International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Efficient Machine Learning Model Training through Data Subsampling: Balancing Performance and Computational Cost

Vibhu Verma[1]

Submitted:11/03/2024 **Revised**: 26/04/2024 **Accepted**: 03/05/2024

Abstract: The increasing complexity of Machine Learning models raises the demanding needs for effective strategies that reduce training time without losing performance. The paper compares a few different approaches for generating subsamples of the data serving as a rich representation of the full dataset, enabling faster training while maintaining model accuracy. By leveraging ML techniques, this approach identifies and extracts representative subsets that preserve the most salient features of the original data. These are then used for training various models, reducing computational costs and time requirements. Experimental results show that across different ML tasks, the proposed approach yields significant regular reductions in training time while retaining comparable predictive performance. The method has the potential to improve the efficiency of large-scale ML workflows in a data-intensive environment.

Keywords*: Machine Learning, Data Subsampling, Training Efficiency, Rich Representations, Computational Optimization.*

1. Introduction

This paper will present various frameworks that could be employed in subsampling a data set, richly representative of the full set. This varies from random sampling to several sampling techniques using machine learning to retrieve a small-sized representation of the actual data.

This can be extremely useful in domains where we have a large corpus of data- Finance, Fraud Modelling, etc.- and training a model on the whole set is not the most cost-efficient way; it also involves high terms of time and money. In this paper, we talk about the comparison between the performance of the model that was trained using data sub-samples, and a model that was trained on the entire data. Besides this, we are comparing the cost in terms of time our algorithms used to come to convergence.

2. Key Concepts

2.1. Sampling Algorithms Outline

2.1.1. UMAP

Uniform Manifold Approximation and Projection is a new kind of dimensionality reduction technique for visualizing data in lower dimensional spaces such as 2D or 3D. It has been highly valuable for maintaining the local structure of the data, making it especially useful when investigating clusters and other interesting features. It has now seen widespread adoption for bioinformatics, natural language processing, and image analysis to show underlying relationships in complicated data sets.

2.1.2. HDBSCAN

HDBSCAN is a clustering algorithm that groups data points into clusters depending on changing density levels. Unlike DBSCAN, it can automatically learn the number of clusters and deal with noise quite well. The ideal cases to use the HDBSCAN algorithm are when working with data with clusters of varying densities; the algorithm requires limited tuning of parameters, hence considered quite popular in exploratory data analysis.

2.1.3. K-MODES

K-MODES is a clustering algorithm designed for categorical data. The method works by assigning data points to the appropriate cluster by taking the mode of their categorical features to minimize dissimilarity within the clusters. It contrasts with algorithms developed for numerical data in that it operates directly upon categorical variables without any transformation. Examples of applications include customer segmentation, text categorization, and demographic analysis.

2.1.4. K-MEANS

K-Means is one of the unsupervised learning algorithms that operate on numerical data to partition into a certain number (K) of clusters. It always minimizes the within-cluster variance, including a step that iteratively updates cluster centroids and reassigns data points. It is fast, efficient, and assumes spherical-shaped and equally sized clusters. It can also be used in market segmentation, image compression, and other pattern recognition tasks.

2.1.5. Gower Distance

This is a similarity measure that was created for mixed data types, including numerical, categorical, and ordinal variables. It explicitly calculates the similarity of every feature and then combines them into an overall distance

[1] *Principal Data Scientist, GWU, Capital One, NY, USA*
ORCID ID : 0009-0002-6232-8688

measure. This technique scales all features to the range of 0 to 1. Therefore, it is usually used in clustering analysis or similar analyses where mixed datasets are dealt with, such as customer profiling or healthcare studies.

2.1.6. ISOLATION FOREST

The Isolation Forest is a method for anomaly detection that isolates anomalies using random splits in the data. Anomalies are more easily separated than normal data points and, therefore, require fewer splits. It is efficient and scalable for big datasets, working well in high-dimensional spaces, hence ideal for fraud detection, cybersecurity, and rare-event analysis.

2.1.7. KS STATISTIC

The Kolmogorov-Smirnov statistic provides the maximum possible deviation between the CDFs of two datasets or distributions. This is applied to establish whether the two distributions differ significantly. The KS statistic consists of a number varying between 0 and 1, where higher values denote a greater divergence.

2.2. Modeling Algorithms Outline

2.2.1. XGBoost

XGBoost (Extreme Gradient Boosting) is a high-performance machine learning algorithm for regression and classification tasks. It builds decision trees sequentially, optimizing them to correct errors in previous iterations. Known for its speed, scalability, and ability to handle missing data, XGBoost is widely used in competitions and real-world applications for its accuracy and regularization to prevent overfitting.

2.2.2. Random Forest

Random Forest is a widely used ensemble learning technique that grows multiple decision trees from random subsets of the data and outputs their combination. It considerably reduces overfitting, makes the model more robust, and works well for regression and classification problems. Random Forest is popular for interpretability, robustness to noise, and good performance in many cases.

2.2.3. Gradient Boosting Machines

GBM is an ensemble technique where models are sequentially built, each of which corrects errors committed by the previous ones. Typically, it combines weak learners, often decision trees, into one strong predictive model. GBM allows much flexibility and support for the optimization of various user-defined loss functions, but it may have problems with overfitting if proper regularization is not performed. GBM is widely used in predictive modeling tasks as it produces accurate results.

2.2.4. Linear Regression

Linear regression is a method in statistics that relates a dependent variable to one or more independent variables. It assumes linearity and works in the direction of least squares to result in the best-fitted line. Linear regression is easy to interpret, and thus, it is extensively used in predictive analytics, economics, and biology for the estimation of continuous outcome variables.

2.2.5. Logistic Regression

Logistic regression is a classification algorithm that predicts binary outputs given some input features. It models the probability of an event taking place using the logistic function, which generates outputs between 0 and 1. Logistic regression is simple, interpretable, and powerful for applications such as spam detection, medical diagnosis, and customer churn. It can also be generalized to multiclass problems.

2.3. Sampling Framework

2.3.1. Random Sampling

Just to set a baseline for our results, we are using a simple random sampling class in Python, which is designed to randomly select a sample percent of rose from a given dataset. We also make sure that we are selecting samples without replacement.

2.3.2. UMAP and Grid Search

This technique aims to create a meaningful yet diverse subset of high-dimensional data for analysis or machine learning. Combining UMAP for dimensionality reduction with stratified sampling in the reduced space, ensures the sampled subset accurately reflects the overall distribution and structure of the original data. UMAP excels here by compressing complex data into 2D or 3D spaces while preserving relationships between data points. Its ability to handle both categorical and continuous data enhances flexibility and applicability across various datasets. Unlike techniques like PCA, which rely on assumptions such as linear relationships between features, UMAP avoids these limitations, making it a robust alternative for dimensionality reduction. Stratified sampling operates by overlaying a grid on the reduced representation and selecting a percentage of points from each grid section, ensuring that all regions of the data are proportionally represented in the sample.

2.3.3. Cluster based Sampling

The Cluster-Based Sampling technique designs a representative subset selection based on clustering algorithms, particularly HDBSCAN for identifying meaningful groups within the dataset. It samples data from a fixed percentage of points in each cluster, including noise points taken as an independent cluster, to ensure that the structural diversity of the original dataset is preserved. It does this by adapting to datasets with mixed data types and computing distances differently, using for instance Gower distance for categorical features. This makes the method

suitable for a wide range of datasets. Especially powerful, HDBSCAN detects clusters of varying densities and handles noise effectively. Sampling within each cluster by its size ensures that small clusters are not smothered by larger ones; this preserves rare but meaningful patterns. This is useful in tasks such as balanced model training, and data reduction with the retention of key insights.

2.3.4. Anomaly Sampler

The Anomaly Sampler is an algorithm to extract a representative subset of data items, utilizing anomaly detection scores from an Isolation Forest. The primary goal of the sampler is to maintain the distributional integrity of the original dataset while retaining a subset with a very similar anomaly score profile. The Isolation Forest assigns anomaly scores to each data point, which indicate how "normal" or "anomalous" they are concerning the data set. The sampler draws subsets in iterations, starting at 5% of the dataset and growing by increments of 5%. It uses a statistical test KS test-to compare the distribution of anomaly scores in the subset to that in the original data to ensure that the sample reflects variability and patterns in the original data. This method serves a specific purpose, especially when working with imbalanced datasets or when the understanding of anomalies is very crucial. It does retain distributional similarities that support tasks such as model training or validation on representative smaller datasets.

2.3.5. Entropy Sampler

The Entropy Sampler is a method that samples data points based on entropy, which is a measure of uncertainty or disorder. The whole idea behind this method is to sample the most "informative" points from a dataset, based on the highest values of entropy. These are the data points that contribute the most uncertainty or complexity to the overall distribution and hence are of value in tasks like active learning or model validation. This method works by first calculating the entropy of each feature, both for continuous and categorical data. For continuous features, it first bins data and then calculates entropy over the binned data; for categorical features, it calculates entropy over unique values and their frequencies. The individual scores are aggregated for each data point to create an overall entropy score. The process of sampling would select points with the highest entropy scores; these would be considered the most "uncertain," or diverse, and thus ensure that the sampled subset represents the most complex and varied aspects of the dataset. This is useful to make the sample retain most of the informational value of the original, which, in many occasions, is important to capture the diversity.

2.3.6. Distance based Sampling

The Distance Based Sampler a focus on the creation of a representative subset of data by making use of distances to cluster centroids. The underlying idea is a sampling of data points that are most distant from cluster centers, with the assurance that these sampled points capture edge cases, anomalies, or diverse patterns that may get lost otherwise. It then clusters the dataset using the K-Means algorithm on numerical data or K-Prototypes when the data is mixed (numerical and categorical). It determines distances from each point to the nearest cluster centroid, thus finding data points that fall on the boundary of a cluster or where there is low density. This will be very useful in situations where the model needs to consider diversity or edge cases, for instance, when testing machine learning models or when one does exploratory data analysis. Distance-based sampling ensures that the subset captures unique and potentially influential points, adding depth to the analysis and applicability of the dataset.

3. Dataset Details

This collection features various datasets intended for classification, regression, and other machine learning applications. The Tuandromd dataset with 241 features and 4,464 samples was mainly intended for a classification task with high-dimensional data. The All Claims dataset supports regression tasks with 130 features and 188,318 samples, perfectly suitable to predict continuous variables. The Android Permissions dataset, intended for classifying app permission data to assess risks, includes 86 features and 29,332 samples. The Student Success dataset has 36 features and 3,630 samples to classify academic outcomes. The Phishing URL dataset consists of 51 features and 235,795 samples to help detect phishing threats, while Parkinson's-19 features and 5,875 samples-and Credit Card-29 features and 284,807 samples-support regression and fraud detection classification, respectively.

Table. 1. Dataset Details

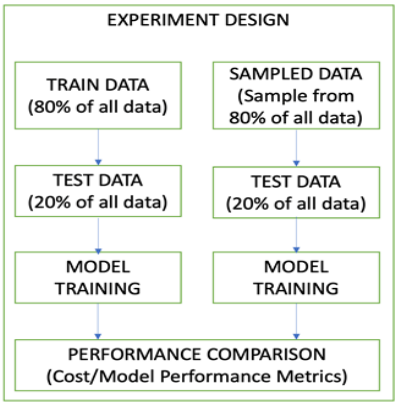
Type	Dataset Details		
	Features	***Dataset Size***	***Dataset Name***
Classification	241	4464	Tuandromd
Regression	130	188318	All Claims
Classification	86	29332	Android Permissions
Classification	36	3630	Student Success
Classification	51	235795	Phishing URL
Regression	19	5875	Parkinsons
Classification	29	284807	Credit Card

4. Experiment Design

This experiment will benchmark the performance of different machine learning models in terms of various data preparation strategies. In this regard, two approaches can be compared: the full data approach and the sampled data

approach.

Fig. 1. Experiment Design



In the full data approach, the data is split into an 80% training set and a 20% testing set. Then, a machine learning model will be trained on the whole training set, and its performance evaluated on the test set.

The sampled data approach randomly samples a subset from the 80% training set. A different model is then trained on the smaller dataset and its performance is also tested on the same 20% test dataset.

By comparing the two models' performance metrics, we want to decide upon the trade-off between the model accuracy and computational efficiency. Some of the key questions to be answered are:

- What is the exact sacrifice of performance due to the smaller, sampled dataset?
- What is the cost saving in computational terms with the sampled data approach?
- Is there an optimal sample size that provides a good balance between performance and efficiency?

These insights will help to form future data preparation and model training strategies, especially in situations in which computation resources are limited or time is tight.

In addition to this, all techniques focus on selecting 20% of the data out of the total training set but the Isolation Forest and KS Sampling Technique, it operates on working on finding a sub sample that is comparable to the actual dataset.

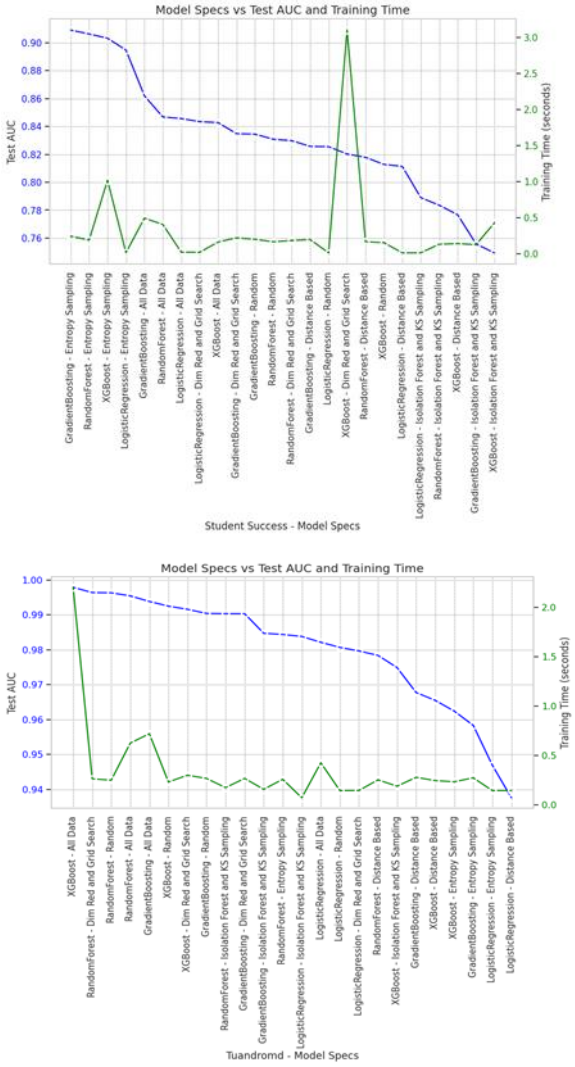


Fig. 2. Model Performance vs Time

When the size of the dataset is small, which is often the case, model performance has less dependence on the selection of a sampling technique. The models that are using the whole data, or sometimes applying dimensionality reduction coupled with grid search, outperform those models that use other sampling techniques. The methods that balance these, such as isolation forests with KS-sampling, might lead to a slight reduction in predictive accuracy. However, these methods enjoy considerable superior speed and are hence desirable when speed is the constraint. Distance-based sampling methods, which select samples based on the proximity between data points, consistently underperform with smaller datasets. These methods often fail to capture the most important features of the data, resulting in longer training times and reduced accuracy.

Isolation forest with KS sampling is faster but tends to result in lower accuracy as its approach focuses on the selection of subsets of data judged to be more relevant by identifying outliers. While these methods speed up training time by working on smaller subsets of data, critical patterns might be missed, which impacts the generalizing capability of the model. This is reflected in lower AUC scores of models

trained with these methods. Thus, though these approaches are faster, they are not that reliable in returning high accuracy.

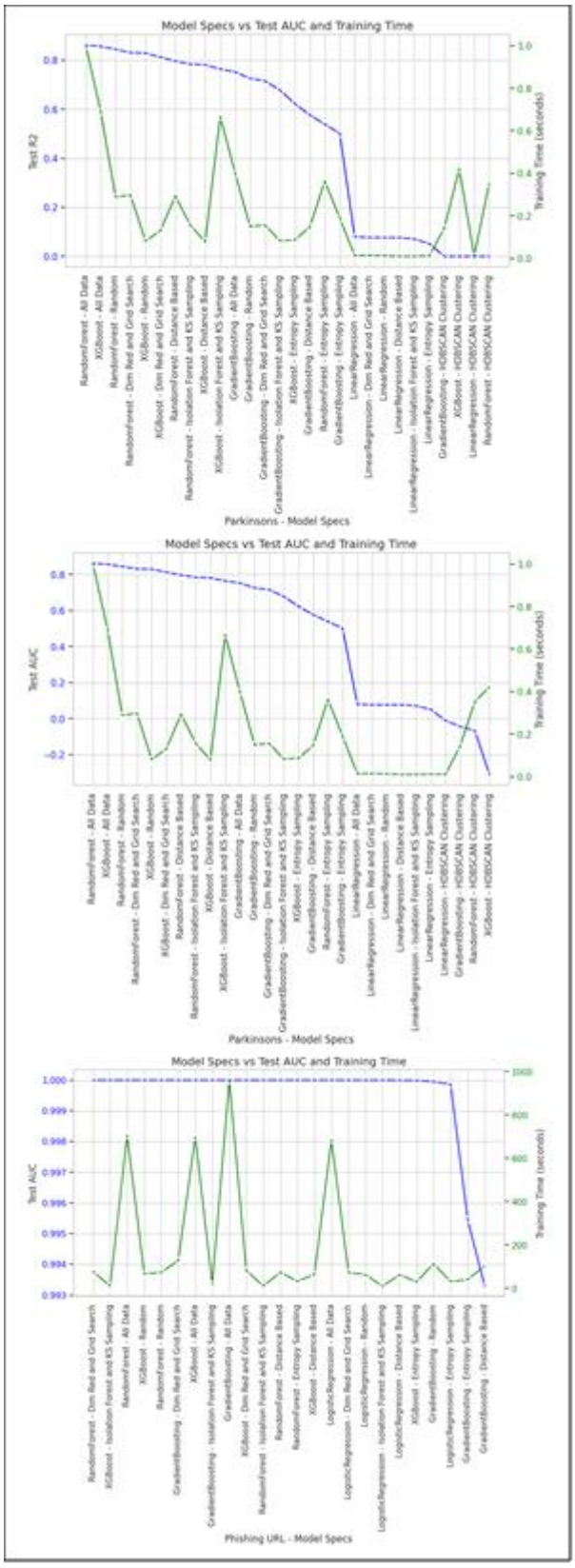


Fig. 3. Model Performance vs Time

Model Performance vs Time

With the increase in dataset size, the impact of sampling techniques becomes more pronounced. For larger datasets, say those from credit card fraud loss detection or phishing URL classification problems, it gets prohibitively time-consuming to train a model on such larger datasets. Sampling methods have come in handy to minimize the dataset size, hence significantly reducing computational times by a margin of up to 60%. Although such methods reduce the data, models trained with these techniques can often perform similarly to those trained with full datasets, provided that main features are kept in the sampling.

For large datasets, dimensionality reduction combined with grid search remains the most effective approach. This combination ensures that the model focuses on the most important features, reducing overfitting and improving predictive accuracy. By reducing the dataset size while preserving its critical features, this method mitigates the computational burden of training on large datasets.

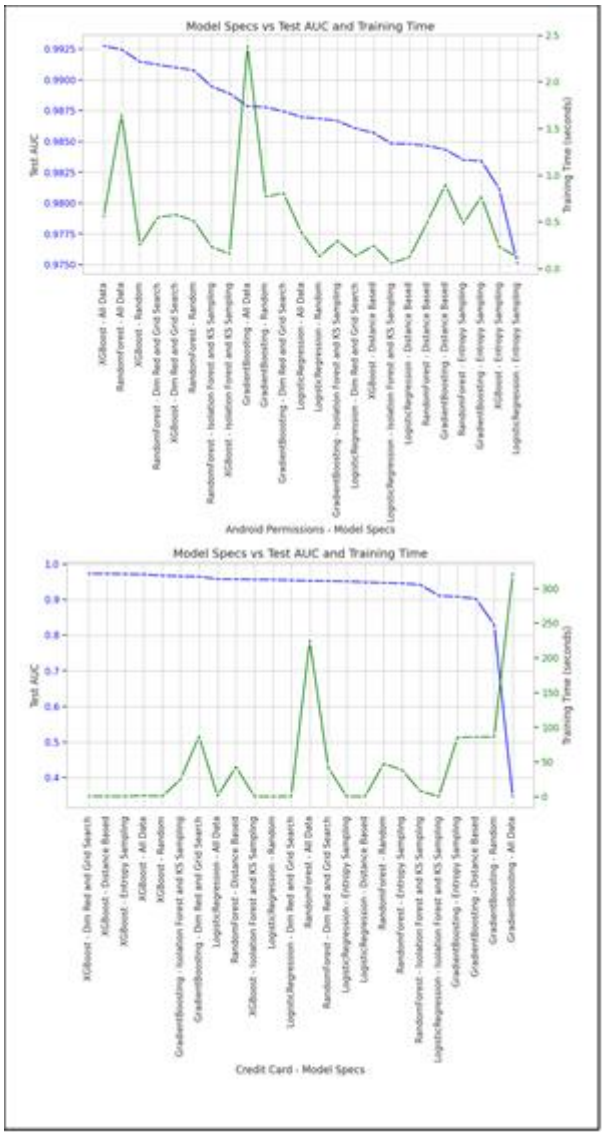


Fig. 4. Model Performance vs Time

However, distance-based sampling still remains inefficient for larger datasets. Models trained on samples selected based on proximity to other samples usually fail to learn the

most important patterns in data and therefore exhibit lower accuracy. Also, the longer training times of distance-based methods become more problematic as the size of the dataset increases.

Model Performance vs Time

It comes with great computational resource cost and time that depend on the size of the dataset. In cases of big datasets, dimensionality reduction combined with grid search will help to reduce dataset size without losing their key features. Further, it will not only cut training time but also provide fast iteration through different models and hyperparameter settings, thus allowing the selection of the best model without excessive computational cost.

Model Performance vs Time

Conclusively, the effectiveness of the different sampling techniques depends on dataset size, available computational resources, and modeling objectives. For small datasets, the use of a full dataset or the application of dimensionality reduction with grid search is usually the most viable, which ensures high accuracy and prevents overfitting. As datasets increase in size, sampling techniques become necessary. Dimensionality reduction combined with grid search remains the most balanced approach to optimize accuracy and computational efficiency. While faster methods do exist, such as isolation forest with KS sampling, these usually come at the expense of lower accuracy and are really best utilized in situations where speed is more important than precision. In the end, the sampling technique used should be determined by dataset size and goals of modeling, and reduction plus grid search can work quite effectively for a wide range of situations.

References

[1] L. McInnes, J. Healy, and N. Saul, "UMAP: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, Feb. 2018.

[2] R. J. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining (PAKDD)*, Ho Chi Minh City, Vietnam, 2013, pp. 160–172, doi: 10.1007/978-3-319-18038-0_14.

[3] Z. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values," *Data Mining Knowl. Discov.*, vol. 2, no. 3, pp. 283–304, Sep. 1998, doi: 10.1023/A:1009769707641.

[4] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Prob.*, Berkeley, CA, USA, 1967, pp. 281–297.

[5] J. C. Gower, "A general coefficient of similarity and some of its properties," *Biometrics*, vol. 27, no. 4, pp. 857–871, Dec. 1971, doi: 10.2307/2528823.

[6] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Pisa, Italy, 2008, pp. 413–422, doi: 10.1109/ICDM.2008.17.

[7] F. J. Massey Jr., "The Kolmogorov-Smirnov test for goodness of fit," *J. Amer. Statist. Assoc.*, vol. 46, no. 253, pp. 68–78, Mar. 1951, doi: 10.1080/01621459.1951.10500769.

[8] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, San Francisco, CA, USA, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[9] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[10] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001, doi: 10.1214/aos/1013203451.

[11] G. A. F. Seber and A. J. Lee, *Linear Regression Analysis*, 2nd ed. Hoboken, NJ, USA: Wiley, 2012.

[12] D. R. Cox, "The regression analysis of binary sequences," *J. Roy. Statist. Soc. Ser. B.*, vol. 20, no. 2, pp. 215–242, 1958.

[13] W. G. Cochran, *Sampling Techniques*, 3rd ed. New York, NY, USA: Wiley, 1977.

[14] J. Zhao, W. Cai, and M. Wang, "Sampling representative data points for clustering based on density and distance," *IEEE Access*, vol. 6, pp. 51977–51987, 2018, doi: 10.1109/ACCESS.2018.2869220.