent and where model robustness is critical.

The efficiency of a machine learning algorithm is not solely dependent on increasing dataset size. Our study suggests that, beyond a certain threshold, simple enlarging the dataset does not proportionally enhance performance. This finding emphasizes the importance of considering other factors such as feature selection, algorithm tuning, and computational efficiency in the data-algorithm interplay. As per the study by Mo, H. et al. [8], the performance of the algorithm can be notably influenced by adjusting hyperparameters based on the dataset size.

Our research contributes significantly to the understanding of machine learning algorithms' performance across various datasets. This emphasizes the significance of selecting an appropriate algorithm that aligns with the distinctive characteristics and scale of the dataset.

References

1. Khanam, J.J., Foo, S.Y.: A comparison of machine learning algorithms for diabetes prediction. ICT Express. 7, 432–439 (2021). https://doi.org/10.1016/j.icte.2021.02.004.
2. Dris, A.B., Alzakari, N., Kurdi, H.: A Systematic Approach to Identify an Appropriate Classifier for Limited-Sized Data Sets. (2019). https://doi.org/10.1109/isncc.2019.8909099.
3. Catal, C., Diri, B.: Investigating the effect of dataset size, metrics sets, and feature selection techniques on software fault prediction problem. Information Sciences. 179, 1040–1058 (2009). https://doi.org/10.1016/j.ins.2008.12.001.
4. Althnian, A., AlSaeed, D., Al-Baity, H., Samha, A., Dris, A.B., Alzakari, N., Abou Elwafa, A., Kurdi, H.: Impact of Dataset Size on Classification Performance: An Empirical Evaluation in the Medical Domain. Applied Sciences. 11, 796 (2021). https://doi.org/10.3390/app11020796.
5. Sample Size Guidelines for Logistic Regression from Observational Studies with Large Population: Emphasis on the Accuracy Between Statistics and Parameters Based on Real Life Clinical Data. MJMS. 25, 122–130 (2018). https://doi.org/10.21315/mjms2018.25.4.12.
6. Bailly, A., Blanc, C., Francis, É., Guillotin, T., Jamal, F., Wakim, B., Roy, P.: Effects of dataset size and interactions on the prediction performance of logistic regression and deep learning models. Computer Methods and Programs in Biomedicine. 213, 106504 (2022). https://doi.org/10.1016/j.cmpb.2021.106504.
7. Olsson, J.S.: An analysis of the coupling between training set and neighborhood sizes for the k NN classifier. (2006). https://doi.org/10.1145/1148170.1148317.
8. Hai, M., Zhang, Y., Zhang, Y.: A Performance Evaluation of Classification Algorithms for Big Data. Procedia Computer Science. 122, 1100–1107 (2017). https://doi.org/10.1016/j.procs.2017.11.479.
9. Austin, A.M., Ramkumar, N., Gladders, B., Barnes, J.A., Eid, M.A., Moore, K.O., Feinberg, M.W., Creager, M.A., Bonaca, M., Goodney, P.P.: Using a cohort study of diabetes and peripheral artery disease to compare logistic regression and machine learning via random forest modeling. BMC Med Res Methodol. 22, (2022). https://doi.org/10.1186/s12874-022-01774-8.
10. Saroj, R.K., Yadav, P.K., Singh, R., Chilyabanyama, O.N.: Machine Learning Algorithms For Understanding The Determinants of Under-Five Mortality. https://doi.org/10.21203/rs.3.rs-1021040/v1.
11. Hu, B.-L., Luo, Y.-W., Zhang, B., Zhang, G.-P.: A Comparative Investigation of Machine Learning Algorithms for Pore-Influenced Fatigue Life Prediction of Additively Manufactured Inconel 718 Based on a Small Dataset. Materials. 16, 6606 (2023). https://doi.org/10.3390/ma16196606.
12. Zhang, Y., Xin, Y., Li, Q., Ma, J., Li, S., Lv, X., Lv, W.: Empirical study of seven data mining algorithms on different characteristics of datasets for biomedical classification applications. BioMed Eng OnLine. 16, (2017). https://doi.org/10.1186/s12938-017-0416-x.
13. Pal, S., Chaurasia, V.: Machine learning algorithms using binary classification and multi model ensemble techniques for skin diseases prediction. IJBET. 34, 57 (2020). https://doi.org/10.1504/ijbet.2020.10032551.
14. Trishna, T.I., Emon, S.U., Ema, R.R., Sajal, G.I.H., Kundu, S., Islam, T.: Detection of Hepatitis (A, B, C and E) Viruses Based on Random Forest, K-nearest and Naïve Bayes Classifier. (2019). https://doi.org/10.1109/icccnt45670.2019.8944455.

15. Ren, J., Lee, S.D., Chen, X., Kao, B., Cheng, R., Cheung, D.: Naive Bayes Classification of Uncertain Data. (2009). https://doi.org/10.1109/icdm.2009.90.
16. Huang, Y., Li, L.: Naive Bayes classification algorithm based on small sample set. (2011). https://doi.org/10.1109/ccis.2011.6045027.
17. Parhusip, H.A., Susanto, B., Linawati, L., Trihandaru, S., Sardjono, Y., Mugirahayu, A.S.: Classification Breast Cancer Revisited with Machine Learning. Int. J. Data. Science. 1, 42–50 (2020). https://doi.org/10.18517/ijods.1.1.42-50.2020.
18. Prakash, K.B.: Analysis, Prediction and Evaluation of COVID-19 Datasets using Machine Learning Algorithms. IJETER. 8, 2199–2204 (2020). https://doi.org/10.30534/ijeter/2020/117852020.

Appendix A. Evaluation metrics math equations

Macro average: Each class contributes equally to the average, regardless of its size or the number of instances.

$$Precision = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i+FP_i} \quad (1)$$

$$Recall = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i+FN_i} \quad (2)$$

$$F1\ Score = 2(Precision*Recall)/(Precision+Recall) \quad (3)$$

Weighted average: The contributions of each class is weighted based on its size or prevalence in the dataset, and the weight assigned to each class is denoted as w_i in the equations provided below.

$$Precision = \sum_{i=1}^{N} w_i \cdot \frac{TP_i}{TP_i+FP_i} \quad (4)$$

$$Recall = \sum_{i=1}^{N} w_i \cdot \frac{TP_i}{TP_i+FN_i} \quad (5)$$

$$F1\ Score = \sum_{i=1}^{N} w_i \cdot 2 \cdot (Precision*Recall)/(Precision+Recall) \quad (6)$$